ALIBABA CLOUD

Alibaba Cloud

ApsaraDB for RDS

RDS MariaDB TX Database

Document Version: 20220622

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 because of product version upgrade, adjustment, or other reasons. Alibaba Cloud reserves the right to modify the content of this document without notice and an updated version of this document will be released through Alibaba Cloud-authorized channels from time to time. You should pay attention to the version changes of this document as they occur and download and obtain the most up-to-date version of this document from Alibaba Cloud-authorized channels.

4. This document serves only as a reference guide for your use of Alibaba Cloud products and services. Alibaba Cloud provides this document based on the "status quo", "being defective", and "existing functions" of its products and services. Alibaba Cloud makes every effort to provide relevant operational guidance based on existing technologies. However, Alibaba Cloud hereby makes a clear statement that it in no way guarantees the accuracy, integrity, applicability, and reliability of the content of this document, either explicitly or implicitly. Alibaba Cloud shall not take legal responsibility for any errors or lost profits incurred by any organization, company, or individual arising from download, use, or trust in this document. Alibaba Cloud shall not, under any circumstances, take responsibility for any indirect, consequential, punitive, contingent, special, or punitive damages, including lost profits arising from the use or trust in this document (even if Alibaba Cloud has been notified of the possibility of such a loss).

5. By law, all the contents in Alibaba Cloud documents, including but not limited to pictures, architecture design, page layout, and text description, are intellectual property of Alibaba Cloud and/or its affiliates. This intellectual property includes, but is not limited to, trademark rights, patent rights, copyrights, and trade secrets. No part of this document shall be used, modified, reproduced, publicly transmitted, changed, disseminated, distributed, or published without the prior written consent of Alibaba Cloud and/or its affiliates. The names owned by Alibaba Cloud shall not be used, published, or reproduced for marketing, advertising, promotion, or other purposes without the prior written consent of Alibaba Cloud. The names owned by Alibaba Cloud include, but are not limited to, "Alibaba Cloud", "Aliyun", "HiChina", and other brands of Alibaba Cloud and/or its affiliates, which appear separately or in combination, as well as the auxiliary signs and patterns of the preceding brands, or anything similar to the company names, trade names, trademarks, product or service names, domain names, patterns, logos, marks, signs, or special descriptions that third parties identify as Alibaba Cloud and/or its affiliates.

6. Please directly contact Alibaba Cloud for any errors of this document.

Document conventions

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

Table of Contents

1.Preface

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

Overview

ApsaraDB for RDS offers stable, reliable, and scalable cloud database services. Based on Apsara Distributed File System and high-performance storage (SSD), ApsaraDB for RDS supports the following database engines: MySQL, SQL Server, PostgreSQL. ApsaraDB for RDS also provides solutions for disaster recovery, backup, database restoration, monitoring, and migration to simplify the database operations and maintenance. For more information about the benefits of ApsaraDB for RDS, see Benefits.

This document describes how to configure ApsaraDB for RDS through the ApsaraDB for RDS console to help you know more about its features and functions. You can also manage ApsaraDB for RDS through APIs and SDKs.

For further assistance, you can log on to the ApsaraDB for RDS console, click More in the top navigation bar, and choose **Support > Open a new ticket**. If your business is complex, you can purchase a support plan to obtain support from IM enterprise groups, technical account managers (TAMs), and service managers.

For more information about ApsaraDB for RDS, see Product Details.

Disclaimer

Some product features or services described in this document may be unavailable in certain regions. See the actual commercial contracts for specific Terms and Conditions. This document serves as a reference guide for your use of Alibaba Cloud products and services. Alibaba Cloud makes every effort to provide relevant operational guidance based on existing technologies. However, Alibaba Cloud hereby states that it in no way guarantees the accuracy, integrity, applicability, and reliability of the content of this document, either explicitly or implicitly.

Terms

- Instance: A database service process that takes up physical memory independently. You can set different memory size, disk space, and database type, where the memory size determines the performance of the instance. After the instance is created, you can change the configuration or delete the instance at any time.
- Database: A database is a logical unit created in an instance. The name of each database under the same instance must be unique.
- Region and zone: Each region is a separate geographic area. Each region has many isolated locations known as zones. The power supply and network of each zone are independent. For more information, see Alibaba Cloud Global Infrastructure.

General terms

Term	Description
On-premise database	Refers to the database deployed in the local server room or the database not on the ApsaraDB for RDS.

Term	Description
ApsaraDB RDS XX (XX represents one of the following database engines: MySQL, SQL Server, PostgreSQL.)	Indicates the ApsaraDB for RDS of a specific database engine. For example, ApsaraDB RDS MySQL means the database engine of the instance enabled on the RDS is MySQL.

2.Limits of RDS MariaDB

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

The following table describes the limits of common actions and configurations in RDS MariaDB.

Item	Limit description
Parameter modification	The RDS console or supported API actions can be used to modify database parameters. But For security parameters, some parameters cannot be modified. For more information, see Use the console to set parameters.
Database root permission	The root and sa permissions are not provided.
Database backup	• Supported CLIs or GUIs can be used for logical data backup. • For physical data backup, the RDS console or supported API actions must be used.
Data restoration	• Supported CLIs or GUIs can be used for logical data restoration. • For physical data restoration, the RDS console or supported API actions must be used.
MariaDB storage engine	• Currently only InnoDB is supported. • For performance and security purposes, we recommend that you use InnoDB. • Memory is not supported. If you create Memory engine tables, they are automatically converted to InnoDB engine tables.
Database replication	MySQL provides a dual-node cluster based on the master/slave replication architecture. The slave instance in the architecture is invisible to you, and your application cannot access to the slave instance directly.
Instance restart	Instances must be restarted through the RDS console or supported API actions.

3.Features of ApsaraDB RDS for MariaDB TX

This topic provides an overview of the features that are supported by ApsaraDB RDS for MariaDB TX. In the following table,the check sign (✔) indicates that the specified feature is supported, and the cross sign (✘) indicates that the specified feature is not supported.

Category	Feature	MariaDB TX 10.3 RDS High-availability Edition Standard SSD/Enhanced SSD
Data migration	Migrate data from a self-managed database to an RDS instance by using mysqldump	✔
Instance management	Create an RDS instance	✔
	Change the configuration of an RDS MariaDB instance	✔
	Switch over workloads between primary and secondary ApsaraDB RDS for MariaDB TX instances	✔
	Restart an ApsaraDB RDS for MySQL instance	✔
	Set the maintenance window of an ApsaraDB RDS instance	✔
	Release an RDS MariaDB instance	✔
	Manage ApsaraDB RDS for MySQL instances that are in the recycle bin	✔
Account management	Create an account on an ApsaraDB RDS for MariaDB instance	✔
	Reset the password of an account on an ApsaraDB RDS for MariaDB TX instance	✔
	Modify the permissions of a standard account on an ApsaraDB RDS for MariaDB TX instance	✔
	Delete an account for an RDS MariaDB instance	✔

Category	Feature	MariaDB TX 10.3 RDS High-availability Edition Standard SSD/Enhanced SSD
Database management	Create a database on an ApsaraDB RDS for MariaDB TX instance	✔
	Delete a database from an ApsaraDB RDS for MariaDB TX instance	✔
Database connection	Connect to an ApsaraDB RDS for MariaDB TX instance	✔
	Configure endpoints	✔
	View and change the internal and public endpoints and ports of an RDS instance	✔
	Apply for or release a public endpoint for an ApsaraDB RDS for MariaDB TX instance	✔
Monitoring and alerting	View the resource and engine metrics of an RDS instance	✔
	Configure alert rules for an ApsaraDB RDS for MariaDB TX instance	✔
Security management	Configure a whitelist for an ApsaraDB RDS for MariaDB instance	✔
Audit	Manage the logs of an RDS instance	✔
Database backup	Perform automatic backup on an RDS instance	✔
	Free quota for backup storage of an RDS instance	✔
	Download the log backup files of an ApsaraDB RDS for MariaDB TX instance	□
Database restoration	Restore the data of an RDS instance	✔
	Create tags	✔

Category	Feature	MariaDB TX 10.3	RDS High-availability Edition	Standard SSD/Enhanced SSD
Tag management				
	Unbind tags from an ApsaraDB RDS for MySQL instance	✔	✔	✔
	Use tags to filter ApsaraDB RDS for MySQL instances	✔	✔	✔

4.Product Specifications

4.1. Primary ApsaraDB RDS for MariaDB TX instance types

This topic provides an overview of primary ApsaraDB RDS for MariaDB TX instance types, which include the most recent and earlier instance types. The overview includes the specifications for each instance type.

5.Quick start

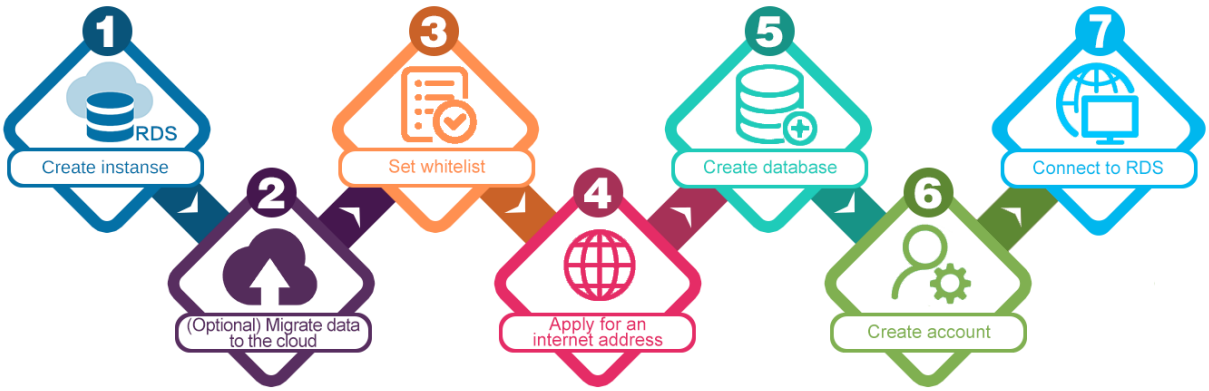
5.1. General workflow to use ApsaraDB RDS for MariaDB TX

This topic describes how to create and use an ApsaraDB RDS for MariaDB TX instance.

Quick start flowchart

If this is the first time that you use ApsaraDB RDS for MariaDB TX, we recommend that you familiarize yourself with the limits of ApsaraDB RDS for MariaDB TX. For more information, see Limits of ApsaraDB RDS for MariaDB TX.

The following flowchart shows the operations that you must perform before you use an ApsaraDB RDS for MariaDB TX instance.



1. Create an ApsaraDB RDS for MariaDB TX instance
2. Configure an IP address whitelist or security group for an ApsaraDB RDS for MariaDB TX instance
3. Apply for or release a public endpoint for an ApsaraDB RDS for MariaDB TX instance
4. Create a database and account on an ApsaraDB RDS for MariaDB instance
5. Connect to an ApsaraDB RDS for MariaDB TX instance

5.2. Create an ApsaraDB RDS for MariaDB TX instance

This topic describes how to create an ApsaraDB RDS for MariaDB TX instance in the ApsaraDB RDS console. You can also create an ApsaraDB RDS for MariaDB TX instance by calling an API operation.

Prerequisites

You have an Alibaba Cloud account. For more information, see Sign up with Alibaba Cloud.

Procedure

1. Go to the ApsaraDB RDS buy page.

2. Select a billing method.
   - **Subscription**: A subscription instance is an instance for which you pay an upfront fee. For long-term use, we recommend that you select the **Subscription** billing method. If you select the subscription billing method, you must also specify the **Duration** parameter in the lower section of the page. The subscription billing method is more cost-effective than the pay-as-you-go billing method. You are offered lower prices for longer subscription periods.
   - **Pay-As-You-Go**: A pay-as-you-go instance is charged per hour based on your actual resource usage. For short-term use, we recommend that you select the **Pay-As-You-Go** billing method. If you no longer need a pay-as-you-go RDS instance, you can release the instance to reduce costs.

   > **Note**
   > - You can create a **pay-as-you-go** RDS instance. After you confirm that the RDS instance that you created meets your business requirements, you can change the billing method of the RDS instance to **subscription**.
   > - If you want to manage the host on which your RDS instance is deployed, you must select Dedicated Cluster (Subscription) to create a host. Then, you can create an RDS instance on the host.

3. Configure the following parameters.

Parameter	Description
Region	The region where the RDS instance resides. - If your application is deployed on an Elastic Compute Service (ECS) instance, the RDS instance must reside in the same region as the ECS instance. For example, the RDS instance and the ECS instance can both reside in the China (Hangzhou) region. If the RDS instance and the ECS instance reside in different regions, they cannot communicate over an internal network and therefore they cannot deliver optimal performance. - If your application is deployed on an on-premises server or computer, we recommend that you select a region that is in close proximity to the on-premises server or computer.
Database Engine	The database engine and version that are run by the RDS instance. Select the **MariaDB TX** database engine. Only MariaDB 10.3 is supported. **Note** The available database engines and versions vary based on the region that you select.
Edition	The RDS edition of the RDS instance. Select **High-availability**. In RDS High-availability Edition, the database system consists of a primary RDS instance and a secondary RDS instance, which work in a high-availability architecture. **Note** The available RDS editions vary based on the region and database engine version that you select. For more information, see Overview of ApsaraDB RDS editions.

Parameter	Description
Storage Type	The type of storage medium that is used by the instance. ApsaraDB RDS for MariaDB TX supports **enhanced SSDs (ESSDs)**, which come in three performance levels (PLs). ○ ESSD PL1: This is the basic PL of ESSDs. ○ ESSD PL2: An ESSD of PL2 delivers IOPS and throughput that are approximately twice higher than the IOPS and throughput delivered by an ESSD of PL1. ○ ESSD PL3: An ESSD of PL3 delivers IOPS that is up to 20 times higher than the IOPS delivered by an ESSD of PL1. An ESSD of PL3 also delivers throughput that is up to 11 times higher than the throughput delivered by an ESSD of PL1. ESSDs of PL3 are suitable for business scenarios in which highly concurrent requests must be processed with high I/O performance and at low read and write latencies. For more information, see Storage types.
Zone	The zone where the RDS instance resides. Each zone is an independent physical location within a region. For example, the China (Hangzhou) region contains Zone H, Zone I, and Zone J. ApsaraDB RDS supports the following two deployment methods: ○ **Multi-zone Deployment**: The primary RDS instance and the secondary RDS instance reside in different zones to provide zone-disaster recovery. This is the recommended deployment method. ○ **Single-zone Deployment**: The primary RDS instance and the secondary RDS instance reside in the same zone. **Note** If you select the RDS Basic Edition, you can select only the **Single-zone Deployment** method.
Instance Type	The instance type of the RDS instance. Before you select an instance type, you must select an instance family. ○ **General-purpose (Entry-level)**: A general-purpose instance exclusively occupies the allocated memory and I/O resources. However, it shares CPU and storage resources with the other general-purpose instances that are deployed on the same host. ○ **Dedicated (Enterprise-level)**: A dedicated instance exclusively occupies the allocated CPU, memory, storage, and I/O resources. The dedicated host instance family is the highest configuration of the dedicated instance family. A dedicated host instance occupies all the CPU, memory, storage, and I/O resources on the host where the instance is deployed. **Note** For more information, see Primary ApsaraDB RDS instance types.

Parameter	Description
Capacity	The maximum amount of storage that is provisioned to store data files, system files, binary log files, and transaction files in the RDS instance. You can adjust the storage capacity at a step size of 5 GB. **Note** A dedicated RDS instance that uses local SSDs exclusively occupies the allocated resources, and its storage capacity varies based on the instance type. For more information, see Primary ApsaraDB RDS instance types.

4. In the lower-right corner of the page, click **Next: Instance Configuration**.
5. Configure the following parameters.

Parameter	Description
Network Type	The network type of the RDS instance. Select **VPC**. A virtual private cloud (VPC) is an isolated virtual network that provides higher security and higher performance than the classic network. If you select the VPC network type, you must specify the **VPC** and **VSwitch of Primary Node** parameters. If you set the **Deployment Method** parameter in the previous step to **Multi-zone deployment**, you must also specify the **VSwitch of Secondary Node** parameter. **Note** The network type of the RDS instance must be the same as the network type of the Elastic Compute Service (ECS) instance that you want to connect. If the RDS instance and the ECS instance reside in VPCs, both instances must reside in the same VPC. If the RDS instance and the ECS instance reside in different VPCs, these instances cannot communicate over an internal network.
	Enable or disable the release protection feature for an ApsaraDB RDS for MySQL instance
Resource Group	The resource group to which the RDS instance belongs. You can retain the default resource group or select a custom resource group based on your business requirements.

6. In the lower-right corner of the page, click **Next: Confirm Order**.
7. Confirm the configuration of the RDS instance in the Parameters section, specify the **Purchase Plan** and **Duration** parameters, read and select **Terms of Service**, and then click **Pay Now**. You need to specify the Duration parameter only when you select the subscription billing method for the RDS instance.

   Note If you select the subscription billing method for the RDS instance, we recommend that you select **Auto-Renew Enabled**. This prevents interruptions to your workloads even if you forget to review the RDS instance.

8. View the RDS instance.

   Go to the Instances page. In the top navigation bar, select the region where the RDS instance

resides. Then, find the RDS instance based on the **Creation Time**. ApsaraDB RDS requires approximately 10 minutes to create an RDS instance.

What to do next

- Configure an IP address whitelist or security group for an ApsaraDB RDS for MariaDB TX instance
- Create a database and account on an ApsaraDB RDS for MariaDB instance
- Apply for or release a public endpoint for an ApsaraDB RDS for MariaDB TX instance
- Connect to an ApsaraDB RDS for MariaDB TX instance

FAQ

- After I create an RDS instance, why does the ApsaraDB RDS console not respond and why am I unable to find the RDS instance?

  This issue may occur due to the following reasons:

  - The region that you selected is not the region where the RDS instance resides.

    In the top navigation bar, select the region where the RDS instance resides. Then, you can find the RDS instance.

  - The zone that you selected cannot provide sufficient resources.

    Resources are dynamically allocated within zones. After you submit the purchase order, the zone that you selected may run out of resources. As a result, the RDS instance cannot be created. We recommend that you select a different zone and try again. If the RDS instance still cannot be created, you can go to the the Orders page in the Billing Management console to view the refunded fee.

- How do I authorize a RAM user to manage my RDS instance?

  For more information, see Use RAM to manage ApsaraDB RDS permissions.

- If my RDS instance resides in a VPC, how many private IP addresses does it have?

  The number of private IP addresses that your RDS instance has varies based on the database engine and RDS edition that are used.

  - MySQL 5.5, 5.6, 5.7, and 8.0 on RDS High-availability Edition with local SSDs: 1
  - MySQL 5.6, 5.7, and 8.0 on RDS Enterprise Edition with local SSDs: 1
  - MySQL 5.7 on RDS Basic Edition with standard SSDs: 1
  - MySQL 8.0 on RDS Basic Edition with standard SSDs: 2
  - MySQL 5.7 and 8.0 on RDS High-availability Edition with standard SSDs or ESSDs: 3
  - MySQL 5.7 and 8.0 on RDS Enterprise Edition with standard SSDs or ESSDs: 1

References

- For more information about how to create an RDS instance by using the ApsaraDB RDS API, see Create an instance.
- For more information about how to create an RDS instance that runs a different database engine, see the following topics:
  - Create an ApsaraDB RDS for SQL Server instance
  - Create an ApsaraDB RDS for PostgreSQL instance
  - Create an ApsaraDB RDS for MariaDB TX instance

5.3. Configure an IP address whitelist or security group for an ApsaraDB RDS for MariaDB TX instance

This topic describes how to configure an IP address whitelist or security group for an ApsaraDB RDS for MariaDB TX instance. Only the devices whose IP addresses are included in an IP address whitelist of your RDS instance can access your RDS instance.

Context

You can control access to your RDS instance by using one of the following methods:

- IP address whitelists

  An IP address whitelist contains the IP addresses of the devices that require access to your RDS instance. The IP address whitelist labeled default contains only the 127.0.0.1 IP address. This IP address indicates that no devices can access your RDS instance.

  Before you configure an IP address whitelist, you must confirm the network isolation mode of your RDS instance. The configuration procedure vary based on the network isolation mode.

  - Standard whitelist mode

    A standard IP address whitelist can contain the IP addresses from both the classic network and virtual private clouds (VPCs). However, the standard whitelist mode may incur security risks. For example, after you add an IP address from a VPC to a standard IP address whitelist, the IP address is granted access over both the VPC and the classic network. Therefore, we recommend that you switch your RDS instance to the enhanced whitelist mode. For more information, see Switch an ApsaraDB RDS for MariaDB TX instance to the enhanced whitelist mode.

    > **Note** RDS instances that run MariaDB TX can be deployed only in VPCs.

  - Enhanced whitelist mode

    An enhanced IP address whitelist can contain only the IP addresses from the classic network or from VPCs. When you create an enhanced IP address whitelist, you must specify its network type. If you add an IP address from a VPC to an enhanced IP address whitelist, the IP address is granted access only over the VPC.

- Security groups

  A security group serves as a virtual firewall to control the inbound and outbound traffic of the ECS instances in that security group. After you add a security group to your RDS instance, all the ECS instances in that security group can access your RDS instance.

  For more information about security groups, see Create a security group.

IP address whitelists help provide high security and efficient protection for your RDS instance. We recommend that you update the configured IP address whitelists on a regular basis. When you configure an IP address whitelist, the workloads on your RDS instance run as normal.

Precautions for configuring an IP address whitelist

- You can modify or clear the IP address whitelist labeled default. However, you cannot delete this IP

address whitelist.

- A maximum of 50 IP address whitelists can be configured for each RDS instance.
- Up to 1,000 IP addresses and Classless Inter-Domain Routing (CIDR) blocks can be granted access to each RDS instance. If you want to add a large number of IP addresses, we recommend that you merge these IP addresses into CIDR blocks, such as 10.10.10.0/24, in which 24 indicates that the prefix of each IP address is 24-bit long. You can replace 24 with a value within the range of 1 to 32. For more information, see CIDR block FAQ.
- When you access an Alibaba Cloud service, the service automatically creates an IP address whitelist. The created IP address whitelist contains the IP address of the server that runs the service. For example, Data Management (DMS) creates an IP address whitelist named **ali_dms_group**, and Database Autonomy Service (DAS) creates an IP address whitelist named **hdm_security_ips**. To ensure that the specified Alibaba Cloud services can be used, do not modify or delete these IP address whitelists.

  Notice Do not add your IP address to these IP address whitelists. If you add your IP address to these IP address whitelists, your IP address may be overwritten by the entries that are updated from the existing IP addresses in these IP address whitelists. If your IP address is overwritten, your workloads are interrupted.

Configure an enhanced IP address whitelist

1.
2. In the left-side navigation pane, click **Data Security**.
3. Confirm the connection scenario and perform the required operations.

Connection scenario	Operation
Your ECS and RDS instances reside in the same VPC. This is the recommended connection scenario.	i. On the **Whitelist Settings** tab of the Data Security page, click **Modify** to the right of the IP address whitelist labeled **default Classic Network**. ii. In the dialog box that appears, enter the private IP address of your ECS instance in the IP Addresses field and click **OK**. **Note** The applications that run on your ECS instance connect to the internal endpoint of your RDS instance.

Connection scenario	Operation
Your ECS and RDS instances reside in different VPCs.	i. On the **Database Connection** page, click **Switch to Classic Network**. In the message that appears, click OK. ii. Click **Switch to VPC**. In the dialog box that appears, select the VPC of your ECS instance and click OK. **Note** Your ECS and RDS instances can reside in the same VPC only when they belong to the same region. If these instances belong to different regions, we recommend that you use Data Transmission Service (DTS) to migrate your RDS instance to the region of your ECS instance. For more information, see Migrate data between ApsaraDB RDS for MariaDB TX instances. iii. On the **Whitelist Settings** tab of the Data Security page, click **Modify** to the right of the IP address whitelist labeled **default VPC**. iv. In the dialog box that appears, enter the private IP address of your ECS instance in the IP Addresses field and click **OK**. **Note** The applications that run on your ECS instance connect to the internal endpoint of your RDS instance.
Your ECS instance resides in the classic network. Your RDS instance resides in a VPC.	i. Migrate your ECS instance to the VPC of your RDS instance. For more information, see Migrate an ECS instance from a classic network to a VPC. **Note** Your ECS and RDS instances can reside in the same VPC only when they belong to the same region. If these instances belong to different regions, we recommend that you use DTS to migrate your RDS instance to the region of your ECS instance. For more information, see Migrate data between ApsaraDB RDS for MariaDB TX instances. ii. On the **Whitelist Settings** tab of the Data Security page, click **Modify** to the right of the IP address whitelist labeled **default VPC**. iii. In the dialog box that appears, enter the private IP address of your ECS instance in the IP Addresses field and click **OK**. **Note** The applications that run on your ECS instance connect to the internal endpoint of your RDS instance.

Connection scenario	Operation
Your host that requires access to your RDS instance resides outside the cloud.	i. On the **Whitelist Settings** tab of the Data Security page, click **Modify** to the right of the IP address whitelist labeled **default Classic Network**. ii. In the dialog box that appears, enter the public IP address of the on-premises server in the IP Addresses field and click **OK**. **Note** ■ The applications that run on your host connect to the public endpoint of your RDS instance. ■ For more information about how to obtain the public IP address of your host, see Why am I unable to connect to my ApsaraDB RDS for MySQL or ApsaraDB RDS for MariaDB instance from a local server over the Internet?

Note

- On the Whitelist Settings tab of the Data Security page, you can click **Create Whitelist**. In the Create Whitelist dialog box, you can set the Network Type parameter to **VPC** or **Classic Network/Public IP**.
- If you enter more than one IP address or CIDR block, you must separate them with commas (,). Example: 192.168.0.1,172.16.213.9.
- If you click **Loading ECS Inner IP**, the IP addresses of all the ECS instances that are created within your Alibaba Cloud account appear. Then, you can select the IP addresses that you want to add to the IP address whitelist.

Configure a standard IP address whitelist

1.
2. In the left-side navigation pane, click **Data Security**.
3. On the **Whitelist Settings** tab of the page that appears, click **Modify** to the IP address whitelist labeled **default**.

   Note You can also click **Create Whitelist** to create an IP address whitelist.

4. In the **Edit Whitelist** dialog box, enter the IP addresses or CIDR blocks that require access to your RDS instance and click **OK**.

> **Note**
>
> - After you add IP addresses or CIDR blocks to the IP address whitelist labeled **default**, the default IP address 127.0.0.1 is automatically deleted from this IP address whitelist.
> - If you enter more than one IP address or CIDR block, you must separate them with commas (,). Do not add spaces preceding or following the commas. Example: `192.168.0.1,172.16.213.9`.
> - If you click **Loading ECS Inner IP**, the IP addresses of all the ECS instances that are created within your Alibaba Cloud account appear. Then, you can select the IP addresses that you want to add to the IP address whitelist.

Common errors

- Your RDS instance has only one IP address whitelist that contains only the default IP address 127.0.0.1 on the **Whitelist Settings** tab of the Data Security page.

  The default IP address 127.0.0.1 indicates that no devices can access your RDS instance. You must add the IP addresses of the devices that require access to your RDS instance to an IP address whitelist.

- An IP address whitelist contains only one entry, 0.0.0.0.

  If you want to grant access from all devices to your RDS instance, enter the 0.0.0.0/0 entry in an IP address whitelist.

  > **Note** The 0.0.0.0/0 entry indicates that all devices can access your RDS instance. Exercise caution when you add this entry.

- When you configure an enhanced IP address whitelist for your RDS instance, IP address errors are reported.

  Check that the enhanced whitelist mode is enabled. For more information, see Switch an ApsaraDB RDS for MariaDB TX instance to the enhanced whitelist mode.

  - If your RDS instance resides in a VPC and is connected by using the internal endpoint, make sure that the private IP address of your ECS instance is added to the IP address whitelist labeled **default VPC**.
  - If your RDS instance resides in the classic network and is connected by using the internal endpoint, make sure that the private IP address of your ECS instance is added to the IP address whitelist labeled **default Classic Network**.
  - If your RDS instance is connected over the Internet, make sure that the public IP address of your ECS instance is added to the IP address whitelist labeled **default Classic Network**. The IP address whitelist labeled default VPC cannot be used to control access over the Internet.

- The public IP addresses that you add to an IP address whitelist are not the actual egress IP addresses of the devices that you want to connect.

  This problem may occur due to the following reasons:

  - Public IP addresses dynamically change.
  - The tool or website that is used to query public IP addresses returns inaccurate results.

For more information, see Why am I unable to connect to my ApsaraDB RDS for MySQL or ApsaraDB RDS for MariaDB instance from a local server over the Internet?

Precautions for configuring a security group

- You can configure both IP address whitelists and security groups for your RDS instance. All the IP addresses in the configured IP address whitelists and all the ECS instances in the configured security groups are granted access to your RDS instance.
- A maximum of 10 security groups can be configured for each RDS instance.
- After the ECS instances in a configured security group are updated, the updates are automatically synchronized to that security group.
- You can configure only a security group that has the same network type as your RDS instance. The network types of your RDS instance and the security group that you want to configure must both be VPC or classic network.

> **Note** After you change the network type of your RDS instance, the security group that you have added becomes invalid. You must add the security group with the required network type again.

Configure a security group

1.
2. In the left-side navigation pane, click **Data Security**.
3. On the **Security Group** tab of the page that appears, click **Add Security Group**.

> **Note** Security groups whose names are followed by a VPC tag contain ECS instances that reside in VPCs.

FAQ

- After I configure an IP address whitelist, does the IP address whitelist immediately take effect?

  No, after you configure an IP address whitelist, the IP address whitelist requires about 1 minute to take effect.

- Why do I find IP address whitelists that I did not create?

  If these IP address whitelists contain private IP addresses, they are probably created by other Alibaba Cloud services, such as DMS and DAS. In this case, these IP address whitelists do not affect your business data, and no further actions are required.

- If I disable Internet access and enable only internal network access, is my RDS instance exposed to security risks?

  Yes, if you disable Internet access and enable only internal network access, your RDS instance is exposed to security risks. We recommend that you change the network type of your RDS instance to VPC. In this case, only an ECS instance in the same VPC can access your RDS instance after the required IP address is added to an IP address whitelist.

5.4. Create a database and account on an ApsaraDB RDS for MariaDB instance

This topic describes how to create a database and account on an RDS for MariaDB instance.

Account types

ApsaraDB RDS for MariaDB supports privileged and standard database accounts. You can manage all accounts and databases in the ApsaraDB for RDS console.

Account type	Description
Privileged account	• You can create and manage a privileged account by using the ApsaraDB for RDS console or APIs. • You can create only one privileged account on each RDS instance. The privileged account can be used to manage all standard accounts and databases on the instance. • A privileged account allows you to manage permissions to a fine level. For example, you can grant each standard account the permissions to query specific tables. • A privileged account has all permissions on databases created on the instance. • A privileged account has permissions to disconnect all standard accounts on the instance.
Standard account	• You can create and manage standard accounts by using the ApsaraDB for RDS console, APIs, or SQL statements. • You can create more than one standard account on each instance. The maximum number of standard accounts varies based on the database engine of the instance. • You must manually grant standard accounts the permissions on specific databases. • You cannot use a standard account to create, manage, or disconnect other accounts from databases.

Create a privileged account

1. Log on to the ApsaraDB for RDS console.
2. In the top navigation bar, select the region where the target RDS instance resides.
3. Find the target RDS instance and click its ID.
4. In the left-side navigation pane, click **Accounts**.
5. Click **Create Account**.
6. In the Create Account pane, configure the following parameters.

Parameter	Description

Parameter	Description
Database Account	Enter the account name. The account name must meet the following requirements: ○ Starts with a letter and ends with a letter or digit. ○ Contains lowercase letters, digits, or underscores (_). ○ Must be 2 to 16 characters in length. **Note** If the name of the privileged account is the same as that of an existing standard account, the privileged account replaces the standard account.
Account Type	Select **Privileged Account**.
Password	Enter the account password. The password must meet the following requirements: ○ Must be 8 to 32 characters in length. ○ Contains at least three of the following character types: uppercase letters, lowercase letters, digits, ○ Special characters include ! @ # $ % ^ & * ( ) _ + - =
Confirm Password	Enter the account password again.
Description	Enter a description that helps identify the account. The description can be up to 256 characters in length.

7. Click **OK**.

Reset permissions of the privileged account

If the privileged account of your RDS instance encounters exceptions, for example, its permissions are revoked by accident, follow these steps to reset the permissions:

1. Log on to the ApsaraDB for RDS console.
2. In the top navigation bar, select the region where the target RDS instance resides.
3. Find the target RDS instance and click its ID.
4. In the left-side navigation pane, click **Accounts**.
5. Find the privileged account, and click **Reset Permissions** in the **Actions** column.
6. Enter the password of the privileged account to reset its permissions.

Create a standard account

1. Log on to the ApsaraDB for RDS console.
2. In the top navigation bar, select the region where the target RDS instance resides.
3. Find the target RDS instance and click its ID.
4. In the left-side navigation pane, click **Accounts**.
5. Click **Create Account**.

6. In the Create Account pane, configure the following parameters.

Parameter	Description
Database Account	Enter the account name. The account name must meet the following requirements: ○ Starts with a letter and ends with a letter or digit. ○ Contains lowercase letters, digits, or underscores (_). ○ Must be 2 to 16 characters in length.
Account Type	Select **Standard Account**.
Authorized Databases	Select one or more databases on which you want to grant permissions to the account. You can leave this parameter empty and grant account permissions on specific databases when you create the databases. i. Select one or more databases from the Unauthorized Databases box and click **the right arrow** to add them to the Authorized Databases box. ii. In the Authorized Databases box, select **Read/Write**, **Read-only**, **DDL Only**, or **DML Only** for each authorized database. If you want to grant the same permissions on multiple authorized databases at a time, select the authorized databases and click the button in the upper-right corner. For example, click **Set All to Read/Write**. **Note** The button in the upper-right corner changes after you click it. For example, after you click **Set All to Read/Write**, the button changes to **Set All to Read-only**.
Password	Enter the account password. The password must meet the following requirements: ○ Must be 8 to 32 characters in length. ○ Contains at least three of the following character types: uppercase letters, lowercase letters, digits, ○ Special characters include ! @ # $ % ^ & * ( ) _ + - =
Confirm Password	Enter the account password again.
Description	Optional. Enter a description that helps identify the account. The description can be up to 256 characters in length.

7. Click **OK**.

Create a database

1. Log on to the ApsaraDB for RDS console.
2. In the top navigation bar, select the region where the target RDS instance resides.
3. Find the target RDS instance and click its ID.
4. In the left-side navigation pane, click **Databases**.

5. Click **Create Database**.
6. Configure the following parameters.

Parameter	Description
Database Name	○ The database name must start with a letter and end with a letter or digit. ○ The database name can contain lowercase letters, digits, underscores (_), and hyphens (-). ○ The database name must be 2 to 64 characters in length.
Supported Character Set	Select utf8, gbk, latin1, or utf8mb4.
Authorized Account	Select one or more accounts that require access to the database. You can leave this parameter empty and configure account permissions after you create the database. **Note** Only **standard accounts** are available in the drop-down list. The privileged account has all permissions on all databases without authorization.
Account Type	Select the permissions that you want to grant to the selected accounts. You can select **Read/Write**, **Read-only**, **DDL Only**, or **DML Only**.
Description	Optional. Enter information that helps identify the database. The description can be up to 256 characters in length.

7. Click **Create**.

Related operations

Operation	Description
CreateAccount	Creates an account.
CreateDatabase	Creates a database.

5.5. Connect to an ApsaraDB RDS for MariaDB TX instance

This topic describes how to connect to an ApsaraDB RDS for MariaDB TX instance. After you complete the initial configuration of your RDS instance, you can connect to your RDS instance from an Elastic Compute Service (ECS) instance or your computer.

Prerequisites

The following operations are complete:

- Create an ApsaraDB RDS for MariaDB TX instance

- Configure an IP address whitelist or security group for an ApsaraDB RDS for MariaDB TX instance
- Create an account for an ApsaraDB RDS for MariaDB TX instance

Use DMS to connect to an RDS instance

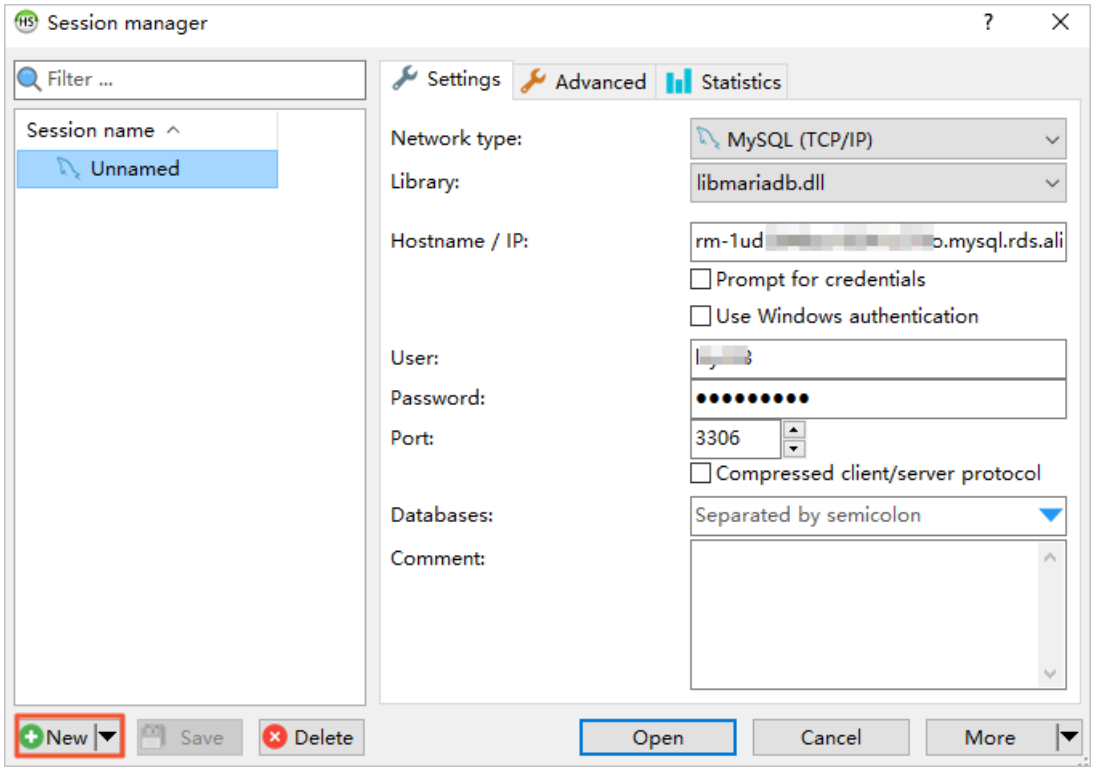
Data Management (DMS) is a graphical data management service that is used to manage relational databases and NoSQL databases. It provides various features, such as data management, schema management, user authorization, security audit, trend analysis, data tracking, business intelligence (BI) charts, and performance analysis and optimization.

Log on the ApsaraDB RDS console, find the RDS instance, and then go to the Databases page. On the **Databases** page, find the database that you want to manage, and click **SQL Query** in the Actions column. On the logon page of DMS, enter the information that is used to connect to the RDS instance.

Use a database client to connect to an RDS instance

ApsaraDB RDS is fully compatible with open source MariaDB. You can connect to the RDS instance from a common database client by using a similar method that can be used to connect to an open source MariaDB database. In the following example, HeidiSQL is used. For more information, visit the HeidiSQL website.

1. Start HeidiSQL.
2. In the lower-left corner of the Session manager dialog box, click **New**.
3. Configure the following parameters.

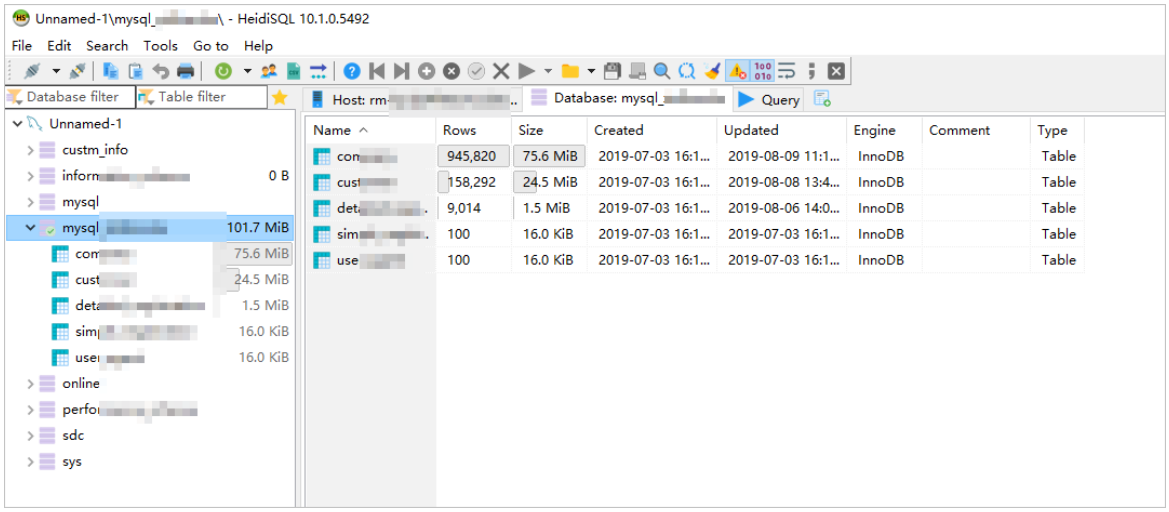


Parameter	Description
Network type	Select the network type of the RDS instance. For this example, select **MariaDB or MySQL (TCP/IP)**.

Parameter	Description
Library	Select the dynamic-link library. For this example, retain the default value.
Hostname / IP	Enter the internal or public endpoint of the RDS instance. Example: `rm-bp1xxxxxxxxxxxxx.mysql.rds.aliyuncs.com`. For more information about how to view the internal and public endpoints of an RDS instance, see View and change the internal and public endpoints and port numbers of an ApsaraDB RDS for MariaDB TX instance. ○ If HeidiSQL runs on an ECS instance that resides in the same region and has the same network type as the RDS instance, enter the internal endpoint of the RDS instance. For example, if the ECS and RDS instances both reside in virtual private clouds (VPCs) that belong to the China (Hangzhou) region, you can use the internal endpoint of the RDS instance to create a secure connection with low latency. ○ In the other scenarios, enter the public endpoint of the RDS instance.
User	Enter the username of the account that is used to connect to the RDS instance. For more information about how to create an account on an RDS instance, see Create a database and account on an ApsaraDB RDS for MariaDB instance.
Password	The password of the preceding account.
Port	Enter the port number that is used to connect to the RDS instance. If you want to connect to the RDS instance over an internal network, enter the internal port number of the RDS instance. If you want to connect to the RDS instance over the Internet, enter the public port number of the RDS instance. For more information, see View and change the internal and public endpoints and port numbers of an ApsaraDB RDS for MariaDB TX instance.

4. Click **Open**.

   If the preceding parameters are properly configured, the RDS instance can be connected.



The following common errors may occur:

- `Unknown MySQL server hose 'xxxxxxxxx'(11001)`

If this error occurs, check that the **Hostname / IP** parameter is properly set. If this parameter is set to the ID or IP address of the RDS instance, the connection fails. You must set this parameter to the internal or public endpoint of the RDS instance.

- `Access denied for user 'xxxxx'@'xxxxx'(using password:YES)`

  If this error occurs, check that the username and password of an account that is created on the RDS instance are properly specified. If you enter the username and password of your Alibaba Cloud account, the connection fails. You can create an account on the **Accounts** page of the RDS instance.

- The response is slow, and the "`Can't connect to MySQL server on 'rm-bp1xxxxxxxxxxxxxx.mysql.rds.aliyuncs.com'(10060)`" message is returned.

  If this error occurs, check that the IP address whitelists of the RDS instance are properly configured. You must make sure that the public IP address of the server that runs HeidiSQL is added to an IP address whitelist of the RDS instance. For more information about how to configure an IP address whitelist for an RDS instance, see Configure an IP address whitelist or security group for an ApsaraDB RDS for MariaDB TX instance.

  > **Note** You can temporarily add the 0.0.0.0/0 entry to an IP address whitelist of the RDS instance. If you can connect to the RDS instance by using HeidiSQL after the 0.0.0.0/0 entry is added, this error occurs due to an improperly configured IP address whitelist. You must obtain the actual public IP address of the server that runs HeidiSQL. For more information, see Why am I unable to connect to my ApsaraDB RDS for MySQL or ApsaraDB RDS for MariaDB instance from a local server over the Internet?.

FAQ

How do I use Function Compute to obtain data from my RDS instance?

You can install third-party dependencies on Function Compute. Then, you can use the built-in modules that are provided by the third-party dependencies in Function Compute to obtain data from your RDS instance. For more information, see Install third-party dependencies.

6.Data migration

6.1. Migrate data from an ApsaraDB RDS for MariaDB TX instance to an ApsaraDB RDS for MySQL instance

This topic describes how to migrate data from an ApsaraDB RDS for MariaDB TX instance to an ApsaraDB RDS for MySQL instance by using Data Transmission Service (DTS). DTS supports schema migration, full data migration, and incremental data migration. When you configure a data migration task, you can select all of the supported migration types to ensure service continuity.

Prerequisites

An ApsaraDB RDS for MySQL instance is created. For more information, see Create an ApsaraDB RDS for MySQL instance.

> **Note** The available storage space of the ApsaraDB RDS for MySQL instance must be larger than the total size of the data in the ApsaraDB RDS for MariaDB TX instance.

Precautions

- DTS uses read and write resources of the source and destination databases during full data migration. This may increase the loads of the database servers. If the database performance is unfavorable, the specification is low, or the data volume is large, database services may become unavailable. For example, DTS occupies a large amount of read and write resources in the following cases: a large number of slow SQL queries are performed on the source database, the tables have no primary keys, or a deadlock occurs in the destination database. Before you migrate data, evaluate the impact of data migration on the performance of the source and destination databases. We recommend that you migrate data during off-peak hours. For example, you can migrate data when the CPU utilization of the source and destination databases is less than 30%.
- The tables to be migrated in the source database must have PRIMARY KEY or UNIQUE constraints and all fields must be unique. Otherwise, the destination database may contain duplicate data records.
- If a data migration task fails, DTS automatically resumes the task. Before you switch your workloads to the destination instance, stop or release the data migration task. Otherwise, the data in the source instance will overwrite the data in the destination instance after the task is resumed.

Billing

Migration type	Task configuration fee	Internet traffic fee
Schema migration and full data migration	Free of charge.	Charged only when data is migrated from Alibaba Cloud over the Internet. For more information, see Pricing.
Incremental data migration	Charged. For more information, see Pricing.	

Migration types

- Schema migration

  DTS migrates the schemas of the required objects to the destination database. DTS supports schema migration for the following types of object: table, view, trigger, stored procedure, and function. DTS does not support schema migration for events.

  > Note
  > - During schema migration, DTS changes the value of the SECURITY attribute from DEFINER to INVOKER for views, stored procedures, and functions.
  > - DTS does not migrate user information. To call a view, stored procedure, or function of the destination database, you must grant the read and write permissions to INVOKER.

- Full data migration

  DTS migrates historical data of the required objects to the destination database.

  > Note
  > - During full data migration, concurrent INSERT operations cause fragmentation in the tables of the destination database. After full data migration is complete, the tablespace of the destination database is larger than that of the source database.
  > - To ensure successful data migration, we recommend that you do not perform DDL operations on the source database during full data migration.

- Incremental data migration

  After full data migration is complete, DTS retrieves binary log files from the source database, and migrates incremental data to the destination database in real time.

SQL operations that can be synchronized during incremental data migration

Operation type	SQL statement
DML	INSERT, UPDATE, DELETE, and REPLACE
DDL	• ALTER TABLE and ALTER VIEW • CREATE FUNCTION, CREATE INDEX, CREATE PROCEDURE, CREATE TABLE, and CREATE VIEW • DROP INDEX and DROP TABLE • RENAME TABLE • TRUNCATE TABLE

Permissions required for database accounts

Database	Schema migration	Full data migration	Incremental data migration
ApsaraDB RDS for MariaDB TX instance	The SELECT permission	The SELECT permission	The REPLICATION CLIENT, REPLICATION SLAVE, SHOW VIEW, and SELECT permissions

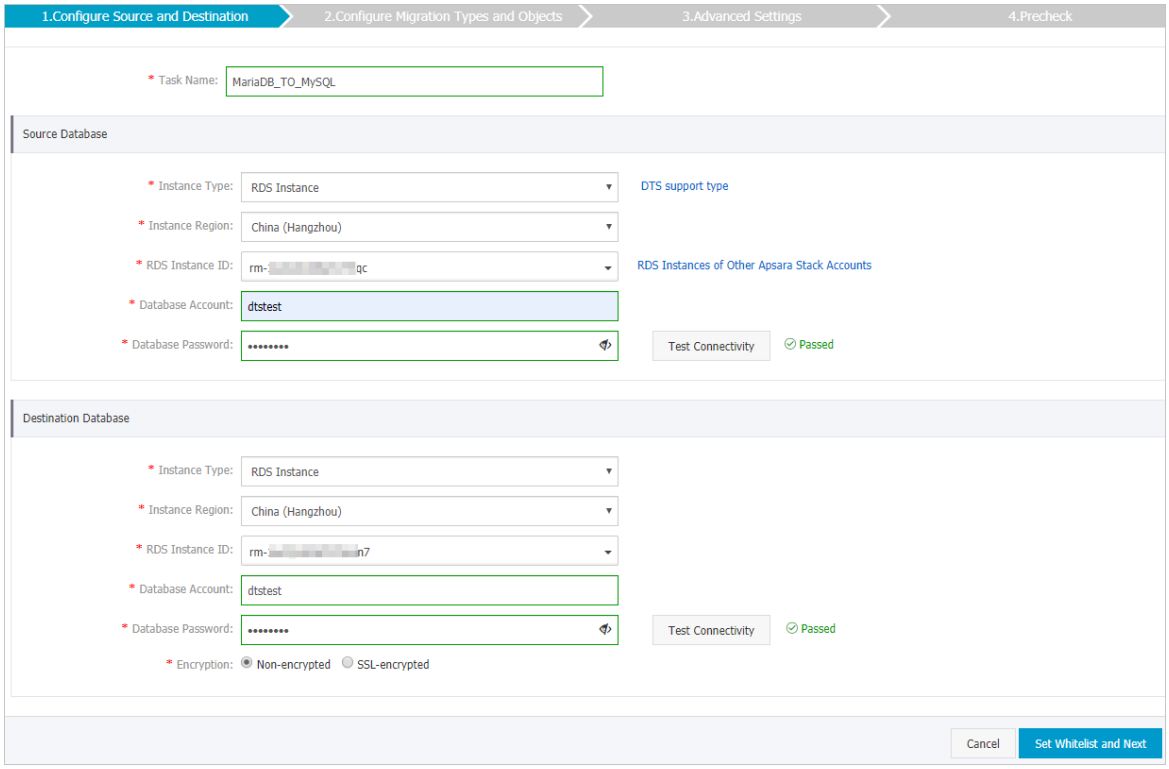
Database	Schema migration	Full data migration	Incremental data migration
ApsaraDB RDS for MySQL instance	The read and write permissions	The read and write permissions	The read and write permissions

For more information about how to create and authorize a database account, see the following topics:

- ApsaraDB RDS for MariaDB TX instance: Create a database and account on an ApsaraDB RDS for MariaDB instance
- ApsaraDB RDS for MySQL instance: Create an account on an ApsaraDB RDS for MySQL instance and Modify the permissions of a standard account on an ApsaraDB RDS for MySQL instance

Procedure

1. Log on to the DTS console.
2. In the left-side navigation pane, click **Data Migration**.
3. At the top of the **Migration Tasks** page, select the region where the destination cluster resides.
4. In the upper-right corner of the page, click **Create Migration Task**.
5. Configure the source and destination databases.



Section	Parameter	Description
N/A	Task Name	DTS automatically generates a task name. We recommend that you specify an informative name for easy identification. You do not need to use a unique task name.

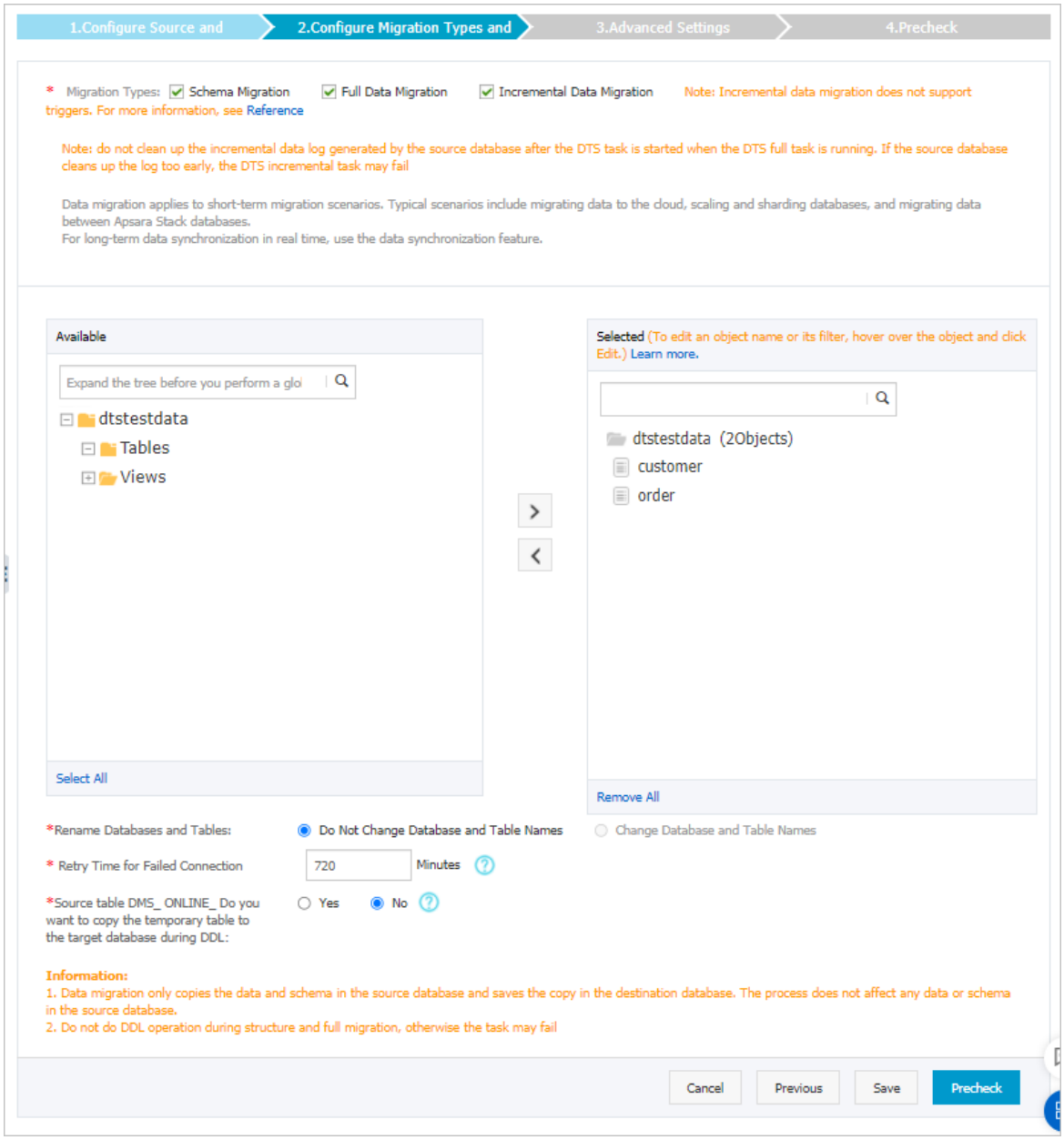
Section	Parameter	Description
Source Database	Instance Type	Select **RDS Instance**.
	Instance Region	Select the region where the source ApsaraDB RDS for MariaDB TX instance resides.
	RDS Instance ID	Select the ID of the source ApsaraDB RDS for MariaDB TX instance.
	Database Account	Enter the database account of the source ApsaraDB RDS for MariaDB TX instance. For more information about the permissions that are required for the account, see Permissions required for database accounts.
	Database Password	Enter the password of the database account. After you specify the source database parameters, click **Test Connectivity** next to **Database Password** to verify whether the specified parameters are valid. **Note** If the specified parameters are valid, the **Passed** message appears. If the **Failed** message appears, click **Check** next to **Failed**. Modify the source database parameters based on the check results.
Destination Database	Instance Type	Select **RDS Instance**.
	Instance Region	Select the region where the destination ApsaraDB RDS for MySQL instance resides.
	RDS Instance ID	Select the ID of the destination ApsaraDB RDS for MySQL instance.
	Database Account	Enter the database account of the destination ApsaraDB RDS for MySQL instance. For more information about the permissions that are required for the account, see Permissions required for database accounts.
	Database Password	Enter the password of the database account. After you specify the destination database parameters, click **Test Connectivity** next to **Database Password** to verify whether the specified parameters are valid. **Note** If the specified parameters are valid, the **Passed** message appears. If the **Failed** message appears, click **Check** next to **Failed**. Modify the destination database parameters based on the check results.

Section	Parameter	Description
	Encryption	Select **Non-encrypted** or **SSL-encrypted**. If you want to select **SSL-encrypted**, you must enable SSL encryption for the ApsaraDB RDS for MySQL instance before you configure the data migration task. For more information, see Configure SSL encryption on an ApsaraDB RDS for MySQL instance. **Note** The **Encryption** parameter is available only for regions in mainland China and the China (Hong Kong) region.

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

   Note DTS adds the CIDR blocks of DTS servers to the whitelists of the source and destination instances. This ensures that DTS servers can connect to the destination instance.

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



Setting	Description
Select the migration types	◦ To perform only full migration, select **Schema Migration** and **Full Data Migration**. ◦ To ensure service continuity during data migration, select **Schema Migration**, **Full Data Migration**, and **Incremental Data Migration**. **Notice** If **Incremental Data Migration** is not selected, we recommend that you do not write data to the source database during full data migration. This ensures data consistency between the source and destination databases.

Setting	Description
Select the objects to be migrated	Select one or more objects from the **Available** section and click the icon to move the objects to the **Selected** section. **Notice** ◦ You can select columns, tables, or databases as the objects to be migrated. ◦ By default, after an object is migrated to the destination database, the name of the object remains unchanged. You can use the object name mapping feature to rename the objects that are migrated to the destination database. For more information, see Object name mapping. ◦ If you use the object name mapping feature to rename an object, other objects that are dependent on the object may fail to be migrated.
Specify whether to rename objects	You can use the object name mapping feature to rename the objects that are migrated to the destination instance. For more information, see Object name mapping.
Specify the retry time for failed connections to the source or destination database	By default, if DTS fails to connect to the source or destination database, DTS retries within the next 720 minutes (12 hours). You can specify the retry time based on your needs. If DTS reconnects to the source and destination databases within the specified time, DTS resumes the data migration task. Otherwise, the data migration task fails. **Note** When DTS retries a connection, you are charged for the DTS instance. We recommend that you specify the retry time based on your business needs. You can also release the DTS instance at your earliest opportunity after the source and destination instances are released.

Setting	Description
Specify whether to copy temporary tables to the destination database when DMS performs online DDL operations on the source table	If you use Data Management (DMS) to perform online DDL operations on the source database, you can specify whether to migrate temporary tables generated by online DDL operations. ○ **Yes**: DTS migrates the data of temporary tables generated by online DDL operations. **Note** If online DDL operations generate a large amount of data, the data migration task may be delayed. ○ **No**: DTS does not migrate the data of temporary tables generated by online DDL operations. Only the original DDL data of the source database is migrated. **Note** If you select No, the tables in the destination database may be locked.

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

   Note
   - Before you can start the data migration task, a precheck is performed. You can start the data migration task only after the task passes the precheck.
   - If the task fails to pass the precheck, you can click the icon next to each failed item to view details.
     - You can troubleshoot the issues based on the causes and run a precheck again.
     - If you do not need to troubleshoot the issues, you can ignore failed items and run a precheck again.

9. After the task passes the precheck, click **Next**.
10. In the **Confirm Settings** dialog box, specify the **Channel Specification** parameter and select **Data Transmission Service (Pay-As-You-Go) Service Terms**.
11. Click **Buy and Start** to start the data migration task.

Stop the migration task

Warning We recommend that you prepare a rollback solution to migrate incremental data from the destination database to the source database in real time. This allows you to minimize the negative impact of switching your workloads to the destination database. For more information, see Switch workloads to the destination database. If you do not need to switch your workloads, you can perform the following steps to stop the migration task.

- Full data migration

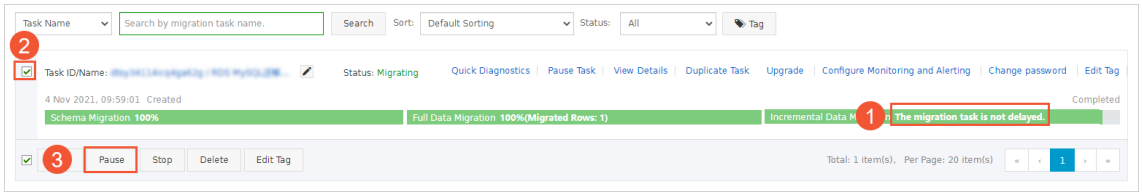
Do not manually stop a task during full data migration. Otherwise, the system may fail to migrate all data. Wait until the migration task automatically ends.

- Incremental data migration

  The task does not automatically end during incremental data migration. You must manually stop the migration task.

  i. Wait until the task progress bar shows **Incremental Data Migration** and **The migration task is not delayed**. Then, stop writing data to the source database for a few minutes. In some cases, the progress bar shows the delay time of **incremental data migration**.

  ii. After the status of **incremental data migration** changes to **The migration task is not delayed**, manually stop the migration task.



What's next

The database accounts that are used for data migration have the read and write permissions. After data migration is complete, you must delete the database accounts of both the ApsaraDB RDS for MariaDB TX instance and the ApsaraDB RDS for MySQL instance to ensure database security.

6.2. Migrate data between ApsaraDB RDS for MariaDB TX instances

This topic describes how to migrate data between ApsaraDB RDS for MariaDB TX instances by using Data Transmission Service (DTS) or the mysqldump plug-in.

Use DTS

For more information about how to use DTS to migrate data between RDS instances, see Migrate data between RDS instances.

Use mysqldump

In this topic, the RDS instances between which you migrate data run MariaDB 10.3.

Prerequisites

- CentOS7 and MySQL 5.7 are installed on your computer or an Elastic Compute Service (ECS) instance.
- The public IP address of your computer or the ECS instance is added to the IP address whitelists of two RDS instances. For more information, see Configure an IP address whitelist or security group for an ApsaraDB RDS for MariaDB TX instance.
- You have applied for public IP addresses for two RDS instances. For more information, see Apply for or release a public endpoint for an ApsaraDB RDS for MariaDB TX instance.

Procedure

1. Use a client to log on to the destination RDS instance and create a database.
2. Use the mysqldump to export the data of the database that you want to migrate of the source RDS instance as a data file.

```
mysqldump -h <Public IP address of the source RDS instance> -P <Port of the source RDS
instance> -u <Privileged account of the source RDS instance> -p<Password of the privile
ged account> --opt --default-character-set=utf8 --hex-blob <Name of the database to be
migrated> --skip-triggers > /tmp/<Name of the database to be migrated>.sql
```

Example

```
mysqldump -h rm-xxx.mariadb.rds.aliyuncs.com -P 3306 -u test -pTestxxx --opt --default-
character-set=utf8 --hex-blob testdb --skip-triggers > /tmp/testdb.sql
```

> **Note** Do not update data during the export process. This step only exports the data. It does not export stored procedures, triggers, or functions.

3. Use the mysqldump to export stored procedures, triggers, and functions.

```
mysqldump -h <Public IP address of the source RDS instance> -P <Port of the source RDS
instance> -u <Privileged account of the source RDS instance> -p<Password of the privile
ged account> --opt --default-character-set=utf8 --hex-blob <Name of the database to be
migrated> -R > /tmp/<Name of the database to be migrated>trigger.sql
```

Example

```
mysqldump -h rm-xxx.mariadb.rds.aliyuncs.com -P 3306 -u test -pTestxxx --opt --default-
character-set=utf8 --hex-blob testdb -R > /tmp/testdbtrigger.sql
```

> **Note** If the database does not have stored procedures, triggers, or functions, skip this step.

4. Execute the following statements to import the data file, stored procedures, triggers, and functions to the destination RDS instance.

```
mysql -h <Public IP address of the destination RDS instance> -P <Port of the destinatio
n RDS instance> -u <Privileged account of the destination RDS instance> -p<Password of
the privileged account> <Database name of the destination RDS instance> < /tmp/<Name of
the database to be migrated>.sql
mysql -h <Public IP address of the destination RDS instance> -P <Port of the destinatio
n RDS instance> -u <Privileged account of the destination RDS instance> -p<Password of
the privileged account> <Database name of the destination RDS instance> < /tmp/<Name of
the database to be migrated>trigger.sql
```

Example

```
mysql -h rm-xxx.mariadb.rds.aliyuncs.com -P 3306 -u test2 -pTest2xxx test001 < /tmp/tes
tdb.sql
mysql -h rm-xxx.mariadb.rds.aliyuncs.com -P 3306 -u test2 -pTest2xxx test001 < /tmp/tes
tdbtriggertrigger.sql
```

6.3. Use mysqldump to migrate data from a self-managed MariaDB database to an ApsaraDB RDS for MariaDB TX instance

This topic describes how to use mysqldump to migrate data from a self-managed MariaDB database to an ApsaraDB RDS for MariaDB TX instance.

Background information

ApsaraDB RDS for MariaDB TX is fully compatible with the native MariaDB database service. The method that is used to migrate data from a self-managed MariaDB database to an ApsaraDB RDS for MariaDB TX instance is similar to the method that is used to migrate data between two MariaDB database servers.

In this topic, the self-managed database is deployed on an on-premises server that runs Linux 7 and MariaDB 10.2.4.

Precautions

After the migration is complete, all table names are in lowercase and are not case-sensitive.

Prerequisites

The IP address of the on-premises server is added to an IP address whitelist of the RDS instance, and a public endpoint is obtained for the RDS instance. For more information, see Configure an IP address whitelist or security group for an ApsaraDB RDS for MariaDB TX instance and Apply for or release a public endpoint for an ApsaraDB RDS for MariaDB TX instance.

Procedure

1. Use a remote connection tool to log on to the RDS instance and create an empty database. The empty database is the destination database to which data is migrated. For example, you can create an empty database named test001. For more information, see Connect to an ApsaraDB RDS for MariaDB TX instance.
2. Log on to the on-premises server. Then, use mysqldump to export the data of the self-managed database as a file. This file is known as a data file.

   ```
   mysqldump -h localhost -u <The username of the account that has permissions on the self
   -managed database> -p --opt --default-character-set=utf8 --hex-blob <The name of the se
   lf-managed database> --skip-triggers > /tmp/<The name of the self-managed database>.sql
   ```

   Example:

   ```
   mysqldump -h localhost -u root -p --opt --default-character-set=utf8 --hex-blob testdb
   --skip-triggers > /tmp/testdb.sql
   ```

   Notice Do not update data during the export process. In this step, only the data is exported. The stored procedures, triggers, and functions are not exported.

3. Use the mysqldump tool to export the stored procedures, triggers, and functions as a file. This file is known as a stored procedure file.

```
mysqldump -h localhost -u <The username of the account that has permissions on the self-managed database> -p --opt --default-character-set=utf8 --hex-blob <The name of the self-managed database> -R | sed -e 's/DEFINER[ ]*=[ ]*[^*]*\*/\*/' > /tmp/<The name of the self-managed database>_trigger.sql
```

Example:

```
mysqldump -h localhost -u root -p --opt --default-character-set=utf8 --hex-blob testdb -R | sed -e 's/DEFINER[ ]*=[ ]*[^*]*\*/\*/' > /tmp/testdb_trigger.sql
```

> **Note** If the self-managed database does not contain stored procedures, triggers, or functions, you can skip this step. In this export process, you must remove DEFINER to ensure compatibility with ApsaraDB RDS for MariaDB TX.

4. Run the following commands to import the data file and the stored procedure file into the RDS instance:

```
mysql -h <The public endpoint of the RDS instance> -P <The public port of the RDS instance> -u <The username of the privileged account of the RDS instance> -p <The name of the destination database on the RDS instance> < /tmp/<The name of the self-managed database>.sql
mysql -h <The public endpoint of the RDS instance> -P <The public port of the RDS instance> -u <The username of the privileged account of the RDS instance> -p <The name of the destination database on the RDS instance> < /tmp/<The name of the self-managed database>trigger.sql
```

Example:

```
mysql -h rm-bpxxxxx.mariadb.rds.aliyuncs.com -P 3306 -u testuser -p test001 < /tmp/testdb.sql
mysql -h rm-bpxxxxx.mariadb.rds.aliyuncs.com -P 3306 -u testuser -p test001 < /tmp/testdb_trigger.sql
```

5. Refresh the remote connection tool and view the tables in the destination database of the RDS instance. If the tables contain data, the migration is successful.

7.Billing

7.1. Change the billing method of an ApsaraDB RDS for MariaDB instance from pay-as-you-go to subscription

This topic describes how to change the billing method of an ApsaraDB RDS for MariaDB instance from pay-as-you-go to subscription.

Impacts

The change of the billing method does not affect the running of your RDS instance.

Precautions

If you upgrade specifications of an RDS instance before you pay for the subscription order, the order becomes invalid. You must cancel this order on the Billing Management page and change the billing method of the RDS instance to subscription again.

Prerequisites

- The type of the RDS instance is not phased out. For more information, see Phased-out instance types. If you need to change the billing method of an RDS instance of a phased-out type to subscription, first change the instance type. For more information, see Change the configuration of an RDS MariaDB instance.
- The billing method of the RDS instance is pay-as-you-go.
- The RDS instance is in the Running state.
- The RDS instance does not have unpaid subscription orders.

Procedure

1. Log on to the ApsaraDB for RDS console.
2. In the top navigation bar, select the region where the target RDS instance resides.
3. Find the instance and go to the **Switch to Subscription Billing** page by using either of the following methods:
   - Click **Switch to Subscription Billing** in the Actions column of the RDS instance.
   - Click the instance ID. In the **Status** section, click **Switch to Subscription Billing**.
4. Select the duration of the subscription.
5. Click **Pay Now**.

   > **Note** A subscription order is generated. If you do not pay for the order or the order becomes invalid, you cannot purchase a new RDS instance or change the billing method of an instance unless you cancel the order. You can pay for or cancel the order on the Billing Management page.

6. Pay for the order.

Related operations

Operation	Description
Change the billing method	Changes the billing method of an ApsaraDB RDS instance.

7.2. Switch an ApsaraDB RDS for MariaDB TX instance from subscription to pay-as-you-go

This topic describes how to switch an ApsaraDB RDS for MariaDB TX instance from the subscription billing method to the pay-as-you-go billing method based on your business requirements.

Prerequisites

- Your RDS instance uses the subscription billing method. For more information about the billing methods of ApsaraDB RDS, see Billable items, billing methods, and pricing.
- Your RDS instance is in the Running state.
- Your RDS instance does not use a phased-out instance type. For more information, see Primary instance types. If your RDS instance uses a phased-out instance type, you must change the instance type before you change the billing method.

Fees

After you switch your RDS instance to the pay-as-you-go billing method, a refund is returned based on the payment method that is used.

Refund = Fee actually paid - Fee for consumed resources

- The fee actually paid is the money that you paid and does not include the part that is covered by coupons or vouchers.
- The fee for consumed resources is calculated based on the following formula: Fee for consumed resources = Daily subscription fee x Consumed subscription duration x Discount for the consumed subscription duration. The daily fee is equal to the order-specific subscription fee divided by 30.

> **Note** The consumed subscription duration is accurate to the day. The part that is less than one day is counted as one day.

Impacts

When you change the billing method, the workloads on your RDS instance run as normal.

> **Note** The subscription billing method is more cost-effective than the pay-as-you-go billing method. In addition, you are offered higher discounts for longer subscription periods. For long-term use, we recommend that you select the subscription billing method.

Procedure

1.
2.
3.
4. In the **Status** section of the page that appears, click **Convert to Pay as you go**.
5. Confirm the instance configuration, read and select Terms of Service, click **Pay Now**, and then complete the payment.

Related operations

Operation	Description
Change the billing method	Changes the billing method of an ApsaraDB RDS instance.

7.3. Manually renew an ApsaraDB RDS for MariaDB instance

This topic describes how to manually renew an ApsaraDB RDS for MariaDB instance that uses the subscription billing method. We recommend that you manually renew your RDS instance before the expiration date. This allows you to prevent service interruptions and data losses.

For more information about the impacts that are caused by subscription expiration, see Unlock or rebuild an expired or overdue ApsaraDB RDS instance.

> **Note** RDS instances that use the pay-as-you-go billing method do not expire and therefore do not require renewal.

You can manually renew your RDS instance before it expires or within 15 days after it expires.

Method 1: Renew an RDS instance in the ApsaraDB RDS console

Renew a single RDS instance

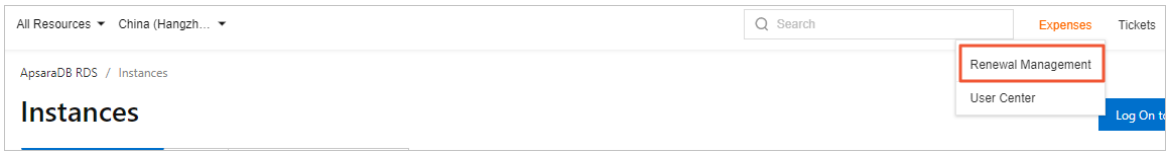
1.
2. In the **Status** section of the page that appears, click **Renew** on the right.
3. On the **Renew** page, configure the **Duration** parameter. You are offered lower prices for longer subscription periods.
4. Read and select Terms of Service, click **Pay Now**, and then complete the payment.

Renew multiple RDS instances at a time

1.
2. Select the RDS instances that you want to renew and click **Renew** below the instance list.
3. In the **Renew** dialog box, confirm the selected RDS instances and click **OK** to go to the **Renewal** page.
4. On the **Manual** tab, select the RDS instances and click **Batch Renew** in the lower part of the page.
5. Configure the **Duration** parameter of each RDS instance, click **Pay**, and then complete the payment.

Method 2: Renew the instance in the Billing Management console

1. Log on to the ApsaraDB RDS console.
2. In the top navigation bar, choose **Expenses > Renewal Management**.



3. On the **Manual** tab of the Renewal page, find the RDS instances that you want to renew. You can renew one or more RDS instances at a time.
   - **Renew a single RDS instance**
     a. Find the RDS instance that you want to renew and click **Renew** in the Actions column.

     > **Note** If the RDS instance is displayed on the **Auto** or **Nonrenewal** tab, you can click **Enable Manual Renewal** in the Actions column and then click **OK** in the message that appears to manually renew the RDS instance.

     b. On the page that appears, configure the Duration parameter, click **Pay Now**, and then complete the payment.
   - **Renew multiple RDS instances at a time**
     a. Select the RDS instances that you want to renew and click **Batch Renew** in the lower part of the page.
     b. Configure the **Duration** parameter of each RDS instance, click **Pay**, and then complete the payment.

Enable auto-renewal for an RDS instance

After auto-renewal is enabled for an RDS instance, you do not need to renew the RDS instance on a regular basis. This allows you to prevent service interruptions that are caused by subscription expiration. For more information, see Enable auto-renewal for an ApsaraDB RDS for MariaDB TX instance.

7.4. Enable auto-renewal for an ApsaraDB RDS for MariaDB TX instance

This topic describes how to enable auto-renewal for an ApsaraDB RDS for MariaDB TX instance that uses the subscription billing method. If you enable auto-renewal for your RDS instance, you do not need to manually renew your subscription or be concerned about service interruptions caused by subscription expiration.

If you do not renew your RDS instance before the expiration date, your RDS instance expires. As a result, your workloads are interrupted and your data may be lost. For more information, see Unlock or rebuild an expired or overdue ApsaraDB for RDS instance.

> **Note** RDS instances that use the pay-as-you-go billing method do not expire and therefore do not require renewal.

Precautions

- If you enable auto-renewal, the first time when the system deducts the subscription fee from your Alibaba Cloud account comes at 08:00:00 three days before your RDS instance expires. If the deduction fails, the system attempts to deduct the subscription fee every day for the next two days.

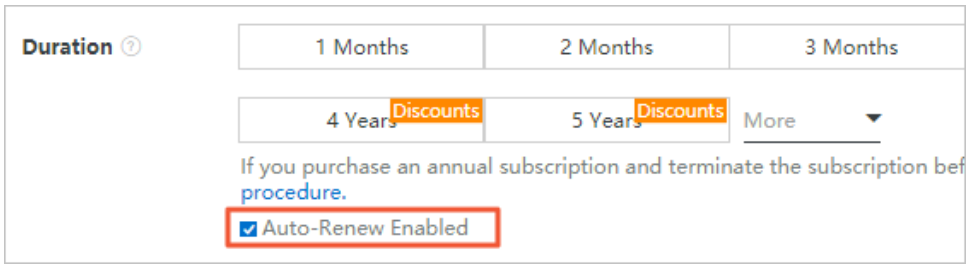
  > **Note** Make sure that the balance of your Alibaba Cloud account is sufficient. Otherwise, the renewal fails. If all the three automatic fee deduction attempts fail, you must manually renew your RDS instance in a timely manner. This allows you to avoid service interruptions and data losses.

- If you manually renew your RDS instance before the system starts automatic fee deduction attempts, the system will automatically renew the instance next time before the expiration date.
- After you enable auto-renewal, it takes effect the next day. If your RDS instance is due to expire the next day, renew it manually to avoid service interruptions. For more information, see Manually renew an ApsaraDB RDS for MariaDB instance.

Enable auto-renewal when you purchase an RDS instance

> **Note** If you select auto-renewal when you purchase an RDS instance, the system automatically renews the RDS instance based on the specified renewal cycle. The renewal cycle is one month or one year. For example, if you select auto-renewal when you purchase an RDS instance with a six-month subscription, the system automatically renews the RDS instance with a one-month subscription each time the instance is due to expire.

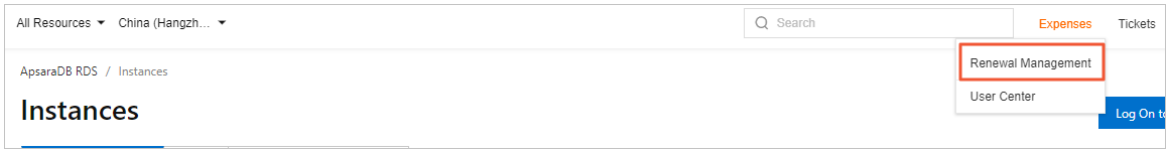
When you purchase a subscription RDS instance, select **Auto-Renew Enabled**.



Enable auto-renewal after you purchase an RDS instance

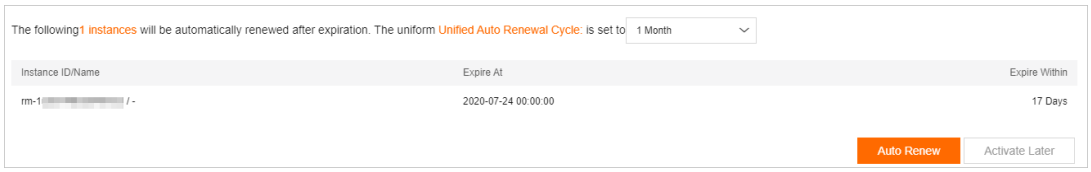
> **Note** After you enable auto-renewal for a created RDS instance, the system automatically renews the RDS instance based on the selected renewal cycle. For example, if you select a three-month renewal cycle, you are charged for a three-month subscription in each renewal cycle.

1. Log on to the ApsaraDB RDS console.
2. In the top navigation bar, choose **Expenses > Renewal Management**.



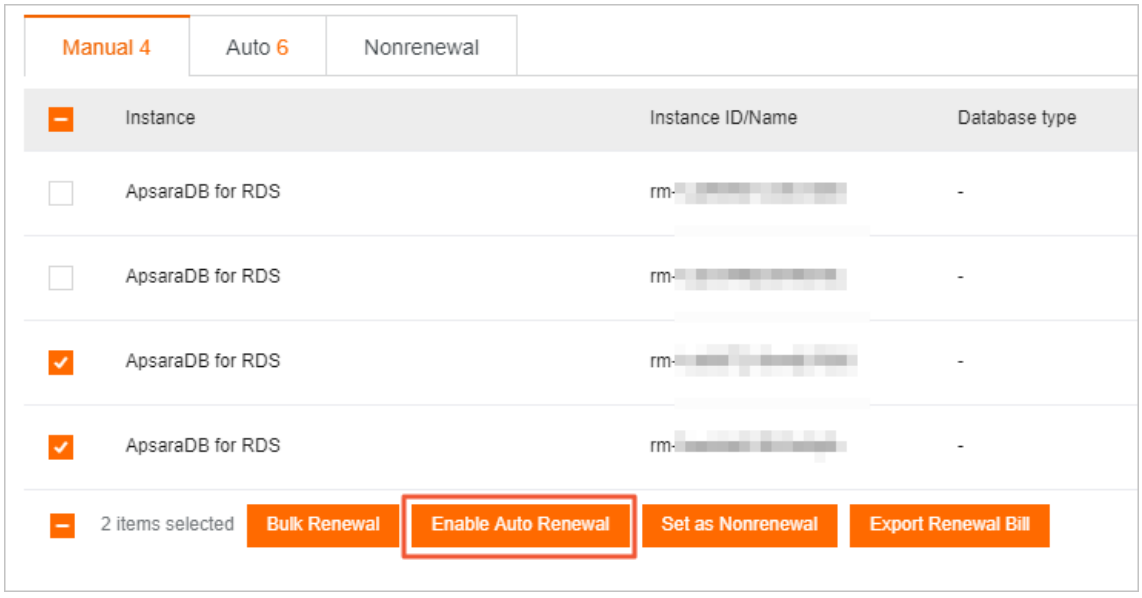
3. On the **Manual** or **Nonrenewal** tab, specify the filter conditions to find the RDS instance for which you want to enable auto-renewal. You can enable auto-renewal for one or more RDS instances at a time.
   - Enable auto-renewal for a single RDS instance.
     a. Find the RDS instance and in the Actions column click **Enable Auto Renewal**.

     b. In the dialog box that appears, specify the **Unified Auto Renewal Cycle** parameter and click **Auto Renew**.

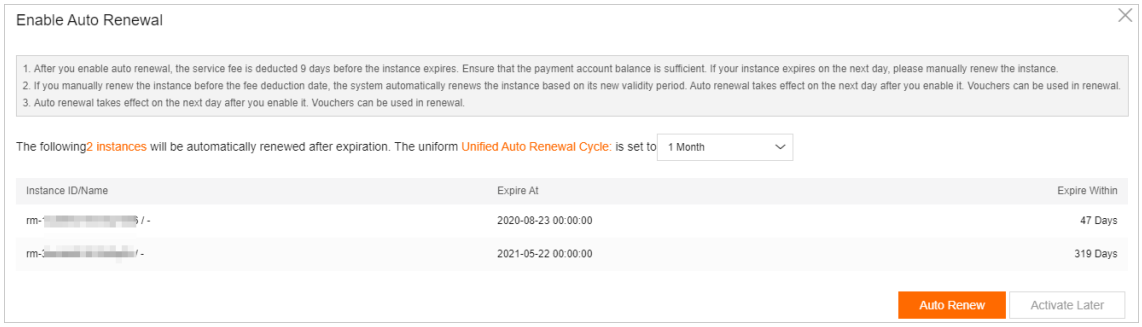


   - Enable auto-renewal for multiple RDS instances.

     Select the RDS instances and click **Enable Auto Renewal** below the instance list.

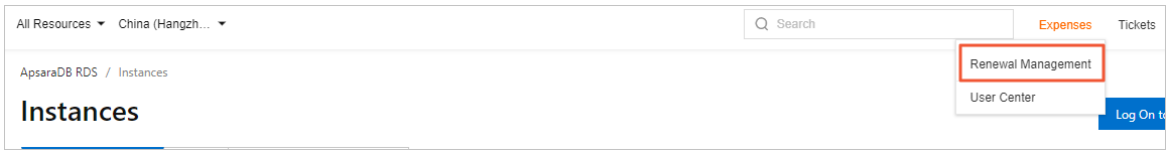


   - In the dialog box that appears, specify the **Unified Auto Renewal Cycle** parameter and click **Auto Renew**.

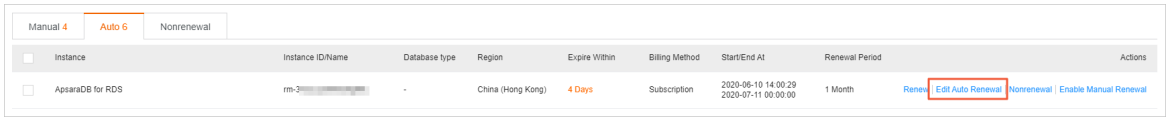


Change the auto-renewal cycle

1. Log on to the ApsaraDB RDS console.
2. In the top navigation bar, choose **Expenses > Renewal Management**.



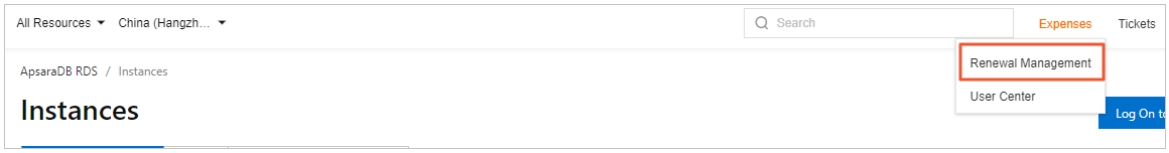
3. On the **Auto** tab, specify filter conditions to find the RDS instance for which you want to enable auto-renewal. Then, select the RDS instance and click **Edit Auto Renewal** in the Actions column.



4. In the dialog box that appears, change the auto-renewal cycle and click **OK**.

Disable auto-renewal

1. Log on to the ApsaraDB RDS console.
2. In the top navigation bar, choose **Expenses > Renewal Management**.



3. On the **Auto** tab, specify filter conditions to find the RDS instance for which you want to enable auto-renewal. Then, select the RDS instance and click **Enable Manual Renewal** in the Actions column.

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

Related operations

Operation	Description

Operation	Description
Create an instance	Creates an ApsaraDB RDS instance. **Note** You can call this operation to enable auto-renewal for an RDS instance that you want to create.
Manually renew an ApsaraDB for RDS instance	Renews an ApsaraDB RDS instance. **Note** You can call this operation to enable auto-renewal for a created RDS instance.

8.Manage pending events

If your ApsaraDB RDS instance has an event pending to be processed, the ApsaraDB RDS console notifies you of the event, so you can handle the event at your earliest opportunity.

You can receive text messages, voice messages, and emails that notify you of pending events such as instance migration and version upgrade events. In addition, after you log on to the ApsaraDB RDS console, you are prompted to manage the pending events. You can view the types, regions, processes, precautions, and affected instances of the pending events. You can also change the value of the Scheduled Disconnection Time parameter.

Prerequisites

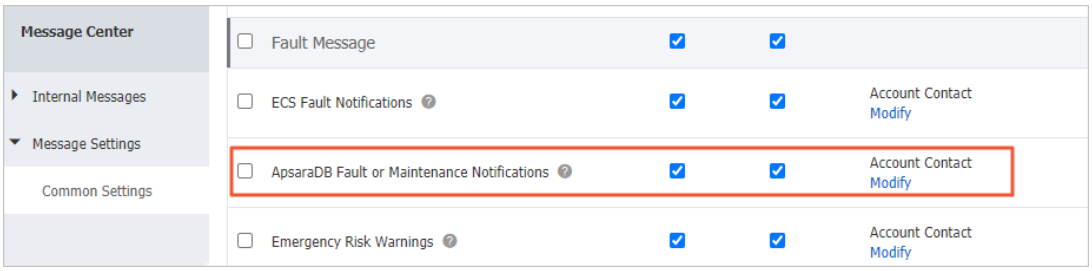
A pending event is found, which is an O&M event.

> **Note** If pending events are found, you can see notification badges on the **Pending Events** button in the upper-right corner of the ApsaraDB RDS homepage.

Precautions

You are notified of ApsaraDB for Redis pending events such as instance migrations or version upgrades at least three days before the events occur. Notifications for high-risk vulnerability fixes are sent three or fewer days before execution due to the urgency of these events. Event notifications are sent by usingphone calls, emails, internal messages, or the ApsaraDB for Redis console. To use this feature, log on to the Message Center console, enable **ApsaraDB Fault or Maintenance Notifications**, and then specify a contact. We recommend that you specify an O&M engineer as the contact.

Message Center settings



Procedure

1. Log on to the ApsaraDB RDS console.
2. Click **Events Center** in the left-side navigation pane or click **Pending Events** upper-right corner of the ApsaraDB RDS homepage,.

   > **Note** If a pending event requires you to schedule the time to handle the event, a message appears, which prompts you to schedule the time at your earliest opportunity.

3. On the **Pending Events** page, select the type and region of the event that you want to handle.

   > **Note** The content of the notification for an event varies based on the value of **Event Type**. The notification provides the process and precautions for the event.

4. View details about the event in the instance list. If you want to change the value of **Scheduled Disconnection Time**, select an RDS instance and click **Specify Disconnection Time**. In the dialog box that appears, specify the time and click **OK**.

> **Note**
> - The information that is displayed for an event varies based on the type of the event.
> - The value of **Scheduled Disconnection Time** cannot be later than the time that is displayed in the **Set Before** column.

Causes and impacts of events

Cause	Impact type	Impact description
Instance migration Primary/secondary switchover SSL certificate update Backup mode change	Transient connections	From the time specified by the RDS instance is subject to the following impacts: • The RDS instance or its database shards experience transient connections and stay in the read-only state for up to 30 seconds until all data is synchronized. We recommend that you perform the operation during off-peak hours and make sure that your application is configured to automatically reconnect to your database system. • The RDS instance cannot work as expected for Data Management (DMS) or Data Transmission Service (DTS). After the operation is complete, the RDS instance is automatically recovered. Scheduled Disconnection Time
Minor engine version update	Transient connections	From the time specified by the RDS instance is subject to the following impacts: • The RDS instance or its database shards experience transient connections and stay in the read-only state for up to 30 seconds until all data is synchronized. We recommend that you perform the operation during off-peak hours and make sure that your application is configured to automatically reconnect to your database system. • The RDS instance cannot work as expected for Data Management (DMS) or Data Transmission Service (DTS). After the operation is complete, the RDS instance is automatically recovered.
	Differences between minor engine versions	Different minor engine versions provide different features. Before you update the minor engine version of the RDS instance, you must take note of the differences between the previous and new minor engine versions. For more information, see the release notes of minor engine versions. • ApsaraDB RDS: Release notes of minor AliSQL versions, Release notes of minor AliPG versions, and Release notes of minor ApsaraDB RDS for SQL Server versions. • Engine release notes, Release notes and Release notes.

Cause	Impact type	Impact description
Proxy version upgrade	Transient connections	From the time specified by the RDS instance is subject to the following impacts: • The RDS instance or its database shards experience transient connections and stay in the read-only state for up to 30 seconds until all data is synchronized. We recommend that you perform the operation during off-peak hours and make sure that your application is configured to automatically reconnect to your database system. • The RDS instance cannot work as expected for Data Management (DMS) or Data Transmission Service (DTS). After the operation is complete, the RDS instance is automatically recovered.
	Differences between proxy versions	Different proxy versions provide different features. Before you upgrade the proxy version of the RDS instance, you must take note of the differences between the previous and new proxy versions.
Network upgrade	Transient connections	From the time specified by the RDS instance is subject to the following impacts: • The RDS instance or its database shards experience transient connections and stay in the read-only state for up to 30 seconds until all data is synchronized. We recommend that you perform the operation during off-peak hours and make sure that your application is configured to automatically reconnect to your database system. • The RDS instance cannot work as expected for Data Management (DMS) or Data Transmission Service (DTS). After the operation is complete, the RDS instance is automatically recovered.
	VIP connection errors	Network upgrades may involve cross-zone data migration. In this case, the virtual IP address (VIP) of the RDS instance changes. If a database client uses a VIP to connect to the RDS instance, the connection is interrupted. **Note** We recommend that you use a domain name to connect to the RDS instance and disable the DNS cache of your application and the DNS cache of the server on which your application runs.

9.Instance

9.1. Create an ApsaraDB RDS for MariaDB TX instance

This topic describes how to create an ApsaraDB RDS for MariaDB TX instance in the ApsaraDB RDS console. You can also create an ApsaraDB RDS for MariaDB TX instance by calling an API operation.

Prerequisites

You have an Alibaba Cloud account. For more information, see Sign up with Alibaba Cloud.

Procedure

1. Go to the ApsaraDB RDS buy page.
2. Select a billing method.
   - **Subscription**: A subscription instance is an instance for which you pay an upfront fee. For long-term use, we recommend that you select the **Subscription** billing method. If you select the subscription billing method, you must also specify the **Duration** parameter in the lower section of the page. The subscription billing method is more cost-effective than the pay-as-you-go billing method. You are offered lower prices for longer subscription periods.
   - **Pay-As-You-Go**: A pay-as-you-go instance is charged per hour based on your actual resource usage. For short-term use, we recommend that you select the **Pay-As-You-Go** billing method. If you no longer need a pay-as-you-go RDS instance, you can release the instance to reduce costs.

   > Note
   > - You can create a **pay-as-you-go** RDS instance. After you confirm that the RDS instance that you created meets your business requirements, you can change the billing method of the RDS instance to **subscription**.
   > - If you want to manage the host on which your RDS instance is deployed, you must select Dedicated Cluster (Subscription) to create a host. Then, you can create an RDS instance on the host.

3. Configure the following parameters.

Parameter	Description

Parameter	Description
Region	The region where the RDS instance resides. ○ If your application is deployed on an Elastic Compute Service (ECS) instance, the RDS instance must reside in the same region as the ECS instance. For example, the RDS instance and the ECS instance can both reside in the China (Hangzhou) region. If the RDS instance and the ECS instance reside in different regions, they cannot communicate over an internal network and therefore they cannot deliver optimal performance. ○ If your application is deployed on an on-premises server or computer, we recommend that you select a region that is in close proximity to the on-premises server or computer.
Database Engine	The database engine and version that are run by the RDS instance. Select the **MariaDB TX** database engine. Only MariaDB 10.3 is supported. **Note** The available database engines and versions vary based on the region that you select.
Edition	The RDS edition of the RDS instance. Select **High-availability**. In RDS High-availability Edition, the database system consists of a primary RDS instance and a secondary RDS instance, which work in a high-availability architecture. **Note** The available RDS editions vary based on the region and database engine version that you select. For more information, see Overview of ApsaraDB RDS editions.
Storage Type	The type of storage medium that is used by the instance. ApsaraDB RDS for MariaDB TX supports **enhanced SSDs (ESSDs)**, which come in three performance levels (PLs). ○ ESSD PL1: This is the basic PL of ESSDs. ○ ESSD PL2: An ESSD of PL2 delivers IOPS and throughput that are approximately twice higher than the IOPS and throughput delivered by an ESSD of PL1. ○ ESSD PL3: An ESSD of PL3 delivers IOPS that is up to 20 times higher than the IOPS delivered by an ESSD of PL1. An ESSD of PL3 also delivers throughput that is up to 11 times higher than the throughput delivered by an ESSD of PL1. ESSDs of PL3 are suitable for business scenarios in which highly concurrent requests must be processed with high I/O performance and at low read and write latencies. For more information, see Storage types.

Parameter	Description
Zone	The zone where the RDS instance resides. Each zone is an independent physical location within a region. For example, the China (Hangzhou) region contains Zone H, Zone I, and Zone J. ApsaraDB RDS supports the following two deployment methods: ○ **Multi-zone Deployment**: The primary RDS instance and the secondary RDS instance reside in different zones to provide zone-disaster recovery. This is the recommended deployment method. ○ **Single-zone Deployment**: The primary RDS instance and the secondary RDS instance reside in the same zone. **Note** If you select the RDS Basic Edition, you can select only the **Single-zone Deployment** method.
Instance Type	The instance type of the RDS instance. Before you select an instance type, you must select an instance family. ○ **General-purpose (Entry-level)**: A general-purpose instance exclusively occupies the allocated memory and I/O resources. However, it shares CPU and storage resources with the other general-purpose instances that are deployed on the same host. ○ **Dedicated (Enterprise-level)**: A dedicated instance exclusively occupies the allocated CPU, memory, storage, and I/O resources. The dedicated host instance family is the highest configuration of the dedicated instance family. A dedicated host instance occupies all the CPU, memory, storage, and I/O resources on the host where the instance is deployed. **Note** For more information, see Primary ApsaraDB RDS instance types.
Capacity	The maximum amount of storage that is provisioned to store data files, system files, binary log files, and transaction files in the RDS instance. You can adjust the storage capacity at a step size of 5 GB. **Note** A dedicated RDS instance that uses local SSDs exclusively occupies the allocated resources, and its storage capacity varies based on the instance type. For more information, see Primary ApsaraDB RDS instance types.

4. In the lower-right corner of the page, click **Next: Instance Configuration**.
5. Configure the following parameters.

Parameter	Description

Parameter	Description
Network Type	The network type of the RDS instance. Select **VPC**. A virtual private cloud (VPC) is an isolated virtual network that provides higher security and higher performance than the classic network. If you select the VPC network type, you must specify the **VPC** and **VSwitch of Primary Node** parameters. If you set the **Deployment Method** parameter in the previous step to **Multi-zone deployment**, you must also specify the **VSwitch of Secondary Node** parameter. **Note** The network type of the RDS instance must be the same as the network type of the Elastic Compute Service (ECS) instance that you want to connect. If the RDS instance and the ECS instance reside in VPCs, both instances must reside in the same VPC. If the RDS instance and the ECS instance reside in different VPCs, these instances cannot communicate over an internal network.
	Enable or disable the release protection feature for an ApsaraDB RDS for MySQL instance
Resource Group	The resource group to which the RDS instance belongs. You can retain the default resource group or select a custom resource group based on your business requirements.

6. In the lower-right corner of the page, click **Next: Confirm Order**.
7. Confirm the configuration of the RDS instance in the Parameters section, specify the **Purchase Plan** and **Duration** parameters, read and select **Terms of Service**, and then click **Pay Now**. You need to specify the Duration parameter only when you select the subscription billing method for the RDS instance.

   Note If you select the subscription billing method for the RDS instance, we recommend that you select **Auto-Renew Enabled**. This prevents interruptions to your workloads even if you forget to review the RDS instance.

8. View the RDS instance.

   Go to the Instances page. In the top navigation bar, select the region where the RDS instance resides. Then, find the RDS instance based on the **Creation Time**. ApsaraDB RDS requires approximately 10 minutes to create an RDS instance.

What to do next

- Configure an IP address whitelist or security group for an ApsaraDB RDS for MariaDB TX instance
- Create a database and account on an ApsaraDB RDS for MariaDB instance
- Apply for or release a public endpoint for an ApsaraDB RDS for MariaDB TX instance
- Connect to an ApsaraDB RDS for MariaDB TX instance

FAQ

- After I create an RDS instance, why does the ApsaraDB RDS console not respond and why am I unable to find the RDS instance?

  This issue may occur due to the following reasons:

- The region that you selected is not the region where the RDS instance resides.

  In the top navigation bar, select the region where the RDS instance resides. Then, you can find the RDS instance.

- The zone that you selected cannot provide sufficient resources.

  Resources are dynamically allocated within zones. After you submit the purchase order, the zone that you selected may run out of resources. As a result, the RDS instance cannot be created. We recommend that you select a different zone and try again. If the RDS instance still cannot be created, you can go to the the Orders page in the Billing Management console to view the refunded fee.

- How do I authorize a RAM user to manage my RDS instance?

  For more information, see Use RAM to manage ApsaraDB RDS permissions.

- If my RDS instance resides in a VPC, how many private IP addresses does it have?

  The number of private IP addresses that your RDS instance has varies based on the database engine and RDS edition that are used.

  - MySQL 5.5, 5.6, 5.7, and 8.0 on RDS High-availability Edition with local SSDs: 1
  - MySQL 5.6, 5.7, and 8.0 on RDS Enterprise Edition with local SSDs: 1
  - MySQL 5.7 on RDS Basic Edition with standard SSDs: 1
  - MySQL 8.0 on RDS Basic Edition with standard SSDs: 2
  - MySQL 5.7 and 8.0 on RDS High-availability Edition with standard SSDs or ESSDs: 3
  - MySQL 5.7 and 8.0 on RDS Enterprise Edition with standard SSDs or ESSDs: 1

References

- For more information about how to create an RDS instance by using the ApsaraDB RDS API, see Create an instance.
- For more information about how to create an RDS instance that runs a different database engine, see the following topics:
  - Create an ApsaraDB RDS for SQL Server instance
  - Create an ApsaraDB RDS for PostgreSQL instance
  - Create an ApsaraDB RDS for MariaDB TX instance

9.2. Restart an ApsaraDB RDS for MySQL instance

This topic describes how to manually restart an ApsaraDB RDS for MySQL instance. This applies if the number of connections exceeds the specified threshold or a performance issue occurs.

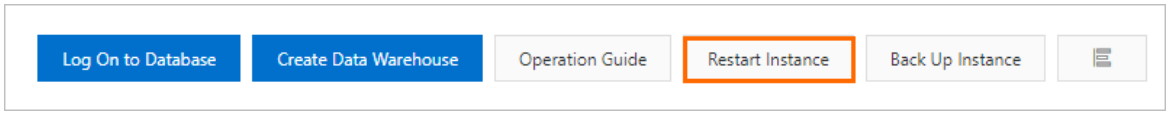
Impacts

A restart causes a network interruption that lasts about 30 seconds. Before you restart your RDS instance, we recommend that you make proper service arrangements. Proceed with caution.

Note The Basic Edition does not provide a secondary RDS instance as a hot standby for the primary RDS instance. If the primary RDS instance unexpectedly exits, your database service may be unavailable for a long period of time. If you change the specifications or upgrade the database engine version of the primary RDS instance, your database service may also be unavailable for a long period of time. If you require high service availability, we recommend that you do not select the Basic Edition. For example, you can select the High-availability Edition. Some primary RDS instances support the upgrade from the Basic Edition to the High-availability Edition. For more information, see Upgrade an RDS instance to the High-availability Edition.

Procedure

1.
2. In the upper-right corner of the Basic Information page, click **Restart Instance**.



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

Related operations

Operation	Description
Restart an ApsaraDB for RDS instance	Restarts an ApsaraDB RDS instance.

9.3. Set the maintenance window of an ApsaraDB RDS instance

This topic describes how to set the maintenance window of an ApsaraDB RDS instance. To ensure database stability, the backend system performs maintenance operations on your RDS instance every day during the maintenance window you specify. The default maintenance window spans from 02:00 to 06:00 UTC+8. We recommend that you set the maintenance window to off-peak hours to avoid interference to your business.

Precautions

- Before the backend system starts maintenance, ApsaraDB for RDS sends notification emails to the contacts listed in your Alibaba Cloud account.
- To ensure smooth maintenance, your RDS instance enters the Instance Maintaining state prior to the maintenance window. While your RDS instance stays in Instance Maintaining state, database access and query operations such as performance monitoring are still available. However, apart from account and database management and IP address whitelist configuration, all other modify operations such as upgrade, downgrade, and restart are temporarily unavailable.
- During the maintenance window, one or two transient disconnections may occur. Make sure that your application is configured to automatically reconnect to your RDS instance.

Procedure

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 resides.
3. Find the target RDS instance. Then, click its ID, or click **Manage** in the **Actions** column.
4. In the **Configuration Information** section of the Basic Information page, click **Configure** next to **Maintenance Window**.
5. Select a maintenance window and click **Save**.

> **Note** The maintenance window is in UTC+8.

Related operations

Operation	Description
Modify the maintenance time	Changes the maintenance window of an ApsaraDB for RDS instance.

9.4. Switch over workloads between primary and secondary ApsaraDB RDS for MariaDB TX instances

ApsaraDB RDS for MySQL provides the primary/secondary switchover feature to ensure high availability. If the primary RDS instance of your database system fails, ApsaraDB RDS automatically switches your workloads over from the primary RDS instance to the secondary RDS instance to ensure service availability. After the primary/secondary switchover is complete, the secondary RDS instance serves as the primary RDS instance. The endpoint that is used to connect to your database system remains unchanged. Your application can automatically connect to the new primary RDS instance by using the endpoint. You can also manually switch your workloads over between the primary RDS instance and the secondary RDS instance.

This topic describes how to switch over workloads between a primary ApsaraDB RDS for MariaDB TX instance and its secondary ApsaraDB RDS for MariaDB TX instance. If you use RDS High-availability Edition, a secondary RDS instance is provided as a standby for the primary RDS instance of your database system. The data of the primary RDS instance is synchronized in real time to the secondary RDS instance. You can access only the primary RDS instance. You cannot access the secondary RDS instance.

If the primary RDS instance fails, your workloads are automatically switched over to the secondary RDS instance.

Precautions

During a primary/secondary switchover, a transient connection may occur. Make sure that your application is configured to automatically reconnect to your database system.

Procedure

1. Log on to the ApsaraDB RDS console.
2. In the top navigation bar, select the region where the primary RDS instance resides.
3. Find the primary RDS instance and click the ID of the instance.

4. In the left-side navigation pane, click **Service Availability**.
5. In the Availability Information section of the page that appears, click **Switch Primary/Secondary Instance**.
6. Specify the time at which you want to perform a switchover. Then, click **OK**.

   During a primary/secondary switchover, you cannot perform a number of operations. For example, you cannot manage databases and accounts or change the network type. We recommend that you perform a primary/secondary switchover during the planned maintenance window.

   > **Note** If you want to change the maintenance window, perform the following operations:
   >
   > i. Click **modify** next to the Switch Within Maintenance Window option.
   >
   > ii. In the Configuration Information section of the page that appears, select a maintenance window and click **OK**.
   >
   > iii. Return to the Service Availability page, refresh the page, and continue with the procedure.

FAQ

- Can I access the secondary RDS instance of my database system?

  No, you cannot access the secondary RDS instance of your database system. You can access only the primary RDS instance of your database system. The secondary RDS instance runs only as a standby.

- Do I need to manually switch my workloads over from the secondary RDS instance to the primary RDS instance after a primary/secondary switchover?

  No, you do not need to manually switch your workloads over from the secondary RDS instance to the primary RDS instance after a primary/secondary switchover. The data in the primary RDS instance is the same as the data in the secondary RDS instance. After a primary/secondary switchover, the secondary RDS instance serves as the new primary RDS instance. No additional operations are required.

- Each time a primary/secondary switchover is performed, my RDS instance does not run as expected 10 minutes after the primary/secondary switchover is complete. What are the possible causes? How do I handle the issue?

  If an exception on your RDS instance triggers a primary/secondary switchover to ensure high availability, your application may fail to identify and respond to the changes to the connections. If no timeout periods are specified for socket connections, your application waits for the database to return the results. In most cases, your application is disconnected after hundreds of seconds. During this period, some connections to the database cannot work as expected, and a large number of SQL statements fail to be executed. To avoid invalid connections, we recommend that you configure the **connectTimeout** and **socketTimeout** parameters to prevent your application from waiting for a long period of time due to network errors. This reduces the time required to recover from failures.

  You must configure these parameters based on your workloads and usage modes. For online transactions, we recommend that you set **connectTimeout** to 1 to 2 seconds and **socketTimeout** to 60 to 90 seconds. This configuration is for reference only.

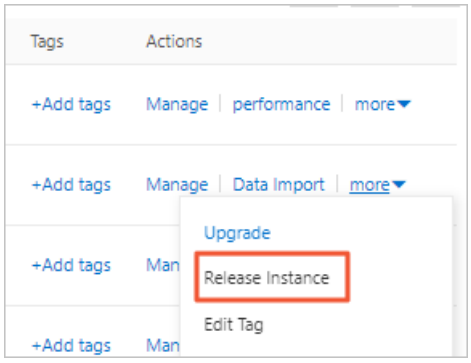
9.5. Release an RDS MariaDB instance

This topic describes how to release an RDS MariaDB 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-based 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 in the **Actions** column choose **More > Release Instance**.



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

Unsubscribe from a subscription RDS instance

If you want to unsubscribe from an RDS instance, submit a .

APIs

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

9.6. Change the configuration of an RDS MariaDB instance

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

You can upgrade or downgrade the configuration of an RDS MariaDB 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

Configuration item	Description
CPU and Memory	All MariaDB DB engine versions and editions support the CPU and memory change.
Capacity	All MariaDB DB engine versions and editions allow you to increase storage capacity. **Note** • For information about the capacity range, see Primary ApsaraDB RDS instance types. • If the storage capacity range of the current specifications cannot meet your requirements, you can change the specifications.

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

Billing

For more information, see Specification change fees.

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.

Procedure

1. Log on to the RDS console.
2. Select the target region.
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:
>
> i. Click **Modify**.
>
> ii. In the **Configuration Information** section, select a maintenance window and click **Save**.
>
> iii. 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

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.

9.7. Modify parameters for an RDS for MariaDB instance

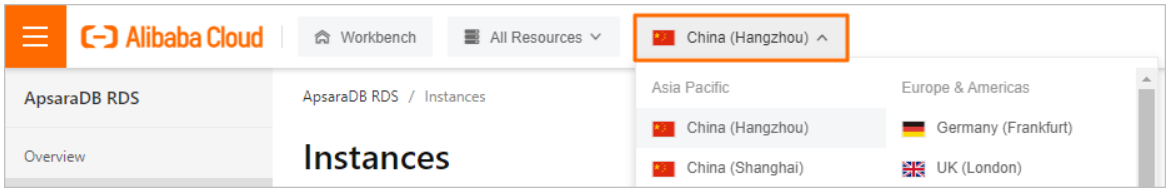
This topic describes how to use the ApsaraDB for RDS console or APIs to view and modify parameters of an RDS for MariaDB instance.

Precautions

- To ensure instance stability, the console allows you to modify only part of the parameters. If you want to modify more parameters, ,submit a ticket.
- When you modify parameters on the **Editable parameters** tab, refer to the **Value Range** column.
- After you modify certain parameters, you must restart the instance for the changes to take effect. For more information, refer to the **Force Restart** column on the **Editable parameters** tab. We recommend that you modify the parameters of an instance during off-peak hours and make sure that your application supports automatic reconnection.

Modify the parameters

1. Log on to the ApsaraDB for RDS console.
2. In the top navigation bar, select the region where the target RDS instance resides.

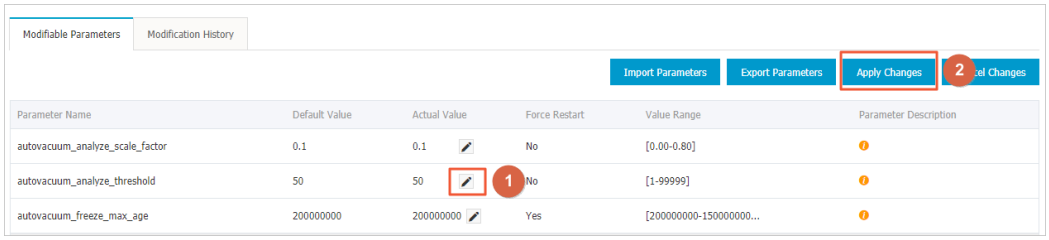


3. Find the target RDS instance and click its ID.
4. In the left-side navigation pane, select **Parameters**.
5. On the **Editable Parameters** tab, modify the target parameters. You can modify one or more

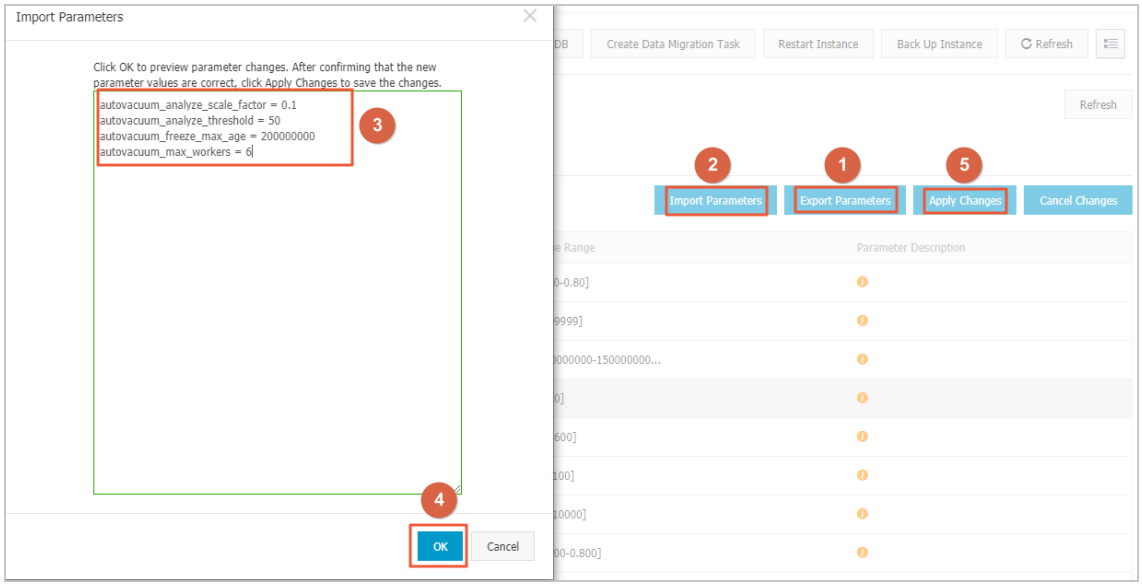
parameters at a time.

- Modify a single parameter.
  a. Click the

  icon following the target parameter.
  b. Enter a new value and click **Confirm**.
  c. In the upper-right corner of the page, click **Apply Changes**.
  d. In the Edit Parameters dialog box, click **Confirm**.

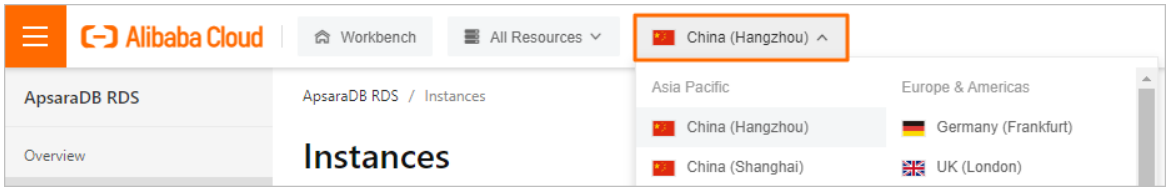


- Modify multiple parameters at a time.
  a. Click **Export Parameters**.
  b. Open the parameter file and modify the parameters.
  c. After you modify the parameters, click **Import Parameters**.
  d. In the **Import Parameters** dialog box, paste the parameters and new values and click **OK**.
  e. Confirm the parameter values in the list and click **Apply Changes**.



Query the parameter modification history

1. Log on to the ApsaraDB for RDS console.
2. In the top navigation bar, select the region where the target RDS instance resides.



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

Parameter description

For more information, see MariaDB parameters.

Related operations

Operation	Description
Modify parameters of an ApsaraDB for RDS instance	Modifies the parameters of an RDS instance.
Query the parameter template of an ApsaraDB for RDS instance	Queries the parameter templates available to an RDS instance.
Query parameter configurations	Queries the parameter settings of an RDS instance.

9.8. Adjust the size of the InnoDB buffer pool for an ApsaraDB RDS for MariaDB TX instance

This topic describes how to configure the innodb_buffer_pool_size parameter of an ApsaraDB RDS for MariaDB TX instance based on your business requirements. This way, you can improve the performance of the instance.

Background information

You can reconfigure the innodb_buffer_pool_size parameter to adjust the size of the InnoDB buffer pool for an RDS instance. The value of this parameter must be calculated by using the following formula:

```
{DBInstanceClassMemory*X/Y}
```

Example:

```
{DBInstanceClassMemory*7/10}
```

> **Note**
>
> - DBInstanceClassMemory is a system variable, which specifies the memory capacity of the RDS instance.
> - X is the numerator, and Y is the denominator.
> - The size of the InnoDB buffer pool must be within the following range: [128 MB, DBInstanceClassMemory × 8/10]. The minimum size is 128 MB, and the maximum size is 80% of the memory capacity that you purchased for the RDS instance.

The default size of the InnoDB buffer pool for an RDS instance is calculated based on the following rules:

- If the RDS instance is equipped with standard SSDs or enhanced SSDs (ESSDs) and the `purchased memory capacity of the RDS instance is less than 16 GB`, the default size of the InnoDB buffer pool is calculated by using the following formula: `Default size of the InnoDB buffer pool = (Purchased memory capacity of the RDS instance - Reserved memory capacity of the RDS instance) × 0.75`.

  > **Note** The reserved memory capacity of the RDS instance is calculated by using the following formula:
  >
  > ```
  > MIN{Purchased memory capacity of the RDS instance × 0.65, [(Purchased memory capacity of the RDS instance/16384) + 1] × 2048}
  > ```

- If the RDS instance is equipped with standard SSDs or ESSDs and the `purchased memory capacity of the RDS instance is greater than or equal to 16 GB`, the default size of the InnoDB buffer pool is calculated by using the following formula: `Default size of the InnoDB buffer pool = Purchased memory capacity of the RDS instance × 0.75`.

> **Note** The default size of the InnoDB buffer pool is an integer multiple of 128. If the calculated result is not an integer multiple of 128, an approximate integer that is an integer multiple of 128 is taken. For example, an RDS instance provides 1,024 MB of memory, the calculated result is 268, and the approximate integer that is a multiple of 128 is 256. In this case, the default size of the InnoDB buffer pool for the RDS instance is 256 MB.

The following table provides the default size and the maximum size of the InnoDB buffer pool for various memory capacities.

Memory capacity (Unit: MB)	Default buffer pool size (Unit: MB)	Maximum buffer pool size (Unit: MB)
2,048	512	512
4,096	1,536	1,536
8,192	4,608	4,608
16,384	12,288	12,288

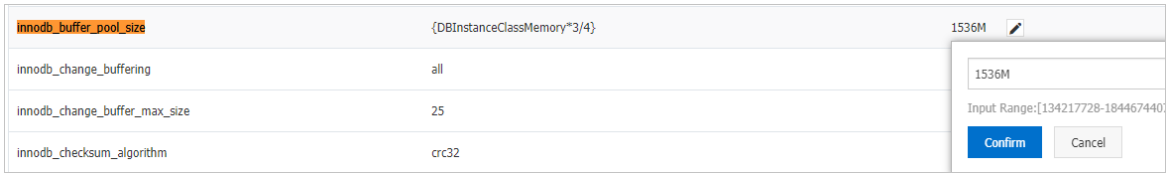
Memory capacity (Unit: MB)	Default buffer pool size (Unit: MB)	Maximum buffer pool size (Unit: MB)
32,768	24,576	25,600
65,536	49,152	52,224
131,072	98,304	104,448
196,608	147,456	156,672
229,376	172,032	183,296
262,144	196,608	208,896
491,520	368,640	393,216

The size of the InnoDB buffer pool must be a multiple of the result that is obtained by using the following formula: `Value of the innodb_buffer_pool_chunk_size parameter × Value of the innodb_buffer_pool_instances parameter`. If the size of the InnoDB buffer pool is not a multiple of the result that you obtain by using the formula, ApsaraDB RDS changes the size to a multiple of the result. For example, if the result that you obtain by using the formula is 1 GB and you set the `innodb_buffer_pool_size` parameter to 1.5 GB, ApsaraDB changes the value of the innodb_buffer_pool_size parameter to 2 GB.

Procedure

1.
2. In the left-side navigation pane, click **Parameters**.
3. Find the **innodb_buffer_pool_size** parameter and click the icon. In the dialog box that appears, enter a new value and click **OK**.

   Warning After you change the value of the **innodb_buffer_pool_size** parameter for an RDS instance, the instance restarts. Proceed with caution.

4. Click **Apply Changes** above the parameter list. In the message that appears, click **OK**. Then, wait for the RDS instance to restart.

9.9. Manage ApsaraDB RDS for MySQL instances that are in the recycle bin

Expired or overdue ApsaraDB for RDS instances are locked in the recycle bin. You can unlock or delete instances in the recycle bin.

Unlock an overdue pay-as-you-go instance

If a pay-as-you-go instance is locked due to overdue payments, check the billing method of your Alibaba Cloud account.

Unlock an expired subscription instance

If a subscription instance is locked because it is expired, you can go to the recycle bin to renew it.

1. Log on to the recycle bin.
2. In the top navigation bar, select the region where the target RDS instance resides.
3. Find the target RDS instance and click **Unlock** to renew it.

   The target RDS instance is unlocked immediately after the renewal.

Rebuild a subscription instance

After a subscription instance expires, it is retained for a specific period of time. After the period of time you specify elapses, the instance is released. The data backup files of the instance are retained for eight days. During the eight-day retention period, you can use the data backup files to rebuild a new instance. For more information, see Unlock or rebuild an expired or overdue ApsaraDB RDS instance.

1. Log on to the recycle bin.
2. In the top navigation bar, select the region where the target RDS instance resides.
3. Find the target RDS instance and click **Recreate Instance**.

   By default, the target RDS instance is rebuilt with the same specifications and in the same zone. You also have the option to rebuild the target RDS instance with different specifications and in a different zone.

Destroy an instance

If an instance is locked due to expiration or overdue payments, you can destroy it in the recycle bin.

> **Warning** Destroying an instance destroys all backups. Proceed with caution.

Procedure

1. Log on to the recycle bin.
2. In the top navigation bar, select the region where the target RDS instance resides.
3. Find the target RDS instance and click **Destroy**.

Related documentation

Unlock or rebuild an expired or overdue ApsaraDB RDS instance

10.Database connection

10.1. Connect to an ApsaraDB RDS for MariaDB TX instance

This topic describes how to connect to an ApsaraDB RDS for MariaDB TX instance. After you complete the initial configuration of your RDS instance, you can connect to your RDS instance from an Elastic Compute Service (ECS) instance or your computer.

Prerequisites

The following operations are complete:

- Create an ApsaraDB RDS for MariaDB TX instance
- Configure an IP address whitelist or security group for an ApsaraDB RDS for MariaDB TX instance
- Create an account on an ApsaraDB RDS for MariaDB TX instance

Use DMS to connect to an RDS instance

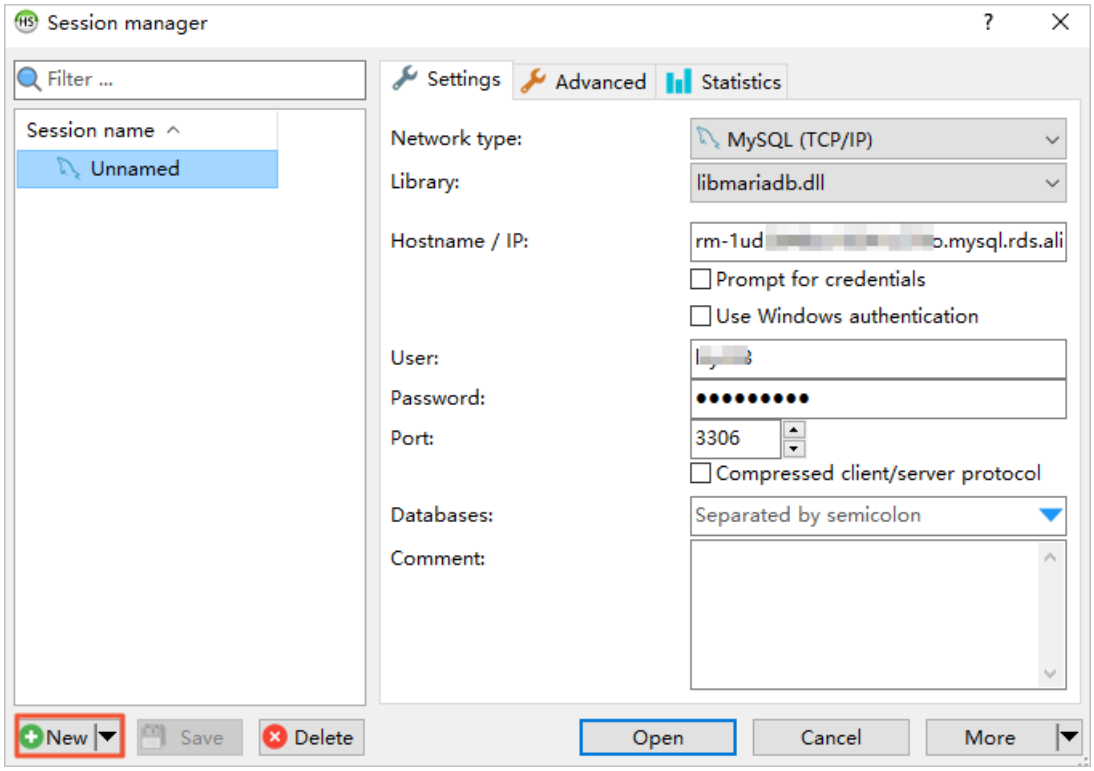
Data Management (DMS) is a graphical data management service that is used to manage relational databases and NoSQL databases. It provides various features, such as data management, schema management, user authorization, security audit, trend analysis, data tracking, business intelligence (BI) charts, and performance analysis and optimization.

Log on the ApsaraDB RDS console, find the RDS instance, and then go to the Databases page. On the **Databases** page, find the database that you want to manage, and click **SQL Query** in the Actions column. On the logon page of DMS, enter the information that is used to connect to the RDS instance.

Use a database client to connect to an RDS instance

ApsaraDB RDS is fully compatible with open source MariaDB. You can connect to the RDS instance from a common database client by using a similar method that can be used to connect to an open source MariaDB database. In the following example, HeidiSQL is used. For more information, visit the HeidiSQL website.

1. Start HeidiSQL.
2. In the lower-left corner of the Session manager dialog box, click **New**.
3. Configure the following parameters.

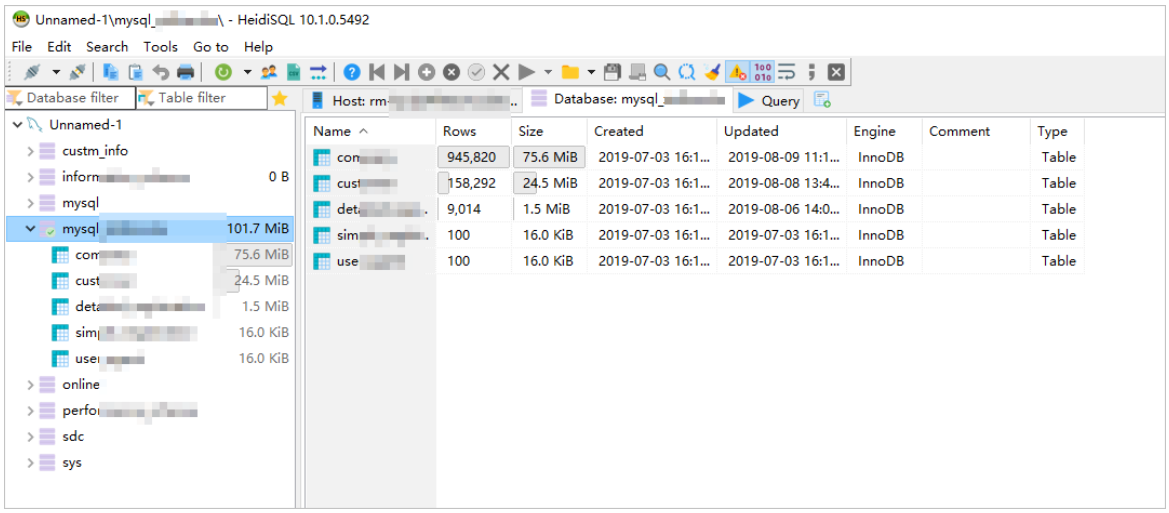


Parameter	Description
Network type	Select the network type of the RDS instance. For this example, select **MariaDB or MySQL (TCP/IP)**.
Library	Select the dynamic-link library. For this example, retain the default value.
Hostname / IP	Enter the internal or public endpoint of the RDS instance. Example: `rm-bp1xxxxxxxxxxxxx.mysql.rds.aliyuncs.com`. For more information about how to view the internal and public endpoints of an RDS instance, see View and change the internal and public endpoints and port numbers of an ApsaraDB RDS for MariaDB TX instance. ○ If HeidiSQL runs on an ECS instance that resides in the same region and has the same network type as the RDS instance, enter the internal endpoint of the RDS instance. For example, if the ECS and RDS instances both reside in virtual private clouds (VPCs) that belong to the China (Hangzhou) region, you can use the internal endpoint of the RDS instance to create a secure connection with low latency. ○ In the other scenarios, enter the public endpoint of the RDS instance.
User	Enter the username of the account that is used to connect to the RDS instance. For more information about how to create an account on an RDS instance, see Create a database and account on an ApsaraDB RDS for MariaDB instance.
Password	The password of the preceding account.

Parameter	Description
Port	Enter the port number that is used to connect to the RDS instance. If you want to connect to the RDS instance over an internal network, enter the internal port number of the RDS instance. If you want to connect to the RDS instance over the Internet, enter the public port number of the RDS instance. For more information, see View and change the internal and public endpoints and port numbers of an ApsaraDB RDS for MariaDB TX instance.

4. Click **Open**.

   If the preceding parameters are properly configured, the RDS instance can be connected.



The following common errors may occur:

- `Unknown MySQL server hose 'xxxxxxxxx'(11001)`

  If this error occurs, check that the **Hostname / IP** parameter is properly set. If this parameter is set to the ID or IP address of the RDS instance, the connection fails. You must set this parameter to the internal or public endpoint of the RDS instance.

- `Access denied for user 'xxxxx'@'xxxxx'(using password:YES)`

  If this error occurs, check that the username and password of an account that is created on the RDS instance are properly specified. If you enter the username and password of your Alibaba Cloud account, the connection fails. You can create an account on the **Accounts** page of the RDS instance.

- The response is slow, and the "`Can't connect to MySQL server on 'rm-bp1xxxxxxxxxxxxxx.mysql.rds.aliyuncs.com'(10060)`" message is returned.

  If this error occurs, check that the public IP address of the server that runs HeidiSQL is added to an IP address whitelist of the RDS instance. For more information about how to configure an IP address whitelist for an RDS instance, see Configure an IP address whitelist or security group for an ApsaraDB RDS for MariaDB TX instance.

> **Note** You can temporarily add the 0.0.0.0/0 entry to an IP address whitelist of the RDS instance. If you can connect to the RDS instance by using HeidiSQL after the 0.0.0.0/0 entry is added, this error occurs due to an improperly configured IP address whitelist. You must obtain the actual public IP address of the server that runs HeidiSQL. For more information, see Why am I unable to connect to my ApsaraDB RDS for MySQL or ApsaraDB RDS for MariaDB instance from a local server over the Internet?.

FAQ

How do I use Function Compute to obtain data from my RDS instance?

You can install third-party dependencies on Function Compute. Then, you can use the built-in modules that are provided by the third-party dependencies in Function Compute to obtain data from your RDS instance. For more information, see Install third-party dependencies.

10.2. Apply for or release a public endpoint for an ApsaraDB RDS for MariaDB TX instance

ApsaraDB RDS for MariaDB TX supports two types of endpoints: internal endpoints and public endpoints. By default, you are provided with an internal endpoint that is used to connect to the RDS instance over an internal network. If you want to connect to the RDS instance over the Internet, you must apply for a public endpoint.

Internal and public endpoints

Endpoint type	Description
Internal endpoint	• An internal endpoint is provided by default. You do not need to apply for this endpoint. In addition, you cannot release this endpoint. You can change the network type of the RDS instance. • If your application is deployed on an Elastic Compute Service (ECS) instance that resides in the same region andhas the same network types as the RDS instance, the ECS and RDS instances can communicate over an internal network. You do not need to apply for a public endpoint for the RDS instance. • For security and performance purposes, we recommend that you connect to the RDS instance by using the internal endpoint.

Endpoint type	Description

Public endpoint	• You must manually apply for a public endpoint. You can release this endpoint if it is no longer required.
	• If you cannot connect to the RDS instance by using the internal endpoint, you must apply for a public endpoint. This includes the following scenarios:
	  ○ Connect to the RDS instance from an ECS instance that resides in a different region or has a different network types from the RDS instance.
	  ○ Connect to the RDS instance from a device outside Alibaba Cloud.
	Note
	• You are not charged for the public endpoint or the traffic that is consumed.
	• If you connect to your RDS instance by using the public endpoint, security is compromised. Proceed with caution.
	• We recommend that you migrate your application to an ECS instance that resides in the same region and has the same network type as your RDS instance. This allows you to connect to your RDS instance by using the internal endpoint. The connection expedites transmission and improves security.

Procedure

1.
2. In the left-side navigation pane, click **Database Connection**.
3. Apply for or release a public endpoint for your RDS instance:
   - If you have not applied for a public endpoint, you can click **Apply for Public Endpoint**.
   - If you have applied for a public endpoint, you can click **Release Public Endpoint**.
4. In the message that appears, click **OK**.

Related operations

Operation	Description
Apply for a public endpoint	Applies for a public endpoint for an ApsaraDB RDS instance.
Release a public endpoint	Releases the public endpoint of an ApsaraDB RDS instance.

10.3. View and change the internal and public endpoints and port numbers of an ApsaraDB RDS for MariaDB TX instance

To connect to an ApsaraDB RDS for MariaDB TX instance, you must enter the internal or public endpoint and port number of the RDS instance. This topic describes how to view and change the internal and public endpoints and port numbers of an ApsaraDB RDS for MariaDB TX instance in the ApsaraDB RDS console.

Change the internal or public endpoint and port number of an RDS instance

1.
2. In the left-side navigation pane, click **Database Connection**.
3. Click **Change Endpoint**.
4. In the dialog box that appears, select a connection type, enter the prefix of the new endpoint, specify the port number, and then click **OK**.

> **Note**
>
> - The prefix can contain lowercase letters, digits, and hyphens (-). The prefix must start with a lowercase letter and end with a lowercase letter or a digit.
> - The prefix must contain at least 8 characters, and the total length of the endpoint cannot exceed 63 characters. The total length includes the prefix and suffix of the endpoint.
> - The port number must be within the range of 1000 to 65534.

FAQ

- After I change an endpoint or port number of my RDS instance, do I need to update the endpoint or port number information on my application?

  Yes, after you change an endpoint or port number of your RDS instance, you must update the endpoint or port number information on your application. Otherwise, your application cannot connect to your RDS instance.

- After I change an endpoint or port number of my RDS instance, is the change immediately applied? Do I need to restart my RDS instance?

  After you change an endpoint or port number of your RDS instance, the change is immediately applied. You do not need to restart your RDS instance.

- After I change or release an endpoint of my RDS instance, can the endpoint be used for another RDS instance?

  Yes, after you change or release an endpoint of your RDS instance, the endpoint can be used for another RDS instance.

11.Account

11.1. Create an account on an ApsaraDB RDS for MariaDB instance

This topic describes how to create an account that is used to manage the databases of an ApsaraDB RDS for MariaDB instance.

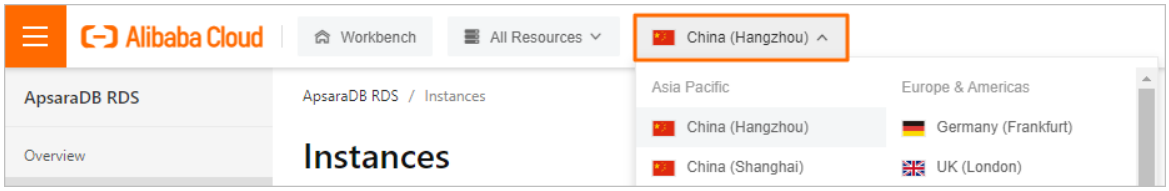
Account types

ApsaraDB RDS for MariaDB supports two types of accounts: privileged accounts and standard accounts. You can manage all accounts and databases of your RDS instance in the ApsaraDB RDS console.

Account type	Description
Privileged account	• You can create and manage privileged accounts by using the ApsaraDB RDS console or API operations. • Only one privileged account is allowed per RDS instance. A privileged account has the permissions to manage all databases and standard accounts of the RDS instance where the privileged account is created. • A privileged account allows you to manage permissions at fine-grained levels. For example, you can grant each standard account the permissions to query specific tables of the RDS instance where the privileged account is created. • A privileged account has all the permissions on the databases of the RDS instance where the privileged account is created. • A privileged account has permissions to disconnect all the standard accounts of the RDS instance where the privileged account is created.
Standard account	• You can create and manage standard accounts by using the ApsaraDB RDS console, API operations, or SQL statements. • More than one standard account is allowed per RDS instance. The maximum number of standard accounts that are allowed varies based on the used database engine. • You must manually grant the permissions on specific databases to each standard account. • A standard account does not have the permissions to create, manage, or disconnect other accounts of the RDS instance where the standard account is created.

Create a privileged account

1. Log on to the ApsaraDB RDS console.
2. In the left-side navigation pane, click **Instances**. In the top navigation bar, select the region where your RDS instance resides.



3. Find your RDS instance and click its ID.
4. In the left-side navigation pane, click **Accounts**.
5. Click **Create Account**.
6. In the Create Account panel, configure the following parameters.

Parameter	Description
Database Account	Enter the username of the account. The username must meet the following requirements: ○ The username must be 2 to 16 characters in length. ○ The username must start with a letter and end with a letter or digit. ○ The username can contain lowercase letters, digits, and underscores (_). ○ The username cannot be the same as that of an existing account. **Note** If the username of the privileged account is the same as that of an existing standard account, the privileged account replaces the standard account.
Account Type	Specify the type of the account. For this example, select **Privileged Account**.
Password	Enter the password of the account. The password must meet the following requirements: ○ The password must be 8 to 32 characters in length. ○ The password must contain at least three of the following character types: uppercase letters, lowercase letters, digits, and special characters. ○ Special characters include: ! @ # $ % ^ & * ( ) _ + - =
Confirm Password	Enter the password of the account again.
Description	Enter a description that helps identify the account. The description can be up to 256 characters in length.

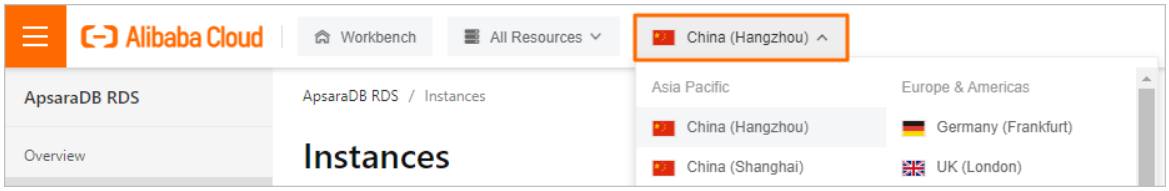
7. Click **OK**.

Reset the permissions of a privileged account

If the permissions of a privileged account are accidentally revoked or encounter other exceptions, perform the following steps to reset the permissions:

1. Log on to the ApsaraDB RDS console.
2. In the left-side navigation pane, click **Instances**. In the top navigation bar, select the region where

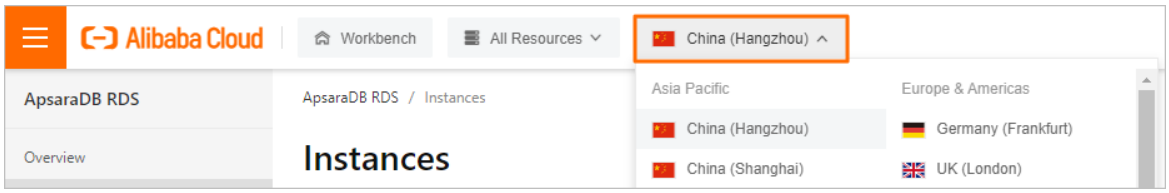
your RDS instance resides.



3. Find your RDS instance and click its ID.
4. In the left-side navigation pane, click **Accounts**.
5. Find the privileged account and click **Reset Permissions** in the **Actions** column.
6. In the dialog box that appears, specify a new password and click OK.

Create a standard account

1. Log on to the ApsaraDB RDS console.
2. In the left-side navigation pane, click **Instances**. In the top navigation bar, select the region where your RDS instance resides.



3. Find your RDS instance and click its ID.
4. In the left-side navigation pane, click **Accounts**.
5. Click **Create Account**.
6. In the Create Account panel, configure the following parameters.

Parameter	Description
Database Account	Enter the username of the account. The username must meet the following requirements: ○ The username must be 2 to 16 characters in length. ○ The username must start with a letter and end with a letter or digit. ○ The username can contain lowercase letters, digits, and underscores (_).
Account Type	Specify the type of the account. For this example, select **Standard Account**.

Parameter	Description
Authorized Databases	Specify the authorized databases of the account. You can specify one or more authorized databases. You can also leave this parameter empty. This is because you can grant the permissions on specific databases to the account after the account is created. i. Select one or more databases from the Unauthorized Databases section and click the **rightwards arrow** to add the selected databases to the Authorized Databases section. ii. In the Authorized Databases section, select the **Read/Write (DDL + DML)**, **Read-only**, **DDL Only**, or **DML Only** permissions for each authorized database. If you want to grant the same permissions on multiple authorized database at a time, select the authorized databases and click the button in the upper-right corner. For example, click **Set All to Read/Write (DDL + DML)**. **Note** The button in the upper-right corner changes after you click it. For example, after you click **Set All to Read/Write**, the button changes to **Set All to Read-only**.
Password	Enter the password of the account. The password must meet the following requirements: ○ The password must be 8 to 32 characters in length. ○ The password must contain at least three of the following character types: uppercase letters, lowercase letters, digits, and special characters. ○ Special characters include: ! @ # $ % ^ & * ( ) _ + - =
Confirm Password	Enter the password of the account again.
Description	Optional. Enter a description that helps identify the account. The description can be up to 256 characters in length.

7. Click **OK**.

Related operations

Operation	Description
CreateAccount	Creates an account that is used to manage the databases of an ApsaraDB RDS instance.

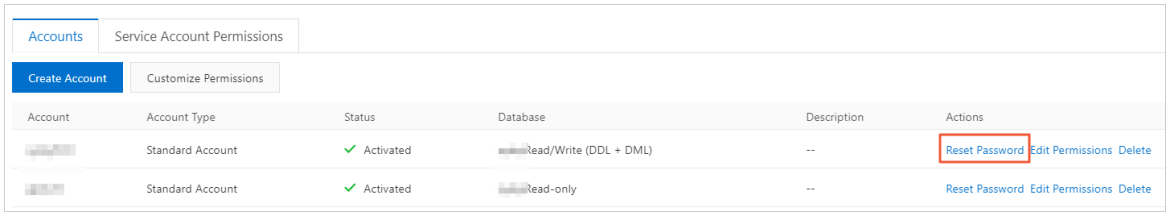
11.2. Reset the password of an account on an ApsaraDB RDS for MariaDB TX instance

This topic describes how to reset the password of an account on your ApsaraDB RDS for MariaDB TX instance. You can reset the password if the password is lost.

Procedure

> **Note** For data security purposes, we recommend that you change the password of each account on a regular basis.

1.
2. In the left-side navigation pane, click **Accounts**.
3. Find the account whose password you want to reset, and click **Reset Password** in the Actions column.



4. In the dialog box that appears, specify a new password, confirm the new password, and then click **Create**.

> **Note** The password must meet the following requirements:
>
> - The password must be 8 to 32 characters in length.
> - The password must contain at least three of the following character types: uppercase letters, lowercase letters, digits, and special characters.
> - The password can contain any of the following characters:
>
>   ! @ # $ % ^ & * ( ) _ + - =

Related operations

Operation	Description
ResetAccountPassword	Resets the password of an account on an ApsaraDB RDS instance.

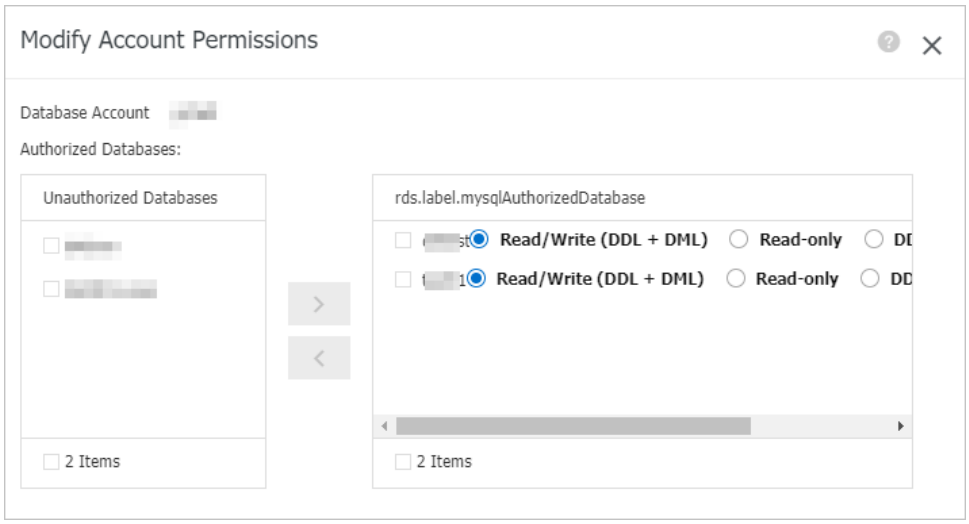
11.3. Modify the permissions of a standard account on an ApsaraDB RDS for MariaDB TX instance

This topic describes how to modify the permissions of a standard account on an ApsaraDB RDS for MariaDB TX instance. You can reset the permissions of a privileged account to the default settings. However, you cannot modify the permissions of a privileged account.

Procedure

1. Go to the **Accounts** page.
   i.
   ii. Find your RDS instance and click its ID. In the left-side navigation pane, click **Accounts**.
2. Find the standard account whose permissions you want to modify. Then, click **Edit Permissions** in the Actions column.
3. In the **Modify Account Permissions** panel, modify the permissions of the standard account.
   - If you want to add or remove an authorized database, select the database and click the **>** or **<** icon.
   - If you want to modify the permissions on an authorized database, select the database. Then, select the **Read/Write (DDL + DML)**, **Read-only**, **DDL Only**, or **DML Only** permissions in the **Authorized Databases** section.

   > **Note** You can use SQL statements to modify the permissions of a standard account at higher levels of granularity.



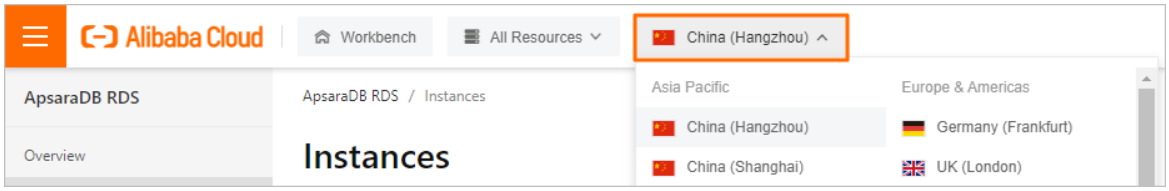
4. Click **OK**.

11.4. Delete an account for an RDS MariaDB instance

This topic describes how to delete an account from an RDS MariaDB 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 **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**.

APIs

API	Description
Delete an account	Used to delete an account for an RDS instance.

12.Database

12.1. Create a database on an ApsaraDB RDS for MariaDB TX instance

This topic describes how to create a database on an ApsaraDB RDS for MariaDB TX instance.

Prerequisites

Create an ApsaraDB RDS for MariaDB TX instance

Terms

- Instance: a virtualized database server, on which you can create and manage a number of databases.
- Database: a set of organized data that can be shared by a number of users. A database provides the minimal redundancy and is independent of applications. You can consider a database to be a warehouse that is used to store data.
- Character set: a collection of letters, special characters, and encoding rules that are used in a database.

Procedure

1.
2. In the left-side navigation pane, click **Databases**.
3. Click **Create Database**.
4. Configure the following parameters.

Parameter	Description
Database Name	◦ The name of the database must be 2 to 64 characters in length. ◦ The name of the database must start with a lowercase letter and end with a lowercase letter or digit. ◦ The name of the database can contain lowercase letters, digits, underscores (_), and hyphens (-). ◦ The name of the database must be unique within your RDS instance.
Supported Character Set	Specify the character set that is supported by the database. you can select utf8, gbk, latin1, or utf8mb4.

Parameter	Description
Authorized Account	Specify the account that is authorized to access the database. You can leave this parameter empty. In this case, you can specify the authorized account of the database after the database is created. For more information, see Modify the permissions of a standard account on an ApsaraDB RDS for MySQL instance. **Note** The **Authorized Account** drop-down list displays only the standard accounts that are created on your RDS instance. The privileged account has all permissions on all databases and does not require authorization.
Account type	Specify the permissions that you want to grant on the database. You can select the **Read/Write**, **Read-only**, **DDL Only**, or **DML Only** permissions. **Note** This parameter is available only when the **Authorized Account** parameter is set.
Description	Enter a description that is used to identify the database. The description can contain up to 256 characters.

5. Click **Create**.

What to do next

Connect to an ApsaraDB RDS for MariaDB TX instance.

Related operations

Operation	Description
CreateDatabase	Creates a database on an ApsaraDB RDS instance.

12.2. Delete a database from an ApsaraDB RDS for MariaDB TX instance

This topic describes how to delete a database from an ApsaraDB RDS for MariaDB TX instance. You can delete a database by using the ApsaraDB RDS console or an SQL statement.

Delete a database by using the ApsaraDB RDS console

1.
2. In the left-side navigation pane, click **Databases**.
3. In the **Actions** column click **Delete**.
4. In the message that appears, click **OK**.

Delete a database by using an SQL statement

1. Connect to your RDS instance by using a client. Note that you cannot connect to an ApsaraDB RDS for MariaDB TX instance by using Data Management (DMS). For more information, see Connect to an ApsaraDB RDS for MariaDB TX instance.
2. Execute the following statement to delete the database:

```
drop database <database name>;
```

Related operations

Operation	Description
DeleteDatabase	Deletes a database from an ApsaraDB RDS instance.

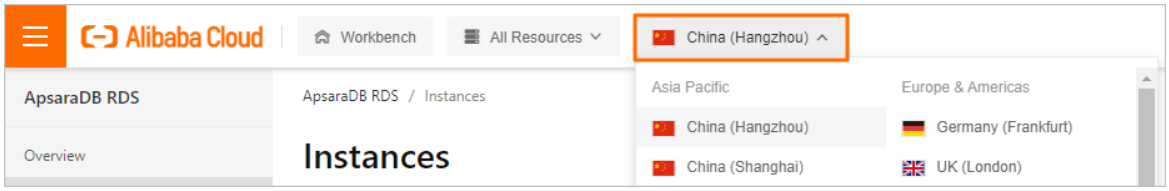
13.Monitoring and alerts

13.1. View the resource and engine metrics of an ApsaraDB RDS for MariaDB instance

This topic describes how to view the resource and engine metrics of an ApsaraDB RDS for MariaDB instance in the ApsaraDB RDS console.

Procedure

1. Log on to the ApsaraDB RDS console.
2. In the left-side navigation pane, click **Instances**. In the top navigation bar, select the region where your RDS instance resides.



3. Find your RDS instance and click its ID.
4. In the left-side navigation pane, click **Monitoring and Alerts**.
5. On the **Monitoring** tab, select the **Resource Monitoring** or **Engine Monitoring** type and specify a time range. Then, you can view the metrics that appear. The following table describes the metrics.

Monitoring type	Metric	Description
Resource Monitoring	IOPS	The numbers of input/output operations per second (IOPS) of the data and log disks.
	CPU Utilization and Memory Usage	The CPU utilization and memory usage of the RDS instance.
	Disk Space	The disk usage of the RDS instance. Unit: MB.
	Total Connections	The number of active connections to the RDS instance and the total number of connections to the RDS instance.
	Network Traffic	The volume of inbound traffic to the RDS instance per second and the volume of outbound traffic from the RDS instance per second. Unit: KB.
	Transactions per Second (TPS)/Queries per Second (QPS)	The average number of transactions per second (TPS) and the average number of SQL statements executed per second.

Monitoring type	Metric	Description
Engine Monitoring	InnoDB Buffer Pool Read Hit Ratio, Usage Ratio, and Dirty Block Ratio	The read hit ratio, dirty ratio, and usage of the InnoDB buffer pool. Unit: %.
	InnoDB Read/Write Volume	The volume of data read from InnoDB per second and the volume of data written to InnoDB per second. Unit: KB.
	InnoDB Buffer Pool Read/Write Frequency	The number of reads from InnoDB per second and the number of writes to InnoDB per second.
	InnoDB Log Read/Write/fsync	The number of physical writes to log files, number of log write requests, and number of writes that are completed by calling the fsync function to log files by InnoDB per second.
	Temporary Tables Automatically Created on Hard Disk when MySQL Statements Are Executed	The number of temporary tables that the RDS instance creates on the hard disk when executing SQL statements.
	MySQL_COMDML	The number of SQL statements that the RDS instance executes per second. ApsaraDB RDS supports the following SQL statements: ◦ INSERT ◦ DELETE ◦ INSERT_SELECT ◦ REPLACE ◦ REPLACE_SELECT ◦ SELECT ◦ UPDATE
	MySQL_RowDML	The number of operations that InnoDB performs per second, including: ◦ The number of physical writes to log files per second. ◦ The number of rows on which InnoDB performs operations per second. This includes the number of rows that are read from InnoDB tables per second, the number of rows that are updated in InnoDB tables per second, the number of rows that are deleted from InnoDB tables per second, and the number of rows that are inserted into InnoDB tables per second.
	MyISAM Read/Write Frequency	The number of reads from the buffer pool by MyISAM per second, the number of writes to the buffer pool by MyISAM per second, the number of reads from the hard disk by MyISAM per second, and the number of writes to the hard disk by MyISAM per second.

Monitoring 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. Configure alert rules for an ApsaraDB RDS for MariaDB TX instance

This topic describes how to configure alert rules for an ApsaraDB RDS for MariaDB TX instance. ApsaraDB for RDS offers the monitoring and alerting feature. If exceptions are detected in your RDS instance or if your RDS instance is locked due to low disk capacity, the system can send notifications to you.

Context

The monitoring and alerting feature of ApsaraDB RDS is implemented by using Cloud Monitor. Cloud Monitor allows you to configure metrics and alert rules. You can also associate alert groups with metrics. If a metric meets the conditions that are specified in an alert rule, alerts are sent as emails to all the contacts in the alert group that is associated with the metric.

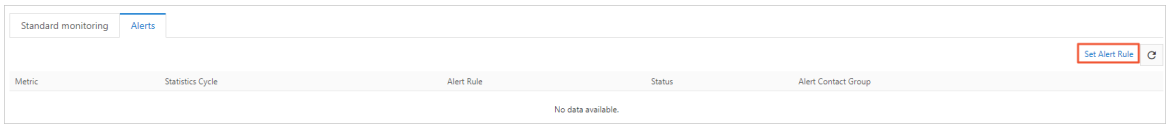
Enable the initiative alert feature

The initiative alert feature allows you to establish an alert system for multiple metrics in RDS. An alert notification is sent if an exception of a key metric occurs. You can then handle the exception at the earliest opportunity. For more information, see Enable the initiative alert feature.

1.
2. In the left-side navigation pane, click **Monitoring and Alerts**.
3. Click the **Alerts** tab.
4. In the right-side section of the page, turn on the **Initiative Alert** switch.

Create an alert rule

1.
2. In the left-side navigation pane, click **Monitoring and Alerts**.
3. Click the **Alerts** tab.
4. Click **Set Alert Rule** to go to the Cloud Monitor console.

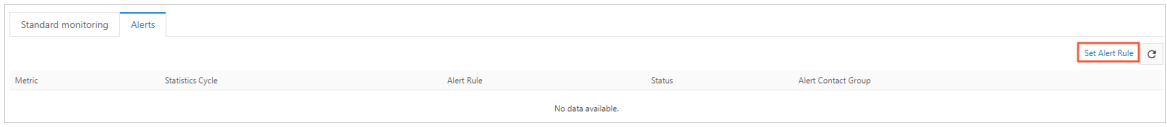


5. Create an alert group. For more information, see Create an alert contact or alert contact group.
6. Create an alert rule. For more information, see Create an alert rule.

> **Note** You can also configure Cloud Monitor to automatically monitor resources based on tags. For more information, see Monitor resources based on tags.

Manage an alert rule

1.
2. In the left-side navigation pane, click **Monitoring and Alerts**.
3. Click the **Alerts** tab.
4. Click **Set Alert Rule** to go to the Cloud Monitor console.



5. On the **Alert Rules** page, find the alert rule that you want to manage, and select one of the following operations in the Actions column:
   - View: View details about the alert rule.
   - Alert Logs: View the alerts that were triggered by the alert rule over a specific time range.
   - Modify: Modify the alert rule. For more information, see Create an alert rule.
   - Disable: Disable the alert rule. After you disable the alert rule, no alerts are triggered even if the metric meets the conditions that are specified in the alert rule.
   - Delete: Delete the alert rule. After you delete the alert rule, the alert rule cannot be restored. You can only re-create the alert rule if necessary.

14.Data security

14.1. Switch an ApsaraDB RDS for MariaDB TX instance to the enhanced whitelist mode

This topic describes how to switch an ApsaraDB RDS for MariaDB TX instance from the standard whitelist mode to the enhanced whitelist mode. The enhanced whitelist mode offers higher security than the standard whitelist mode.

> **Note** The enhanced whitelist mode is unavailable due to a network link upgrade. You will be immediately notified when the enhanced whitelist mode is available.

Network isolation modes

RDS instances support the following two network isolation modes:

- Standard whitelist

  IP addresses from both the classic network and virtual private clouds (VPCs) can be added to the same IP address whitelist. The standard whitelist mode is less secure than the enhanced whitelist mode. We recommend that you switch to the enhanced whitelist mode.

- Enhanced whitelist

  IP addresses from the classic network and VPCs must be added to different IP address whitelists. When you create an IP address whitelist, you must specify its network type.

Changes incurred

If the RDS instance resides in a VPC, an IP address whitelist of the VPC network type is created. The new IP address whitelist contains all the IP addresses from the original IP address whitelists.

> **Note** After you switch to the enhanced whitelist mode, the configured ECS security groups remain unchanged.

Precautions

- After you switch to the enhanced whitelist mode, you cannot roll the instance back to the standard whitelist mode.
- In enhanced whitelist mode, an IP address whitelist of the classic network type can also be used to allow access over the Internet. If you want to access your RDS instance from a host over the Internet, you must add the public IP address of the host to an IP address whitelist of the classic network type.

Procedure

1.
2. In the left-side navigation pane, click **Data Security**.

3. On the **Whitelist Settings** tab, click **Switch to Enhanced Whitelist (Recommended)**.
4. In the message that appears, click **Confirm**.

14.2. Configure a whitelist for an ApsaraDB RDS for MariaDB instance

This topic describes how to configure a whitelist for an RDS MariaDB instance.

Context

You can control access to your RDS instance by using one of the following methods:

- IP address whitelists

  An IP address whitelist contains the IP addresses of the devices that require access to your RDS instance. The IP address whitelist labeled default contains only the 127.0.0.1 IP address. This IP address indicates that no devices can access your RDS instance.

  Before you configure an IP address whitelist, you must confirm the network isolation mode of your RDS instance. The configuration procedure vary based on the network isolation mode.

  - Standard whitelist mode

    A standard IP address whitelist can contain the IP addresses from both the classic network and virtual private clouds (VPCs). However, the standard whitelist mode may incur security risks. For example, after you add an IP address from a VPC to a standard IP address whitelist, the IP address is granted access over both the VPC and the classic network. Therefore, we recommend that you switch your RDS instance to the enhanced whitelist mode. For more information, see Switch an ApsaraDB RDS for MariaDB TX instance to the enhanced whitelist mode.

    > **Note** RDS instances that run MariaDB TX can be deployed only in VPCs.

  - Enhanced whitelist mode

    An enhanced IP address whitelist can contain only the IP addresses from the classic network or from VPCs. When you create an enhanced IP address whitelist, you must specify its network type. If you add an IP address from a VPC to an enhanced IP address whitelist, the IP address is granted access only over the VPC.

- Security groups

  A security group serves as a virtual firewall to control the inbound and outbound traffic of the ECS instances in that security group. After you add a security group to your RDS instance, all the ECS instances in that security group can access your RDS instance.

  For more information about security groups, see Create a security group.

IP address whitelists help provide high security and efficient protection for your RDS instance. We recommend that you update the configured IP address whitelists on a regular basis. When you configure an IP address whitelist, the workloads on your RDS instance run as normal.

Precautions for configuring an IP address whitelist

- You can modify or clear the IP address whitelist labeled default. However, you cannot delete this IP address whitelist.

- A maximum of 50 IP address whitelists can be configured for each RDS instance.
- Up to 1,000 IP addresses and Classless Inter-Domain Routing (CIDR) blocks can be granted access to each RDS instance. If you want to add a large number of IP addresses, we recommend that you merge these IP addresses into CIDR blocks, such as 10.10.10.0/24, in which 24 indicates that the prefix of each IP address is 24-bit long. You can replace 24 with a value within the range of 1 to 32. For more information, see CIDR block FAQ.
- When you access an Alibaba Cloud service, the service automatically creates an IP address whitelist. The created IP address whitelist contains the IP address of the server that runs the service. For example, Data Management (DMS) creates an IP address whitelist named **ali_dms_group**, and Database Autonomy Service (DAS) creates an IP address whitelist named **hdm_security_ips**. To ensure that the specified Alibaba Cloud services can be used, do not modify or delete these IP address whitelists.

  Notice Do not add your IP address to these IP address whitelists. If you add your IP address to these IP address whitelists, your IP address may be overwritten by the entries that are updated from the existing IP addresses in these IP address whitelists. If your IP address is overwritten, your workloads are interrupted.

Configure an enhanced IP address whitelist

1.
2. In the left-side navigation pane, click **Data Security**.
3. Confirm the connection scenario and perform the required operations.

Connection scenario	Operation
Your ECS and RDS instances reside in the same VPC. This is the recommended connection scenario.	i. On the **Whitelist Settings** tab of the Data Security page, click **Modify** to the right of the IP address whitelist labeled **default Classic Network**. ii. In the dialog box that appears, enter the private IP address of your ECS instance in the IP Addresses field and click **OK**. **Note** The applications that run on your ECS instance connect to the internal endpoint of your RDS instance.

Connection scenario	Operation
Your ECS and RDS instances reside in different VPCs.	i. On the **Database Connection** page, click **Switch to Classic Network**. In the message that appears, click OK. ii. Click **Switch to VPC**. In the dialog box that appears, select the VPC of your ECS instance and click OK. **Note** Your ECS and RDS instances can reside in the same VPC only when they belong to the same region. If these instances belong to different regions, we recommend that you use Data Transmission Service (DTS) to migrate your RDS instance to the region of your ECS instance. For more information, see Migrate data between ApsaraDB RDS for MariaDB TX instances. iii. On the **Whitelist Settings** tab of the Data Security page, click **Modify** to the right of the IP address whitelist labeled **default VPC**. iv. In the dialog box that appears, enter the private IP address of your ECS instance in the IP Addresses field and click **OK**. **Note** The applications that run on your ECS instance connect to the internal endpoint of your RDS instance.
Your ECS instance resides in the classic network. Your RDS instance resides in a VPC.	i. Migrate your ECS instance to the VPC of your RDS instance. For more information, see Migrate an ECS instance from a classic network to a VPC. **Note** Your ECS and RDS instances can reside in the same VPC only when they belong to the same region. If these instances belong to different regions, we recommend that you use DTS to migrate your RDS instance to the region of your ECS instance. For more information, see Migrate data between ApsaraDB RDS for MariaDB TX instances. ii. On the **Whitelist Settings** tab of the Data Security page, click **Modify** to the right of the IP address whitelist labeled **default VPC**. iii. In the dialog box that appears, enter the private IP address of your ECS instance in the IP Addresses field and click **OK**. **Note** The applications that run on your ECS instance connect to the internal endpoint of your RDS instance.

Connection scenario	Operation
Your host that requires access to your RDS instance resides outside the cloud.	i. On the **Whitelist Settings** tab of the Data Security page, click **Modify** to the right of the IP address whitelist labeled **default Classic Network**. ii. In the dialog box that appears, enter the public IP address of the on-premises server in the IP Addresses field and click **OK**. **Note** ■ The applications that run on your host connect to the public endpoint of your RDS instance. ■ For more information about how to obtain the public IP address of your host, see Why am I unable to connect to my ApsaraDB RDS for MySQL or ApsaraDB RDS for MariaDB instance from a local server over the Internet?

Note

- On the Whitelist Settings tab of the Data Security page, you can click **Create Whitelist**. In the Create Whitelist dialog box, you can set the Network Type parameter to **VPC** or **Classic Network/Public IP**.
- If you enter more than one IP address or CIDR block, you must separate them with commas (,). Example: 192.168.0.1,172.16.213.9.
- If you click **Loading ECS Inner IP**, the IP addresses of all the ECS instances that are created within your Alibaba Cloud account appear. Then, you can select the IP addresses that you want to add to the IP address whitelist.

Configure a standard IP address whitelist

1.
2. In the left-side navigation pane, click **Data Security**.
3. On the **Whitelist Settings** tab of the page that appears, click **Modify** to the IP address whitelist labeled **default**.

   Note You can also click **Create Whitelist** to create an IP address whitelist.

4. In the **Edit Whitelist** dialog box, enter the IP addresses or CIDR blocks that require access to your RDS instance and click **OK**.

> **Note**
>
> - After you add IP addresses or CIDR blocks to the IP address whitelist labeled **default**, the default IP address 127.0.0.1 is automatically deleted from this IP address whitelist.
> - If you enter more than one IP address or CIDR block, you must separate them with commas (,). Do not add spaces preceding or following the commas. Example: `192.168.0.1,172.16.213.9`.
> - If you click **Loading ECS Inner IP**, the IP addresses of all the ECS instances that are created within your Alibaba Cloud account appear. Then, you can select the IP addresses that you want to add to the IP address whitelist.

Common errors

- Your RDS instance has only one IP address whitelist that contains only the default IP address 127.0.0.1 on the **Whitelist Settings** tab of the Data Security page.

  The default IP address 127.0.0.1 indicates that no devices can access your RDS instance. You must add the IP addresses of the devices that require access to your RDS instance to an IP address whitelist.

- An IP address whitelist contains only one entry, 0.0.0.0.

  If you want to grant access from all devices to your RDS instance, enter the 0.0.0.0/0 entry in an IP address whitelist.

  > **Note** The 0.0.0.0/0 entry indicates that all devices can access your RDS instance. Exercise caution when you add this entry.

- When you configure an enhanced IP address whitelist for your RDS instance, IP address errors are reported.

  Check that the enhanced whitelist mode is enabled. For more information, see Switch an ApsaraDB RDS for MariaDB TX instance to the enhanced whitelist mode.

  - If your RDS instance resides in a VPC and is connected by using the internal endpoint, make sure that the private IP address of your ECS instance is added to the IP address whitelist labeled **default VPC**.
  - If your RDS instance resides in the classic network and is connected by using the internal endpoint, make sure that the private IP address of your ECS instance is added to the IP address whitelist labeled **default Classic Network**.
  - If your RDS instance is connected over the Internet, make sure that the public IP address of your ECS instance is added to the IP address whitelist labeled **default Classic Network**. The IP address whitelist labeled default VPC cannot be used to control access over the Internet.

- The public IP addresses that you add to an IP address whitelist are not the actual egress IP addresses of the devices that you want to connect.

  This problem may occur due to the following reasons:

  - Public IP addresses dynamically change.
  - The tool or website that is used to query public IP addresses returns inaccurate results.

For more information, see Why am I unable to connect to my ApsaraDB RDS for MySQL or ApsaraDB RDS for MariaDB instance from a local server over the Internet?

Precautions for configuring a security group

- You can configure both IP address whitelists and security groups for your RDS instance. All the IP addresses in the configured IP address whitelists and all the ECS instances in the configured security groups are granted access to your RDS instance.
- A maximum of 10 security groups can be configured for each RDS instance.
- After the ECS instances in a configured security group are updated, the updates are automatically synchronized to that security group.
- You can configure only a security group that has the same network type as your RDS instance. The network types of your RDS instance and the security group that you want to configure must both be VPC or classic network.

> **Note** After you change the network type of your RDS instance, the security group that you have added becomes invalid. You must add the security group with the required network type again.

Configure a security group

1.
2. In the left-side navigation pane, click **Data Security**.
3. On the **Security Group** tab of the page that appears, click **Add Security Group**.

> **Note** Security groups whose names are followed by a VPC tag contain ECS instances that reside in VPCs.

FAQ

- After I configure an IP address whitelist, does the IP address whitelist immediately take effect?

  No, after you configure an IP address whitelist, the IP address whitelist requires about 1 minute to take effect.

- Why do I find IP address whitelists that I did not create?

  If these IP address whitelists contain private IP addresses, they are probably created by other Alibaba Cloud services, such as DMS and DAS. In this case, these IP address whitelists do not affect your business data, and no further actions are required.

- If I disable Internet access and enable only internal network access, is my RDS instance exposed to security risks?

  Yes, if you disable Internet access and enable only internal network access, your RDS instance is exposed to security risks. We recommend that you change the network type of your RDS instance to VPC. In this case, only an ECS instance in the same VPC can access your RDS instance after the required IP address is added to an IP address whitelist.

15.Backup

15.1. Automatically back up the data of an RDS MariaDB instance

This topic describes how to set backup policies for an RDS MariaDB instance. The system backs up the instance data according to the backup policies. MariaDB TX does not support manual backup.

Precautions

- The backup files occupy the backup space of the RDS instance. If the used backup space exceeds the quota of free backup space, additional fees are incurred. For more information, see View the free quota for backup storage of an ApsaraDB RDS for MariaDB TX instance.
- For information about the billing method and billable items, see Billable items, billing methods, and pricing.
- For information about the pricing of backup space, see ApsaraDB RDS MySQL pricing.
- 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.

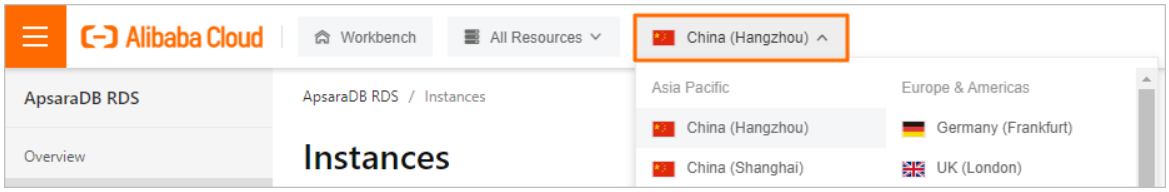
Overview

DB engine	Data backup	Log backup
MariaDB	Supports snapshot backup, but does not support physical backup or logical backup.	• Binary log files occupy the disk space of the RDS instance. • When the size of the existing binary log file reaches 500 MB or the duration of data write into the existing binary log file reaches 6 hours, the system starts to write data into a new binary log file. The earlier binary log file then is uploaded asynchronously. • You can uploaded binary log files to buckets in OSS. **Note** You cannot access the buckets for storing the uploaded binary log files in OSS.

Procedure

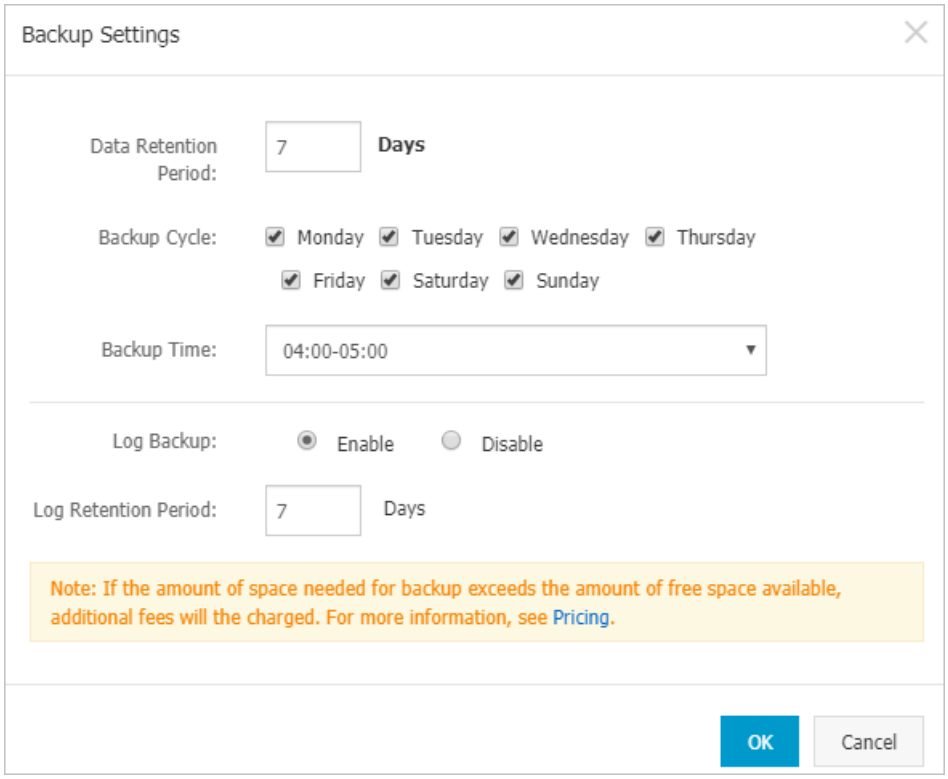
ApsaraDB for RDS can automatically back up databases according to the backup policy you set.

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 **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. 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.
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.



FAQ

1. Can I disable the data backup function for an RDS MariaDB TX instance?

   No, the data backup function must be enabled, and the backup file retention period ranges from 7 days to 730 days.

2. Can I disable the log backup function for an RDS MariaDB TX instance?

   Yes, you can disable the log backup function as needed.

APIs

API	Description
Create data backup	Used to create a backup file for an RDS instance.
Query the data backup files	Used to view the list of backup files for an RDS instance.
Query backup settings	Used to view the backup settings of an RDS instance.
Modify backup settings	Used to modify the backup settings of an RDS instance.
Query backup tasks	Used to obtain the list of backup tasks for an RDS instance.
Query log backup files	Used to view binlogs of an RDS instance.

15.2. View the free quota for backup storage of an ApsaraDB RDS for MariaDB TX instance

This topic describes how to view the free quota for backup storage of an ApsaraDB RDS for MariaDB TX instance. It also provides more information about how to calculate your excess backup storage usage. The free quota can vary based on the instance configuration.

Context

Backup files occupy backup storage. Each RDS instance is allocated with a free quota for backup storage. If the total size of backup files exceeds the free quota, additional fees are incurred.

Formula

RDS instances that are equipped with standard or enhanced SSDs support only snapshot backups. The free quota that is provisioned to store snapshot backup files is calculated based on the following formula: Free quota = 200% × Storage capacity purchased during instance creation (unit: GB; rounded up to the next integer).

The excess backup storage usage for which you must pay an hourly rate is calculated based on the following formula: Excess backup storage usage = Size of data backup files + Size of log backup files - Free quota.

If the size of data backup files is 150 GB, the size of log backup files is 50 GB, and the storage capacity purchased during instance creation is 60 GB, then the excess backup storage usage for which you must pay an hourly rate is 80 GB based on the following calculation: `Excess backup storage usage = 150 + 50 - 200% × 60 = 80 (GB)`. This indicates that you must pay for an additional 80 GB of backup storage per hour.

> **Note**
>
> - For more information about the hourly rate of excess backup storage usage, see the pricing information at ApsaraDB for RDS.
> - The RDS Basic Edition that is used with some database engines supports a free-of-charge retention period of seven days. For more information, see the ApsaraDB for RDS console.

Procedure

1.
2. In the **Usage Statistics** section at the lower part of the page, view the free quota for backup storage to the right of **Backup Size**.

   > **Note** The free quota for backup storage can vary based on the instance configuration.

FAQ

- Will backup files occupy the storage that I purchased during instance creation?

  No, backup files do not occupy the storage that you purchased during instance creation.

- Can I purchase subscription-billed backup storage?

  No, you cannot purchase backup storage.

15.3. Download the log backup files of an ApsaraDB RDS for MariaDB TX instance

This topic describes how to download the unencrypted log backup files of an ApsaraDB RDS for MariaDB TX instance. You can use these log backup files with snapshot backup files to restore the data of the RDS instance.

Limits

A Resource Access Management (RAM) user that has only the read permissions is not authorized to download backup files. You can grant the required permissions to the RAM user by using the RAM console.

Database engine	Download of data backup files	Download of log backup files
MariaDB	You cannot download the data backup files of your RDS instance. However, you can use the **restoration** feature to restore the data of your RDS instance to the same RDS instance or to a new RDS instance.	You can download the log backup files of your RDS instance.

Procedure

1. Go to the **Backup and Restoration** page.
   i.
   ii. Find your RDS instance and click its ID. In the left-side navigation pane, click **Backup and Restoration**.
2. On the **Log Backup** tab of the page that appears, select a time range, find the log backup file that you want to download, and then click **Download** in the **Actions** column.

   > **Note** If the log backup file is used to restore the data of your RDS instance to an on-premises database, you must make sure that the following requirements are met:
   >
   > - The instance ID of the log backup file must be the same as the instance No. of the data backup file that you select. The selected data backup file is used together with the log backup file to restore the data of your RDS instance.
   > - The start time of the log backup file must be later than the end time of the data backup file that you select and earlier than the point in time to which you want to restore the data of your RDS instance.

3. In the **Download Instance Backup Set** or **Download Binary Log** dialog box, click .

Download method	Description
	Use a browser to download the backup file.
	Copy the internal URL from which you can download the backup file. If your ECS and RDS instances reside in the same region, you can log on to the ECS instance and then use the internal URL to download the backup file. This method is faster and more secure than the other download methods.
	Copy the public URL to download the backup file. If you want to use other tools to download the backup file, select this download method.

Note In a Linux operating system, you can run the following command to download a log backup file:

```
wget -c '<Download URL of the log 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.

16.Restoration

16.1. Restore the data of an ApsaraDB RDS for MariaDB TX instance

This topic describes how to restore the data of an ApsaraDB RDS for MariaDB TX instance. If data backups are created for your RDS instance, you can restore the data of your RDS instance by using a data backup file.

The entire restoration process consists of the following steps:

1. Restore the data from a data backup file to a new RDS instance. This process was known as instance cloning.
2. Log on to the new RDS instance and verify the data.
3. Migrate the data to the original RDS instance.

Precautions

- The new RDS instance has the same IP address whitelists, backup settings, and parameter settings as the original RDS instance.
- The information about the data of the new RDS instance is the same as the information about the data in the data backup file that you select.
- The new RDS instance contains the account information in the data backup file that you select.

Billing

The restoration fee is the same as the fee that is required to purchase an RDS instance. For more information, visit the ApsaraDB RDS buy page.

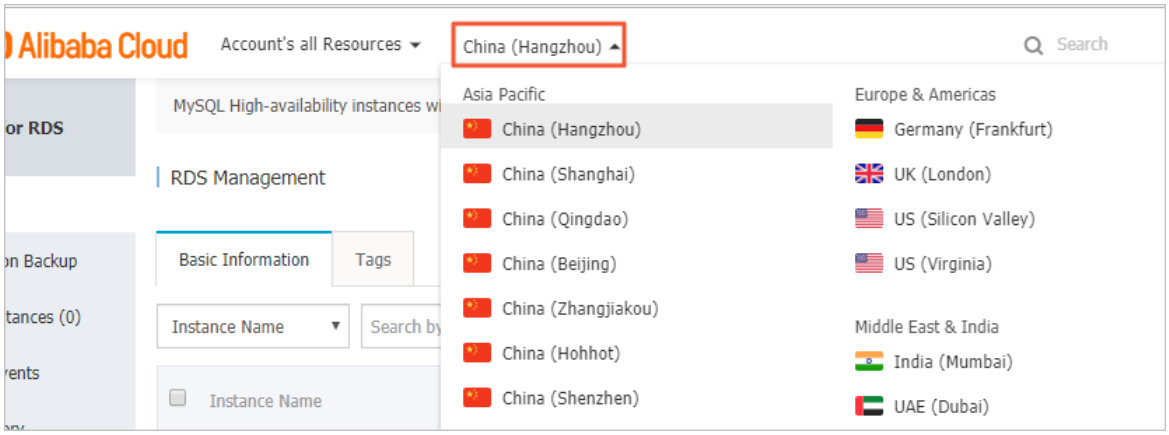
Prerequisites

The original RDS instance must meet the following requirements:

- The original RDS instance is in the Running state and is not locked.
- No migration tasks are being performed for the original RDS instance.
- At least one data backup file is available for the original RDS instance.

Restore the data to a new RDS instance

1. Log on to the ApsaraDB RDS console.
2. Select the region where the original RDS instance resides.



3. Find the original RDS instance and click its ID.
4. In the left-side navigation pane, click **Backup and Restoration**.
5. In the upper-right corner of the page that appears, click **Restore Database (Previously Clone Database)**.
6. Configure the following parameters.

Parameter	Description
Billing Method	○ **Subscription**: A subscription instance is an instance that you can subscribe to for a specified period of time and pay for up front. For long-term use, the subscription billing method is more cost-effective than the pay-as-you-go billing method. You can receive larger discounts for longer subscription periods. ○ **Pay-As-You-Go**: A pay-as-you-go instance is charged per hour based on your actual resource usage. We recommend that you select the pay-as-you-go billing method for short-term use. If you no longer need a pay-as-you-go instance, you can release it to reduce costs.
Restore Mode	The method that is used to restore data. Select **By Backup Set**.
Edition	The RDS edition of the new RDS instance. Select **High-availability**. In this edition, your database system consists of one primary RDS instance and one secondary RDS instance. These instances work in the high-availability architecture. **Note** The available RDS editions vary based on the region and database engine version that you select. For more information, see Overview of ApsaraDB RDS editions.
Zone of Primary Node	The zone of the new RDS instance. Each zone is an independent physical location within a region. Zones in the same region provide the same services. The multi-zone deployment method provides zone-level disaster recovery. If you select the single-zone deployment method, you need only to specify the Zone of Primary Node parameter. If you select the multi-zone deployment method, you need to specify the Zone of Primary Node parameter and ApsaraDB RDS automatically specifies the Zone of Secondary Node parameter.

Parameter	Description
Instance Type	○ **General-purpose (Entry-level)**: belongs to the general-purpose instance family. A general-purpose instance exclusively occupies the allocated memory and I/O resources. However, it shares CPU and storage resources with the other general-purpose instances that are deployed on the same server. ○ **Dedicated (Enterprise-level)**: belongs to the dedicated instance family or the dedicated host instance family. A dedicated instance exclusively occupies the allocated CPU, memory, storage, and I/O resources. The dedicated host instance family is the highest configuration of the dedicated instance family. A dedicated host instance occupies all the CPU, memory, storage, and I/O resources on the server where it is deployed. **Note** Each instance type supports a specific number of CPU cores, memory capacity, maximum number of connections, and maximum input/output operations per second (IOPS). For more information, see Primary ApsaraDB RDS instance types.
Capacity	The storage capacity that the new RDS instance has available to store data files, system files, binary log files, and transaction files. The storage capacity increases in increments of 5 GB. **Note** The dedicated instance family supports the exclusive allocation of resources. Therefore, the storage capacity of each instance type with local SSDs in this family is fixed. For more information, see Primary ApsaraDB RDS instance types.

7. Click **Next: Instance Configuration**.
8. Configure the following parameters.

Parameter	Description
Network Type	The network type of the new RDS instance. ApsaraDB RDS for MariaDB TX supports virtual private clouds (VPCs). A VPC is an isolated virtual network that provides higher security and higher performance than the classic network. **Note** You must make sure that an RDS instance has the same network type as the Elastic Compute Service (ECS) instance that you want to connect to the RDS instance. If the network types of these instances are VPC, you must also make sure that these instances reside in the same VPC. Otherwise, these instances cannot communicate over an internal network.
Resource Group	The resource group to which the new RDS instance belongs.

9. Click **Next: Confirm Order**.
10. Confirm the settings in the **Parameters** section, specify the **Purchase Plan** and **Duration** parameters, read and select Terms of Service, click **Pay Now**, and then complete the payment. You must specify the Duration parameter only when the new RDS instance uses the subscription billing method.

Verify the data on the new RDS instance

For more information, see Connect to an ApsaraDB RDS for MariaDB TX instance.

Migrate the data to the original RDS instance

After you verify the data on the new RDS instance, you can migrate the data from the new RDS instance to the original RDS instance. For more information, see Migrate data between RDS instances.

> **Note** During the migration process, your workloads on the original RDS instance run as normal.

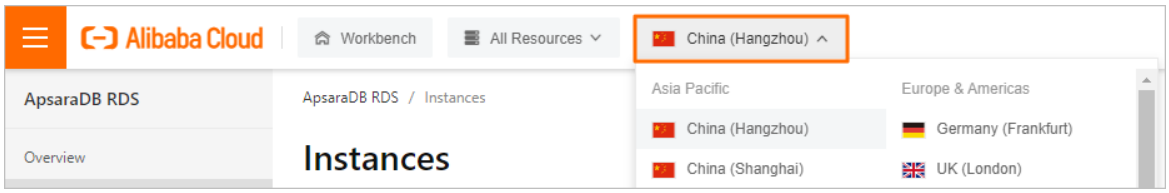
17.Manage logs

This topic describes how to manage the error logs and slow query logs of an ApsaraDB RDS MariaDB instance through the ApsaraDB for RDS console.

> **Note** For more information about how to manage binary logs, see Automatically back up the data of an RDS MariaDB instance and Download the log backup files of an ApsaraDB RDS for MariaDB TX instance.

Procedure

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



3. Find the target RDS instance and click its ID.
4. In the left-side navigation pane, click **Logs**.
5. On the **Logs** page that appears, click the Error Log, Slow Query Log or Slow Query Log Summary tab, select a time range, and click **Search**.

Tab	Description
Error Log	Records database running errors that occurred within the last month.
Slow Query Log	Records SQL statements that each took more than 1 second to run within the last month. Duplicate SQL statements are deleted. You can change this 1-second threshold by reconfiguring the long_query_time parameter. For more information, see Reconfigure parameters for an RDS MariaDB instance.
Slow Query Log Summary	Provides statistics and analysis reports on SQL statements that each took more than 1 second to run within the last month. You can change this 1-second threshold by reconfiguring the long_query_time parameter. For more information, see Reconfigure parameters for an RDS MariaDB instance.

18.View the event history of an ApsaraDB RDS instance

This topic describes how to view the operation and maintenance (O&M) events that are performed by users and Alibaba Cloud on an ApsaraDB RDS for SQL Server instance. These events include instance creation and parameter reconfiguration.

Billing

The event history feature is free of charge in the public preview phase, but starts to be charged after the public preview phase ends.

Scenarios

- Track instance management operations.
- Audit the security of instance management operations.
- Audit the compliance of the instance management operations that are performed by Alibaba Cloud. This applies to security-demanding sectors, such as finance and government affairs.

View the event history feature

1. Log on to the RDS management console , in the let-side navigation pane, click **Event Center**, and then select a region above.
2. Click the **Historical Events** tab.

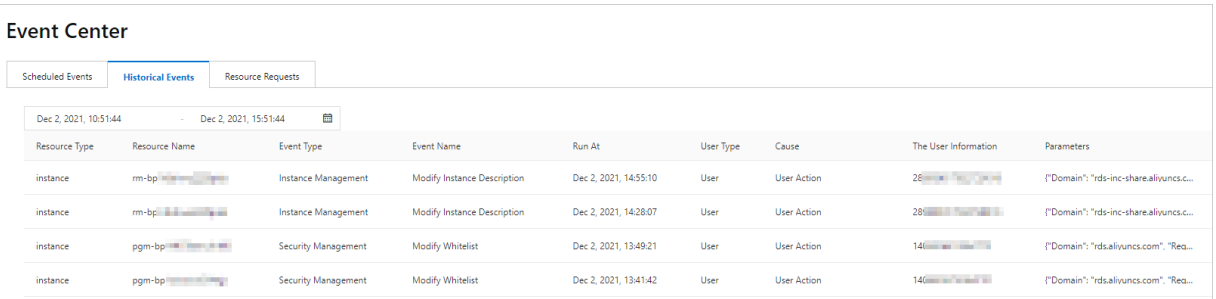
Introduction to the Historical Events page

The Historical Events page shows details about historical events in the selected region. These details include the resource types, resource names, and event types. The following table describes the parameters of a historical event.

Parameter	Description
Resource Type	The type of the RDS resource managed in the event. Only the **Instance** resource type is supported.
Resource Name	The name of the RDS resource managed in the event. If the value of the **Resource Type** parameter is **Instance**, the **Resource Name** column displays the ID of the involved RDS instance.
Event Type	The type of the event, for example, **Instance Management**, **Database Management**, **Read-write Splitting**, and **Network Management**. For more information, see Events.
Event Name	The operation executed in the event. For example, if the event type is **Instance Management**, supported operations include **Create Instance**, **Delete Instance**, **Change Specifications**, and **Restart Instance**. For more information, see Events.
Run At	The time when the event was executed.

Parameter	Description
User Type	The initiator of the event. Valid values: • User: initiates operations by using the ApsaraDB RDS console or the API. • System: initiates automatic O&M operations or periodic tasks. • O&M Administrator: initiates manual O&M operations.
Cause	The cause of the event. Valid values: • User Action: The event was initiated from a user by using the ApsaraDB RDS console or the API. • System Action or O&M Action: The event was initiated from the system or an O&M administrator.
The user information	The ID of the account that is used by a user to perform the event.
Parameters	The request parameters used by a user to initiate the event in the ApsaraDB RDS console.

> **Note**
> - The Historical Events page shows the historical events that were generated about 5 minutes earlier.
> - Historical Events are presented specific to regions. You can select a region in the top navigation bar and then view the historical events in the selected region.



Events

Event type	Operation
	Restart Instance (RestartDBInstance)
	Renew (RenewInstance)
	Change Specifications (ModifyDBInstanceSpec)
	Migrate Across Zones (MigrateToOtherZone)
	Shrink Log (PurgeDBInstanceLog)
	Upgrade Kernel Version (UpgradeDBInstanceEngineVersion)

Event type	Operation
Instance Management	Modify Instance Description (ModifyDBInstanceDescription)
	Modify Maintenance Window (ModifyDBInstanceMaintainTime)
	Create Read-only Instance (CreateReadOnlyDBInstance)
	Destroy Instance (DestroyDBInstance)
	Modify Upgrade Mode of Kernel Version (ModifyDBInstanceAutoUpgradeMinorVersion)
	Edit Parameters (ModifyParameter)
CloudDBA	Create Diagnostics Report (CreateDiagnosticReport)
Database Management	Create Database (CreateDatabase)
	Delete Database (DeleteDatabase)
	Modify Database Description (ModifyDBDescription)
	Replicate Database Between Instances (CopyDatabaseBetweenInstances)
	Modify System Collation and Time Zone (ModifyCollationTimeZone)
Read-write Splitting	Create Read-write Splitting Endpoint (AllocateReadWriteSplittingConnection)
	Query System-assigned Weight (CalculateDBInstanceWeight)
	Modify Read-write Splitting Policy (ModifyReadWriteSplittingConnection)
	Release Read-write Splitting Endpoint (ReleaseReadWriteSplittingConnection)
Security Management	Enable Enhanced Whitelist (MigrateSecurityIPMode)
	Enable SSL (ModifyDBInstanceSSL)
	Enable TDE (ModifyDBInstanceTDE)
	Modify Whitelist (ModifySecurityIps)
Account Management	Create Account (CreateAccount)
	Delete Account (DeleteAccount)
	Authorize Account to Access Database (GrantAccountPrivilege)
	Revoke Database Permissions from Account (RevokeAccountPrivilege)

Event type	Operation
	Modify Description of Database Account (ModifyAccountDescription)
	Reset Account Password (ResetAccountPassword)
	Reset Permissions of Superuser Account (ResetAccount)
High Availability (HA)	Trigger Switchover Between Primary and Secondary Instances (SwitchDBInstanceHA)
	Modify HA Mode (ModifyDBInstanceHAConfig)
Network Management	Apply for Public Endpoint (AllocateInstancePublicConnection)
	Modify Expiry Time of Endpoint (ModifyDBInstanceNetworkExpireTime)
	Modify Endpoint and Port (ModifyDBInstanceConnectionString)
	Switch Network Type (ModifyDBInstanceNetworkType)
	Release Public Endpoint (ReleaseInstancePublicConnection)
	Switch Between Internal and Public Endpoints (SwitchDBInstanceNetType)
Log Management	Enable/disable Log Audit (ModifySQLCollectorPolicy)
Backup Restoration	Create Data Backup (CreateBackup)
	Clone Instance (CloneDBInstance)
	Create Temporary Instance (CreateTempDBInstance)
	Modify Backup Policy (ModifyBackupPolicy)
	Restore Backup Set to Original Instance (RestoreDBInstance)
	Delete Data Backup (DeleteBackup)
	Restore Database (RecoveryDBInstance)
Cross-region Backup Restoration	Restore Data to New Instance Across Regions (CreateDdrInstance)
	Modify Cross-region Backup Settings (ModifyInstanceCrossBackupPolicy)
SQL Server Backup Migration to Cloud	Restore Backup File in OSS to RDS Instance (CreateMigrateTask)
	Make Database Available While Migrating Backup Data to Cloud (CreateOnlineDatabaseTask)
Monitoring	Set Monitoring Frequency (ModifyDBInstanceMonitor)

Event type	Operation
Data Migration	Create Upload Path for SQL Server (CreateUploadPathForSQLServer)
	Import Data from Other RDS (ImportDatabaseBetweenInstances)
	Cancel Migration Task (CancelImport)
Tag Management	Bind Tags to Instance (AddTagsToResource)
	Remove Tag (RemoveTagsFromResource)

Related operations

Operation	Description
Query historical events	Queries the events of an ApsaraDB RDS instance.
Query status of the event history feature	Queries the status of the historical events feature of an ApsaraDB RDS instance.
Enable or disable the event history feature	Enables or disables the historical events feature of an ApsaraDB RDS instance.

19.Tag

19.1. Create tags

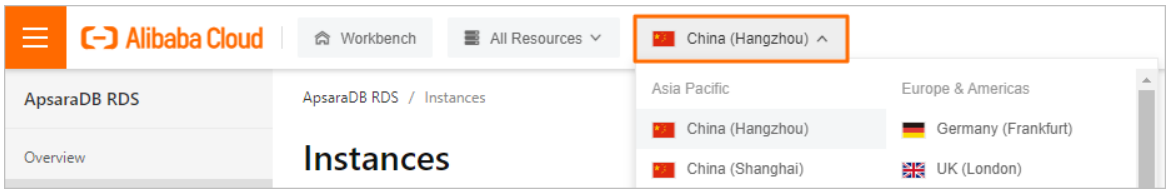
This topic describes how to create tags for one or more RDS instances. If you have a large number of RDS instances, you can create tags and then bind the tags to the instances so that you can classify and better manage the instances. Each tag consists of a key and a value.

Limits

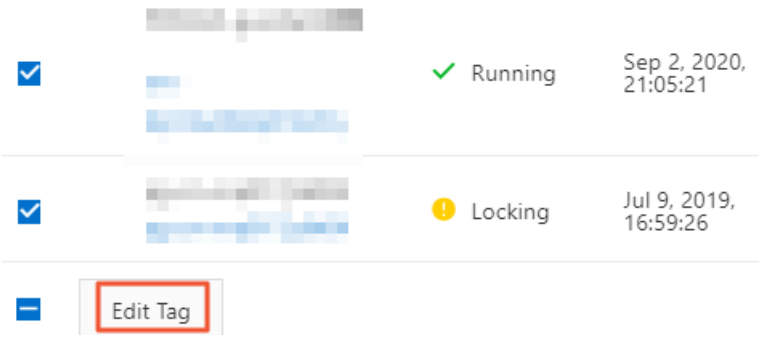
- Up to 10 tags can be bound to each RDS instance, and each tag must have a unique key. Tags with the same key are overwritten.
- You can bind up to five tags at a time.
- Tag information is independent in different regions.
- After you unbind a tag from an RDS instance, the tag is deleted if it is not bound to any other RDS instance.

Procedure

1. Log on to the ApsaraDB for RDS console. In the left-side navigation pane, click **Instances**. In the top navigation bar, select the region where your RDS instance resides.

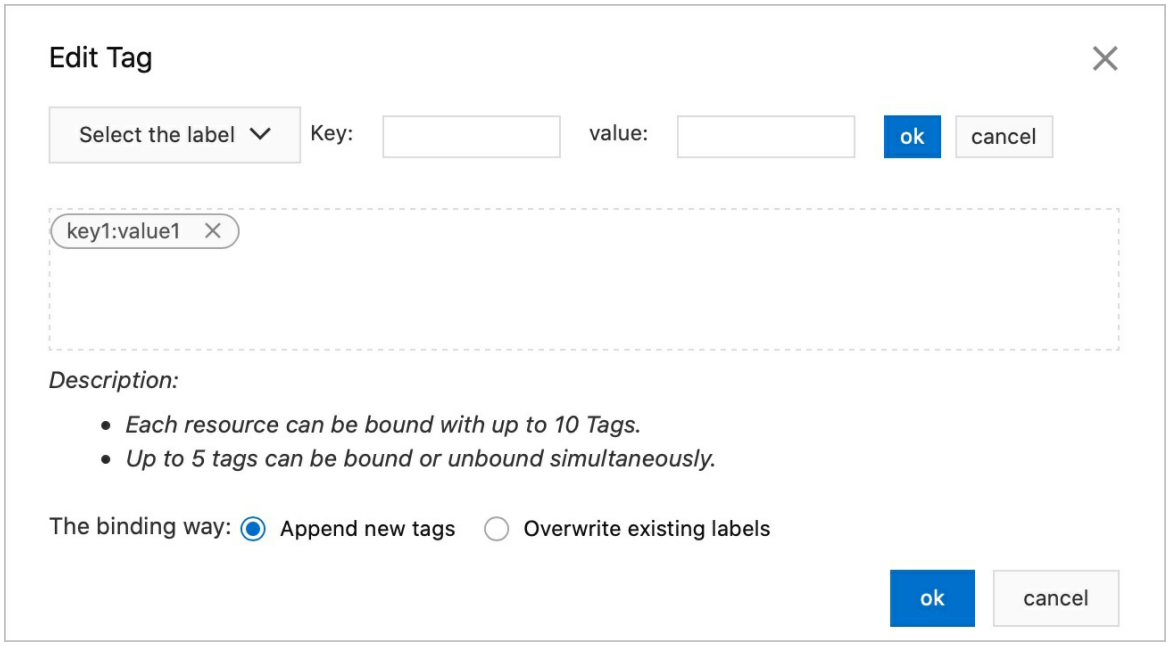


2. Specify the method of adding tags.
   - If you want to add tags to only one RDS instance, find the RDS instance and in the **Actions** column choose **More > Edit Tag**.
   - If you want to add tags to more than one RDS instance, select the RDS instances and click **Edit Tag**



3. 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.



4. After you add all the tags you need, click **Confirm**.

APIs

API	Description
AddTagsToResource	Used to bind a tag to RDS instances.

19.2. Unbind tags from an ApsaraDB RDS for MySQL instance

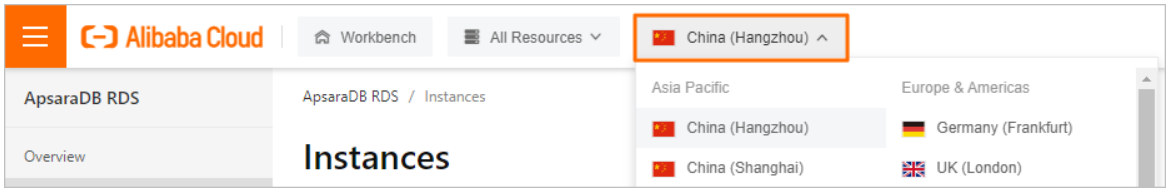
This topic describes how to unbind tags from an ApsaraDB RDS for MySQL instance. You may need to unbind tags due to instance configuration adjustments or if these tags are no longer needed.

Limits

- You can unbind a maximum of 20 tags at a time.
- After you unbind a tag from an RDS instance, ApsaraDB RDS checks whether the tag is bound to any other RDS instance. If the tag is not bound to any other RDS instance, ApsaraDB RDS deletes the tag.

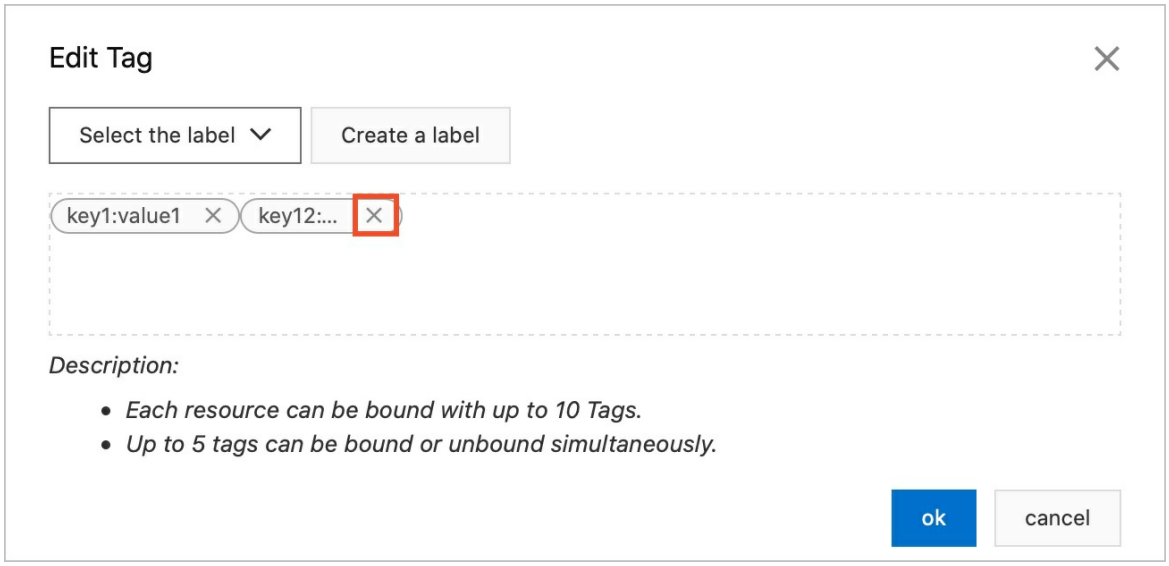
Procedure

1. Log on to the ApsaraDB for RDS console. In the left-side navigation pane, click **Instances**. In the top navigation bar, select the region where your RDS instance resides.



2. Find your RDS instance and in the Actions column choose **More > Edit Tag**.

3. In the dialog box that appears, click the **X** icon next to each tag that you want to unbind.



4. Click **OK**.

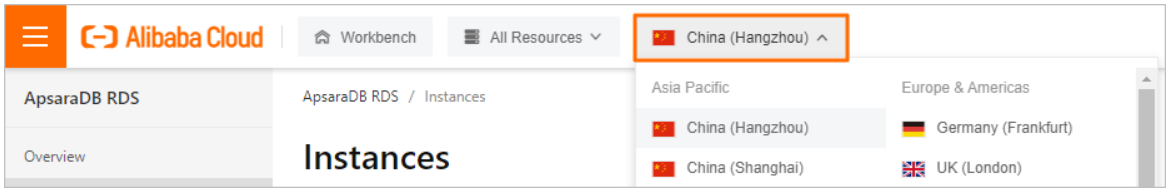
Related operations

Operation	Description
Unbind tags	Unbinds tags from one or more ApsaraDB RDS instances.

19.3. Use tags to filter ApsaraDB RDS for MySQL instances

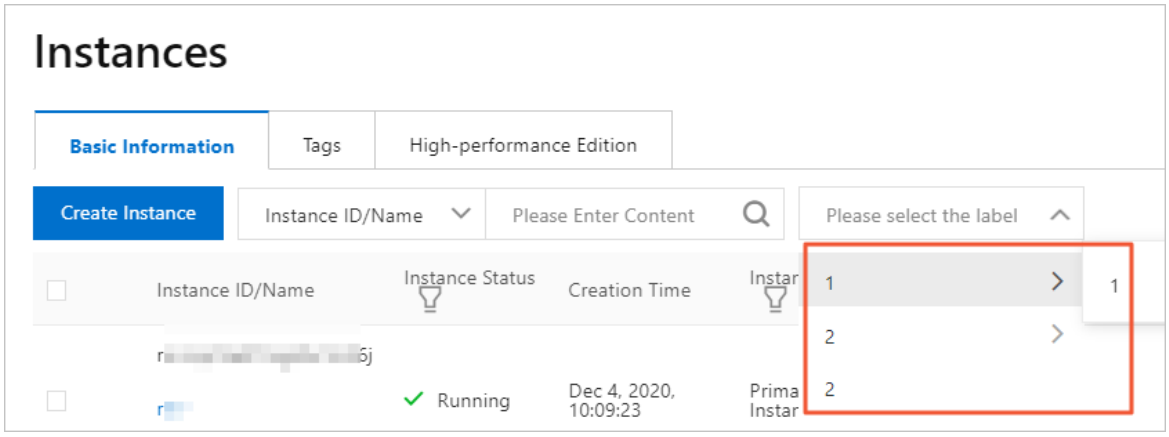
This topic describes how to filter ApsaraDB RDS for MySQL instances based on the tags that are bound to these instances.

1. Log on to the ApsaraDB for RDS console. In the left-side navigation pane, click **Instances**. In the top navigation bar, select the region where your RDS instance resides.



2. On the **Basic Information** tab, specify the **keys** and **values** of the tags based on which you want to filter RDS instances.

> **Note** To delete the filter condition that is specified by a tag, you can click the X icon next to the tag.



Related operations

Operation	Description
Query the tags of ApsaraDB RDS instances	Queries the tags that are bound to one or more ApsaraDB RDS instances.