ALIBABA CLOUD

Alibaba Cloud

ApsaraDB for PolarDB

User Guide

Document Version: 20220712

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.Overview

is a next-generation independently developed by Alibaba Group. decouples computing from storage and uses integrated software and hardware. provides a database service solution that enables auto scaling, high performance, mass storage, and high security and reliability. is fully compatible with MySQL 5.6, MySQL 5.7, MySQL 8.0, and PostgreSQL 11. is also highly compatible with Oracle.

uses an architecture that decouples computing from storage. All compute nodes share one set of data. allows you to upgrade or downgrade specifications in minutes, and supports fault recovery in seconds. ensures global data consistency, and offers data backup and disaster recovery. The feature of data backup and disaster recovery is free of charge. combines the benefits of commercial databases with the benefits of open source cloud databases. The benefits of commercial databases include stability, reliability, high performance, and scalability. The benefits of open source cloud databases include ease of use, openness, and self-iteration. is fully compatible with native MySQL databases and ApsaraDB RDS for MySQL databases. You can migrate data from a MySQL database to a database of the without the need to modify the code or configuration of your applications.

Terms

- cluster

  A cluster of the contains one primary node and a maximum of 15 read-only nodes. A minimum of one read-only node is required to provide high availability in active-active mode. If a cluster ID starts with `pc`, it indicates that the cluster is a cluster.

- node

  A node is a database service process that exclusively occupies physical memory. If a node ID starts with `pi`, it indicates that the node is a instance.

- database

  A database is a logical unit that is created on a node. You can create multiple databases on a node. The name of each database on the node must be unique.

- region and zone

  A region is a geographic area where a data center resides. A zone is a geographic area in a region. This area has an independent power supply and network. For more information, see Alibaba Cloud's Global Infrastructure.

Console

Alibaba Cloud offers an easy-to-use web console to help you manage various Alibaba Cloud services and products. This includes the cloud database service . In the console, you can create, connect to, and configure databases.

Log on to the console: PolarDB console

2.Feature list

This topic describes the features that are supported by different editions and versions of .

MySQL 8.0

Type	Feature	MySQL 8.0			
Data migration	Overview of data migration or synchronization solutions	Supported	Supported	Supported	Supported
Cluster management	Purchase a pay-as-you-go cluster and Purchase a subscription cluster	Supported	Not supported	Supported	Supported
	Purchase a storage plan	Supported	Supported	Supported	Supported
	Change the billing method from subscription to pay-as-you-go	Supported	Supported	Supported	Supported
	Change the billing method of a cluster from pay-as-you-go to subscription	Supported	Supported	Supported	Supported
	Manual renewal	Supported	Supported	Supported	Supported
	Auto-renewal	Supported	Supported	Supported	Supported
	Release a cluster	Supported	Supported	Supported	Supported
	Specify cluster and node parameters	Supported	Supported	Supported	Supported
	Configure an IP whitelist	Supported	Supported	Supported	Supported
	Clone a cluster	Supported	Supported	Supported	Supported
	Bind a tag	Supported	Supported	Supported	Supported
	Filter clusters by tag	Supported	Supported	Supported	Supported
	View tags bound to a cluster	Supported	Supported	Supported	Supported
	Unbind a tag	Supported	Supported	Supported	Supported
	Register and log on to an Alibaba Cloud account	Supported	Supported	Supported	Supported
	Create and authorize a RAM user	Supported	Supported	Supported	Supported

Account Type	Feature	MySQL 8.0			
	Create a database account	Supported	Supported	Supported	Supported
	Manage database accounts for a cluster	Supported	Supported	Supported	Supported
Database	Database Management	Supported	Supported	Supported	Supported
PolarProxy	Apply for a cluster endpoint or a primary endpoint	Supported	Supported	Supported	Supported
	Connect to a cluster	Supported	Supported	Supported	Supported
	Private domain names	Supported	Supported	Supported	Supported
	Read/write splitting	Supported	Not supported	Supported	Not supported
	Dynamic data masking	Supported	Not supported	Supported	Not supported
	Configure PolarProxy	Supported	Not supported	Supported	Not supported
Cluster configuration changes	Manually upgrade or downgrade a PolarDB cluster	Supported	Not supported	Supported	Not supported
	Automatic configuration changes (auto scaling)	Supported	Not supported	Supported	Not supported
	Automatically scale local resources	Supported	Not supported	Not supported	Not supported
	Add or remove read-only nodes	Supported	Not supported	Supported	Not supported
	Perform a temporary cluster upgrade	Supported	Not supported	Not supported	Not supported
	Upgrade an Archive Database Standalone Edition cluster to an Archive Database Cluster Edition cluster	Not supported	Supported	Not supported	Not supported
High-availability deployment	Multi-zone deployment and Change the primary zone and vSwitch of a cluster	Supported	Not supported	Not supported	Not supported

deployment architecture Type	Feature	MySQL 8.0			
	Multi-node deployment and Automatic failover and manual failover	Supported	Not supported	Supported	Not supported
Global database network (GDN)	Create and release a GDN	Supported	Not supported	Not supported	Not supported
	Add and remove secondary clusters	Supported	Not supported	Not supported	Not supported
	Connect to a GDN	Supported	Not supported	Not supported	Not supported
Data security	Configure TDE	Supported	Not supported	Not supported	Supported
	Configure SSL encryption	Supported	Supported	Supported	Supported
Backup and restoration	Automatic backup	Supported	Supported	Supported	Supported
	Manual backup	Supported	Supported	Supported	Supported
	Cluster restoration: Restore data from a backup set	Supported	Supported	Supported	Supported
	Cluster restoration: Restore data to a previous point in time	Supported	Supported	Supported	Supported
	Database or table restoration: Restore data from a backup set	Supported	Not supported	Not supported	Not supported
	Database or table restoration: Restore data to a previous point in time	Supported	Not supported	Not supported	Not supported
Cluster recycle bin	Restore a released cluster	Supported	Supported	Supported	Supported
	Delete a released cluster	Supported	Supported	Supported	Supported
	SQL Explorer	Supported	Supported	Not supported	Supported
	View performance monitoring data	Supported	Supported	Supported	Supported
	Slow SQL query	Supported	Supported	Not supported	Not supported

Type	Feature	MySQL 8.0			
Diagnosis and optimization	Autonomy center	Supported	Supported	Not supported	Not supported
	Session Management	Supported	Supported	Not supported	Supported
	Real-time Monitoring	Supported	Supported	Not supported	Supported
	Storage analysis	Supported	Supported	Not supported	Supported
	Deadlock analysis	Supported	Supported	Not supported	Supported
	Diagnostic reports	Supported	Supported	Not supported	Supported
	Performance Insight	Supported	Supported	Not supported	Not supported
Database engine	Version Management	Supported	Supported	Supported	Supported
	Enable binary logging	Supported	Not supported	Not supported	Supported
	Use ROLLUP to improve performance	Supported	Supported	Supported	Supported
	Recycle bin	Supported	Supported	Supported	Supported
	Fast query cache	Supported	Supported	Supported	Supported
	Parallel query	Supported	Not supported	Not supported	Supported
	Statement Outline	Supported	Supported	Supported	Supported
	Hot row optimization	Supported	Supported	Supported	Supported
	Performance Agent	Supported	Supported	Supported	Supported
Other features	Set a maintenance window	Supported	Supported	Supported	Supported
	View and manage scheduled events	Supported	Supported	Supported	Supported
	Restart nodes	Supported	Supported	Supported	Supported

MySQL 5.7

Type	Feature	MySQL 5.7	
Data migration	Overview of data migration or synchronization solutions	Supported	Supported
Cluster management	Purchase a pay-as-you-go cluster and Purchase a subscription cluster	Supported	Supported
	Purchase a storage plan	Supported	Supported
	Change the billing method from subscription to pay-as-you-go	Supported	Supported
	Change the billing method of a cluster from pay-as-you-go to subscription	Supported	Supported
	Manual renewal	Supported	Supported
	Auto-renewal	Supported	Supported
	Release a cluster	Supported	Supported
	Specify cluster and node parameters	Supported	Supported
	Configure an IP whitelist	Supported	Supported
	Clone a cluster	Supported	Supported
	Bind a tag	Supported	Supported
	Filter clusters by tag	Supported	Supported
	View tags bound to a cluster	Supported	Supported
	Unbind a tag	Supported	Supported
Account	Register and log on to an Alibaba Cloud account	Supported	Supported
	Create and authorize a RAM user	Supported	Supported
	Create a database account	Supported	Supported
	Manage database accounts for a cluster	Supported	Supported
Database	Database Management	Supported	Supported
	Apply for a cluster endpoint or a primary endpoint	Supported	Supported
	Connect to a cluster	Supported	Supported
	Private domain names	Supported	Supported

PolarProxy Type	Feature	MySQL 5.7	
	Read/write splitting	Supported	Not supported
	Dynamic data masking	Supported	Not supported
	Configure PolarProxy	Supported	Not supported
Auto scaling	Manually upgrade or downgrade a PolarDB cluster	Supported	Not supported
	Automatic configuration changes (auto scaling)	Supported	Not supported
	Automatically scale local resources	Supported	Not supported
	Add or remove read-only nodes	Supported	Not supported
	Perform a temporary cluster upgrade	Supported	Not supported
High-availability deployment architecture	Multi-zone deployment and Change the primary zone and vSwitch of a cluster	Supported	Not supported
	Multi-node deployment and Automatic failover and manual failover	Supported	Not supported
GDN	Create and release a GDN	Not supported	Not supported
	Add and remove secondary clusters	Not supported	Not supported
	Connect to a GDN	Not supported	Not supported
Data security	Configure TDE	Supported	Supported
	Configure SSL encryption	Supported	Supported
Backup and restoration	Automatic backup	Supported	Supported
	Manual backup	Supported	Supported
	Cluster restoration: Restore data from a backup set	Supported	Supported
	Cluster restoration: Restore data to a previous point in time	Supported	Supported
	Database or table restoration: Restore data from a backup set	Supported	Not supported
	Database or table restoration: Restore data to a previous point in time	Supported	Not supported
	Restore a released cluster	Supported	Supported

Cluster recycle bin Type	Feature	MySQL 5.7	
	Delete a released cluster	Supported	Supported
Diagnosis and optimization	SQL Explorer	Supported	Supported
	View performance monitoring data	Supported	Supported
	Slow SQL query	Supported	Not supported
	Autonomy center	Supported	Not supported
	Session Management	Supported	Supported
	Real-time Monitoring	Supported	Supported
	Storage analysis	Supported	Supported
	Deadlock analysis	Supported	Supported
	Diagnostic reports	Supported	Supported
	Performance Insight	Supported	Not supported
Database engine	Version Management	Supported	Supported
	Enable binary logging	Supported	Supported
	Use ROLLUP to improve performance	Not supported	Not supported
	Recycle bin	Not supported	Not supported
	Fast query cache	Supported	Not supported
	Parallel query	Not supported	Not supported
	Statement Outline	Supported	Supported
	Hot row optimization	Not supported	Not supported
	Performance data monitoring	Not supported	Not supported
Other features	Set a maintenance window	Supported	Supported
	View and manage scheduled events	Supported	Supported
	Restart nodes	Supported	Supported

MySQL 5.6

Type	Feature	MySQL 5.6	
Data migration	Overview of data migration or synchronization solutions	Supported	Supported
Cluster management	Purchase a pay-as-you-go cluster and Purchase a subscription cluster	Supported	Supported
	Purchase a storage plan	Supported	Supported
	Change the billing method from subscription to pay-as-you-go	Supported	Supported
	Change the billing method of a cluster from pay-as-you-go to subscription	Supported	Supported
	Manual renewal	Supported	Supported
	Auto-renewal	Supported	Supported
	Release a cluster	Supported	Supported
	Specify cluster and node parameters	Supported	Supported
	Configure an IP whitelist	Supported	Supported
	Clone a cluster	Supported	Supported
	Bind a tag	Supported	Supported
	Filter clusters by tag	Supported	Supported
	View tags bound to a cluster	Supported	Supported
	Unbind a tag	Supported	Supported
Account	Register and log on to an Alibaba Cloud account	Supported	Supported
	Create and authorize a RAM user	Supported	Supported
	Create a database account	Supported	Supported
	Manage database accounts for a cluster	Supported	Supported
Database	Database Management	Supported	Supported
	Apply for a cluster endpoint or a primary endpoint	Supported	Supported
	Connect to a cluster	Supported	Supported
	Private domain names	Supported	Supported

PolarProxy Type	Feature	MySQL 5.6	
	Read/write splitting	Supported	Not supported
	Dynamic data masking	Supported	Not supported
	Configure PolarProxy	Supported	Not supported
Auto scaling	Manually upgrade or downgrade a PolarDB cluster	Supported	Not supported
	Automatic configuration changes (auto scaling)	Supported	Not supported
	Automatically scale local resources	Supported	Not supported
	Add or remove read-only nodes	Supported	Not supported
	Perform a temporary cluster upgrade	Supported	Not supported
High-availability deployment architecture	Multi-zone deployment and Change the primary zone and vSwitch of a cluster	Supported	Not supported
	Multi-node deployment and Automatic failover and manual failover	Supported	Not supported
GDN	Create and release a GDN	Supported	Not supported
	Add and remove secondary clusters	Supported	Not supported
	Connect to a GDN	Supported	Not supported
Data security	Configure TDE	Supported	Supported
	Configure SSL encryption	Supported	Supported
Backup and restoration	Automatic backup	Supported	Supported
	Manual backup	Supported	Supported
	Cluster restoration: Restore data from a backup set	Supported	Supported
	Cluster restoration: Restore data to a previous point in time	Supported	Supported
	Database or table restoration: Restore data from a backup set	Supported	Not supported
	Database or table restoration: Restore data to a previous point in time	Supported	Not supported
	Restore a released cluster	Supported	Supported

Type	Feature	MySQL 5.6	
Cluster recycle bin			
	Delete a released cluster	Supported	Supported
Diagnosis and optimization	SQL Explorer	Supported	Supported
	View performance monitoring data	Supported	Supported
	Slow SQL query	Supported	Not supported
	Autonomy center	Supported	Not supported
	Session Management	Supported	Supported
	Real-time Monitoring	Supported	Supported
	Storage analysis	Supported	Supported
	Deadlock analysis	Supported	Supported
	Diagnostic reports	Supported	Supported
	Performance Insight	Supported	Not supported
Database engine	Version Management	Supported	Supported
	Enable binary logging	Supported	Supported
	Use ROLLUP to improve performance	Not supported	Not supported
	Recycle bin	Not supported	Not supported
	Fast query cache	Supported	Not supported
	Parallel query	Not supported	Not supported
	Statement Outline	Supported	Supported
	Hot row optimization	Supported	Supported
	Performance data monitoring	Not supported	Not supported
Other features	Set a maintenance window	Supported	Supported
	View and manage scheduled events	Supported	Supported
	Restart nodes	Supported	Supported

3.Data Migration or Synchronization

3.1. Overview of data migration or synchronization solutions

provides multiple data migration and synchronization solutions to meet different business requirements, such as migrating data to the cloud, migrating data between different cloud service providers, and synchronizing data. This allows you to migrate and synchronize databases to Alibaba Cloud. This does not affect your business.

Data Transmission Service (DTS) of Alibaba Cloud supports schema migration, full migration, and real-time data synchronization for . For more information about DTS, see DTS.

Data migration

Use scenario	Document link
Migrate data from ApsaraDB RDS for MySQL to	• Create a PolarDB for MySQL cluster by migrating an ApsaraDB RDS for MySQL instance (recommended for smooth migration) • Create a PolarDB for MySQL cluster by cloning an ApsaraDB RDS for MySQL instance • Migrate data from an ApsaraDB RDS for MySQL instance to a PolarDB for MySQL cluster
Migrate data from to ApsaraDB RDS for MySQL	Migrate data from a PolarDB for MySQL cluster to an ApsaraDB RDS for MySQL instance
Migrate data from a user-created database to	Migrate data from a user-created MySQL database to a PolarDB for MySQL cluster
Migrate data from a third-party cloud database to	Migrate data from an Amazon Aurora MySQL cluster to a PolarDB for MySQL cluster
Migrate data between clusters	Migrate data between PolarDB for MySQL clusters

Data synchronization

Use scenario	Document link
Synchronize data from ApsaraDB RDS for MySQL to	Synchronize data from an ApsaraDB RDS for MySQL instance to an Apsara PolarDB for MySQL cluster
Synchronize data between clusters	• Configure one-way data synchronization between PolarDB for MySQL clusters • Configure two-way data synchronization between PolarDB for MySQL clusters

Use scenario	Document link
Synchronize data from a user-created database to	Synchronize data from a self-managed MySQL database hosted on ECS to a PolarDB for MySQL cluster
Synchronize data from to ApsaraDB RDS for MySQL	Synchronize data from an Apsara PolarDB for MySQL cluster to an ApsaraDB RDS for MySQL instance
Synchronize data from to an analytical database	• Synchronize data from a PolarDB for MySQL cluster to an AnalyticDB for MySQL cluster • Synchronize data from a PolarDB for MySQL cluster to an AnalyticDB for PostgreSQL instance
Synchronize data from to DataHub	Synchronize data from an Apsara PolarDB for MySQL cluster to a DataHub instance
Synchronize data from to Kafka	Synchronize data from a PolarDB for MySQL cluster to a self-managed Kafka cluster

3.2. Considerations for data migration from MySQL 5.7 to PolarDB for MySQL 8.0

8.0 is fully compatible with MySQL 5.7. You can migrate data from MySQL 5.7 to 8.0. This does not cause data loss. However, you must pay attention to the compatibility between the MySQL client version and 8.0.

For more information about how to migrate data from MySQL 5.7 to 8.0, see the following topics:

- Migrate data from an ApsaraDB RDS for MySQL instance to a PolarDB for MySQL cluster
- Migrate data from an Amazon Aurora MySQL cluster to a PolarDB for MySQL cluster
- Migrate data between PolarDB for MySQL clusters
- Migrate data from a user-created MySQL database to a PolarDB for MySQL cluster

Client versions

You must upgrade your MySQL client to the following versions:

- Java: MySQL Connector/J 8.0 or later.
- ODBC: MySQL Connector/ODBC 8.0 or later.
- CPP: MySQL Connector/CPP 8.0 or later.
- .NET: MySQL Connector/NET 8.0 or later.
- Nodejs: MySQL Connector/Nodejs 8.0 or later.
- Python: MySQL Connector/Python 8.0 or later.
- Python: mysql-connector-Python 8.0.5 or later.
- Golang: go-sql-driver/mysql 1.4.0 or later.
- PHP: mysqlnd 7.4 or later.

- C/CPP: libmysqlclient 8.0 or later.

Known issues of the client

- Issue: An exception occurs when the client connects to the MySQL database. The `query_cache_size` parameter cannot be identified.
  - Driver version: mysql-connector-java:5.1.42
  - Database version: mysql 8.0.13
  - Solution: Use mysql-connector-java:5.1.42 or later. For more information about the updates of the version, see Changes in MySQL Connector/J 5.1.43.
- Issue: The flag of `COM_STMT_EXECUTE` is invalid and the `COM_STMT_FETCH` statement is not executed to retrieve the result set. As a result, the Python driver for MySQL 8.0 fails to return the result. However, the result is returned as normal in MySQL 5.6 or MySQL 5.7.
  - Driver version: mysql-connector-2.2.9.
  - Database version: mysql 8.0.13.
  - Solution: Install the Python driver for MySQL 8.0. For more information, see Python driver.
- Issue: When you execute SQL statements with `kickout` in 8.0 clusters, the following error is returned:

  ```
  ERROR 1064 (42000): You have an error in your SQL syntax;
  ```

  - Solution: We recommend that you upgrade the 8.0 cluster to the latest revision. For more information, see Version Management.

3.3. Migrate data from ApsaraDB RDS to PolarDB

3.3.1. Create a PolarDB for MySQL cluster by migrating an ApsaraDB RDS for MySQL instance

allows you to create a cluster by migrating an ApsaraDB RDS for MySQL instance. The created cluster uses the accounts, databases, IP address whitelist, and required parameters of the source ApsaraDB RDS for MySQL instance.

Benefits

- The endpoint of the source database is retained. You can switch to without changing the connection settings of your applications.
- You can complete data migration in the console. No data migration tools such as Data Transmission Service (DTS) are required.
- The data migration feature is free of charge to use.
- No data loss occurs during the migration.
- Incremental data migration is supported. This allows you to migrate data with a service downtime of less than 10 minutes.
- Hot migration is supported. Only a single transient disconnection occurs during the data migration

from ApsaraDB RDS for MySQL to .

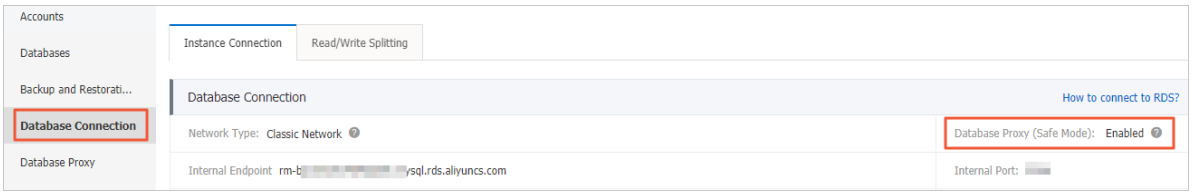
- Migration rollbacks are supported. If a migration fails, the migration can be rolled back within 10 minutes.

Prerequisites

- The edition of the source RDS instance is ApsaraDB RDS for MySQL 5.6 or 5.7 High-availability Edition, and the storage type is local SSD.
- ○ For ApsaraDB RDS for MySQL 5.6, the minor version of the kernel must be 20190815 or later.
  ○ For ApsaraDB RDS for MySQL 5.7, the minor version of the kernel must be 20200331 or later.

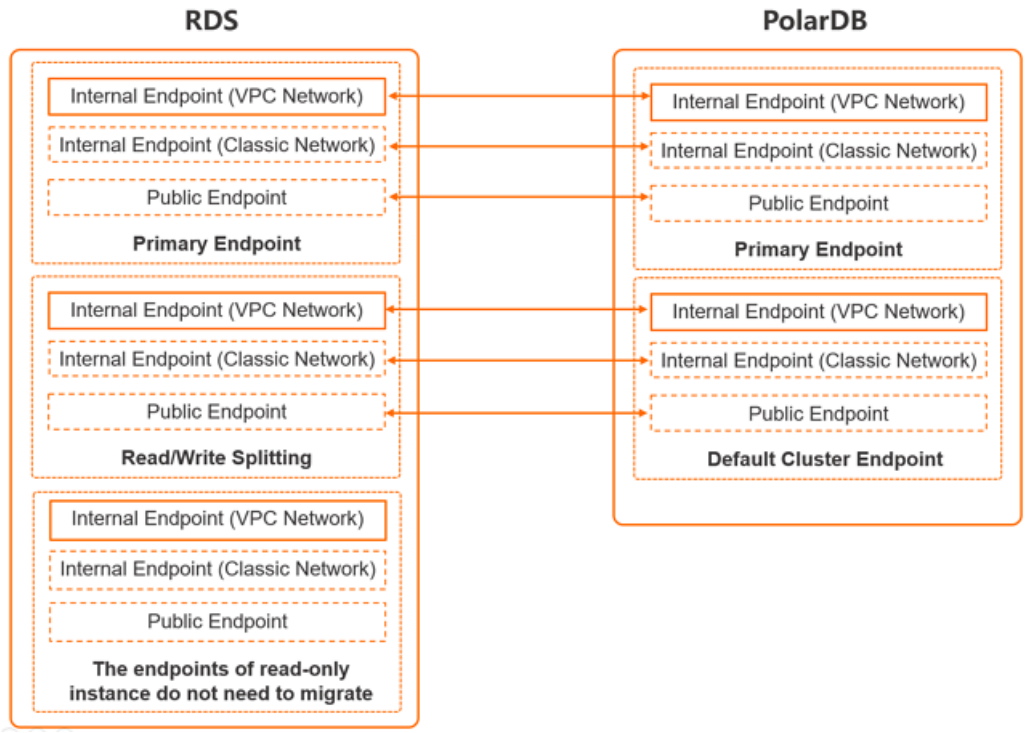
> **Note** You can run the `show variables like '%rds_release_date%';` command to view the minor version of the kernel for the source RDS instance. If the minor version of the kernel for the source RDS instance is earlier than the preceding version, you can upgrade the minor version of the kernel to the latest version. For more information, see Update the minor engine version of an ApsaraDB RDS for MySQL instance.

- Transparent Data Encryption (TDE) and SSL are disabled on the source RDS instance. For more information, see TDE and SSL.
- The table storage engine for the source RDS instance is InnoDB.
- If Database Proxy (Safe Mode) is enabled for the RDS instance, a privileged account is created or the network connection mode of the RDS instance is switched to high-performance mode. For more information, see Create an account and [Important] RDS network link upgrade.



Switching with endpoints

When you perform data migration from ApsaraDB RDS for MySQL to , you can select Switch with Endpoints (Connection Changes Not Required). Then, the system interchanges the endpoints between the RDS instance and the cluster. In this case, you do not need to modify the configurations of your applications to connect to the cluster. The following figure shows the rules for endpoint interchanges between the RDS instance and the cluster.



To switch with endpoints, take note of the following limits:

- Only the endpoints of the RDS instance and the cluster are interchanged. The other configurations such as the vSwitches and virtual IP addresses remain the same as before.
- Endpoints can be interchanged only if both the source RDS instance and the destination cluster have endpoints. By default, only the primary endpoints in the internal network can be interchanged.
- If you want to switch to a different endpoint, you must create these endpoints before the switchover. For more information about how to create endpoints for the cluster, see Configure PolarProxy. For more information about how to create endpoints for the RDS instance, see Configure endpoints for an RDS instance.
- The port numbers are not interchanged between the RDS instance and the cluster. You must make sure that the port number of the RDS instance is the same as that of the cluster. The default port number used by RDS and is 3306. If the port numbers are different, change one of the port numbers. For more information, see View and change the internal and public endpoints and port numbers of an ApsaraDB RDS for MySQL instance and Modify or delete an endpoint.
- After the endpoints are interchanged, issues may occur due to the expiration of the Domain Name System (DNS) cache data. The databases in the cluster may fail to be connected or support only read operations. We recommend that you refresh the DNS cache data of your server to fix this issue.
- If you want to use Data Management (DMS) to log on to the database after the endpoints are interchanged, you must use the latest version of DMS and the cluster ID to log on to the database. You cannot log on to the database by using the endpoint.

Limits

- Cross-region data migration is not supported.
- You cannot set the parameters of the source RDS instance during data migration.

Pricing

You are not charged for data migration from the source RDS instance to the cluster. You are charged only for the purchased cluster. For more information, see Billable items.

Procedure

Step	Description
1. Migrate data from the source RDS instance	In this step, a cluster is created. The cluster stores the same data as the source RDS instance. Then, incremental data is synchronized from the source RDS instance to the cluster in real time.
2. Switch over services	• When you switch over services, you can select Switch with Endpoints (Connection Changes Not Required). Then, the system automatically interchanges the endpoints between the RDS instance and the cluster. The cluster must have an endpoint, such as the endpoint in the classic network or the Internet. In this case, you do not need to modify the configurations of your applications to connect to the cluster. • After the switchover, the read/write state of the source RDS instance changes to Read Only, and the read/write state of the cluster changes to Read and Write. Incremental data on the cluster is synchronized to the source RDS instance. **Note** If data errors occur after the switchover is complete, you can roll back the migration. This allows you to restore the database and data to the state before the data migration. For more information, see Roll back the migration.
3. Complete the migration	• We recommend that you test the cluster to confirm that the cluster runs as expected before you click Complete Migration. You must click Complete Migration within seven days after the cluster is created. • After you click Complete Migration, the system stops synchronizing data between the source RDS instance and the cluster. The read/write state of the source RDS instance is changed to Read and Write.

Step 1: Migrate data from the source RDS instance

In this step, a cluster is created. The cluster stores the same data as the source RDS instance. Then, incremental data is synchronized from the source RDS instance to the cluster in real time.

1.
2.
3. Click **Create Cluster**.
4. Set **Product Type** to **Subscription** or **Pay-As-You-Go**.
   - **Subscription**: If you select this billing method, you must pay for compute nodes when you create the cluster. You are charged by hour for the amount of storage space consumed by your data. The fees are deducted from your account on an hourly basis.
   - **Pay-As-You-Go**: If you use this billing method, you are charged for the used resources. An upfront payment is not required. You are charged by hour for the compute nodes and the amount of storage space consumed by your data. These fees are deducted from your account on an hourly basis.

5. Set the following parameters.

Parameter	Description
Region	The region where the source ApsaraDB RDS for MySQL instance resides. **Note** The cluster to be created must also be deployed in this region.
Create Type	Select **Migration from RDS**. All of the data is cloned from the source RDS instance. Then, incremental data is synchronized from the source RDS instance to the cluster in real time. By default, the cluster enters read-only mode and the binary logging feature is enabled for the cluster before the switchover.
RDS Engine Type	The database engine of the source RDS instance. The default value of this parameter is **MySQL** and cannot be changed.
RDS Engine Version	The engine version of the source RDS instance. You can select **5.6** or **5.7**.
Source RDS Instance	The source RDS instance. The available source RDS instances exclude read-only instances.
Primary Availability Zone	Each zone is an independent geographical location in a region. All zones in a region provide the same services. You can choose to create your cluster in the same zone as the Elastic Compute Service (ECS) instance or in the zone that is different from the zone of this instance. You need to specify only the primary zone. The system selects a secondary zone.
Network Type	The network type of the cluster. The value of this parameter cannot be changed.
VPC **VSwitch**	The VPC and the vSwitch to which the cluster belongs. Make sure that the cluster and the ECS instance to be connected are deployed in the same VPC. Otherwise, the cluster and the ECS instance cannot communicate over the internal network to achieve the optimal performance.
Compatibility	The database engine version of the cluster. The default version is the same as the engine version of the source RDS instance and cannot be changed.
Edition	The edition of the cluster. This parameter is automatically set to . You do not need to specify this parameter.
Node Specification	The node specifications of the cluster. You can specify node specifications based on your business requirements. We recommend that you select specifications that are the same as or more advanced than the specifications of the source RDS instance. For more information about node specifications, see Specifications of compute nodes.

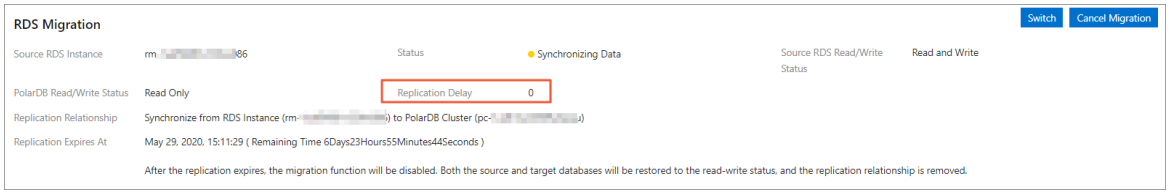
Parameter	Description
Nodes	The number of nodes to be created in the cluster. You do not need to specify this parameter. The system creates a read-only node of the same specifications as the primary node.
Storage Cost	The storage cost. You do not need to specify this parameter. You are charged by hour for the storage space that is consumed by the actual amount of data. For more information, see Specifications and pricing. **Note** You do not need to specify the storage capacity when you create a cluster. The system scales the storage capacity when the amount of data is increased or decreased.
Time Zone	The time zone of the cluster. The default value is **UTC+08:00**.
Table Name Case Sensitivity	Specifies whether the table names of the cluster are case-sensitive. The default value is **Not Case-sensitive (Default)**. If your on-premises database has case-sensitive table names, select **Case-sensitive** to facilitate data migration. **Note** The value of this parameter cannot be changed after the cluster is created. Proceed with caution.
Release Cluster	The backup retention policy that is used when you delete or release the cluster. Default value: **Retain Last Automatic Backup (Automatic Backup before Release) (Default)**. Valid values: ○ **Retain Last Automatic Backup (Automatic Backup before Release) (Default)**: retains the last backup when you delete the cluster. ○ **Retain All Backups**: retains all backups when you delete the cluster. ○ **Delete All Backups (Cannot be restored)**: deletes all backups when you delete the cluster. **Note** If you choose to retain backups when you delete the cluster, you may be charged for the backups. You can delete the backups to reduce costs. For more information, see Billing rules of backup storage that exceeds the free quota.
Cluster Name	○ The name of the new cluster. It must be 2 to 128 characters in length and can contain letters, digits, periods (.), underscores (_), and hyphens (-). It must start with a letter. ○ If you leave this parameter empty, the system generates a cluster name. You can change the cluster name after the cluster is created.

Parameter	Description
Resource Group	The resource group of the cluster. Select a resource group from available resource groups. For more information, see Create a resource group. **Note** A resource group contains a group of resources that belong to an Alibaba Cloud account. Resource groups allow you to manage these resources in a centralized manner. A resource belongs to only one resource group. For more information, see Use RAM to create and authorize resource groups.

6. If you create a **subscription** cluster, set **Purchase Plan** and **Number** and click **Buy Now** in the right corner.
7. On the **Confirm Order** page, confirm your order information. Read and accept the terms of service, and then click **Activate Now**.

   After you complete the activation, it takes 10 to 15 minutes to create the cluster. Then, the newly created cluster is displayed on the **Clusters** page.

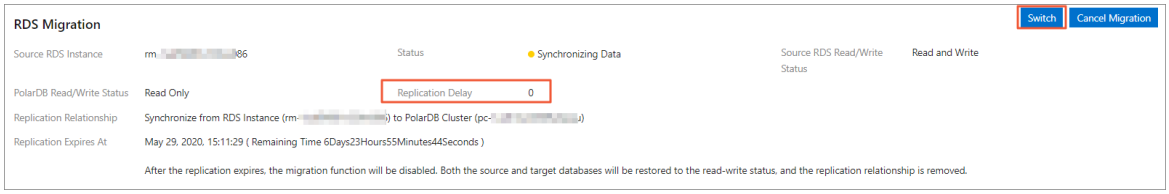
   Note
   - If nodes in the cluster are in the **Creating** state, the cluster is being created and unavailable. The cluster is available only if it is in the **Running** state.
   - Make sure that you have selected the region where the cluster is deployed. Otherwise, you cannot view the cluster.
   - We recommend that you purchase subscription storage plans to store a large amount of data. Storage plans are more cost-effective than pay-as-you-go storage. You are offered larger discounts if you purchase storage plans that provide larger storage capacities. For more information, see Billing method 12: pay-as-you-go.
8. On the **Purchase** page, confirm the unpaid order and the payment method and click **Purchase**.
9. After the cluster is created, log on to the PolarDB console and check that the **Replication Delay** of the cluster is less than 60 seconds. Then, you can go to Step 2: Switch over services.



   Note
   - After the cluster is created, the system starts to migrate data from the RDS instance to the cluster. You must click Complete Migration in the PolarDB console within seven days after the cluster is created. Otherwise, the migration task is automatically terminated after seven days.
   - You can also cancel the migration task. For more information about the impacts of canceling a migration task, see FAQ.

Step 2: Switch over services

1. Log on to the PolarDB console.
2. Find the cluster and click the cluster ID.
3. On the **Overview** page, click **Switch**.

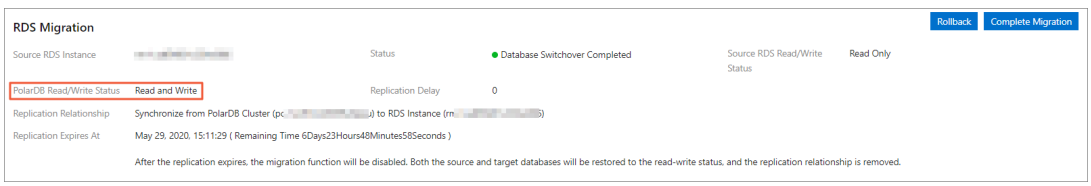


> **Note**
>
> - In most cases, it requires less than 5 minutes for the system to compete the switchover.
> - After the switchover, the read/write states of the source RDS instance and the destination cluster are interchanged. The read/write state of the source RDS instance is changed to Read Only, and the read/write state of the cluster is changed to Read and Write. The replication direction is also changed. This way, the incremental data on the cluster is synchronized to the RDS instance.

4. In the **Start Switching** dialog box, select **Switch with Endpoints (Connection Changes Not Required)** or **Switch without Endpoints (Connection Changes Required)**.
   - If you select **Switch with Endpoints (Connection Changes Not Required)**, perform the following steps:
     a. After you select **Switch with Endpoints (Connection Changes Not Required)**, the system interchanges the endpoints between the RDS instance and the cluster. You do not need to modify the configurations of your applications to connect to the cluster.

     > **Notice** Before you select **Switch with Endpoints (Connection Changes Not Required)**, make sure that you have read the related notes.

     b. Click **OK**.
   - If you select **Switch without Endpoints (Connection Changes Required)**, perform the following steps:
     a. Select **Switch without Endpoints (Connection Changes Required)**. After the switchover, you must change the database endpoint in your application at the earliest opportunity.
     b. Click **OK**.
     c. Refresh the page. After **PolarDB Read/Write Status** becomes **Read and Write**, change the database endpoint in your application at the earliest opportunity.



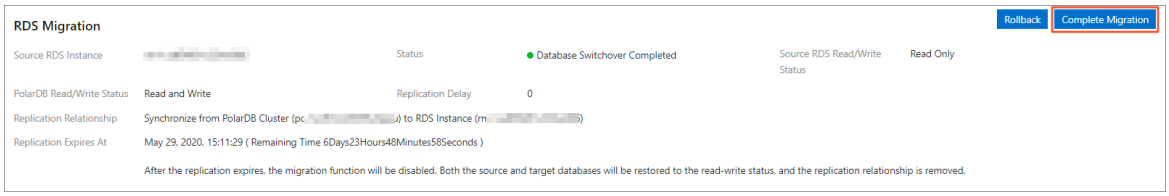
> **Note** If data errors are found after the switchover, you can roll back the database to the state before the data migration. For more information, see Roll back the migration.

Step 3: Complete the migration

After you perform the operations that are described in Migrate data from the source RDS instance, you must click **Complete Migration** to complete the migration within seven days after the cluster is created.

> **Warning** After the migration is complete, the system stops synchronizing data between the RDS instance and the cluster, and the migration rollback feature becomes unavailable. We recommend that you test the cluster to confirm that the cluster runs as expected before you complete the migration.

1. Log on to the PolarDB console.
2. Find the cluster and click the cluster ID.
3. On the **Overview** page, click **Complete Migration**. In the message that appears, click **OK**.

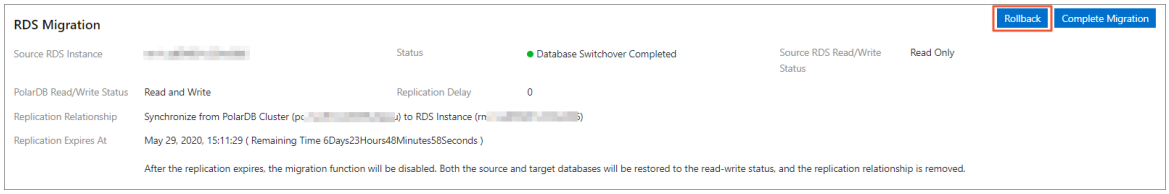


> **Note**
>
> - After you click **OK**, the system stops data synchronization within about 2 minutes. During this process, the state displayed is **Disabling Synchronization**. Wait until the migration is complete.
> - In the **Complete Migration** dialog box, you can specify whether to disable the binary logging feature for the cluster. If this feature is disabled, the write performance can be slightly improved. However, the cluster is automatically restarted.
> - If you no longer need to use the source RDS instance, you can release the instance.For more information, see Release or unsubscribe from an ApsaraDB RDS for MySQL instance.

Roll back the migration (optional)

If data errors occur before the migration is complete, you can roll back the migration. This allows you to restore the database and data to the state before the data migration. After the rollback, the read/write state of the source RDS instance is changed to Read and Write, and the read/write state of the cluster is changed to Read Only. The system synchronizes data from the RDS instance to the cluster.

1. Log on to the PolarDB console.
2. Find the cluster and click the cluster ID.
3. On the **Overview** page, click **Rollback**.



4. In the **Switch Back** dialog box, select **Switch Back with Endpoints (Connection Changes Not Required)** or **Switch Back without Endpoints (Connection Changes Required)**.
   - If you select **Switch Back with Endpoints (Connection Changes Not Required)**, perform the following steps:
     a. After you select **Switch Back with Endpoints (Connection Changes Not Required)**, the system interchanges the endpoints between the RDS instance and the cluster. You do not need to modify the configurations of your applications to connect to the RDS instance.
     b. Click **OK**. This way, the read/write state of the source RDS instance is changed to Read and Write, and the read/write state of the cluster is changed to Read Only. The system synchronizes data from the RDS instance to the cluster.
   - If you select **Switch Back without Endpoints (Connection Changes Required)**, perform the following steps:
     a. Select **Switch Back without Endpoints (Connection Changes Required)**. After the switchover, you must change the database endpoint in your application at the earliest opportunity.
     b. Click **OK**. This way, the read/write state of the source RDS instance is changed to Read and Write, and the read/write state of the cluster is changed to Read Only. The system synchronizes data from the RDS instance to the cluster.
     c. Refresh the page. After **Source RDS Read/Write Status** becomes **Read and Write**, change the database endpoint in your application to the endpoint of the RDS instance at the earliest opportunity.

FAQ

- What are the differences among creating a PolarDB for MySQL cluster by migrating an ApsaraDB RDS for MySQL instance, creating an cluster by using the Clone from RDS method (Create a PolarDB for MySQL cluster by cloning an ApsaraDB RDS for MySQL instance), and migrating data from an ApsaraDB RDS for MySQL instance to a cluster (Migrate data from an ApsaraDB RDS for MySQL instance to a PolarDB for MySQL cluster)?

  The following table describes the differences.

Item	Create a cluster by migrating an ApsaraDB RDS for MySQL instance	Create a PolarDB for MySQL cluster by cloning an ApsaraDB RDS for MySQL instance	Migrate data from an ApsaraDB RDS for MySQL instance to a PolarDB for MySQL cluster
Whether DTS is required	No	No	Yes
Whether incremental data migration or synchronization is supported	Yes	No	Yes

Item	Create a cluster by migrating an ApsaraDB RDS for MySQL instance	Create a PolarDB for MySQL cluster by cloning an ApsaraDB RDS for MySQL instance	Migrate data from an ApsaraDB RDS for MySQL instance to a PolarDB for MySQL cluster
Whether operations on the source RDS instance are affected	No	No	No
Whether the source and destination databases can use different MySQL versions	No	No	Yes

Note

- Do the node specifications of need to be the same as those of the source ApsaraDB RDS for MySQL instance?

  You can select the specifications of based on your business requirements. We recommend that you select specifications that are the same as or more advanced than those of the source RDS instance. All nodes in the cluster are dedicated nodes. The dedicated nodes provide stable and reliable performance. For more information, see Billable items.

- Do I need to purchase a cluster before the upgrade from RDS to PolarDB?

  You do not need to purchase a cluster in advance. You can perform the operations that are described in Step 1: Migrate data from the source RDS instance to purchase and create a cluster that has the same data as the source RDS instance.

- Is the source RDS instance affected when data is migrated from the RDS instance?

  No, the source RDS instance is not affected when data is migrated from the RDS instance.

- What are the impacts of smooth migration on the RDS instance?

  The migration does not affect the operations on the source RDS instance. However, data migration involves read operations, and read operations affect query performance on the source RDS instance.

- What are the impacts of smooth migration on my workloads that run on the databases?

  A smooth migration ensures that no data is lost during the migration. The service downtime is less than 10 minutes. During service downtime, the service is suspended and does not generate incremental data but the database is not stopped. You can roll back the migration based on your business requirements.

- What happens if I cancel the migration?

  After the migration is canceled, you can modify the parameters of the source RDS instance. The read/write state of the cluster is changed to Read and Write, and the data in the cluster is not deleted. If you manually cancel the migration, you can specify whether to disable the binary logging feature for the cluster. Binary logging is not disabled if the migration is automatically canceled.

- Do I need to change the endpoint in my applications after the service is switched over to ?

  When you switch over services, you can select **Switch with Endpoints (Connection Changes Not Required)**. Then, the system interchanges the endpoints between the RDS instance and the cluster. In this case, you do not need to modify the configurations of your applications to connect to the cluster.

- I select **Switch with Endpoints (Connection Changes Not Required)** when I switch over services. Why do I need to use the new endpoint to connect to the cluster?

  Endpoints can be interchanged only if both the source RDS instance and the destination cluster have endpoints. By default, only the primary endpoint in the internal network is interchanged. If you want to switch to a different endpoint, you must create these endpoints before the switchover. For more information about how to create endpoints for the cluster, see Apply for a cluster endpoint or a primary endpoint. For more information about how to create endpoints for the RDS instance, see Configure endpoints for an RDS instance.

- If the source RDS instance contains read-only instances, can the endpoints of read-only instances be switched when **Switch with Endpoints (Connection Changes Not Required)** is selected?

  No, the endpoints of read-only instances cannot be switched. The endpoints of the source RDS read-only instances are not migrated when you upgrade from RDS to . Only the endpoints and read/write splitting endpoints of the source RDS primary instance are migrated. You must manually modify the endpoints of the source RDS read-only instances in your applications to the endpoints of the cluster.

- After the switchover, connections to the databases cannot be made or data cannot be written to the databases. Why?

  After the endpoints are interchanged, issues may occur due to the expiration of the DNS cache data. The databases in the cluster may fail to be connected or support only read operations. We recommend that you refresh the DNS cache to fix this issue.

- Can I perform a compatibility test and evaluate the workloads before the migration to the cluster?

  You can first perform the operations described in Create a PolarDB for MySQL cluster by cloning an ApsaraDB RDS for MySQL instance to clone data to the cluster for compatibility tests and workload evaluation. Then perform the operations in this topic to upgrade to .

- Why does the **Complete Migration** button not appear in the console after I switch over services?

  If you complete the **Complete Migration** step, this button disappears. This avoids repeated operations.

- After the upgrade to , do I need to create the same username and password in the cluster as those in the source RDS instance?

  No, you do not need to create the same username and password in the cluster. The created cluster retains the accounts, databases, IP address whitelist, and required parameters of the source ApsaraDB RDS for MySQL instance.

- How do I perform migration from a source RDS instance that has TDE or SSL enabled to the cluster?

  RDS instances that have TDE or SSL enabled cannot be upgraded to clusters. You can use DTS to migrate data from the source RDS instance to the cluster. For more information, see Migrate data from an ApsaraDB RDS for MySQL instance to a PolarDB for MySQL cluster.

- Can I perform a cross-version upgrade when I create a cluster by migrating an ApsaraDB RDS for MySQL instance? For example, can I upgrade from ApsaraDB RDS for MySQL 5.6 to 8.0?

  No, you cannot perform a cross-version upgrade. To upgrade ApsaraDB RDS for MySQL 5.6 to 8.0, you must use DTS to migrate data. For more information, see Migrate data from an ApsaraDB RDS for MySQL instance to a PolarDB for MySQL cluster.

- If a DTS data synchronization task is started for the source RDS instance before the upgrade to , is the task affected during the upgrade?

No, when you perform operations described in this topic, full data is first cloned from the source RDS instance to the new cluster. Then, incremental data is synchronized from the source RDS instance to the cluster in real time. The data source of the DTS task is still the source RDS instance. The data migration to the cluster does not affect the operations on the source RDS instance.

However, after the data is migrated, if you switch your services to the new cluster and the source RDS instance is no longer used, the data source of DTS is not automatically changed to the new cluster. In this case, you must create a DTS synchronization task and change the data source to the cluster.

References

API operation	Description
CreateDBCluster	Creates a cluster. **Note** If you create a cluster by migrating an ApsaraDB RDS for MySQL instance, set CreationOption to MigrationFromRDS.
DescribeDBClusterMigration	Queries the migration state of a specified cluster.
ModifyDBClusterMigration	Switches or rolls back the task that migrates data from RDS to .
CloseDBClusterMigration	Cancels or completes the migration for a cluster.

3.3.2. Create a PolarDB for MySQL cluster by cloning an ApsaraDB RDS for MySQL instance

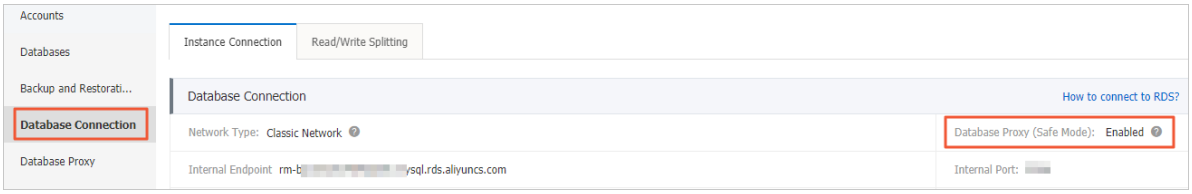
allows you to create a cluster by using the Clone from RDS method. You can use the Clone from RDS method to create a cluster that has the same data as the source ApsaraDB RDS instance. The created cluster retains the accounts, databases, IP whitelist, and required parameters of the source RDS instance.

Prerequisites

- The edition of the source RDS instance is ApsaraDB RDS for MySQL 5.6 or 5.7 High-availability Edition, and the storage type is local SSD.
- - For ApsaraDB RDS for MySQL 5.6, the minor version of the kernel must be 20190815 or later.
  - For ApsaraDB RDS for MySQL 5.7, the minor version of the kernel must be 20200331 or later.

  Note You can run the `show variables like '%rds_release_date%';` command to view the minor version of the kernel for the source RDS instance. If the minor version of the kernel for the source RDS instance is earlier than the preceding version, you can upgrade the minor version of the kernel to the latest version. For more information, see Update the minor engine version of an ApsaraDB RDS for MySQL instance.
- Transparent Data Encryption (TDE) and SSL are disabled on the source RDS instance. For more information, see TDE and SSL.
- The table storage engine for the source RDS instance is InnoDB.

- If Database Proxy (Safe Mode) is enabled for the RDS instance, a privileged account is created or the network connection mode of the RDS instance is switched to the high-performance mode. For more information see Create an account and [Important] RDS network link upgrade.



Context

is a next-generation relational cloud database that is developed by Alibaba Cloud. The database service provides the following benefits:

- Large storage capacity: up to 200 TB.
- High performance: up to six times higher than the performance of MySQL.
- Serverless storage: You do not need to purchase the storage capacity in advance. The storage capacity can be automatically scaled. You are charged for the amount of storage space that you use.
- Temporary upgrade: supports temporary specification upgrades to handle short-term business peaks.

For more information, see Benefits.

Precautions

The incremental data of the source RDS instance cannot be synchronized to the created cluster.

> **Note** If you want to synchronize the incremental data of the source RDS instance to the cluster in real time when you create the cluster, follow the instructions in Create a PolarDB for MySQL cluster by migrating an ApsaraDB RDS for MySQL instance. This implements smooth migration without downtime.

Benefits

- The cloning feature is provided free of charge.
- No data loss occurs during the migration.

Procedure

1.
2.
3. Click **Create Cluster**.
4. Set **Product Type** to **Subscription** or **Pay-As-You-Go**.
   - **Subscription**: If you use this billing method, you must pay for compute nodes when you create the cluster. You are charged by hour for the amount of storage space consumed by your data. These fees are deducted from your account on an hourly basis.
   - **Pay-As-You-Go**: If you use this billing method, you are charged for the used resources. An upfront payment is not required. You are charged by hour for the compute nodes and the amount of storage space consumed by your data. These fees are deducted from your account

on an hourly basis.

5. Set the following parameters.

Parameter	Description
Region	The region where the source ApsaraDB RDS for MySQL instance is located. **Note** The cluster to be created must also be deployed in this region.
Create Type	Select **Clone from RDS**.
RDS Engine Type	The database engine of the source RDS instance. The default value of this parameter is **MySQL** and cannot be changed.
RDS Engine Version	The engine version of the source RDS instance. You can select **5.6** or **5.7**.
Source RDS Instance	The source RDS instance. Read-only instances are not included in the available source RDS instances.
Primary Availability Zone	Each zone is an independent geographical location in a region. All zones in a region provide the same services. You can choose to create your cluster within the same zone as the Elastic Compute Service (ECS) instance or in a different zone. You need to specify only the primary zone. The system selects a secondary zone.
Network Type	The network type of the cluster. The value of this parameter cannot be changed.
VPC **VSwitch**	The virtual private cloud (VPC) and the vSwitch to which the cluster belongs. Make sure that the cluster and the ECS instance to be connected are deployed within the same VPC. Otherwise, the cluster and the ECS instance cannot communicate over the internal network for optimal performance.
Compatibility	The database engine version of the cluster. The default version is the same as the engine version of the source RDS instance and cannot be changed.
Edition	The edition of the cluster. This parameter is automatically set to . You do not need to specify this parameter.
Node Specification	The node specifications of the cluster. You can specify node specifications based on your business requirements. We recommend that you select specifications that are the same as or more advanced than the specifications of the source RDS instance. For more information about node specifications, see Specifications of compute nodes.
Nodes	The number of nodes to be created in the cluster. You do not need to specify this parameter. The system creates a read-only node of the same specifications as the primary node.

Parameter	Description
Storage Cost	The storage cost. You do not need to specify this parameter. You are charged by hour for the amount of storage space consumed by your data. For more information, see Specifications and pricing. **Note** You do not need to specify the storage capacity when you create a cluster. The system scales the storage capacity when the amount of data is increased or decreased.
Time Zone	The time zone of the cluster. The default value is **UTC+08:00**.
Table Name Case Sensitivity	Specifies whether table names of the cluster are case-sensitive. The default value is **Not Case-sensitive (Default)**. If table names are case-sensitive in your on-premises database, select **Case-sensitive** to facilitate data migration. **Note** The value of this parameter cannot be changed after the cluster is created. Proceed with caution.
Release Cluster	The backup retention policy that is used when you delete or release the cluster. Default value: **Retain Last Automatic Backup (Automatic Backup before Release) (Default)**. Valid values: ◦ **Retain Last Automatic Backup (Automatic Backup before Release) (Default)**: retains the last backup when you delete the cluster. ◦ **Retain All Backups**: retains all backups when you delete the cluster. ◦ **Delete All Backups (Cannot be restored)**: deletes all backups when you delete the cluster. **Note** You may be charged for the backups that are retained after you delete or release a cluster. For more information, see Release a cluster.
Cluster Name	◦ The name of the new cluster. It must be 2 to 128 characters in length and can contain letters, digits, periods (.), underscores (_), and hyphens (-). It must start with a letter. ◦ If you leave this parameter empty, the system generates a cluster name. You can change the cluster name after the cluster is created.

Parameter	Description
Resource Group	The resource group of the cluster. Select a resource group from available resource groups. For more information, see Create a resource group. **Note** A resource group contains a group of resources that belong to an Alibaba Cloud account. Resource groups allow you to manage these resources in a centralized manner. A resource belongs to only one resource group. For more information, see Use RAM to create and authorize resource groups.

6. If you create a **subscription** cluster, set **Purchase Plan** and **Number** and click **Buy Now** in the right corner.
7. On the **Confirm Order** page, confirm your order information. Read and accept the terms of service, and then click **Activate Now**.

   After you complete the activation, it takes 10 to 15 minutes to create the cluster. Then, the newly created cluster is displayed on the **Clusters** page.

   Note
   - If nodes in the cluster are in the **Creating** state, the cluster is being created and unavailable. The cluster is available only if it is in the **Running** state.
   - Make sure that you have selected the region where the cluster is deployed. Otherwise, you cannot view the cluster.
   - We recommend that you purchase subscription storage plans to store a large amount of data. Storage plans are more cost-effective than pay-as-you-go storage. You are offered larger discounts if you purchase storage plans that provide larger storage capacities. For more information, see Billing method 12: pay-as-you-go.
8. On the **Purchase** page, confirm the unpaid order and the payment method and click **Purchase**.
9. Log on to the PolarDB console and check the status of the new cluster.

FAQ

Is the source RDS instance affected when data is cloned from the RDS instance?

No, the source RDS instance is not affected when data is cloned from the RDS instance.

References

API	Description
CreateDBCluster	Creates a cluster. **Note** If you create a cluster by cloning an ApsaraDB RDS for MySQL instance, set CreationOption to CloneFromRDS.

What's next

You must change the endpoint over which your application accesses the database to the endpoint of the cluster at the earliest opportunity. For more information, see Apply for a cluster endpoint or a primary endpoint.

3.3.3. Migrate data from an ApsaraDB RDS for MySQL instance to a PolarDB for MySQL cluster

You can use Data Transmission Service (DTS) to migrate data from a MySQL database to a cluster.

Supported source databases

You can use DTS to migrate data from the following types of MySQL databases to a cluster. This topic uses an ApsaraDB RDS for MySQL instance as an example to describe how to configure a data migration task. You can also follow the procedure to configure data migration tasks for other types of MySQL databases.

- instance
- Self-managed databases:

Prerequisites

- The source instance is created. For more information, see Create an ApsaraDB RDS for MySQL instance.
- The destination cluster is created. For more information, see Purchase a pay-as-you-go cluster and Purchase a subscription cluster.
- The available storage space of the destination cluster is larger than the total size of the data in the source instance.

Limits

Migration types

SQL operations that can be migrated

Permissions required for database accounts

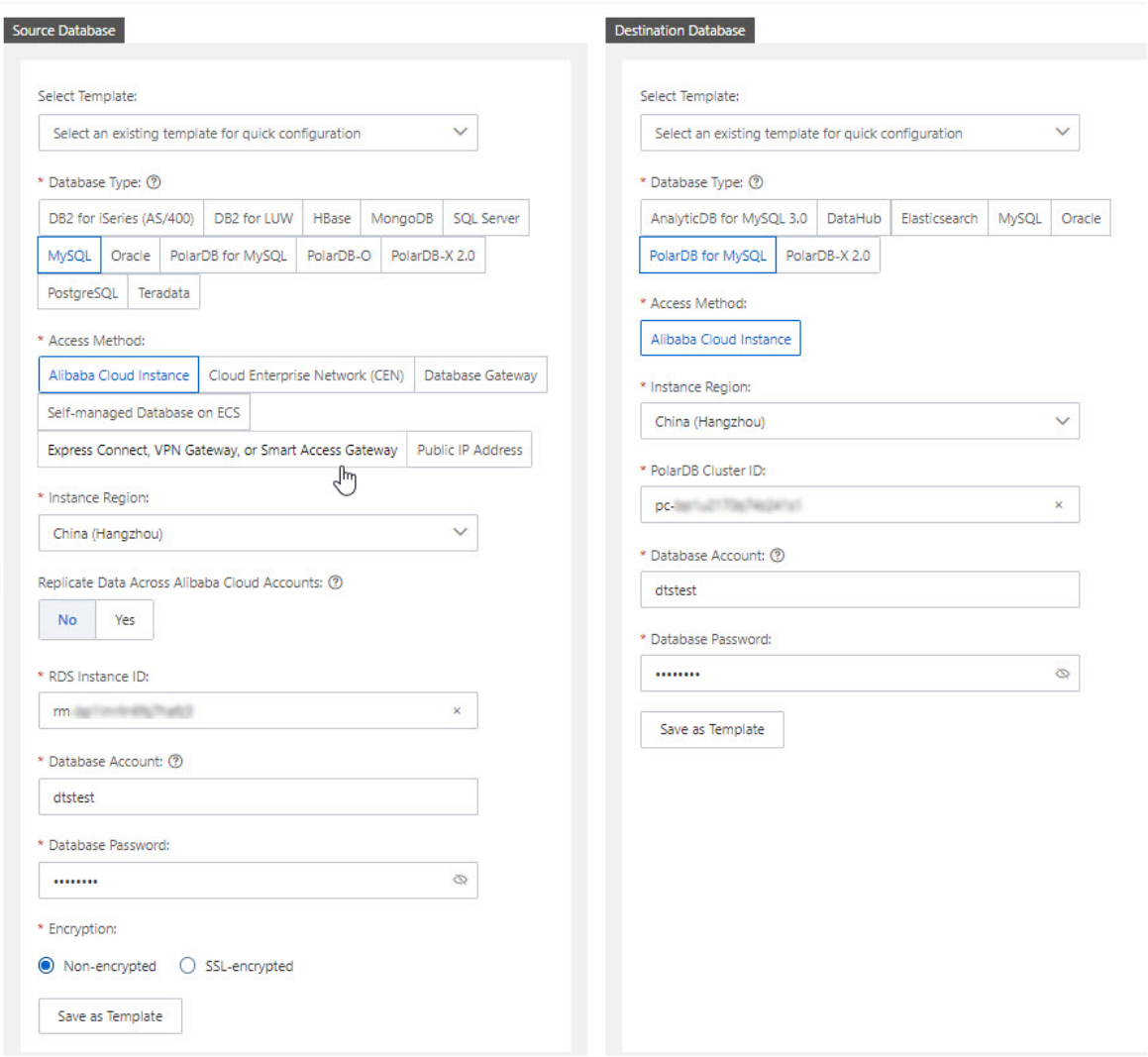
Database	Schema migration	Full data migration	Incremental data migration
instance	The SELECT permission	The SELECT permission	The REPLICATION CLIENT, REPLICATION SLAVE, SHOW VIEW, and SELECT permissions
cluster	The read and write permissions		

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

- 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
- cluster: Create a database account

Procedure

1.
2.
3.
4.
5. Warning



Section	Parameter	Description
N/A		
		Select **MySQL**.
		Select **Alibaba Cloud Instance**.
		Select the region where the source ApsaraDB RDS for MySQL instance resides.

Section	Parameter	Description
		In this example, select **No** because data is migrated within the same Alibaba Cloud account.
	RDS Instance ID	Select the ID of the source ApsaraDB RDS for MySQL instance.
		Enter the database account of the source ApsaraDB RDS for MySQL instance. For information about the permissions that are required for the account, see Permissions required for database accounts.
		Select **PolarDB for MySQL**.
		Select **Alibaba Cloud Instance**.
		Select the region where the destination cluster resides.
	PolarDB Cluster ID	Select the ID of the destination cluster.
		Enter the database account of the destination cluster. For information about the permissions that are required for the account, see Permissions required for database accounts.

6.

7. ○ **Basic Settings**

Parameter	Description
	Note
	In the **Selected Objects** section, right-click an object. In the dialog box that appears, select the DDL and DML operations that you want to migrate. For more information, see SQL operations that can be migrated.

○ **Advanced Settings**

Parameter	Description

8.
9.
10.
11.
12.

3.4. Migrate data between PolarDB databases

3.4.1. Migrate data between PolarDB for MySQL clusters

This topic describes how to migrate data between clusters by using Data Transmission Service (DTS).

> **Note** A cluster of an earlier version cannot be upgraded to 8.0. However, you can create a cluster of version 8.0, and then migrate data to this cluster. Before you migrate data across different versions of PolarDB for MySQL clusters, we recommend that you create a pay-as-you-go cluster to test the compatibility. After the test is complete, you can release the cluster.

Prerequisites

- The source and destination clusters are created. For more information, see Purchase a pay-as-you-go cluster and Purchase a subscription cluster.
- The available storage space of the destination cluster is larger than the total size of the data in the source cluster.

Limits

Migration types

SQL operations that can be migrated

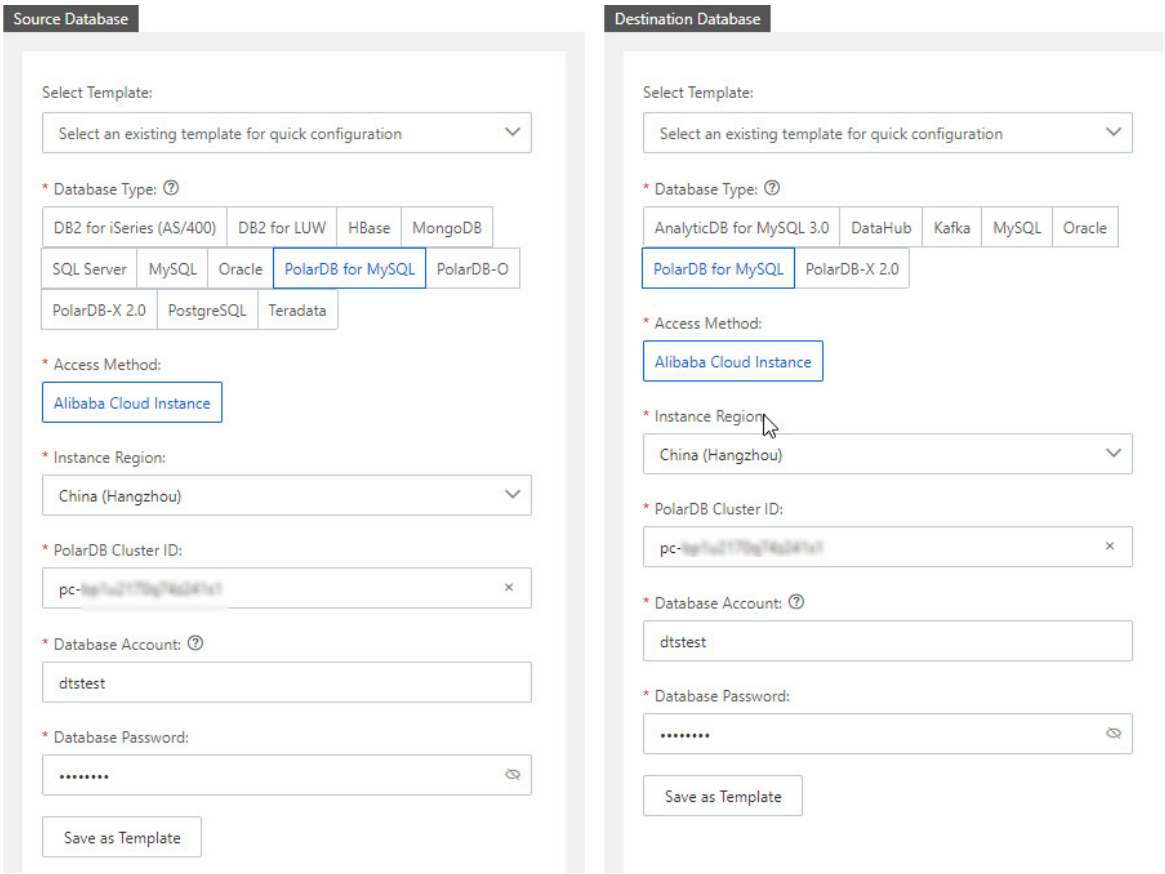
Permissions required for database accounts

Database	Required permissions
Source cluster	The read permissions on the objects to be migrated
Destination cluster	The read and write permissions on the objects to be migrated

For more information about how to create a database account for a cluster, see Create a database account.

Procedure

1.
2.
3.
4.
5. Warning



Section	Parameter	Description
N/A		
		Select **PolarDB for MySQL**.

Section	Parameter	Description
		Select **Alibaba Cloud Instance**.
		Select the region where the source cluster resides.
	PolarDB Cluster ID	Enter the ID of the source cluster.
		Enter the database account of the source cluster. For information about the permissions that are required for the account, see Permissions required for database accounts.
		Select **PolarDB for MySQL**.
		Select **Alibaba Cloud Instance**.
		Select the region where the destination cluster resides.
	PolarDB Cluster ID	Select the ID of the destination cluster.
		Enter the database account of the destination cluster. For information about the permissions that are required for the account, see Permissions required for database accounts.

6.

7. ○ **Basic Settings**

Parameter	Description
	Note
	In the **Selected Objects** section, right-click an object. In the dialog box that appears, select the DDL and DML operations that you want to migrate. For more information, see SQL operations that can be migrated.

○ **Advanced Settings**

Parameter	Description

Parameter	Description

8.
9.
10.
11.
12.

3.5. Migrate data from other databases to PolarDB

3.5.1. Migrate data from a self-managed MySQL database to a PolarDB for MySQL cluster

This topic describes how to migrate data from a self-managed MySQL database to a cluster by using Data Transmission Service (DTS).

Supported source databases

You can use DTS to migrate data from the following types of MySQL databases to a cluster. In this example, the source database is a self-managed MySQL database with a public IP address. You can also follow the procedure to configure data migration tasks for other types of MySQL databases.

- instance
- Self-managed databases:

Prerequisites

-
-
- The destination cluster is created. For more information, see Purchase a pay-as-you-go cluster and Purchase a subscription cluster.
- The available storage space of the cluster is larger than the total size of the data in the self-managed MySQL database.

Limits

Migration types

SQL operations that can be migrated

Permissions required for database accounts

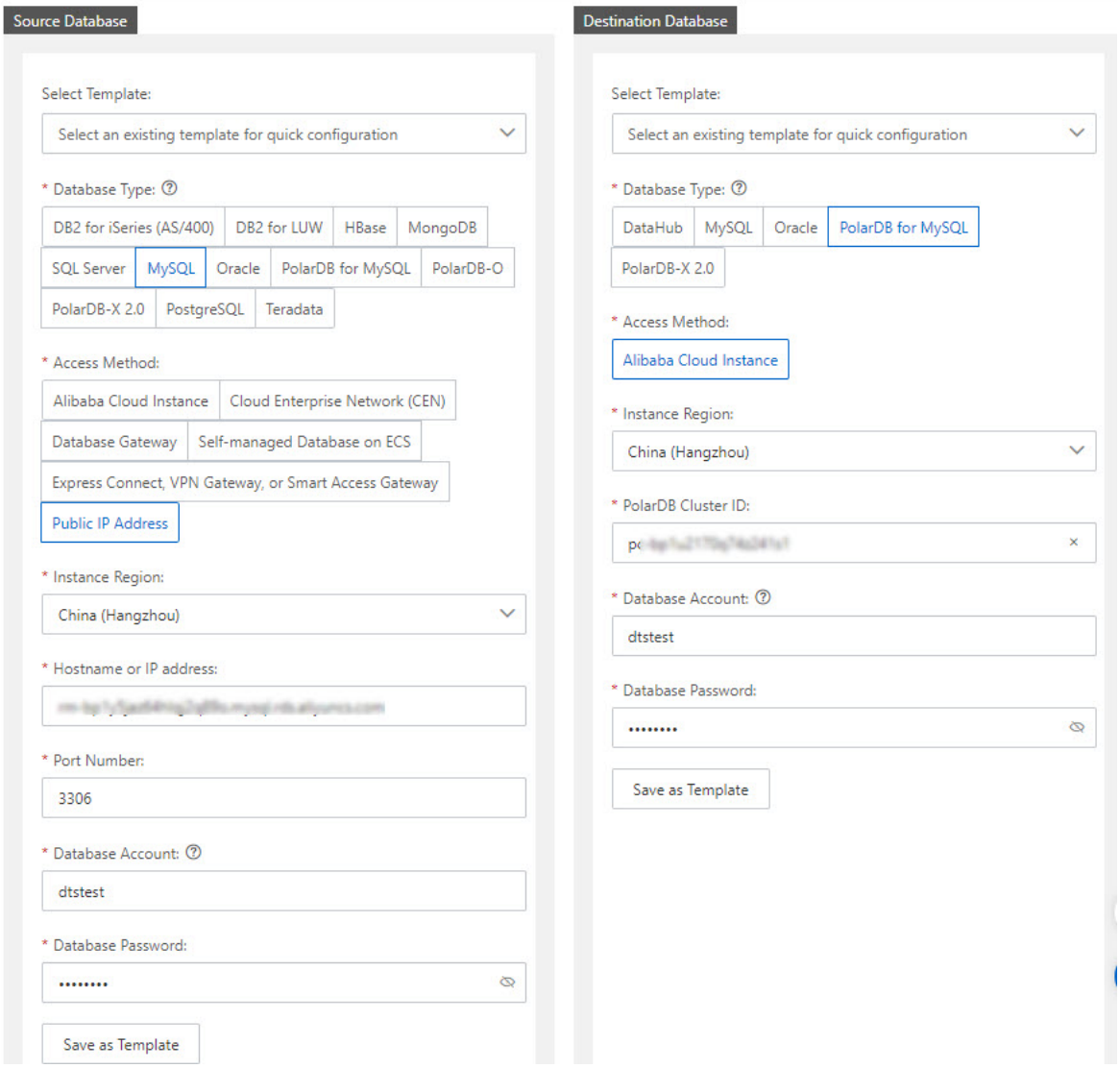
Database			
Self-managed MySQL database	The SELECT permission	The SELECT permission	Full data migration: the SELECT permission on the objects to be migrated The REPLICATION CLIENT, REPLICATION SLAVE, and SHOW VIEW permissions The permissions to create databases and tables. The permissions allow DTS to create a database named dts to record heartbeat data during migration.
cluster	The read and write permissions		

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

-
- cluster: Create a database account

Procedure

1.
2.
3.
4.
5. Warning



Section	Parameter	Description
N/A		
		Select **MySQL**.
		Select an access method based on the deployment of the source database. In this example, select **Public IP Address**. **Note** If you select other types of self-managed databases, you must deploy the network environment for the source database. For more information, see Preparation overview.
		Select the region where the self-managed MySQL database resides.
	Hostname or IP Address	Enter the endpoint that is used to access the self-managed MySQL database. In this example, enter the public IP address.

Section	Parameter	Description
	Port Number	Enter the service port number of the self-managed MySQL database. The port must be accessible over the Internet. The default port number is **3306**.
		Enter the account of the self-managed MySQL database. For information about the permissions that are required for the account, see Permissions required for database accounts.
		Select **PolarDB for MySQL**.
		Select **Alibaba Cloud Instance**.
		Select the region where the destination cluster resides.
	PolarDB Cluster ID	Select the ID of the destination cluster.
		Enter the database account of the destination cluster. For information about the permissions that are required for the account, see Permissions required for database accounts.

6.
7.
8. ○ **Basic Settings**

Parameter	Description
	Note
	In the **Selected Objects** section, right-click an object. In the dialog box that appears, select the DDL and DML operations that you want to migrate. For more information, see SQL operations that can be migrated.

○ **Advanced Settings**

Parameter	Description

9.
10.
11.
12.
13.

3.5.2. Migrate data from a self-managed MySQL database to a PolarDB for MySQL cluster by using mysqldump

This topic describes how to use mysqldump to migrate data from a self-managed MySQL database to .

Prerequisites

The following steps have been completed for the cluster:

- Create a database
- Create a database account
- Configure an IP whitelist
- Apply for a public endpoint

Compare migration methods

You can migrate data from a self-managed MySQL database to by using mysqldump or Data Transmission Service (DTS). The following table describes the differences between the two migration methods.

Item	mysqldump	DTS
The version of self-managed MySQL database	Unlimited	The version of the self-managed MySQL database is 5.1, 5.5, 5.6, 5.7, or 8.0.
Schema migration and full data migration	Supported	Supported
Incremental data migration	Not supported	Supported
Migration without service interruption	Not supported	Supported

> **Note** For more information about how to use DTS, see Migrate data from a user-created MySQL database to a PolarDB for MySQL cluster.

Considerations

After the migration is complete, the destination table names are not case-sensitive. All table names are provided in lowercase.

Procedure

> **Note** MySQL 8.0 and the Linux operating system are used in the example of this topic.

1. Use mysqldump to export data, stored procedures, triggers, and functions of the self-managed MySQL database.

   > **Note** Do not update data until the data export task is complete.

   i. In the Linux command-line interface (CLI), run the following command to export data:

   > **Note** When you enter the endpoint for the self-managed database, take note of the following rules:
   >
   > - If the self-managed MySQL database is deployed on an ECS instance, enter `127.0.0.1`.
   > - If the self-managed MySQL database is deployed on an on-premises machine, enter the public endpoint of the database.

   ```
   mysqldump -h <The endpoint of your database>  -u root -p --opt --default-character-set=utf8 --hex-blob <The name of your database> --skip-triggers --skip-lock-tables > /tmp/<The name of your database>.sql
   ```

   Example

   ```
   mysqldump -h 127.0.0.1 -u root -p --opt --default-character-set=utf8 --hex-blob testdb --skip-triggers --skip-lock-tables > /tmp/testdb.sql
   ```

ii. In the Linux CLI, run the following command to export stored procedures, triggers, and functions. This is an optional step.

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

```
mysqldump -h 127.0.0.1 -u root -p --opt --default-character-set=utf8 --hex-blob <The name of your database> -R | sed -e 's/DEFINER[ ]*=[ ]*[^*]*\*/\*/' > /tmp/<The name of your database>Trigger.sql
```

Example

```
mysqldump -h 127.0.0.1 -u root -p --opt --default-character-set=utf8 --hex-blob testdb -R | sed -e 's/DEFINER[ ]*=[ ]*[^*]*\*/\*/' > /tmp/testdbTrigger.sql
```

2. Run the following commands to import the exported files into the cluster:

```
mysql -h <The endpoint of the PolarDB cluster> -P <The port of the PolarDB cluster> -u <The username of the account of the PolarDB cluster> -p <The name of the PolarDB database> < /tmp/<The name of your database>.sql
mysql -h <The endpoint of the PolarDB cluster> -P <The port of the PolarDB cluster> -u <The username of the account of the PolarDB cluster> -p <The name of the PolarDB database> < /tmp/<The name of your database>Trigger.sql
```

Example

```
mysql -h polardbtest.mysql.polardb.rds.aliyuncs.com -P 3306 -u testuser -p testdb  < /tmp/testdb.sql
mysql -h polardbtest.mysql.polardb.rds.aliyuncs.com -P 3306 -u testuser -p testdb  < /tmp/testdbTrigger.sql
```

> **Note**
> - The database name must be the name of the database that is created on the cluster. For more information about how to create a database, see Create a database.
> - The account of the cluster must be a privileged account or a standard account that has the read and write permissions.

3. After the data is imported, you can log on to the cluster database to check the data. For more information, see Connect to a cluster.

FAQ

What do I do if the error message `Access denied; you need (at least one of) the SUPER privilege(s) for this operation` is returned?

SUPER permissions are required to execute the SQL statements in the script. You can first delete the statements that require the SUPER permissions, and execute the script.

3.5.3. Migrate data from an Amazon Aurora MySQL cluster to a PolarDB for MySQL cluster

This topic describes how to migrate data from an Amazon Aurora MySQL cluster to a PolarDB for MySQL cluster by using Data Transmission Service (DTS). DTS supports schema migration, full data migration, and incremental data migration. You can select all of the supported migration types to ensure service continuity.

Prerequisites

- The Public accessibility option of the Amazon Aurora MySQL cluster is set to **Yes**. The setting ensures that DTS can access the Amazon Aurora MySQL cluster over the Internet.
- A PolarDB for MySQL cluster is created. For more information, see Create a PolarDB for MySQL cluster.
- The available storage space of the PolarDB for MySQL cluster is larger than the total size of the data in the Amazon Aurora MySQL cluster.

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 source database must have PRIMARY KEY or UNIQUE constraints and all fields must be unique. Otherwise, the destination database may contain duplicate data records.
- DTS uses the `ROUND(COLUMN,PRECISION)` function to retrieve values from columns of the FLOAT or DOUBLE data type. If you do not specify a precision, DTS sets the precision for the FLOAT data type to 38 digits and the precision for the DOUBLE data type to 308 digits. You must check whether the precision settings meet your business requirements.
- If the name of the source database is invalid, you must create a database in the PolarDB for MySQL cluster before you configure a data migration task.

  > **Note** For more information about how to create a database and the database naming conventions, see Create a database on an ApsaraDB RDS for MySQL instance.

- 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 PolarDB for MySQL cluster. DTS supports schema migration for the following types of objects: 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 from the Amazon Aurora MySQL cluster to the PolarDB for MySQL cluster.

  > **Note** During full data migration, concurrent INSERT operations cause fragmentation in the tables of the destination instance. After full data migration is complete, the tablespace of the destination instance is larger than that of the source instance.

- Incremental data migration

  After full data migration is complete, DTS retrieves binary log files from the Amazon Aurora MySQL cluster. Then, DTS synchronizes incremental data from the Amazon Aurora MySQL cluster to the PolarDB for MySQL cluster. Incremental data migration allows you to ensure service continuity when you migrate data between MySQL databases.

Permissions required for database accounts

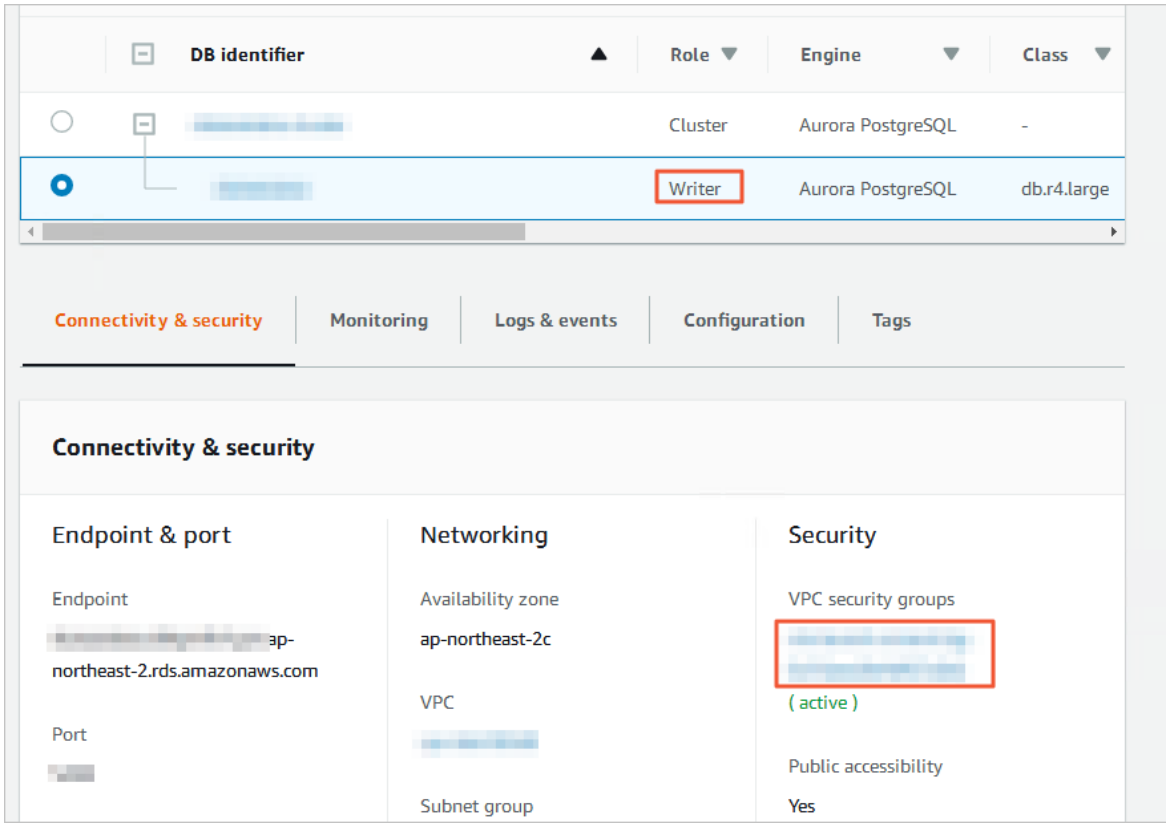
Database	Schema migration	Full data migration	Incremental data migration
Amazon Aurora MySQL	The SELECT permission on the objects to be migrated	The SELECT permission on the objects to be migrated	The SELECT permission on the objects to be migrated, the REPLICATION SLAVE permission, the REPLICATION CLIENT permission, and the SHOW VIEW permission
PolarDB for MySQL	The read and write permissions on the objects to be migrated	The read and write permissions on the objects to be migrated	The read and write permissions on the objects to be migrated

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

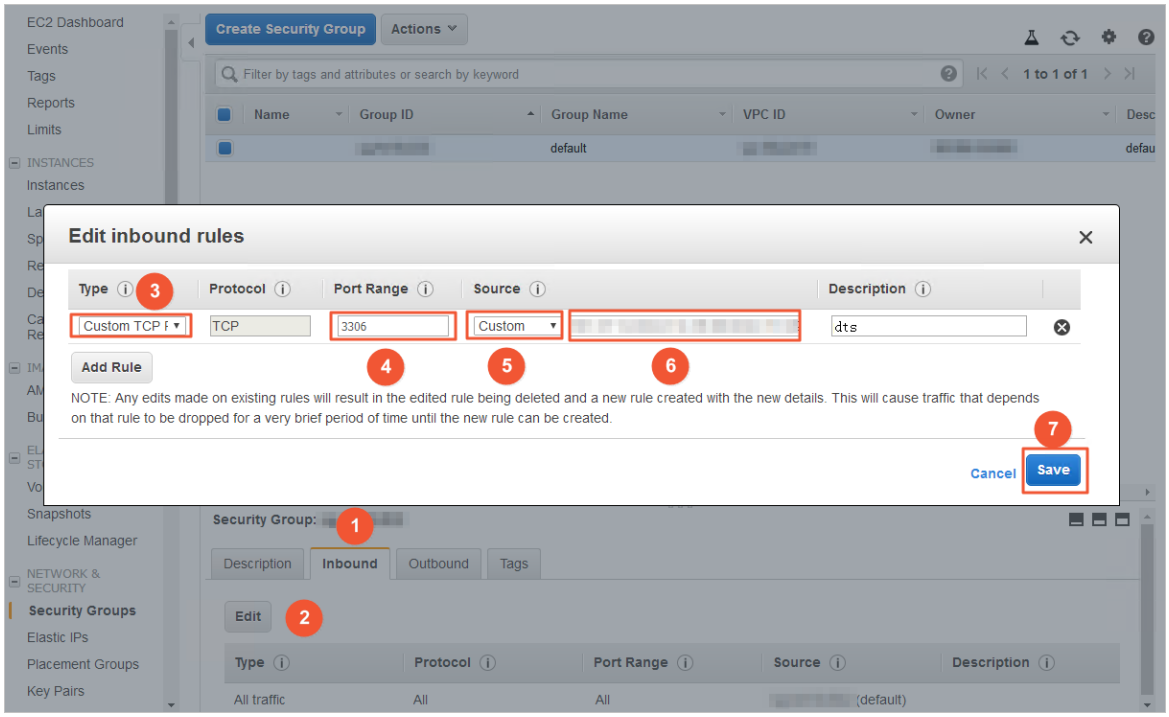
- Amazon Aurora MySQL cluster: Create an account for a user-created MySQL database and configure binary logging
- PolarDB for MySQL cluster: Create a database account.

Before you begin

1. Log on to the Amazon Aurora console.
2. Go to the **Basic information** page of the Amazon Aurora MySQL cluster.
3. Select the node that assumes the **writer** role.
4. In the **Connectivity & security** section, click the name of the VPC security group that corresponds to the writer node.



5. On the **Security Groups** page, click the Inbound tab in the Security Group section. On the Inbound tab, click Edit to add the CIDR blocks of DTS servers in the corresponding region to the inbound rule. For more information, see Add the CIDR blocks of DTS servers to the security settings of on-premises databases.



> **Note**
>
> - You need to add only the CIDR blocks of DTS servers that reside in the same region as the destination database. For example, the source database resides in the Singapore (Singapore) region and the destination database resides in the China (Hangzhou) region. You need to add only the CIDR blocks of DTS servers that reside in the China (Hangzhou) region.
> - You can add all of the required CIDR blocks to the inbound rule at a time.

6. Log on to the Amazon Aurora MySQL database and specify the number of hours to retain binary log files. Skip this step if you do not need to perform incremental data migration.

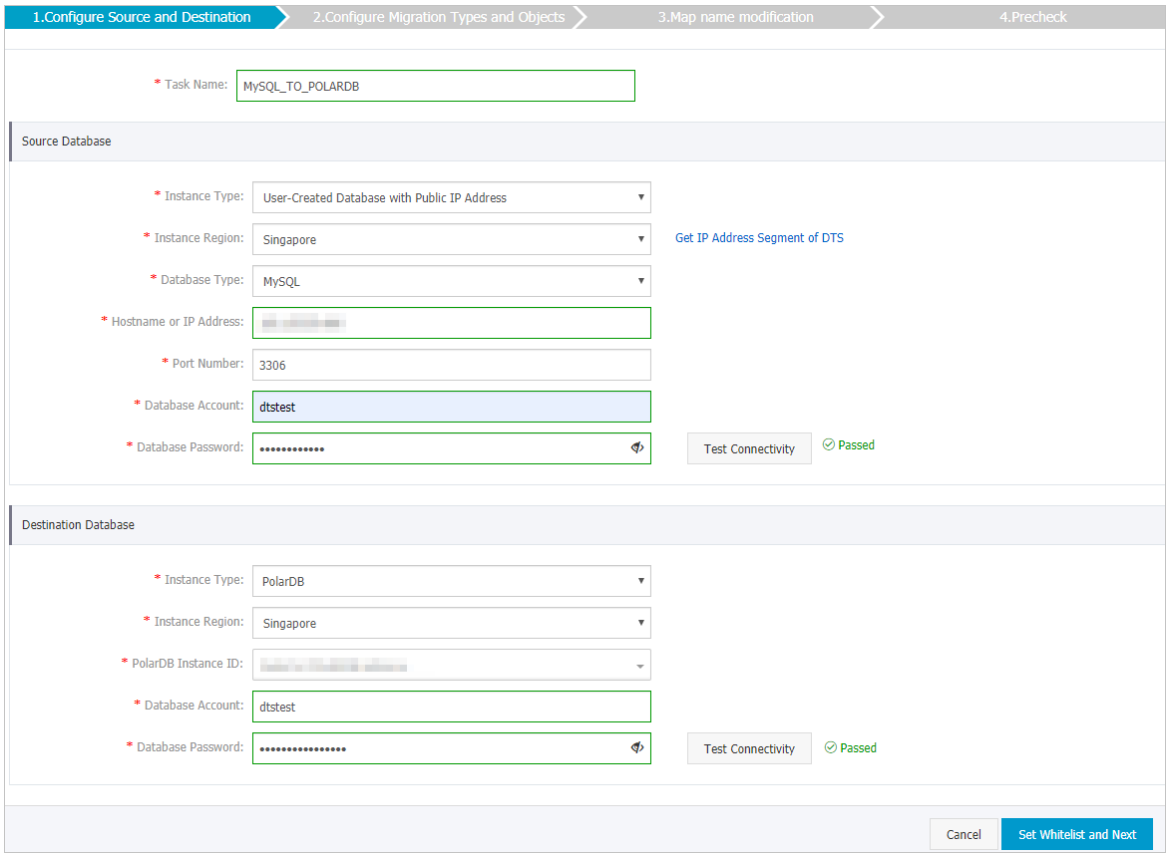
```
call mysql.rds_set_configuration('binlog retention hours', 24);
```

> **Note**
>
> - The preceding command sets the retention period of binary log files to 24 hours. The maximum value is 168 hours (7 days).
> - The binary logging feature of the Amazon Aurora MySQL cluster must be enabled and the value of the binlog_format parameter must be set to row. If the MySQL version is 5.6 or later, the value of the binlog_row_image parameter must be set to full.

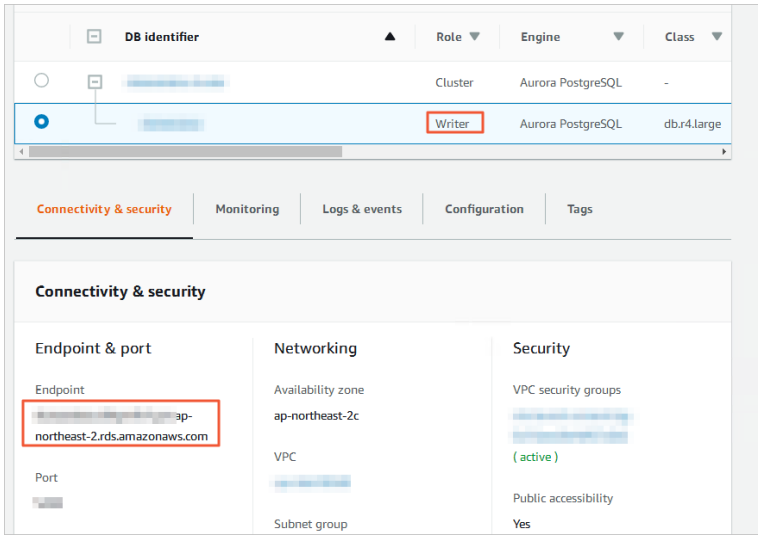
Procedure

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.
	Instance Type	Select **User-Created Database with Public IP Address**.
	Instance Region	If the instance type is set to **User-Created Database with Public IP Address**, you do not need to specify the **instance region**.
	Database Type	Select **MySQL**.

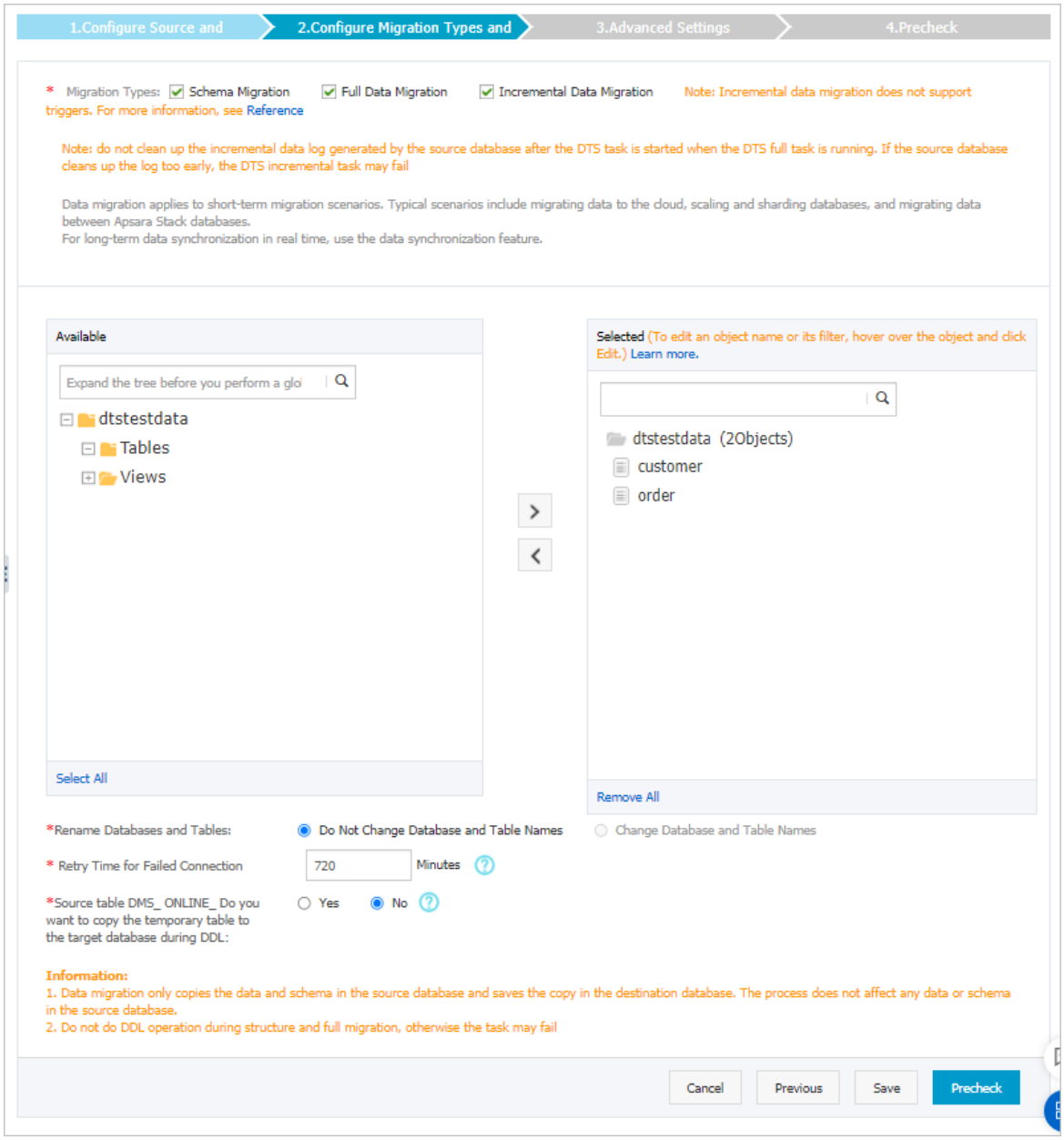
Section	Parameter	Description
Source Database	Hostname or IP Address	Enter the endpoint that is used to access the Amazon Aurora MySQL cluster. **Note** You can obtain the endpoint on the **Basic information** page of the Amazon Aurora MySQL cluster. 
	Port Number	Enter the service port number of the Amazon Aurora MySQL cluster. The default port number is **3306**.
	Database Account	Enter the database account of the Amazon Aurora MySQL cluster. For 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. **Note** After you specify the source database parameters, click **Test Connectivity** next to **Database Password** to verify whether the specified parameters are valid. 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.
	Instance Type	Select **PolarDB**.
	Instance Region	Select the region where the PolarDB for MySQL cluster resides.
	PolarDB Instance ID	Select the ID of the PolarDB for MySQL cluster.

Section	Parameter	Description
Destination Database	Database Account	Enter the database account of the PolarDB for MySQL cluster. For 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. **Note** After you specify the destination database parameters, click **Test Connectivity** next to **Database Password** to verify whether the specified parameters are valid. 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.

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

   Note In this step, DTS adds the CIDR blocks of DTS servers to the whitelist of the PolarDB for MySQL cluster. This ensures that DTS servers can connect to the cluster.

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



Setting	Description
Select the migration types	◦ To perform only full data 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**. **Note** If **Incremental Data Migration** is not selected, we recommend that you do not write data to the Amazon Aurora MySQL cluster 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. **Note** ◦ 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 12 hours. You can specify the retry time based on your needs. By default, if DTS fails to connect to the source or destination database, DTS retries within the next 12 hours. You can specify the retry time based on your needs. 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.
    - Schema migration and full data migration

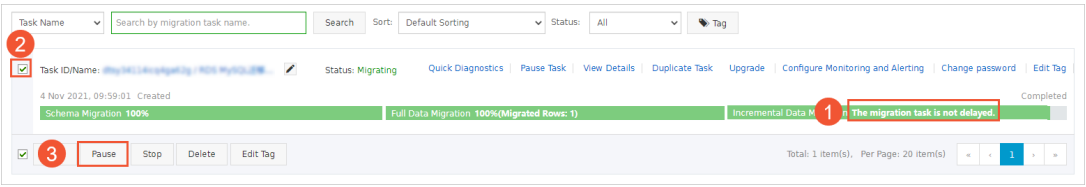
      We recommend that you do not manually stop the task during full data migration. Otherwise, the data migrated to the destination database will be incomplete. You can wait until the data migration task automatically stops.
    - Schema migration, full data migration, and incremental data migration

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

> **Notice** We recommend that you select an appropriate time to manually stop the data migration task. For example, you can stop the task during off-peak hours or before you switch your workloads to the destination cluster.

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

b. Wait until the status of **incremental data migration** changes to **The migration task is not delayed** again. Then, manually stop the migration task.



12. Switch your workloads to the PolarDB for MySQL cluster.

3.6. Migrate data from PolarDB to other databases

3.6.1. Migrate data from a PolarDB for MySQL cluster to an ApsaraDB RDS for MySQL instance

You can use Data Transmission Service (DTS) to migrate data from a PolarDB for MySQL cluster to a self-managed MySQL database or an instance.

Supported destination databases

You can use DTS to migrate data from a PolarDB for MySQL cluster to the following types of MySQL databases. This topic uses an instance as an example to describe how to configure a data migration task. You can also follow the procedure to configure data migration tasks for other types of MySQL databases.

- instance
- Self-managed databases:

Prerequisites

- The source cluster is created. For more information, see Purchase a pay-as-you-go cluster and Purchase a subscription cluster.
- The available storage space of the destination instance is larger than the total size of the data in the source cluster.

Limits

Migration types

SQL operations that can be migrated

Permissions required for database accounts

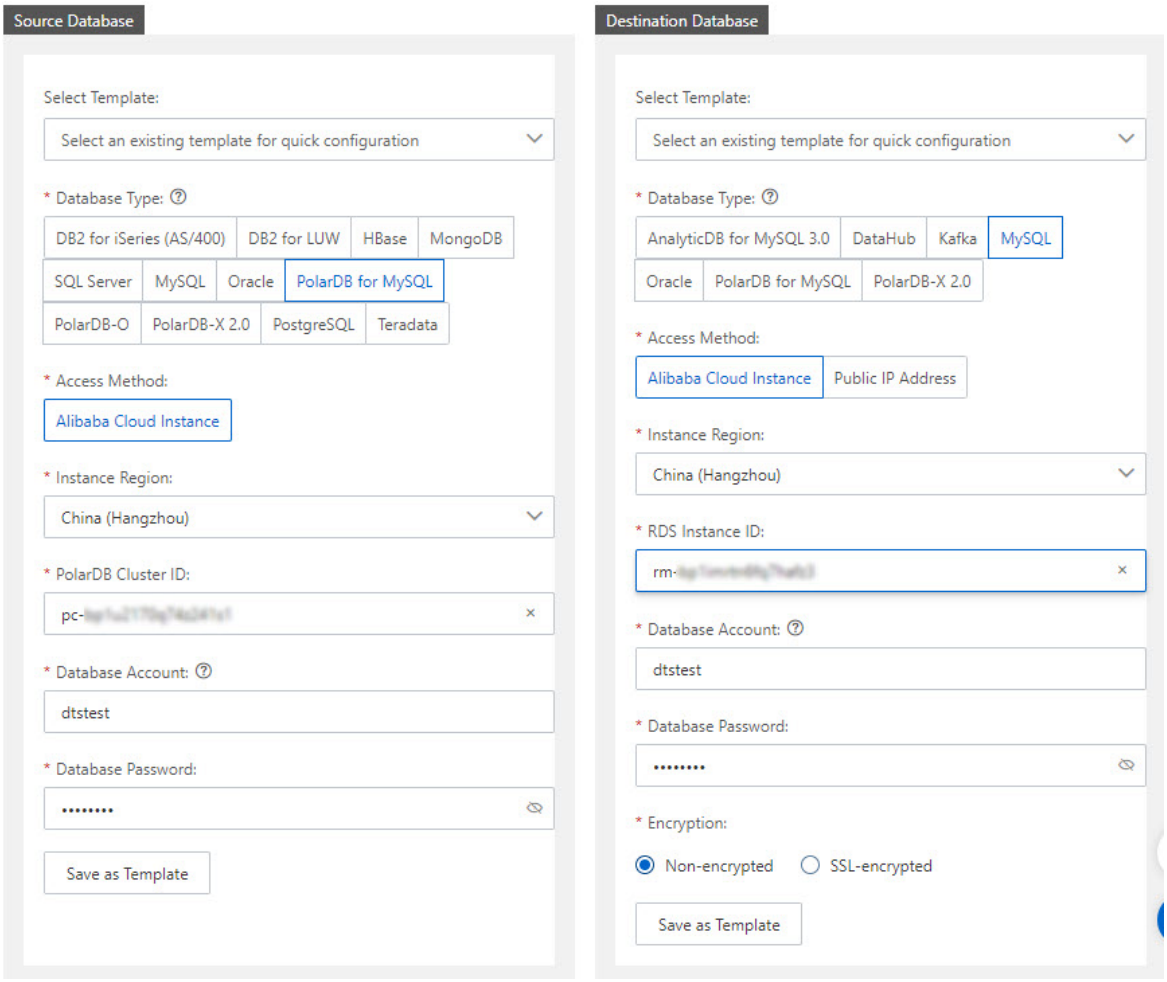
Database	Required permissions
cluster	The read permissions on the objects to be migrated
instance	The read and write permissions on the objects to be migrated

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

- cluster: Create a database account
- 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.
2.
3.
4.
5. Warning



Section	Parameter	Description
N/A		
		Select **PolarDB for MySQL**.
		Select **Alibaba Cloud Instance**.
		Select the region where the source cluster resides.
	PolarDB Cluster ID	Enter the ID of the source cluster.
		Enter the database account of the source cluster. For information about the permissions that are required for the account, see Permissions required for database accounts.
		Select **MySQL**.
		Select **Alibaba Cloud Instance**.
		Select the region where the destination instance resides.

Section	Parameter	Description
	RDS instance ID	Select the ID of the destination instance.
		Enter the database account of the destination instance. For information about the permissions that are required for the account, see Permissions required for database accounts.

6.
7. ○ **Basic Settings**

Parameter	Description
	Note
	In the **Selected Objects** section, right-click an object. In the dialog box that appears, select the DDL and DML operations that you want to migrate. For more information, see SQL operations that can be migrated.

○ **Advanced Settings**

Parameter	Description

8.
9.
10.
11.
12.

4.Account Management

4.1. Overview

This topic describes the Alibaba Cloud console accounts and the database accounts.

Console accounts

You can use the following accounts to log on to the console:

- Alibaba Cloud account: This type of account allows flexible control over all your Alibaba Cloud resources and is used for billing purposes. You must create an Alibaba Cloud account before you purchase Alibaba Cloud services.
- RAM user. You can create and manage RAM users in the Resource Access Management (RAM) console to share resources among multiple users. RAM users do not own resources. The charges of the resources consumed by RAM users are billed to the corresponding Alibaba Cloud account.

Database accounts

You can use the following accounts to log on to databases in the cluster. For more information, see Create a database account.

Account type	Description
Privileged Account	• You can use only the console to create and manage privileged accounts. • You can create only one privileged account for each cluster. A privileged account can manage all the standard accounts and databases in the corresponding cluster. • You can create a database and a standard account and authorize the standard account to perform add, delete, modify, and view operations on the database. • A privileged account has more permissions than before. This allows you to implement fine-grained control over user permissions based on your business requirements. For example, you can grant different users the permissions to query different tables. • A privileged account has the permissions to disconnect all standard accounts on the instance.
Standard Account	• You can create and manage standard accounts in the console or by using SQL statements. • You can create multiple standard accounts for each cluster. The maximum number of standard accounts that you can create depends on the database engine. • A standard account cannot be used to create databases or standard accounts. A standard account can only manage databases on which they have permissions. • A standard account does not have permissions to manage or disconnect the other accounts of the RDS instance on which the standard account is created.

Related operations

API	Description
CreateAccount	Creates an account.

API	Description
DescribeAccounts	Queries the accounts of the specified cluster.
ModifyAccountDescription	Modifies the description of a database account for the specified cluster.
ModifyAccountPassword	Changes the password of a database account for the specified cluster.
GrantAccountPrivilege	Grants a specified standard account the permissions on one or more databases of the specified cluster.
RevokeAccountPrivilege	Revokes the permissions on one or more databases from the specified standard account.
ResetAccount	Resets the permissions of a privileged account for the specified cluster.
DeleteAccount	Deletes an account.

4.2. Alibaba Cloud accounts

4.2.1. Register and log on to an Alibaba Cloud account

This topic describes how to register and log on to an Alibaba Cloud account.

Register an Alibaba Cloud account

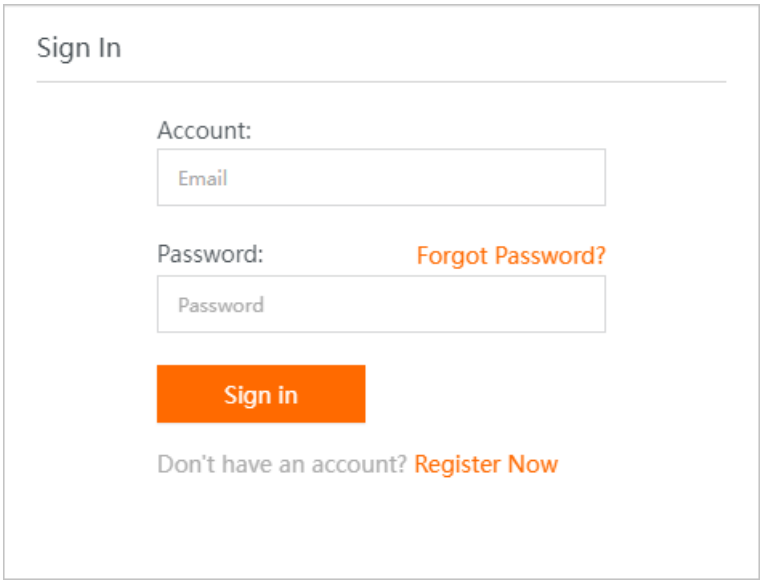
You can register an Alibaba Cloud account by using the following two methods:

- On the Alibaba Cloud International site, click **Free Account** in the upper-right corner.
- Directly go to the Alibaba Cloud account registration page.

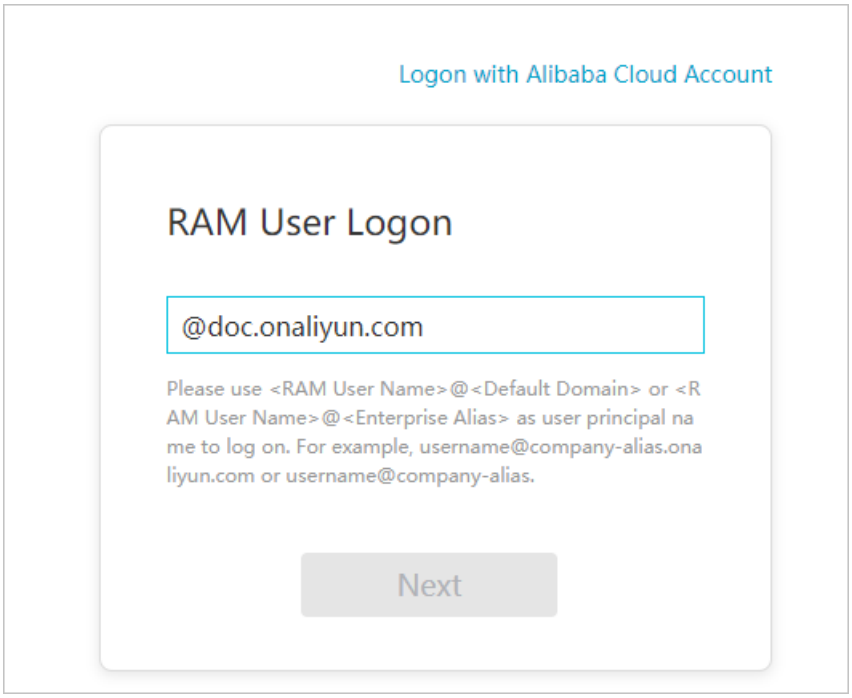
Log on to your Alibaba Cloud account

Your Alibaba Cloud account and Resource Access Management (RAM) user have different logon pages.

- The following figure shows the logon page for an Alibaba Cloud account.



- The following figure shows the logon page for a RAM user.



4.2.2. Create and authorize a RAM user

You can use your Alibaba Cloud account to access your resources. If you want to share the resources within your Alibaba Cloud account with other users, you must create and authorize Resource Access Management (RAM) users. After the authorization, the RAM users can access the specified resources. This topic describes how to create and authorize a RAM user.

Prerequisites

Log on to the console by using an Alibaba Cloud account or as a RAM user.

- For more information about how to use an Alibaba Cloud account, see Log on to the console with an Alibaba Cloud account.

- For more information about how to log on as a RAM user, see Log on to the console as a RAM user.

> **Note** The username of a RAM user must be in the format of `RAM username@enterprise alias`.

Procedure

- Create a RAM user. For more information, see Create a RAM user.
- Grant permissions to a RAM user on the Users page. For more information, see Grant permissions to a RAM user on the Users page.
- Grant permissions to a RAM user on the Grants page. For more information, see Grant permissions to a RAM user on the Grants page.
- Log on to the console as a RAM user. For more information, see Log on to the Alibaba Cloud Management Console as a RAM user.

Related operations

You can also add a RAM user to a group, assign roles to a RAM user, and authorize a user group or roles. For more information, see RAM User Guide.

4.2.3. Authorize RAM users to manage PolarDB by using custom policies

This topic describes how to authorize Resource Access Management (RAM) users to manage PolarDB by using custom policies. If the system policies that are provided by RAM cannot meet your business requirements, you can create custom policies to manage permissions. For example, you can create custom policies to grant permissions on specific resources and operations.

Prerequisites

Make sure that an Alibaba Cloud account is created before you use RAM to manage permissions. If not, go to the Sign up to Alibaba Cloud page.

Context

- A policy defines a set of permissions that are described based on the policy structure and syntax. A policy describes the authorized resource sets, authorized operation sets, and the authorization conditions. For more information, see Policy structure and syntax.
- Before you use custom policies for fine-grained permission management, familiarize yourself with how to specify resources for RAM users in policies. For more information, see Use RAM for resource authorization.

> **Note** To customize permissions or grant the specific permissions on tables, you can use the permission management feature of Database Management Service (DMS). For more information, see Manage user permissions on MySQL databases.

Procedure

1. Create a custom policy. For more information, see Create a custom policy.

Sample custom policies:

- Example 1: Authorize a RAM user to manage the two specified clusters.

  Assume that you have multiple clusters within your Alibaba Cloud account. You want to authorize a RAM user to use only two clusters whose IDs are i-001 and i-002. In this case, you can create the following policy:

```
{
  "Statement": [
    {
      "Action": "polardb:*",
      "Effect": "Allow",
      "Resource": [
                  "acs:polardb:*:*:*/i-001",
                  "acs:polardb:*:*:*/i-002"
                  ]
    },
    {
      "Action": "polardb:Describe*",
      "Effect": "Allow",
      "Resource": "*"
    }
  ],
  "Version": "1"
}
```

  Note

  - The authorized RAM user can view all the clusters and resources, but can manage only the two clusters whose IDs are i-001 and i-002. You can still manage the two clusters by using API operations, command-line interfaces (CLIs), or software development kits (SDKs).
  - The policy must include `Describe*`. Otherwise, the authorized RAM user cannot view clusters in the PolarDB console.

- Example 2: Authorize a RAM user to use only specific features of .

  If you want to authorize a RAM user to use only some features of , you can create the following policy:

```
{
    "Statement": [
        {
            "Action": [
              "polardb:Describe*",
              "polardb:CreateBackup",
              "polardb:DeleteBackup",
              "polardb:ModifyDBClusterAccessWhitelist"
            ],
            "Resource": "*",
            "Effect": "Allow"
        }
    ],
    "Version": "1"
}
```

> **Note**
>
> - The authorized RAM user can only query cluster information and backups, create and delete backups, and modify whitelists for all the clusters within your account.
> - allows you to specify whether RAM users can perform specific operations on PolarDB resources. You can specify API operations in policies for fine-grained permission management. For more information, see Services that work with RAM and API overview.

2. Attach the custom policy to a RAM user. For more information, see Grant permissions to a RAM user.

4.3. Database accounts

4.3.1. Create a database account

This topic describes how to create accounts and explains the differences between privileged accounts and standard accounts.

Context

You can create and manage privileged accounts and standard accounts in the console.

> **Note** To avoid security risks, does not provide root accounts.

Account type	Description

Account type	Description
Privileged Account	• You can use only the console to create and manage privileged accounts. • You can create only one privileged account for each cluster. A privileged account can manage all the standard accounts and databases in the corresponding cluster. • You can create a database and a standard account and authorize the standard account to perform add, delete, modify, and view operations on the database. • A privileged account has more permissions than before. This allows you to implement fine-grained control over user permissions based on your business requirements. For example, you can grant different users the permissions to query different tables. • A privileged account has the permissions to disconnect all standard accounts on the instance.
Standard Account	• You can create and manage standard accounts in the console or by using SQL statements. • You can create multiple standard accounts for each cluster. The maximum number of standard accounts that you can create depends on the database engine. • A standard account cannot be used to create databases or standard accounts. A standard account can only manage databases on which they have permissions. • A standard account does not have permissions to manage or disconnect the other accounts of the RDS instance on which the standard account is created.

Create a privileged account

1.
2.
3.
4. In the left-side navigation pane, choose **Settings and Management > Accounts**.
5. Click **Create Account**.
6. In the **Create Account** panel, specify the following parameters. The following table describes the parameters.

Parameter	Description
Account Name	Enter the username of the account. The username must meet the following requirements: ◦ It must start with a lowercase letter and end with a letter or a digit. ◦ It can contain lowercase letters, digits, and underscores (_). ◦ It must be 2 to 16 characters in length. ◦ It cannot be root, admin, or another username that is reserved by the system.

Parameter	Description
Account Type	Specify the type of the account. Select **Privileged Account**. **Note** If you have already created a privileged account, you cannot select **Privileged Account**. You can create only one privileged account for each cluster.
Password	Enter a password for the account. The password must meet the following requirements: ○ It must contain at least three of the following character types: uppercase letters, lowercase letters, digits, and special characters. ○ It must be 8 to 32 characters in length. ○ It can contain the following special characters: !@#$%^&*()_+-=
Confirm Password	Enter the logon password again.
Description	The information that can help you manage the account. It must meet the following requirements: ○ It cannot start with `http://` or `https://`. ○ It must be 2 to 256 characters in length.

7. Click **OK**.

Create a standard account

1.
2.
3.
4. In the left-side navigation pane, choose **Settings and Management > Accounts**.
5. Click **Create Account**.
6. In the **Create Account** panel, specify the following parameters.

Parameter	Description
Account Name	Enter the username of the account. The username must meet the following requirements: ○ It must start with a lowercase letter and end with a letter or a digit. ○ It can contain lowercase letters, digits, and underscores (_). ○ It must be 2 to 16 characters in length. ○ It cannot be root, admin, or another username that is reserved by the system.

Parameter	Description
Account Type	Specify the type of the account. Select **Standard Account**.
Databases	You can grant permissions on one or more databases to the account. You can leave this parameter empty. You can grant the account the database permissions after the account is created. i. Select one or more databases from the **Databases Not Assigned** list and click the icon. Then, the selected databases are added to the **Assigned Databases** list. ii. In the **Assigned Databases** list, specify the permissions on the selected databases. To specify the permissions, select one of the following options: **Read&Write**, **ReadOnly**, **DMLOnly**, **DDLOnly**, and **ReadOnly&Index**. **Note** If you need to customize the permissions or grant the account specific table permissions, click **Customize Permissions** below the **Databases Not Assigned** list. On the page that appears, you can use the permission management feature of Database Management Service (DMS) to manage the account permissions. For more information, see Manage user permissions on MySQL databases.
Password	Enter a password for the account. The password must meet the following requirements: ○ It must contain at least three of the following character types: uppercase letters, lowercase letters, digits, and special characters. ○ It must be 8 to 32 characters in length. ○ It can contain the following special characters: !@#$%^&*()_+-=
Confirm Password	Enter the logon password again.
Description	The information that can help you manage the account. It must meet the following requirements: ○ It cannot start with `http://` or `https://`. ○ It must be 2 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, you can reset the permissions to restore the privileged account to the initial state. To reset the permissions of the account, perform the following steps:

1.

2.
3.
4. In the left-side navigation pane, choose **Settings and Management > Accounts**.
5. On the page that appears, find the privileged account that you want to manage. In the **Actions** column for the privileged account, click **Reset Permissions**
6. In the dialog box that appears, enter the password of the privileged account. Then, click **OK** to reset the permissions of the account.

What to do next

Apply for a cluster endpoint or a primary endpoint

Related operations

API	Description
CreateAccount	Creates an account.
DescribeAccounts	Queries the accounts of a specified cluster.
ModifyAccountDescription	Modifies the description of a database account for a PolarDB cluster.
ModifyAccountPassword	Changes the password of a database account.
GrantAccountPrivilege	Grants a specified standard account the permissions on one or more databases of a specified PolarDB cluster.
RevokeAccountPrivilege	Revokes the permissions on one or more databases from a specified PolarDB standard account.
ResetAccount	Resets the permissions of a privileged account for a specified PolarDB cluster.

4.3.2. Manage database accounts for a cluster

This topic describes how to manage database accounts for a cluster. For example, you can reset permissions of the privileged account, modify permissions of standard accounts, change a password, and delete accounts.

Context

supports two types of accounts: privileged account and standard account. You can manage all accounts and databases in the console.

Precautions

- To ensure data security, you cannot create and use a root account in .
- If you execute the UPDATE or INSERT statement to modify permission tables in a MySQL database to change the password or permissions of an account, data cannot be synchronized to read-only nodes. We recommend that you use the following methods to change the password or permissions:

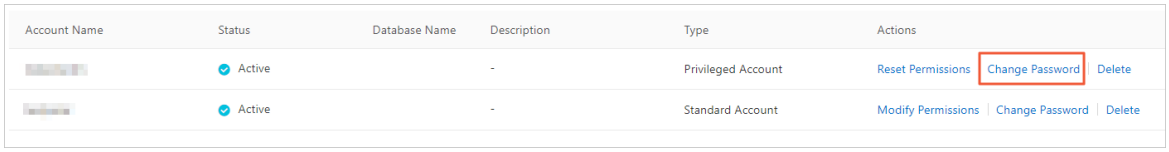
- Use the console to change the password or permissions. For more information, see Change the password of an account.
- Use the command-line interface (CLI) to change the password or permissions as a privileged user. For more information, see Run commands to change the password or permissions of an account.

Create a database account

For more information, see Create a database account.

Change the password of an account

1.
2.
3.
4. In the left-side navigation pane, choose **Settings and Management > Accounts**.
5. Find the target account and click **Change Password** in the **Actions** column.

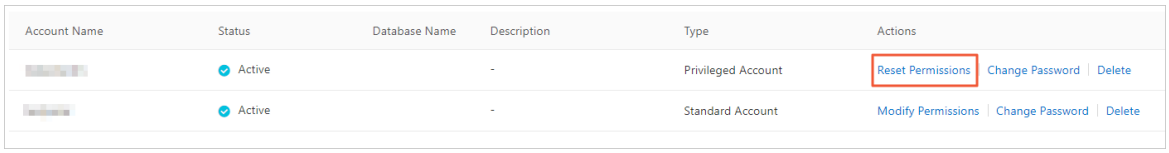


6. In the dialog box that appears, enter and confirm the new password, and click **OK**.

Reset permissions of the privileged account

If an issue occurs on the privileged account, for example, permissions are unexpectedly revoked, you can enter the password of the privileged account to reset permissions.

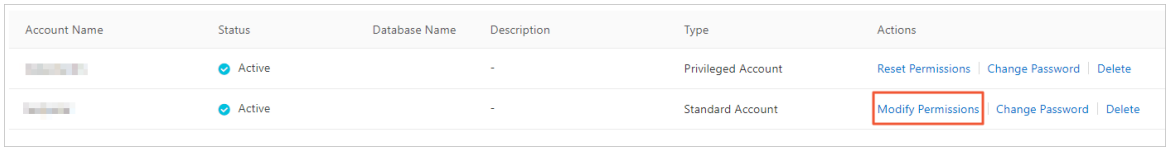
1.
2.
3.
4. In the left-side navigation pane, choose **Settings and Management > Accounts**.
5. Find the target account and click **Reset Permissions** in the **Actions** column.



6. In the dialog box that appears, click **OK** .

Modify the permissions of a standard account

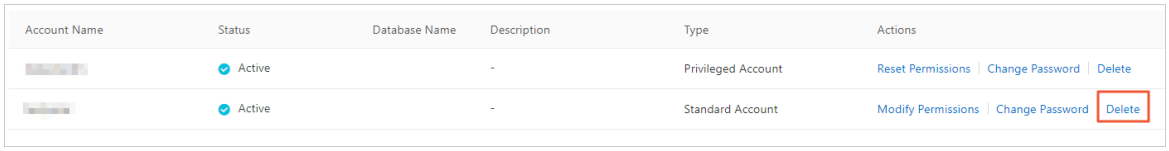
1.
2.
3.
4. In the left-side navigation pane, choose **Settings and Management > Accounts**.
5. Find the target account and click **Modify Permissions** in the **Actions** column.



6. In the dialog box that appears, modify permissions of authorized databases and unauthorized databases, and click **OK**.

Delete an account

1.
2.
3.
4. In the left-side navigation pane, choose **Settings and Management > Accounts**.
5. Find the target account and click **Delete** in the **Actions** column.



6. In the dialog box that appears, click **OK**.

Run commands to change the password or permissions of an account

- You can log on to the cluster with the privileged account and run the following command to change the password of an account:
  - 8.0:

```
ALTER USER {username} IDENTIFIED BY '{password}'
```

Parameter	Description
username	The account for which you want to change the password.
password	The password of the account.

- 5.6 or 5.7:

```
SET PASSWORD FOR 'username'@'host' = PASSWORD('password')
```

Parameter	Description
username	The account for which you want to change the password.
host	The host from which the account can be used to log on to the database. If you set this parameter to a percent sign (%), you can log on to the database from all hosts by using the account.
password	The password of the account.

- You can log on to the cluster with the privileged account and run the following command to change permissions of an account:

```
GRANT privileges ON databasename.tablename TO 'username'@'host' WITH GRANT OPTION;
```

Parameter	Description
privileges	The operations that are granted to the account, such as SELECT, INSERT, and UPDATE. If you set this parameter to ALL, you can manage all databases.
databasename	The name of the database. If you set this parameter to an asterisk (*), the account can be used to manage all databases.
tablename	The name of a table. If you set this parameter to an asterisk (*), the account can be used to manage all tables.
username	The account to be authorized.
host	The host from which the account can be used to log on to the database. If you set this parameter to a percent sign (%), you can log on to the database from all hosts by using the account.
WITH GRANT OPTION	Grants the GRANT command permissions to the account. This parameter is optional.

Related operations

API	Description
CreateAccount	Creates an account for a specified cluster.
DescribeAccounts	Queries the accounts of a specified cluster.

API	Description
ModifyAccountDescription	Modifies the description of an account for a specified cluster.
ModifyAccountPassword	Changes the password of an account for a specified cluster.
GrantAccountPrivilege	Grants access permissions on one or more databases in a specified cluster to an account.
RevokeAccountPrivilege	Revokes access permissions on one or more databases from an account for a specified cluster.
ResetAccount	Resets permissions of an account.
DeleteAccount	Deletes an account.

4.3.3. Account permissions

This topic describes the permissions of privileged accounts and standard accounts of .

Privileged accounts

The privileged accounts of have the following permissions.

Account type	Permission
Privileged account	SELECT, INSERT, UPDATE, DELETE, and CREATE DROP, RELOAD, PROCESS, REFERENCES, and INDEX ALTER, LOCK TABLES, EXECUTE, TRIGGER, and CREATE TEMPORARY TABLES REPLICATION SLAVE, REPLICATION CLIENT, CREATE VIEW, and SHOW VIEW CREATE ROUTINE, ALTER ROUTINE, CREATE USER, and EVENT

Standard accounts

The standard accounts of have the following permissions.

Permission type	Permission
Read and write	SELECT, INSERT, UPDATE, DELETE, CREATE, and EXECUTE DROP, REFERENCES, INDEX, ALTER, and CREATE TEMPORARY TABLES EVENT, CREATE VIEW, SHOW VIEW, CREATE ROUTINE, and ALTER ROUTINE LOCK TABLES, TRIGGER, PROCESS, REPLICATION SLAVE, and REPLICATION CLIENT

Permission type	Permission
Read-only	SELECT, LOCK TABLES, and SHOW VIEW PROCESS, REPLICATION SLAVE, and REPLICATION CLIENT
Data manipulation language (DML) only	SELECT, INSERT, UPDATE, DELETE, and LOCK TABLES CREATE TEMPORARY TABLES, EXECUTE, TRIGGER, and EVENT SHOW VIEW, PROCESS, REPLICATION SLAVE, and REPLICATION CLIENT
Data definition language (DDL) only	CREATE, DROP, INDEX, ALTER, and CREATE TEMPORARY TABLES CREATE VIEW, SHOW VIEW, CREATE ROUTINE, and ALTER ROUTINE LOCK TABLES, PROCESS, REPLICATION SLAVE, and REPLICATION CLIENT
Read-only and index	SELECT, INDEX, LOCK TABLES, and SHOW VIEW

4.3.4. System Accounts

provides multiple system accounts for operations such as background O&M. In most cases, you do not need to manage the permissions and authorized operations of these system accounts.

Account	Description
root (MySQL 5.6 and 8.0) aliyun_root (MySQL 5.7)	The account that is used to locally manage instances. For example, you can use this account to reconfigure the parameters that are related to the database engine and query the status of instances.
aurora	The account that is used to remotely manage instances. If an instance is faulty, you can use this account to log on to and manage the instance. For example, you can perform a primary/secondary switchover and monitor instances.
replicator	The account that is used to replicate or synchronize data between the primary and secondary instances.

Note Do not use one of the names of the preceding system accounts when you create a database account.

5.Data Security and Encryption

5.1. Configure a whitelist for a cluster

5.1.1. Configure an IP whitelist

After you create a cluster, you must configure an IP whitelist, and create an account for logging on to the cluster. Only IP addresses in the IP whitelists or Elastic Compute Service (ECS) instances in the security groups of the cluster can access the cluster. This topic describes how to configure an IP whitelist.

Scenarios

An IP whitelist contains IP addresses or CIDR blocks that are allowed to access a cluster. You can configure an IP whitelist to reinforce the security of a cluster. We recommend that you update the IP whitelist on a regular basis. In most cases, you must configure an IP whitelist in the following scenarios:

- You want to connect your ECS instance to a cluster. You can find the IP addresses of the ECS instance in the **Configuration Information** section on the **Instance Details** page. Then, add one of the IP addresses to the IP whitelist of the cluster.

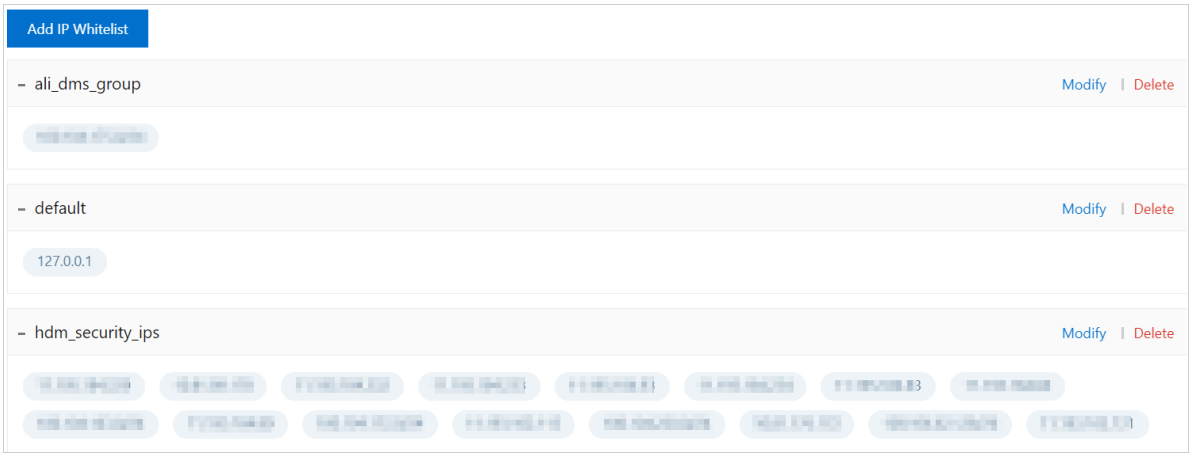
  > **Note** If the ECS instance and the cluster are deployed in the same region, such as the China (Hangzhou) region, add the private IP address of the ECS instance to the IP whitelist. If the ECS instance and the cluster are deployed in different regions, add the public IP address of the ECS instance to the IP whitelist. You can also migrate the ECS instance to the region where the cluster is deployed and then add the private IP address of the ECS instance.

- If you want to connect on-premises servers, computers, or other cloud instances to the cluster, add the relevant IP addresses to the IP whitelist of the cluster.

Precautions

- cannot automatically obtain the private IP addresses of ECS instances in virtual private clouds (VPCs). If you want to use the private IP address of an ECS instance to access a cluster, you must manually add the private IP address to the IP whitelist of the cluster.
- You can configure both IP whitelists and security groups. After you add IP addresses to IP whitelists and add ECS instances to security groups of a cluster, the specified IP addresses and ECS instances can access the cluster.
- The **ali_dms_group** (for Data Management), **hdm_security_ips** (for Database Autonomy Service), and **dtspolardb** (for Data Transmission Service) whitelists are automatically created when you use the relevant services. To ensure that the services can be used as normal, do not modify or delete these IP whitelists.

  > **Notice** Do not add your service IP addresses to these IP whitelists. Otherwise, your service IP addresses may be overwritten when the related services are updated. Consequently, service interruption may occur.

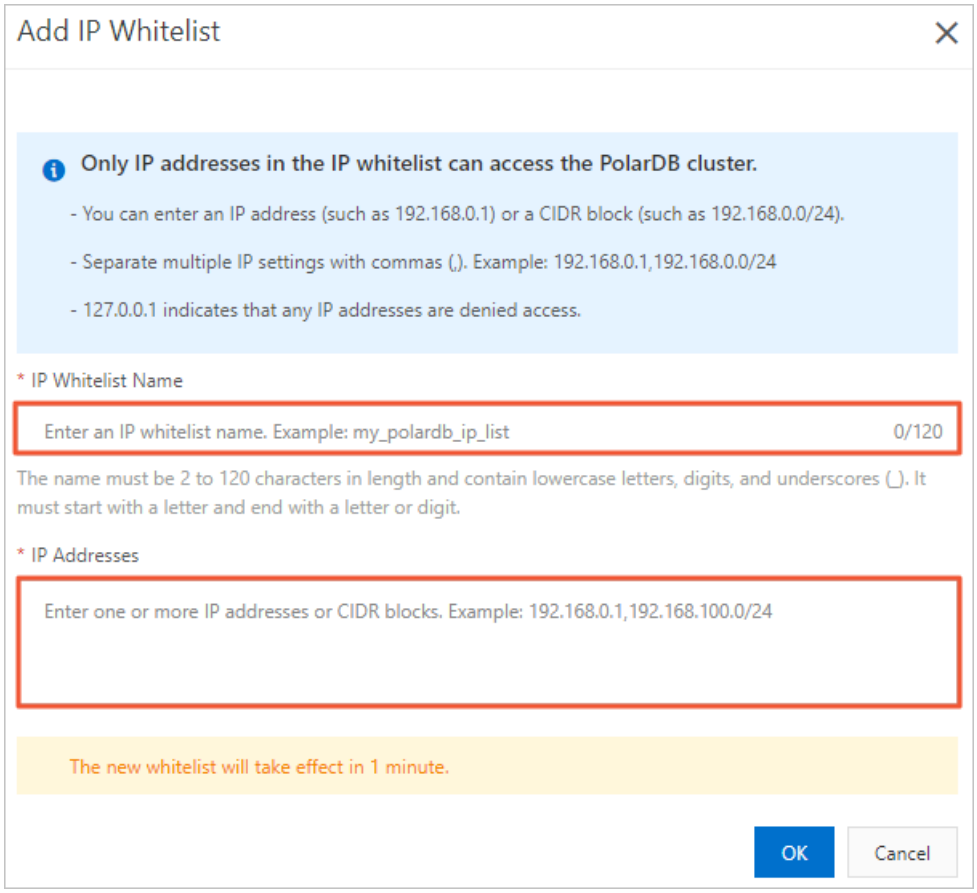


Procedure

1.
2.
3.
4. In the left-side navigation pane, choose **Settings and Management > Whitelists**.
5. On the **Whitelists** page, you can click **Add IP Whitelist** to add an IP whitelist or click **Modify** to modify an existing IP whitelist.

   - Add an IP whitelist
     a. Click **Add IP Whitelist**.

b. In the **Add IP Whitelist** panel, specify the name of the IP whitelist and enter the IP addresses that are allowed to access the cluster.

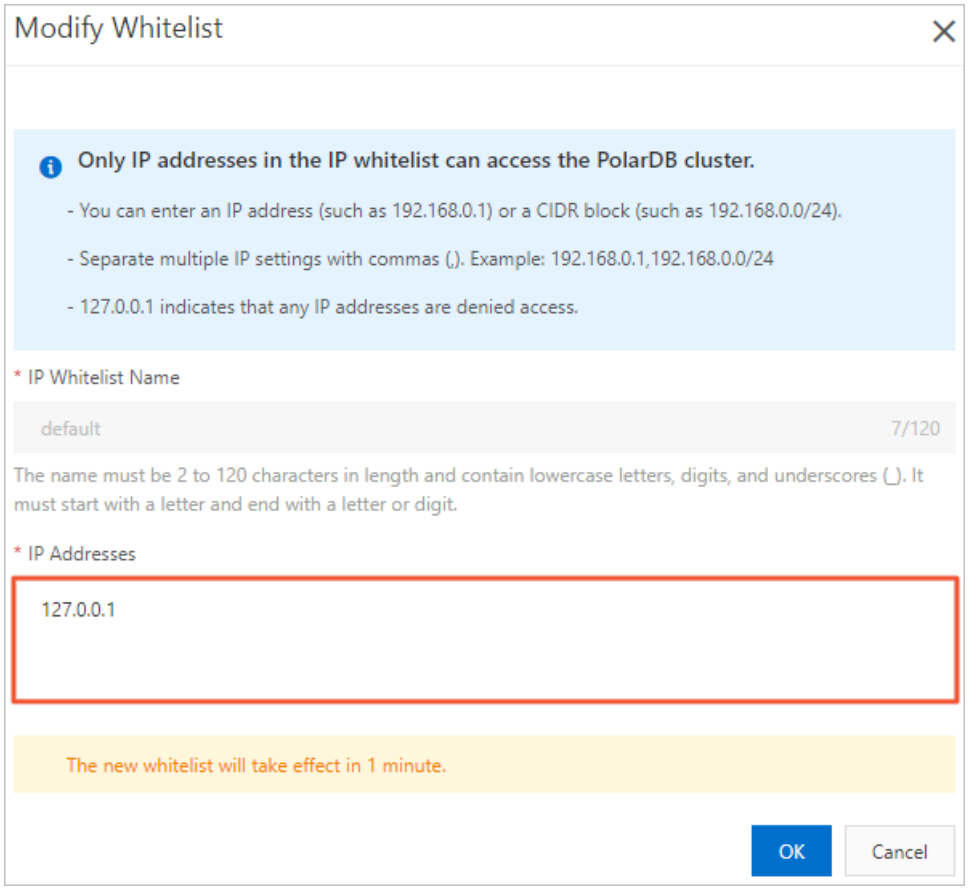


> **Note** The name of the IP whitelist must meet the following requirements:
> - The name can contain lowercase letters, digits, and underscores (_).
> - The name must start with a letter and end with a letter or digit.
> - The name must be 2 to 120 characters in length.

- Modify an IP whitelist

  a. On the right side of an IP whitelist name, click **Modify**.

b. In the **Modify Whitelist** panel, enter the IP addresses that are allowed to access the cluster.



> **Note**
>
> - A `default` IP whitelist that contains only the IP address `127.0.0.1` is automatically created for each cluster. This IP whitelist blocks all IP addresses.
> - If you set an IP whitelist to a percent sign (`%`) or `0.0.0.0/0`, all IP addresses are allowed to access the cluster. We recommend that you do not use this configuration unless necessary because it compromises database security.

6. Click **OK**.

> **Note** You can create at most 50 IP whitelists and add at most 1,000 IP addresses or CIDR blocks to the 50 IP whitelists.

Related API operations

API operation	Description
DescribeDBClusterAccessWhitelist	Queries the IP addresses that are allowed to access a specified PolarDB for MySQL cluster.

API operation	Description
ModifyDBClusterAccessWhitelist	Modifies the IP addresses that are allowed to access a specified PolarDB for MySQL cluster.

5.1.2. Configure a security group

An Elastic Compute Service (ECS) security group is a virtual firewall that is used to control the inbound and outbound traffic of ECS instances in the security group. This topic describes how to configure a security group.

Scenarios

After you create a cluster, you cannot connect to the cluster. You must configure a security group for the cluster. Then, the ECS instances in the security group can access the cluster.

> **Note**
>
> - For more information about security groups and how to configure a security group in the ECS console, see Create a security group.
> - You can configure both IP whitelists and security groups. After you add IP addresses to IP whitelists and add ECS instances to security groups of a cluster, the specified IP addresses and ECS instances can access the cluster.

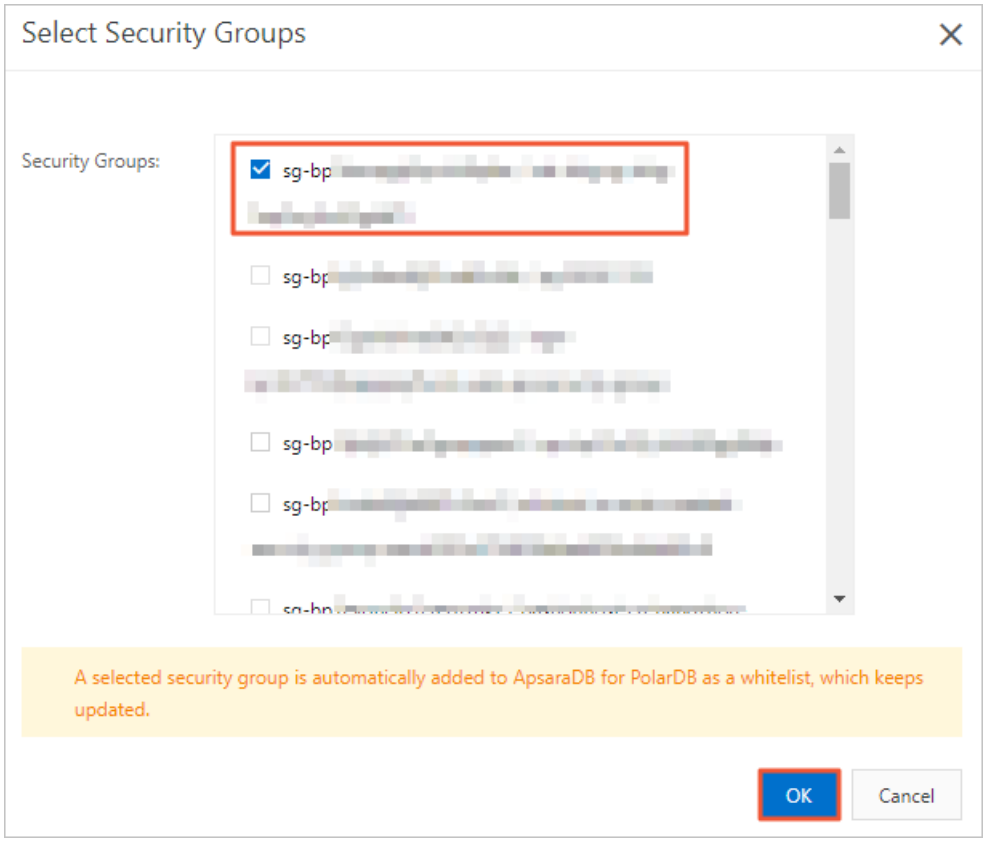
Precautions

- The network types of a cluster and its security groups must be the same. For example, if your PolarDB for MySQL cluster is deployed in a virtual private cloud (VPC), you can add only security groups of the VPC type.
- You can create at most 10 security groups for each cluster.

Procedure

1.
2.
3.
4. In the left-side navigation pane, choose **Settings and Management > Whitelists**.
5. On the **Whitelists** page, you can click **Select Security Group** to add a security group. You can also click **Modify** in the **Actions** column to change the security groups that you have added.

6. In the **Select Security Groups** panel, select one or more security groups and click **OK**.



Note For more information about how to create a security group, see Create a security group.

Related API operations

API operation	Description
DescribeDBClusterAccessWhitelist	Queries the IP addresses that are allowed to access a specified PolarDB for MySQL cluster.
ModifyDBClusterAccessWhitelist	Modifies the IP addresses that are allowed to access a specified PolarDB for MySQL cluster.

5.2. Configure SSL encryption

This topic describes how to make data transmission more security by configuring SSL encryption. You must enable SSL encryption and install SSL certificates that are issued by certificate authorities (CAs) in the required applications. SSL is used to encrypt connections at the transport layer and enhance the security and integrity of the transmitted data. However, SSL encryption increases the round-trip time.

Background information

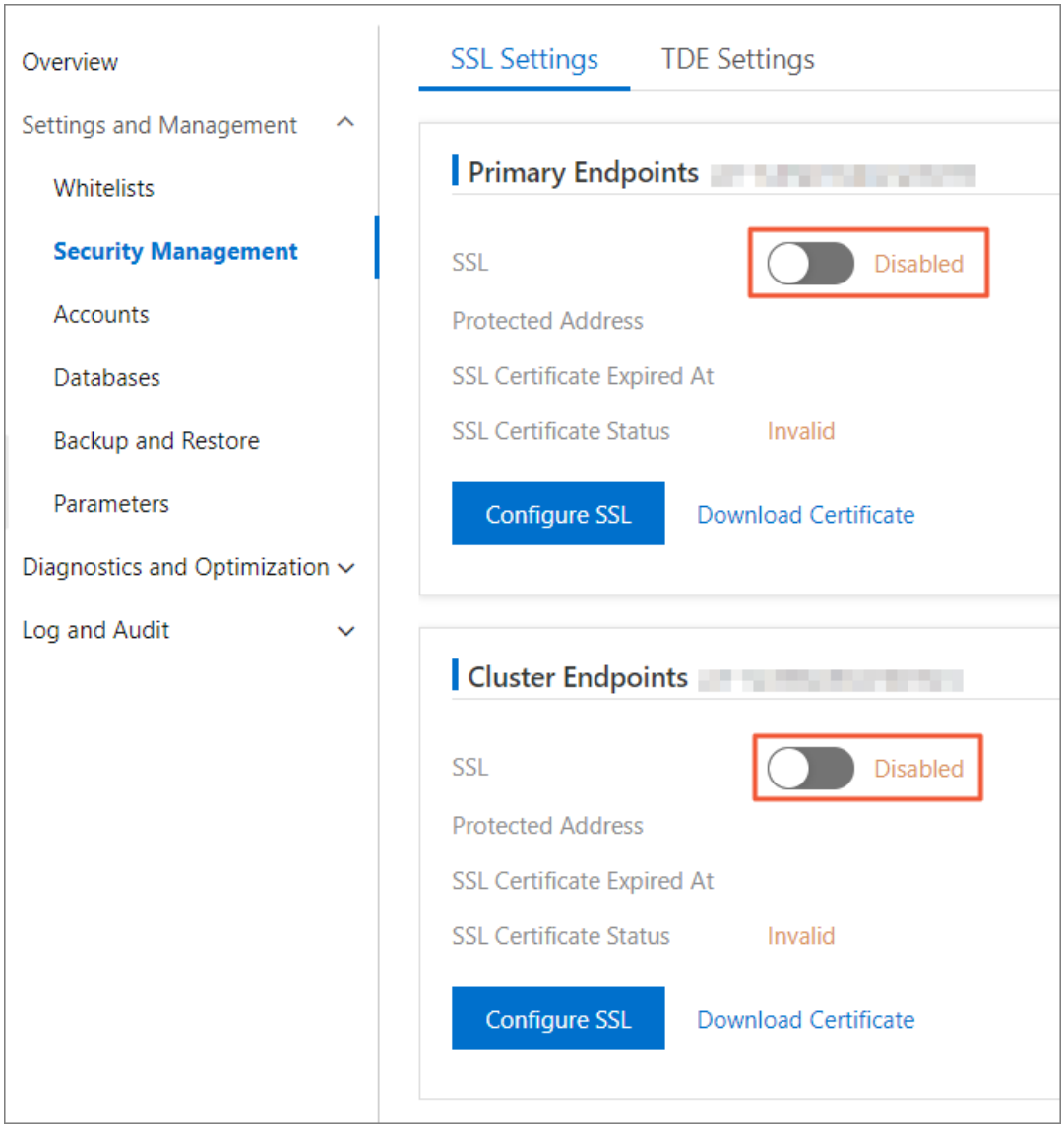
SSL is developed by Netscape to allow encrypted communication between a web server and a browser. SSL supports various encryption algorithms, such as RC4, MD5, and RSA. The Internet Engineering Task Force (IETF) upgraded SSL 3.0 to TLS. The term "SSL encryption" is still used in the industry. In this topic, SSL encryption refers to TLS encryption.

Precautions

- The validity period of an SSL certificate is one year. You must renew an SSL certificate before the SSL certificate expires. In addition, you must download the required SSL certificate file and configure the SSL certificate again after you renew the SSL certificate. Otherwise, clients that are connected to your PolarDB cluster through encrypted connections are disconnected. For more information about how to renew an SSL certificate, see Renew an SSL certificate.
- SSL encryption may cause a sharp increase in CPU utilization. We recommend that you enable SSL encryption only if you want to encrypt the connections that are established to the public endpoint of your instance. In most cases, connections that are established to the internal endpoint of your instance are secure and do not require SSL encryption.
- After you change the endpoint for which SSL encryption is enabled, the SSL certificate is automatically renewed and the cluster is restarted. Proceed with caution.
- After you renew an SSL certificate, the cluster is automatically restarted. Proceed with caution.
- To enable SSL encryption, the endpoint of the cluster must be less than 64 characters in length. For more information about how to modify an endpoint, see Configure PolarProxy.

Enable SSL encryption and download an SSL certificate

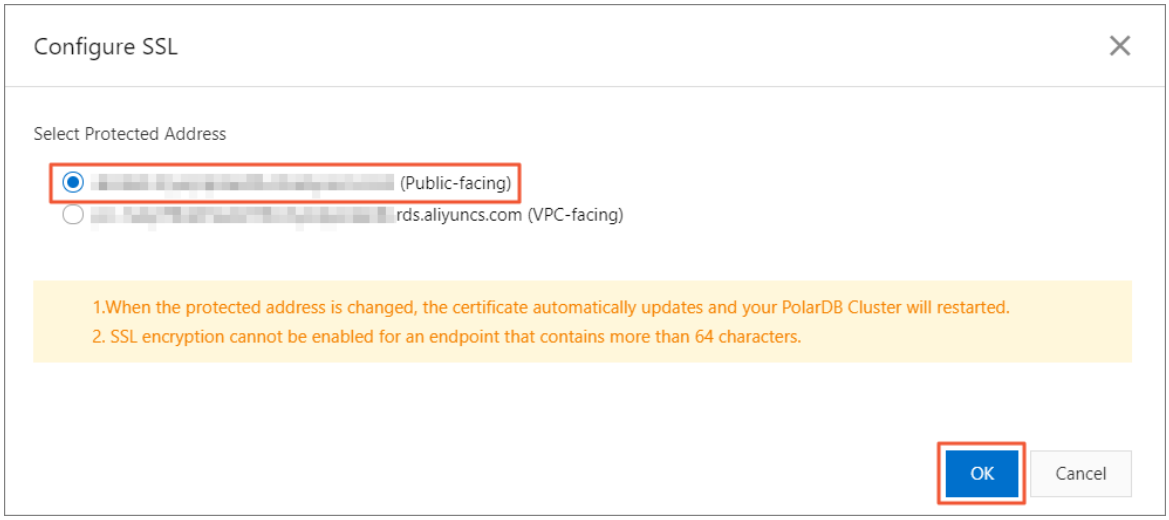
1.
2.
3.
4. In the left-side navigation pane, choose **Settings and Management > Security Management**.
5. On the **SSL Settings** tab, turn on **SSL** to enable SSL encryption.



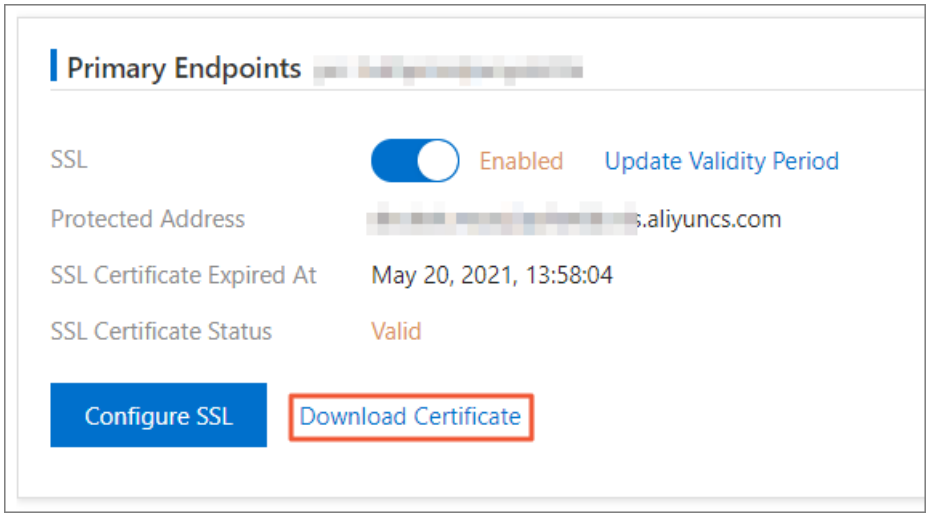
> **Note**
>
> - You can enable SSL encryption only for the primary endpoints of 5.6 and 5.7 clusters.
> - You can enable SSL encryption for the primary endpoints, cluster endpoints, and custom endpoints of 8.0 clusters.

6. In the **Configure SSL** dialog box, select the endpoint for which you want to enable SSL encryption and click **OK**.

> **Note** You can select a public endpoint or an internal endpoint as needed. However, you can select only one endpoint.



7. After the status of SSL encryption changes to **Enabled**, click **Download Certificate**.

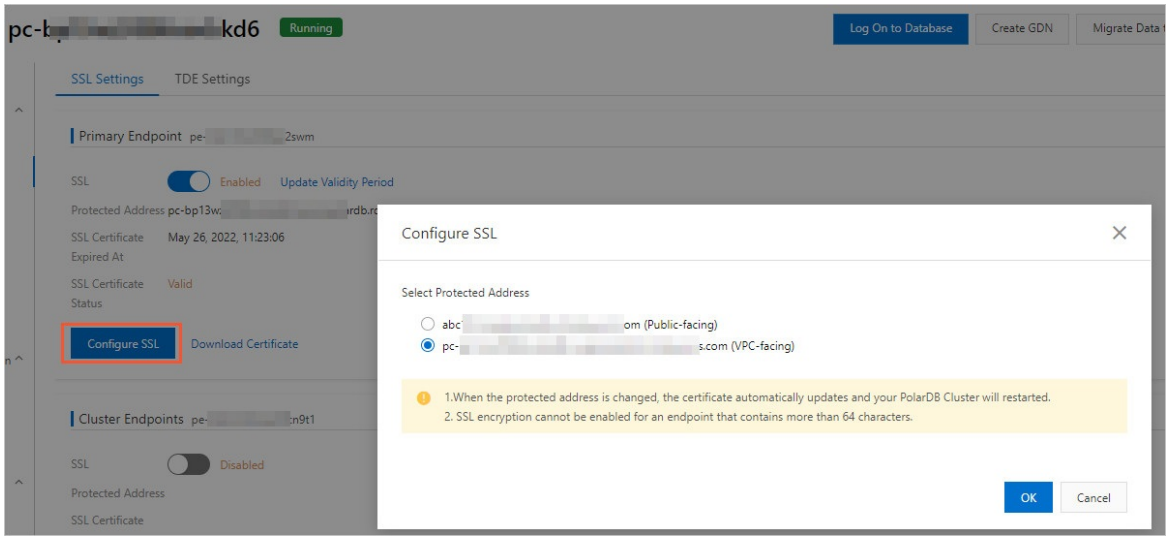


The downloaded package contains the following files:

- A P7B file. This file is used to import the CA certificate to a Windows system.
- A PEM file. This file is used to import the CA certificate to other operating systems or applications.
- A JKS file. This file is a truststore for Java. The password is apsaradb. The file is used to import the CA certificate chain to Java programs.

> **Note** When the JKS file is used in Java, you must modify the default JDK security configuration in JDK 7 and JDK 8. Open the `jre/lib/security/java.security` file on the server that is connected to the cluster and modify the following configurations:
>
> ```
> jdk.tls.disabledAlgorithms=SSLv3, RC4, DH keySize < 224
> jdk.certpath.disabledAlgorithms=MD2, RSA keySize < 1024
> ```
>
> If you do not modify these configurations, the following error is returned. In most cases, similar errors are caused by invalid Java security configurations.
>
> ```
> javax.net.ssl.SSLHandshakeException: DHPublicKey does not comply to algorithm con
> straints
> ```

If you want to change the endpoint for which SSL encryption is enabled, click **Configure SSL**.



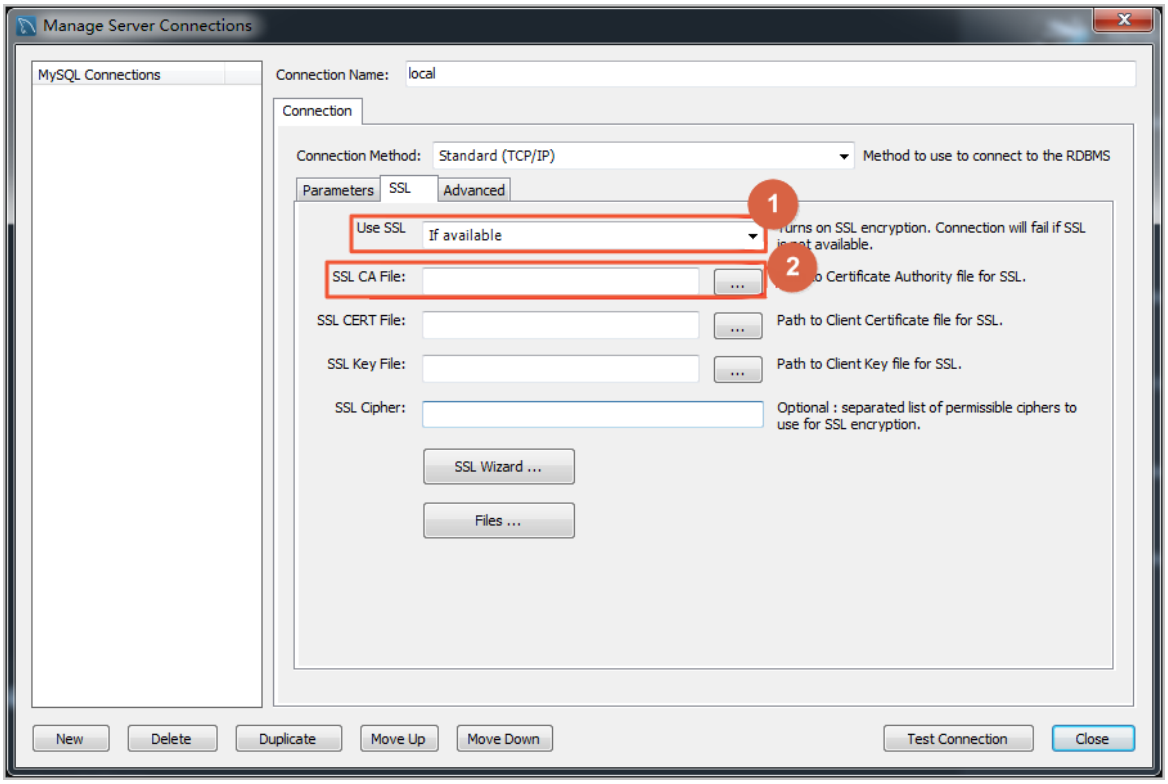
> **Notice** After you change the endpoint for which SSL encryption is enabled, the SSL certificate is automatically renewed and the cluster is restarted. Proceed with caution.

Configure an SSL certificate

After you enable SSL encryption, you must configure an SSL certificate. The SSL certificate is required for your application or client to connect to your cluster. In this section, MySQL Workbench and Navicat are used as examples to describe how to configure an SSL certificate. If you want to use other applications or clients, see the related instructions.

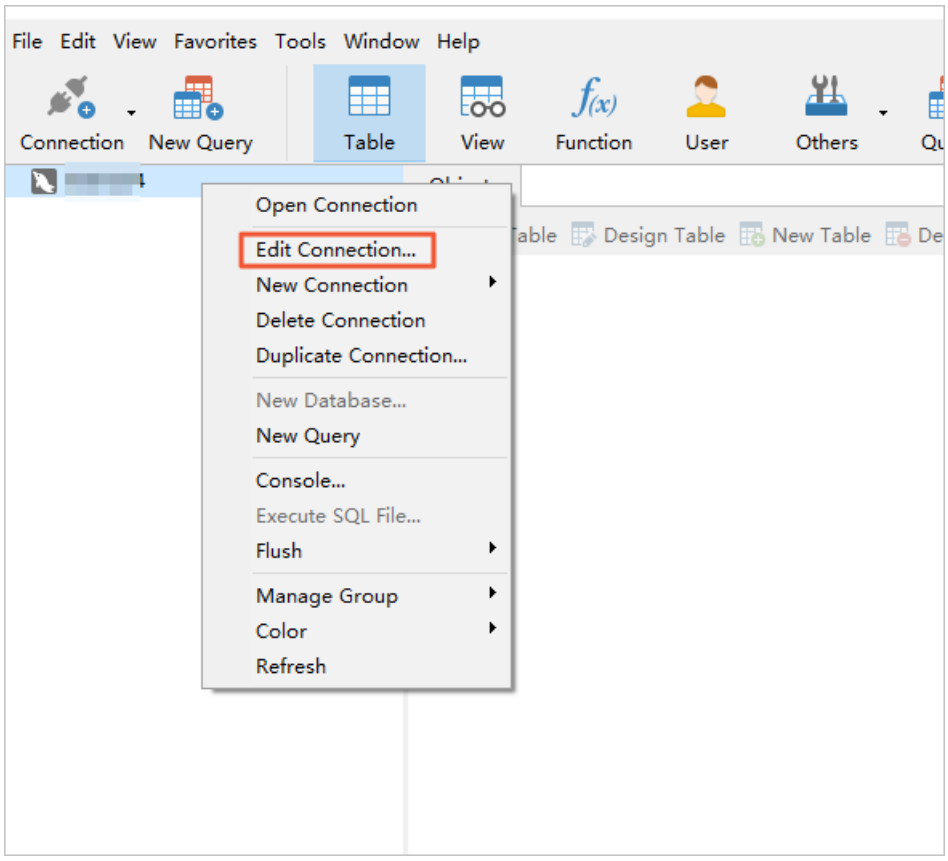
Perform the following steps to configure an SSL certificate on MySQL Workbench:

1. Start MySQL Workbench.
2. Choose **Database > Manage Connections**.
3. Enable **Use SSL** and import the SSL certificate file, as shown in the following figure.

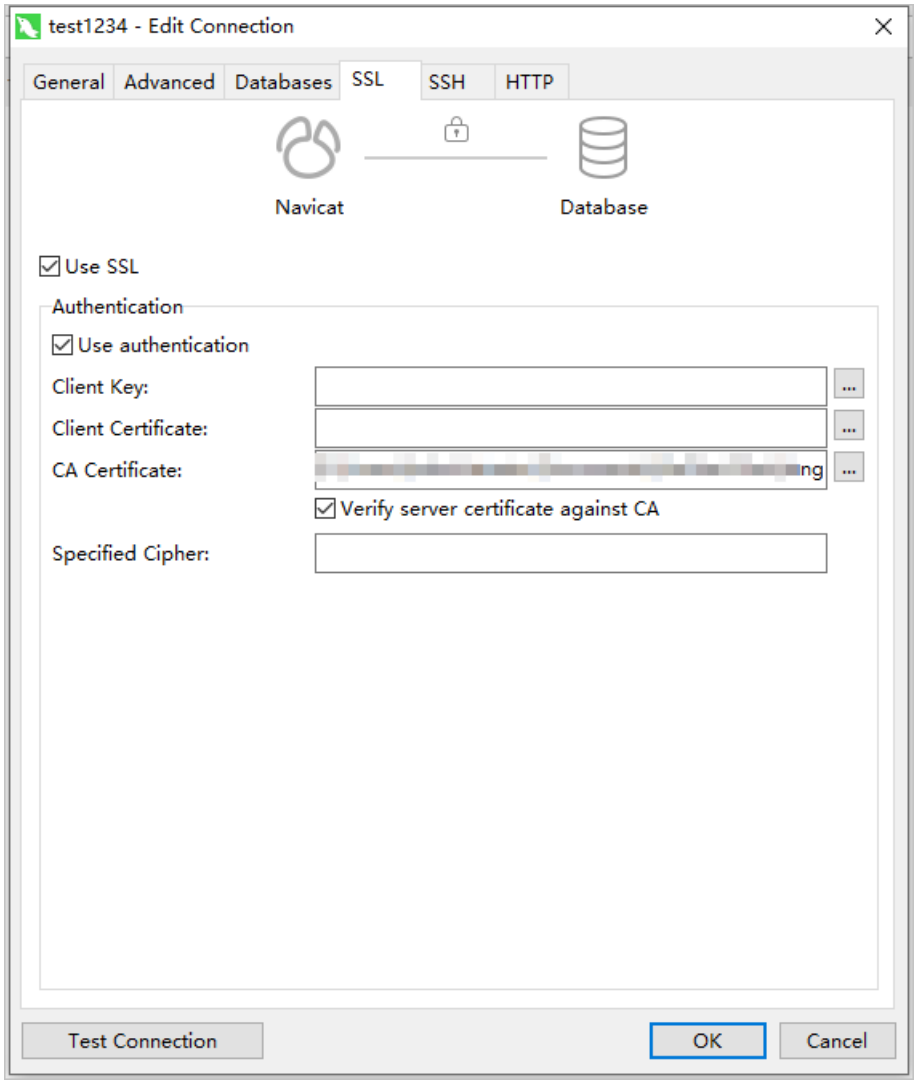


Perform the following steps to configure an SSL certificate on Navicat:

1. Start Navicat.
2. Right-click the database and click **Edit Connection**.



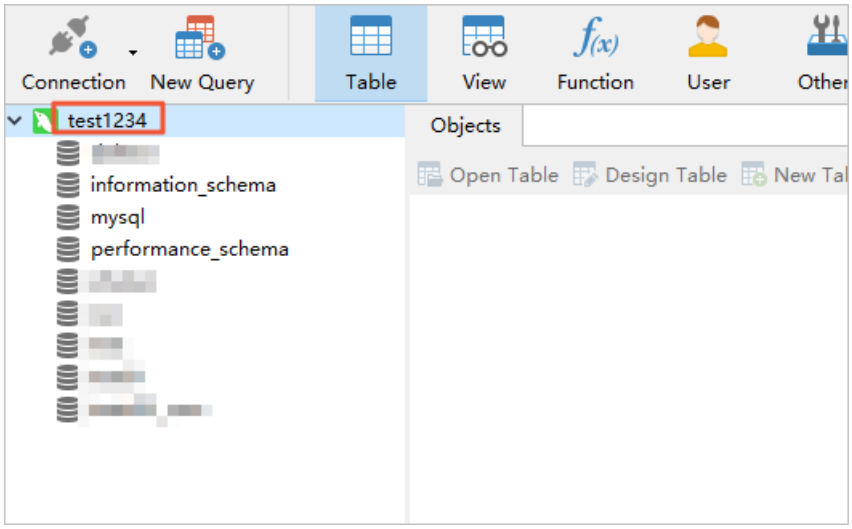
3. Click the **SSL** tab and select the path of the PEM certificate file, as shown in the following figure.



4. Click **OK**.

> **Note** If the system displays the `Connection with same connection name already exists in the project.` error, this indicates that you did not close the exiting connection. Close Navicat and open it again.

5. Double-click your database to check whether Navicat can connect to the database.

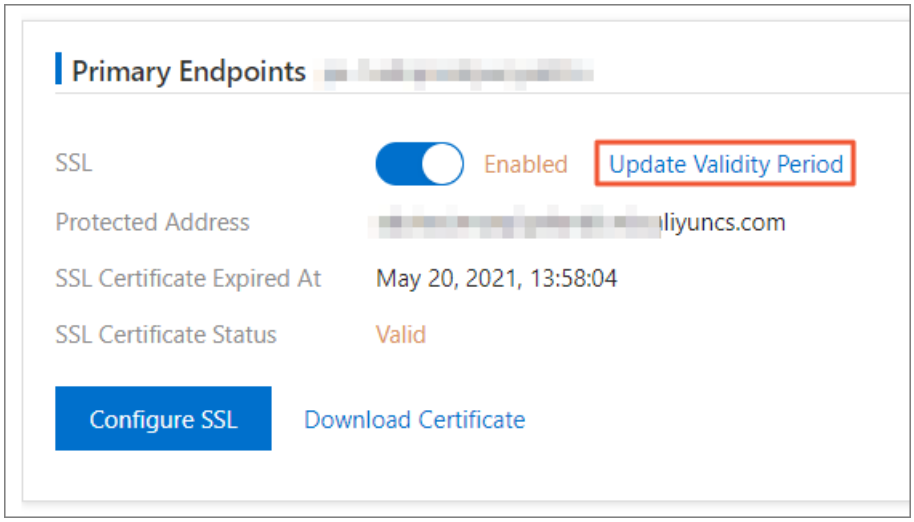


Renew an SSL certificate

This section describes how to renew an SSL certificate. After you change the endpoint for which SSL encryption is enabled or when the expiration date of the SSL certificate is close, you must renew the SSL certificate.

> **Note** After you renew an SSL certificate, the cluster is automatically restarted. Proceed with caution.

1.
2.
3.
4. In the left-side navigation pane, choose **Settings and Management > Security Management**.
5. On the **SSL Settings** tab, click **Update Validity Period**.



6. In the message that appears, click **OK**.
7. After the SSL certificate is renewed, download and configure the SSL certificate again.

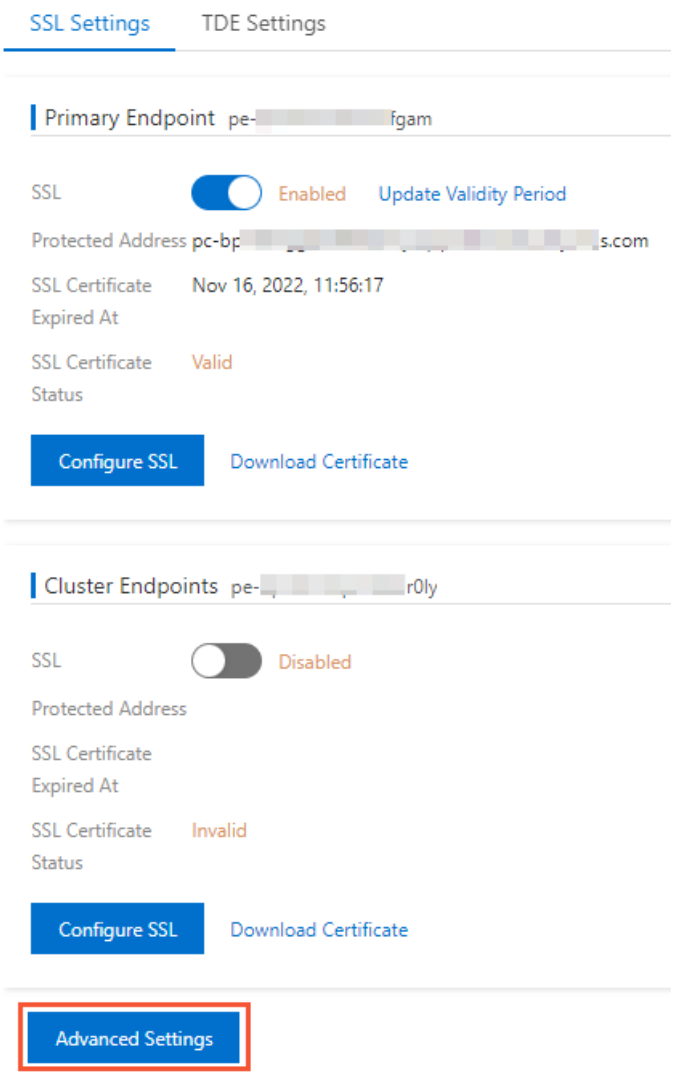
> **Note**
> - For more information about how to download an SSL certificate, see Step 7 in Enable SSL encryption and download an SSL certificate.
> - For more information about how to configure an SSL certificate, see Configure an SSL certificate.

Enable automatic certificate rotation

After you enable automatic certificate rotation, automatically renews an SSL certificate within 10 days before the SSL certificate expires. The certificate is renewed at a point in time within the maintenance window of the cluster.

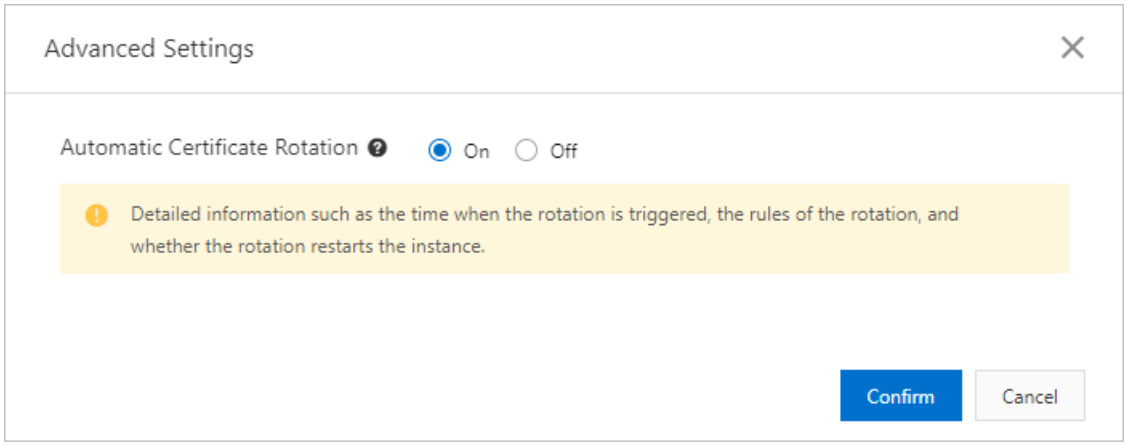
> **Note** If you enable automatic certificate rotation, your cluster is automatically restarted after the SSL certificate is renewed. Proceed with caution.

1. On the **SSL Settings** tab, click **Advanced Settings**.



2. In the **Advanced Settings** dialog box, select On for Automatic Certificate Rotation and click

Confirm.



Disable SSL encryption

> **Note**
> - After you disable SSL encryption, the cluster is restarted. We recommend that you perform this operation during off-peak hours.
> - After SSL encryption is disabled, the performance of your cluster is improved but data security is compromised. We recommend that you disable SSL encryption only in secure environments.

1.
2.
3.
4. In the left-side navigation pane, choose **Settings and Management > Security Management**.
5. On the **SSL Settings** tab, turn off **SSL** to disable SSL encryption.
6. In the message that appears, click **OK**.

Related API operations

API	Description
DescribeDBClusterSSL	Queries the SSL encryption settings of a specified cluster.
ModifyDBClusterSSL	Enables SSL encryption, disables SSL encryption, or renews the SSL certificate for a specified cluster.

5.3. Configure TDE

You can use Transparent Data Encryption (TDE) to encrypt data files when the files are written to disks and decrypt data files when the files are loaded to the memory from disks. If you use TDE, the sizes of the data files do not increase. Developers do not need to modify applications to use TDE.

Prerequisites

- Only clusters whose Editions are Cluster Edition or Single Node Edition support TDE. TDE is not supported by Archive Database Edition.

  clusters of the and versions must meet specific requirements. The following table describes the requirements based on the edition and version.

- Alibaba Cloud Key Management Service (KMS) is activated. For more information, see Activate KMS.
- ApsaraDB RDS is authorized to access KMS. For more information, see Authorize an ApsaraDB RDS for MySQL instance to access KMS.

Context

TDE for adopts the Advanced Encryption Standard (AES) algorithm. The key length is 256 bits. The keys that are used in TDE are generated and managed by KMS. does not provide keys or certificates. In some zones, you can use the keys that are automatically generated by Alibaba Cloud. You can also use your own key materials to generate keys. Then, authorize to use these keys.

Note

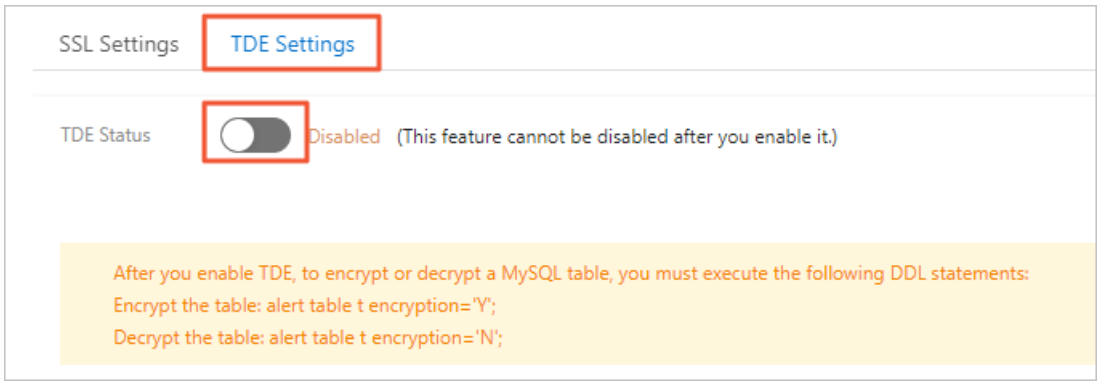
- In I/O bound scenarios, TDE may adversely affect the performance of your databases.
- You cannot enable TDE for clusters that are connected to a global database network (GDN). Clusters for which TDE is enabled cannot be connected to a GDN.

Procedure

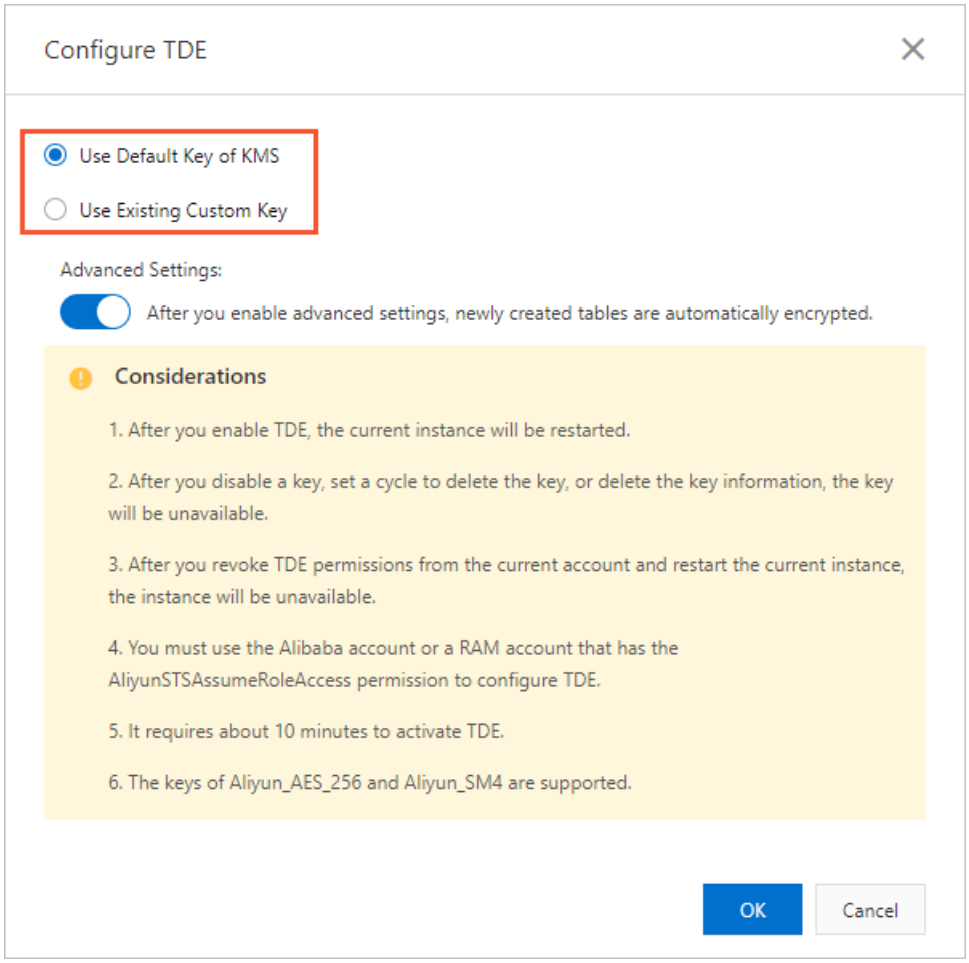
Notice

- After you enable TDE for a PolarDB cluster, the cluster is automatically restarted. Proceed with caution.
- After TDE is enabled, you cannot disable TDE.

1.
2.
3.
4. In the left-side navigation pane, choose **Settings and Management > Security Management**.
5. On the **TDE Settings** tab, turn on **TDE Status**.

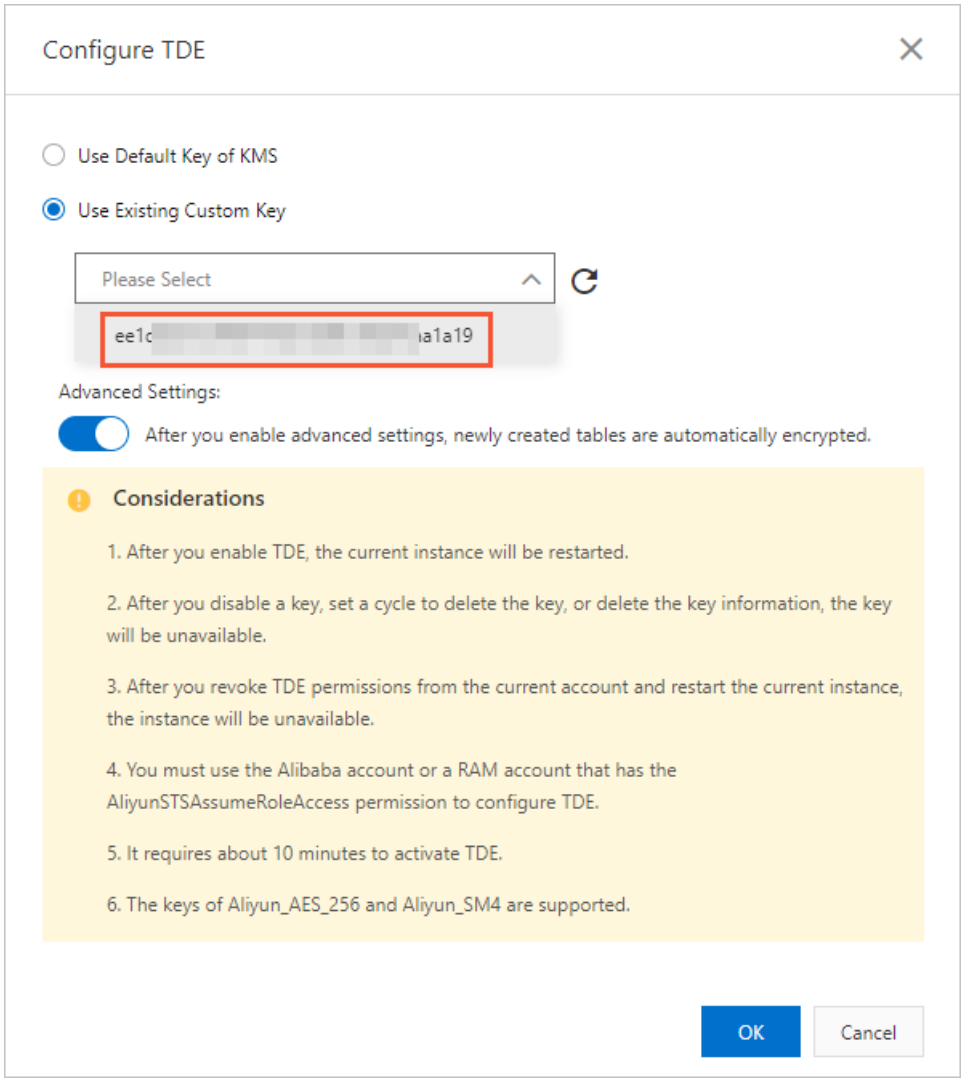


6. In the **Configure TDE** dialog box, select **Use Default Key of KMS** or **Use Existing Custom Key**.



Note TDE supports the following keys: `Aliyun_AES_256` and `Aliyun_SM4`.

- In the dialog box that appears, select **Use Default Key of KMS** and click **OK**.
- If you choose **Use Existing Custom Key**, select a key generated by KMS from the drop-down list and click **OK**.



Note

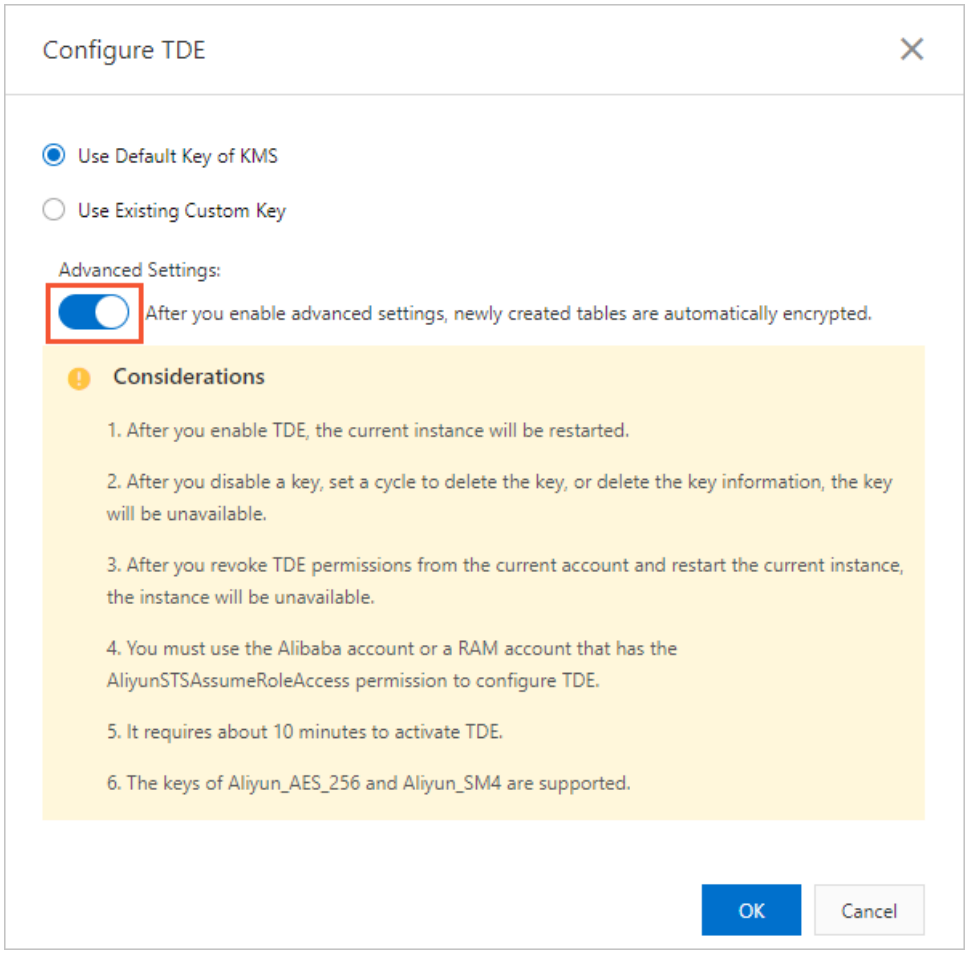
- If you do not have a custom key, click **Create Custom Key**. In the KMS console, create a key and import your key materials. For more information, see Create a CMK.
- When you use an existing custom key, you must take note of the following limits:
  - If you disable a key, configure a key deletion plan, or delete the key materials, the key becomes unavailable.
  - If you revoke the authorization to a cluster of the ApsaraDB PolarDB MySQL-compatible edition, the cluster becomes unavailable after you restart the cluster.
  - You must use an Alibaba Cloud account or an account that has the AliyunSTSAssumeRoleAccess permission.

It requires approximately 10 minutes to enable TDE.

Advanced settings

> **Note** You can enable the **Advanced Settings** feature only when the cluster version is 8.0 and the minor kernel version is 8.0.1.1.15 or later.

When you enable TDE, you can enable the **Advanced Settings** feature in the **Configure TDE** dialog box. After this feature is enabled, all newly created tables are automatically encrypted.



Encrypt and decrypt tables

> **Note** If you turn on **Advanced Settings**, created tables are automatically encrypted and you do not need to manually encrypt the created tables. For existing tables, you need to perform specific operations to encrypt data.

To encrypt or decrypt MySQL tables after you enable TDE, you must log on to the database and execute the relevant DDL statements. The following table lists the DDL statements that are executed to encrypt and decrypt tables in the ApsaraDB PolarDB MySQL-compatible edition of different kernel versions.

Operation	ApsaraDB PolarDB MySQL-compatible edition 5.6	ApsaraDB PolarDB MySQL-compatible edition 5.7 & ApsaraDB PolarDB MySQL-compatible edition 8.0

Operation	ApsaraDB PolarDB MySQL-compatible edition 5.6	ApsaraDB PolarDB MySQL-compatible edition 5.7 & ApsaraDB PolarDB MySQL-compatible edition 8.0
Encryption	`alter table <tablename> block_format=encrypted;`	`alter table <tablename> encryption= 'Y';`
Decryption	`alter table <tablename> block_format=default;`	`alter table <tablename> encryption= 'N';`

Note When you execute the preceding **alter table** statements to encrypt or decrypt a table, the table is locked.

5.4. SQL firewalls

5.4.1. Overview

PolarProxy provides the SQL firewall feature, which can identify SQL statements to be allowed and blocked after you configure blacklist and whitelist rules. This topic describes the SQL firewall feature.

Background information

When your database is illegally accessed, attackers may obtain all data in the database. Even in routine O&M, if you execute a **DELETE** statement which does not contain `WHERE` clauses, a table in the database may be accidentally deleted.

To prevent such high-risk operations, PolarProxy provides the SQL firewall feature, which enables you to configure blacklist and whitelist rules to allow and block SQL statements.

Prerequisites

PolarProxy must be V2.5.1 or later. For more information about how to view and upgrade the version of PolarProxy, see Version Management.

Note If you want to use this feature when PolarProxy is earlier than V2.5.1, to contact technical support to upgrade PolarProxy.

Limits

- This feature is applicable only to newly purchased clusters. Existing clusters are not supported.
- After the SQL firewall feature is enabled, PolarProxy creates the `proxy_auditing` database and two tables in the current cluster: `sql_list` and `org_sql_list`. The former table saves parameterized SQL statements in blacklist and whitelist rules and the latter table saves SQL statements in blacklist rules. Each table can save up to 500,000 SQL statements.
- The first time you create a rule, the rule is applicable only to new connections. After the rule is modified, it becomes applicable to existing connections after five seconds.

- The SQL statements in a blacklist rule created in custom parameterized SQL mode or custom SQL mode are not blocked when the rule is created, but are blocked after the rule is created.
- For the same account and endpoint, you cannot enable both a blacklist rule and a whitelist rule.
- Multi-statements are not supported. All multi-statements are blocked.
- SQL statements that cannot be parsed by PolarProxy are recorded in firewall audit logs but are not blocked. For example, if a **SELECT** statement contains syntax errors, the statement will not be blocked.
- The SQL statements in a blacklist rule created in custom parameterized SQL mode or custom SQL mode and in a whitelist are retained even if the blacklist or whitelist rule is disabled or deleted. If the blacklist or whitelist rule is enabled again, the rule is still applicable to the same accounts. If you want to completely delete the blacklist or whitelist rule, you can connect to the primary node in the cluster by using the super administrator account and delete the SQL statements.
- Blacklist and whitelist rules can only block or allow common SQL commands (3) or PREPARE commands (22). Other commands are not blocked or allowed. For example, after you connect to a cluster by using the MySQL command line, the **USE db;** statement is converted into a **COM_INIT_DB(1)** command by the MySQL command line. This command is not blocked.

Impacts on performance

- After you enable the blacklist rule feature, performance overheads of less than 10% are incurred.
- If the whitelist rule feature is enabled and no alert logs are generated, performance overheads of about 10% are incurred.
- If alert logs are generated for all executed SQL statements, the request rate decreases by 20% to 30% in normal business conditions. However, not all SQL statements can trigger alert logs. Only some can trigger alert logs. Therefore, in normal business conditions, the blacklist or whitelist rule feature does not have significant impact on the request rate.

Blacklist or whitelist rule feature

- **Blacklist rule feature**: You can configure blacklist rule to block specified types of SQL statements or specific SQL statements. When an SQL statement that is added to the blacklist rule is executed, PolarProxy blocks the SQL statement. For more information, see Configure blacklist rules.
- **Whitelist rule feature**: Business-related SQL statements are trained before they can be allowed by a whitelist rule. After the protection mode is enabled, only SQL statements that are added to the whitelist rule can be allowed. For more information, see Configure whitelist rules.

5.4.2. Configure blacklist rules

You can create, modify, delete, enable, and disable blacklist rules in the console. This topic describes the concept of blacklist rules and how to configure blacklist rules.

Blacklist rules

PolarProxy allows you to configure blacklist rules to block specified types of SQL statements or specific SQL statements.

You can configure blacklist rules in the following ways:

- **Fixed rule mode**: You can configure common blacklist rules in the console. Each rule can be effective for an account or a cluster. For more information about common rules, see Parameters for a blacklist rule.

- **Custom parameterized SQL mode**: You can parameterize all variables in SQL statements that you execute in a database, generate a parameterized template, and record the template in the database. PolarProxy blocks SQL statements that meet the parameterized template.
- **Custom SQL mode**: You can specify SQL statements to be blocked without parameterizing their variables. SQL statements that use other parameters are not blocked.

Add a blacklist rule

1.
2.
3.
4. In the left-side navigation pane, choose **Settings and Management > Security Management**.
5. On the **SQL Firewall** tab, click **Add** in the upper-left corner.
6. In the **Create a Rule** dialog box, set the parameters based on the mode that you select.
   - **Fixed rule mode.**
     a. If you select the **fixed rule mode**, set the following parameters. Parameters for a blacklist rule

Parameter		Required	Description
Basic Information	**Rule Name**	Yes	The name of the rule. The name must meet the following requirements: ■ The name can contain digits and letters. ■ The name can be up to 30 characters in length.
	Description	No	The description of the rule. **Note** The description can be up to 64 characters in length.
	Endpoint	Yes	The endpoint to which the current rule is applied.
	Rule Type	Yes	The type of the rule. Select **Blacklist Rule**.
	Current Mode	No	The mode of the rule. Set the value to **Protection Mode**. PolarProxy blocks SQL statements that meet the blacklist rule.

Parameter		Required	Description
	Database Account Name	No	The name of the database account to which the rule is applied. Valid values: ■ **All Accounts**: indicates that the rule applies to all database accounts in the cluster. The text box on the right must be left empty. ■ **Include**: indicates that the rule applies only to specified database accounts. You must specify at least one database account name in the text box on the right. Separate multiple accounts with commas (,). ■ **Exclude**: indicates that the rule applies only to database accounts that are not specified here. You must specify at least one database account name in the text box on the right. Separate multiple accounts with commas (,). **Note** The database account name can be in one of the following formats: ■ `Username`. Example: `user`. ■ `Username@IP address`. Example: `user@10.0.0.0`.
	Block SQLs With Asterisks (*)	No	Specifies whether to block SQL statements that contain asterisk signs (`*`). Valid values: ■ **Enable**: blocks SQL statements that contain asterisk signs (`*`). ■ **Disable**: does not block SQL statements that contain asterisk signs (`*`).

Parameter		Required	Description
Configurations	**Block SQLs of Specific Types**	No	Specifies whether to block SQL statements of specific types. Valid values: ■ **Enable**: blocks SQL statements of specific types. If you select **Enable**, select at least one type. The following types are supported: ■ `CREATE` ■ `DROP` ■ `ALTER` ■ `TRUNCATE` ■ `RENAME` ■ `INSERT` ■ `UPDATE` ■ `SELECT` ■ `DELETE` ■ **Disable**: does not block SQL statements of specific types.
	Block SQLs Without WHERE	No	Specifies whether to block SQL statements that do not contain WHERE clauses. Valid values: ■ **Enable**: blocks SQL statements of specific types that do not contain WHERE clauses. If you select **Enable**, select at least one type. The following types are supported: ■ `UPDATE` ■ `SELECT` ■ `DELETE` ■ **Disable**: does not block SQL statements of specific types that do not contain WHERE clauses. **Note** This parameter is valid only for the `SELECT`, `UPDATE`, and `DELETE` statements that contain at least one table name. The `SELECT 1;` statement is not blocked by PolarProxy.

Parameter		Required	Description
	Block SQLs With Specific Columns	No	Specifies whether to block SQL statements that contain specific column names. Valid values: ■ **Enable**: blocks SQL statements that contain specific column names. If you select **Enable**, the following options are supported: ■ **All**: indicates that the rule applies to all column names in the cluster. The text box on the right must be left empty. ■ **Include**: indicates that the rule applies only to specified column names. You must specify at least one database column name in the text box on the right. Separate multiple column names with commas (,). ■ **Exclude**: indicates that the rule applies only to column names that are not specified here. You must specify at least one column name in the text box on the right. Separate multiple column names with commas (,). ■ **Disable**: does not block SQL statements that contain specific column names.
	Block SQLs With Specific Functions	No	Specifies whether to block SQL statements that contain specific functions. Valid values: ■ **Enable**: blocks SQL statements that contain specific functions. If you select **Enable**, the following options are supported: ■ **All**: indicates that the rule applies to all functions in the cluster. The text box on the right must be left empty. ■ **Include**: indicates that the rule applies only to specified functions. You must specify at least one function in the text box on the right. Separate multiple functions with commas (,). ■ **Exclude**: indicates that the rule applies only to functions that are not specified here. You must specify at least one function in the text box on the right. Separate multiple functions with commas (,). ■ **Disable**: does not block SQL statements that contain specific functions.

Parameter		Required	Description
	Block SQLs With Specific Columns and Specific Functions	No	Specifies whether to block SQL statements that contain specific functions and specific column names. Valid values: ■ **Enable**: blocks SQL statements that contain specific functions and specific column names. If you select **Enable**, you must specify at least one **function** and one **column name** in the text box on the right. ■ If you set **Function Name** to **Include** and **Column Name** to **Include**, the rule is effective for SQL statements that contain specific functions and specific column names. ■ If you set **Function Name** to **Include** and **Column Name** to **Exclude**, the rule is effective for SQL statements that contain specific functions and that do not contain specific column names. ■ If you set **Function Name** to **Exclude** and **Column Name** to **Include**, the rule is effective for SQL statements that do not contain specific functions and that contain specific column names. ■ If you set **Function Name** to **Exclude** and **Column Name** to **Exclude**, the rule is effective for SQL statements that do not contain specific functions or specific column names. ■ **Disable**: does not block SQL statements that contain specific functions and specific column names.

b. Click **OK**.

- **Custom parameterized SQL mode**

a. If you select **custom parameterized SQL mode**, set the required parameters. For more information about the parameters, see Parameters for a blacklist rule.

> **Note** When you select **custom parameterized SQL mode**, you can disable all blacklist rules in the **Configurations** section.

b. Click **OK**.

c. Connect to the specified database endpoint by using the previously defined **database account name**. You can specify a SQL statement to be blocked by adding the following `hint` command before the SQL statement: `hint(/* store_to_blacklist */)`. For example, to block the `select id from sqlblack_test where id = 1;` statement, run the following command:

```
/* store_to_blacklist */ select id from sqlblack_test where id = 1;
```

The parameterized template:

```
select id from sqlblack_test where id = ?
```

`?` indicates any value.

Wait for five seconds. When the account executes a SQL statement that meets the preceding parameterized template on the specified cluster, PolarProxy blocks the statement. The following information is displayed after a SQL statement is blocked:

```
ERROR 1141 (HY000): This SQL is rejected by SQL Firewall. Access denied for user
'xxx'@'x.x.x.x' to database 'xzh': This SQL is in blacklist bl_test.
```

`bl_test` is the name of the blacklist rule table.

> **Note**
> - If you use the MySQL command line, you must add the `-c` option. Otherwise, the `hint` command does not take effect.
> - The parameterized SQL statement takes effect after five seconds.

- **Custom SQL mode**

a. If you select **custom SQL mode**, set the required parameters. For more information about the parameters, see Parameters for a blacklist rule.

> **Note** When you select **custom SQL mode**, you can disable all options in the **Configurations** section.

b. Click **OK**.

c. Connect to the specified database endpoint by using the previously defined **database account name**. You can specify a SQL statement to be blocked by adding the following `hint` command before the SQL statement: `hint(/* orginal_store_to_blacklist */)`. For example, to block the `update t set k = 2 where id = 2;` statement, run the following command:

```
/* orginal_store_to_blacklist */ update t set k = 2 where id = 2;
```

Wait for five seconds. When the account executes the `update t set k = 2 where id = 2;` statement on the specified cluster, PolarProxy blocks the statement. However, SQL statements that use other parameters are not blocked. The following information is displayed after a SQL statement is blocked:

```
ERROR 1141 (HY000): This SQL is rejected by SQL Firewall. Access denied for user
'xxx'@'x.x.x.x' to database 'xzh': This SQL is in blacklist bl_test.
```

`bl_test` is the name of the blacklist rule table.

> **Note**
> - If you use the MySQL command line, you must add the `-c` option. Otherwise, the `hint` command does not take effect.
> - The parameterized SQL statement takes effect after five seconds.

Enable or disable a blacklist rule

1.
2.
3.
4. In the left-side navigation pane, choose **Settings and Management > Security Management**.
5. On the **SQL Firewall** tab, find the rule and turn on or off **Enable/Disable**.

> **Note** You can select multiple rules in the rule table and then click **Enable** or **Disable** to batch enable or disable the rules.

6. In the **Enable** or **Disable** message, click **OK**.

> **Note** When you disabled a blacklist rule created in custom parameterized SQL mode or custom SQL mode, the SQL statements in the blacklist rule table in the database are retained even if the blacklist rule is disabled. If the blacklist rule is enabled again, the rule is still applicable to the same accounts. If you want to completely disable the blacklist rule, you can connect to the primary node in the cluster by using the super administrator account and delete the SQL statements from the `proxy_auditing.sql_list` table. When you delete the SQL statements this way, the SQL statements will not be blocked after five seconds. When you delete the SQL statements from the proxy_auditing.sql_list table, do not execute the `DROP` statement to delete the table.

Modify a blacklist rule

1.
2.
3.
4. In the left-side navigation pane, choose **Settings and Management > Security Management**.
5. On the **SQL Firewall** tab, find the rule and click **Modify** in the **Actions** column. In the **Modify a Rule** dialog box, modify the parameters based on your business requirements. For more information about the parameters, see Parameters for a blacklist rule.

> **Note** When you modify a rule, you cannot modify the **rule name**.

6. Click **OK**.

> **Note** The parameterized SQL statements in a rule created in **custom parameterized SQL mode** rule and the SQL statements in a rule created in **custom SQL mode** cannot be modified in the console. You must delete the SQL statements from the table and then add them again.

Delete a blacklist rule

1.
2.
3.
4. In the left-side navigation pane, choose **Settings and Management > Security Management**.
5. On the **SQL Firewall** tab, find the rule and click **Delete** in the **Actions** column.

> **Note** You can select multiple rules in the rule table and then click **Delete** to batch delete the rules.

6. In the **Delete** message, click **OK**.

> **Note** When you delete a blacklist rule created in custom parameterized SQL mode or custom SQL mode, the SQL statements in the blacklist rule table in the database are retained in the `proxy_auditing.sql_list` table even if the blacklist rule is deleted. If you want to completely delete the blacklist rule, you can connect to the primary node in the cluster by using the super administrator account and delete the SQL statements from the `proxy_auditing.sql_list` table. When you delete the SQL statements this way, the SQL statements will not be blocked after five seconds. When you delete the SQL statements from the proxy_auditing.sql_list table, do not execute the `DROP` statement to delete the table.

Cancel a blacklist rule created in custom parameterized SQL mode or custom SQL mode

- Cancel a blacklist rule created in **custom parameterized SQL mode**

  You can use one of the following methods to cancel a blacklist rule created in **custom parameterized SQL mode**:

  - Cancel a blacklist rule created in **custom parameterized SQL mode** as stated in Enable or disable a blacklist rule or Delete a blacklist rule.

    > **Note** If you only disable a blacklist rule created in custom parameterized SQL mode in the console and do not delete the parameterized SQL statements from the `proxy_auditing.sql_list` table, the rule of the same account still take effect when the blacklist rule is enabled again in the console.

  - Connect to the primary node in the cluster by using the super administrator account and delete the SQL statements from the `proxy_auditing.sql_list` table. The parameterized SQL statement will not be blocked after five seconds.

    > **Note** When you delete the SQL statements from the proxy_auditing.sql_list table, do not execute the `DROP` statement to delete the table.

- Cancel a blacklist rule created in **custom SQL mode**

  You can use one of the following methods to cancel a blacklist rule created in **custom SQL mode**:

  - Cancel a blacklist rule created in **custom SQL mode** as stated in Enable or disable a blacklist rule or Delete a blacklist rule.

    > **Note** If you only disable a blacklist rule created in custom SQL mode in the console and do not delete the SQL statements from the `proxy_auditing.org_sql_list` table, the rule of the same account still take effect when the blacklist rule is enabled again in the console.

  - Connect to the primary node in the cluster by using the super administrator account and delete the SQL statements from the `proxy_auditing.org_sql_list` table. The SQL statement will not be blocked after five seconds.

    > **Note** When you delete the SQL statements from the proxy_auditing.org_sql_list table, do not execute the `DROP` statement to delete the table.

5.4.3. Configure whitelist rules

You can create, modify, delete, enable, and disable whitelist rules in the console. This topic describes the concept of whitelist rules and how to configure whitelist rules.

Whitelist rules

After you configure a whitelist rule, SQL statements that are not added to the whitelist rule are blocked or alerted. This can protect the account for your business. This account only executes business-related SQL statements, but not SQL statements that are irrelevant to your business. Many SQL statements may be used in actual business and it takes some time to enter even a single SQL statement. To improve efficiency and user experience, PolarProxy provides the following whitelist modes:

- **Training mode**: PolarProxy only collects SQL statements, but does not block SQL statements or generate alerts.
- **Detection mode**: PolarProxy records SQL statements that are not added to the whitelist rule when PolarProxy detects them. However, PolarProxy does not block such SQL statements.
- **Protection mode**: PolarProxy records and blocks SQL statements that are not added to the whitelist rule when PolarProxy detects them.

You can also configure multiple whitelist rules in the console. Each whitelist rule can be trained by using a separate account. After the **detection mode** or **protection mode** is enabled, you can also specify the accounts to which each whitelist rule is applicable.

Add a whitelist rule

1.
2.
3.
4. In the left-side navigation pane, choose **Settings and Management > Security Management**.
5. On the **SQL Firewall** tab, click **Add** in the upper-left corner.
6. In the **Create a Rule** dialog box, set the parameters based on the mode that you select.

   Parameters for a whitelist rule

Parameter		Required	Description
Basic Information	**Rule Name**	Yes	The name of the rule. The name must meet the following requirements: ○ The name can contain digits and letters. ○ The name can be up to 30 characters in length.
	Description	No	The description of the rule. **Note** The description can be up to 64 characters in length.
	Endpoint	Yes	The endpoint to which the current rule is applied.
	Rule Type	Yes	The type of the rule. Select **Whitelist Rule**.
	Current Mode	No	The mode of the rule. Valid values: ○ **Training Mode**: collects SQL statements, but does not block SQL statements or generate alerts. ○ **Detection Mode**: records SQL statements that are not added to the whitelist rule when detecting them, but does not block such SQL statements. ○ **Protection Mode**: records and blocks SQL statements that are not added to the whitelist rule when detecting them.

Parameter		Required	Description
Configurations	**Database Account Name**	No	The name of the database account to which the rule is applied. Valid values: ○ **All Accounts**: indicates that the rule applies to all database accounts in the cluster. The text box on the right must be left empty. ○ **Include**: indicates that the rule applies only to specified database accounts. You must specify at least one database account name in the text box on the right. Separate multiple accounts with commas (,). ○ **Exclude**: indicates that the rule applies only to database accounts that are not specified here. You must specify at least one database account name in the text box on the right. Separate multiple accounts with commas (,). **Note** The database account name can be in one of the following formats: ○ `Username`. Example: `user`. ○ `Username@IP address`. Example: `user@10.0.0.0`.

7. Click **OK**.
8. Perform the steps based on the mode of the rule that you select.
   - If **Current Mode** is set to **Training Mode**, perform the following steps:
     a. Connect to the specified database endpoint by using the previously defined **database account name**.
     b. Use this account to execute business-related SQL statements that are added to the whitelist rule. PolarProxy parameterizes the SQL statements and saves them to the whitelist rule of the database. Example:

        ```
        update t set k = 2 where id = 2;
        ```

        The parameterized SQL statement:

        ```
        update t set k = ? where id = ?
        ```

        `?` indicates any value. The parameterized SQL statement `update t set k = ? where id = ?` is saved to the whitelist rule.

     Note You can also add the following `hint` command before a business-related SQL statement to parameterize it and add it to the whitelist rule: `hint(/* store_to_whitelist */)`.

   - If **Current Mode** is set to **Detection Mode**, perform the following steps:

a. Connect to the specified database endpoint by using the previously defined **database account name**.

b. Use this account to execute business-related SQL statements. PolarProxy detects whether the SQL statements are not added to the whitelist rule. Example:

```
update t set k = 2 where k = 2
```

If a SQL statement is not added to the whitelist rule, PolarProxy allows and records the SQL statement. A similar result is displayed:

```
Query OK, 0 rows affected (0.03 sec)
Rows matched:1 Changed: 0 Warnings:0
```

- If **Current Mode** is set to **Protection Mode**, perform the following steps:

a. Connect to the specified database endpoint by using the previously defined **database account name**.

b. Use this account to execute business-related SQL statements. Example:

```
select id from t where id = 1;
```

If a SQL statement is not added to the whitelist rule, PolarProxy records and blocks the SQL statement. A similar result is displayed:

```
ERROR 1141 (HY000): This SQL is rejected by SQL Firewall. Access denied for user
'xzh'@'x.x.x.x' to database 'xzh': This SQL is not in whitelist wl_test.
```

> **Note**
> - Each time you update your business, you must train business-related SQL statements. Otherwise, the SQL statements cannot be executed.
> - You can also configure multiple whitelist rules in the console. Each whitelist rule can be trained by using a separate account. After the **detection mode** or **protection mode** are enabled, you can also specify the accounts to which each whitelist rule is applicable.

Enable or disable a whitelist rule

1.
2.
3.
4. In the left-side navigation pane, choose **Settings and Management > Security Management**.
5. On the **SQL Firewall** tab, find the rule and turn on or off **Enable/Disable**.

> **Note** You can select multiple rules in the rule table and then click **Enable** or **Disable** to batch enable or disable the rules.

6. In the **Enable** or **Disable** message, click **OK**.

Note When you disabled a whitelist rule, the SQL statements in the whitelist rule table in the database are retained even if the whitelist rule is disabled. If the whitelist rule is enabled again, the rule is still applicable to the same accounts. If you want to completely disable the whitelist rule, you can connect to the primary node in the cluster by using the super administrator account and delete the SQL statements from the `proxy_auditing.sql_list` table. When you delete the SQL statements this way, the SQL statements will not be allowed after five seconds. When you delete the SQL statements from the proxy_auditing.sql_list table, do not execute the `DROP` statement to delete the table.

Modify a whitelist rule

1.
2.
3.
4. In the left-side navigation pane, choose **Settings and Management > Security Management**.
5. On the **SQL Firewall** tab, find the rule and click **Modify** in the **Actions** column. In the **Modify a Rule** dialog box, modify the parameters based on your business requirements. For more information about the parameters, see Parameters for a whitelist rule.

Note When you modify a rule, you cannot modify the **rule name**.

6. Click **OK**.

Note The parameterized SQL statements in a rule in the database cannot be modified in the console. You must delete the SQL statements from the `proxy_auditing.sql_list` table and then add them again.

Delete a whitelist rule

1.
2.
3.
4. In the left-side navigation pane, choose **Settings and Management > Security Management**.
5. On the **SQL Firewall** tab, find the rule and click **Delete** in the **Actions** column.

Note You can select multiple rules in the rule table and then click **Delete** to batch delete the rules.

6. In the **Delete** message, click **OK**.

> **Note** When you delete a whitelist rule, the SQL statements in the whitelist rule are retained in the `proxy_auditing.sql_list` table even if the whitelist rule is deleted. If you want to completely delete the whitelist rule, you can connect to the primary node in the cluster by using the super administrator account and delete the SQL statements from the `proxy_auditing.sql_list` table. When you delete the SQL statements this way, the SQL statements will not be allowed after five seconds. When you delete the SQL statements from the proxy_auditing.sql_list table, do not execute the `DROP` statement to delete the table.

5.5. FAQ about data security and encryption

This topic provides answers to some frequently asked questions about data security and encryption.

About IP whitelists

- How can I allow a server to access only a specified node in a cluster?

  You can use a custom cluster endpoint. Servers that connect to a custom cluster endpoint of a cluster are allowed to access only a specified node in the cluster.

- How many IP addresses or CIDR blocks can I add to all IP whitelists?

  You can add at most 1,000 IP addresses or CIDR blocks to all IP whitelists. Security groups do not have this limit.

- Why am I unable to connect an Elastic Compute Service (ECS) instance to a cluster after I add the IP address of the ECS instance to an IP whitelist?

  You can perform the following steps to locate the cause:

  i. Check whether the setting of the IP whitelist is correct. If you want to connect the ECS instance to the internal endpoint of the cluster, you must add the private IP address of the ECS instance to the IP whitelist. If you want to connect the ECS instance to the public endpoint of the cluster, you must add the public IP address of the ECS instance to the IP whitelist.

  ii. Check whether the ECS instance and the PolarDB for MySQL cluster are deployed in the same type of network. If the ECS instance runs in a classic network, you can migrate the ECS instance to the virtual private cloud (VPC) where the cluster is deployed. For more information, see Overview of migration solutions.

  > **Note** Do not migrate the ECS instance if you want to connect the ECS instance to other cloud services in the classic network. The ECS instance cannot access these services after it is migrated to the VPC.

  You can also use the ClassLink feature to connect the classic network to the VPC.

  iii. Check whether the ECS instance and the PolarDB for MySQL cluster run in the same VPC. If the ECS instance and the cluster run in different VPCs, you must purchase another cluster in the VPC of the ECS instance, or activate Cloud Enterprise Network (CEN) to connect the two VPCs.

- Why am I unable to connect to the public endpoint of a cluster?

  You may fail to connect to the public endpoint of a cluster due to the following reasons:

i. If you connect an ECS instance to the public endpoint of the cluster, you must add the public IP address of the ECS instance to an IP whitelist of the cluster. Do not add the private IP address of the ECS instance.

ii. Set the IP whitelist to `0.0.0.0/0` and try to access the cluster. If you can access the cluster, the public endpoint that was used before is incorrect. For more information about how to check the public endpoint, see Apply for a cluster endpoint or a primary endpoint.

- How can I allow a user account to access a cluster from only a specified IP address?

  You can create a privileged account by running the following commands. Then, you can log on with the privileged account and specify the IP address that standard accounts can use to access the cluster.

```
1
2
3 CREATE USER 'alitest'@'192.168.1.101' ;
4
5
6 select * from mysql.user where user='alitest';
```

About SSL encryption

What will happen if I do not renew an expired SSL certificate? Will my instance stop running or data security be compromised?

If you do not renew the SSL certificate after it expires, your instance can still run as normal and data security is not compromised. However, applications that connect to your instance through encrypted connections are disconnected.

About transparent data encryption (TDE)

- Can I continue to use common database tools, such as Navicat, after I enable TDE?

  Yes, you can continue to use common database tools after you enable TDE.

- Why is data still displayed in plaintext after it is encrypted?

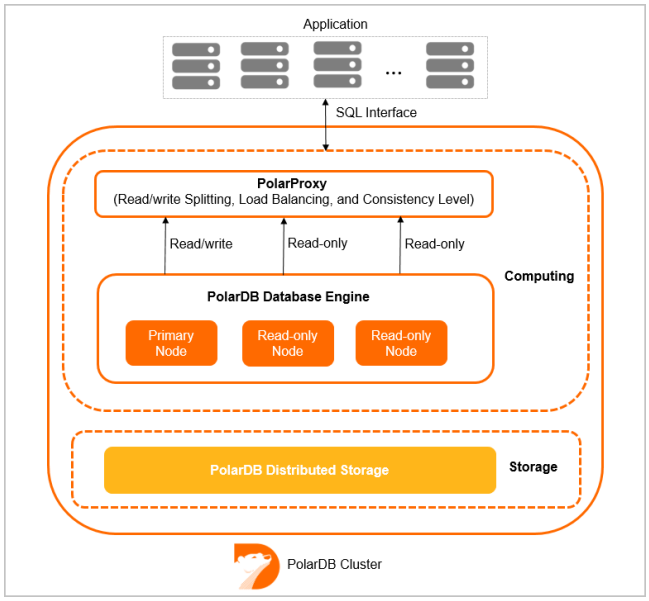
  When data is queried, the data is decrypted and loaded to the memory. Therefore, the data is displayed in plaintext. After TDE is enabled, the stored data is encrypted.

6.PolarProxy Enterprise Edition

6.1. Overview

PolarProxy serves as a proxy between a database and an application in a cluster. PolarProxy receives and routes all the requests from the application. PolarProxy supports advanced features, such as automatic read/write splitting, load balancing, consistency levels, and connection pools. PolarProxy is easy to use and maintain, and provides high-availability and high-performance services. You can connect to cluster endpoints to use the features of PolarProxy.

Architecture



A cluster of consists of a primary node and up to 15 read-only nodes. By default, PolarDB provides two types of endpoints: the primary endpoint and cluster endpoints. PolarProxy provides cluster endpoints for . Cluster endpoints include read/write endpoints and read-only endpoints. Read/write endpoints support read/write splitting. Read-only endpoints allow PolarDB clusters to distribute read requests to read-only nodes based on the number of connections.

PolarProxy Enterprise Editions

The following PolarProxy Enterprise Editions are available: and .

- is used by clusters of the type of PolarDB for MySQL Cluster Edition, which shares CPU resources with smart elastic scaling within seconds provided based on business loads.
- is used by clusters of the type of PolarDB for MySQL Cluster Edition, which exclusively uses allocated CPU resources and provides better stability.

The following table describes the differences between the two editions.

Item		

Item		
Billing method	Currently free of charge. In the future, when to charge the fees is to be determined.	
Resource type	Shared CPU physical resources with smart elastic scalability within seconds are provided based on business loads.	Exclusive use of physical resources with better performance stability provided.
Architecture	High-availability redundant architecture.	
Cluster specifications	Minimum configuration: 2 cores.	
Performance	Compared with previous editions, the maximum IOPS of cluster storage is increased by 50%. For more information about the maximum IOPS of different cluster specifications, see Specifications of compute nodes.	
Read-only node specifications	The configurations are not necessarily consistent between the read-only nodes and the primary nodes. You can downgrade configurations based on business loads to save costs.	
Number of read-only nodes	Up to 15 read-only nodes.	
Endpoint	One primary endpoint and seven cluster endpoints.	
Failover between the primary and read-only nodes	Connection and transaction not interrupted and brief block of 5 to 10 seconds. This feature is expected to be available at the end of April 2022.	
Consistency	• Eventual consistency. • Session consistency. • Global consistency.	
Connection pool	Supported.	
Transaction splitting	Supported.	
Interruption protection (connection retention)	Supported.	
Data masking (for security)	Supported.	
Business stability during configuration change	Supported.	
Multi-master architecture (available soon)	Supported. For more information about the multi-master architecture, see Release announcements for the multi-master architecture.	

Item		

Compute node scale-out within seconds (available soon)	Supported.	Exclusive use of resources to ensure sufficient resources.
Proxy throttling protection (available soon)	Supported. This feature is expected to be available at the end of May 2022.	

Billing

At present, you can use PolarProxy Enterprise Edition free of charge. In the future, when to charge the fees is to be determined.

Edition switchover policies

The following table describes the edition switchover policies for PolarProxy Enterprise Edition.

Edition	Purchase type	Policy
• Single Node Edition • Archive Database Standalone Edition	Single Node Edition and Archive Database Standalone Edition do not support the Database Proxy Enterprise Edition feature regardless of existing or newly purchased instances.	
• Cluster Edition • Archive Database Cluster Edition	Newly purchased cluster	Newly purchased clusters only support PolarProxy Enterprise Edition.
	Existing pay-as-you-go cluster	Existing pay-as-you-go clusters are automatically switched to PolarProxy Enterprise Edition.
	Existing subscription cluster	

Limits

Only clusters of Edition and support cluster endpoints and PolarProxy. clusters of the and editions use a single-node architecture, and do not support PolarProxy and cluster endpoints.

Precautions

- PolarProxy does not support compression protocols for the default and custom cluster endpoints.
- If you do not enable the transaction splitting feature after cluster endpoints are used, all the requests in a transaction are forwarded to the primary node.

- When you execute the `SHOW PROCESSLIST` statement after cluster endpoints are used, the system returns the results of all the nodes.
- If you execute multi-statement or call stored procedures, all subsequent requests for the current connection are forwarded to the primary node. For more information, see Multi-Statement. To use the read/write splitting feature, you must close the current connection and reconnect to the cluster.
- The maximum number of connections to a cluster endpoint depends on the specifications of compute nodes in backend databases. For a cluster endpoint that supports read/write splitting, an application establishes a connection to each compute node in the backend database. Therefore, the maximum number of connections that an application can use is the maximum number of connections of a single compute node. For a cluster endpoint that is in read-only mode, an application connection only establishes a connection to one compute node in the backend database. The maximum number of connections that an application can use is the sum of the maximum numbers of connections for all read-only nodes. For a cluster endpoint that is in read-only mode, you can use the Transaction-level connection pools feature to increase the maximum number of connections that an application can use.
- If a session that supports read/write splitting is created after you add or restart a read-only node, read requests are forwarded to the read-only node. If a session that supports read/write splitting is created before you add or restart a read-only node, read requests are not forwarded to the read-only node. To forward these read requests to the read-only node, you must close the connection and reconnect to the cluster. For example, you can restart your application to establish a new connection.
- Do not modify environment variables when you call stored procedures or execute multi-statement, such as `set names utf8mb4;select * from t1;`. Otherwise, the data read from the primary node and read-only nodes is inconsistent.

Read/write splitting

clusters of Edition support read/write splitting. This feature allows a PolarDB cluster to distribute read and write requests from applications by using a cluster endpoint. Write requests are forwarded to the primary node. Read requests are forwarded to the primary node or read-only nodes based on the loads on each node. The number of pending requests on a node indicates the loads on the node. For more information, see Read/write splitting.

Load balancing

supports automatic scheduling based on the load of each node. Read requests are automatically forwarded to read-only nodes based on the number of active connections. This ensures load balancing among read-only nodes. Load balancing allows you to offload reads from the primary node to read-only nodes and split transactions.

- **Primary Node Accepts Read Requests**

  SQL query statements are sent to read-only nodes when data consistency and transaction correctness are achieved. This reduces the loads on the primary node and makes the primary node stable. For more information, see Offload reads from the primary node.

- **Transaction Splitting**

  supports transaction splitting. This feature ensures data consistency in a session and allows PolarDB to send read requests to read-only nodes. This reduces the loads on the primary node. For more information, see Split transactions.

Consistency levels

asynchronously replicates the updates from the primary node to read-only nodes. In read/write splitting mode, a read request that follows a write request may fail to obtain the latest data. This causes data inconsistency. provides the eventual consistency, session consistency, and global consistency options. For more information, see Consistency levels.

Connection pools

supports session-level connection pools and transaction-level connection pools. You can select a connection pool based on your business requirements to reduce the database loads that are caused by a large number of connections. For more information, see Connection pools.

Persistent connections

supports the persistent connection feature to prevent transient disconnections or temporary failures in new connections. These issues can be caused by O&M operations, such as configuration upgrades, failover, and minor version upgrades. Issues can also be caused by other reasons. For example, the server on which nodes are deployed is unavailable. The persistent connection feature improves the high availability of . For more information, see Persistent connections.

Dynamic data masking

When your application initiates a data query request, PolarDB masks the sensitive data that is queried before PolarDB returns the data to the application. To achieve this, you need to specify the database account, the database name, and the table or column that requires data masking before the data is queried. For more information, see Dynamic data masking.

Related API operations

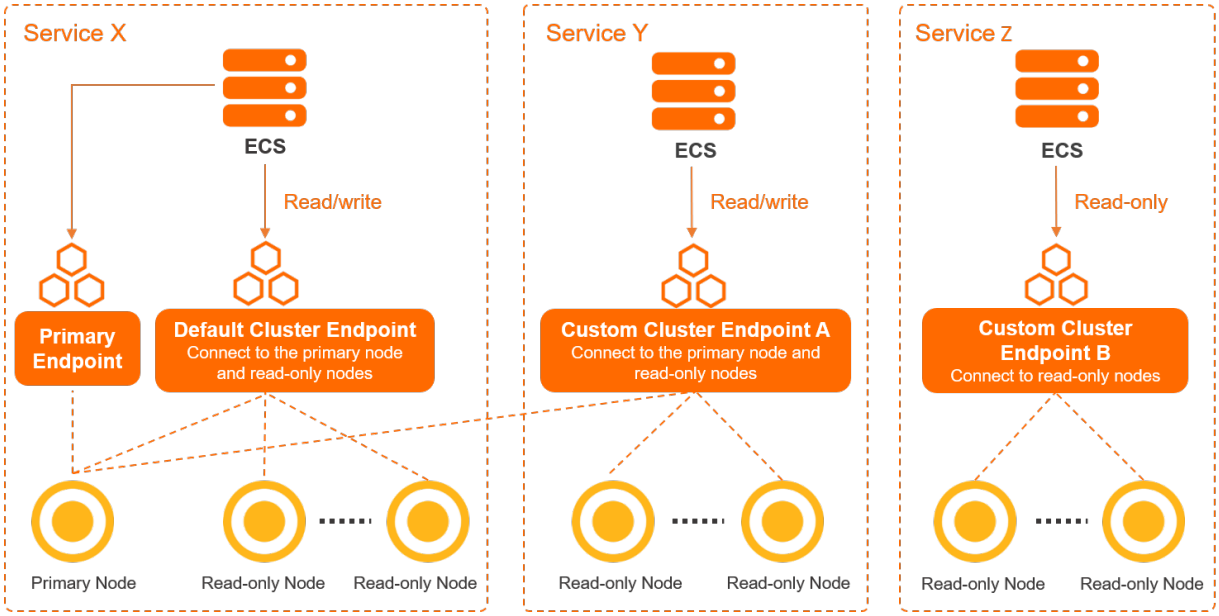
Operation	Description
CreateDBEndpointAddress	Creates a public endpoint for a specified cluster.
CreateDBClusterEndpoint	Creates a custom cluster endpoint for a specified cluster.
DescribeDBClusterEndpoints	Queries the information about the endpoints of a specified cluster.
ModifyDBClusterEndpoint	Modifies the configuration of a cluster endpoint for a specified cluster.
ModifyDBEndpointAddress	Modifies the endpoints such as custom cluster endpoints of a specified cluster.
DeleteDBEndpointAddress	Deletes a cluster endpoint of a specified cluster. This operation cannot be used to delete private custom cluster endpoints.
DeleteDBClusterEndpoint	Deletes a custom cluster endpoint of a specified cluster.

6.2. Connect to PolarDB

6.2.1. Cluster endpoints and primary endpoints

The endpoints of a cluster are classified into two types: cluster endpoint and primary endpoint. To connect to a cluster, you can use either the primary endpoint or a cluster endpoint of the cluster. This topic describes the use scenarios of and differences between cluster endpoints and primary endpoints.

Comparison between cluster endpoints and primary endpoints



Endpoint type	Description	Scenario	Supported network type

Endpoint type	Description	Scenario	Supported network type
Cluster endpoint (recommended)	The cluster endpoint is implemented by using PolarProxy. Cluster endpoints have the following features: • PolarProxy provides the read/write splitting feature. This feature enables PolarDB clusters to distribute read and write requests from applications by using cluster endpoints. The built-in proxy of a PolarDB cluster forwards write requests to the primary node, and forwards read requests to the primary node or read-only nodes based on the loads on nodes. • A cluster provides one cluster endpoint by default and allows you to create up to six cluster endpoints based on your business requirements. When you create a cluster endpoint, you can configure the read/write mode for the cluster endpoint and specify the nodes to which the cluster endpoint can connect. • To use the features of PolarProxy, you must use a cluster endpoint to connect to the cluster. For more information, see PolarProxy. **Note** allows you to create single-node cluster endpoints. If the node that is associated with a single-node cluster endpoint is faulty, the single-node cluster endpoint may remain unavailable for up to 1 hour. We recommend that you do not use single-node cluster endpoints in your production environment.	• Scenarios that require data isolation. You can use different cluster endpoints to connect your services to a cluster based on your business requirements. • Cluster endpoints can be configured in one of two read/write modes: **read and write (automatic read/write splitting)** or **read-only**. As a result, cluster endpoints can also be used for read-only services. For example, assume that you have purchased a cluster that contains one primary node and four read-only nodes. You want to connect Service A and Service B to this cluster. Service A is a read-only service and Service B is a read/write service. You can create cluster endpoint A that runs in read-only mode for Service A and associate cluster endpoint A with read-only node 1 and read-only node 2. Then, create cluster endpoint B that runs in read and write (automatic read/write splitting) mode for Service B and associate cluster endpoint B with read-only node 3 and read-only node 4. This way, the data of Service A is physically isolated from that of Service B.	• Internal network • Internet

Endpoint type	Description	Scenario	Supported network type
Primary endpoint	Each cluster supports only a single primary endpoint. The primary endpoint has the following features: • The primary endpoint allows you to connect to the primary node of the cluster. The primary endpoint can be used for read and write operations. • When the primary node is faulty, the primary endpoint is switched to a new primary node.	Scenarios that do not require read/write splitting.	

Private endpoints and public endpoints

Network type	Description	Scenario
Internal network	• A cluster achieves optimal performance when the cluster is connected by using a private endpoint. • When you create a cluster, a default private endpoint is created. This endpoint can be modified but not deleted. For more information, see Modify an endpoint.	Examples: • If your Elastic Compute Service (ECS) instance is deployed in the same virtual private cloud (VPC) as the cluster, your ECS instance can connect to the cluster by using a private endpoint. • You can use Data Management (DMS) to connect to the cluster over an internal network.
Internet	• You can apply for or delete a public endpoint. For more information, see Apply for a cluster endpoint or a primary endpoint. • A cluster cannot achieve optimal performance when it is connected by using a public endpoint.	For example, you can connect to your cluster by using a public endpoint to perform O&M operations.

Read/Write modes for cluster endpoints

You can set the read/write mode to **Read and Write (Automatic Read-write Splitting)** or **Read Only** for a cluster endpoint. The following table describes the differences between cluster endpoints that use different read/write modes.

> **Note** For information about how to configure the read/write mode for a cluster endpoint, see Configure PolarProxy.

Item	Read and Write (Automatic Read-write Splitting)	Read Only
Associated nodes	Nodes can be associated with the cluster endpoint in one of the following three configurations: • Only the primary node • One or more read-only nodes • The primary node and one or more read-only nodes **Note** In read and write mode: • All write requests are sent only to the primary node, regardless of whether the primary node has been added to the list of service nodes. • For read requests, you can configure **Primary Node Accepts Read Requests** to specify whether the primary node processes read requests.	Nodes can be associated with the cluster endpoint by one of the following two configurations: • One or more read-only nodes • The primary node and one or more read-only nodes **Note** In read-only mode: • Requests are forwarded to read-only nodes in load balancing mode. • Read requests are not forwarded to the primary node. Even if the primary node is added to the list of service nodes, read requests are still not forwarded to it. • You cannot create a cluster endpoint with only one primary node.
Primary Node Accepts Read Requests	Supported. For more information, see Read/Write splitting.	This feature is used to reduce the loads on the primary node. However, the primary node that is associated with a **Read Only** cluster endpoint does not process read or write requests. Therefore, this feature is redundant.
Transaction Splitting	Supported. For more information, see Split transactions. **Note** This configuration is supported only if **Consistency Level** is **Session Consistency (Medium)** or **Global Consistency (Strong)**.	This feature is used to reduce the loads on the primary node. However, the primary node that is associated with a **Read Only** cluster endpoint does not process read or write requests. Therefore, this feature is redundant.

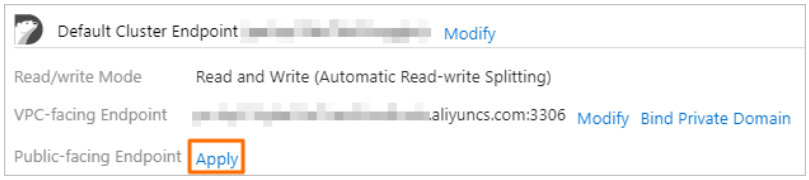
Item	Read and Write (Automatic Read-write Splitting)	Read Only
Consistency Level	**Eventual Consistency (Weak)**, **Session Consistency (Medium)**, and **Global Consistency (Strong)** are supported. For more information, see Consistency levels.	This parameter is set to **Eventual Consistency (Weak)**. The cluster endpoint in this mode does not process write requests.
Connection Pool	**Session-level** and **Transaction-level** connection pools are supported. For more information, see Connection pools. **Note** • This connection pool feature is provided by PolarProxy of . This feature does not affect the connection pool feature in your clients. If the client provides a connection pool, you can set Connection Pool to **Off** in the PolarDB console to disable the connection pool feature of PolarProxy. • After you set Connection Pool to **Off**, PolarProxy sends a request from the client to all nodes that are associated with the cluster endpoint. These nodes include the primary node and the read-only nodes. The total number of available connections is limited by the maximum number of connections that can be established to the primary node.	Not supported. **Note** evenly distributes requests among all read-only nodes that are associated with the **Read Only** cluster endpoint. The primary node does not process requests. PolarProxy sends a request from a client to only a single node. The total number of available connections is equal to the total number of connections to all read-only nodes.
Parallel Query	Not supported. The use of parallel query adversely impacts the primary node.	Supported. For more information, see Parallel query.

6.2.2. Apply for a cluster endpoint or a primary endpoint

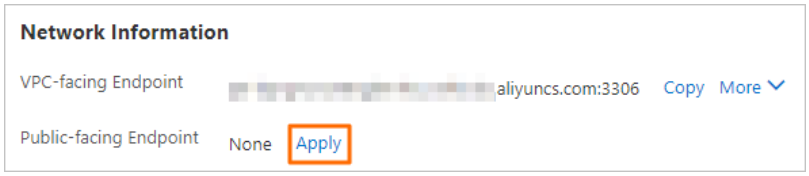
You can use a cluster endpoint or a primary endpoint of a cluster to connect to the cluster. This topic describes how to apply for and manage a cluster endpoint or a primary endpoint.

Apply for a cluster endpoint or a primary endpoint

1.
2.
3.
4. In the **Endpoints** section of the **Overview** page, you can apply for an endpoint by using one of the following methods:
   - Method 1:
     a. In the upper-right corner of the **Endpoints** section, click the icon to switch the view.
     b. Click **Apply**.



   - Method 2:
     a. On the right side of the cluster endpoint, click **Modify**.
     b. In the **Network Information** section of the dialog box that appears, click **Apply**.



> **Note** You can apply only for **Public-facing Endpoint** endpoints. After you create a cluster, a default **VPC-facing Endpoint** endpoint is provided. You do not need to apply for this endpoint.

5. In the dialog box that appears, specify a prefix for the endpoint and click **OK**.

> **Note** The prefix of the endpoint must meet the following requirements:
> - The prefix must be 6 to 30 characters in length, and can contain lowercase letters, digits, and hyphens (-).
> - The prefix must start with a lowercase letter and end with a digit or a lowercase letter.

After an application for the cluster endpoint is approved, all features that can be provided by the proxy are supported. You can modify the parameters of the features provided by the proxy based on your business requirements. For more information, see Configure PolarProxy.

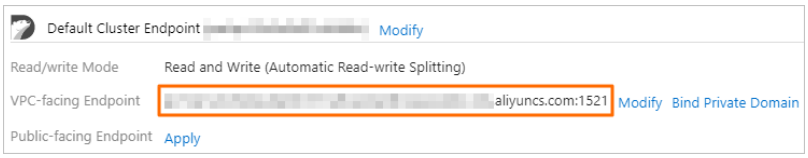
Create a custom cluster endpoint

In the **Endpoints** section of the **Overview** page, click **Create Custom Cluster Endpoint**. In the dialog box that appears, create a custom cluster endpoint and configure the features provided by the proxy for the custom cluster endpoint. For more information, see Configure PolarProxy.

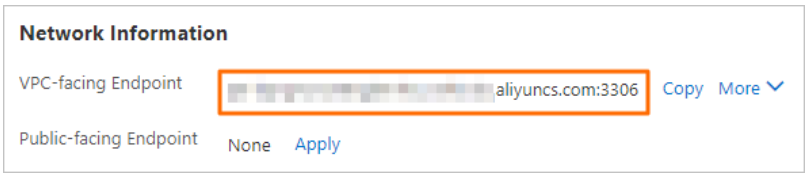
View an endpoint

You can view the domain name and port number of an endpoint by using one of the following methods:

- In the upper-right corner of the **Endpoints** section, click the icon to switch the view. Then, you can view the domain name and port number of the endpoint.



- On the right side of the cluster endpoint, click **Modify**. In the **Network Information** section of the dialog box that appears, you can view the domain name and port number of the endpoint.

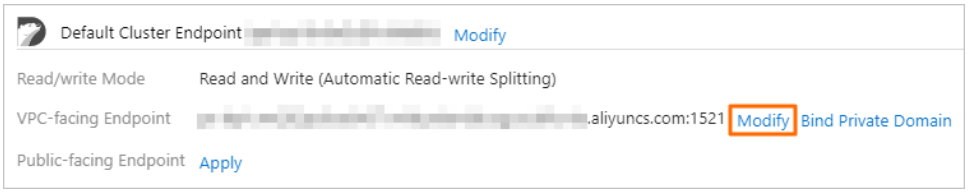


> **Note**
>
> - If you use a domain name to connect to a database, you can click **Bind Private Domain** to bind the domain name to a private endpoint. This allows you to retain the original database domain name after the database is migrated to the cloud. Only **VPC-facing Endpoint** endpoints can be bound to private domain names. For more information, see Private domain names.
> - The default port number of an endpoint that is used by an cluster is 3306. You can change the port number. For more information, see Modify an endpoint.

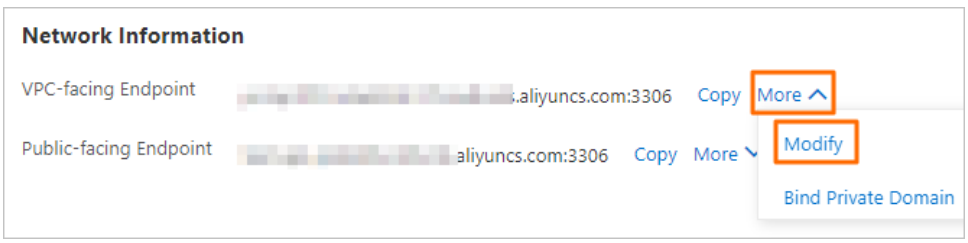
Modify an endpoint

You can change the domain name and port number for an endpoint by using one of the following methods:

- In the upper-right corner of the **Endpoints** section, click the icon to switch the view. Then, find the endpoint that you want to manage and click **Modify**.



- On the right side of the cluster endpoint, click **Modify**. In the **Network Information** section of the dialog box that appears, choose **More > Modify**.



Notice

- The prefix of the endpoint must meet the following requirements:
  - The prefix must be 6 to 30 characters in length, and can contain lowercase letters, digits, and hyphens (-).
  - The prefix must start with a lowercase letter and end with a digit or a lowercase letter.
- The port number ranges from 3000 to 3500.
- If SSL is enabled for the endpoint, the cluster restarts after you modify the endpoint.
- If SSL is enabled for the endpoint, the total length of the new endpoint cannot exceed 64 characters.

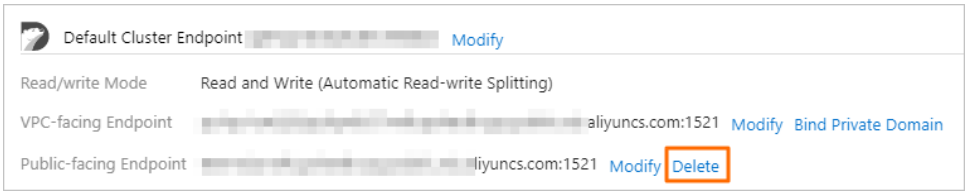
Delete an endpoint

Warning

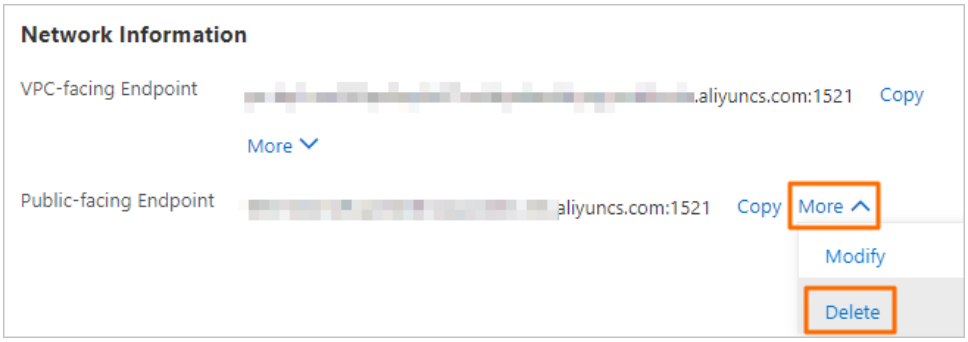
- Before you delete an endpoint, make sure that your application is connected to the cluster by using another endpoint.
- A deleted endpoint cannot be recovered. If you need an endpoint, click **Apply** to apply for a new endpoint. For more information, see Apply for a cluster endpoint or a primary endpoint.
- The default cluster endpoint can be modified but cannot be deleted. Custom cluster endpoints can be deleted.

You can delete an endpoint by using one of the following methods:

- In the upper-right corner of the **Endpoints** section, click the icon to switch the view. Then, find the endpoint that you want to manage and click **Delete**.



- On the right side of the cluster endpoint, click **Modify**. In the **Network Information** section of the dialog box that appears, choose **More > Delete**.



> **Note** You can delete only **Public-facing Endpoint** endpoints.

What to do next

Connect to a cluster

Related API operations

API	Description
DescribeDBClusterEndpoints	Queries the endpoints of a specified cluster.
CreateDBEndpointAddress	Creates a public endpoint for a specified cluster.
ModifyDBEndpointAddress	Modifies the default endpoint of a specified cluster.
DeleteDBEndpointAddress	Deletes a cluster endpoint of a specified cluster.

6.2.3. Connect to a cluster

This topic describes how to use Data Management (DMS) or a MySQL client to connect to a cluster of .

Prerequisites

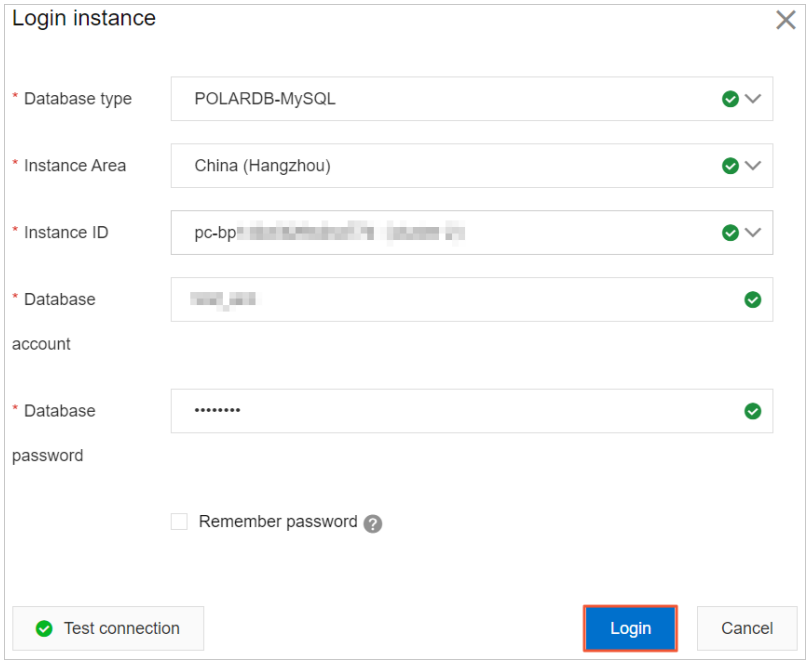
A privileged account or a standard account is created for the cluster. For more information, see Create a database account.

Use DMS to connect to a cluster

DMS is a visualized data management service provided by Alibaba Cloud. DMS provides various management services such as data management, schema management, access control, business intelligence (BI) charts, data trends, data tracking, performance optimization, and server management. You can use DMS to manage relational databases such as MySQL, SQL Server, and PostgreSQL databases and NoSQL databases such as MongoDB and Redis databases. You can also use DMS to manage Linux servers.

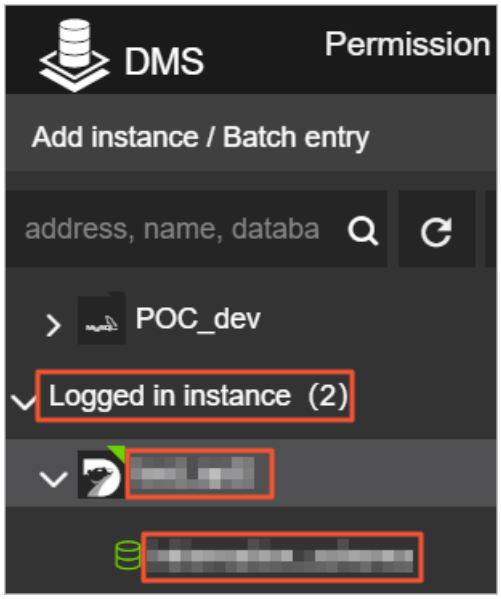
1.
2.
3.
4. In the upper-right corner of the **Overview** page, click **Log On to Database**.

5. In the dialog box that appears, enter the database account and database password that you created for the cluster and click **Login**.



> **Note** The logon account must be granted management permissions on the database. Otherwise, you cannot find the database in the left-side navigation pane of the DMS console. For information about how to modify the permissions granted to an account, see Modify the permissions of a standard account.

6. After you log on to DMS, refresh the page. In the left-side navigation pane, click **Instances Connected**.
7. In the **Instances Connected** list, click the cluster name, find the database, and then double-click the database name. Then, you can manage the database.

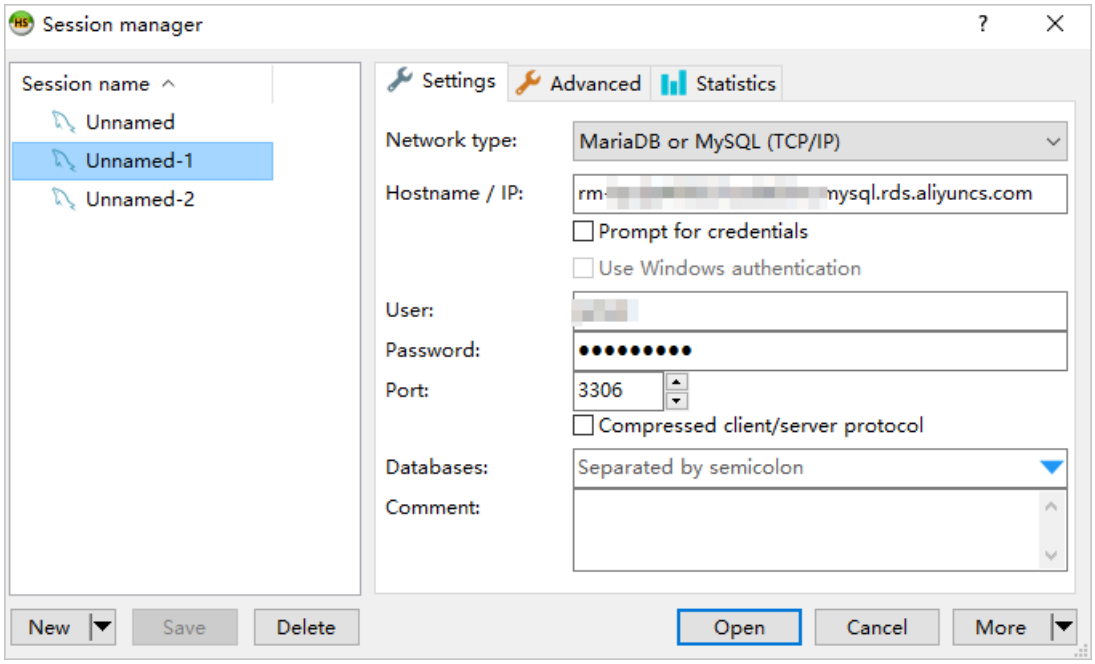


Use a client to connect to a cluster

You can use a MySQL client to connect to a cluster. The client used in the following example is HeidiSQL.

1. Start HeidiSQL.

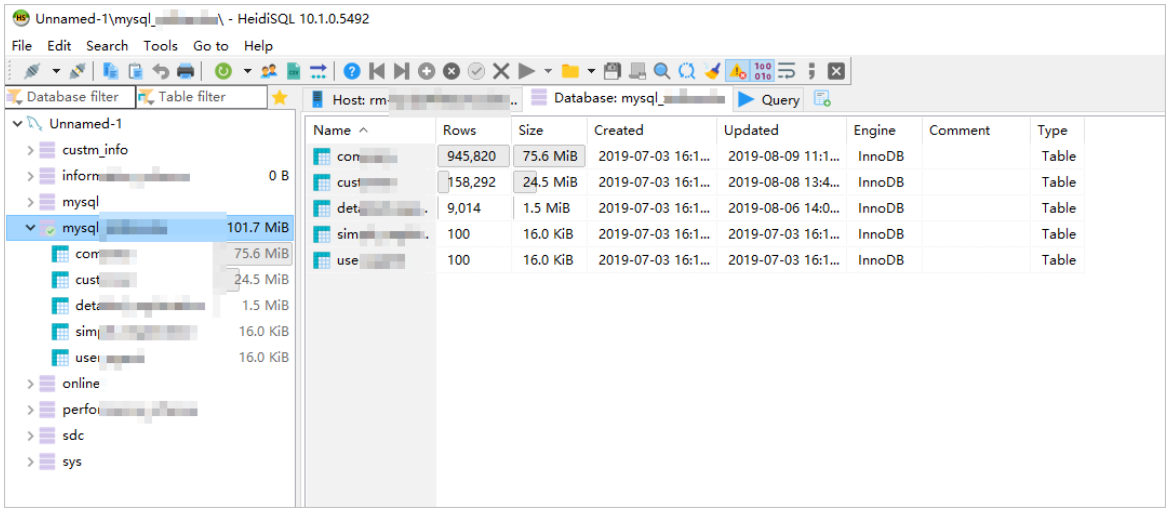
2. In the lower-left corner of the session manager, click **New**.



3. Specify the information about the cluster to which you want to connect. The following table describes the parameters.

Parameter	Description
Network type	The network protocol type that is used to connect to the database. Select MariaDB or MySQL (TCP/IP).
Hostname / IP	Enter a public or private endpoint of the cluster. ○ If the client runs on an Elastic Compute Service (ECS) instance that is deployed in the same region and has the same network type as the cluster, use a private endpoint. For example, if the ECS instance and the cluster are deployed in a virtual private cloud (VPC) in the China (Hangzhou) region, you can use a private endpoint to establish a secure and fast connection. ○ In other scenarios, use a public endpoint. To view the endpoint and port information about the cluster, perform the following steps: i. Log on to the PolarDB console. ii. In the upper-left corner of the page, select the region in which the cluster is deployed. iii. Find the cluster and click the cluster ID. iv. On the **Overview** page, view the endpoint and port information.
Users	The name of the account that is used to connect to the cluster.
Password	The password of the account.
Port	The port number in the public or private endpoint that is used to connect to the cluster. The default port number is 3306.

4. Click **Open**. If the connection information is valid, the client is connected to the cluster.



Use the CLI to connect to a cluster

If MySQL is installed on your server, you can run the following command in the CLI to connect to a cluster:

```
mysql -h<Endpoint> -P<Port number> -u<Username>
```

```
Example:
mysql -hpc-bp199s527********.rwlb.rds.aliyuncs.com -p3306 -upolardb
```

> **Note** After you run the preceding command, `Enter password` appears. Enter the database password of the cluster as prompted to connect to the cluster.

```
[root@iZbp1g                    i28gZ ~]# mysql -hpc-bp                    ]d.rwlb.rds.aliyuncs.com -p3306 -uroot12 -pPas      233
Warning: Using a password on the command line interface can be insecure.
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 536887153
Server version: 5.7.28-log Source distribution

Copyright (c) 2000, 2016, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

Parameter	Description	Example
-h	The public or private endpoint of the cluster. For more information about endpoints, see Cluster endpoints and primary endpoints.	`pc-bpxxxxxxxxxxxxxx.mysql.polardb.rds.aliyuncs.com`

Parameter	Description	Example
-P	The port number that is used to connect to the cluster. • If you use the private endpoint to connect to the cluster, enter the private port number in the private endpoint. • If you use a public endpoint to connect to the cluster, enter the port number in the public endpoint. **Note** • The default port number is 3306. • If you want to use the default port, you do not need to specify a value for this parameter.	`3306`
-u	The name of the database account that is used to connect to the cluster.	`root`

6.2.4. Troubleshoot cluster connection failures

This topic describes common causes for the failures in connecting Data Management (DMS) and common MySQL clients to clusters and provides solutions for these failures.

A whitelist that is not configured or is incorrectly configured for a cluster

Causes:

- The default whitelist contains only the IP address `127.0.0.1`, which indicates that no IP addresses are allowed to access the cluster. The IP addresses of the clients that require access to a cluster are not added to the whitelist of the cluster.
- The formats of the IP addresses that are specified in the whitelist are invalid.
- The public IP addresses that are added to the whitelist of a cluster are not the outbound IP addresses of the clients that require access to the cluster.

Solutions:

- Add the IP addresses of the clients that require access to the cluster to the whitelist of the cluster. For more information, see Configure an IP whitelist.
- Specify IP addresses for the whitelist in a valid format. For example, change `0.0.0.0` to `0.0.0.0/0`.
- Obtain the correct public IP addresses of the clients that require access to the cluster and add the correct public IP addresses to the whitelist of the cluster.

No database accounts are created or the current account does not have permission to access the database

Causes:

- No database accounts are created.
- The current account does not have permission to access the database.

Solution:

- Create a database account in the cluster and grant the account permission to access the database. For more information, see Create a database account.
- Modify the permission of the current database account in the cluster. For more information, see Reset permissions of the privileged account and Modify the permissions of a standard account.

A private or public endpoint that is incorrectly used

Cause: A private or public endpoint is incorrectly used.

Solution: Make sure that you use the correct endpoint to connect to your cluster. If you want to access the cluster over a virtual private cloud (VPC), use a private endpoint of the cluster. If you want to access the cluster over the Internet, use a public endpoint of the cluster.

Network type mismatch

Cause: The network type of the Elastic Compute Service (ECS) instance to which your cluster is connected is different from that of your cluster. The ECS instance is deployed in the classic network, and the cluster is deployed in a VPC.

Solutions:

- We recommend that you migrate your ECS instance from the classic network to a VPC. For more information, see Migrate an ECS instance from a classic network to a VPC.

  > **Note** Your ECS instance and cluster must be deployed in the same VPC. Otherwise, they cannot communicate with each other over the VPC.

- Use the ClassicLink feature to establish an internal network connection between the ECS instance in the classic network and the PolarDB cluster in the VPC.
- Use the public endpoint of the cluster to connect the ECS instance to the cluster over the Internet. This solution does not provide optimal performance or high security and stability.

6.2.5. Private domain names

Assume that you use domain names to connect to databases and you want to retain the original domain names of the databases after the databases are migrated to the cloud. In this case, you can bind the private domain names by using the private domain name feature.

Scenarios

You can bind a private domain name to each VPC-facing endpoint of . Private domain names take effect in only the VPC that you specify in the current region. Private domain names have a higher priority for resolution than the domain names that take effect in the globe.

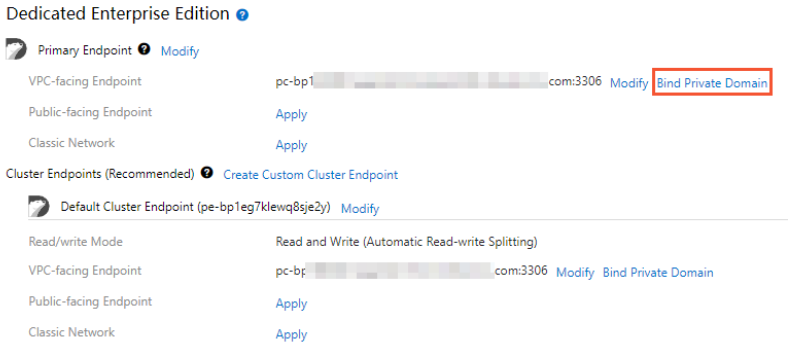
For example, the original domain name of a database is developer.aliyundoc.com, and the database is migrated to the cluster. The endpoint of the cluster is image.developer.aliyundoc.com. To allow the original domain name to remain unchanged, you can create a private domain name to bind developer.aliyundoc.com that is a CNAME record to image.developer.aliyundoc.com. After the domain name is bound to the endpoint, you can access the cluster by visiting developer.aliyundoc.com in the specified VPC, as shown in the following figure.

Billing description

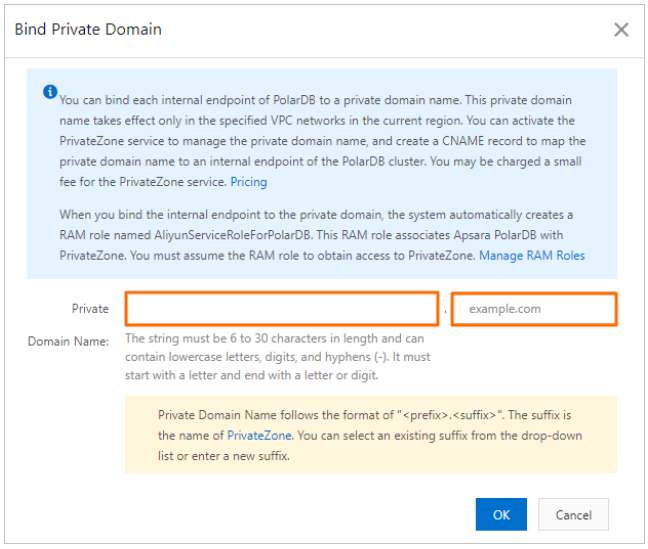
The private domain name feature of is realized by mapping the private domain names that are managed by PrivateZone to the VPC-facing endpoints of . PrivateZone charges a small amount of fee. For more information about pricing, see Pricing.

Bind a private domain name

1.
2.
3.
4. In the upper-right corner of the **Endpoints** section on the **Overview** page, click the icon to switch the view.
5. On the right side of the VPC-facing endpoint, click **Bind Private Domain**.



6. In the **Bind Private Domain** dialog box, enter the prefix and the suffix of the private domain name.



The format of private domain names is `<prefix>.<suffix>`. The following table describes the format of the private domain names.

Configuration	Description
Prefix of a private domain name	The prefix of the private domain name must be 6 to 30 characters in length and can contain at least one of the following types of characters: lowercase letters, digits, and hyphens (-). The prefix must start with a letter and end with a digit or a letter.
Suffix of the private domain name (zone)	You can select an existing zone from the drop-down list or enter a new zone. For more information about zones, see PrivateZone. **Note** ◦ If the VPC where your cluster resides is not in the configured zone, the system automatically binds the VPC to the zone. ◦ You canview and manage zones in the PrivateZone console.

Note When you bind a private domain name, the system automatically creates an AliyunServiceRoleForPolarDB role. For more information, see RAM role linked to Apsara PolarDB.

7. Click **OK**.
8. In the **Bind Private Domain** dialog box, confirm the information about the domain name again and click **OK**.

Related API operations

API	Description
ModifyDBEndpointAddress	Modifies the endpoints of a cluster, including the primary endpoint, default cluster endpoint, custom cluster endpoint, and private domain name.

6.2.6. FAQ

This topic provides answers to frequently asked questions about how to connect to clusters.

- Am I charged for data traffic if my application uses a public endpoint to connect to a cluster?

  No, you are not charged for data traffic that is incurred by using public endpoints of clusters.

- What is the maximum number of single-node cluster endpoints that I can create for a cluster?

  You can create up to six custom cluster endpoints for a cluster. The custom cluster endpoints can be single-node cluster endpoints. For more information about how to create a single-node cluster endpoint, see Create a custom cluster endpoint.

  > **Warning** If you create a single-node cluster endpoint for a read-only node and the read-only node becomes faulty, the single-node cluster endpoint may be unavailable for up to 1 hour. We recommend that you do not create single-node cluster endpoints in your production environment.

- If a single-node cluster endpoint is created for a read-only node, can the read-only node be used as the new primary node after a failover?

  The read-only node for which a single-node cluster endpoint is created cannot be automatically used as the new primary node after a failover. However, you can manually promote the read-only node as the new primary node. For more information, see Automatic failover and manual failover.

- What is the maximum number of cluster endpoints for a cluster?

  A cluster can have a maximum of seven cluster endpoints. One cluster endpoint is the default cluster endpoint and the other endpoints are custom cluster endpoints.

- Can I modify a cluster endpoint?

  Yes, you can modify the default cluster endpoint and custom cluster endpoints. For more information, see Modify an endpoint.

- Can I delete a cluster endpoint?

  Yes, you can delete only custom cluster endpoints. However, you cannot delete the default cluster endpoint. For more information, see Delete an endpoint.

6.3. Read/write splitting

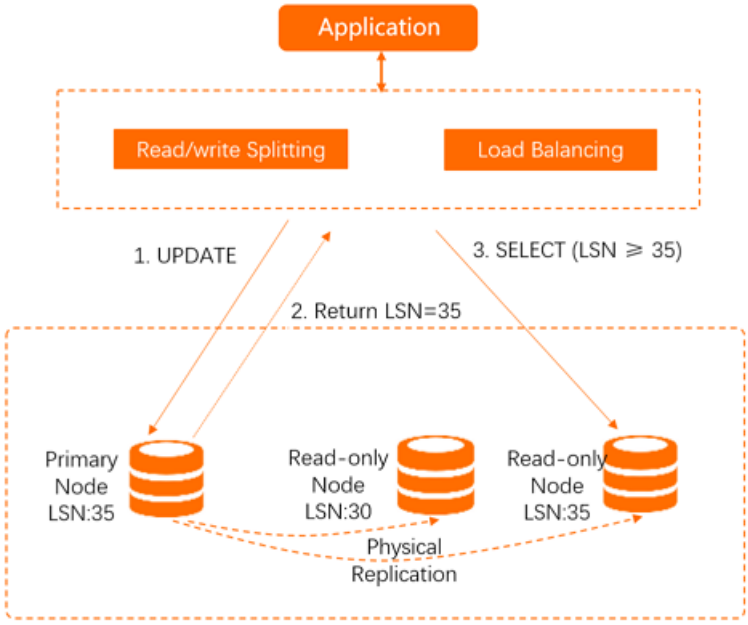
6.3.1. Overview

A cluster supports the read/write splitting feature. In read/write splitting mode, you need only to use a cluster endpoint to send requests from applications to a cluster. In this mode, write requests are automatically forwarded to the primary node. Read requests are automatically forwarded to the primary node or read-only nodes based on the load of each node. The load of a node is based on the number of pending requests.

Benefits

- **Read consistency**

When a client connects to a backend server by using a cluster endpoint, the built-in database proxy for read/write splitting automatically establishes connections between the primary node and read-only nodes. In a session, the built-in database proxy first selects an appropriate node based on the data synchronization progress of each database node. Then, the proxy forwards read and write requests to the nodes whose data is up-to-date and valid. This helps balance read and write requests among the nodes.



- **Native read/write splitting for enhanced performance**

  You can create your own proxy in the cloud to implement read/write splitting. However, high latency may occur because data is parsed and forwarded by multiple components before the data is written to a database. uses a built-in proxy that is deployed in existing secure links to implement read/write splitting and ensure that data does not pass through multiple components. This reduces the latency and speeds up data processing.

- **Easy maintenance**

  For traditional database services, read/write splitting is time-consuming. You must specify the endpoints of the primary node and each read-only node in applications. You also need to configure forwarding rules to send write requests to the primary node and read requests to read-only nodes.

  provides a cluster endpoint that can be used by an application to connect to the cluster. After the application is connected to the cluster, you can send read and write requests from the application to the cluster. Write requests are automatically forwarded to the primary node, and read requests are automatically forwarded to the primary node or read-only nodes. The process of read/write splitting is transparent to users. This reduces maintenance costs.

  You need only to add read-only nodes to scale the processing capabilities of your cluster. You do not need to modify your application.

- **Node health checks for enhanced database availability**

  The read/write splitting module of automatically performs health checks on all nodes in a cluster. If a node fails or the latency exceeds a specified threshold, stops sending read requests to this node. Write and read requests are sent to other healthy nodes. This ensures that applications can access the cluster even if a read-only node fails. After the node recovers, automatically adds the node to the list of nodes that are available to receive requests.

- **Free feature that reduces resource and maintenance costs**

  The read/write splitting feature is available free of charge.

Logic to forward requests

Forwarding logic in read/write splitting mode:

- The following requests are forwarded only to the primary node:
  - Requests for DML statements such as INSERT, UPDATE, DELETE, and SELECT FOR UPDATE
  - All data definition language (DDL) statements used to perform operations such as creating databases or tables, deleting databases or tables, and changing schemas or permissions.
  - All requests that are encapsulated in transactions
  - Requests for user-defined functions
  - Requests for stored procedures
  - Requests for EXECUTE statements
  - Requests for multi-statements
  - Requests that involve temporary tables
  - Requests for SELECT last_insert_id() statements
  - All requests to query or modify user environment variables
  - All requests for KILL statements in SQL (not KILL commands in Linux)
- The following requests are forwarded to the primary node or read-only nodes:

  > **Note** The following requests are forwarded to the primary node only after **Primary Node Accepts Read Requests** is disabled. By default, the primary node does not process read requests.

  - Read requests that are not encapsulated in transactions
  - Requests for COM_STMT_EXECUTE statements
- The following requests are forwarded to all nodes:
  - All requests to modify system environment variables
  - Requests for USE statements
  - Requests for COM_STMT_PREPARE statements
  - Requests that are sent to execute COM_CHANGE_USER, COM_QUIT, and COM_SET_OPTION statements.
  - Requests for SHOW PROCESSLIST statements

    > **Note** After a SHOW PROCESSLIST statement is executed, returns all processes that are running on any nodes in your database system.

Forwarding logic in read-only mode:

- DDL and DML operations are not supported.
- Requests are forwarded to read-only nodes in load balancing mode.
- Read requests are not forwarded to the primary node. Even if the primary node is added to the list of service nodes, read requests are still not forwarded to it.

Features

provides the following features for read/write splitting:

- **Load balancing**

  supports automatic scheduling based on the load of each node. Read requests are automatically forwarded to read-only nodes based on the number of active connections. This ensures load balancing between read-only nodes.

  Load balancing includes the **Primary Node Accepts Read Requests** and **Transaction Splitting** features:

  - **Primary Node Accepts Read Requests**

    After you enable the feature that offloads read requests from the primary node to read-only nodes, common read requests are no longer forwarded to the primary node. In a transaction, read requests that require high consistency are still forwarded to the primary node to meet business requirements. If all read-only nodes fail, read requests are forwarded to the primary node. If your workloads do not require high consistency, you can set the consistency level to eventual consistency to reduce the number of read requests that are forwarded to the primary node. You can also use the transaction splitting feature to reduce the number of read requests that are forwarded to the primary node before a transaction is started. However, broadcast requests such as SET and PREPARE requests are forwarded to the primary node.

    > **Note**
    > - The **Primary Node Accepts Read Requests** parameter is available only if the **Read/write Mode** parameter is set to **Read and Write (Automatic Read-write Splitting)**. The **Primary Node Accepts Read Requests** feature is disabled by default. For information about how to modify **Primary Node Accepts Read Requests** settings, see Configure PolarProxy.
    > - The configuration of **Primary Node Accepts Read Requests** immediately takes effect after it is modified.

  - **Transaction Splitting**

    Before transaction splitting is enabled, the database proxy sends all requests in a transaction to the primary node. This ensures the read and write consistency of transactions in a session. However, this causes heavy loads on the primary node. After transaction splitting is enabled, the database proxy identifies the transaction status. Then, read requests that are sent before the transaction is started are forwarded to read-only nodes by using the load balancing module. During this process, the read and write consistency is ensured. For more information, see Split transactions.

- **Consistency Level**

  asynchronously replicates the updates from the primary node to read-only nodes. In read/write splitting mode, a read request that follows a write request may fail to fetch the latest data. provides the eventual consistency, session consistency, and global consistency options. For more information, see Consistency levels.

- **Connection Pool**

  supports session-level connection pools and transaction-level connection pools. You can select a connection pool based on your business requirements to reduce the database loads that are caused by a large number of connections. For more information, see Connection pools.

- **Persistent connections**

  adds the persistent connection feature to prevent temporary service interruptions or connection failures. These issues may be caused by O&M operations, such as specification upgrades, switchovers, and minor version updates. The issues may also be caused by anomalies such as server malfunctions. Persistent connections can improve the availability of . For more information, see Persistent connections.

Hints

supports the following hints:

> **Notice**
>
> - Hints have the highest routing priority and are not limited by consistency levels or transaction splitting. Before you use hints, perform an evaluation.
> - You can use hints only after you set the read/write mode to **Read and Write (Automatic Read-write Splitting)** for a cluster endpoint. Hints are not supported when the read/write mode of a cluster endpoint or a primary endpoint is set to **Read Only**. For more information about the read/write mode of a cluster endpoint, see Read/Write modes for cluster endpoints.

- You can add `/*FORCE_MASTER*/` or `/*FORCE_SLAVE*/` to an SQL statement to forcibly specify the routing direction for the SQL statement.

  For example, assume that `SELECT * FROM test` is routed to a read-only node. If the SQL statement is changed to `/*FORCE_MASTER*/ SELECT * FROM test`, the statement is routed to the primary node.

- You can add `/*force_node='<Node ID>'*/` to an SQL statement to forcibly specify a node to execute the SQL statement.

  For example, `/*force_node='pi-bpxxxxxxxx'*/ show processlist` specifies that the `show processlist` statement is executed on a node named `pi-bpxxxxxxxx`. If the node is unavailable, the error message `force hint server node is not found, please check.` is returned.

- You can add `/*force_proxy_internal*/set force_node = '<Node ID>'` to an SQL statement to forcibly specify a node to execute all SQL statements.

  For example, if you execute the `/*force_proxy_internal*/set force_node = 'pi-bpxxxxxxxx'` statement, all read requests are routed to a node named `pi-bpxxxxxxxx`. If the node fails, the error message `set force node 'rr-bpxxxxx' is not found, please check.` is returned.

> **Note**
>
> - If you want to execute the preceding statement that contains the hint on the official command line of MySQL, add the -c parameter in the statement. Otherwise, the hint becomes invalid because the official command line of MySQL filters out the hint. For more information, see mysql Client Options.
> - We recommend that you do not use `/*force_proxy_internal*/` in SQL statements. Otherwise, all subsequent SQL statements are routed to the specified node and the read/write splitting feature becomes invalid.
> - Hints cannot contain statements that change environment variables. For example, if you use `/*FORCE_SLAVE*/ set names utf8;`, errors may occur.

6.3.2. Load balancing

supports automatic distribution based on the loads of nodes. Read requests are automatically forwarded to read-only nodes based on the number of active connections. This ensures load balancing across read-only nodes. This topic describes how to offload reads from the primary node and spit transactions for load balancing.

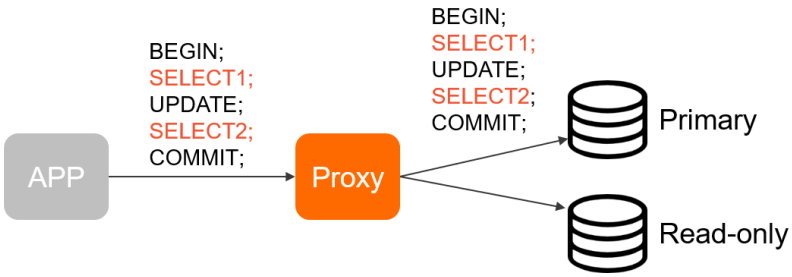
Offload reads from the primary node

After you enable the feature to offload reads from the primary node, common read requests are no longer forwarded to the primary node. Within a transaction, read requests that have strict consistency requirements are still forwarded to the primary node to meet the business requirements. If all read-only nodes are faulty, read requests are forwarded to the primary node. If your workloads do not require high consistency, you can set the consistency level to eventual consistency to reduce the read requests that are forwarded to the primary node. You can also use the transaction splitting feature to reduce the read requests that are forwarded to the primary node before a transaction is started. However, broadcast requests such as SET and PREPARE requests are forwarded to the primary node. For more information about how to modify the **Primary Node Accepts Read Requests** configuration, see Configure PolarProxy.

> **Note**
>
> - The **Primary Node Accepts Read Requests** parameter is available only if the **Read/write Mode** parameter is set to **Read and Write (Automatic Read-write Splitting)**.
> - The modification of the **Primary Node Accepts Read Requests** configuration immediately takes effect.

Split transactions

If the cluster endpoint that is used to connect to the cluster is in read/write mode, PolarProxy forwards read and write requests to the primary node and read-only nodes. To ensure data consistency among transactions within a session, PolarProxy sends all requests in transactions of the session to the primary node. For example, database client drivers such as the Java Database Connectivity (JDBC) encapsulate requests in a transaction. In this case, all requests from applications are sent to the primary node. This results in heavy loads on the primary node. However, no requests are sent to read-only nodes. The following figure shows the process.



To fix this issue, provides the transaction splitting feature. This feature ensures data consistency in a session and allows to send read requests to read-only nodes to reduce the loads on the primary node. You can reduce the read loads on the primary node without the need to modify the code or configuration of your application. This way, the stability of the primary node is improved.

> **Note** Only transactions in the sessions that are at the Read Committed isolation level can be split.

To reduce the load of the primary node, PolarProxy sends read requests that are received before the first write request in a transaction is sent to read-only nodes. Uncommitted data in transactions cannot be queried from read-only nodes. To ensure data consistency in transactions, all read and write requests that are received after the first write request are still forwarded to the primary node. For more information about how to enable transaction splitting, see Configure PolarProxy.

6.3.3. Consistency levels

provides three consistency levels to meet your different consistency requirements. The three consistency levels are eventual consistency, session consistency, and global consistency.

Issues and solutions

MySQL provides a proxy that supports read/write splitting. The proxy establishes connections from applications to MySQL and parses SQL statements. Then, the proxy forwards requests for write operations such as UPDATE, DELETE, INSERT, and CREATE operations to the primary database, and requests for SELECT operations to secondary databases. The replication delay increases if the loads on databases are heavy. For example, when you execute DDL statements to add columns to a large table or insert a large amount of data, a large replication delay occurs. In this case, you cannot retrieve the latest data from read-only nodes. The read/write splitting feature cannot solve this issue.

uses asynchronous physical replication to synchronize data among the primary and read-only nodes. After the data on the primary node is updated, the updates are synchronized to read-only nodes. The replication delay varies based on the write loads on the primary node. The replication delay is just a few milliseconds. The asynchronous replication ensures eventual consistency among the primary and read-only nodes. provides the following three consistency levels to meet your different consistency requirements:

- Eventual consistency

- Session consistency
- Global consistency

> **Note** For more information about how to change the consistency level, see Configure PolarProxy.

Eventual consistency

- Description

  runs in a read/write splitting architecture. Traditional read/write splitting ensures only eventual consistency. The retrieved results from different nodes may be different due to a primary/secondary replication delay. For example, if you repeatedly execute the following statements within a session, the result returned by each SELECT statement may be different. The actual query result depends on the replication delay.

  ```
  INSERT INTO t1(id, price) VALUES(111, 96);
  UPDATE t1 SET price = 100 WHERE id=111;
  SELECT price FROM t1;
  ```

- Scenarios

  To reduce loads on the primary node and send as many read requests as possible to read-only nodes, we recommend that you select eventual consistency.

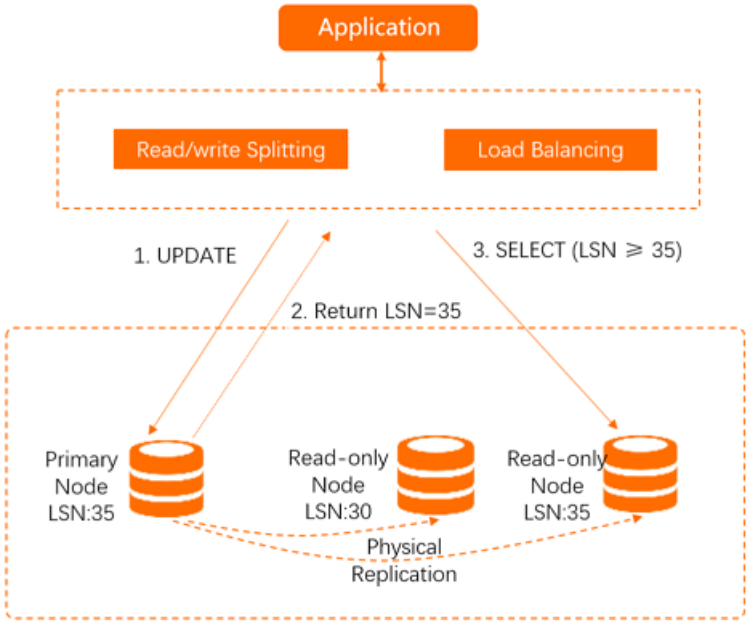
Session consistency

- Description

  To eliminate data inconsistencies caused by eventual consistency, requests are split in most cases. The requests that require high consistency are sent to the primary node. The requests that require at least eventual consistency are sent to read-only nodes by using the read/write splitting feature. However, this increases the loads on the primary node, reduces read/write splitting performance, and complicates application development.

  To solve the issue, provides session consistency. Session consistency is also known as causal consistency. Session consistency ensures that the data that is updated before read requests are sent within a session can be obtained. This ensures that data is monotonic.

  uses PolarProxy to achieve read/write splitting. PolarProxy tracks redo logs that are applied on each node and records each log sequence number (LSN). When the data in the primary node is updated, records the LSN of the new update as a session LSN. When a new read request arrives, compares the session LSN with the LSN on each node and forwards the request to a node where the LSN is greater than or equal to the session LSN. This ensures session consistency. implements efficient physical replication.



To ensure efficient synchronization, data is being replicated to other read-only nodes when the read-only node returns the result to the client. This allows data to be updated on read-only nodes before subsequent read requests arrive. In most scenarios, a large number of read requests and a small number of write requests exist. Therefore, this mechanism can ensure session consistency, read/write splitting, and load balancing based on the verification result.

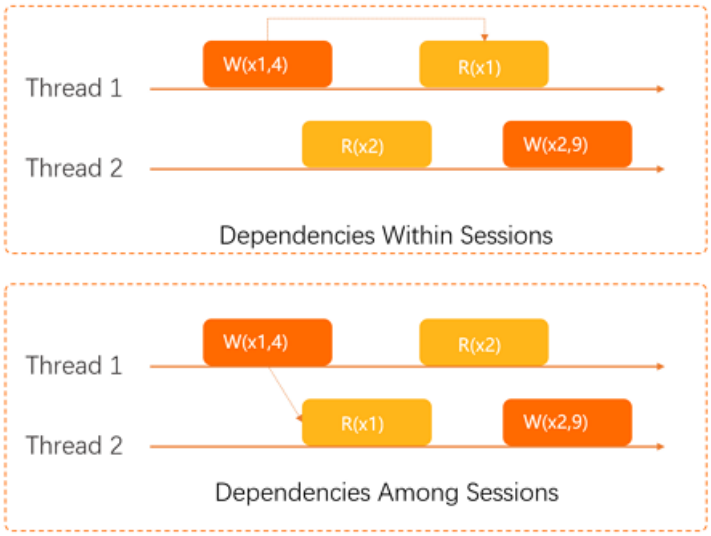
- Scenarios

  A higher consistency level of an cluster indicates heavier loads on the primary database and lower cluster performance. We recommend that you use session consistency. This consistency level minimizes the impact on cluster performance and meets the requirements of most scenarios.

Global consistency

- Description

  In some scenarios, dependencies exist within individual sessions and between different sessions. For example, if you use a connection pool, requests that run on the same thread may be sent by using different connections. These requests belong to different sessions in the database. However, these requests depend on each other in the business process and session consistency cannot ensure data consistency. To solve this issue, provides global consistency.



After PolarProxy in your cluster receives a read request, PolarProxy checks the latest LSN on the primary node. For example, the latest LSN is LSN0. Internal batch operations are optimized to reduce the number of times that PolarProxy queries the latest LSN on the primary node. Then, after the LSNs of all read-only nodes are updated to LSN0, PolarProxy sends the read request to read-only nodes. This way, the data returned for the read request is the latest data updated before the read request is initiated.

The following table describes the two configuration parameters for global consistency.

Parameter	Description
ConsistTimeout	**Global Consistency Timeout**: The timeout period for updating the LSNs of read-only nodes to the latest LSN of the primary node. If the update operation times out, PolarProxy provided by performs the operation that is specified by the ConsistTimeoutAction parameter. Valid values: 0 to 300000. Default value: 20. Unit: milliseconds.
ConsistTimeoutAction	**Global Consistency Timeout Policy**: If the LSNs of read-only nodes cannot be updated to the latest LSN of the primary node within the timeout period specified by the ConsistTimeout parameter, PolarProxy provided by performs the operation that is specified by the ConsistTimeoutAction parameter. Valid values: ○ 0: PolarProxy sends read requests to the primary node. This is the default value. ○ 1: PolarProxy returns the `wait replication complete timeout, please retry` error message to the application.

> **Note** For more information about how to modify **Global Consistency Timeout** and **Global Consistency Timeout Policy**, see Configure PolarProxy.

- Scenarios

If the primary/secondary replication delay is high, a large number of requests may be forwarded to the primary node when you use global consistency. This increases the loads on the primary node and may increase service latency. Therefore, in scenarios in which a large number of read requests and a small number of write requests are processed, we recommend that you use global consistency.

Best practices for consistency levels

- A higher consistency level of an cluster indicates lower cluster performance. We recommend that you use session consistency. This consistency level minimizes the impact on cluster performance and meets the requirements of most scenarios.
- If you require high data consistency between different sessions, you can select one of the following solutions:
  - Use hints to forcibly send specific queries to the primary node.

    ```
    /*FORCE_MASTER*/ select * from user;
    ```

    > **Note**
    > - If you want to execute the preceding statement that contains the hint on the official command line of MySQL, add the -c parameter in the statement. Otherwise, the hint becomes invalid because the official command line of MySQL filters out the hint. For more information, see mysql Client Options.
    > - Hints have the highest priority for routing and are not limited by consistency levels or transaction splitting. Before you use hints, evaluate the impacts on your business.
    > - Hints cannot contain statements that change environment variables. For example, if you execute the `/*FORCE_SLAVE*/ set names utf8;` statement, errors may occur.

  - Use global consistency.

6.3.4. Connection pools

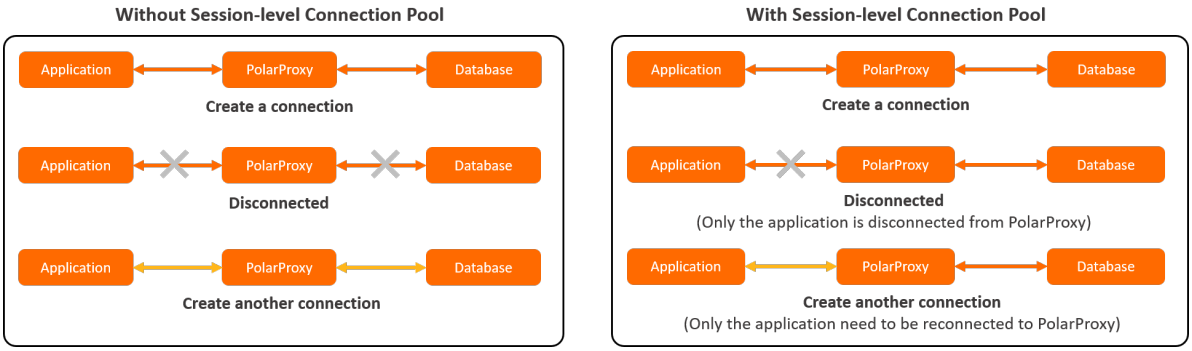
supports session-level connection pools and transaction-level connection pools. You can select a connection pool based on your business requirements to reduce the database loads caused by a large number of connections.

Note

- If you modify the connection pool configuration, the new configuration takes effect only on the connections created after the modification. For more information, see Configure PolarProxy.
- If you need to enable the connection pool feature for an account of a database, the client IP addresses that you use to connect to the database must be granted the same permissions. If you enable the connection pool feature and grant different database or table permissions to these IP addresses, a permission error may occur. For example, `user@192.xx.xx.1` is granted the `database_a` permission and `user@192.xx.xx.2` is not granted the `database_a` permission. In this case, a permission error may occur if the connection from one of the client IP addresses to the database is reused.
- This topic describes the connection pool feature provided by PolarProxy. You can use this feature at the same time as the connection pool feature of your client. If your client provides a connection pool, you do not need to enable the connection pool feature of PolarProxy.

Session-level connection pools

- How a session-level connection pool works

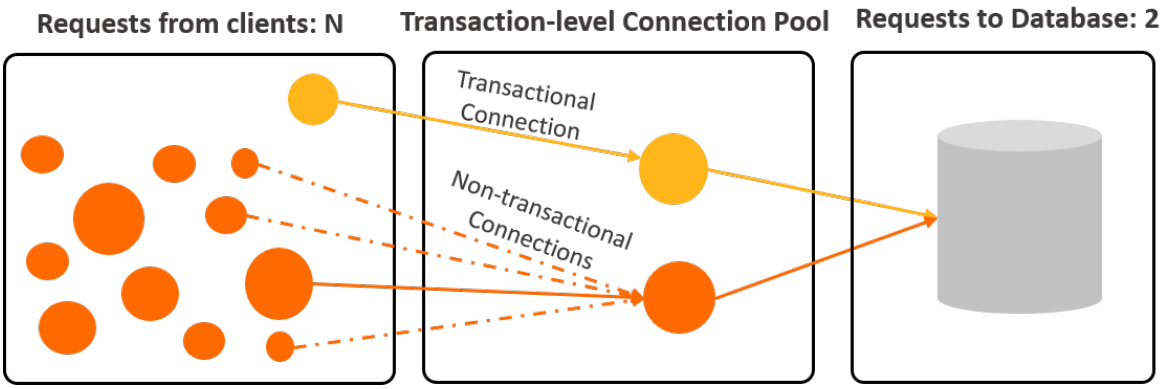


Frequent short-lived connections increase the loads on a database. Session-level connection pools allow you to reduce the loads that are caused by frequent short-lived connections. After a client is disconnected, the system checks whether the connection is idle. If the connection is idle, PolarProxy retains the connection in the connection pool for a short period. When a new request is received, the system provides the idle connection if the idle connection matches the conditions specified by the request parameters such as `user`, `clientip`, and `dbname`. This reduces the overhead required to establish a new connection to a database. If no idle connection in the connection pool matches the conditions, the system creates a new connection to the database.

- Limits
  - Session-level connection pools cannot be used to reduce the number of concurrent connections to a database. Session-level connection pools can be used to reduce only the frequency at which connections are established from applications to the database. This way, the number of main threads consumed in MySQL is reduced and the service performance is improved. However, the connections to the database still include the idle connections that are retained in the connection pool.
  - Session-level connection pools cannot resolve the issue of pending connections that are caused by a large number of slow SQL statements. The key to addressing this issue is to minimize the number of slow SQL statements.

Transaction-level connection pools

- How a transaction-level connection pool works



Transaction-level connection pools are used to reduce the number of direct connections to a database and the loads that are caused by frequent short-lived connections.

After the transaction-level connection pool feature is enabled, you can establish thousands of connections between clients and PolarProxy. However, only dozens or hundreds of connections are established between PolarProxy and backend databases.

The maximum number of connections to a cluster endpoint varies based on the specifications of the compute nodes in backend databases. If the transaction-level connection pool feature is disabled, the system must create a connection on the primary node and a connection on each read-only node each time a client sends a request.

After the transaction-level connection pool feature is enabled, the client that sends a request first connects to PolarProxy. PolarProxy does not immediately establish a connection between the client and the database. PolarProxy checks whether an idle connection in the transaction-level connection pool matches the conditions specified by the request parameters such as `user`, `dbname`, and the system variable. If no idle connection matches the conditions, PolarProxy creates a new connection to the database. If an idle connection matches the conditions, PolarProxy reuses the connection. After the transaction is committed, the connection is retained in the connection pool for other requests.

- Limits
  - If you perform one of the following operations, the connection is locked until the connection is closed. The locked connection is no longer retained in the connection pool and becomes unavailable to other requests.
    - Execute a PREPARE statement.
    - Create a temporary table.
    - Modify user variables.
    - Receive a large number of log entries. For example, you can receive log entries of more than 16 MB.
    - Execute a LOCK TABLE statement.
    - Execute multiple statements by using one statement string.
    - Call a stored procedure.
  - The FOUND_ROWS, ROW_COUNT, and LAST_INSERT_ID functions are not supported. These functions can be called, but may return inaccurate results. The following list provides the compatibility details about these functions:
    - PolarProxy V1.13.11 or later allows you to execute the `SELECT FOUND_ROWS()` statement that directly follows the `SELECT SQL_CALC_FOUND_ROWS * FROM t1 LIMIT *` statement. However, we recommend that you use `SELECT COUNT(*) FROM tb1` instead of `SELECT FOUND_ROWS()` in queries. For more information, see FOUND_ROWS().
    - The `INSERT` `SELECT LAST_INSERT_ID()` statement can be executed to ensure that the results are correct.
  - If the `wait_timeout` parameter is specified for a connection, the connection to the client may not time out. The `wait_timeout` parameter does not affect the connection between PolarProxy and the client. This is because the system assigns a connection in the connection pool to each request. If the time specified by `wait_timeout` is reached, the system closes only the connection between PolarProxy and the database, but retains the connection between PolarProxy and the client.

- The connection pool matches requests to connections by using the `sql_mode`, `character_set_server`, `collation_server`, and `time_zone` variables. If the requests include other session-level system variables, you must execute SET statements to specify these variables after the connections are established. Otherwise, the connection pool may reuse connections for which system variables are changed.
- Connections may be reused. Therefore, after you execute the `SELECT CONNECTION_ID()` statement, different thread IDs may be returned for the same connection.
- Connections may be reused. Therefore, the IP addresses and port numbers in the output of `show processlist` statement or the IP addresses and port numbers displayed on the **SQL Explorer** page may be different from those of the client.
- PolarProxy merges the results of the SHOW PROCESSLIST statement executed on each node and then returns the final result to the client. After the transaction-level connection pool feature is enabled, the thread ID of the connection between the client and PolarProxy is different from that between PolarProxy and the database. As a result, when you run the KILL command, an error may be returned even if the command is run as expected. You can execute the SHOW PROCESSLIST statement to check whether the connection is closed.

How to select a connection pool

You can determine whether to enable the connection pool feature and select a type of connection pool based on the following recommendations:

- Your service requires a small number of connections and most of the required connections are persistent connections, or a connection pool is available for your service. In this case, you do not need to enable the connection pool feature provided by .
- Your service requires a large number of connections such as tens of thousands of connections, or your service is a serverless service that does not run in scenarios described in the limits of transaction-level connection pools. In this case, you can enable the transaction-level connection pool feature. In a serverless service, the number of connections linearly increases based on the scaling-up or scaling-out of servers.
- Your service requires only short-lived connections and runs in a scenario that is described in the limits of transaction-level connection pools. In this case, you can enable the session-level connection pool feature.

6.3.5. Persistent connections

supports the persistent connection feature to prevent temporary service interruptions or connection failures. These issues can be caused by O&M activities, such as specification upgrades, switchovers, and minor version upgrades. The issues can also be caused anomalies such as server malfunctions. Persistent connections can improve the availability of .

Prerequisites

- The PolarProxy version of the cluster is 2.4.7 or later.

  > **Note** If the PolarProxy version is earlier than 2.4.7 and you require persistent connections, .

- The cluster is a 5.6, 5.7, or 8.0 cluster of .

Background information

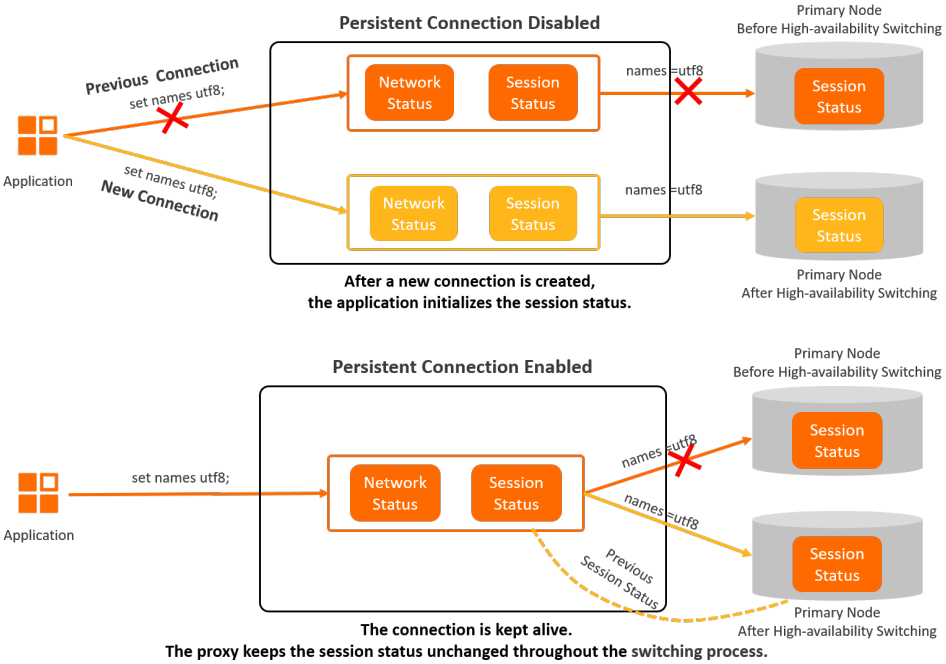
supports primary/secondary failovers by using high-availability components. This ensures that the clusters are highly available. However, the failovers may adversely affect your service and cause issues, such as temporary service interruptions or connection failures. Your application may be temporarily disconnected from the cluster in the following scenarios:

- Switchovers: Switchovers are triggered by O&M activities performed in the console or by the backend controller, such as specification upgrades, Automatic failover and manual failover, and minor version upgrades.
- Failovers: Failovers are triggered by anomalies, such as primary node failures or server malfunctions.

In most cases, you can restart the application or configure the application with the automatic reconnection mechanism to resolve these issues. However, these issues may not be taken into account in the early stages of the development due to the short development lifecycle. This leads to a large number of exceptions or service interruptions. supports the persistent connection feature to prevent connectivity issues caused by O&M activities or anomalies, such as specification upgrades, switchovers, minor version upgrades, or server malfunctions. Persistent connections can improve the availability of .

How connections are kept alive

Each session in a cluster consists of a frontend connection between the application and PolarProxy, and a backend connection between PolarProxy and the backend database. After the persistent connection feature is enabled, when PolarProxy disconnects from the current primary node and connects to a new primary node, the connection (the session shown in the application) between PolarProxy and the application is kept alive. PolarProxy establishes a connection to the new primary node and then restores the session to the state before the switchover is performed. This makes the entire switchover transparent to the application.



Typically, a MySQL session includes the following information: system variables, user variables, temporary tables, character set encoding, transaction status, and PREPARE statement status. In this topic, the status of character set encoding is used as an example to demonstrate how the status of a session changes before and after persistent connection is enabled.

A connection is established between the application and PolarProxy and the `set names utf8;` statement is executed. In this case, the session is in the `names=utf8` state. When PolarProxy connects to a new primary node, the session status must remain unchanged. Otherwise, a character set encoding error occurs. To prevent these errors, the session status must be kept unchanged after the switchover is complete.

> **Note** When PolarProxy connects to a new primary node, the original and new databases become temporarily inaccessible to both read and write requests. The downtime depends the database loads. During the database downtime, PolarProxy stops routing requests to both databases. PolarProxy resumes request distribution based on the following conditions:
>
> - If the new database recovers within 60 seconds, PolarProxy routes requests to the new database.
> - If the new database fails to recover within 60 seconds, PolarProxy disconnects from the application. The application must reconnect to PolarProxy. This issue also occurs when persistent connection is disabled.

Enable the persistent connection feature

- For a cluster that uses PolarProxy 2.4.7 or later, the persistent connection feature is enabled by default.
- For a cluster that uses a PolarProxy version earlier than 2.4.7, you must to enable the persistent connection feature.

Usage notes

Connections in the following scenarios cannot be kept alive:

- When PolarProxy connects to a new primary node, temporary tables exist within the session.
- When PolarProxy connects to a new primary node, a result message is in the process of being delivered from the database to PolarProxy. However, PolarProxy has received only a portion of the message. For example, after you execute a SELECT statement, a result message of 100 MB in size is returned to PolarProxy. However, PolarProxy receives only 10 MB of the message when the switchover is triggered.
- When PolarProxy connects to a new primary node, transactions in progress exist within the session, such as `begin;insert into;` .

> **Note** In the last two scenarios, if the switchover is triggered by a scheduled plan other than an anomaly, PolarProxy retains the connection for 2 seconds. Then, if the remaining message can be delivered or the transaction can be completed within 2 seconds, PolarProxy connects to the new primary node after the waiting period expires. This way, connections can be kept alive in more scenarios.

Performance benchmarking

- Environment
  - The following cluster is used for benchmarking:
    - A 8.0 cluster. By default, the cluster contains one primary node and two read-only nodes.
    - The node specification is 4-core 16 GB (polar.mysql.x4.large).

  - Test tool: SysBench.
  - Test data:
    - 20 tables are used in the test. Each table contains 10,000 rows.
    - The degree of parallelism is 20.

- Procedure

  Test the ratios of connections that are kept alive in the cluster before and after an O&M activity is performed.

- Test result

  In the following scenarios, the ratios of connections that are kept alive in the cluster are 100%.

  > **Note**
  > - The ratio of connections that are kept alive is 100% only when you upgrade cluster specifications tier by tier, such as from 4 cores to 8 cores. If you upgrade the cluster specifications from 4 cores to 16 cores or higher, a service interruption may occur.
  > - If the database proxy node is downgraded when a read-only node is removed, some connections may be closed.
  > - In this section, only the minor version upgrade of the database kernel engine is tested. This test does not include the minor version upgrade of the database proxy. During the minor version upgrade of the database proxy, network interruptions may occur.

Scenario	Ratio of connections that are kept alive
Switch to a new primary node	100%
Upgrade the minor version of the database kernel engine	100%
Upgrade cluster specifications	100%
Add or remove nodes	100%

6.4. Dynamic data masking

6.4.1. Overview

This topic provides an overview of the dynamic data masking feature provided by the proxy.

Prerequisites

The version of the proxy must be V2.4.12 or later. For more information about how to view and upgrade the version of the proxy, see Version Management.

> **Note** For the proxy of V1.x. x, you cannot upgrade the version of the proxy to V2.4.12 or later in the console. To upgrade the version of the proxy from V1.x. x to V2.4.12 or later, to contact technical support.

Data masking solutions

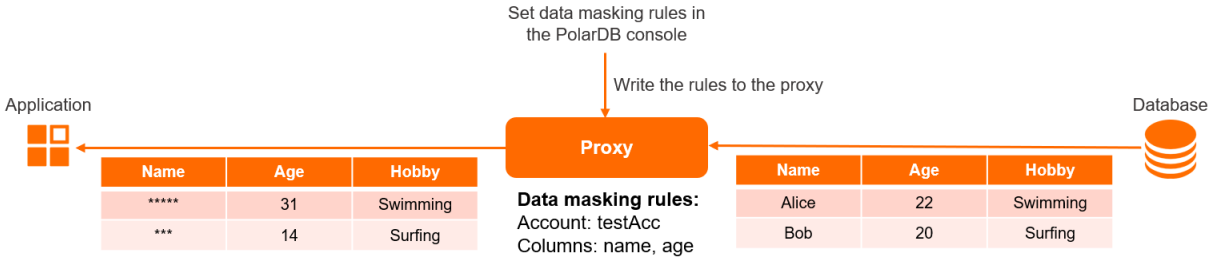
If you want to use third parties to generate reports, analyze data, and perform development and test activities, you may need to obtain the latest customer data from databases in the production environment in real time. To avoid disclosing personal information, data must be masked before it is provided to third parties. Alibaba Cloud provides the following data masking solutions: dynamic data masking and static data masking. The proxy uses dynamic data masking.

Comparison of data masking solutions

Data masking solution	Description	Advantage	Limits
Dynamic data masking	When your application initiates a data query request, the proxy masks the sensitive data that is queried before the proxy returns the data to the application. Before your application queries data, you need only to specify the database account and the name of the database, table, or column that requires data masking.	• You do not need to change code in your business system. This reduces costs. • Your application can query the real-time data from production databases.	Compared with mirror databases, production databases have lower query performance because the proxy masks the sensitive real-time data in the production databases.
Static data masking	The proxy exports all data in a production database to a mirror database, and encrypts or masks the sensitive data during the export.	Your application queries data from mirror databases instead of production databases. In this case, data masking does not affect the services that require access to production databases.	• You must develop a set of components used for masking the sensitive data in the data import toolkit. This incurs high development costs. • Data in mirror databases is not as up-to-date as data in production databases.

How it works

After you configure data masking rules in the console, the console writes these rules to the proxy. When your application connects to a database by using the account specified in the data masking rules and queries the specified columns, the proxy masks the data that is queried from the database and returns the masked data to the client.



The preceding figure shows the following data masking rules:

- The data masking rules take effect only when you use the `testAcc` account to query data from a database.
- The proxy masks only the data that is queried in the `name` and `age` columns.

If your application uses the `testAcc` account to connect to a database and queries data in the `name`, `age`, and `hobby` columns of a table, the proxy masks data in the `name` and `age` columns and returns the masked data together with the unmasked data in the `hobby` column.

The proxy uses different methods to mask different types of data. The following table describes data masking methods.

Data type	Data masking method	Example
Integer data types: TINYINT, SMALLINT, MEDIUMINT, INT, and BIGINT	The proxy returns a random value in the format defined in the data type of the raw data.	• Raw value: 12345 • Masked value that is randomly selected: 28175
Decimal data types: DECIMAL, FLOAT, and DOUBLE		• Raw value: 1.2345 • Masked value that is randomly selected: 8.2547
Date and time data types: DATE, TIME, DATETIME, TIMESTAMP, and YEAR		• Raw value: 2021-01-01 00:00:00 • Masked value that is randomly selected: 4926-12-13 17:23:07
Other data types	The proxy replaces the data with asterisks (*).	• Raw value: John Smith • Masked value: *********

Additional considerations

- The dynamic data masking feature applies only to cluster endpoints. Cluster endpoints consist of the default cluster endpoint and custom cluster endpoints. If you use the primary endpoint to connect to a database and query data from the database, the dynamic data masking feature does not take effect. For more information about how to view a cluster endpoint, see View an endpoint.

- If query results contain data that must be masked and the size of a single row exceeds 16 MB, the query session is closed.

  For example, you want to query data in the `name` and `description` columns of the `person` table. In this table, the sensitive data in the `name` column must be masked. The size of the data in a row of the `description` column exceeds 16 MB. In this case, when you execute the `SELECT name, description FROM person` statement, the query session is closed.

- If a column in which you want to mask the sensitive data is used as the value of an input parameter in a function, data masking does not take effect.

  For example, a data masking rule is created to mask the sensitive data in the `name` column. When you execute the `SELECT CONCAT(name, '') FROM person` statement, your application can still read the raw values of the `name` column.

- If a column in which you want to mask the sensitive data is used together with the UNION operator, data masking may not take effect.

  For example, a data masking rule is created to mask the sensitive data in the `name` column. When you execute the `SELECT hobby FROM person UNION SELECT name FROM person` statement, your application can still read the raw values of the `name` column.

Enable the dynamic data masking feature

For more information, see Manage data masking rules.

Appendix: Impacts on cluster performance

The dynamic data masking feature affects the performance of clusters in the following scenarios.

> **Note** In this example, the read-only queries per second (QPS) of clusters are used to show the difference in performance.

Scenario		Impact on performance
Whether your account is included in the data masking rule	Whether your query hits the data masking rule	
No	No	Data masking does not take effect on queries made by your account. This way, the performance of your cluster is not affected.
	No	The proxy analyzes only the column definition data in the result set and does not mask the raw data in the query results. This results in performance overhead of approximately 6%. After the dynamic data masking feature is enabled, the read-only QPS decreases by approximately 6%.

Scenario		
Whether your account is included in the data masking rule	Whether your query hits the data masking rule	Impact on performance
Yes	Yes	The P proxy analyzes the column definition data in the result set and masks the raw data in the query results. In this case, performance overhead is based on the size of the result set. A larger number of rows in the query results cause greater performance overhead. If the query result of a single row is returned, the performance overhead of approximately 6% occurs.

6.4.2. Manage data masking rules

You can create, modify, delete, enable, and disable data masking rules in the console. This topic describes how to manage data masking rules.

Prerequisites

The version of the proxy must be 2.4.12 or later. For more information about how to view and upgrade the version of PolarDB proxy, see Version Management.

Considerations

- The dynamic data masking feature applies only to cluster endpoints, including default cluster endpoints and customized cluster endpoints. When you query data from a primary endpoint, the dynamic data masking feature is not applied. For more information about how to view and apply for a cluster endpoint, see View an endpoint.
- If the query results contain data that needs to be masked and the size of a single row exceeds 16 MB, the query session is closed.

  For example, you want to query the `name` and `description` columns of the `Person` table in which the `name` column needs to be masked. However, the size of the data in a row of the `description` column exceeds 16 MB. In this case, the query session is closed when you execute the `SELECT name, description FROM person` statement.
- If the data column you want to mask is used as a function parameter, data masking is not applied.

  For example, if a rule has been created to mask data in the `name` column, your application can still read the actual value of the `name` column when you execute the `SELECT CONCAT(name, '') FROM person` statement.
- If the data column you want to mask is used in the UNION operator, data masking is not applied.

For example, if a rule has been created to mask data in the `name` column, your application can still read the actual value of the `name` column when you execute the `SELECT hobby FROM person UNION SELECT name FROM person` statement.

Create a data masking rule

1.
2.
3.
4. In the left-side navigation pane, choose **Settings and Management > Rules**.
5. In the upper-left corner of the page, click **Add**. In the **Create Data Masking Rule** dialog box, set the following parameters.

   Parameters for a data masking rule

Parameter		Required	Description
Basic Information	**Rule Name**	Yes	The name of the data masking rule. The name can be up to 30 characters in length.
	Description	No	The description of the data masking rule. The description is up to 64 characters in length.
	Enable/Disable	N/A	The **Enable/Disable** switch. **Note** When you create a data masking rule, the **Enable/Disable** switch is turned on by default.

Parameter		Required	Description
Configurations	Database Account Name	No	The name of the database account to which the rule is applied. The type of the account that is used to connect to the sandbox instance. Valid values: ◦ **All Accounts**: indicates that the data masking rule applies to all database accounts in the cluster. The text box on the right must be left empty. ◦ **Include**: indicates that the data masking rule applies only to specified database accounts. You must specify at least one database account name in the text box on the right. Separate multiple accounts with commas (,). ◦ **Exclude**: indicates that the data masking rule applies only to database accounts that are not specified in this section. You must specify at least one database account name in the text box on the right. Separate multiple accounts with commas (,). **Note** The database account names can be in one of the following formats: ◦ `account name`. Example: `user` ◦ `account name@full IP address`. Example: `user@1.1.1.1` ◦ `account name@IP address with wildcard characters`. Example: `user@1.1.1.%`, `user@%.1.1.1`, or `user@1.%.1` ◦ `account name@IP/subnet mask`. Example: `user@1.1.1.0/255.255.255.0`
	Database Name	No	The name of the database to which the rule is applied. The type of the account that is used to connect to the sandbox instance. Valid values: ◦ **All Databases**: indicates that the data masking rule applies to all the databases in the cluster. The text box on the right must be left empty. ◦ **Include**: indicates that the data masking rule applies only to specified databases. You must specify at least one database name in the text box on the right. Separate multiple database names with commas (,).

Parameter		Required	Description
	Table Name	No	The name of the table to which the rule is applied. The type of the account that is used to connect to the sandbox instance. Valid values: ○ **All tables**: indicates that the data masking rule applies to all the tables in the cluster. The text box on the right must be left empty. ○ **Include**: indicates that the data masking rule applies only to specified tables. You must specify at least one table name in the text box on the right. Separate multiple table names with commas (,).
	Column Name	Yes	The name of the field to which the rule is applied. You can specify more than one field name and separate multiple field names with commas (,).

6. Click **OK**.

Enable or disable a data masking rule

1.
2.
3.
4. In the left-side navigation pane, choose **Settings and Management > Rules**.
5. Locate the rule that you want to manage and turn **Enable/Disable** on or off.

> **Note**
> ○ You can select multiple rules in the rule list and then click **Enable** or **Disable** below the list to **Enable** or **Disable** the rules in batches.
> ○ **Disable** data masking rules will not be deleted. You can **Enable** the rules again based on your requirements.

6. In the dialog box that appears, click **OK**.

Modify a data masking rule

1.
2.
3.
4. In the left-side navigation pane, choose **Settings and Management > Rules**.
5. Locate the rule that you want to modify and click **Modify** in the right-side **Actions** column. In the dialog box that appears, configure the parameters based on your requirements. For more information about parameter descriptions, see Parameters for a data masking rule.

> **Note** You can modify the parameters only in **Description** and **Configurations**. Parameters in **Rule Name** cannot be modified.

6. Click **OK**.

Delete a data masking rule

1.
2.
3.
4. In the left-side navigation pane, choose **Settings and Management > Rules**.
5. Locate the rule that you want to delete and click **Delete** in the right-side **Actions** column.

> **Note** You can select multiple rules in the rule list. Then, click **Delete** below the list to delete the rules in batches.

6. In the dialog box that appears, click **OK**.

Related API operations

Operation	Description
DescribeMaskingRules	Queries the data masking rules that apply to a cluster or the details of a specified masking rule.
ModifyMaskingRules	Modifies or adds a data masking rule.
DeleteMaskingRules	Deletes a specified data masking rule.

6.5. Configure PolarProxy

This topic describes how to configure PolarProxy for a cluster by modifying the configuration of the cluster endpoint.

Prerequisites

A cluster of or is created. This feature is not supported for clusters of or . For more information about cluster editions, see Editions.

Precautions

- You can enable hybrid transaction/analytical processing (HTAP) and configure the degree of parallelism only when you configure PolarProxy for a 8.0 cluster.

Procedure

1.
2.
3.
4. In the **Endpoints** section on the **Overview** page, find the cluster endpoint that you want to modify and click **Modify** to the right of the cluster endpoint.
5. In the dialog box that appears, modify the configuration of the cluster endpoint based on your business requirements. The following table describes the parameters.

   Parameters

Parameter		Description
Network Information		provides a private endpoint for each cluster by default. You can modify the private endpoint or apply for a public endpoint. For more information, see Apply for a cluster endpoint or a primary endpoint.
Cluster Settings	**Read/write Mode**	The read/write mode of the cluster endpoint. You can select **Read Only** or **Read and Write (Automatic Read-write Splitting)**. **Note** You can change the read/write mode after a custom cluster endpoint is created. After you change the read/write mode, the new mode takes effect only on newly created connections. Existing connections remain in the original mode.
	Endpoint Name	The name of the cluster endpoint.
Node Settings	**Unselected Nodes** and **Selected Nodes**	Select the nodes that you want to associate with the cluster endpoint to process read requests from the **Unselected Nodes** list on the left. The Unselected Nodes list contains the primary node and all read-only nodes. Then click > to add the nodes to the **Selected Nodes** list on the right. **Note** The type of nodes that you select does not affect the read/write mode. ○ If you set the read/write mode to **Read and Write (Automatic Read-write Splitting)**, write requests are sent only to the primary node regardless of whether the primary node is selected. ○ If the read/write mode is **Read Only**, all read requests are forwarded to read-only nodes instead of the primary node in load balancing mode. Even if the primary node is added to the **Selected Nodes** list, read requests are not forwarded to it.

Parameter		Description
	Automatically Associate New Nodes	Specifies whether to automatically associate a new node with the cluster endpoint.
SLB Settings	**Load Balancing Policy**	Specifies the load balancing policy that is used to distribute read requests to multiple read-only nodes when read/write splitting is enabled. The default value is **Load-based Automatic Scheduling** and cannot be changed.
	Primary Node Accepts Read Requests	◦ If you set this parameter to **No**, read requests are sent only to read-only nodes to reduce the loads of the primary node. ◦ If you set this parameter to **Yes**, read requests are sent to the primary node and read-only nodes. For more information, see Offload reads from the primary node. **Note** This parameter is available only if the read/write mode is set to **Read and Write (Automatic Read-write Splitting)**.
	Transaction Splitting	Specifies whether to enable the transaction splitting feature. For more information, see Split transactions. **Note** This parameter is available only if the read/write mode is set to **Read and Write (Automatic Read-write Splitting)**.
Consistency Settings	**Consistency Level**	◦ If you set the read/write mode to **Read and Write (Automatic Read-write Splitting)**, the following consistency levels are available: **Eventual Consistency (Weak)**, **Session Consistency (Medium)**, and **Global Consistency (Strong)**. For more information, see Consistency levels. ◦ If you set the read/write mode to **Read Only**, the default consistency level is **Eventual Consistency (Weak)** and cannot be changed. **Note** Changes to the consistency level immediately take effect on all connections.
	Global Consistency Timeout	The timeout period of global consistency among read-only nodes. Unit: milliseconds. Valid values: 0 to 6000. Default value: 20. **Note** This parameter is available only if you set **Consistency Level** to **Global Consistency (Strong)**.

Parameter		Description
	Global Consistency Timeout Policy	The default policy to be applied if fails to achieve global consistency among read-only nodes within the specified timeout period. Valid values: ○ **Send Requests to Primary Node (Default)** ○ **SQL Exception: Wait replication complete timeout, please retry.** **Note** This parameter is available only if you set **Consistency Level** to **Global Consistency (Strong)**.
Connection Pool Settings	Connection Pool	You can set the parameter to **Off** (default), **Session-level**, or **Transaction-level**. For more information about connection pools, see Connection pools. **Note** This parameter is available only if you set the read/write mode to **Read and Write (Automatic Read-write Splitting)**.
TAP Optimization	Parallel Query	Specifies whether to enable parallel query and set the degree of parallelism. By default, the parallel query feature is disabled. For more information, see Parallel query. **Note** This parameter is available only if the read/write mode of the custom cluster endpoint of your cluster is **Read Only** and the cluster is a 8.0 cluster.
Dynamic Data Masking		You can create, enable, or disable data masking rules in the console. For more information, see Manage data masking rules. **Note** To use dynamic data masking, the PolarProxy version of the cluster must be 2.4.12 or later.

6. Click **OK**.

Related API operations

API	Description
DescribeDBClusterEndpoints	Queries the cluster endpoints of a PolarDB cluster.
ModifyDBClusterEndpoint	Modifies cluster endpoints.
DeleteDBClusterEndpoint	Deletes a custom cluster endpoint of a PolarDB cluster.

6.6. Upgrade the specifications of PolarProxy

The default specifications of PolarProxy Enterprise Edition are sufficient for most application scenarios that you may encounter. However, when you perform stress tests or have high service throughput, you should consider upgrading the specifications of PolarProxy Enterprise Edition to meet your business requirements. This topic describes the procedure to upgrade PolarProxy.

Context

The specifications of PolarProxy Enterprise Edition are automatically upgraded or downgraded based on the changes in the number of database nodes and node specifications. PolarProxy specification upgrade rules:

- For clusters, the number of vCPUs of the cluster nodes occupied by PolarProxy is rounded up from one sixth of the total vCPUs of the cluster nodes. It is rounded up to the nearest multiple of 2. The number of vCPUs can be 2 to 64.
- For clusters, the number of vCPUs of the cluster nodes occupied by PolarProxy is rounded up from one fourth of the total vCPUs of the cluster nodes. It is rounded up to the nearest multiple of 2. The number of vCPUs can be 2 to 128.

For example, a cluster has two nodes, and each node has four vCPUs. In this case, PolarProxy occupies two vCPUs.

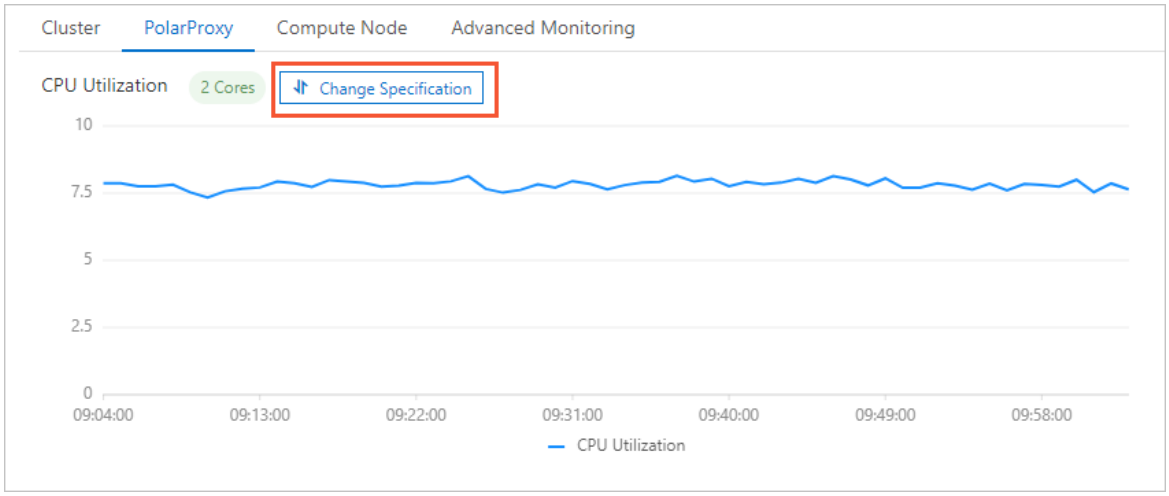
Precautions

During the upgrade, new sessions are established to the new PolarProxy. Sessions established to the original PolarProxy are automatically closed 24 hours after the upgrade starts. We recommend that you perform this upgrade operation during off-peak hours and make sure that your applications can automatically reconnect to the database service.

Procedure

This topic takes PolarProxy Enterprise Edition as an example to describe the procedure to upgrade PolarProxy.

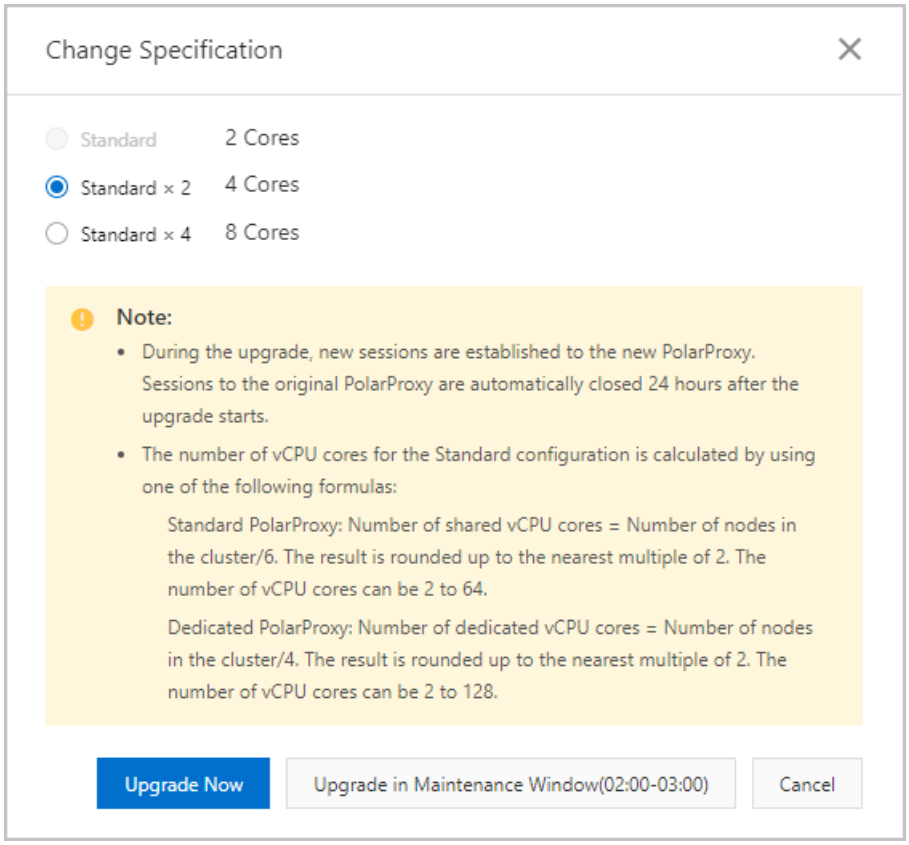
1.
2.
3. On the **Clusters** page, click the ID of the cluster.
4. In the left-side navigation pane, choose **Diagnostics and Optimization > Monitoring**. On the **PolarProxy** tab, click **Change Specification**.



5. In the **Change Specification** dialog box, select an upgrade plan.

   PolarProxy specifications upgrade plans:

   - Standard specification: For clusters, the number of vCPUs of the cluster nodes occupied by PolarProxy is rounded up from one sixth of the total vCPUs of the cluster nodes. For clusters, the number of vCPUs of the cluster nodes occupied by PolarProxy is rounded up from one fourth of the total vCPUs of the cluster nodes.
   - Standard specification × 2: The number of vCPUs is twice that of the standard specification.
   - Standard specification × 4: The number of vCPUs is four times that of the standard specification.



6. Click **Upgrade Now** or **Upgrade in Maintenance Window**.

   If you select **Upgrade in Maintenance Window**, you can view the details about the task or cancel the task on the **Scheduled Tasks** page. For more information, see View or cancel a

scheduled task.

7. In the dialog box that appears, click **OK**.

6.7. Performance Monitoring

PolarProxy Enterprise Edition allows you to monitor a variety of performance metrics and view monitoring data at intervals of seconds. You can monitor the status of your clusters and locate faults based on fine-grained monitoring data.

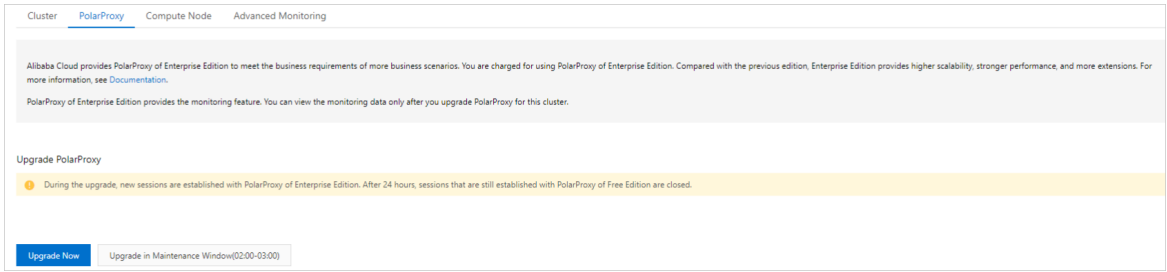
Precautions

- For PolarDB for MySQL clusters that were purchased before December 9, 2021, you must upgrade PolarProxy to PolarProxy Enterprise Edition to use the performance monitoring feature. For PolarDB for MySQL clusters that were purchased after December 9, 2021, the performance monitoring feature can be used directly.
- During the upgrade, new sessions are established to the new PolarProxy. Sessions established to the original PolarProxy are automatically closed 24 hours after the upgrade starts. We recommend that you perform this upgrade operation during off-peak hours and make sure that your applications can automatically reconnect to the database service.

Procedure

1. Enter the monitoring page of PolarProxy.
   i. Log on to the PolarDB console.
   ii. In the upper-left corner of the console, select the region in which the cluster that you want to manage is deployed.
   iii. Find the cluster and click the cluster ID.
   iv. In the left-side navigation pane, choose **Diagnostics and Optimization > Monitoring**.
   v. On the **Monitoring** page, click **PolarProxy**.
2. (Optional) Upgrade PolarProxy to PolarProxy Enterprise Edition.

   > **Note** For PolarDB clusters that were purchased before December 9, 2021, perform the following steps to upgrade PolarProxy. For PolarDB clusters that were purchased after December 9, 2021, ignore this operation.

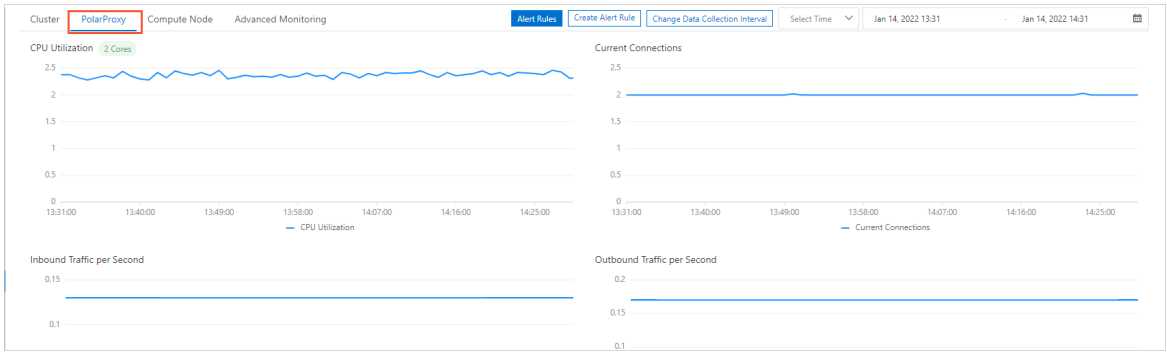
   i. On the **PolarProxy** tab, click **Change Specification** and then click Upgrade Now or **Upgrade in Maintenance Window**.

      If you select **Upgrade in Maintenance Window**, you can view the details about the task or cancel the task on the **Scheduled Tasks** page. For more information, see View or cancel a scheduled task.

ii. In the dialog box that appears, click **OK**.

3. View the monitoring data of the PolarProxy performance metrics.

   On the **PolarProxy** tab, you can view the PolarProxy monitoring metrics based on your business requirements.



   CPU Utilization indicates the usage of vCPUs of the cluster nodes occupied by PolarProxy. For clusters, the number of vCPUs of the cluster nodes occupied by PolarProxy is rounded up from one sixth of the total vCPUs of the cluster nodes. For clusters, the number of vCPUs of the cluster nodes occupied by PolarProxy is rounded up from one fourth of the total vCPUs of the cluster nodes.

   For example, a cluster has two nodes, and each node has four vCPUs. In this case, PolarProxy occupies two vCPUs.

6.8. FAQ

This topic provides answers to frequently asked questions (FAQ) about PolarProxy provided by .

- Why am I unable to retrieve a record immediately after I insert the record?

  In a read/write splitting architecture, a delay occurs when data is being replicated among the primary node and read-only nodes. supports session consistency to ensure that you can query updates within a session. You can retrieve the inserted record after the replication is complete. For more information, see the "Session consistency" section in Session consistency.

- Can data be read immediately after it is written into ?

  No, data cannot be read immediately after the data is written into . A delay of a few milliseconds occurs when you read data by using an endpoint for which read/write splitting is enabled, even if the loads on the primary node and read-only nodes of an cluster are not heavy. To eliminate this delay, you can use the primary endpoint to connect to the cluster. This way, read and write requests are sent to the primary node. For more information about how to view the primary endpoint, see View an endpoint.

- Why do low loads exist on read-only nodes when the loads on the primary node are high?

  By default, requests in transactions are routed only to the primary node. To balance loads across the primary and read-only nodes, you can use the following solutions:

  - For stress testing that uses Sysbench, add `--oltp-skip-trx=on` to your code if Sysbench 0.5 is used or `--skip-trx=on` if Sysbench 1.0 is used. This way, you do not need to execute the BEGIN and COMMIT statements.
  - In actual usage, a large number of transactions can cause heavy loads on the primary node. In this case, you can enable the transaction splitting feature to reduce the loads on the primary node. For more information, see Split transactions.

- Why does a node receive more requests than others?

  Requests are distributed to each node based on loads. The nodes on which lighter loads exist receive more requests.

- Does a new read-only node automatically receive read requests?

  This depends on whether a session that supports read/write splitting is created after you add the read-only node. If yes, requests are automatically forwarded to the read-only node. If no, read requests are not forwarded to the read-only node. In this case, you can close a connection and then reconnect to your cluster. This way, read requests sent over the connection are forwarded to the read-only node. For example, you can restart your application to establish a new connection.

- What are the differences between PolarProxy Enterprise Edition and PolarProxy Enterprise Edition ?
  - is used by clusters of the General-purpose type of Cluster Edition, which shares CPU resources with smart elastic scaling within seconds provided based on business loads.
  - is used by clusters of the Dedicated type of Cluster Edition, which exclusively uses allocated CPU resources and provides better stability.

- Does the previous free edition of PolarProxy continue to be available?

  No,

  - for existing clusters, PolarProxy was switched to PolarProxy Enterprise Edition on January 7, 2022. Currently, PolarProxy is free of charge.
  - For newly purchased clusters, they use only PolarProxy Enterprise Edition. Currently, PolarProxy is free of charge.

7.Database Management

You can create and manage all databases in the console.

Create a database

1.
2.
3.
4. In the left-side navigation pane, choose **Settings and Management > Databases**.
5. Click **Create Database**.
6. In the **Create Database** panel, configure the following parameters.

Parameter	Description
Database Name	○ The name must start with a letter and end with a letter or a digit. ○ The name can contain lowercase letters, digits, underscores (_), and hyphens (-). ○ The name can contain up to 64 characters in length. ○ The name must be unique in your PolarDB instance. **Note** Do not use reserved words as database names, such as `test` or `mysql`.
Supported Character Set	Select the character set supported by the database. The database supports the **utf8mb4**, **utf8**, **gbk**, and **latin1** character sets.
Authorized Account	Select the account that you want to authorize to access this database. You can leave this parameter empty and bind an account after the database is created. **Note** Only standard accounts are available in the drop-down list. Privileged accounts have all the permissions on all databases. You do not need to authorize the privileged accounts to access the database.
Account Permission	Select the permission that you want to grant to the selected account. Valid values: **Read&Write**, **ReadOnly**, **DMLOnly**, **DDLOnly**, or **ReadOnly&Index**.
Description	Enter a description for the database. The description helps facilitate subsequent database management. The description must meet the following requirements: ○ It cannot start with `http://` or `https://`. ○ It must be 2 to 256 characters in length.

7. Click **OK**.

Delete a database

1.
2.
3.
4. In the left-side navigation pane, choose **Settings and Management > Databases**.
5. Find the database that you want to delete and click **Delete** in the **Actions** column.
6. In the dialog box that appears, click **OK**.

Related API operations

API	Description
CreateDatabase	Creates a database.
DescribeDatabases	Queries the database list.
ModifyDBDescription	Modifies the description of a database.
DeleteDatabase	Deletes a database.

8.Modify Cluster Configurations

8.1. Overview

This topic describes the cluster specification change feature and its common scenarios.

allows you to dynamically change the specifications of a cluster within just a few minutes. You do not need to lock databases when you change the specifications of the cluster. The cluster specification change feature supports the following three types of scaling:

- Vertical scaling of computing capacity: The specifications of the cluster are upgraded or downgraded. You can change the specifications of the primary and read-only nodes separately. This means that the specifications of read-only nodes do not have to be consistent with those of the primary node.

  > **Note** clusters provide a more flexible specification change solution. The specifications of read-only nodes can be changed separately and do not need to be consistent with the specifications of the primary node. You can specify different specifications for different nodes based on your business requirements.

- Horizontal scaling of the computing capacity: Read-only nodes are added or removed. A maximum of 16 computing nodes can be added.
- Horizontal scaling of storage spaces: A serverless architecture is adopted. This allows the storage space to be automatically resized as the data volume changes. The maximum storage space for a single instance is 100 TB. If you need to store a large volume of data, we recommend that you purchase a storage plan to reduce costs. For more information, see Purchase a storage plan.

Cluster specification changes support the following scenarios:

- **Manually change specifications**: You can manually change the specifications of your clusters. For more information, see Manually upgrade or downgrade a PolarDB cluster.

  > **Note** Only Edition or are supported. and are not supported.

- **Automatically change specifications (auto-scaling)**: You can specify the parameters of the automatic specification change feature. Edition instances automatically scale out and scale in based on the parameters that you configured. For more information, see Automatic configuration changes (auto scaling) and Automatically scale local resources.
- **Temporary cluster upgrade**: To improve the overall performance, you can temporarily upgrade the specifications of clusters. When the rollback time is reached, the cluster is automatically rolled back to the original specifications. For more information, see Perform a temporary cluster upgrade.

  > **Note** Only subscription clusters support temporary upgrades.

- **Add and delete nodes**: You can manually add or delete read-only nodes. For more information, see Add or remove read-only nodes.

  > **Note** Only Edition, Edition, or are supported. and are not supported.

- **Upgrade Archive Database Single Node Edition to Archive Database Cluster Edition**: You

can upgrade existing clusters to clusters. For more information, see Upgrade an Archive Database Standalone Edition cluster to an Archive Database Cluster Edition cluster.

8.2. Manually upgrade or downgrade a PolarDB cluster

allows you to change the specifications of a cluster to meet business requirements. If your PolarDB cluster does not need to handle workload fluctuations, we recommend that you manually upgrade or downgrade the cluster. This topic describes how to manually upgrade or downgrade a cluster.

Prerequisites

- The cluster is of the Edition or . and are not supported. For more information about cluster editions, see Editions.
- No upgrade or downgrade task is being performed for the cluster.

Billing

For information about the billing rules, see Configuration change fees.

Precautions

- When you upgrade or downgrade a cluster, data stored in the cluster is not affected.
- The time for specification changes is related to the number of compute nodes in the cluster. It takes about 5 minutes per compute node. For example, if a cluster contains two compute nodes, it takes about 10 minutes to change the specifications. In addition, the time for specification changes is also related to factors such as database loads and the number of databases and tables.
- During the upgrade or downgrade process, a transient connection to each endpoint occurs for no more than 30 seconds. We recommend that you upgrade or downgrade your cluster during off-peak hours and make sure that your applications are configured with the automatic reconnection mechanism.
- A cluster upgrade or downgrade has minor impacts on the primary node of a cluster. However, the upgrade or downgrade degrades the performance of read-only nodes in the cluster. As a result, the read-only nodes require more time to handle requests during the upgrade or downgrade process.
- You can upgrade or downgrade both clusters and individual nodes in a cluster.

Select different specifications for primary node and read-only nodes

For a cluster, you can separately upgrade or downgrade the primary node and read-only nodes. This means that you can select different specifications for the primary node and read-only nodes.

Limits

- In a cluster, ensure that at least one read-only node has the same specifications as the primary node.
- We recommend that you select close specifications for the primary node and read-only nodes:
  - The memory of a read-only node cannot be less than half of the memory of the primary node.

- The following table lists the correspondence of CPU specifications for the primary node and read-only nodes.

CPU cores for primary node	Minimum CPU cores for read-only node
2	2
4	4
8	4
16	8
32	16
64	32
88	32

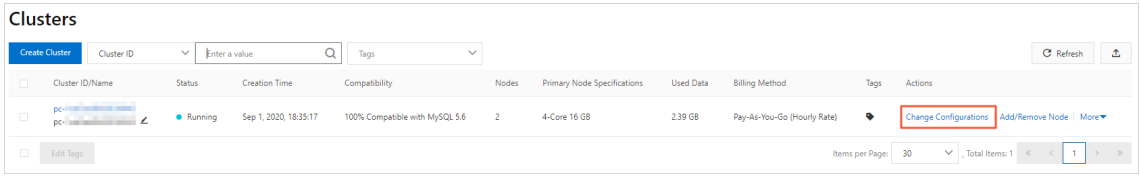
- Read-only nodes must be of the same specifications as the primary node if hot standby nodes are configured.
- We recommend that read-only column store nodes use higher specifications than the primary node if read-only column store nodes are added.

> **Note** Edition clusters are not subject to the preceding limits.

Procedure

1.
2.
3. You can open the **Change Configurations** dialog box by using one of the following methods:
   - Method 1

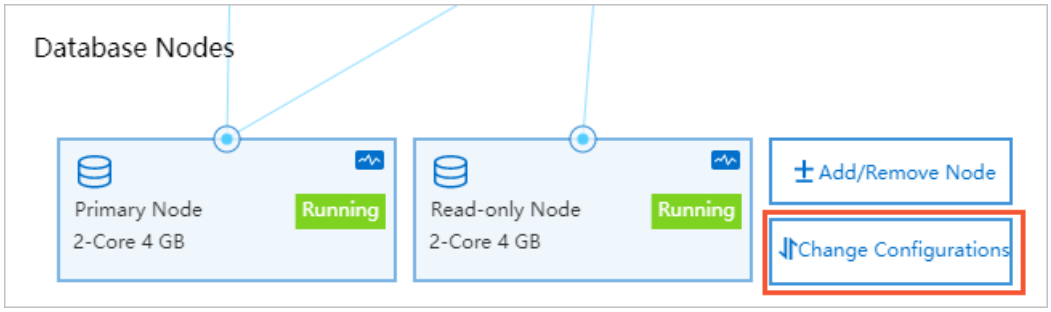
     On the **Clusters** page, find the cluster that you want to manage and click **Change Configurations** in the **Actions** column.

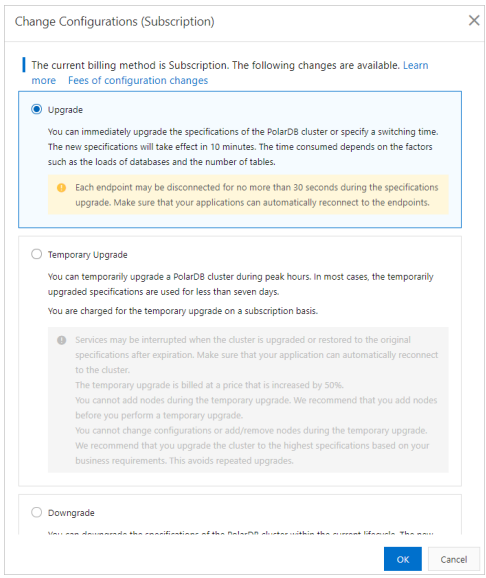
   - Method 2

     a. On the **Clusters** page, click the ID of the cluster.

b. On the **Overview** page, click **Change Configurations** in the **Database Nodes** section.



4. Select **Upgrade** or **Downgrade** based on your business requirements and click **OK**.



> **Note** Only subscription clusters support **Temporary Upgrade**. For more information, see Perform a temporary cluster upgrade.

5. On the **Upgrade/Downgrade** page, select a node specification, specify **Switching Time**, read and accept the terms of service, and then click **Buy Now**.

> **Note**
>
> - In a cluster, ensure that at least one read-only node has the same specifications as the primary node.
> - You can upgrade or downgrade both clusters and individual nodes in a cluster.
> - You can specify **Switching Time** only after you select **Upgrade** or **Downgrade**. If you select **Temporary Upgrade**, you cannot specify Switching Time.
> - You can set **Switching Time** to **Switch Now** or **Switch At**. If you select **Switch At**, you can specify a point in time within the following 24 hours. Your cluster will be upgraded or downgraded within 30 minutes after the specified point in time. You can view the details about the scheduled task on the **Scheduled Tasks** page, or cancel the task. For more information, see View or cancel a scheduled task.

6. On the **Purchase** page, confirm the order information and click **Purchase**.

> Note
>
> - During the upgrade or downgrade process, your applications are temporarily disconnected from each endpoint for no more than 30 seconds. We recommend that you upgrade or downgrade your cluster during off-peak hours and make sure that your applications are configured with the automatic reconnection mechanism.
> - The time for specification changes is related to the number of compute nodes in the cluster. It takes about 5 minutes per compute node. For example, if a cluster contains two compute nodes, it takes about 10 minutes to change the specifications. In addition, the time for specification changes is also related to factors such as database loads and the number of databases and tables.

Related API operations

Operation	Description
ModifyDBNodeClass	Changes the specifications of a cluster.
ModifyDBNodesClass	Change the specifications of a single node in a cluster.

8.3. Automatically change specifications

Context

1.

8.3.1. Automatic configuration changes (auto scaling)

You can configure the parameters for automatic configuration changes on the **Basic Information** page of your cluster. The cluster automatically scales out or scales in based on the parameters that you configure. This topic describes how to configure the parameters for automatic configuration changes on the **Basic Information** page of a cluster.

Limits

- The edition of the cluster is or . and are not supported. For more information about cluster editions, see Editions.
- No upgrade or downgrade task is being performed for the cluster.
- clusters that use the pay-as-you-go and subscription billing methods all support auto scaling.

Billing

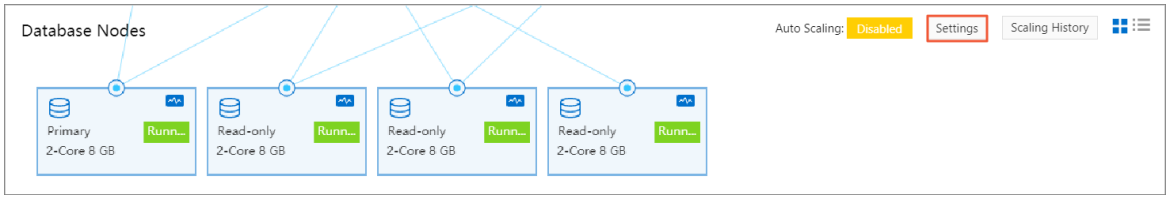
For information about the billing rules, see Configuration change fees.

Precautions

- You can upgrade or downgrade only clusters. You cannot upgrade or downgrade a single node in a cluster.
- When you upgrade or downgrade a cluster, data stored in the cluster is not affected.
- During the upgrade or downgrade process, your applications are temporarily disconnected from the cluster for no more than 30 seconds. We recommend that you upgrade or downgrade your cluster during off-peak hours and make sure that your applications are configured with the automatic reconnection mechanism.
- A cluster upgrade or downgrade has minor impacts on the primary node of a cluster. However, the upgrade or downgrade degrades the performance of read-only nodes in the cluster. As a result, the read-only nodes require more time to handle requests during the upgrade or downgrade process.

Procedure

1.
2.
3. On the **Clusters** page, click the ID of the cluster.
4. On the **Overview** page, click **Settings** in the upper-right corner of the **Database Nodes** section.



5. In the dialog box that appears, specify the following parameters. The following table describes the parameters.

Parameter	Description
Auto Scaling-out	Specifies whether to enable the auto scale-out feature.
Observation Period	If the CPU utilization is greater than or equal to the specified value during the observation period, automatically adds nodes or upgrades the specifications of your cluster after the observation period expires. This ensures that the cluster can handle the incoming read and write requests. For example, if the observation period is 5 minutes and the time required to complete a scaling activity is 10 minutes, you must wait 15 minutes before you can check the scaling result.
CPU Usage	The threshold that is used to trigger upgrades or scale-out activities. If the **CPU Usage** is greater than or equal to the specified value, an auto scale-out activity is triggered.
Maximum Specification	The highest specifications to which a cluster can be upgraded. After an upgrade is triggered, the specifications of a cluster is upgraded tier by tier until the highest specifications are applied. For example, the cluster is upgraded from 4 cores to 8 cores, and then to 16 cores.

Parameter	Description
Maximum Number of Read-only Nodes	The maximum number of read-only nodes that can be automatically added to the cluster. After a scale-out activity is triggered, read-only nodes are automatically added to a cluster one after another until the specified upper limit is reached. **Note** The nodes that are automatically added are associated with the default cluster endpoint. If the cluster uses a custom endpoint, you must specify whether to associate these nodes with the endpoint by specifying the **Automatically Associate New Nodes** parameter. For more information about **Automatically Associate New Nodes**, see Configure PolarProxy.
Auto Scaling-in	Specifies whether to enable the auto scale-in feature. **Note** After you turn on Auto Scaling-in, if the CPU utilization remains at less than 30% for more than 99% of the silent period, an automatic scale-in activity is triggered after the silent period ends. The specifications of the cluster are scaled in to the original specifications in small increments.
Quiescent Period	The minimum interval between two scaling activities. During a silent period, monitors the resource usage of a cluster but does not trigger scaling activities. If a quiescent period and an observation period end at the same time and the CPU utilization reaches the threshold value within the observation period, automatically triggers the auto scaling operation.

6. Click **OK**.

8.3.2. Configure the auto scaling feature of DAS

The **diagnostics** feature of clusters is integrated with some features of Database Autonomy Service (DAS), such as auto scaling. You can enable the auto scaling feature on the **Autonomy Center** tab in the PolarDB console. This topic describes how to enable the auto scaling feature.

Limits

- The cluster is of the Edition. , , and are not supported. For more information about cluster editions, see Editions.
- No upgrade or downgrade task is being performed for the cluster.
- clusters that use the pay-as-you-go and subscription billing methods all support auto scaling.

Billing

For information about the billing rules, see Configuration change fees.

Precautions

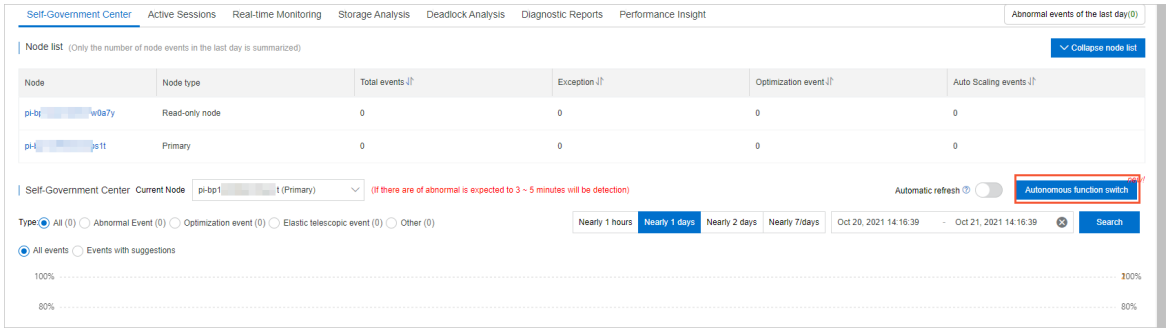
- The auto scaling feature of DAS **does not support separate auto scale-out or scale-up of the primary node and read-only nodes. After an auto scale-in or scale-down task is performed, all read-only nodes use the same specification as the primary node. Proceed with caution.**

  > **Note** In the following case: A cluster contains more than two nodes, read-only nodes use different specifications from the primary node, the primary node and read-only node specifications are inconsistent, **Automatic Scale-up/out** is enabled, and the average CPU utilization is greater than or equal to the specified value within the entire observation window. can select one of the following scale-out or scale-up methods depending on the real-time read/write traffic of the cluster:
  >
  > - **Add a read-only node that uses the same specification as the primary node.**
  > - **Add a read-only node that uses the same specification as an existing read-only node in the cluster.**
  >
  > **Note** This method is applicable to the scenario where the average CPU utilization of read-only nodes is greater than or equal to the specified value.
  >
  > - **Upgraded read-only nodes to the same specification as the primary node.**
  >
  > If the CPU utilization remains at less than 30% for more than 99% of the silent period after you enable **Automatic Scale-down/in**, auto scale-in or scale-down tasks are triggered step by step for the cluster when the silent period ends. The following scale-in or scale-down methods may be used:
  >
  > - **Delete a read-only node that you have added.**
  >
  > **Note** The specifications of remaining read-only nodes stay unchanged.
  >
  > - **Downgrade all read-only nodes to the same specifications as the primary node.**

- When you upgrade or downgrade a cluster, data stored in the cluster is not affected.
- During the upgrade or downgrade process, your applications are temporarily disconnected from the cluster for no more than 30 seconds. We recommend that you upgrade or downgrade your cluster during off-peak hours and make sure that your applications are configured with the automatic reconnection mechanism.
- A cluster upgrade or downgrade has minor impacts on the primary node of a cluster. However, the upgrade or downgrade degrades the performance of read-only nodes in the cluster. As a result, the read-only nodes require more time to handle requests during the upgrade or downgrade process.

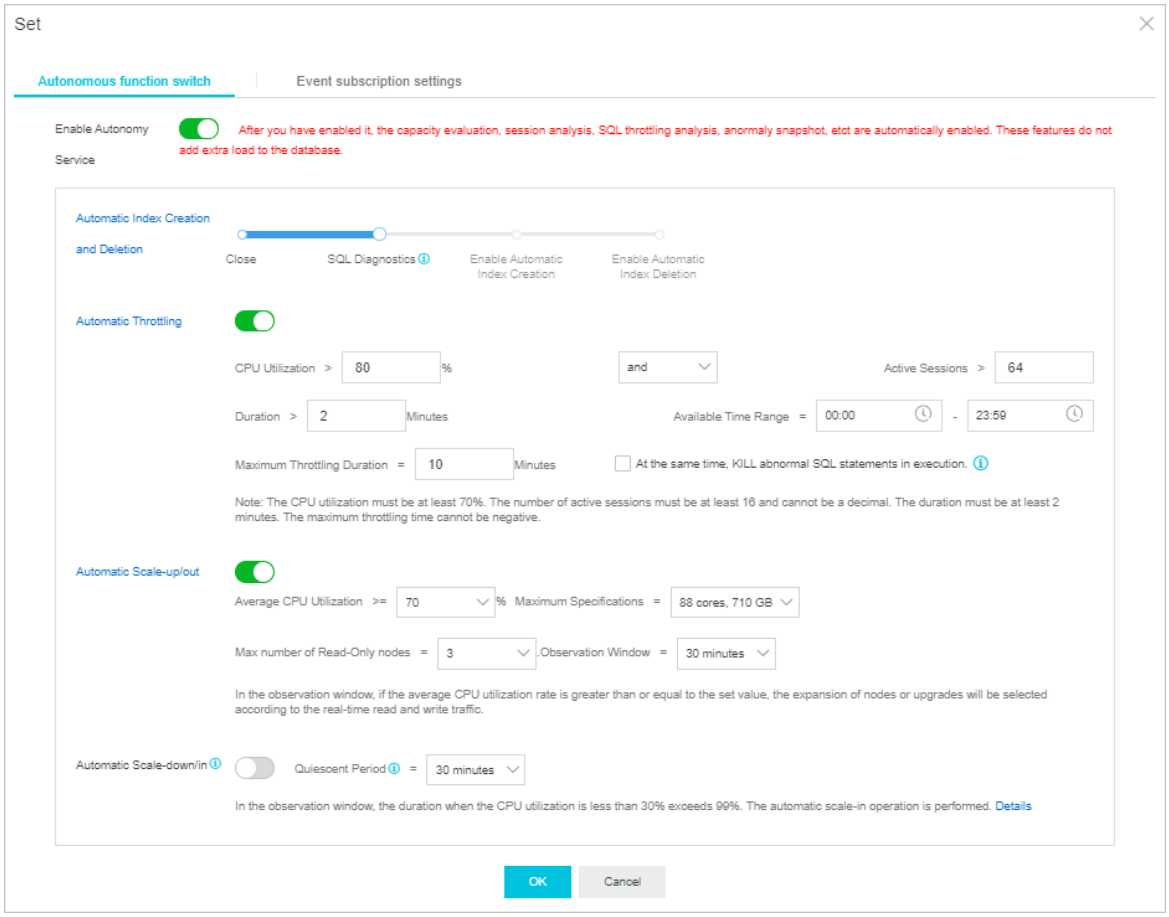
Procedure

1.
2.
3. On the **Clusters** page, click the ID of the cluster.
4. In the left-side navigation pane, choose **Diagnostics and Optimization > Diagnosis**.
5. On the page that appears, click the **Autonomy Center** tab.
6. In the upper-right corner of the **Autonomy Center** section, click **Autonomy Service Settings**.



7. On the **Autonomy Service Settings** tab of the **Settings** dialog box, turn on **Enable Autonomy Service** and then turn on **Automatic Scale-up/out** and **Automatic Scale-down/in**.

> **Note**
>
> - After you turn on **Automatic Scale-up/out**, if the average CPU utilization reaches or exceeds the specified threshold during the observation period, automatically upgrades or scales out the cluster after the observation period expires. This ensures that the cluster can handle all read and write requests. For example, if the observation period is 5 minutes and the time required to complete a scaling activity is 10 minutes, you must wait 15 minutes before you can check the scaling result.
> - If the CPU utilization remains at less than 30% for more than 99% of the silent period after you enable **Automatic Scale-down/in**, auto scale-in or scale-down tasks are triggered step by step for the cluster when the silent period ends. The auto scale-in process ends when the cluster uses its initial specifications.



8. Specify the **Automatic Scale-up/out** and **Automatic Scale-down/in** parameters and click **OK**.

Parameter	Description
Average CPU Utilization	The threshold that is used to trigger automatic scale-up. When the average CPU utilization reaches or exceeds the specified threshold, an upgrade or scale-out activity is automatically triggered.
Maximum Specifications	The highest specifications to which a cluster can be upgraded. After an upgrade is triggered, the specifications of a cluster is upgraded tier by tier until the highest specifications are applied. For example, the cluster is upgraded from 4 cores to 8 cores, and then to 16 cores.

Parameter	Description
Max number of Read-Only Nodes	The maximum number of read-only nodes that can be automatically added to the cluster. After a scale-out activity is triggered, read-only nodes are automatically added to a cluster one after another until the specified upper limit is reached. **Note** The nodes that are automatically added are associated with the default cluster endpoint. If the cluster uses a custom endpoint, you must specify whether to associate these nodes with the endpoint by specifying the **Automatically Associate New Nodes** parameter. For more information about how to configure **Automatically Associate New Nodes**, see Configure PolarProxy.
Observation Window	If the average CPU utilization reaches or exceeds the specified threshold during the observation period, automatically upgrades or scales out the cluster after the observation period expires. This ensures that the cluster can handle all read and write requests. For example, if the observation period is 5 minutes and the time required to complete a scaling activity is 10 minutes, you must wait 15 minutes before you can check the scaling result.
Quiescent Period	The minimum interval between two automatic scale-up operations or two automatic scale-down operations. During a silent period, monitors the resource usage of a cluster but does not trigger scaling activities. If a silent period and an observation period expire at the same time and the average CPU utilization within the observation period reaches the threshold, automatically triggers an upgrade.

8.3.3. Automatically scale local resources

PolarDB for MySQL allows you to scale local resources within several seconds. When the average CPU utilization of a PolarDB cluster within a sampling duration reaches the specified threshold, PolarDB for MySQL increases the number of CPU cores of the cluster to the specified value.

Prerequisites

- Only standard PolarDB for MySQL clusters of the Cluster Edition support this feature.

- To use this feature, you must create an AliyunServiceRoleForDAS role in the Database Autonomy Service (DAS) console. For more information, see AliyunServiceRoleForDAS role.

Usage notes

- This feature is provided free of charge.
- If you want to enable automatic scaling for local resources, we recommend that you disable automatic configuration changes for PolarDB clusters. These two features are not compatible with each other. If you enable automatic configuration changes for PolarDB clusters when automatic scaling for local resources is enabled, automatic scaling for local resources will fail.

Comparison between automatic scaling for local resources and automatic configuration changes for PolarDB clusters

The following table describes the differences between automatic scaling for local resources and Automatic configuration changes (auto scaling) for PolarDB clusters:

Item	Automatic scaling for local resources	Automatic configuration changes for PolarDB clusters
Scalable resources	You can scale only CPU cores and IOPS, but cannot scale memory and connections.	You can scale CPU cores, IOPS, memory, and connections at the same time.
Scaling speed	Quick scaling: • The minimum sampling duration is 30 seconds. • When a scale-up task is triggered, it can be completed within several seconds.	Slow scaling: • The minimum sampling duration is 5 minutes. • Scaling takes a longer time because the specifications of the cluster are changed.
Scaling process	No network interruptions occur during the scaling process because specification changes are not required.	Network interruptions occur because specification changes are required.

Based on the preceding comparison, automatic scaling for local resources has the following advantages:

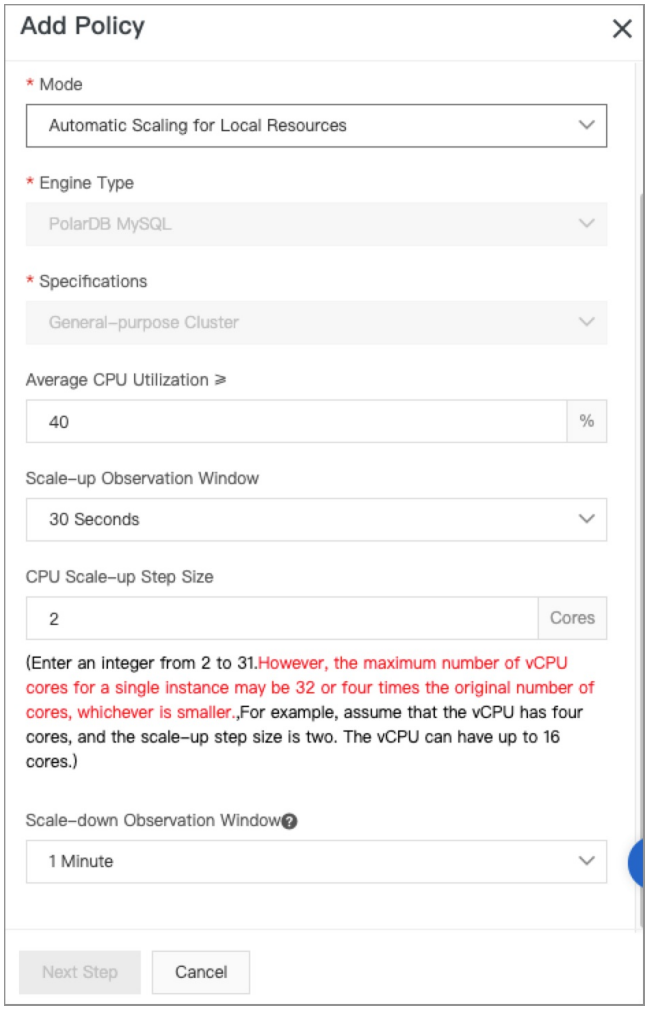
- A scale-up task can be completed within several seconds.
- No network interruptions occur during the scaling process.

Procedure

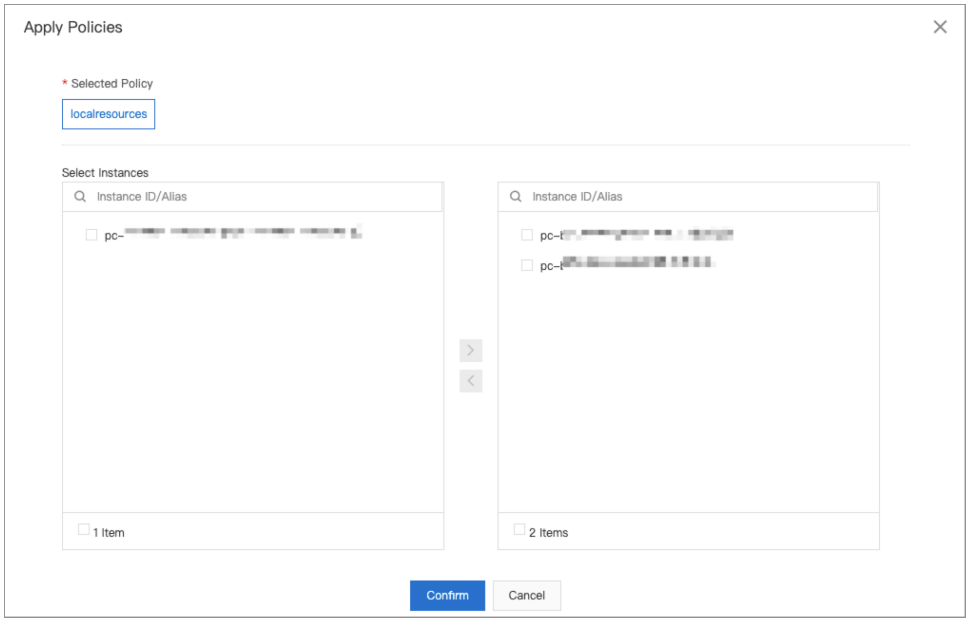
1. Log on to the DAS console.
2. In the left-side navigation pane, click **Management and Settings**.
3. In the **Auto Scaling Policies** section, click **Add Policy**.
4. In the **Add Policy** panel, configure the following parameters.

Parameter	Description
Policy Name	The name of the policy.

Parameter	Description
Mode	The mode of the policy. Select **Automatic Scaling for Local Resources**.
Engine Type	The type of the database engine. Only PolarDB MySQL is supported.
Specifications	The specifications of the selected database engine. Only General-purpose Cluster is supported.
Average CPU Utilization	The threshold that is used to trigger automatic scale-up. If the average CPU utilization is greater than or equal to the specified value, automatic scale-up is triggered.
Scale-up Observation Window	The sampling duration for scale-up. DAS monitors the CPU utilization of the instance during the sampling duration. When the CPU utilization reaches the value specified in the Average CPU Utilization parameter, a scale-up task is triggered.
CPU Scale-up Step Size	The number of cores to add during each CPU scale-up. The value must be an integer ranging from 2 to 31. For example, if the number of CPU cores of an instance is 4 and the **CPU Scale-up Step Size** value is 2, the number of CPU cores is increased to 6 after a CPU scale-up. If the average CPU utilization still meets the conditions for triggering automatic scale-up after a scale-up is complete, another scale-up is performed until the maximum number of CPU cores is reached. **Note** The maximum number of CPU cores for a single database instance is 32, which is calculated based on the following formula: 4 × Original number of CPU cores. For example, if the original number of CPU cores of an instance is 4, the CPU of the instance can be scaled up to a maximum of 16 cores.
Scale-down Observation Window	The sampling duration for scale-down. DAS monitors the CPU utilization of the instance during the sampling duration. When the CPU utilization is below 30% for more than 99% of the sampling duration, a scale-down task is triggered. The number of CPU cores is decremented by the CPU Scale-up Step Size value during each scale-down until the number of CPU cores return to the original value.



5. Click **Next Step**.
6. On the Auto Scaling Policies page, find the policy you created and click **Apply** in the **Actions** column.
7. In the **Apply Policies** dialog box, select the database instances to which you want to apply the policy, and click >.



8. Click **Confirm**. Then, the policy is applied to the selected database instances.

View the results of automatic scaling for local resources

1. In the left-side navigation pane of the DAS console, click **Instance Monitoring**.
2. On the page that appears, find the database instance for which you want to enable the auto scaling feature and click the instance ID. The instance details page appears.
3. In the left-side navigation pane, click **Autonomy Center**.
4. On the **Autonomy Center** page, select a time range and filter the auto scaling events that occurred in the selected time range.
5. Click **Details** in the **Auto-Scaling Events** section to view the details of auto scaling events.

8.4. Perform a temporary cluster upgrade

For a cluster you subscribed to, you can temporarily upgrade the specifications of the cluster to meet the business peak requirements in the specified validity period of the temporary upgrade.

Prerequisites

- The cluster is a subscription cluster.
- The cluster does not have pending renewal, upgrade, downgrade, or temporary upgrade orders.
- The cluster is of the Edition or . and are not supported. For more information about cluster editions, see Editions.

Background information

You can temporarily upgrade the specifications of a PolarDB cluster or a single node in a PolarDB cluster. When the specified rollback time is reached, the cluster is automatically rolled back to the original specifications.

> **Note** PolarDB clusters do not support temporary downgrades. For more information about downgrades, see Manually upgrade or downgrade a PolarDB cluster.

Precautions

- When PolarDB temporarily upgrades a cluster or rolls back a cluster to the original specifications, transient connections to the cluster may occur. Make sure that your application is configured with the automatic reconnection mechanism.
- The rollback time must be at least one day before the expiration date of the cluster. For example, if a cluster expires on January 10, the restoration time must be on January 9 at latest.
- If a cluster is temporarily upgraded, you cannot manually upgrade or downgrade the cluster, add nodes to the cluster, or remove nodes from the cluster. In addition, the cluster does not support automatic scaling. For more information about how to manually upgrade or downgrade a cluster and how to add or move nodes, see Manually upgrade or downgrade a PolarDB cluster and Add or remove read-only nodes.
- The earliest rollback time that you can specify is 1 hour after the cluster is upgraded. We recommend that you set the validity period of a temporary upgrade to no more than 14 days.
- After you perform a temporary upgrade, if the cluster still does not meet your business requirements or you want to extend validity period of the temporary upgrade, you can perform another temporary upgrade before the first temporary upgrade is rolled back. The rollback time of the second temporary upgrade must not be earlier than that of the first temporary upgrade.

Select different specifications for primary node and read-only nodes

For a cluster, you can perform separate temporary upgrades for the primary node and read-only nodes. This means that you can select different specifications for the primary node and read-only nodes.

Limits

- In a cluster, ensure that at least one read-only node has the same specifications as the primary node.
- We recommend that you select close specifications for the primary node and read-only nodes:
  - The memory of a read-only node cannot be less than half of the memory of the primary node.

- The following table lists the correspondence of CPU specifications for the primary node and read-only nodes.

CPU cores for primary node	Minimum CPU cores for read-only node
2	2
4	4
8	4
16	8
32	16
64	32
88	32

- Read-only nodes must be of the same specifications as the primary node if hot standby nodes are configured.
- We recommend that read-only column store nodes use higher specifications than the primary node if read-only column store nodes are added.

> **Note** Edition clusters are not subject to the preceding limits.

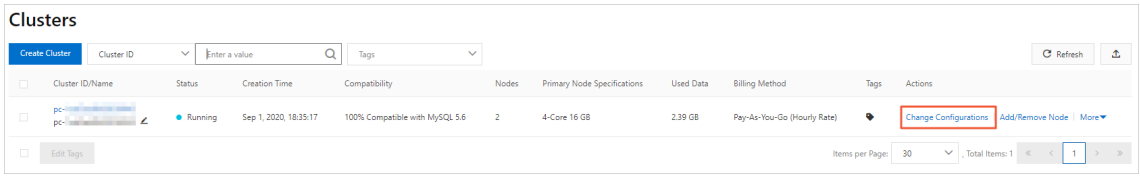
Billing

A temporary upgrade fee is 1.5 times the price difference between the original and new specifications. For example, the validity period of a temporary upgrade is N days. You can calculate the temporary upgrade fee by using the following formula:

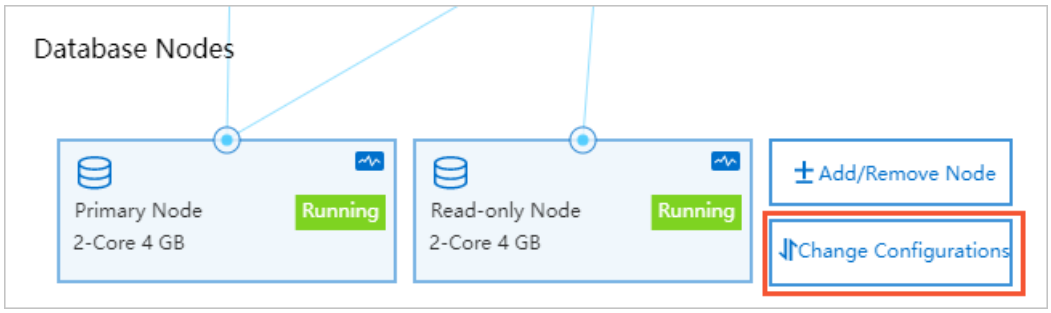
Temporary upgrade fee = (Monthly subscription fee for the new specifications - Monthly subscription fee for the original specifications)/30 × 1.5 × N

Procedure

1.
2.
3. On the **Clusters** page, find the cluster for which you want to perform a temporary upgrade.
4. Open the **Change Configurations (Subscription)** dialog box by using one of the following two methods:
   - Find the cluster that you want to upgrade and click **Change Configurations** in the **Actions** column.



- a. Find the cluster that you want to upgrade and click the cluster ID to navigate to the **Overview** page.
- b. In the **Database Nodes** section, click **Change Configurations**.



5. On the **Change Configurations (Subscription)** page, select **Temporary Upgrade** and click **OK**.

> **Note** **Temporary Upgrade** is available only for subscription clusters.

6. On the page that appears, specify the following parameters.

Parameter	Description
Node	The new node specifications that you want to use. **Note** Make sure that one read-only node has the same specifications as the primary node and you can determine the specifications of other read-only nodes.
Restore Time	The time when the temporary upgrade expires. When the time is reached, the cluster is restored to the original specifications. **Note** - After you perform a temporary upgrade, if the cluster still does not meet your business requirements or you want to extend the validity period of the temporary upgrade, you can perform another temporary upgrade before the first temporary upgrade is rolled back. **The rollback time of the second temporary upgrade** must not be earlier than that of the first temporary upgrade. - The minimum validity period for a temporary upgrade is one hour. We recommend that you set the validity period to up to 14 days. It is because the restoration time cannot be changed after it is specified. - The rollback time must be at least one day before the expiration date of the cluster.

7. Read and accept the terms of service and then click **Buy Now**.
8. On the **Purchase** page, confirm the order information and click **Purchase**.

8.5. Add or remove read-only nodes

This topic describes how to manually add or remove read-only nodes after you create a cluster.

Prerequisites

- The edition of the cluster is , , or . and are not supported. For more information about cluster editions, see Editions.
- No cluster specification change tasks are being performed for the cluster.

Separately change the specifications of the primary node and read-only nodes

You can add read-only nodes based on your business requirements. This means that the specifications of read-only nodes do not have to be consistent with those of the primary node.

Limits

- A PolarDB cluster can contain up to 15 read-only nodes. To ensure high availability, each cluster must have at least one read-only node.
- The specifications of the newly added read-only node should not differ too greatly from those of the primary node.
  - The memory of the newly added read-only node must be at least half that of the primary node.
  - The following table describes the mapping between the number of vCPUs of the newly added read-only node and the number of vCPUs of the primary node:

Number of vCPUs of the primary node	Minimum number of vCPUs of read-only nodes
2	2
4	4
8	4
16	8
32	16
64	32
88	32

  - In a PolarDB cluster, at least one read-only node should have the same specifications as the primary node.
  - The specifications of hot standby nodes must be the same as those of the primary node.
  - We recommend that you specify higher specifications for read-only column store nodes than the primary node. For more information, see Add a read-only column store node.

> **Note** Edition is not subject to the preceding limits.

Billing rules

You are charged for newly added nodes based on the following billing rules:

- If you add a node to a **subscription** cluster, the node is billed on a **subscription** basis.
- If you add a node to a **pay-as-you-go** cluster, the node is billed on a **pay-as-you-go** basis.

> **Note**
> - You can release read-only nodes that use the subscription and pay-as-you-go methods as needed. After you release a node, the system refunds fees for the remaining subscription period or stops billing. For more information, see Configuration change fees.
> - The nodes that you add are billed based on the node specifications. For more information, see Billable items. The storage fee varies based on the usage of the storage space, regardless of the number of nodes.

Impacts of node quantity on cluster performance

For more information, see OLTP performance test tools and methods.

Precautions

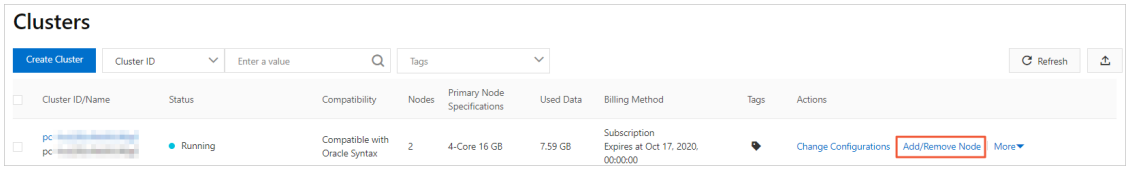
- You can add multiple read-only nodes at a time to Cluster Edition or clusters. Each cluster can contain at most 15 read-only nodes.
- Only clusters support the concurrent removal of multiple read-only nodes. However, you must keep at least one read-only node in the cluster to ensure high availability.
- It takes about 5 minutes to add a read-only node. The time consumption varies based on multiple factors, such as the number of nodes that are added, the numbers of tables and databases, and the database loads. When PolarDB adds nodes to a cluster, the databases in the cluster are not affected.
- When PolarDB removes a read-only node from a cluster, connections to the node are closed. The connections to other nodes in the cluster are not affected. We recommend that you remove nodes during off-peak hours and make sure that your applications can automatically reconnect to the cluster. If you remove a node from a cluster and your application is connected to the cluster endpoint, the removed node is transparent to the application. Therefore, you do not need to modify the configurations of the application.

Add a read-only node

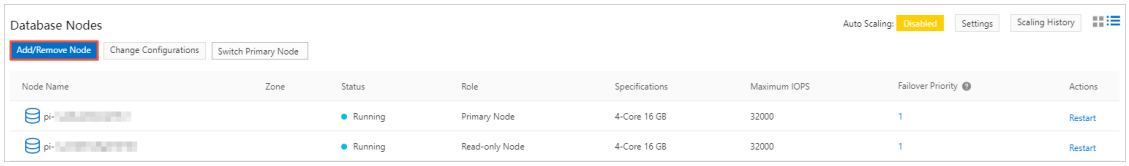
> **Note** After you add a read-only node, a read session is established to forward read requests to the read-only node. However, existing read sessions are not redirected to the read-only node. To resolve this issue, you can close and re-create these sessions by restarting your applications. Then, sessions are established to the newly added node.

1.
2.
3. Open the **Add/Remove Node** dialog box by using one of the following methods:
   - Open the **Add/Remove Node** dialog box on the **Clusters** page.

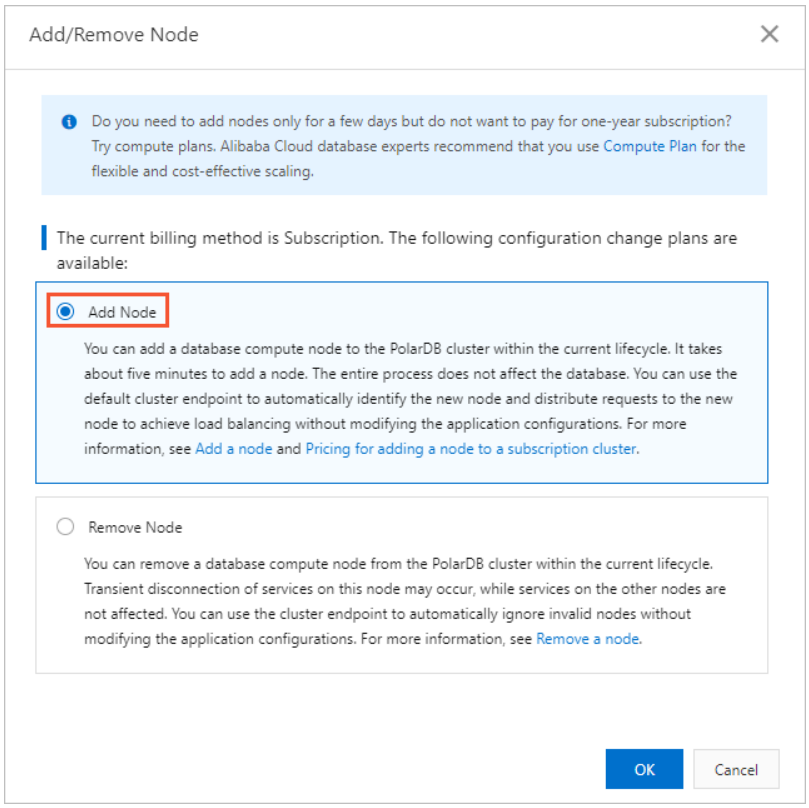
     Find the cluster that you want to manage and click **Add/Remove Node** in the **Actions** column.



- Open the **Add/Remove Node** dialog box on the **Overview** page of the cluster.
  a. Find the cluster that you want to manage and click the cluster ID. The **Overview** page appears.
  b. In the **Database Nodes** section, click the icon to change the display mode.
  c. Click **Add/Remove Node**.



4. Select **Add Node** and click **OK**.



5. Click **+ Add a read-only node**, specify **Switching Time**, read and accept the terms of service, and then click **Buy Now**.

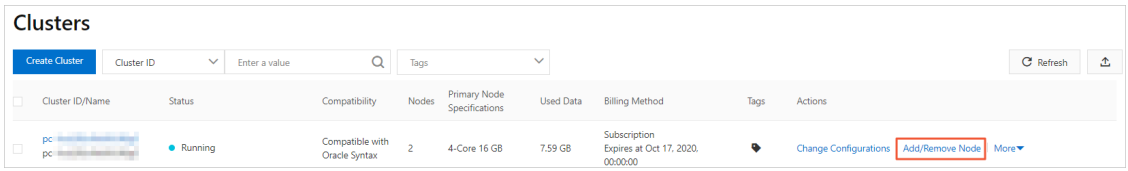
> **Note**
>
> - If you want to add multiple nodes to your cluster at a time, click **+ Add a read-only node** again to add more nodes.
> - You can set **Switching Time** to **Switch Now** or **Switch At**. If you select **Switch At**, you can specify a point in time within the next 24 hours. The nodes will be added within 30 minutes after the specified point in time. You can view the details about the scheduled task on the **Scheduled Tasks** page, or cancel the task. For more information, see View or cancel a scheduled task.

6. On the **Purchase** page, confirm the order information and click **Purchase**.

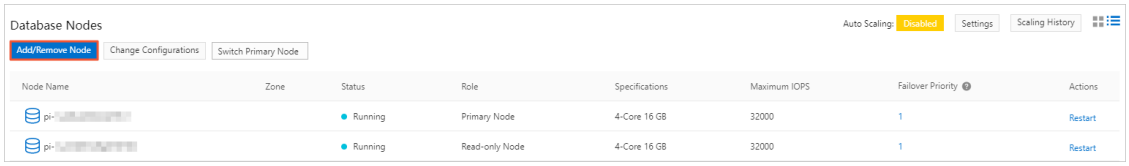
Remove a read-only node

1.
2.
3. Open the **Add/Remove Node** dialog box by using one of the following methods:
   - Open the **Add/Remove Node** dialog box on the **Clusters** page.

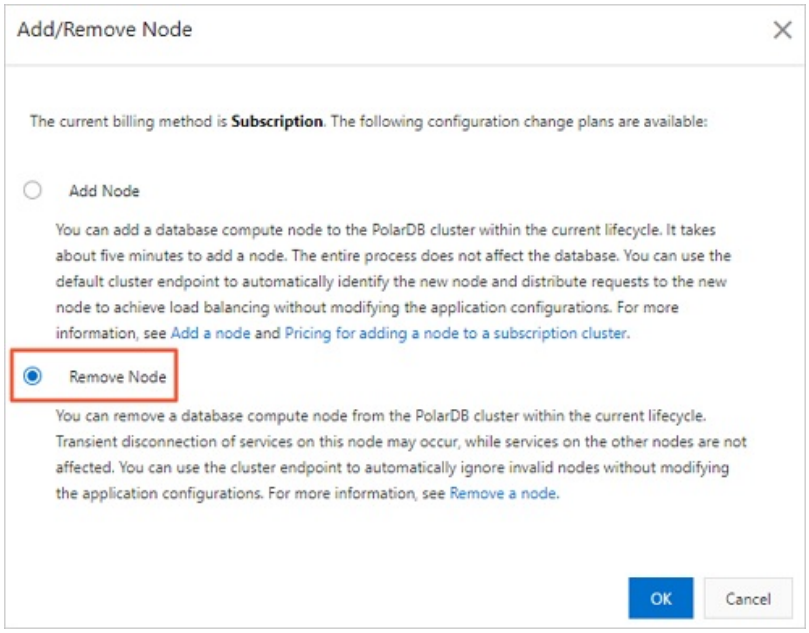
     Find the cluster that you want to manage and click **Add/Remove Node** in the **Actions** column.

   - Open the **Add/Remove Node** dialog box on the **Overview** page of the cluster.
     a. Find the cluster that you want to manage and click the cluster ID. The **Overview** page appears.
     b. In the **Database Nodes** section, click the icon to change the display mode.
     c. Click **Add/Remove Node**.

4. Select **Remove Node** and click **OK**.



5. Click the

   icon to the left of a node name to remove the node.

   > **Note** Only clusters support the concurrent removal of multiple read-only nodes. However, you must keep at least one read-only node in the cluster to ensure high availability.

6. Read and accept the terms of service and click **Buy Now**.

   > **Note** After a node is removed, the system refunds fees for the remaining subscription period or stops billing. For more information, see Configuration change fees.

Related API operations

API	Description
CreateDBNodes	Adds read-only nodes to a cluster.
ModifyDBNodesClass	Changes the specifications of a single node in a cluster independently.
ModifyDBNodeClass	Changes the node specifications of a cluster.
RestartDBNode	Restarts a specified node in a cluster.
DeleteDBNodes	Removes a read-only node from a cluster.

8.6. Upgrade an Archive Database Standalone Edition cluster to an Archive Database Cluster Edition cluster

supports and . This topic describes how to upgrade to .

Prerequisites

Before you use this feature, to contact technical support for data processing.

Context

clusters are unavailable for purchase. However, existing clusters remain available for use. You can upgrade existing clusters to clusters. For more information about , see Overview.

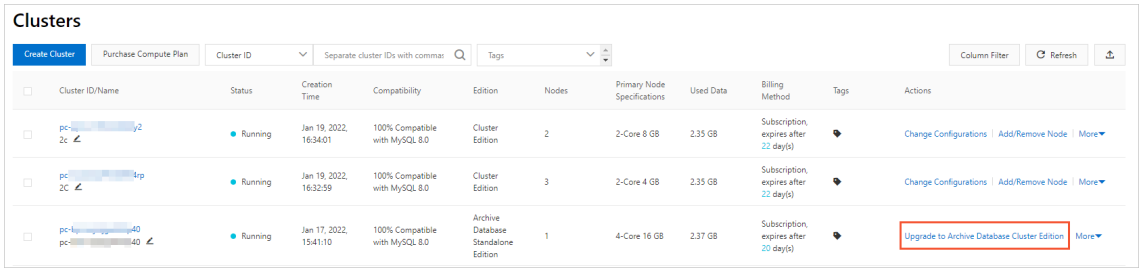
Precautions

- After you upgrade to , the system automatically restarts nodes. In most cases, the restart takes 3 to 5 minutes based on the redo logs that accumulate before the upgrade. In extreme cases, the restart may take up to 30 minutes. During this period, your business is unavailable. Exercise caution when you perform this upgrade. We recommend that you perform this upgrade during off-peak hours.
- After the upgrade is successful, all nodes in the cluster continue to use the primary endpoint. To use a cluster endpoint, you must configure the cluster endpoint on your own.
- After you upgrade an Archive Database Standalone Edition to an cluster, the Archive Database Cluster Edition cluster cannot be reverted to the cluster. If the reversion is required, we recommend that you add nodes. After you add nodes, existing nodes are not restarted. The operations of adding and removing nodes and precautions are consistent with those of Cluster Edition. For more information, see Add or remove read-only nodes.

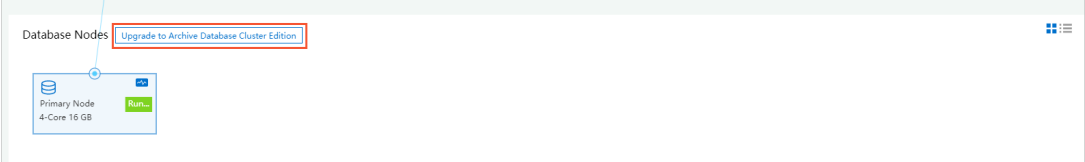
Procedure

1.
2.
3. Open the **Change Configurations** dialog box by using one of the following methods:
   - Method 1

     On the **Clusters** page, find the cluster that you want to manage and click **Upgrade to Archive Database Cluster Edition** in the **Actions** column.



- Method 2
  a. On the **Clusters** page, click the ID of the cluster.
  b. In the **Database Nodes** section of the **Overview** page, click **Upgrade to Archive Database Cluster Edition**.



4. Click **+ Add a read-only node**, specify **Switching Time**, read and accept the terms of service, and then click **Buy Now**.

> **Note**
> - If you want to add multiple nodes to your cluster at a time, click **+ Add a read-only node** again to add more nodes.
> - You can set **Switching Time** to **Switch Now** or **Switch At**. If you select **Switch At**, you can specify a point in time within the following 24 hours. The nodes will be added within 30 minutes after the specified point in time. You can view the details about the scheduled task on the **Scheduled Tasks** page, or cancel the task. For more information, see View or cancel a scheduled task.

5. On the **Purchase** page, confirm the order information and click **Purchase**.

9.Backup and Restoration

9.1. Overview

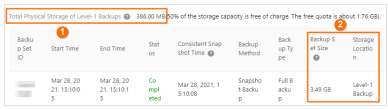
Data is a core asset for enterprises. As enterprise business grows, its data increases exponentially. This requires business applications to be able to process data online and in real time. It becomes more challenging for database O&M personnel to protect the core data of their enterprises, because various factors, such as accidental data deletion, system vulnerabilities, ransomware, hardware failures, and natural disasters, may cause data losses. Therefore, data backup and restoration are important features of databases.

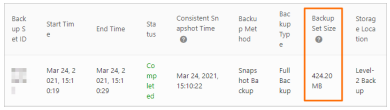
supports data backup and redo log backup. Backing up data is a process of creating a backup set (snapshot) of all data on a cluster at a certain point in time. Backing up redo logs is a process of recording the new data after a backup set is created. You can restore your cluster or a specific instance or table in your cluster to any point in time by using a full data backup set and the redo logs generated after the backup set is created.

Data backup

Data backups are divided into level-1 backups and level-2 backups by storage location.

The location where backup sets are stored	Default configuration	Retention period	Benefit	How to view the size of a backup set

The location where backup sets are stored	Default configuration	Retention period	Benefit	How to view the size of a backup set
Level-1 backup	Enabled	3~14 days	• Level-1 backups are created based on Redirect-on-Write (ROW) snapshots. These snapshots are stored in the distributed file system of . The system does not replicate data when it saves a data block to a snapshot. When a data block is modified, the system saves one of the previous versions of the data block to a snapshot and creates a new data block that is redirected by the original data block. Therefore, you can create backups within a few seconds regardless of the size of your database storage. • The backup and restoration features of clusters use multi-threading parallel processing and other innovative technologies. This allows you to restore data from a backup set (snapshot) to a new cluster within 10 minutes. **Note** By default, the level-1 backup feature is enabled, and you cannot disable this feature.	The following figure shows the total physical storage of level-1 backups.  **Note** The **total size of level-1 backups** of a cluster is the sum of the dedicated physical storage occupied by all level-1 backups, as shown in part ①. It is not the sum of the logical data sizes of all level-1 backups, as shown in part ②. The data of the cluster and multiple level-1 backups (snapshots) can be stored in the same physical data block that is billed only once. For more information, see FAQ.

The location where backup sets are stored	Default configuration	Retention period	Benefit	How to view the size of a backup set
Level-2 backup	Disabled	• 30 to 7,300 days • Enable the **Retained Before Cluster Is Deleted** feature to save level-2 backups permanently.	• Level-2 backups are level-1 backups that are compressed and then stored in on-premises storage. Level-2 backups are slower to restore than level-1 backups. However, level-2 backups are more cost-effective than level-1 backups. • If you enable this feature, expired level-1 backups are transferred to on-premises storage and stored as level-2 backups. The backups are transferred at a rate of approximately 150 MB/s.	The following figure shows the total size of level-2 backups. The total size of level-2 backups is the sum of the data sizes of all level-2 backups. 

The location where backup sets are stored	Default configuration	Retention period	Note If a level-1 backup expires before the previous one is transferred to a level-2 backup, the level-1 backup is deleted and is not transferred to a level-2 backup. For example, a cluster creates level-1 backups at 01:00 every day and retains the backups for 24 hours. The cluster creates Level-1 Backup A at 01:00 on January 1 and creates Level-1 Backup B at 01:00 on January 2. Level-1 Backup A expires at 01:00 on January 2 and starts to be transferred to a level-2 backup. However, Level-1 Backup A is large in amount of data and the transfer task is not completed by 01:00 on January 3. In this case, Level-1 Backup B is deleted after it expires at 01:00 on January 3 and is not transferred to a level-2 backup.	Benefit	How to view the size of a backup set

Redo log backup

- Benefit

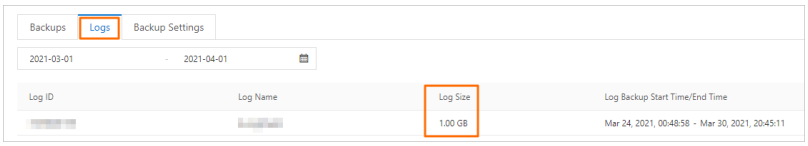
  The log backup feature allows you to create backups by uploading real-time redo logs to Object Storage Service (OSS) in parallel. The feature is enabled by default, and log backups are retained for 3 to 7,300 days. You can save the backups permanently by enabling the **Retained Before Cluster Is Deleted** feature.

  Note By default, log backup is enabled, and you cannot disable this feature.

  Log backups help consistent point-in-time recovery. Based on a full backup set (snapshot) and the redo logs generated after the backup set is created, you can perform point-in-time recovery (PITR) for a cluster. Log backups can prevent data loss caused by user errors and ensure the security of data that is generated within a period of time. If you perform PITR, you must consider the amount of time that is required to query redo logs. Redo logs are queried at a rate of 1 GB every 20 seconds to 70 seconds. The total restoration duration is the sum of the time required to restore backup sets and the time required to query redo logs.

- View the size of a backup set

  The following figure shows that the total size of log backups is the sum of the size of each log backup file.



9.2. Pricing

This topic describes the pricing rules of the backup and restoration features of .

Prices

allows you to use the backup and restoration features free of charge. However, backup files consume storage space. Storage types include level-1 backup, level-2 backup, and log backup based on backup types. Prices vary with different storage types.

- provides free storage quotas for both level-1 backups and log backups. If your usage exceeds the quotas, you are charged by for the amount of additional storage space consumed by the backups and the length of time for which the backups are retained.
- Level-2 backups are paid services. You are charged by for the amount of storage space consumed by the backups and the length of time for which the backups are retained.

The following table describes the prices.

Prices (Unit: USD/GB/hour)

Region	Level-1 backup	Level-2 backup	Log backup
Chinese mainland	0.000464	0.0000325	0.0000325
China (Hong Kong) and regions outside China	0.000650	0.0000455	0.0000455

Note After the free quota is consumed, you are charged for additional storage space on a pay-as-you-go basis. To reduce the costs of level-1 backups, we recommend that you use storage plans, which are cost-effective. For more information, see Billing rules of backup storage that exceeds the free quota.

If you modify data after the snapshots of your cluster are created, the amount of snapshot backups increases. In this case, you are charged for the additional backups. If data is modified during defragmentation, the number of snapshot backups increases.

For example, if your database has 100 GB of data and you modify 10 GB of the data after a snapshot is created:

- You are charged for 100 GB of data storage and 10 GB of snapshot storage.
- If you choose to retain the snapshots when you delete your database, you are charged for 100 GB of snapshot storage.

Reduce the charges for backup storage usage

does not support the download of backup files to your PC. To reduce backup fees, you can use one of the following methods:

- Select an appropriate backup frequency: In the **Backup Policy Settings** dialog box, set **Backup Frequency** to an appropriate value.
- Shorten the retention period of level-1 backups: In the **Backup Policy Settings** dialog box, set **Level-1 Backup** in the **Data Backup Retention Period** section.
- Purchase storage plans to offset level-1 backup fees: You can use storage plans to offset the fees generated by level-1 backups, which are cost-effective. For more information, see Billing rules of backup storage that exceeds the free quota.

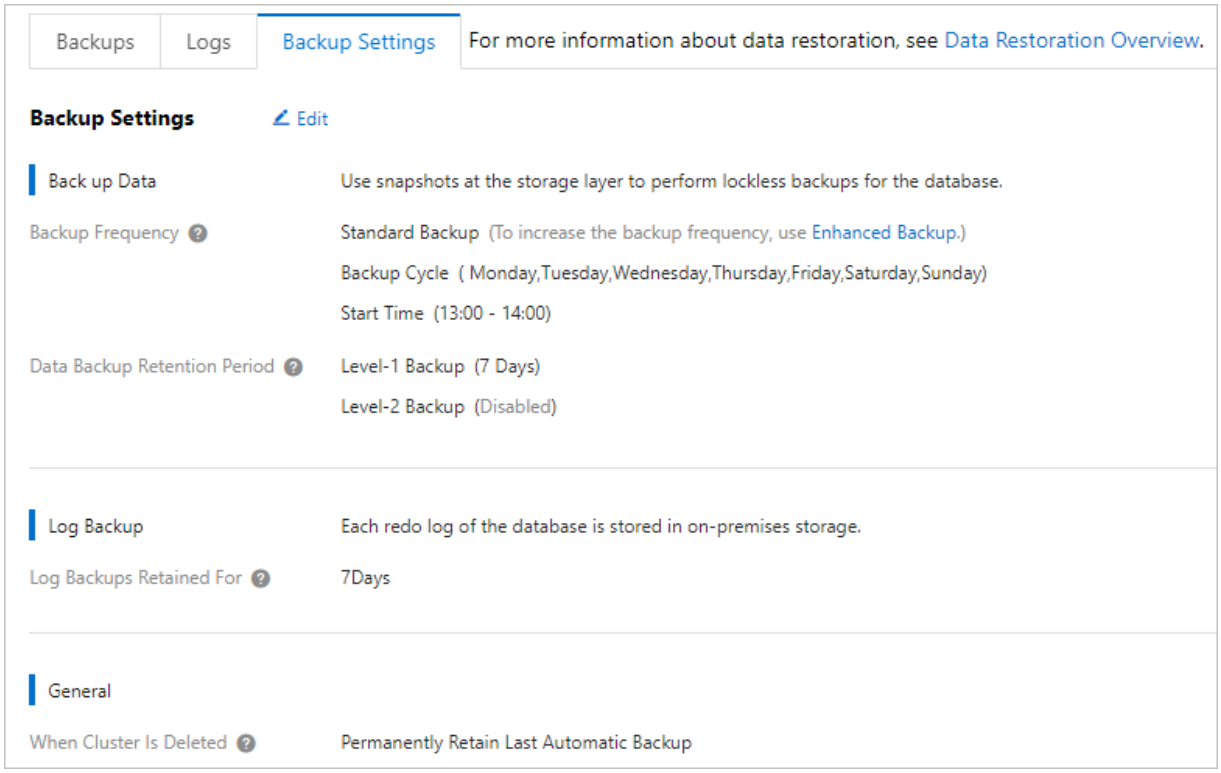
9.3. Backup methods

9.3.1. Backup settings

supports data backup and redo log backup. When you back up data, you create a backup set of all data on a cluster at a specific point in time. A backup set is also known as a snapshot. When you back up redo logs, you record the incremental data that is generated after a backup set is created. You can configure policies for data backup and redo log backup. For example, you can specify the frequency of automatic data backups, the retention period of data backup files, the storage location, and the retention period of log backup files.

Navigate to the details page of the cluster for which you want to configure backup settings. In the left-side navigation pane, choose **Settings and Management > Backup and Restore**. Click the **Backup Policy Settings** tab.

Click **Edit**. In the dialog box that appears, configure parameters in the **Back up Data**, **Log Backup**, and **General** sections.



Parameters in the Back up Data section

To configure a data backup policy, specify the frequency of automatic backups and the storage location and retention period of the backup files generated by automatic backups and manual backups.

Parameter	Description
Backup Frequency	The frequency of automatic backups. You can select **Standard Backup (at specified intervals)** or **High-frequency Backup**. • **Standard Backup (at specified intervals)**: If you select this option, you must specify the frequency of automatic backups and the time when automatic backups start. **Note** To prevent data loss, perform automatic backup at least two times a week. • **High-frequency Backup**: If you select this option, you must specify the backup frequency. You can select **Last 24 Hours, Every 2 Hours**, **Last 24 Hours, Every 3 Hours**, or **Last 24 Hours, Every 4 Hours**.

Parameter	Description
Data Backup Retention Period	The storage location and retention period of the data backup files generated by automatic backups and manual backups. You can specify **Level-1 Backup** or **Level-2 Backup** as the storage location. For more information, see Data backup. • **Level-1 Backup**: If you select this option, you must specify the retention period for level-1 backups. Note ◦ By default, level-1 backup is enabled. The default retention period of level-1 backups is 7 days. ◦ A backup can be retained for 3 to 14 days. • **Level-2 Backup**: You can enable or disable level-2 backup. Note ◦ By default, level-2 backup is disabled. If you enable the feature, storage fees are incurred. You can delete backup files to reduce costs. For more information about the pricing of level-2 backup, see Billing rules of backup storage that exceeds the free quota. ◦ Level-2 backups can be retained for 30 to 7,300 days. ◦ If you want to permanently retain level-2 backups, select **Retained Before Cluster Is Deleted**. After you select this option, you cannot specify the retention period of level-2 backups.

Parameters in the Log Backup section

When you configure a redo log backup policy, you must specify the retention period of redo logs.

Parameter	Description

Parameter	Description
Log Retention Period (Days)	The retention period of log backup files. **Note** • By default, log backup is enabled. The default retention period of a log backup file is 7 days. • Log backup files can be retained for 3 to 7,300 days. • To permanently retain log backup files, select **Retained Before Cluster Is Deleted**. The Retention Period parameter becomes unavailable if you select this option.

Parameters in the General section

You can configure a backup retention policy that applies when you delete a cluster.

Parameter	Description

Parameter	Description
When Cluster Is Deleted	The backup retention policy that applies when you delete a cluster. • **Permanently Retain All Backups**: retains all backups after you delete a cluster. • **Permanently Retain Last Automatic Backup**: retains the latest backup after you delete a cluster. • **Immediately Delete All Backups**: does not retain backups after you delete a cluster. Note • If you select the **Permanently Retain All Backups** or **Permanently Retain Last Automatic Backup** policy, the system runs an automatic backup task to retain all data when you delete the cluster. • After you delete a cluster, level-1 backups are automatically transferred to level-2 backups. You can go to the **Cluster Recycle** page to view all backups. For more information, see Restore a released cluster.

Related API operations

Operation	Description
DescribeBackupPolicy	Queries the backup policy of a specified cluster.
ModifyBackupPolicy	Modifies the backup policy of a specified cluster.

9.3.2. Backup method 1: Automatic backup

By default, automatic backup is enabled. automatically backs up data based on the specified backup policy. This way, data security is ensured by periodic and scheduled backups. When a cluster is created, automatically backs up data once a day. You can configure parameters such as the frequency of automatic backup and the retention period of backup files in the console based on your business requirements.

Note

- The backup files that are automatically created cannot be deleted. You can configure the retention

period of automatic backup files in the **Data Backup Retention Period** section in the Backup Settings dialog box.

- If you modify data after snapshots of your cluster are created, the amount of snapshot backups increases. As a result, the backup cost increases. If data is modified during defragmentation, the amount of snapshot backups increases.

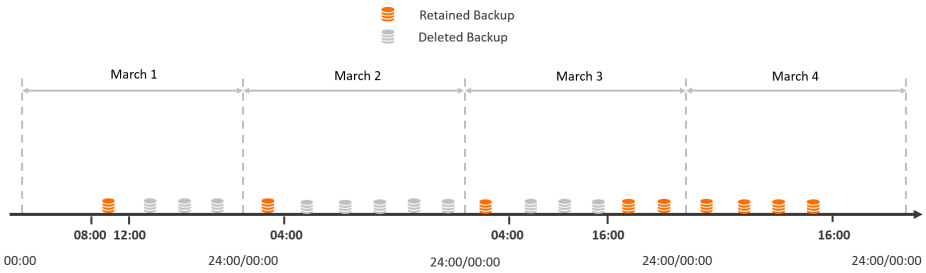
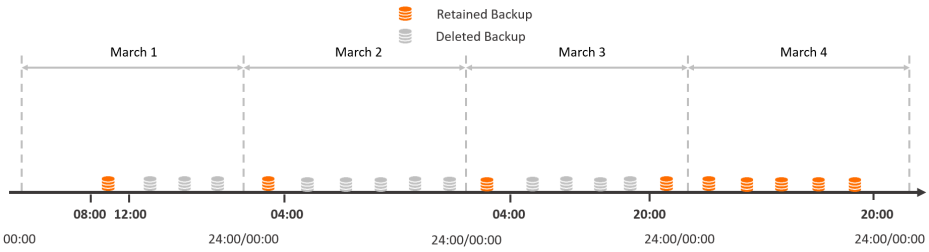
  For example, if your database has 100 GB of data and you modify 10 GB of the data after a snapshot is created,

  - you are charged for 100 GB of data storage and 10 GB of snapshot storage.
  - If you choose to retain the snapshots when you delete your database, you are charged for 100 GB of snapshot storage.

Standard backup and enhanced backup

To enable backup at different frequencies, supports two types of automatic backup: standard backup and enhanced backup.

Backup Frequency	Description
Standard Backup (at specified intervals)	In this mode, the data is automatically backed up only once a day. You can specify which days of the week to back up data and the time period during which data is backed up. **Note** To prevent data loss, perform automatic backup at least two times a week.

Backup Frequency	Description
High-frequency Backup	supports enhanced backup. This feature increases backup frequency to speed up data restoration. In this mode, the system automatically backs up data multiple times in a day. You can configure the system to perform an enhanced backup once every two, three, or four hours based on your business requirements. After you enable enhanced backup, all backups are retained for 24 hours. When the retention period expires, the backups are automatically deleted. The system permanently retains the first backup that is created after 00:00 each day. For example, if you specify a backup frequency of **every 4 hours** at 08:00 on March 1, the system automatically creates the first backup before 12:00 on March 1. Then, the system continues to create a backup every four hours. If the current time is 16:00 on March 4, the system retains the following backups: • The backups created over the past 24 hours (from 16:00 on March 3 to 16:00 on March 4). • The backups created from 00:00 to 04:00 on March 3. • The backups created from 00:00 to 04:00 on March 2. • The backups created from 08:00 to 12:00 on March 1.  Then, after four hours (at 20:00 on March 4), the system retains the following backups: • The backups created within the last 24 hours (from 20:00 on March 3 to 20:00 on March 4). • The backups created from 00:00 to 04:00 on March 3. • The backups created from 00:00 to 04:00 on March 2. • The backups created from 08:00 to 12:00 on March 1. 

Configure an automatic backup policy

You can select Standard Backup or Enhanced Backup based on your business requirements. You can specify the storage location and retention period of data backup files generated by automatic backups.

1. Navigate to the details page of the cluster for which you want to configure backup settings. In the left-side navigation pane, choose **Settings and Management > Backup and Restore**.
2. Click the **Backup Policy Settings** tab.
3. Click **Edit** on the right of **Backup Policy Settings**. In the dialog box that appears, configure the following parameters.

Parameter	Description
Backup Frequency	The frequency of automatic backups. You can select **Standard Backup (at specified intervals)** or **High-frequency Backup**. ○ **Standard Backup (at specified intervals)**: If you select this option, you must specify the frequency of automatic backups and the time when automatic backups start. **Note** To prevent data loss, perform automatic backup at least two times a week. ○ **High-frequency Backup**: If you select this option, you must specify the backup frequency. You can select **Last 24 Hours, Every 2 Hours**, **Last 24 Hours, Every 3 Hours**, or **Last 24 Hours, Every 4 Hours**.

Parameter	Description
Data Backup Retention Period	The storage location and retention period of the data backup files generated by automatic backups and manual backups. You can specify **Level-1 Backup** or **Level-2 Backup** as the storage location. For more information, see Data backup. ○ **Level-1 Backup**: If you select this option, you must specify the retention period for level-1 backups. Note ■ By default, level-1 backup is enabled. The default retention period of level-1 backups is 7 days. ■ A backup can be retained for 3 to 14 days. ○ **Level-2 Backup**: You can enable or disable level-2 backup. Note ■ By default, level-2 backup is disabled. If you enable the feature, storage fees are incurred. You can delete backup files to reduce costs. For more information about the pricing of level-2 backup, see Billing rules of backup storage that exceeds the free quota. ■ Level-2 backups can be retained for 30 to 7,300 days. ■ If you want to permanently retain level-2 backups, select **Retained Before Cluster Is Deleted**. After you select this option, you cannot specify the retention period of level-2 backups.

Note For information about other parameters, see Backup settings.

4. After you complete the backup settings, click **OK**.

Related API operations

Operation	Description
CreateBackup	Creates a full backup of a specified cluster.
DescribeBackups	Queries the backup information of a specified cluster.
DeleteBackup	Deletes the backups of a specified cluster.

Operation	Description
DescribeBackupPolicy	Queries the automatic backup policy of a specified cluster.
ModifyBackupPolicy	Modifies the automatic backup policy of a specified cluster.

9.3.3. Backup method 2: Manual backup

Manual backups are backups triggered by you. You can manually back up data at any time based on your business requirements to ensure data reliability. This topic describes how to configure the manual backup settings.

Precautions

- You can manually create up to three backups for a cluster.
- Manual backup files can be deleted.
- If you modify data after snapshots of your cluster are created, the amount of snapshot backups increases. As a result, the backup cost increases. If data is modified during defragmentation, the amount of snapshot backups increases.

  For example, if your database has 100 GB of data and you modify 10 GB of the data after a snapshot is created,

  - you are charged for 100 GB of data storage and 10 GB of snapshot storage.
  - If you choose to retain the snapshots when you delete your database, you are charged for 100 GB of snapshot storage.

Configure manual backup

1.
2.
3.
4. In the left-side navigation pane, choose **Settings and Management > Backup and Restore**.
5. In the **Data Backups** tab, click **Create Backup**.
6. In the dialog box that appears, click **OK**.

Related API operations

API	Description
CreateBackup	Creates a full backup for a specified cluster.
DescribeBackups	Queries the backup information of a specified cluster.
DeleteBackup	Deletes the backups of a specified cluster.

9.4. Restoration methods

9.4.1. Method 1 for cluster restoration: Restore from a backup set

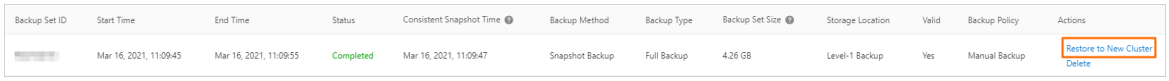
Cluster restoration refers to restoring all data of a cluster to a new cluster. After you verify the new cluster's data accuracy, you can migrate the restored data to the original cluster. Cluster restoration supports two methods: restore from a backup set and restore data to a previous point in time. This topic describes how to restore all historical data of a cluster from a backup set.

Precautions

Only the data and account information of the original cluster can be restored to a new cluster. The parameters of the original cluster cannot be restored to the new cluster.

Procedure

1.
2.
3.
4. In the left-side navigation pane, choose **Settings and Management > Backup and Restore**.
5. Find the backup set that you want to restore and click **Restore to New Cluster** in the **Actions** column.



6. On the **Clone Instance** page, select a **billing method** for the new cluster.
7. Configure the following parameters.

Parameter	Description
Clone Source Type	Select **Backup Set**.
Region	By default, the region of the new cluster is the same as that of the original cluster. This setting cannot be changed.
Clone Source Backup Set	Select the backup set from which you want to restore data. **Note** The **Start Time** of each backup set is displayed. You can determine whether to select the backup set based on this backup time.
Primary Availability Zone	Select the primary zone where the cluster is deployed. **Note** In regions that have two or more zones, automatically replicates data to the secondary zone for disaster recovery.
Network Type	This parameter can only be set to **VPC**.

Parameter	Description
VPC VSwitch	Select a **VPC** and a **VSwitch** for the cluster. We recommend that you use the same VPC and vSwitch that are used for the original cluster. **Note** Make sure that the cluster and the ECS instance you want to connect to the cluster are deployed in the same VPC. Otherwise, the cluster and the ECS instance cannot communicate over the internal network, which results in decreased performance.
Compatibility	By default, the new cluster has the same **compatibility** as that of the original cluster. For example, if the **compatibility** of the original cluster is **MySQL 8.0** (fully compatible with MySQL 8.0), the **compatibility** of the new cluster is also **MySQL 8.0**.
Specification Type	has the following two types of specifications: **General Specification** and **Dedicated Specification**. For more information about the two types of specifications, see Comparison between general-purpose and dedicated compute nodes. **Note** This parameter is available only when the **edition** of the original cluster is .
Node Specification	Select a **node specification**. The maximum storage capacity and performance of clusters vary based on node specifications. For more information, see Specifications of compute nodes. **Note** We recommend that you select a **node specification** that is the same or higher than the node specification of the original cluster. This ensures that the new cluster runs as expected.
Nodes	◦ The default number of nodes of the **edition** is **2**. You do not need to change this parameter value. **Note** By default, a new cluster of contains one primary node and one read-only node. After a cluster is created, you can add nodes to the cluster. A cluster can contain one primary node and up to 15 read-only nodes. For more information about how to add nodes, see Add or remove read-only nodes. ◦ The default number of nodes of the and **editions** is **1**. You do not need to change this parameter value.

Parameter	Description
Storage Cost	You do not need to select the storage capacity when you purchase clusters. You are charged for the storage capacity used on an hourly basis. You can also purchase a storage plan based on your business requirements. For more information about how to purchase a storage plan, see Purchase a storage plan.
Cluster Name	The name of the cluster. The name must meet the following requirements: ○ It cannot start with `http://` or `https://`. ○ It must be 2 to 256 characters in length. If you do not specify this parameter, the system automatically generates a cluster name. You can change the name after the cluster is created.
Purchase Plan	Specify **Purchase Plan** for the cluster. **Note** This parameter is available only when the **Billing Method** parameter is set to **Subscription**.
Number	Select the **number** of clusters you want to purchase.

8. Read and accept the terms of service, and complete the rest of the steps based on the **billing method** of the cluster.
   - **Pay-as-you-go**

     Click **Buy Now**.
   - **Subscription**
     a. Click **Buy Now**.
     b. On the **Purchase** page, confirm the information of the unpaid order and the payment method and click **Purchase**.

Note After you complete the payment, it requires 10 to 15 minutes to create the cluster. Then, you can view the new cluster on the **Clusters** page.

What to do next

1. Log on to the new cluster and verify data accuracy. For more information about how to log on to the cluster, see Connect to a cluster.
2. Migrate data to the original cluster.

   After you verify the data on the new cluster, you can migrate the data from the new cluster back to the original cluster. For more information, see Migrate data between PolarDB for MySQL clusters.

Note Data migration is a process of replicating data from a cluster to another cluster. During data migration, your services in the original cluster are not affected.

Related API operations

API operation	Description
CreateDBCluster	Restores the data of a cluster. **Note** You must set **CreationOption** to **CloneFromPolarDB**.

9.4.2. Method 2 for full restoration: point-in-time restore

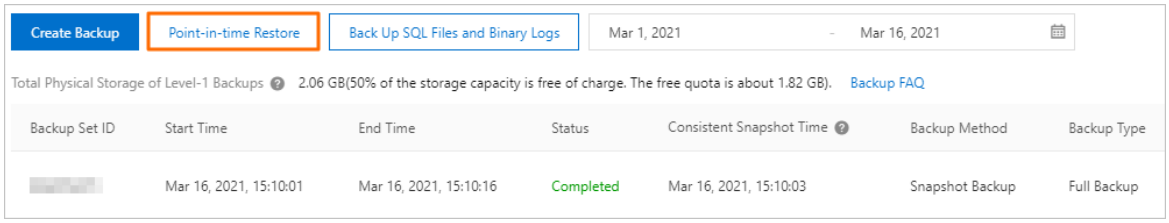
Full restoration is a method of restoring all historical data of a cluster to a new cluster. After you verify the accuracy of the data in the new cluster, you can migrate the restored data to the original cluster. All historical data of a PolarDB cluster can be restored from a backup set or to an earlier point in time. This topic describes how to restore all historical data of a cluster to an earlier point in time.

Note

- Only the data and account information of the original cluster can be restored to a new cluster. The parameters of the original cluster cannot be restored to the new cluster.
- You can restore data to a specified point in time within a specific time range. The time range varies based on the **Log Retention Period (Days)** parameter in the backup settings. The default time range is 7 days.

Procedure

1.
2.
3.
4. In the left-side navigation pane, choose **Settings and Management > Backup and Restore**.
5. On the **Backup and Restore** page, click **Point-in-time Restore**.



6. On the **Clone Instance** page, select a **billing method** for the new cluster.
7. Configure the following parameters.

Parameter	Description
Clone Source Type	Select **Backup Timepoint**.

Parameter	Description
Backup Timepoint	Select the point in time to which you want to restore data. **Note** The time range within which data can be restored to a specified point in time varies based on the **Log Retention Period (Days)** parameter in the backup settings. The default time range is 7 days.
Region	This parameter is automatically set to the region of the original cluster. You do not need to change this value.
Primary Availability Zone	Select the primary zone where you want to deploy the new cluster. **Note** In regions that have two or more zones, automatically replicates data to the secondary zones for disaster recovery.
Network Type	This parameter is automatically set to **VPC**. You do not need to change this value.
VPC	Select a **VPC** and a **vSwitch** for the new cluster. We recommend that you select the same VPC and vSwitch that are connected to the original cluster. **Note** Make sure that the cluster and the ECS instance to be connected to the cluster are deployed in the same VPC. Otherwise, the cluster and the ECS instance cannot communicate over a VPC. As a result, the cluster performance is compromised.
VSwitch	
Compatibility	This parameter is automatically set to the value of **Compatibility** that is specified for the original cluster. You do not need to change this value. For example, if the value of **Compatibility** that is specified for the original cluster is **MySQL 8.0**, **Compatibility** is automatically set to **MySQL 8.0** for the new cluster. The value of MySQL 8.0 specifies that the cluster is fully compatible with MySQL 8.0.
Specification Type	Select **General-purpose** or **Dedicated** for . Valid values: For more information about the comparison between the types of specifications, see Comparison between general-purpose and dedicated compute nodes. **Note** This parameter is available only if the **edition** of the original cluster is .

Parameter	Description
Node Specification	Select a **node specification**. The maximum storage and performance of a cluster vary based on the node specification. For more information, see Specifications of compute nodes. **Note** We recommend that you select a **node specification** that is higher than the node specification of the original cluster. This ensures that the new cluster runs as expected.
Nodes	○ If the **edition** of the original cluster is , the number of nodes is automatically set to **2** for the new cluster. You do not need to change this value. **Note** By default, a new cluster of the contains one primary node and one read-only node. After the new cluster is created, you can add only read-only nodes to the new cluster. Each cluster can contain a maximum of 15 read-only nodes. For more information, see Add or remove read-only nodes. ○ If the **edition** of the original cluster is or , the number of nodes is automatically set to **1** for the new cluster. You do not need to change this value.
Storage Cost	You do not need to specify the required storage when you purchase the cluster. You are charged for storage usage on an hourly basis. In addition, you can purchase storage plans based on your business requirements. For more information, see Purchase a storage plan.
Cluster Name	Enter the name of the new cluster. The name must meet the following requirements: ○ The name cannot start with `http://` or `https://`. ○ The name must be 2 to 256 characters in length. If you do not configure this parameter, the system automatically generates a cluster name. You can change the cluster name after the new cluster is created.
Purchase Plan	Select a **purchase plan** for the new cluster. **Note** This parameter is available only if **Billing Method** is set to **Subscription**.
Number	Select the **number** of clusters that you want to purchase.

8. Read and accept the terms of service, and complete the rest of the steps based on the **billing method** of the cluster.
   - **Pay-as-you-go**

     Click **Buy Now**.

- **Subscription**
  a. Click **Buy Now**.
  b. On the **Purchase** page, confirm the information of the unpaid order and the payment method and click **Purchase**.

> **Note** After you complete the payment, it requires 10 to 15 minutes to create the cluster. Then, you can view the new cluster on the **Clusters** page.

What to do next

1. Log on to the new cluster and verify data accuracy. For more information about how to log on to the cluster, see Connect to a cluster.
2. Migrate data to the original cluster.

   After you verify the data on the new cluster, you can migrate the data from the new cluster back to the original cluster. For more information, see Migrate data between PolarDB for MySQL clusters.

> **Note** Data migration is a process of replicating data from a cluster to another cluster. During data migration, your services in the original cluster are not affected.

Related API operations

API operation	Description
CreateDBCluster	Restores the data of a cluster. **Note** You must set **CreationOption** to **CloneFromPolarDB**.

9.4.3. Method 1 for database and table restoration: Restore data from a backup set

Database and table restoration is a process of restoring only specified databases or tables in a cluster. For example, assume that you are the database administrator of a gaming company, you can use the database and table restoration feature to restore the data of a player or a group of players. You can restore databases and tables by using two methods: restore from a backup set and restore to a point in time. This topic describes how to restore a specified database or table from a backup set.

Limits

- Only Cluster Edition supports database and table restoration, and one of the following clusters must be used:
  - A cluster of 5.6 whose minor version of the kernel is 5.6.1.0.25 or later. For more information, see Version Management.
  - A cluster of 5.7 whose minor version of the kernel is 5.7.1.0.8 or later.

- A cluster of 8.0 whose minor version of the kernel is 8.0.1.1.14 or later.

  > **Note** Database and table restoration is not supported on clusters of 8.0.2.

- Single-node Edition, Archive Database Standalone Edition, and Archive Database Cluster Edition clusters do not support the database and table restoration feature.
- Database and table restoration is not supported on clusters with transparent data encryption (TDE) enabled.
- Clusters in the Global Database Network (GDN) do not support the database and table restoration feature.

Limits

- You can restore databases and tables only from level-1 backup sets. Level-2 backup sets are not supported.
- Only the tables that you specify are restored. Make sure that you select only the tables that you want to restore.

  > **Note** If you are unable to select the tables that you want to restore, we recommend that you restore all data in the current cluster to a new cluster and then migrate the data to the current cluster. For more information, see Method 1 for cluster restoration: Restore from a backup set and Method 2 for full restoration: point-in-time restore.

- If the specified tables are deleted within the time period from the point in time when the most recent backup set is generated to the point in time that you specified, you cannot restore the data by **Backup Timepoint**.
- You can restore at most 100 tables at a time. If you restore a database, all tables in the database are restored.

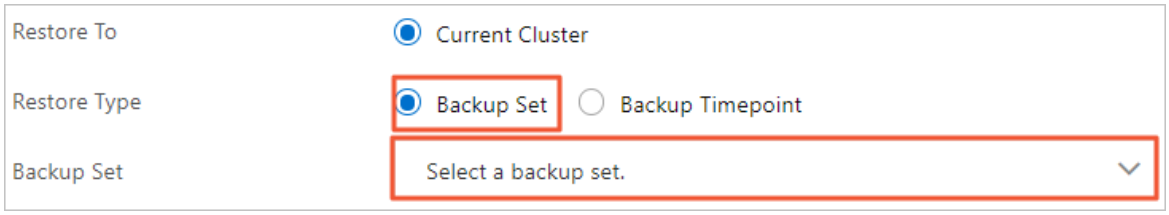
  > **Note** To restore a large number of tables, we recommend that you restore all data in your cluster to a new cluster. For more information, see Method 1 for cluster restoration: Restore from a backup set and Method 2 for full restoration: point-in-time restore.

- The database and table restoration feature is applicable only to clusters that store 50,000 tables or less. If the cluster stores more than 50,000 tables, this feature cannot be used.
- You cannot use the database and table restoration feature to restore triggers. If the table that you restore contains a trigger, the trigger cannot be restored.
- You cannot use the database and table restoration feature to restore foreign keys. If the table that you restore contains a foreign key, the foreign key cannot be restored.

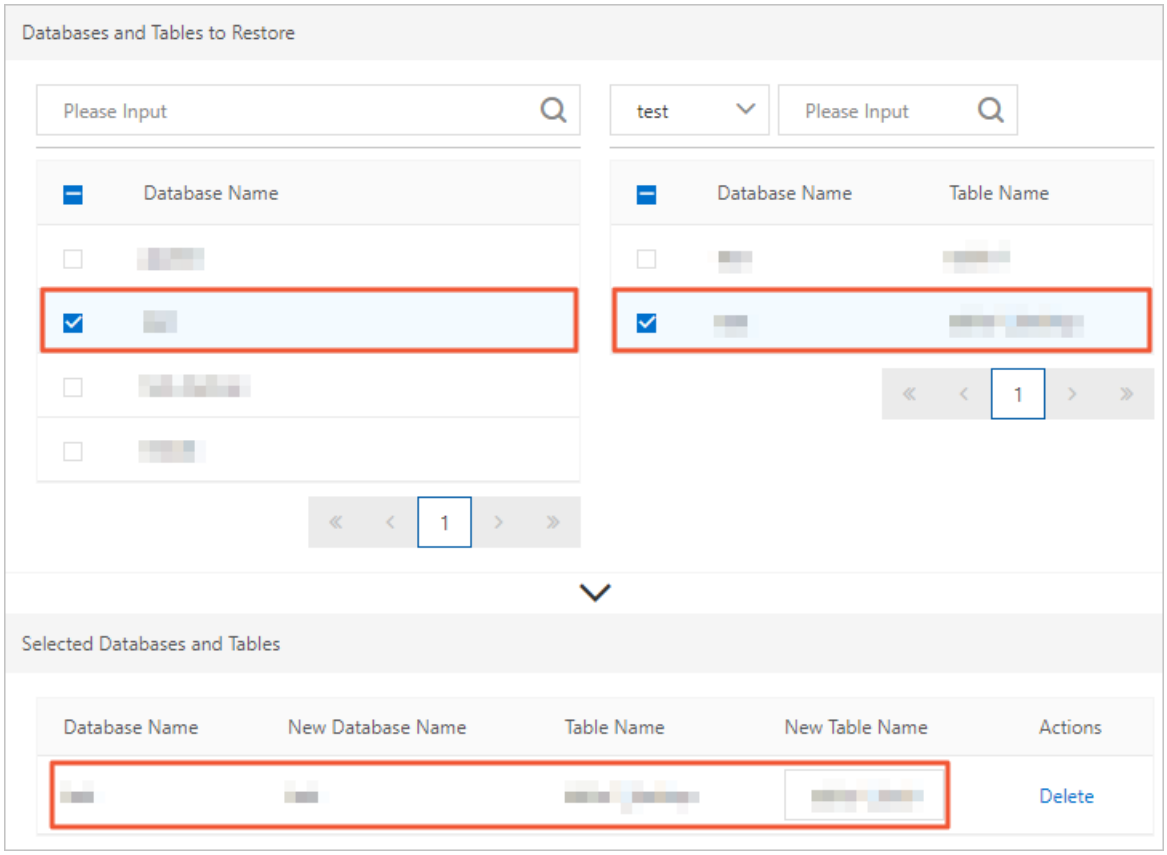
Procedure

1.
2.
3.
4. In the left-side navigation pane, choose **Settings and Management > Backup and Restore**.
5. On the **Backup and Restore** page, click **Restore Databases/Tables**.
6. In the dialog box that appears, select **Backup Set** in the **Restore Type** section and select the

backup set that you want to use from the backup set drop-down list.



7. On the left side of the **Databases and Tables to Restore** section, select the database that you want to restore. On the right side, select the table that you want to restore.



> **Note**
>
> - If you do not select a table after you select a database, all tables in the database are restored.
> - After you select a database or table, the system automatically generates the name of the new database or table by appending the `_backup` suffix to the name of the selected database or table. For example, if the name of the selected table is `test`, the new table is named `test_backup`. You can also change the name of the new database or table.

8. Click **OK**.

Related operations

API	Precautions
DescribeMetaList	Queries the metadata of the database or table that you want to restore.
RestoreTable	Restores the databases or tables of a cluster.

9.4.4. Method 2 for database and table restoration: Restore data to a point in time

Instance and table restoration is a process of restoring a specified instance or table of a cluster. For example, if you are the database administrator of a games company, you can use the database and table restoration feature to restore the data of a player or a group of players. An instance or a table can be restored from a backup set or to a point in time. This topic describes how to restore the data of a database or table to a previous point in time.

Limits

- Only Cluster Edition supports database and table restoration, and one of the following clusters must be used:
  - A cluster of 5.6 whose minor version of the kernel is 5.6.1.0.25 or later. For more information, see Version Management.
  - A cluster of 5.7 whose minor version of the kernel is 5.7.1.0.8 or later.
  - A cluster of 8.0 whose minor version of the kernel is 8.0.1.1.14 or later.

    > **Note** Database and table restoration is not supported on clusters of 8.0.2.

- Single-node Edition, Archive Database Standalone Edition, and Archive Database Cluster Edition clusters do not support the database and table restoration feature.
- Database and table restoration is not supported on clusters with transparent data encryption (TDE) enabled.
- Clusters in the Global Database Network (GDN) do not support the database and table restoration feature.

Precautions

- You can restore only specified tables, so you must confirm that you have selected all the tables that you want to restore.

  > **Note** If you cannot determine which tables to restore, we recommend that you restore all the data of your cluster to a new cluster and then migrate the data back to the original cluster. For more information, see Method 1 for cluster restoration: Restore from a backup set and Method 2 for full restoration: point-in-time restore.

- If the specified tables are deleted during the time period between the generation of the last backup set and the specified time, you cannot restore the data by **Backup Timepoint**.
- You can restore up to 100 tables at a time. If you choose to restore an instance, all tables in the

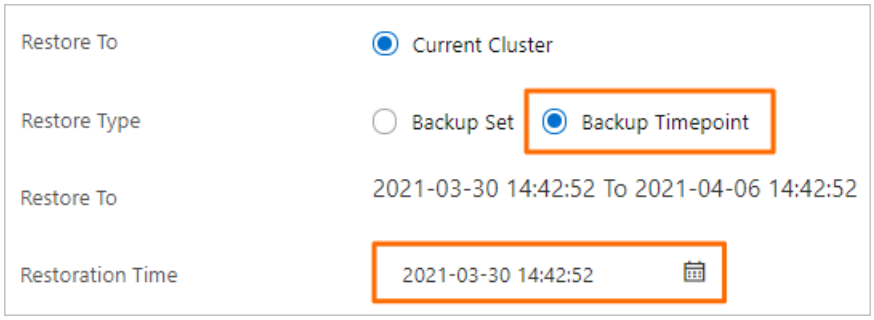
instance can be restored.

> **Note** To restore a large number of tables, we recommend that you restore all data of your cluster to a new cluster. For more information, see Method 1 for cluster restoration: Restore from a backup set and Method 2 for full restoration: point-in-time restore.

- The instance and table restoration feature only applies to clusters that have no more than 50,000 tables.
- The instance and table restoration feature does not restore triggers. If the original table has a trigger, the trigger will not be restored.
- The instance and table restoration feature does not restores foreign keys. If the original table has a foreign key, the foreign key will not be restored.

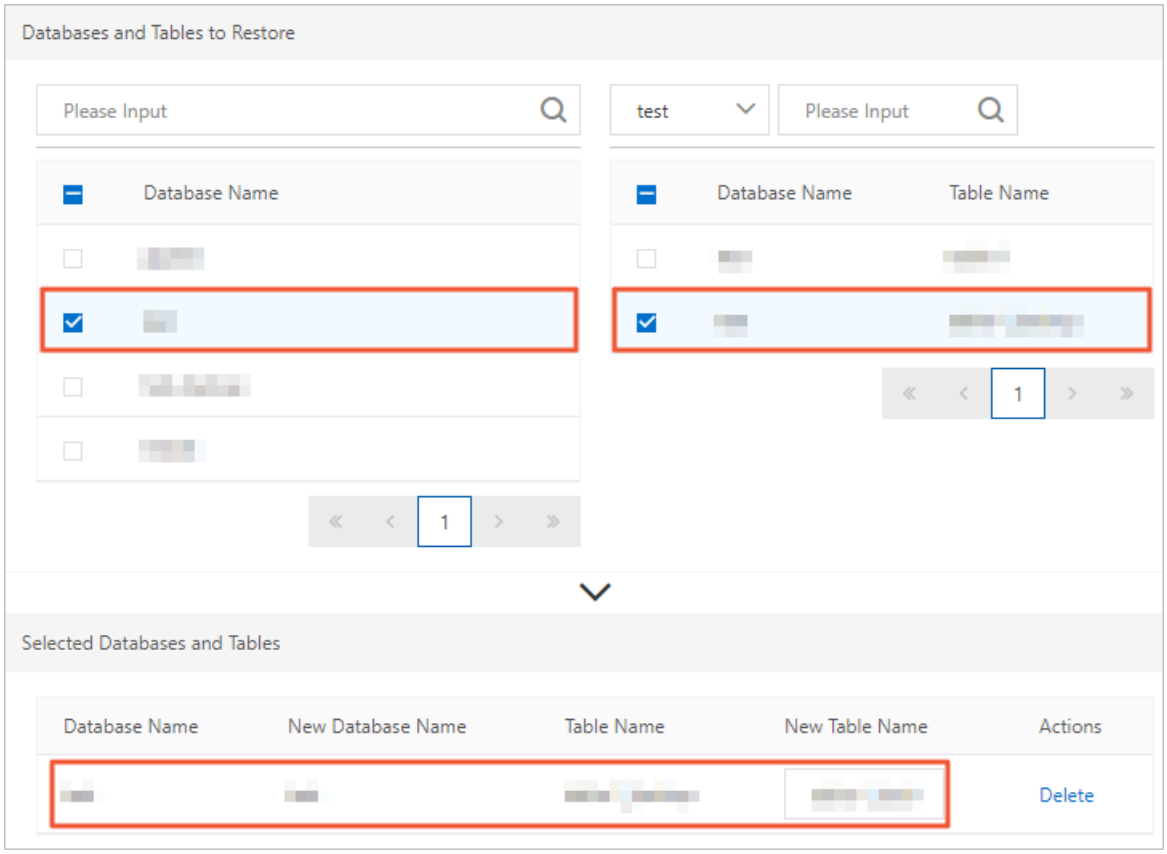
Procedure

1.
2.
3.
4. In the left-side navigation pane, choose **Settings and Management > Backup and Restore**.
5. On the **Backup and Restore** page, click **Restore Databases/Tables**.
6. In the dialog box that appears, set **Restore Type** to **Backup Timepoint** and configure **Restoration Time**.



> **Note** The point in time specified by **Restore To** must fall within the time range that is specified by **Restoration Time**. The full backup set that is most recent to the specified point in time must contain the table that you want to restore. This way, the **Backup Timepoint** restore type can be used. The time range specified by the Restore To parameter is determined by the value of **Log Retention Period (Days)**. The default value of the Log backup retention parameter is 7 days.

7. In the **Databases and Tables to Restore** section, select a database to restore in the left and then select tables to restore in the right.



> **Note**
>
> - If you select no tables after you select a database, all data in the database is restored.
> - After you select a database or a table, the system specifies the `_backup` suffix in the name of the original database or the original table to name the new database or the new table. For example, if the name of the original table is `test`, the new table is named `test_backup`. You can customize names for new databases and new tables.

8. Click **OK**.

Related API operations

API operation	Description
DescribeMetaList	You can call the DescribeMetaList operation to query the metadata of the instance or table that you want to restore.
RestoreTable	You can call the RestoreTable operation to restore the instances or tables of a cluster.

9.4.5. Restore data that is deleted accidentally

There are various scenarios where data is deleted or modified accidentally, such as deleting, modifying, or overwriting a column or row of data in a table, and deleting a table, database, or cluster accidentally.

provides multiple methods to restore data for different scenarios and database engine versions.

For more information about the data restoration methods and steps, see Data restoration methods for different scenarios.

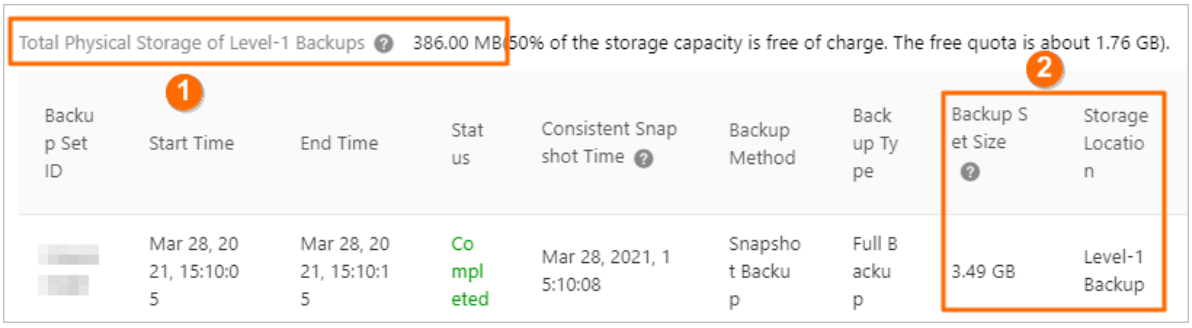
9.5. FAQ

This topic provides answers to frequently asked questions about the backup and restoration features of .

Data backup FAQ

- Is the total size of level-1 backups (snapshots) equal to the sum of the sizes of all level-1 backups (snapshots)?

  No, the total size of level-1 backups (snapshots) is not equal to the sum of the sizes of all level-1 backups (snapshots). The total size of level-1 backups (snapshots) is displayed in part ①, as shown in the following figure.



- Why is the total size of level-1 backups smaller than the sum of the sizes of all level-1 backups?

  The size of level-1 backups in is measured in two forms: the logical size of backups and the total physical storage of backups. uses snapshot chains to store level-1 backups. Only one record is generated for each data block. Therefore, the total physical storage of all level-1 backups is smaller than the total logical size of all level-1 backups. In some cases, the total physical storage of all level-1 backups is smaller than the logical size of a single backup.

- How am I charged for backups in ?

  You are charged for storage space of level-1, level-2, and log backups. By default, the level-1 backup and log backup features are enabled, and a free storage quota is provided. By default, the level-2 backup feature is disabled.

- How are the fees of level-1 backups calculated?

  The fee is calculated based on the following formula: Storage fee per hour = (Total size of level-1 backups-Used database storage space × 50%) × Price per hour. For example, the total size of level-1 backups of a database is 700 GB, and the used database storage space is 1,000 GB. Then, the storage fee per hour is calculated based on the following formula: [700 GB-500 GB] × USD 0.000464/GB = USD 0.0928.

- Can I use a storage plan to offset the storage fees of backups?

Yes, you can purchase a storage plan to offset the storage space used by all clusters within your account. The remaining capacity of the storage plan is automatically used to offset the storage space that exceeds the free quota for level-1 backups at a ratio of 1:1.6 until the storage plan is exhausted. If the remaining capacity of the storage plan is insufficient to offset the storage space of level-1 backups, you are charged for additional storage space on a pay-as-you-go basis. For more information, see Storage plans .

- Are level-1 backups the only type of backup that can be manually created?

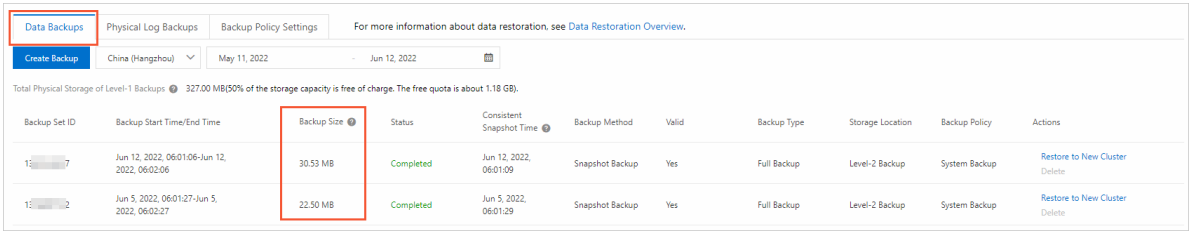
  Yes, only level-1 backups can be manually created.

- How long are manually created backups retained?

  The retention period of manually created backups is specified by the Level-1 Backup parameter in the Data Backup Retention Period section.

- How do I view the size of a level-2 backup?

  You can view the size of a level-2 backup on the **Backups** tab in the console.



- How do I download backup sets to the on-premises file system?

  uses a snapshot backup mechanism at the storage layer. Exported files cannot be directly restored to the on-premises file system. To download a backup set, you can use the mysqldump tool or use Database Backup (DBS) to back up a PolarDB for MySQL cluster and manually download the backup. For more information, see Back up a PolarDB for MySQL instance and Manually download a backup set.

Data restoration FAQ

- How can I restore data that was deleted or modified by accident?

  You can choose different methods to restore data based on your business scenario and database engine version. For more information about the data restoration methods and steps, see Data restoration methods for different scenarios.

- Why is the database and table restoration feature temporarily unavailable?

  Check whether your cluster has an excessive number of tables. The database and table restoration feature can be applied only to clusters that have 50,000 tables or less. Databases or tables cannot be restored for a cluster that has more than 50,000 tables.

- Can I customize the names of restored databases or tables?

  Yes, you can customize the names of restored databases or tables.

- If my cluster does not have a data backup, can I restore the data to a previous point in time?

  No. To restore data to a previous point in time, you must restore the data of a full backup that was created before the specified point in time. Then, you must restore the data generated after the backup that was created and before the specified point in time based on the redo logs.

10.Failover with hot standby

10.1. Overview

provides the failover with hot standby feature. You can enable hot standby for the read-only nodes in your cluster to improve failover speed and avoid service interruptions.

also provides an advanced feature for high availability scenarios. The advanced feature is fast failover with hot standby. This feature uses a number of technologies to improve failover speed and user experience in scenarios such as scheduled switchover (cluster configuration changes) and unexpected switchover (failover for disaster recovery). For more information about the core technologies utilized in the failover with hot standby feature, see How failover with hot standby works.

Supported versions

Only Edition 5.7 and 8.0 clusters support the failover with hot standby feature.

Scenarios

Usage improvement for idle nodes

To simplify business logic, the primary endpoint is used to provide services in scenarios where strong consistency is required for read requests, more write requests need to be processed than read requests, and data is frequently updated. By default, creates at least one read-only node for the purpose of disaster recovery. You can enable failover with hot standby for read-only nodes that do not provide scalable read services. This helps improve the performance of dirty page flushing, availability, and disaster recovery capabilities of the primary node.

Cluster configuration changes

Hot standby nodes use a global prefetching system to speed up failover, and provide persistent connections and transaction resumable upload to implement O&M without interrupting your services. After hot standby and transaction resumable upload are enabled, provides persistent connections and reliable transactions in the following scenarios: **active O&M**, **cluster configuration changes**, and **minor version upgrade**, which significantly reduces the number of errors reported by clients.

Data import

A large volume of data needs to be imported during off-peak hours or when new instances are created. You can temporarily enable hot standby during the data import phase. This will allocate more resources to the primary node to process write requests, improving the performance of the cluster.

High-availability read-only nodes

In scenarios where users have high requirements for the availability of analytical services, supports dynamic switchover between read-only nodes and hot standby nodes. You can configure an additional hot standby node. This hot standby node can serve as a backup node for the primary node or a backup node for the read-only node of an analytical service.

10.2. Billing

This topic describes the billing rules of the failover with hot standby feature.

The failover with hot standby feature is free of charge. You only pay for read-only nodes where the failover with hot standby feature is enabled. For more information about the fees of read-only nodes, see Billing rules of pay-as-you-go compute nodes and Billing rules of subscription compute nodes.

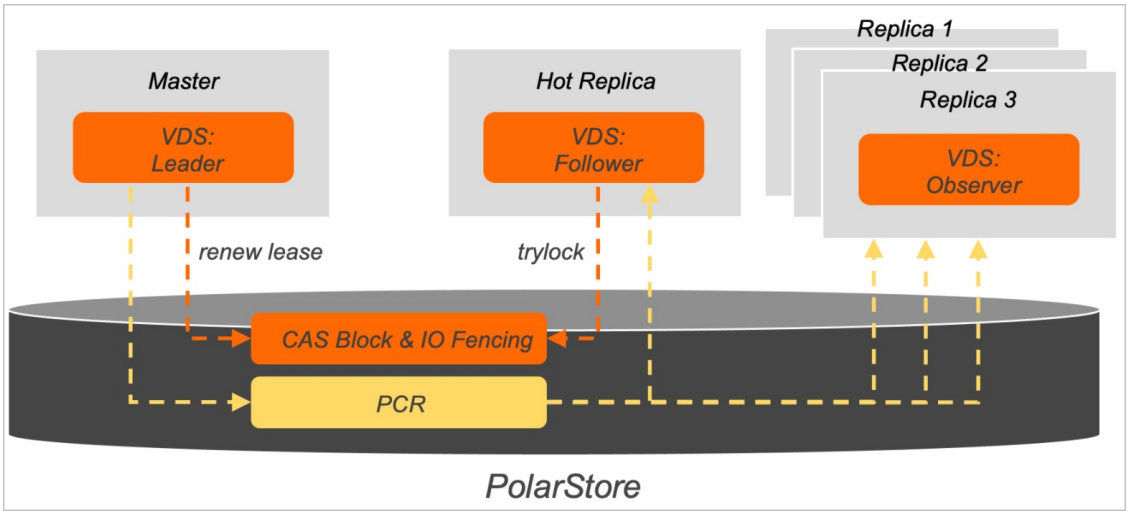
10.3. How failover with hot standby works

This topic describes how the failover with hot standby feature of works.

The following core technologies are utilized to implement the failover with hot standby feature of :

- Voting Disk Service (VDS): VDS is a high availability module based on the shared disk architecture that can be used to implement autonomous management of cluster nodes. VDS greatly shortens the time required for fault detection and primary node selection.
- Global prefetching: Powered by global prefetching, hot standby nodes can prefetch multiple modules inside the storage engine, reducing the time required for failovers.
- Persistent connections and transaction resumable upload: After persistent connections and transaction resumable upload are enabled, PolarDB implements active O&M without interrupting your services during cluster configuration changes or minor version upgrade.

New high availability module: VDS



After hot standby is enabled, activates VDS. With the shared disk architecture of , VDS provides autonomous management, fault detection, and primary node selection of cluster nodes. The following items describe the architecture of VDS:

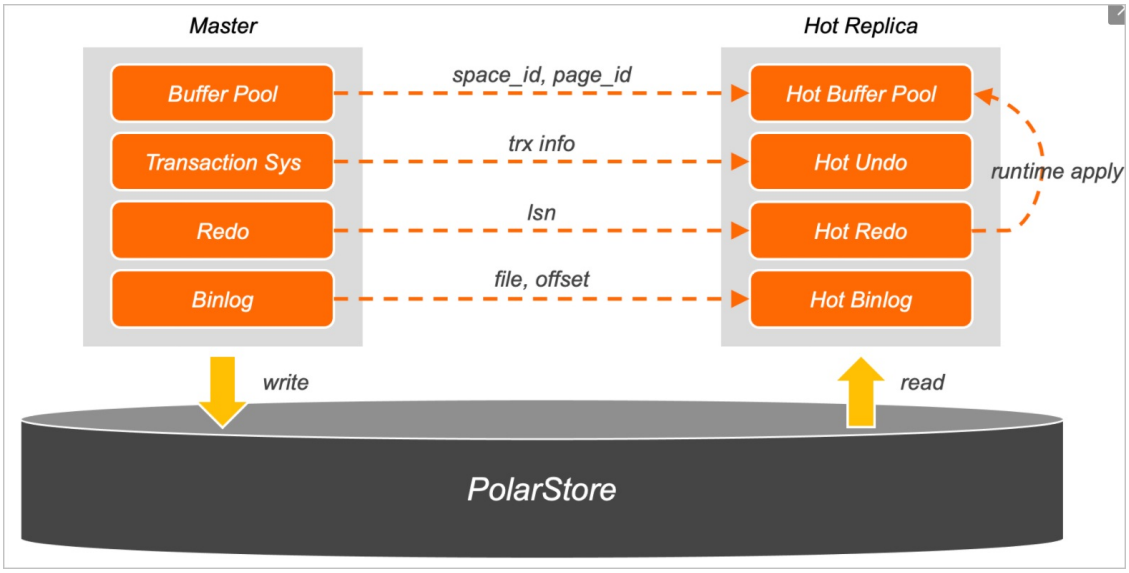
- Each compute node in VDS has an independent thread. VDS threads are classified into three categories: Leader, Follower, and Observer. In a cluster, Leader threads run on the primary node, Follower threads run on the hot standby node, and Observer threads run on read-only nodes. A cluster can contain one Leader thread, one Follower thread, and multiple Observer threads.
- VDS creates two data modules - Compare-and-Swap (CAS) Block and Polar Cluster Registry (PCR) - on PolarStore.
  - CAS Block is an atomic data block that supports CAS operations provided by PolarStore. CAS Block allows for lease-based distributed locks in VDS and records metadata such as lock holder and lease time. The primary node and hot standby node of a cluster use lock acquirement and lock renewal semantics to detect faults and select the primary node.

- PCR is a file that stores the metadata of node management, such as the topology status of a cluster. Leader threads have the permission to write data to PCR, while Follower and Observer threads have only the permission to read data from PCR. When a Follower thread is designated as a Leader thread, the original Leader thread can no longer write data to PCR. Only the latest Leader thread has the permission to write data to PCR.

In general, the primary node provides read and write services, and the corresponding Leader thread regularly renews its lock in VDS. When the primary node becomes unavailable, a hot standby node takes over. The process is described in the following section:

1. After the lease of the primary node expires, a Follower thread is locked and is elected as the Leader thread. At this point, the hot standby node becomes the primary node.
2. When the original primary node recovers, it fails to acquire a lock. Then, the original primary node is degraded to a hot standby node.
3. When the primary node selection process is completed, PCR broadcasts the new topology information to all Observer threads. In this way, read-only nodes can automatically connect to the new primary node and restore the synchronization links for log sequence numbers (LSNs) and binary logs.

Global prefetching system



Unlike read-only nodes, hot standby nodes do not provide read services. Hot standby nodes are mainly used to improve failover speed and improve availability. The global prefetching system is the most important module in failover with hot standby. It synchronizes the metadata of the primary node in real time and prefetches key data into the memory to improve failover speed. The global prefetching system consists of four modules: Hot Buffer Pool, Hot Undo, Hot Redo, and Hot Binlog.

- **Hot Buffer Pool**

  The Hot Buffer Pool module monitors the linked list that is used to implement the Least Recently Used (LRU) algorithm in the buffer pool for the primary node in real time and sends relevant data to the hot standby node. The hot standby node selects frequently accessed pages and prefetches them to the memory to avoid performance degradation caused by a significant drop in the buffer pool hit rate when a read-only node is elected as the primary node.

- **Hot Undo**

  The Hot Undo module prefetches data in the transaction system. During a failover, needs to find the

pending transactions from undo pages and roll back the transactions. Read-only nodes process only large-scale analytical query requests and do not access uncommitted transactions of the primary node. This leads to long I/O wait time for undo pages. The Hot Undo module prefetches undo pages and transaction information, improving failover speed.

- Hot Redo

  The Hot Redo module caches the redo logs of hot standby and read-only nodes in the redo hash table of the memory in real time.

- Hot Binlog

  After Hot Binlog is enabled, InnoDB transactions in the Prepare state decide whether to commit or roll back a transaction based on binary logs. When a large number of transactions are executed, the system may require several seconds or minutes to read and parse all the binary logs. Hot standby nodes use background threads to asynchronously cache the latest binary logs in the I/O cache and parse them in advance to improve failover speed.

Persistent connections and transaction resumable upload

Failover occurs in scenarios such as cluster configuration changes, minor version upgrade, active O&M, and disaster recovery. However, the failover may affect your service and cause issues such as temporary service interruptions and connection failures. This increases the complexity and risks of application development.

supports the Persistent connections feature. Persistent connections are implemented in the way that the database proxy serves as a connecting bridge between your application and . When the database performs a primary/secondary failover, the database proxy connects database nodes to your application and restores the previous session, including the original system variables, user variables, character set encoding, and other information.

The persistent connections function can be applied to only idle connections. If the current session has a transaction that is being executed at the moment when the node is switched, the database proxy cannot retrieve the original transaction context from . The new primary node will roll back the uncommitted transactions and release the row locks held by these transactions. In this case, persistent connections cannot be maintained. To solve this issue, provides the transaction resumable upload function. Transaction resumable upload, together with the persistent connections function, allows for fast failover to deliver high availability without interrupting your services.

Unlike logical replication based on binary logs, the physical replication architecture allows to rebuild the same transactions on the hot standby node as on the primary node.

For example, the process of committing a transaction in an application is BEGIN > INSERT > UPDATE > COMMIT. After the transaction starts to be executed, the database proxy caches the most recently executed SQL statement while forwarding the SQL statement to the primary node. After the insert operation is executed on the primary node, automatically saves the savepoint of the latest statement as part of the transaction information. The session tracker returns the current session and transaction information to the database proxy. Then, the database proxy temporarily saves the data to the internal cache. The session information, such as character sets and user variables, is used for maintaining connections. The transaction information, such as trx_id and undo_no, is used for transaction resumable upload. In addition, the transaction information is continuously synchronized to the hot standby node through physical replication.

Assume that the primary node is unavailable when executes the update operation on your application. After the failover, the new primary node can build all uncommitted transactions based on redo logs and asynchronously wait for uncommitted transactions without rolling back the transactions. When the database proxy detects the failover, it will use the session and transaction information cached by itself to rebuild the transaction by calling the Attach Trx API of . determines whether the transaction information is valid based on the proxy information. If the transaction information is valid, it will be bound to the connection and rolled back to the savepoint of the undo_no corresponding to the last statement (the UPDATE statement).

After the transaction is rebuilt, the database proxy can resend the latest UPDATE statement that failed to be executed to the new primary node from the SQL statement cache. During the failover process, no connection or transaction errors are reported on your application. The only difference is that the UPDATE statement may be slower than normal.

10.4. Cluster node performance comparison before and after hot standby is enabled

This topic describes the performance differences before and after hot standby is enabled.

Comparison overview

The following table lists the performance differences between hot standby nodes (with the hot standby feature enabled) and read-only nodes:

Item	Hot standby node	Read-only node
Service capabilities	Does not provide services	Provides read-only services
Switchover (active O&M, such as minor version upgrade and cluster configuration changes)	5 seconds before transactions per second (TPS) drops to zero	Over 10 seconds before TPS drops to zero
Failover (disaster recovery, such as recovering the primary node)	5 seconds before TPS drops to zero	60 seconds of service interruption

The preceding table indicates that:

- In active O&M scenarios, read-only nodes with hot standby enabled experience shorter service interruption time.
- In disaster recovery scenarios, read-only nodes with hot standby enabled have shorter service interruption time. In addition, services and transactions are not interrupted, and the number of errors reported by clients is significantly reduced.

Detailed information

The following section provides detailed information about failover speed before and after hot standby is enabled:

- **Switchover (active O&M, such as minor version upgrade and cluster configuration changes)**

- Normal read-only nodes (when hot standby and transaction resumable upload are disabled): over 10 seconds before TPS drops to zero

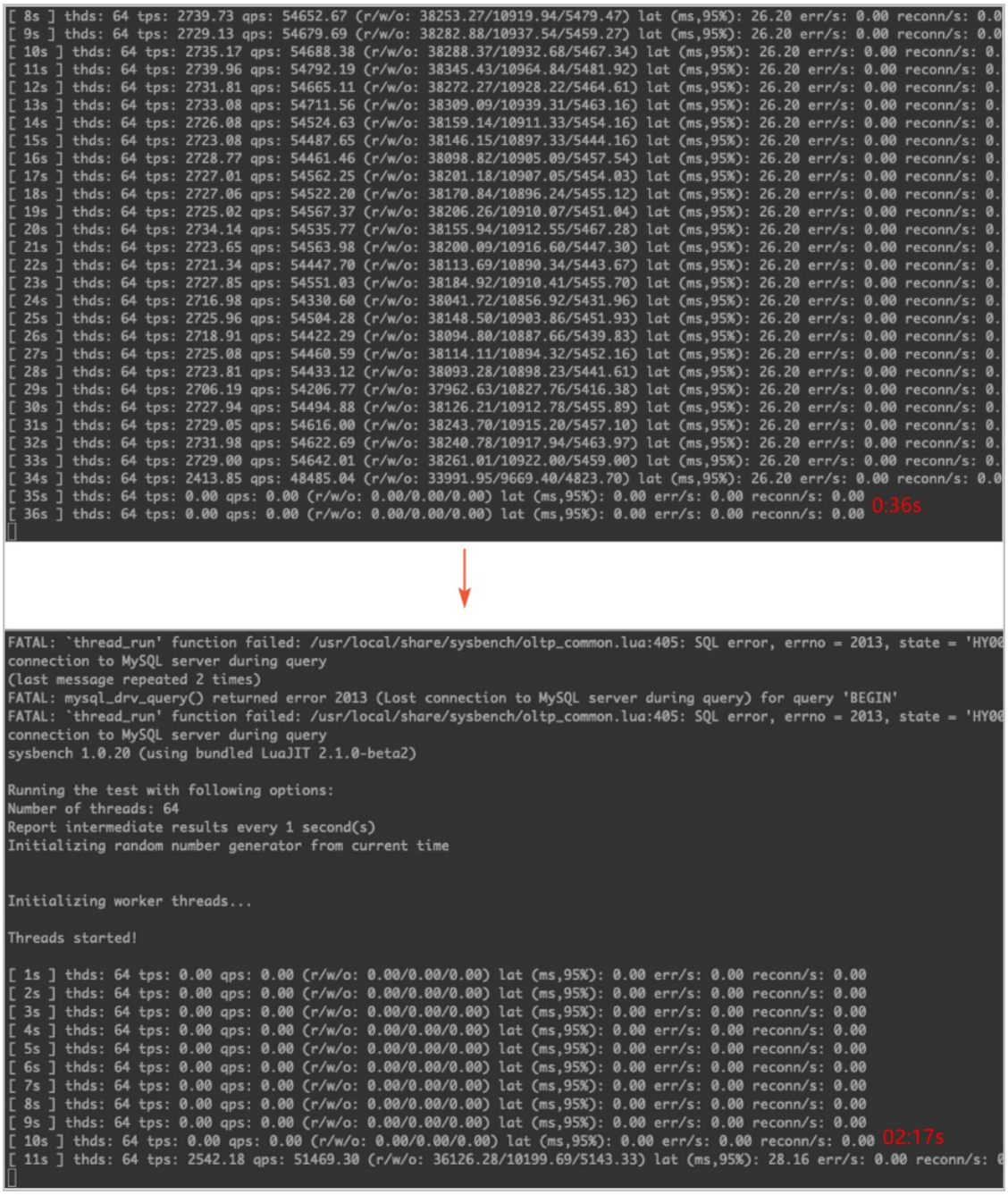
```
[ 36s ] thds: 64 tps: 2708.19 qps: 54140.88 (r/w/o: 37904.72/10819.78/5416.39) lat (ms,95%): 26.20 err/s: 0.00 reconn/s: 0.
[ 37s ] thds: 64 tps: 2703.00 qps: 54030.10 (r/w/o: 37810.07/10815.02/5405.01) lat (ms,95%): 26.20 err/s: 0.00 reconn/s: 0.
[ 38s ] thds: 64 tps: 2698.80 qps: 54018.04 (r/w/o: 37814.23/10806.21/5397.60) lat (ms,95%): 26.20 err/s: 0.00 reconn/s: 0.
[ 39s ] thds: 64 tps: 2703.02 qps: 54084.38 (r/w/o: 37854.27/10824.08/5406.04) lat (ms,95%): 26.20 err/s: 0.00 reconn/s: 0.
[ 40s ] thds: 64 tps: 2694.98 qps: 53889.61 (r/w/o: 37730.73/10768.92/5389.96) lat (ms,95%): 26.68 err/s: 0.00 reconn/s: 0.
[ 41s ] thds: 64 tps: 2704.02 qps: 54137.33 (r/w/o: 37921.23/10806.07/5410.03) lat (ms,95%): 26.20 err/s: 0.00 reconn/s: 0.
[ 42s ] thds: 64 tps: 2703.02 qps: 53980.34 (r/w/o: 37753.24/10822.07/5405.03) lat (ms,95%): 26.68 err/s: 0.00 reconn/s: 0.
[ 43s ] thds: 64 tps: 2696.99 qps: 54029.83 (r/w/o: 37838.88/10796.97/5393.98) lat (ms,95%): 26.20 err/s: 0.00 reconn/s: 0.
[ 44s ] thds: 64 tps: 2715.99 qps: 54255.82 (r/w/o: 37964.88/10859.96/5430.98) lat (ms,95%): 26.20 err/s: 0.00 reconn/s: 0.
[ 45s ] thds: 64 tps: 2714.98 qps: 54295.60 (r/w/o: 38013.72/10851.92/5429.96) lat (ms,95%): 26.20 err/s: 0.00 reconn/s: 0.
[ 46s ] thds: 64 tps: 2711.02 qps: 54248.37 (r/w/o: 37974.26/10851.07/5423.04) lat (ms,95%): 26.20 err/s: 0.00 reconn/s: 0.
[ 47s ] thds: 64 tps: 2714.02 qps: 54231.31 (r/w/o: 37947.22/10856.06/5428.03) lat (ms,95%): 26.20 err/s: 0.00 reconn/s: 0.
[ 48s ] thds: 64 tps: 2713.99 qps: 54246.76 (r/w/o: 37977.84/10839.95/5428.98) lat (ms,95%): 26.20 err/s: 0.00 reconn/s: 0.
[ 49s ] thds: 64 tps: 2700.02 qps: 54050.31 (r/w/o: 37839.21/10816.06/5395.03) lat (ms,95%): 26.20 err/s: 0.00 reconn/s: 0.
[ 50s ] thds: 64 tps: 2710.98 qps: 54088.55 (r/w/o: 37840.68/10823.91/5423.95) lat (ms,95%): 26.20 err/s: 0.00 reconn/s: 0.
[ 51s ] thds: 64 tps: 2550.12 qps: 50488.44 (r/w/o: 35268.70/10179.49/5040.24) lat (ms,95%): 26.20 err/s: 0.00 reconn/s: 0.
[ 52s ] thds: 64 tps: 0.00 qps: 0.00 (r/w/o: 0.00/0.00/0.00) lat (ms,95%): 0.00 err/s: 0.00 reconn/s: 0.00
[ 53s ] thds: 64 tps: 0.00 qps: 0.00 (r/w/o: 0.00/0.00/0.00) lat (ms,95%): 0.00 err/s: 0.00 reconn/s: 0.00
[ 54s ] thds: 64 tps: 0.00 qps: 0.00 (r/w/o: 0.00/0.00/0.00) lat (ms,95%): 0.00 err/s: 0.00 reconn/s: 0.00
[ 55s ] thds: 64 tps: 0.00 qps: 0.00 (r/w/o: 0.00/0.00/0.00) lat (ms,95%): 0.00 err/s: 0.00 reconn/s: 0.00
[ 56s ] thds: 64 tps: 0.00 qps: 0.00 (r/w/o: 0.00/0.00/0.00) lat (ms,95%): 0.00 err/s: 0.00 reconn/s: 0.00
[ 57s ] thds: 64 tps: 0.00 qps: 0.00 (r/w/o: 0.00/0.00/0.00) lat (ms,95%): 0.00 err/s: 0.00 reconn/s: 0.00
[ 58s ] thds: 64 tps: 0.00 qps: 0.00 (r/w/o: 0.00/0.00/0.00) lat (ms,95%): 0.00 err/s: 0.00 reconn/s: 0.00
[ 59s ] thds: 64 tps: 0.00 qps: 0.00 (r/w/o: 0.00/0.00/0.00) lat (ms,95%): 0.00 err/s: 0.00 reconn/s: 0.00
[ 60s ] thds: 64 tps: 0.00 qps: 0.00 (r/w/o: 0.00/0.00/0.00) lat (ms,95%): 0.00 err/s: 0.00 reconn/s: 0.00
[ 61s ] thds: 64 tps: 0.00 qps: 0.00 (r/w/o: 0.00/0.00/0.00) lat (ms,95%): 0.00 err/s: 0.00 reconn/s: 0.00
[ 62s ] thds: 64 tps: 0.00 qps: 0.00 (r/w/o: 0.00/0.00/0.00) lat (ms,95%): 0.00 err/s: 0.00 reconn/s: 0.00
[ 63s ] thds: 64 tps: 0.00 qps: 0.00 (r/w/o: 0.00/0.00/0.00) lat (ms,95%): 0.00 err/s: 0.00 reconn/s: 0.00
[ 64s ] thds: 64 tps: 0.00 qps: 0.00 (r/w/o: 0.00/0.00/0.00) lat (ms,95%): 0.00 err/s: 0.00 reconn/s: 0.00
```

- Hot standby nodes (when hot standby and transaction resumable upload are enabled): 5 seconds before TPS drops to zero

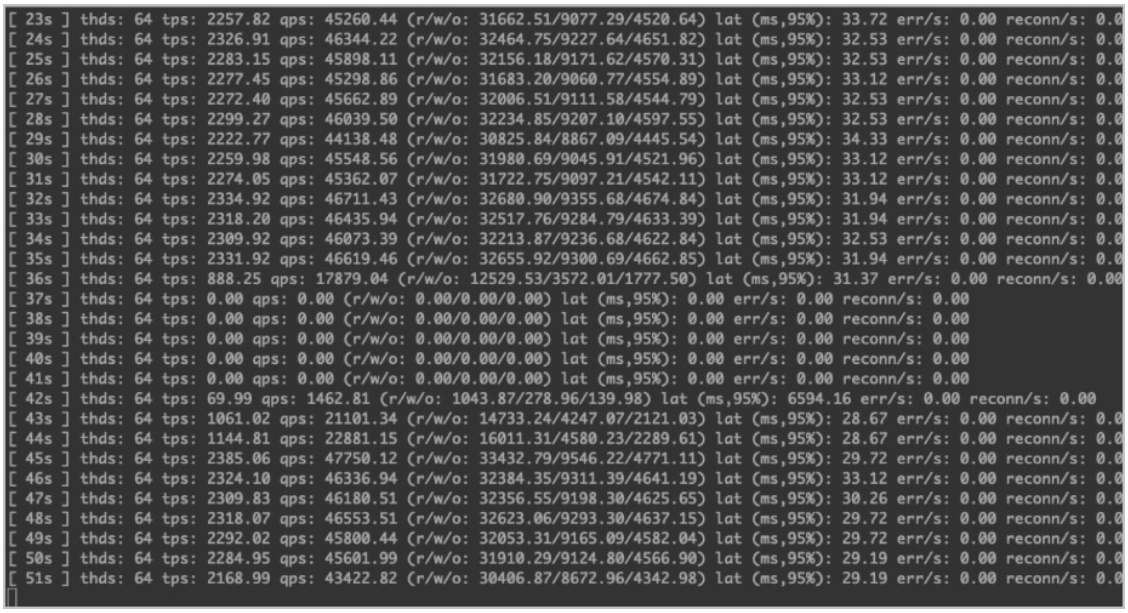
```
[ 33s ] thds: 64 tps: 2427.11 qps: 48339.19 (r/w/o: 33788.53/9696.44/4854.22) lat (ms,95%): 30.81 err/s: 0.00 reconn/s: 0.0
[ 34s ] thds: 64 tps: 2391.22 qps: 48200.34 (r/w/o: 33794.04/9621.87/4784.43) lat (ms,95%): 30.81 err/s: 0.00 reconn/s: 0.0
[ 35s ] thds: 64 tps: 2427.71 qps: 48447.26 (r/w/o: 33919.98/9671.85/4855.43) lat (ms,95%): 30.81 err/s: 0.00 reconn/s: 0.0
[ 36s ] thds: 64 tps: 2456.12 qps: 48941.31 (r/w/o: 34219.61/9813.46/4908.23) lat (ms,95%): 30.26 err/s: 0.00 reconn/s: 0.0
[ 37s ] thds: 64 tps: 2402.02 qps: 48214.46 (r/w/o: 33790.32/9616.09/4808.05) lat (ms,95%): 30.81 err/s: 0.00 reconn/s: 0.0
[ 38s ] thds: 64 tps: 2377.98 qps: 47430.59 (r/w/o: 33179.71/9494.92/4755.96) lat (ms,95%): 31.37 err/s: 0.00 reconn/s: 0.0
[ 39s ] thds: 64 tps: 2395.97 qps: 48032.41 (r/w/o: 33600.59/9640.88/4790.94) lat (ms,95%): 30.81 err/s: 0.00 reconn/s: 0.0
[ 40s ] thds: 64 tps: 2360.57 qps: 47201.33 (r/w/o: 33074.94/9407.26/4719.13) lat (ms,95%): 31.37 err/s: 0.00 reconn/s: 0.0
[ 41s ] thds: 64 tps: 2465.96 qps: 49155.27 (r/w/o: 34367.49/9855.85/4931.93) lat (ms,95%): 29.72 err/s: 0.00 reconn/s: 0.0
[ 42s ] thds: 64 tps: 2436.03 qps: 48872.56 (r/w/o: 34237.39/9764.11/4871.06) lat (ms,95%): 29.72 err/s: 0.00 reconn/s: 0.0
[ 43s ] thds: 64 tps: 2311.01 qps: 46276.24 (r/w/o: 32385.17/9265.05/4626.02) lat (ms,95%): 33.12 err/s: 0.00 reconn/s: 0.0
[ 44s ] thds: 64 tps: 2382.00 qps: 47578.10 (r/w/o: 33299.07/9519.02/4760.01) lat (ms,95%): 31.37 err/s: 0.00 reconn/s: 0.0
[ 45s ] thds: 64 tps: 2303.99 qps: 46139.88 (r/w/o: 32303.92/9223.98/4611.99) lat (ms,95%): 33.12 err/s: 0.00 reconn/s: 0.0
[ 46s ] thds: 64 tps: 2266.00 qps: 45233.03 (r/w/o: 31658.02/9044.01/4531.00) lat (ms,95%): 33.72 err/s: 0.00 reconn/s: 0.0
[ 47s ] thds: 64 tps: 2448.98 qps: 48876.56 (r/w/o: 34201.69/9776.91/4897.96) lat (ms,95%): 29.72 err/s: 0.00 reconn/s: 0.0
[ 48s ] thds: 64 tps: 2443.01 qps: 49073.25 (r/w/o: 34396.17/9791.05/4886.02) lat (ms,95%): 29.72 err/s: 0.00 reconn/s: 0.0
[ 49s ] thds: 64 tps: 2369.98 qps: 47161.57 (r/w/o: 32979.70/9444.91/4736.96) lat (ms,95%): 31.37 err/s: 0.00 reconn/s: 0.0
[ 50s ] thds: 64 tps: 2412.03 qps: 48338.63 (r/w/o: 33826.44/9692.13/4820.06) lat (ms,95%): 30.26 err/s: 0.00 reconn/s: 0.0
[ 51s ] thds: 64 tps: 2453.00 qps: 49064.93 (r/w/o: 34346.95/9805.99/4911.99) lat (ms,95%): 29.72 err/s: 0.00 reconn/s: 0.0
[ 52s ] thds: 64 tps: 2462.99 qps: 49209.86 (r/w/o: 34455.90/9826.97/4926.99) lat (ms,95%): 29.72 err/s: 0.00 reconn/s: 0.0
[ 53s ] thds: 64 tps: 2450.97 qps: 48999.48 (r/w/o: 34314.64/9789.90/4894.95) lat (ms,95%): 29.72 err/s: 0.00 reconn/s: 0.0
[ 54s ] thds: 64 tps: 350.02 qps: 6975.32 (r/w/o: 4876.22/1394.06/705.03) lat (ms,95%): 30.26 err/s: 0.00 reconn/s: 0.00
[ 55s ] thds: 64 tps: 0.00 qps: 0.00 (r/w/o: 0.00/0.00/0.00) lat (ms,95%): 0.00 err/s: 0.00 reconn/s: 0.00
[ 56s ] thds: 64 tps: 0.00 qps: 0.00 (r/w/o: 0.00/0.00/0.00) lat (ms,95%): 0.00 err/s: 0.00 reconn/s: 0.00
[ 57s ] thds: 64 tps: 0.00 qps: 0.00 (r/w/o: 0.00/0.00/0.00) lat (ms,95%): 0.00 err/s: 0.00 reconn/s: 0.00
[ 58s ] thds: 64 tps: 484.98 qps: 9434.61 (r/w/o: 6520.73/1950.92/962.96) lat (ms,95%): 4683.57 err/s: 0.00 reconn/s: 0.00
[ 59s ] thds: 64 tps: 2108.03 qps: 42603.70 (r/w/o: 29945.49/8433.14/4225.07) lat (ms,95%): 38.94 err/s: 0.00 reconn/s: 0.0
[ 60s ] thds: 64 tps: 2245.97 qps: 44820.47 (r/w/o: 31301.63/9032.89/4485.95) lat (ms,95%): 34.33 err/s: 0.00 reconn/s: 0.0
[ 61s ] thds: 64 tps: 2187.98 qps: 43689.61 (r/w/o: 30575.73/8737.92/4375.96) lat (ms,95%): 34.95 err/s: 0.00 reconn/s: 0.0
```

- **Failover (disaster recovery, such as recovering the primary node)**

- Normal read-only nodes (when hot standby and transaction resumable upload are disabled): 60 seconds of service interruption



- Hot standby nodes (when hot standby and transaction resumable upload are enabled): 5 seconds before TPS drops to zero



10.5. Configure hot standby nodes

You cannot directly set hot standby nodes when you create a cluster. You can only enable the failover with hot standby feature for existing read-only nodes in your cluster.

Prerequisites

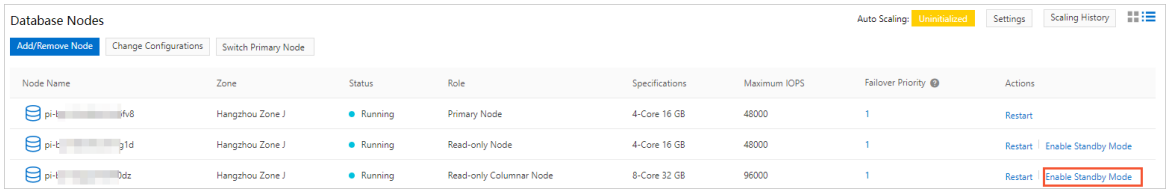
Only Edition 5.7 and 8.0 clusters support the failover with hot standby feature.

Precautions

- If the failover with hot standby feature is not enabled for read-only nodes, your services may be interrupted for about 20 to 30 seconds when failover occurs. Therefore, make sure that your applications can be automatically reconnected to the cluster. If the hot standby feature is enabled for read-only nodes, failover can be completed within 3 to 10 seconds.
- After the failover with hot standby feature is enabled for a read-only node, the node will no longer process read requests.
- The specifications of hot standby nodes must be the same as those of the primary node.

Procedure

1.
2.
3.
4. In the **Database Nodes** section of the **Overview** page, click the icon in the upper-right corner of the section to switch views.
5. In the **Database Nodes** section, find the read-only node for which you want to enable hot standby and click **Enable Hot Standby** in the **Actions** column.



Note When the failover with hot standby feature is enabled for the first time, all nodes in the cluster are restarted one by one. When the failover with hot standby feature is enabled or disabled in the future, the nodes will not be restarted.

11.Flashback queries

PolarDB for MySQL allows you to use the flashback query feature to retrieve data from instances, databases, and data tables as the data was at a past point in time in an efficient manner.

Syntax

```
SELECT <column_name_list> FROM <table_name> AS OF <TIMESTAMP>;
```

Description

- 5.6, 5.7, and 8.0 support this feature.
- Before you use this feature, you must set innodb_backquery_enable to ON in **Parameters** of the requird cluster.

  > **Note** If you execute a flashback query before you set innodb_backquery_enable to ON, the " `ERROR 1815 (HY000): Internal error: the backquery_time set is out of range, too old` " message is returned.

- provides the parameters to implement precise control over the flashback query feature. The following table describes the parameters.

Parameter	Data type	Description
innodb_backquery_enable	BOOL	Specifies whether to enable the flashback query feature. Default value: OFF. Valid values: ○ ON ○ OFF
innodb_backquery_window	ULONG	Specifies the time range based on which to implement the flashback query feature. Valid values: 1 to 2592000. Unit: seconds. Default value: 86400.

Example

- Prepare data for the test:

  Create a table named `products` at 13:51 on `August 31, 2021`. Insert data into the table.

```
create table products (
       prod_id bigint(10) primary key NOT NULL,
       prod_name varchar(20) NOT NULL,
       cust_id bigint(10) NULL,
       createtime datetime NOT NULL DEFAULT NOW()
);
INSERT INTO  products(prod_id,prod_name,cust_id,createtime)
values
(101,'Book',1,NOW()),(102,'Apple',1,NOW()),(103,'Beef',2,NOW()),(104,'Bread',3,NOW()),(10
5,'Cheese',4,NOW());
```

- Query the data in the `products` table.

```
SELECT * FROM products;
+---------+-----------+---------+---------------------+
| prod_id | prod_name | cust_id | createtime          |
+---------+-----------+---------+---------------------+
|     101 | Book      |       1 | 2021-08-31 13:51:22 |
|     102 | Apple     |       1 | 2021-08-31 13:51:24 |
|     103 | Beef      |       2 | 2021-08-31 13:51:26 |
|     104 | Bread     |       3 | 2021-08-31 13:51:27 |
|     105 | Cheese    |       4 | 2021-08-31 13:51:29 |
+---------+-----------+---------+---------------------+
5 rows in set (0.00 sec)
```

- Update the data for the test:

  Update the `products` table at 14:18 on `August 31, 2021`.

```
UPDATE products SET prod_id = 110, createtime = NOW() WHERE prod_name = "Book";
UPDATE products SET prod_id = 119, createtime = NOW() WHERE prod_name = "Apple";
SELECT * FROM products;
+---------+-----------+---------+---------------------+
| prod_id | prod_name | cust_id | createtime          |
+---------+-----------+---------+---------------------+
|     103 | Beef      |       2 | 2021-08-31 13:51:26 |
|     104 | Bread     |       3 | 2021-08-31 13:51:27 |
|     105 | Cheese    |       4 | 2021-08-31 13:51:29 |
|     110 | Book      |       1 | 2021-08-31 14:18:21 |
|     119 | Apple     |       1 | 2021-08-31 14:18:22 |
+---------+-----------+---------+---------------------+
5 rows in set (0.00 sec)
```

- Execute the flashback query:

  View the data in the `products` table based on `2021-08-31 14:00:00`.

```
SELECT * FROM products AS of TIMESTAMP '2021-08-31 14:00:00';
+---------+-----------+---------+---------------------+
| prod_id | prod_name | cust_id | createtime          |
+---------+-----------+---------+---------------------+
|     101 | Book      |       1 | 2021-08-31 13:51:22 |
|     102 | Apple     |       1 | 2021-08-31 13:51:24 |
|     103 | Beef      |       2 | 2021-08-31 13:51:26 |
|     104 | Bread     |       3 | 2021-08-31 13:51:27 |
|     105 | Cheese    |       4 | 2021-08-31 13:51:29 |
+---------+-----------+---------+---------------------+
5 rows in set (0.00 sec)
```

12.High-availability deployment architecture

12.1. Multi-zone deployment architecture

12.1.1. Multi-zone deployment

allows you to deploy a cluster across multiple zones. Compared with single-zone clusters, multi-zone clusters can enhance disaster recovery capabilities and withstand data center-level faults. This topic describes how to deploy a cluster across multiple zones and change the primary zone.

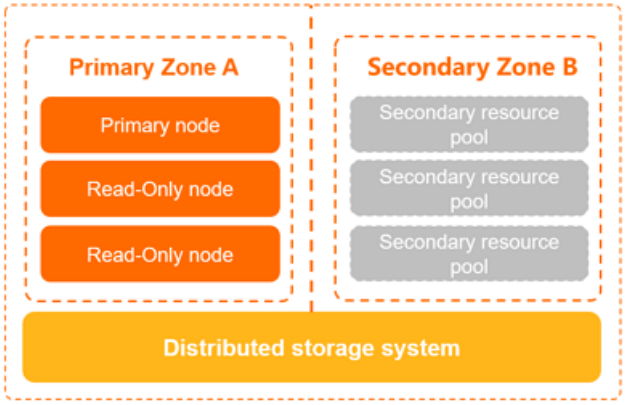
Prerequisites

- The region in which the cluster is deployed contains two or more zones.
- The destination zone has sufficient computing resources.

Multi-zone architecture

When a multi-zone cluster is deployed, data is distributed across zones. Compute nodes must be deployed in the primary zone. reserves sufficient resources in a secondary zone to implement a failover if the primary zone fails.

The following figure shows the multi-zone deployment architecture.



Establish a multi-zone deployment architecture

By default, if the prerequisites are met, a multi-zone cluster is created. For more information about how to create a cluster, see Purchase a pay-as-you-go cluster and Purchase a subscription cluster.

The existing single-zone clusters are upgraded to multi-zone clusters. The upgrades are automatically performed by migrating data online. This does not affect your services.

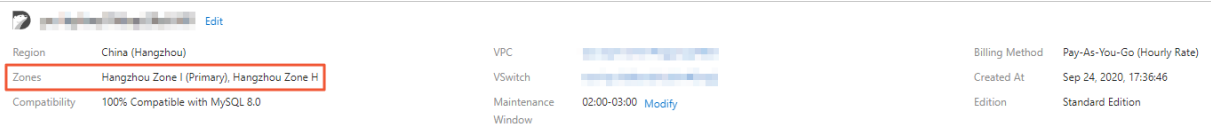
Pricing for multi-zone deployment

No additional fee is charged for multi-zone deployment.

> **Note** You can upgrade your current single-zone cluster to a multi-zone cluster for free.

View the zones of a cluster

On the **Overview** page of the cluster, you can view the value of the **Zones** parameter for the cluster.

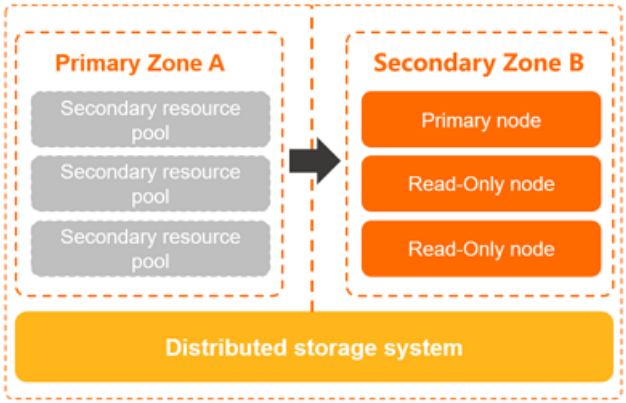


12.1.2. Change the primary zone and vSwitch of a cluster

allows you to change the primary zone and vSwitch of a cluster. You can use this feature to migrate the compute nodes of your cluster to another zone.

Scenarios and note

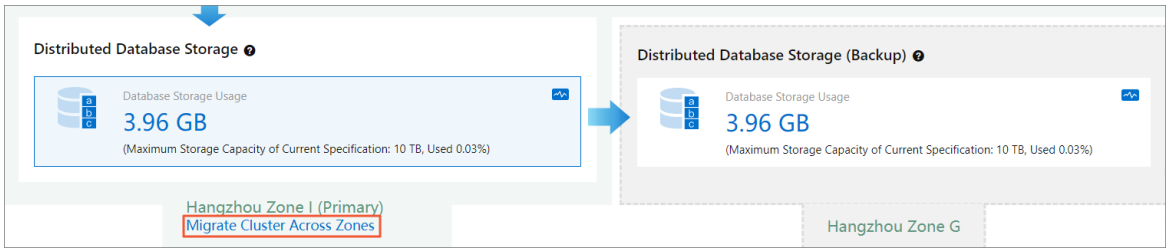
This feature is suitable for disaster recovery and scenarios in which you want to use an Elastic Compute Service (ECS) instance to connect to the PolarDB cluster node in the nearest zone.



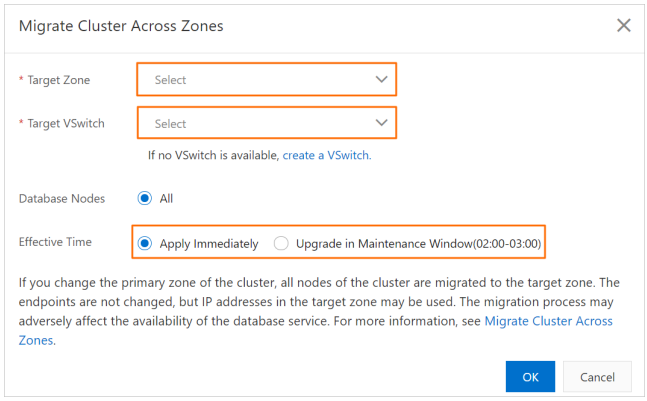
- When services are switched from the primary zone to a secondary zone, one or two transient connections occur. Each transient connection lasts approximately 30s. We recommend that you switch your services in off-peak hours and make sure that your applications can be automatically reconnected to the cluster.
- If the destination zone is a secondary zone, data migration is not required. The system migrates only the compute nodes. This way, only an average of 5 minutes is required to migrate each compute node across data centers. In most cases, this operation is performed in disaster recovery drills.
- If the destination zone is not a secondary zone, the data in the cluster must be migrated. The time required to migrate data depends on the volume of the data. Several hours may be required to migrate the data. Proceed with caution. In most cases, this operation is performed to adjust the layout of applications and databases distributed among zones so that applications can access databases from the nearest zone.
- After the primary zone is changed, the primary endpoint and cluster endpoints that are used to connect to databases remain unchanged. However, the vSwitch and the IP address may change. This operation may affect the availability of the databases. Proceed with caution.

Procedure

1.
2.
3.
4. On the **Overview** page, click **Migrate Cluster Across Zones**.



5. In the dialog box that appears, specify **Target Zone** and **Target VSwitch**, and configure **Effective Time** based on your business requirements.

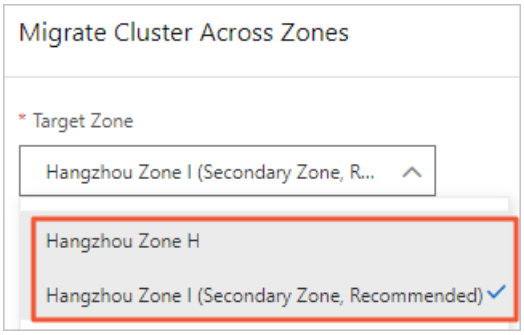


> **Note**
>
> - If the destination zone is a secondary zone, data migration is not required. The system migrates only the compute nodes. This way, only an average of 5 minutes is required to migrate each compute node across data centers. In most cases, this operation is performed in disaster recovery drills.
> - If the destination zone is not a secondary zone, the data in the cluster must be migrated. The time required to migrate data depends on the volume of the data. Several hours may be required to migrate the data. Proceed with caution. In most cases, this operation is performed to adjust the layout of applications and databases distributed among zones so that applications can access databases from the nearest zone.
> - If no vSwitches are available in the destination zone, you must create a vSwitch. For more information, see Create a vSwitch.
> - You can set **Effective Time** to **Apply Immediately** or **Upgrade in Maintenance Window**. If you select **Upgrade in Maintenance Window**, you can view the details about the scheduled task or cancel the task on the **Scheduled Tasks** page. For more information, see View or cancel a scheduled task.

6. Click **OK**.

FAQ

- How much time is required to change the primary zone for a cluster?
- ○ If the destination zone is a secondary zone, data is not migrated. The system migrates only the compute nodes. This way, only an average of 5 minutes is required to migrate each compute node across data centers. In most cases, this operation is performed in disaster recovery drills.
  ○ If the destination zone is not a secondary zone, the data in the cluster must be migrated. The time required to migrate data depends on the volume of the data. If the amount of your data is large, several hours may be required. Proceed with caution. In most cases, this operation is performed to adjust the layout of applications and databases distributed among zones so that applications can access databases from the nearest zone.



- Is the time required to change the primary zone equal to the service downtime? For example, I want to use a secondary zone as the primary zone for a four-node cluster. The average time required to migrate a node is 5 minutes. In this case, are my services unavailable for approximately 20 minutes?

  No, the time required to change the primary zone is not equal to the service downtime. Only one or two transient connections occur. Each transient disconnection lasts approximately 30s. We recommend that you change the primary zone during off-peak hours and make sure that your applications can be automatically reconnected to the cluster.

- How does the system ensure that the cluster runs as expected when the primary zone is being changed?

  After you change the primary zone for a cluster, the primary endpoint and cluster endpoints of the cluster remain unchanged. Therefore, you can still use these endpoints to connect to the cluster. However, the vSwitch and IP address of the cluster may change. This operation may affect the availability of the databases. Proceed with caution.

12.2. Multi-node deployment architecture

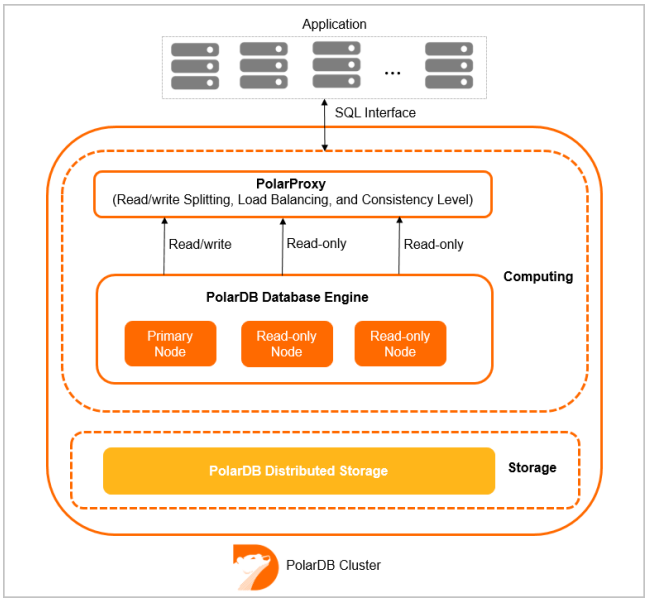
12.2.1. Multi-node deployment

Multiple compute nodes are deployed in the primary zone of a cluster of . These compute nodes consist of one primary node and one or more read-only nodes.

Limits

Only clusters of and support multi-node deployment. and do not support this feature. For more information, see Editions.

Multi-node architecture



Each cluster contains one primary node and one or more read-only nodes. A cluster can contain at most 15 read-only nodes and must contain at least one read-only node. All of the nodes in the same cluster use the same specifications.

The multi-node architecture ensures the high availability of PolarDB clusters. When the primary node in a cluster fails, the cluster can automatically fail over to a read-only node. Then, the read-only node serves as the new primary node.

In addition, PolarProxy of a cluster can be used to perform read/write splitting. For more information, see PolarProxy.

Add or remove read-only nodes

You can manually add or remove read-only nodes to adjust the cluster performance based on your requirements. For more information, see Add or remove read-only nodes.

> **Note**
>
> - The system requires about 5 minutes to add a read-only node. The time consumption depends on multiple factors, such as the number of newly added nodes, the numbers of databases and tables, and the database loads. When adds nodes to a cluster, the databases in the cluster are not affected.
> - When removes a read-only node from a cluster, connections to the node are closed. The connections to other nodes in the cluster are not affected. We recommend that you remove nodes during off-peak hours and make sure that your applications are configured with the automatic reconnection mechanism.
> - We recommend that you connect your application to the cluster endpoint because can automatically re-create connections after you add or remove read-only nodes. This way, you no longer need to modify the application configurations. After you add a read-only node, automatically discovers the node and then balances traffic loads to the node. After you remove a read-only node, automatically filters out the node.

12.2.2. Automatic failover and manual failover

When a system failure occurs, a cluster can automatically switch services from the primary node to a read-only node. You can specify a read-only node as the new primary node to switch services from the primary node to the read-only node.

Precautions

During an automatic failover or a manual failover, transient connections may occur. Each transient connection lasts 20s to 30s. Make sure that your applications can be automatically reconnected to the cluster.

Automatic failover

A cluster of the Cluster Edition uses an active-active architecture that ensures high availability. If the primary node that supports reads and writes is faulty, services are automatically switched to the read-only node that is elected by the system as the new primary node.

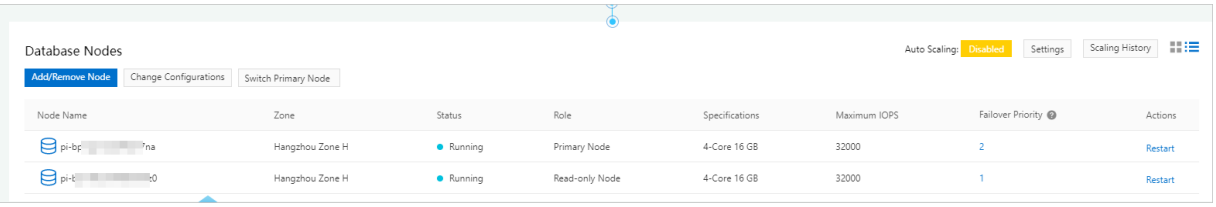
A failover priority is assigned by the system to each node in a cluster. During a failover, a node is elected as the primary node based on the probability that is determined by this priority. The probability of being elected as the primary node is the same for the nodes that are assigned the same failover priority.

The system performs the following steps to elect the primary node:

1. Find all the available read-only nodes that can be elected as the primary node.
2. Select the read-only nodes that are assigned the highest failover priority.
3. If the failover to the first read-only node fails due to network issues, abnormal replication status, or other reasons, the system attempts to switch your services to another read-only node until the failover succeeds.

> **Note** During the failover, transient connections may occur. Each transient connection lasts 20s to 30s. Make sure that your applications can be automatically reconnected to the cluster.

In the **Database Nodes** section of the **Overview** page for the cluster, you can view and configure the failover priority of each node in the cluster.

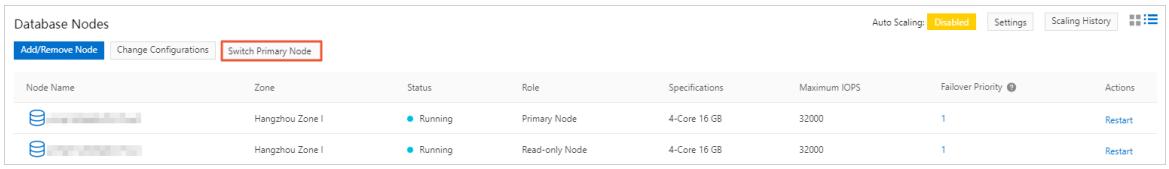


Manual failover

You can specify a read-only node as the new primary node to switch services from the primary node to the read-only node. Manual failovers are suitable for scenarios in which you need to test the high availability of a cluster or specify a read-only node as the primary node of a cluster.

1.
2.
3.

4. In the **Database Nodes** section of the **Overview** page, click in the upper-right corner of the section to switch views.
5. Click **Switch Primary Node**.



6. In the dialog box that appears, specify **New Primary Node** and click **OK**.

> **Note** During the failover, transient connections may occur. Each transient connection lasts 20s to 30s. Make sure that your applications can be automatically reconnected to the cluster.

Related API operations

Operation	Description
FailoverDBCluster	Manually switches services from the primary node to a read-only node of a specified cluster. You can specify a read-only node as the new primary node.

13.Multi-master Architecture

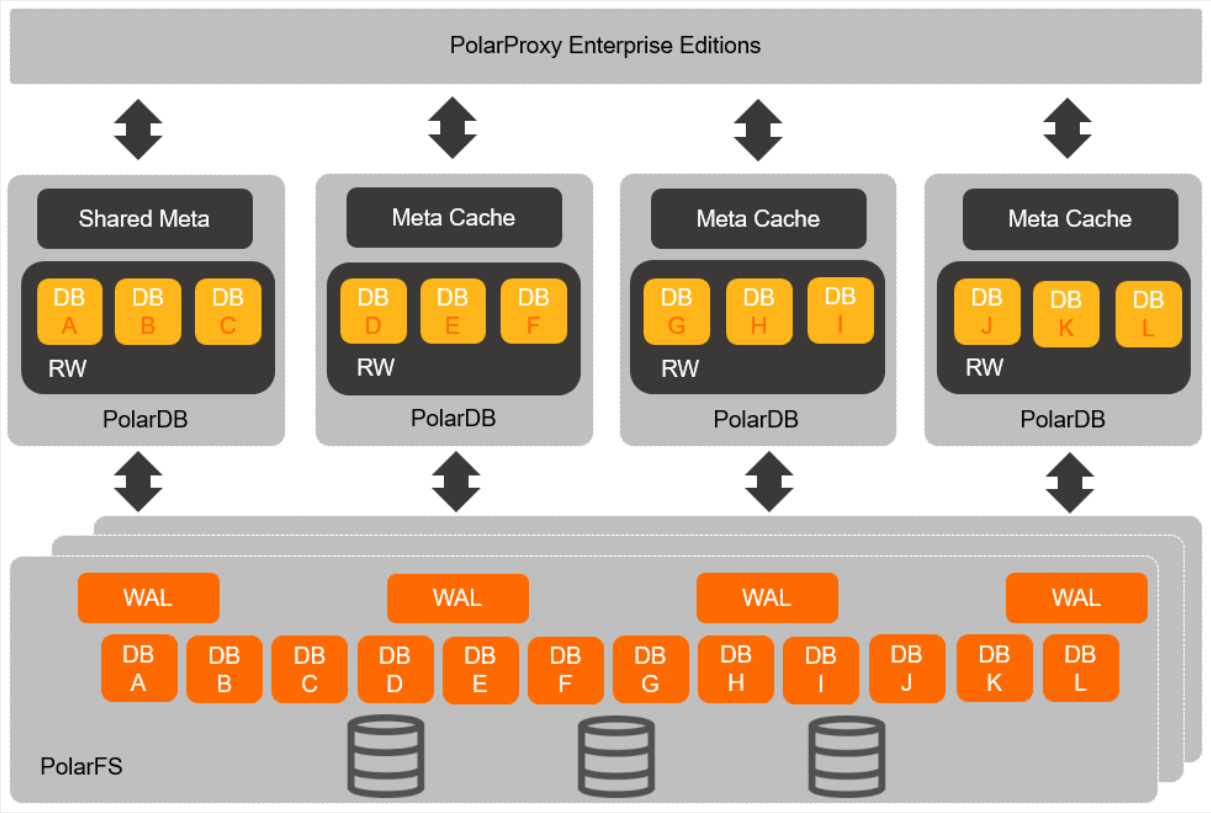
13.1. Multi-master Cluster Edition

This topic describes the .

With the growth of customers, especially tier-1 customers, the architecture that consists of one primary node and multiple read-only nodes cannot provide sufficient write performance that is necessary for the large-scale customer business.

Therefore, provides the multi-master architecture that contains multiple primary nodes and read-only nodes. The new architecture is suitable for high concurrent read and write scenarios such as multitenancy, gaming, and e-commerce.

The following figure shows the multi-master architecture.



All data files in a cluster are stored in PolarStore. Each primary node uses PolarFile System to share data files. You can access all nodes in a cluster by using the cluster endpoint. The database proxy automatically forwards SQL statements to the required primary node.

Core advantages

- Concurrent data writes to databases on different nodes are supported.
- supports one primary node and a maximum of 15 read-only nodes. supports concurrent data writes to databases from a maximum of 32 primary nodes.
- Dynamic failover of nodes for databases can be implemented within seconds to improve the overall concurrent read and write capabilities of clusters.

Scenarios

The multi-master architecture is suitable for scenarios such as multitenancy in software as a service (SaaS), gaming, and e-commerce. These scenarios feature high concurrent read and write requests.

- **Multitenancy in SaaS: high concurrency and load balance between tenants**

  Scenario: The number of databases of tenants rapidly changes, and the load volume undergoes substantial changes. Users must schedule database resources among different instances to deliver optimal experience.

  Solution: The multi-master architecture helps customers to switch between different read-only nodes of databases of tenants. This implements load balance.

- **Gaming applications deployed on different servers: better performance and scalability and global servers supported**

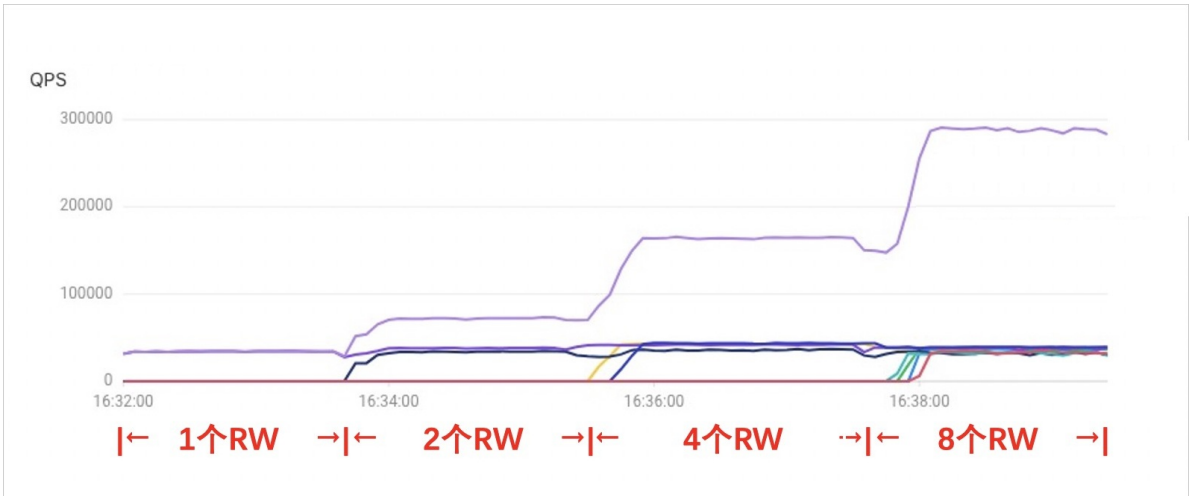
  Scenario: During the growth period of a game, database loads are heavy and feature continual increase. During this period, the number of databases keeps growing. As a result, the loads of primary nodes also increase. During the decline period of a game, database loads are significantly reduced, and databases are merged. As a result, the loads of primary nodes are also decreased.

  Solution: During the growth period, you can switch some databases to new primary nodes to implement load balance. During the decline period, you can aggregate databases to a few primary nodes to reduce operating costs.

Performance improvement

After tests, the overall concurrent read and write capabilities of a cluster show a linear increase because the databases of the cluster are switched to more primary nodes. The following code snippet provides an example of stress testing:

- Test background: The cluster contains eight databases and eight primary nodes.
- Test procedure: At the beginning of a test, eight databases share one primary node. Data is synchronized to all databases at the same time to perform the same stress test. During the stress testing period, eight databases are scheduled to two primary nodes, four primary nodes, and eight primary nodes respectively. View the change trend of the overall performance of the cluster.
- The following figure shows the change trend of QPS.



In the preceding figure, as databases are scheduled to more primary nodes, the overall concurrent read and write capabilities of the cluster are significantly improved and show a linear increase.

Supported kernel versions

Only 8.0 supports Edition.

Node specifications and pricing

supports seven node types of clusters. For more information, see Specifications of compute nodes.

For more information about the billing of , see Billable items.

Usage

The feature is still in the canary release stage. You can submit a ticket to apply for the feature. For more information, see Usage.

> **Note** Only the canary release version of the Multi-master Cluster Edition feature is available for enterprise users.

13.2. Usage

increases the number of primary nodes from which you can write data to databases. The architecture supports concurrent data writes to databases from different primary nodes. The architecture also helps you dynamically switch the primary nodes of databases within seconds to improve the overall concurrent read and write capabilities of clusters. This topic describes how to use .

Prerequisites

- A is purchased. For more information, see Purchase a pay-as-you-go cluster and Purchase a subscription cluster.
- A privileged account is created. For more information, see Create a privileged account.
- You are connected to the cluster. For more information, see Connect to a cluster.

Limits

- The data of each database can be written from only one node. You can read or write data only from a node which have database assigned.
- You can query only the data within one primary node. If you execute a SQL statement to query data from databases on multiple primary nodes, the system reports an error. We recommend that you modify the endpoints of all databases to one primary node before you query data.
- Only cluster endpoints are provided. Primary endpoints are not supported.

Specify the primary node when you create a database

You can create a database on a specified primary node by executing the following statement:

```
CREATE DATABASE name [POLARDB_WRITE_NODE master_id];
```

> **Note** The data of each database can be written from only one node. You can read or write data only from a node which have database assigned.

Example: Create the `db1` database on the RW1 node.

```
CREATE DATABASE db1 POLARDB_WRITE_NODE 1;
```

To create the `db1` database on RW2, replace 1 with 2 in the preceding statement.

Delete a database on a specified primary node

You can delete a database on a specified primary node by executing the following statement:

```
DROP DATABASE name;
```

Example: Delete the `db1` database on the RW1 node.

```
DROP DATABASE db1;
```

When you delete a database, you do not need to specify the POLARDB_WRITE_NODE parameter.

Switch the primary node of a database

You can switch the endpoint of the database to another primary node by executing the following statement:

```
ALTER DATABASE name POLARDB_WRITE_NODE master_id;
```

Example: Switch the endpoint of the `db1` database to the RW2 node.

```
ALTER DATABASE db1 POLARDB_WRITE_NODE 2;
```

Specify the primary node where a SQL statement is executed

Notice This feature is only applicable to the statements not to query data, such as those to query information_schema or status variables. For the statements to query data such as `SELECT * FROM table1`, you do not need to specify a primary node. The database proxy automatically selects the required primary node.

To send an SQL statement to a specified primary node, execute the following SQL statement:

```
ALTER SESSION POLARDB_WRITE_NODE master_id;
```

Example: Query the value of the `innodb_buffer_pool_size` variable on the RW1 node.

```
ALTER SESSION POLARDB_WRITE_NODE 1; # Send the SQL statement to the RW1 node.
SHOW VARIABLES LIKE 'innodb_buffer_pool_size'; # Query the value of the innodb_buffer_pool_
size variable on the RW1 node.
```

Note If you do not specify a primary node where you execute an SQL statement, the database proxy randomly selects a primary node to execute the SQL statement.

Execute the following statement to unlock the primary node where the specified SQL statement is executed:

```
RESET SESSION POLARDB_WRITE_NODE;
```

Query the information about a node

- Execute the following statement to query the database distribution on a primary node:

  ```
  ALTER SESSION POLARDB_WRITE_NODE master_id;
  SELECT * FROM INFORMATION_SCHEMA.INNODB_MASTER_GLOBAL_LOCK_INFO;
  ```

  Example: Query the database distribution on the RW1 node.

  ```
  ALTER SESSION POLARDB_WRITE_NODE 1;
  SELECT * FROM INFORMATION_SCHEMA.INNODB_MASTER_GLOBAL_LOCK_INFO;
  ```

  A similar result is returned:

  ```
  +----------------------+--------------------+----------+--------------+-----------+----
  ---------+
  | table_name           | table_id           | space_id | s_lock_count | lock_mode | cur
  rent_lsn |
  +----------------------+--------------------+----------+--------------+-----------+----
  ---------+
  | mysql/global_ddl_lock | 1043956280258737140 |        0 |            0 | SLS_X     |
  24311657 |
  | db2                  |   27980076883382651 |        0 |            0 | SLS_X     |
  36087702 |
  | db1                  |   27980076883383418 |        0 |            0 | SLS_X     |
  34339564 |
  | mysql                | 3381631115268247737 |        0 |            0 | SLS_IX    |
  0 |
  +----------------------+--------------------+----------+--------------+-----------+----
  ---------+
  4 rows in set (0.00 sec)
  ```

  Each row in the preceding result is the information of a database, although the column name is table_name. `mysql/global_ddl_lock` and `mysql` are the internal databases of MySQL, not custom databases.

- Execute the following statement to query the distribution of all databases in the cluster:

  > **Note** You can query database information only by using a privileged account, but not a custom account.

  ```
  SELECT * FROM INFORMATION_SCHEMA.INNODB_CC_GLOBAL_LOCK_INFO;
  ```

  A similar result is returned:

```
+-----------+----------------------+--------------------+-----------+-------------+
| master_id | table_name           | table_id           | lock_mode | current_lsn |
+-----------+----------------------+--------------------+-----------+-------------+
|         2 | db5                  |   27980076883382398 | SLS_X     |    18866566 |
|         1 | mysql/global_ddl_lock | 1043956280258737140 | SLS_X     |    24311657 |
|         1 | db1                  |   27980076883383418 | SLS_X     |    34339564 |
|         1 | db2                  |   27980076883382651 | SLS_X     |    36087702 |
|         2 | db4                  |   27980076883383165 | SLS_X     |    18855954 |
|         1 | mysql                | 3381631115268247737 | SLS_IX    |           0 |
+-----------+----------------------+--------------------+-----------+-------------+
6 rows in set (0.00 sec)
```

Each row in the preceding result is the information of a database, although the column name is table_name. The result indicates that the databases are on the primary nodes specified by the master_id parameter. `mysql/global_ddl_lock` and `mysql` are the internal databases of MySQL, not custom databases.

Query the metadata of a database

Execute the following statement to query the information of tables on all primary nodes:

```
SELECT * FROM INFORMATION_SCHEMA.TABLES;
```

Execute the following statement to query the information of the primary node where the SQL statement is executed:

```
SELECT * FROM INFORMATION_SCHEMA.INNODB_TABLES;
```

14.Archive Database Edition

14.1. Archive Database Edition

This topic describes the benefits, architectures, and scenarios of .

Challenges and requirements for archiving historical data

- Challenges

  In most cases, new data is read or updated more frequently than historical data. Historical data such as messages or orders generated one year ago is seldom accessed. A large volume of data that is not often accessed or never accessed is stored in your database system as your business develops. This can cause the following issues:

  - Historical data and new data are stored in the same database system. This can result in insufficient disk space.
  - A large volume of data shares the memory, cache space, and disk IOPS capabilities of the database system. This can deteriorate the database performance.
  - The operation to back up a large volume of data requires a long period of time and can fail. Even if the operation is successful, the storage of the backup files is an issue that needs to be solved.

  These issues can be resolved by archiving historical data. Historical data can be stored as files by using low-cost storage services, such as Object Storage Service (OSS) or Database Backup (DBS). In real business scenarios, historical data is not completely static. Historical data generated multiple months or years ago may be queried or updated in real time or occasionally. For example, historical data such as historical orders in Taobao or Tmall, historical messages in DingTalk, and historical Cainiao logistics orders can be queried within Alibaba Group.

- Requirements

  To resolve the issues related to reads and updates of historical data, a separate database can be used as an archive database that stores only archived data. An archive database must meet the following requirements:

  - It must provide a large storage capacity to save online data that is continuously generated. This way, you do not need to worry about the storage capacity.
  - It must provide the same interfaces as your online databases. For example, the archive database must support MySQL protocols in the same manner as the online databases. This ensures that your applications can access the online databases and archive database, without the need to modify your code.
  - It must be cost-efficient. For example, you can compress data to reduce the consumed disk space and use low-cost storage media to store large volumes of data.
  - It must provide read and write capabilities that meet the requirements of low-frequency reads and writes.

MySQL fails to provide a solution that meets all of the previous requirements, though MySQL is the most widely used open source database system in the world. Engines such as TokuDB and MyRocks provide high compression ratios. However, the volume of data that can be stored by using one of these engines is limited by the disk capacity of each physical machine.

Solution: Archive Database Edition

To address the preceding challenges and meet the requirements to store archived data, provides the Edition. This edition provides features that are developed based on the following technological innovations and breakthroughs:

- This edition uses X-Engine as the storage engine. X-Engine is developed by Alibaba Cloud based on the log-structured merge-tree (LSM tree). X-Engine provides powerful data compression capabilities that allow you to use archive databases at a low cost. X-Engine uses the LSM tree and the Zstandard (ZSTD) data compression algorithm to increase the data compression ratio. Compared to InnoDB, X-Engine helps you save up to 70% storage space. For more information about X-Engine, see Introduction to X-Engine. The Edition has limits, especially in terms of the compatibility with MySQL, due to the use of X-Engine. For more information, see Limits.
- supports online expansion of the storage capacity based on shared distributed storage. PolarDB connects computing resources and storage resources over a high-speed network and transmits data by using the remote direct memory access (RDMA) protocol. This eliminates the bottleneck of I/O performance. X-Engine integrated in provides these benefits.

  X-Engine is integrated in by using the following technological innovations. This enables to run in a dual-engine architecture.

  - The write-ahead logging (WAL) log streams of X-Engine are combined with the redo log streams of InnoDB. This way, the same log streams and transmission channels are used to support InnoDB and X-Engine. The management logic and the logic of interaction with the shared storage remain unchanged. This architecture can be reused by other engines that are introduced later.
  - The I/O module of X-Engine is adapted to Polar File System (PFS) of InnoDB. This ensures that InnoDB and X-Engine use the same distributed storage. Backups are accelerated based on the underlying distributed storage.

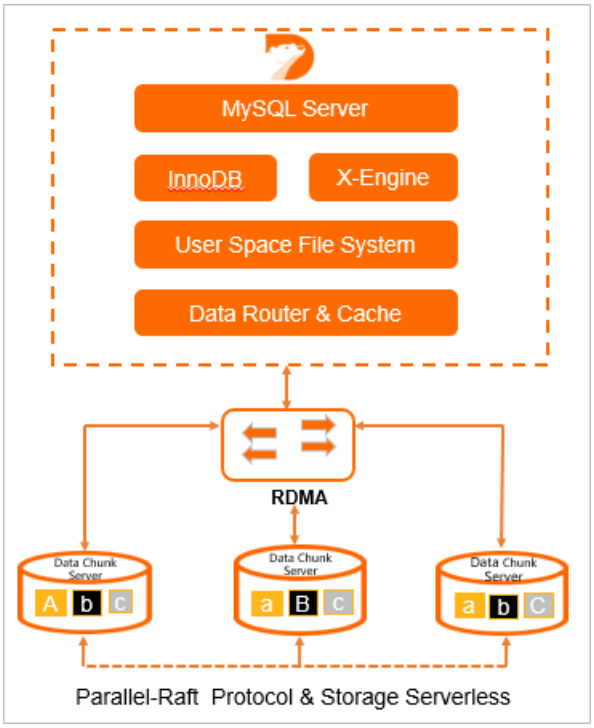
Compute node architectures of

supports and . uses the single-node architecture. An cluster provides a primary node and multiple read-only nodes. The primary node processes read and write requests, and an Archive Database cluster contains at least one read-only node. An Archive Database cluster supports the and specifications.

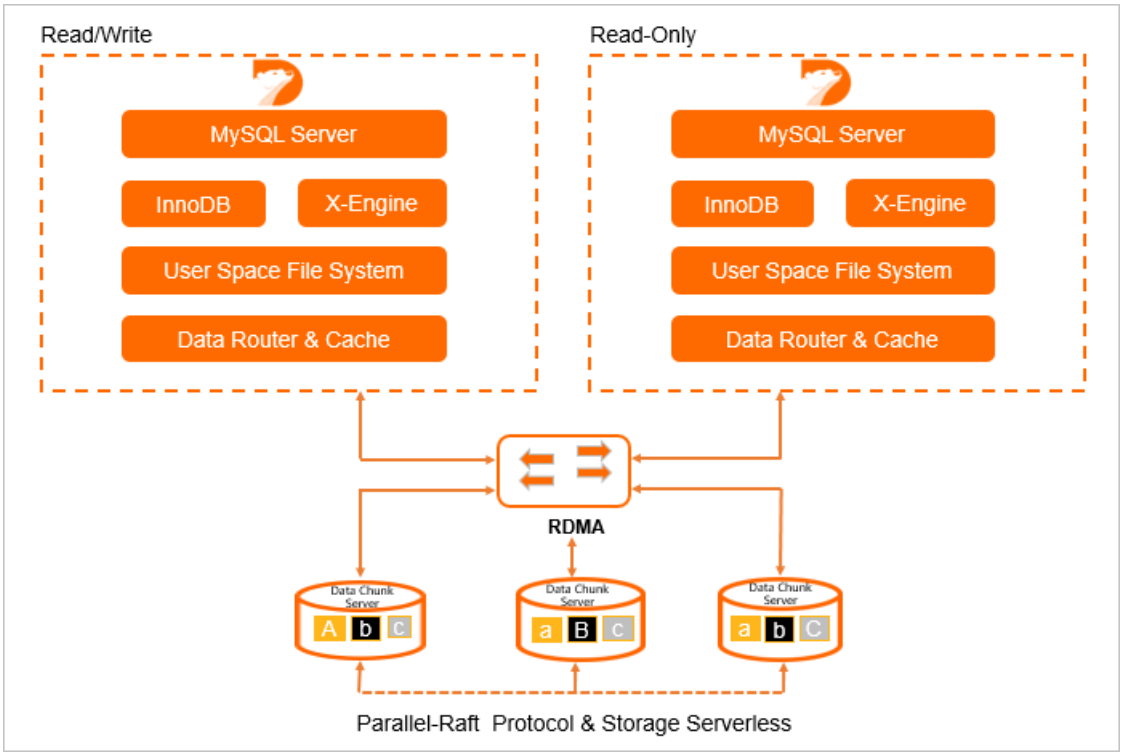
> **Note** clusters are unavailable for purchase. However, existing clusters remain available for use. You can update to . For more information, see Upgrade an Archive Database Standalone Edition cluster to an Archive Database Cluster Edition cluster.

- By default, a cluster of the Edition contains one compute node. The compute node is a dedicated node that reduces the costs incurred on PolarProxy and the overheads of synchronizing redo logs. However, in scenarios that require large storage capacity and fewer reads and writes, computing resources of the primary node cannot be used up. Therefore, the read capability that is provided by read-only nodes is unnecessary. If the specifications of the primary node and read-only nodes are the same, 50% of the computing resources are wasted. The Edition can help reduce storage costs based on the data compression capability of X-Engine. The Edition uses only the primary node to provide services. This eliminates the costs of computing resources offered by read-only nodes. A longer time is required for clusters that do not have read-only nodes to recover in disaster recovery scenarios where the primary node stops providing services. However, still ensures 99.95% availability based on the high availability capabilities that are provided by the underlying distributed storage. In most cases, business scenarios do not require high-availability services for low-frequency reads and writes. In these scenarios, data is imported to in batches in an asynchronous manner and archive databases are suitable.

Read-only nodes are not provided in the single-node architecture of . When you perform O&M operations on a node, such as restart the node after the minor version upgrade, the temporary read-only node deployed within the system is upgraded to the primary node to reduce adverse impacts on reads and writes to the . The following figure shows the single-node architecture of .



- A cluster of the Edition consists of one primary node and at least one read-only node based on shared storage. The primary node can handle read and write requests. A read-only node can handle only read requests. The multi-node architecture of inherits the advantages of as well as ensures the high availability of PolarDB clusters. When the primary node in a cluster fails, the cluster can automatically fail over to a read-only node. Then, the read-only node serves as the new primary node. This ensures that the service availability is at least 99.99%. The following figure shows the multi-node architecture of .



Benefits

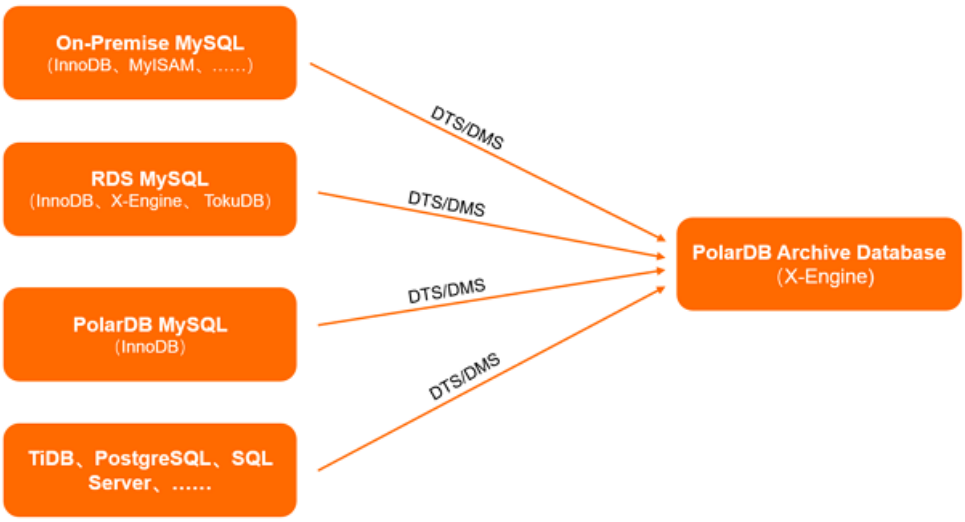
- The Archive Database Edition provides a large storage capacity. Based on the 200 TB storage capacity and the compression capability of X-Engine, a PolarDB cluster of the Archive Database Edition can store more than 500 TB raw data. The Archive Database Edition uses a serverless architecture so that the storage capacity can automatically increase as the data volume increases. This way, you do not need to specify the storage capacity when you purchase the PolarDB cluster. You are charged for the actual storage capacity that you use.
- Edition supports the official MySQL protocols. Compared to other solutions that back up historical data to NoSQL services such as HBase, the Archive Database Edition allows applications to access both online databases and archive databases without the need to modify the code.
- The Archive Database Edition uses the backup capability provided by the underlying distributed storage of to back up a large volume of data in a short period. The backup files can be uploaded to and permanently stored in low-cost storage, such as OSS.
- The multi-node architecture of uses X-Engine, which provides powerful data compression capabilities to reduce storage costs and ensure high availability of clusters. When the primary node in a cluster fails, the cluster can automatically fail over to a read-only node. Then, the read-only node serves as the new primary node. This ensures that the service availability is at least 99.99%.

Scenarios

Edition provides a large storage capacity and can be used to store the historical data of multiple services. This ensures centralized storage and management for all historical data. The Archive Database Edition is suitable for the following scenarios:

- Edition is used to store cold data of self-managed databases. The self-managed databases can be MySQL, TiDB, PostgreSQL, SQL Server, or other relational databases.
- Edition is used to store archived data for ApsaraDB RDS for MySQL or . You can migrate the historical data that is not often accessed to X-Engine. This way, the storage space of online databases can be released to reduce costs and improve performance.
- Edition is used as a relational database service that provides a large storage capacity. This is applicable to scenarios in which a large volume of data needs to be written but the data is accessed at a low frequency, such as monitoring logs.

You can use Data Transmission Service (DTS) to continuously migrate data from the online database to in real time. You can also use Data Management Service (DMS) to periodically import online data to .



Supported kernel versions

Only that runs MySQL 8.0 is supported.

Node specifications and pricing

supports and . For more information, see Specifications of compute nodes.

For more information about billing rules for and , see Billing rules of compute nodes.

FAQ

How does the Edition ensure service availability and data reliability when only one primary node is used?

The Edition is a database service that is used to store data for a specific purpose and contains only one compute node. uses new technologies such as computing scheduling within seconds and distributed multi-replica storage to ensure high service availability and high data reliability.

14.2. Usage instructions

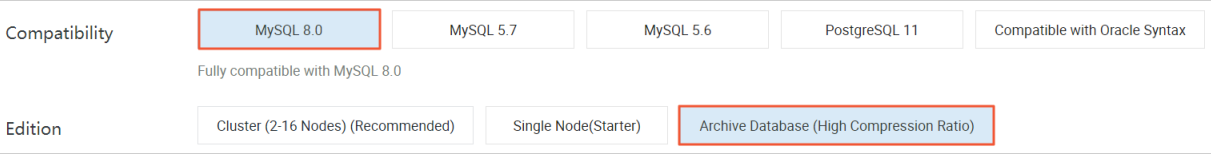
uses X-Engine as the default storage engine instead of InnoDB. Archive Database has a high compression ratio and is applicable to services that do not have high requirements for computing but need to store archived data, such as DingTalk messages. This topic describes the feature of .

Prerequisites

The is supported by only 8.0. For more information, see Archive Database.

Create a (X-Engine)

You can create a in the same way you create a cluster. You only need to set **Compatibility** to **MySQL 8.0** and set **Edition** to **Archive Database (High Compression Ratio)** on the buy page. For more information, see Purchase a pay-as-you-go cluster.



Create tables in

You can create tables in in the same way you create tables in a database that uses the InnoDB engine. The default engine for is X-Engine. If no engine is specified when you create a table, a table that uses the X-Engine engine is created. For example, run the following command to create a table with no engine specified:

```
CREATE TABLE test_arc.t1 (id int PRIMARY KEY,c1 varchar(10));
```

Run the `show create table` command to view the details of the statement:

Table	t1
Create Table	CREATE TABLE `t1` (`id` int(11) NOT NULL, `c1` varchar(10) COLLATE utf8mb4_general_ci DEFAULT NULL, PRIMARY KEY (`id`))ENGINE=XENGINE DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci

After you create the table, data is stored in X-Engine. You can use the table in the same way that you use a table in InnoDB.

> **Note**
>
> - You can also create InnoDB tables in . For example, when you use Data Transmission Service (DTS) to migrate data, the tables to be migrated may still use InnoDB. To convert tables from InnoDB to X-Engine, see Convert tables from InnoDB, TokuDB, or MyRocks to X-Engine.
> - You can run the following command to view the default engine of the current database:
>
>   ```
>   show variables like '%default_storage_engine%';
>   ```

Limits

- Limits on resource allocations if X-Engine and InnoDB are used together

When you use X-Engine, 95% of the memory is used as the write cache and block cache to speed up reads and writes. The InnoDB buffer pool does not consume much memory. We recommend that you do not use tables that use InnoDB to store a large volume of data within a cluster that uses X-Engine. Otherwise, the X-Engine performance may deteriorate due to a low cache hit ratio. We recommend that you make sure that all tables use the X-Engine engine when you use . Otherwise, the performance may deteriorate.

- Limits on engine features

  The following table describes the limits on X-Engine.

Category	Feature	Description
SQL features	Foreign key	Not supported.
	Temporary table	Not supported.
	Partition table	Not supported. X-Engine does not support the creation, addition, deletion, modification, or query of partitions.
	Generated Column	Not supported.
	Handler API	Not supported.
Column properties	Maximum column size (LONGBLOB/LONGTEXT/JSON)	32MB
	GIS data type	Not supported. X-Engine does not support the following GIS data types: GEOMETRY, POINT, LINESTRING, POLYGON, MULTIPOINT, MULTILINESTRING, MULTIPOLYGON, and GEOMETRYCOLLECTION.
Index	Hash index	Not supported.
	Spatial index	Not supported. X-Engine does not support creating or using full-text indexes.
Transaction	Transaction isolation level	Supports the following two isolation levels: ◦ Read Committed (RC) ◦ Repeatable Read (RR)
	The maximum data volume supported by a transaction	32 MB
	Savepoint	Not supported.
	XA transaction	Available soon.

Category	Feature	Description
Lock	Lock granularity	○ Supports table-level and row-level locks. ○ GAP locks are not supported.
	Skip Locked	Not supported.
	Lock Nowait	Not supported.
Character set	Character sets supported by non-indexed columns	Supported.
	Character sets supported by indexed columns	○ Latin1 (latin1_bin) ○ GBK (gbk_chinese_ci and gbk_bin) ○ UTF-8 (utf8_general_ci and utf8_bin) ○ UTF-8MB4 (utf8mb4_0900_ai_ci, utf8mb4_general_ci, and utf8mb4_bin)
Primary/secondary replication	Binary log formats	Supports the following formats: ○ stmt ○ row ○ mixed **Note** The default binary log format is the row format. The stmt and mixed formats may cause data security issues in specific concurrency scenarios.

Note By default, the features that are not listed in X-Engine are the same as those in the InnoDB.

- Limits on large transactions

  X-Engine does not support large transactions. When the number of rows modified in a transaction is equal to or greater than 10,000, X-Engine enables the `commit in middle` feature. This way, X-Engine internally commits the transaction and starts a sub-transaction to continue to perform the transaction. However, the `commit in middle` feature does not follow the atomicity of transactions in a strict sense. Therefore, note the following limits:

- Assume that you want to start a transaction to insert a large amount of data. During the insertion, a part of the data has been submitted due to the `commit in middle` feature. Then, the inserted data can be queried by other requests.
- Assume that you want to start a transaction to modify more than 10,000 rows of data. The transaction on which X-Engine enables `commit in middle` cannot be rolled back.

```
drop table t1;
create table t1(c1 int primary key , c2 int)ENGINE=xengine;
begin;
call insert_data(12000); // 12,000 rows are inserted and a commit in middle operation i
s triggered. As a result, the first 10,000 rows of data are committed.
rollback;// Only the last 2,000 rows can be rolled back.
select count(*) from t1; // The committed 10,000 rows of data can be queried.
+----------+
| count(*) |
+----------+
|    10000 |
+----------+
1 row in set (0.00 sec)
```

- Assume that you want to start a transaction to modify and delete a large amount of data. The DELETE operation cannot read the inserted rows in this transaction due to the `commit in middle` feature. Therefore, the newly inserted data cannot be deleted.

```
drop table t1;
create table t1(c1 int primary key , c2 int)ENGINE=xengine;
call insert_data(10000);
begin;
insert into t1 values(10001,10001), (10002,10002);
delete from t1 where c1 >= 0;// The deletion triggers a commit in middle operation, and
the two rows of data inserted by the current transaction are not deleted.
commit;
select * from t1;
+-------+-------+
| c1    | c2    |
+-------+-------+
| 10001 | 10001 |
| 10002 | 10002 |
+-------+-------+
2 rows in set (0.00 sec)
```

Parameters

Note

- You can modify the table parameters based on your business requirements. For more information, see Specify cluster and node parameters.
- All parameters in the table have been added the MySQL configuration file compatibility prefix `loose_` in the console.

Category	Parameter	Description	Modifiable	Restart after the parameter modification
Performance	xengine_batch_group_max_group_size	The maximum number of groups in a transaction pipeline.	No	N/A
	xengine_batch_group_max_leader_wait_time_us	The maximum pending time of a transaction pipeline.	No	N/A
	xengine_batch_group_slot_array_size	The maximum batch size of a transaction pipeline.	No	N/A
	xengine_parallel_read_threads	The number of parallel read threads.	Yes	No
	xengine_parallel_wal_recovery	Parallel recovery	No	N/A
Memory	xengine_block_cache_size	The size of the read block cache.	No	N/A
	xengine_row_cache_size	The size of the row cache.	No	N/A
	xengine_write_buffer_size	The maximum size of a memory table.	No	N/A
	xengine_block_size	The size of the data block on a disk.	No	N/A
	xengine_db_write_buffer_size	The maximum size of the active memory tables in all subtables.	No	N/A
	xengine_db_total_write_buffer_size	The maximum size of the active memory tables and immutable memory tables in all subtables.	No	N/A
	xengine_scan_add_blocks_limit	The number of blocks that can be added to the block cache during each range-based scan request.	Yes	No
compaction	xengine_flush_delete_percent_trigger	If the number of records in a memory table exceeds the value of this parameter, the xengine_flush_delete_record_trigger parameter takes effect.	No	N/A
Lock	xengine_max_row_locks	The maximum number of rows that can be locked in a single SQL request.	No	N/A
	xengine_lock_wait_timeout	The timeout period of a lock wait.	Yes	No

15.Global Database Networks

15.1. Overview

A global database network (GDN) consists of multiple clusters that are deployed in multiple regions across the globe. This topic describes GDNs and their features.

All clusters in a GDN handle read and write requests, and replicate data between each other. This allows GDN to also provide geo-disaster recovery. GDN is ideal for the following scenarios:

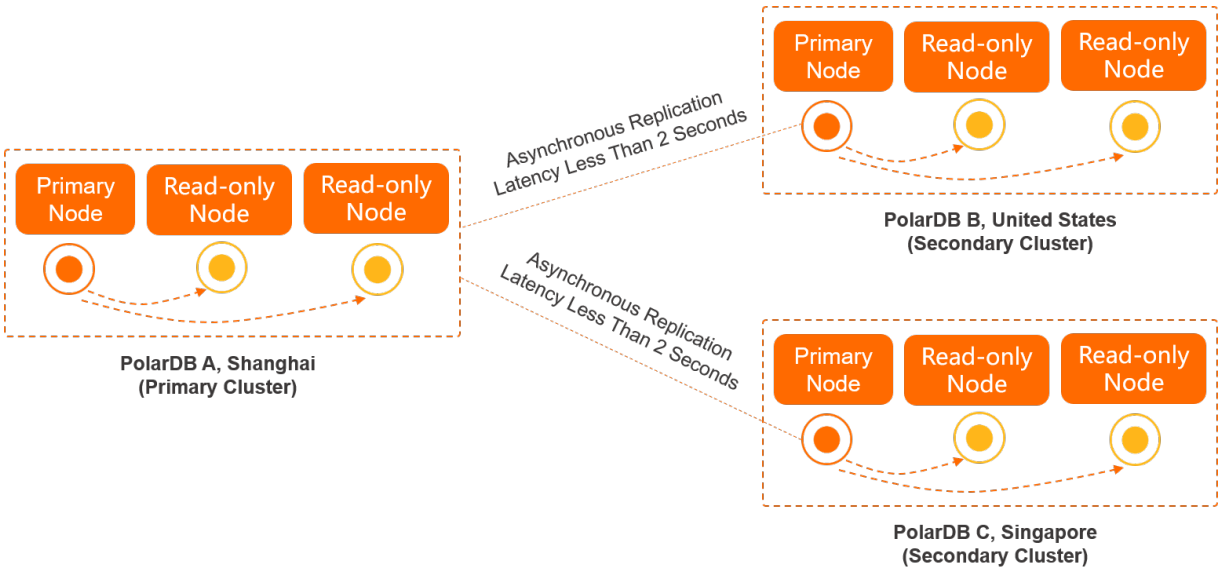
- Active geo-redundancy

  If you deploy applications in multiple regions but deploy only databases in the primary region, applications that are not deployed in the primary region must communicate with the databases, which may be located in a geographically distant region. This results in high latency and poor performance. GDN replicates data across regions with low latency and provides cross-region read/write splitting. GDN allows applications to read data from a database local to the region. This shortens the time required to access the database to less than 2 seconds.

- Geo-disaster recovery

  GDN supports geo-disaster recovery regardless of whether your applications are deployed in one or more regions. If a fault occurs in the region where the primary cluster is deployed, you need to manually switch your service to a secondary cluster.

  > **Note** A typical failover can be completed in 10 minutes. In practice, most failovers can be completed within 5 minutes. During the failover, services may be interrupted for up to 60 seconds. We recommend that you perform the switchover during off-peak hours and make sure that your applications are configured to automatically reconnect to the database service.



Request routing

The read and write requests on each cluster in a GDN are routed based on the cluster endpoint configuration. Imagine a scenario where you have three GDN clusters, of which Cluster 1 is the primary cluster and Cluster 2 and Cluster 3 are secondary clusters. If the endpoint of Cluster 2 can handle read and write requests but is configured to allow read requests to the primary cluster, read requests are routed to the primary node of the primary cluster. This results in a higher request latency. If the endpoint of Cluster 3 is configured to handle only read requests, read requests are routed only to the read-only nodes of Cluster 3. For more information about how to configure the endpoint of a cluster, see Configure PolarProxy.

> **Note** If the endpoint of a secondary cluster is configured to handle both read and write requests, write requests and other broadcast requests (such as SET statements) are routed to the primary node of the primary cluster. If **session consistency** is enabled for the secondary cluster, read requests on the cluster may be routed to the primary node of the primary cluster.

Advantages

- Zero code modification for deployment: If an application is deployed in one region, you can deploy it in multiple regions without the need to modify code. For more information, see Cross-region deployment.
- Cross-region read/write splitting: GDN clusters can handle both read and write requests. Read requests are sent to the cluster in the same region while write requests are forwarded to the primary cluster. For more information, see Cross-region read/write splitting.
- Flexible configuration: The primary and secondary clusters can be configured separately. The configuration of a cluster includes cluster specifications, whitelists, and parameter values. For more information, see Create a GDN.
- Data of clusters in different regions can be replicated with low latency. Physical replication is performed over multiple channels, which allows data to be replicated across all nodes with a latency of less than 2 seconds even under heavy load. For more information, see Low-latency synchronization across regions.

Billing

You are not charged for the traffic that is generated during cross-region data transmission within a GDN. You are charged only for the use of clusters in the GDN. For more information about the pricing rules of clusters, see Billable items.

Supported regions and clusters

- Regions: GDN is available in more than 10 regions, including regions inside the Chinese mainland, the China (Hong Kong) region, and regions outside China.
- A cluster in a GDN must use one of the following versions:
  - A 8.0 cluster
  - A 5.7 cluster whose engine minor version is 5.7.1.0.13 or later
  - A 5.6 cluster whose engine minor version is 5.6.1.0.27 or later
- The primary cluster and secondary clusters must have the same database engine version, that is MySQL 8.0, MySQL 5.7, or MySQL 5.6.

Get started with GDN

For more information, see Create and release a GDN.

Related videos

GDN

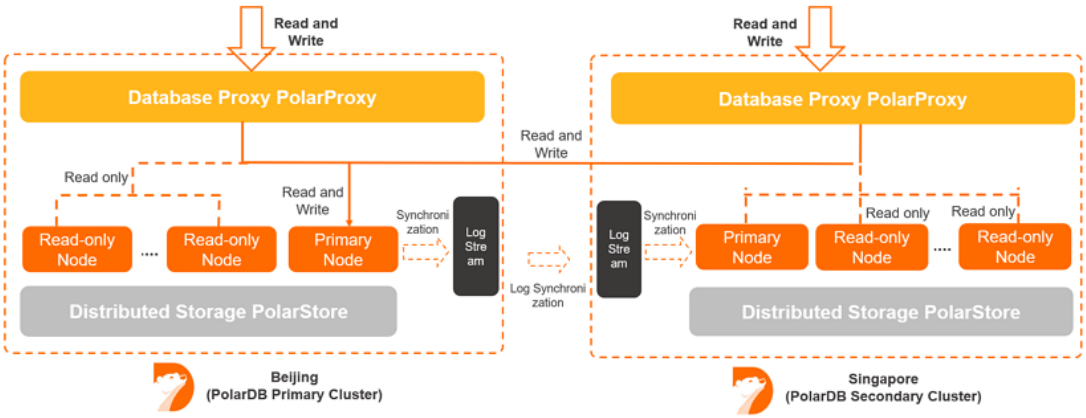
References

- Technical architecture
- Typical scenarios
- Create and release a GDN
- Add and remove secondary clusters
- Connect to a GDN

15.2. Technical architecture

This topic describes the technical architecture of global database networks (GDNs).

GDN architecture



Cross-region deployment

- A GDN consists of one primary cluster and multiple secondary clusters. Data is synchronized among clusters in each GDN.

  > **Note** A GDN can contain one primary cluster and up to four secondary clusters. To add more secondary clusters, for technical support.

- By default, each cluster in a GDN contains two nodes. You can add up to 16 nodes. For more information, see Add a read-only node.

Low-latency synchronization across regions

GDNs use an asynchronous replication mechanism to replicate data across regions. GDNs also reduce the latency of cross-region replication between the primary and secondary clusters by using technologies, such as physical logs and parallel processing. Data is synchronized between clusters and the network latency is limited to less than 2 seconds. This way, read requests from applications in non-central regions can be processed with the minimum latency. If you create cross-region secondary clusters and synchronize data, the stability and performance of the current primary cluster are not affected.

> **Note** When you create a secondary cluster, we recommend that you select the same node specification as the primary cluster. This ensures low-latency synchronization. For information about how to create a secondary cluster, see Add a secondary cluster.

The following table describes the test results of the low-latency synchronization from the US (Silicon Valley) region to the China (Zhangjiakou) region in a GDN. These regions are used in the example.

Specification and topology of the test clusters	Sysbench stress testing	Peak QPS/TPS	Synchronization latency from the secondary cluster in the US (Silicon Valley) region to the primary cluster in the China (Zhangjiakou) region
GDN that covers the China (Zhangjiakou) region and the US (Silicon Valley) region 16-Core 128 GB	OLTP_INSERT	82655/82655	Less than 1s
	OLTP_WRITE_ONLY	157953/26325	Less than 1s
	OLTP_READ_WRITE	136758/6837	Less than 1s

Cross-region read/write splitting

- Key features
  - Each cluster is in read and write mode.
  - In most cases, read requests are forwarded to the secondary cluster in the same region. Write requests are forwarded to the primary cluster.

    > **Note** The primary node in the secondary cluster is used to asynchronously replicate data from the primary cluster. By default, read requests are sent to the read-only nodes in the same region to reduce the latency of physical replication across regions.

  - You do not need to modify the code of your applications to achieve read/write splitting.
- How to implement read/write splitting

  The cross-region read/write splitting feature of a GDN must be implemented based on the cluster endpoints of clusters. For more information about how to manage a cluster endpoint for a GDN, see Connect to a GDN.
- Forwarding rules

Node	Forwarded request

Node	Forwarded request
Only the primary node	○ All data manipulation language (DML) operations, such as INSERT, UPDATE, DELETE, and SELECT FOR UPDATE operations. ○ All data definition language (DDL) operations, such as creating databases or tables, deleting databases or tables, and changing table schemas or permissions. ○ All requests in transactions. ○ Queries by using user-defined functions. ○ Queries by using stored procedures. ○ EXECUTE statements. ○ Multi-statements. ○ Requests that involve temporary tables. ○ SELECT last_insert_id(). ○ All requests to query or modify user variables. ○ SHOW PROCESSLIST statements. ○ KILL statements in Structured Query Language (SQL) statements (not KILL commands in Linux).
The primary node or read-only nodes **Note** Requests are sent to the primary node only if the **Primary Node Accepts Read Requests** feature is disabled.	○ Non-transactional read requests. ○ COM_STMT_EXECUTE commands.
All nodes	○ All requests to modify system variables. ○ USE statements. ○ COM_STMT_PREPARE commands. ○ COM_CHANGE_USER, COM_QUIT, and COM_SET_OPTION, and other commands.

Note The primary node in the secondary cluster is used to asynchronously replicate data from the primary cluster, and does not process read and write requests. Therefore, the primary node in the table refers to the primary node in the primary cluster, and read-only nodes refer to the read-only nodes in the secondary cluster.

References

- Overview
- Typical scenarios
- Create and release a GDN
- Add and remove secondary clusters

15.3. Typical scenarios

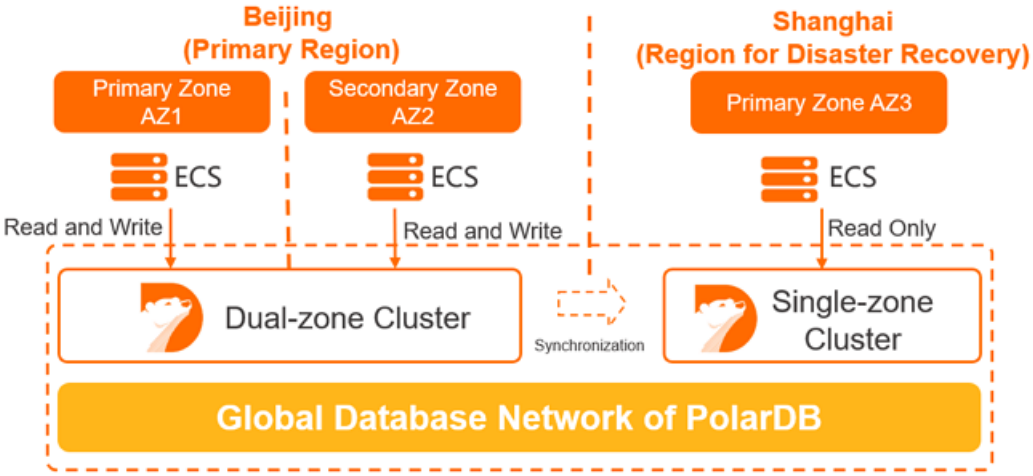
Geo-disaster recovery and cross-region deployment are typical scenarios in which global database networks (GDNs) are used. This topic also describes the service architectures and how to deploy GDNs in these scenarios.

Geo-disaster recovery

The geo-disaster recovery feature allows you to achieve high availability across regions. This enhances data security and improves service availability. If a data center breakdown occurs, services can be rapidly recovered. Architectures can be implemented, such as three data centers across two zones, four data centers across two zones, and six data centers across three zones.

- Typical industries include banking, securities, insurance, and fintech.
- The following example shows the service architecture of three data centers across two zones:

  The databases are deployed in two clusters:



  - The cluster in the China (Beijing) region is deployed in two zones: AZ 1 and AZ 2.
  - The cluster in the China (Shanghai) region is deployed in a single zone.

  The application is deployed in the China (Beijing) region and performs local read and write operations on the database in AZ 1.

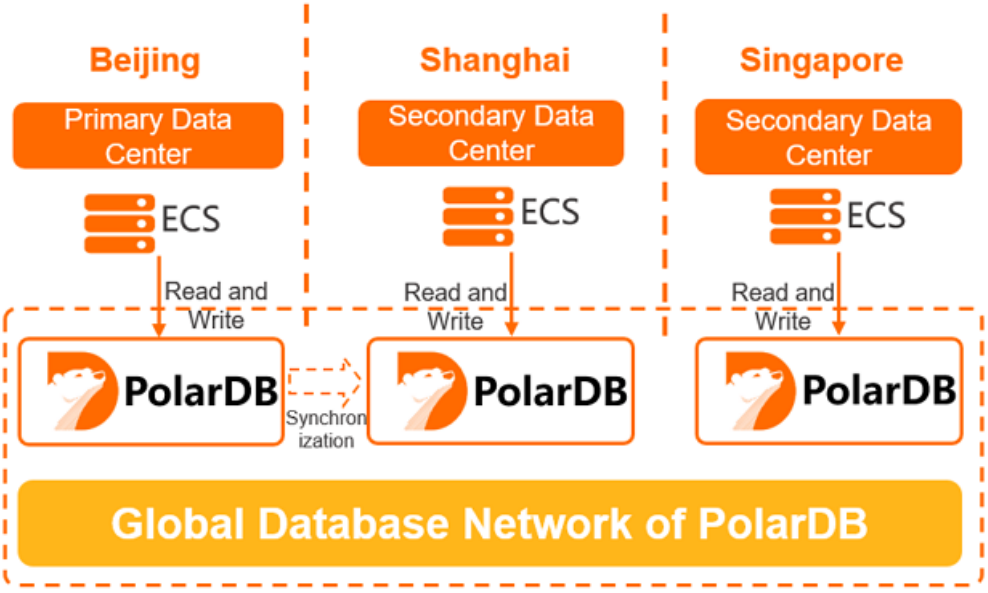
  - If AZ 1 in the China (Beijing) region fails, the service is preferentially switched to AZ 2 in the China (Beijing) region.
  - If AZ 1 and AZ 2 in the China (Beijing) region fail, the service is switched to AZ 3 in the China (Shanghai) region.

Cross-region deployment (active geo-redundancy)

In some cases, services of enterprises are deployed across the country or on a global scale. Therefore, data must be synchronized to achieve cross-region reads and writes. The database can be accessed from global regions. However, read and write requests are sent to the clusters that are deployed in the same region as the application.

- Applicable industries: gaming, cross-border e-commerce, local services (takeout), and new retail (outlets).
- The following example shows the service architecture:

- The applications in each region read data from and write data to the database that is deployed in the same region. This ensures optimal performance.
- You need to configure only one connection string for an application. You can expand your services cross regions, from one data center to two, three, or even more data centers, without modifying the code.
- The cluster specifications in the China (Shanghai) and the Singapore (Singapore) regions do not need to be consistent with the specifications in the China (Beijing) region. You can select the specifications based on your requirements.



Procedure

1. Create a GDN and select an existing cluster as the primary cluster in the GDN. For more information, see Create a GDN.
2. In the GDN, add a secondary cluster. For more information, see Add a secondary cluster.
3. Connect to the GDN. For more information, see Connect to a GDN.

For more information about the best practices for deploying GDNs across regions, see Best practices for deploying a GDN across regions.

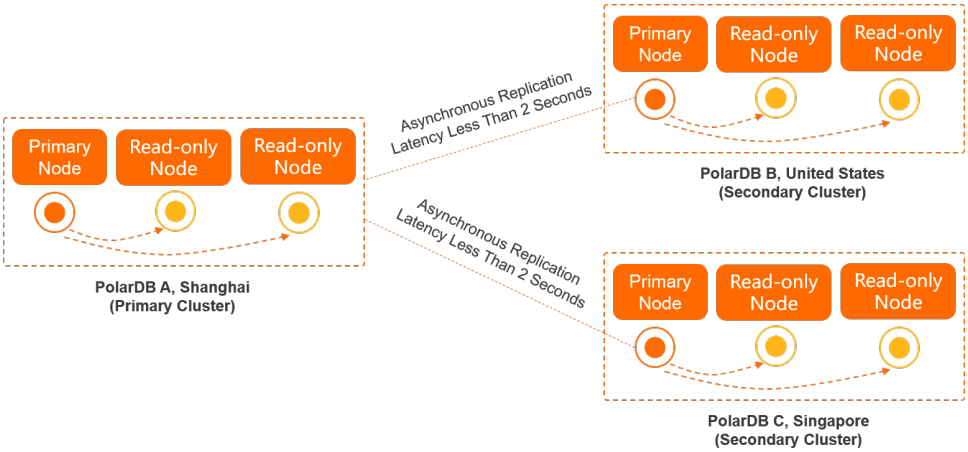
15.4. Best practices for deploying a GDN across regions

This describes how to deploy a Global Database Network (GDN) across regions.

Deployment plan

- Architecture

  A gaming company provides services in Beijing and plans to provide services in Shanghai and Singapore. Data needs to be shared across regions and all regions of the world can connect to the same database. Read and write requests are sent to the clusters that is deployed in the same region as the clients.



- Procedure
  i. Create a GDN.
  ii. Add a secondary cluster.
  iii. Connect to a GDN.
  iv. (Optional) Purchase a storage plan.

Create a GDN

After you create a GDN, it contains a primary cluster. For more information, see Create a GDN.

> **Note** The cross-region data transfer within a GDN is free of charge. You are charged for only clusters in the GDN. For more information about the pricing of clusters, see Billable items.

Add a secondary cluster

Add a secondary cluster to the GDN to form a global database network. For more information, see Add a secondary cluster.

> **Note**
> - When you create a secondary cluster, we recommend that you set the same node specifications as those of the primary cluster to ensure low-latency synchronization. You can specify the number of read-only nodes based on read requests to the secondary cluster that is deployed in the same region as the client.
> - GDN supports low-latency synchronization across regions. Secondary clusters automatically synchronize data from the primary cluster. You do not need to purchase additional services, such as Data Transmission Service (DTS).

Connect to a GDN

After the GDN is created, you can use the cluster endpoint to connect to and manage the GDN. For more information, see Connect to a GDN.

(Optional) Purchase a storage plan

You are eligible for storage plans when you use the following resources:

- Cluster storage

  The storage of each cluster is charged based on the volume and storage duration of data. If you require a large storage capacity, such as 1,000 GB or more, we recommend that you use storage plans to reduce costs. Storage fees are offset at a ratio of 1:1. This means that a 1-GB storage plan can be used to offset the fees of 1 GB storage for level-1 backups.

- Level-1 backups that exceed the free quota

  Storage plans are applied to all clusters that belong to your account. The remaining capacity of a storage plan is automatically used to offset the level-1 backup storage space that exceeds the free quota at a ratio of 1:1.6. In this case, every 1 GB of the storage plan is used to offset 1.6 GB of level-1 backup storage. If the remaining capacity of the storage plan is insufficient to offset the level-1 backup storage space, you are charged for additional storage space on a pay-as-you-go basis.

For more information, see Purchase a storage plan.

> **Note**
>
> - You can select the following types of storage plans: storage plans that apply to regions inside **mainland China**, and storage plans that apply to **China (Hong Kong) and regions outside China**. You can purchase only one storage plan for each type.
> - A storage plan can be used by all the clusters in the regions that are specified by the **Plan Type** parameter. You can select **Mainland China** or **China (Hong Kong) and regions outside China**. For more information, see Scale-out and scale-in.
> - If the capacity of your storage plan is insufficient, you can upgrade the storage plan. However, storage plans cannot be downgraded.
> - You are charged for the storage that is not covered by the storage plan on a pay-as-you-go basis.
>
> For example, each cluster in the GDN requires 300 GB of storage. You can purchase a 500 GB of **mainland China** storage plan to deduct the storage of the primary cluster in the China (Beijing) region and the secondary cluster in the China (Shanghai) region. Then, you can purchase a 300 GB of **China (Hong Kong) and regions outside China** storage plan to deduct the storage of the secondary cluster in Singapore. The 100 GB of storage that exceeds the capacity of the storage plan is charged on a pay-as-you-go basis.

15.5. Create and release a GDN

A global database network (GDN) consists of multiple clusters that are deployed in multiple regions across the globe. Data is replicated across all clusters in each GDN. This allows applications that are deployed across different regions to access a database that is local to their region, which provides reliable access with low latency. This topic describes how to create and release a GDN.

Prerequisites

A primary cluster is created. For more information, see Purchase a pay-as-you-go cluster and Purchase a subscription cluster.

Limits

- A cluster in a GDN must use one of the following versions:
  - A 8.0 cluster

  - A 5.7 cluster whose engine minor version is 5.7.1.0.13 or later
  - A 5.6 cluster whose engine minor version is 5.6.1.0.27 or later
- The primary cluster and secondary clusters must have the same database engine version, that is MySQL 8.0, MySQL 5.7, or MySQL 5.6.
- A GDN consists of a single primary cluster and up to four secondary clusters. Multiple clusters can be deployed within the same zone of a region.

  > **Note** To add more secondary clusters, to contact technical support.

- A cluster belongs to only one GDN.
- Clusters in the GDN do not support specifications of 2 cores and 4 GB of memory or 2 cores and 8 GB of memory.
- Clusters in the GDN do not support the database and table restoration feature.

Billing

You are not charged for the traffic that is generated during cross-region data transmission within a GDN. You are charged only for the use of clusters in the GDN. For more information about the pricing rules of clusters, see Billable items.

Create a GDN

1.
2. In the left-side navigation pane, click **Global Database Network**.
3. On the **Global Database Network** page, click **Create GDN**.

4. In the **Create GDN** dialog box, specify the following parameters.

Parameter	Description
Name	The name of the GDN that you want to create. We recommend that you set a descriptive name that makes it easy to identify. GDN names do not have to be unique.
Primary Region	The region where the primary cluster is deployed. **Note** Select the region where the primary cluster is deployed.
Primary Cluster	Select an existing cluster as the primary cluster of the GDN.

5. After you specify the preceding parameters, click **OK**.

> **Note** After you create a GDN, you can add secondary clusters. For more information, see Add and remove secondary clusters.

Release a GDN

1. Log on to the PolarDB console.
2. In the left-side navigation pane, click **Global Database Network**.
3. Find the GDN that you want to release and click **Delete** in the **Actions** column.

   > Notice
   > - A GDN can be released if it contains only the primary cluster.
   > - A GDN cannot be restored after it is released. Proceed with caution.
   > - Applications that are connected to the endpoints of a GDN cannot access databases after the GDN is released. Modify the application code to change the connection string of the GDN at your earliest opportunity.

4. In the **Delete GDN** dialog box, click **OK**.

FAQ

- How many GDNs can I create for an Alibaba Cloud account?

  The number of GDNs that you can create is unlimited.

- After a GDN is created, can I change the primary cluster of the GDN?

  No, you cannot change the primary cluster after the GDN is created. You can create another GDN, select the required cluster as the primary cluster, and then release the current GDN.

  > Note Before you release the GDN, remove all secondary clusters from the GDN.

15.6. Add and remove secondary clusters

A global database network (GDN) consists of multiple clusters that are deployed in multiple regions across the globe. Data is replicated across all clusters in each GDN. This allows applications that are deployed across different regions to access a database that is local to their region, which provides reliable access with low latency. This topic describes how to add and remove secondary clusters.

Prerequisites

Create a GDN

Precautions

- A cluster in a GDN must use one of the following versions:
  - A 8.0 cluster
  - A 5.7 cluster whose engine minor version is 5.7.1.0.13 or later
  - A 5.6 cluster whose engine minor version is 5.6.1.0.27 or later
- The primary cluster and secondary clusters must have the same database engine version, that is MySQL 8.0, MySQL 5.7, or MySQL 5.6.

- You can create only secondary clusters. You cannot specify existing clusters as secondary clusters.
- When you create a secondary cluster, we recommend that you set the same node specifications as those of the primary cluster to ensure data replication with low latency. You can specify the number of read-only nodes based on read requests to the secondary cluster that is deployed in the same region as the client.
- A GDN consists of a single primary cluster and up to four secondary clusters. Multiple clusters can be deployed within the same zone of a region.

  > **Note** To add more secondary clusters, to contact technical support.

- A cluster belongs to only one GDN.
- Clusters in the GDN do not support specifications of 2 cores and 4 GB of memory or 2 cores and 8 GB of memory.

Billing

You are not charged for the traffic that is generated during cross-region data transmission within a GDN. You are charged only for the use of clusters in the GDN. For more information about the pricing rules of clusters, see Billable items.

Add a secondary cluster

1.
2. In the left-side navigation pane, click **Global Database Network**.
3. Find the GDN to which you want to add a secondary cluster and click **Add Secondary Cluster** in the **Actions** column.

> **Note** You cannot specify existing clusters as secondary clusters.

4. On the buy page, select **Subscription** or **Pay-As-You-Go**.
5. Configure the parameters described in the following table.

Parameter	Description
Region	The region where you want to create a cluster. You cannot change the region after the cluster is created. **Note** Make sure that the cluster and the ECS instance to which you want to connect are deployed in the same region. Otherwise, the cluster and the ECS instance can communicate only over the Internet, which results in decreased cluster performance.
Create Type	The type of cluster to be created. Select **Create Secondary Cluster**.

Parameter	Description
GDN	The GDN in which you want to create a secondary cluster. **Note** By default, the GDN that you select before you create the secondary cluster is used.
Primary Availability Zone	The primary zone where the cluster is deployed. ○ A zone is an independent geographical location in a region. All of the zones in a region provide the same level of service performance. ○ You can deploy your cluster and ECS instance in the same zone or in different zones. ○ You need to specify only the primary zone. The system automatically selects a secondary zone.
Network Type	This parameter can be set only to **VPC**. You do not need to change this parameter value. **Note** Before you use the classic network, you must select a virtual private cloud (VPC). After the cluster is created, configure the classic network. For more information, see Cluster endpoints and primary endpoints.
VPC **VSwitch**	Make sure that the cluster and the ECS instance to which you want to connect are deployed in the same VPC. Otherwise, the cluster and the ECS instance cannot communicate over a VPC, which results in decreased cluster performance. ○ If you have an existing VPC that meets your network requirements, select the VPC. For example, if you have created an ECS instance and the VPC to which the ECS instance is connected meets your network requirements, select this VPC. ○ Otherwise, use the default VPC and the default vSwitch. ■ Default VPC: ■ Only one VPC is specified as the default VPC in the region that you select. ■ The default VPC uses a 16-bit subnet mask. For example, the CIDR block 172.31.0.0/16 provides up to 65,536 internal IP addresses. ■ The default VPC does not consume the quota of the VPCs that you can create on Alibaba Cloud. ■ Default vSwitch: ■ Only one vSwitch is specified as the default vSwitch in the zone that you select. ■ The default VPC uses a 20-bit subnet mask. For example, the CIDR block 172.16.0.0/20 provides up to 4,096 internal IP addresses. ■ The default vSwitch does not consume the quota of the vSwitches that you can create in a VPC. ○ If the default VPC and vSwitch cannot meet your business requirements, you can create your own VPC and vSwitch. For more information, see Create and manage a VPC.

Parameter	Description
Compatibility	**MySQL 8.0**, **MySQL 5.7**, and **MySQL 5.6** are supported. The value of this parameter must be specified the same as the compatibility of the primary cluster.
Edition	This parameter can only be set to **Cluster (2-16 Nodes) (Recommended)**. You do not need to change this parameter value.
Node Specification	Specify the node specification based on your business requirements. For more information, see Specifications of compute nodes.
Nodes	By default, each **Cluster (2-16 Nodes) (Recommended)** cluster consists of one primary node and one read-only node. Both of the nodes have the same specifications. Keep the default setting. **Note** If the primary node fails, the system upgrades the read-only node to a primary node and creates another read-only node. For more information about read-only nodes, see Architecture.
Storage Cost	The storage cost. You do not need to change this parameter value. You are charged by hour for the actual volume of storage space that is consumed. For more information, see Billable items. **Note** You do not need to specify the storage capacity when you create a cluster. The system scales the storage capacity when the amount of data is increased or decreased.
Time Zone	The time zone of the cluster. The default value is **UTC+08:00**.
Table Name Case Sensitivity	Specifies whether table names are case-sensitive. The default value is **Not Case-sensitive**. If the table names of your on-premises database are case-sensitive, we recommend that you select Case-sensitive. This ensures that data is migrated smoothly. **Note** After the cluster is created, you cannot change the value of this parameter. We recommend that you configure this parameter based on your business requirements.

Parameter	Description
Release Cluster	The backup retention policy that is used when the cluster is deleted or released. The default value is **Retain Last Automatic Backup (Automatic Backup before Release) (Default)**. ○ **Retain Last Automatic Backup (Automatic Backup before Release) (Default)**: The system retains the last backup when you release the cluster. ○ **Retain All Backups**: The system retains all backups when you release the cluster. ○ **Delete All Backups (Cannot be restored)**: The system retains no backups when you release the cluster. **Note** You may be charged for the backups that are retained after you delete or release a cluster. For more information, see Release a cluster.
Cluster Name	○ The name of the new cluster. It must be 2 to 128 characters in length and can contain letters, digits, periods (.), underscores (_), and hyphens (-). It must start with a letter. ○ If you leave this parameter empty, the system generates a cluster name. You can change the cluster name after the cluster is created.
Resource Group	Select a resource group from available resource groups. For more information, see Create a resource group. **Note** A resource group is a group of resources that belong to an Alibaba Cloud account. Resource groups allow you to manage these resources in a centralized manner. A resource belongs to only one resource group. For more information, see Use RAM to create and authorize resource groups.

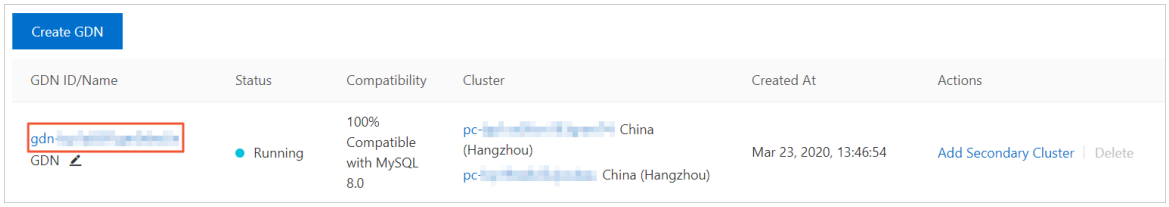
6. If you create a **subscription** cluster, set **Purchase Plan** and **Number** and click **Buy Now** on the right side.
7. On the **Confirm Order** page, confirm your order information. Read and accept the terms of service.
   - If **Product Type** is set to **Pay-As-You-Go**, click **Activate Now**.
   - If **Product Type** is set to **Subscription**, click **Pay**. On the **Purchase** page, confirm the order information and payment method, and click **Purchase**.

   After you complete the payment, it requires 10 to 15 minutes to create the cluster. Then, the newly created cluster is displayed on the **Clusters** page.

> **Note**
>
> - If nodes in your cluster are in the **Creating** state, the cluster is being created and unavailable. The cluster is available only when it is in the **Running** state.
> - Make sure that you have selected the region where the cluster is deployed. Otherwise, you cannot view the cluster.
> - We recommend that you purchase storage plans if you want to store a large volume of data. Storage plans are more cost-effective than pay-as-you-go storage. Larger storage plans provide more storage for lower costs. For more information, see Combination with storage plans.

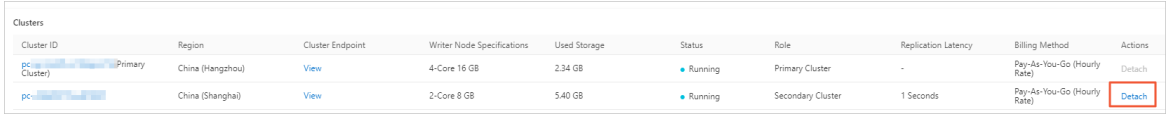
Remove a secondary cluster

1.
2. In the left-side navigation pane, click **Global Database Network**.
3. Find the GDN from which you want to remove a secondary cluster and click **GDN ID/Name**.



4. In the **Clusters** section, find the **Secondary Cluster** that you want to remove and click **Detach** in the **Actions** column.

> **Note**
>
> - It requires about 5 minutes to remove a secondary cluster.
> - During this process, the endpoints of all clusters in the GDN (including the secondary cluster that is being removed) are available. You can still use the endpoints to access databases.
> - Only secondary clusters can be removed from a GDN. The primary cluster cannot be removed from a GDN.
> - After a secondary cluster is removed from a GDN, the secondary cluster stops replicating data from the primary cluster. In this case, the secondary cluster is set to the read/write mode.
> - After a secondary cluster is removed from a GDN, the cluster can no longer be used as a secondary cluster for the GDN. Proceed with caution.



5. In the dialog box that appears, click **OK**.

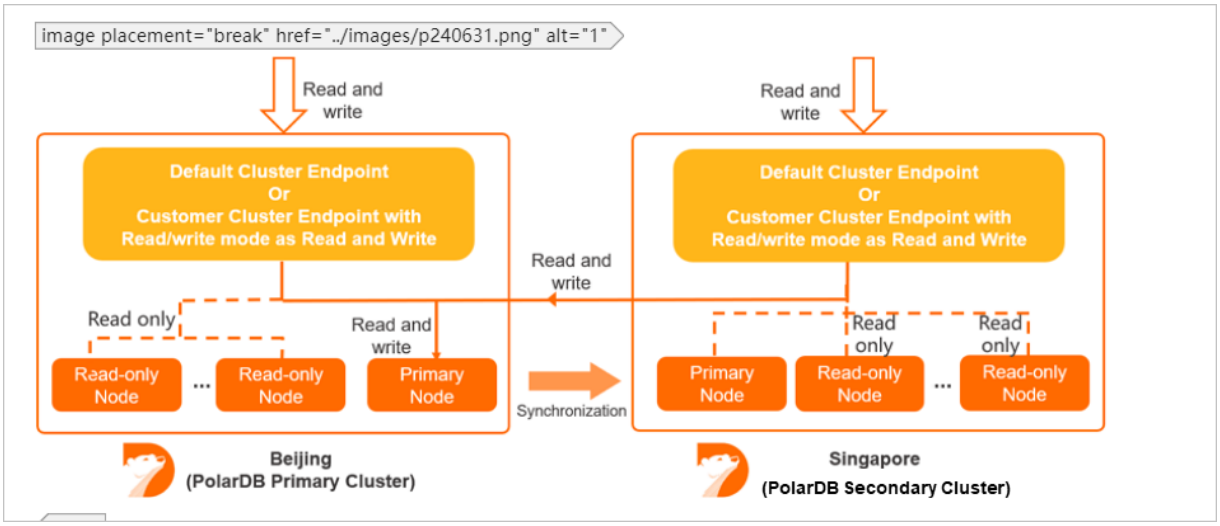
15.7. Connect to a GDN

A global database network (GDN) consists of multiple clusters that are distributed across regions around the world. This topic describes how to view cluster endpoints of a GDN and how to connect to a GDN.

Endpoints of a GDN

A GDN does not provide an endpoint. However, each cluster in the GDN provides a separate cluster endpoint. A GDN consists of the primary cluster and secondary clusters. Applications in each region use the endpoint of the cluster that is deployed in the same region to connect to the GDN.

Data is synchronized from the primary cluster to all secondary clusters in a GDN. In most cases, read requests are forwarded to the secondary cluster in the same region. Write requests are forwarded to the primary cluster. For more information about the read/write splitting feature of GDN, see Cross-region read/write splitting.



Only the following cluster endpoints support read/write splitting:

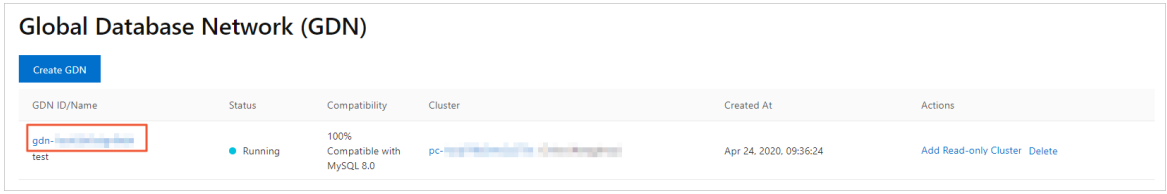
- The default cluster endpoint.
- Custom cluster endpoints whose **Read/write Mode** is set to **Read and Write (Automatic Read-write Splitting)**.

> **Note**
> - The primary endpoint and custom cluster endpoints in **Read Only** mode do not support read/write splitting.
> - For more information about how to change **Read/write Mode**, see Configure PolarProxy.
> - We recommend that you set **Primary Node Accepts Read Requests** to **No** and **Consistency Level** to **Eventual Consistency (Weak)** when you configure the custom cluster endpoint for a secondary cluster. This mitigates the impact of the replication latency between the primary cluster and secondary clusters on your business. For more information about how to configure the custom cluster endpoint, see Configure PolarProxy.

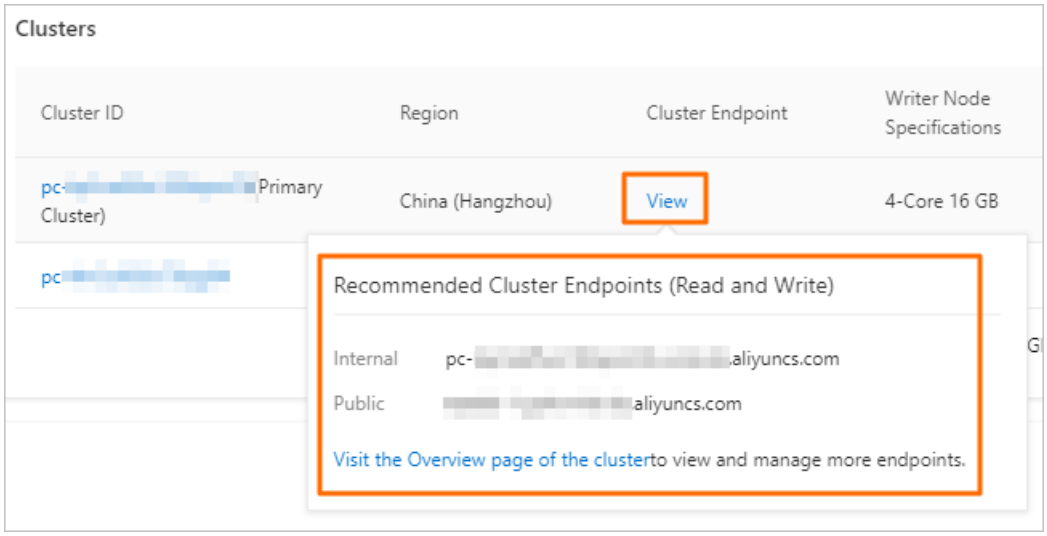
View cluster endpoints

1. Log on to the PolarDB console.
2. In the left-side navigation pane, click **Global Database Network**.

3. Find the GDN that you want to manage and click **GDN ID/Name**.



4. In the **Clusters** section, find the cluster for which you want to view the endpoints.
5. Click **View** in the **Cluster Endpoint** column. In the dialog box that appears, you can view the cluster endpoints.



> **Note**
>
> - You can view only the default cluster endpoint that consists of the private endpoint and public endpoint.
> - To view other endpoints, click **Visit the Overview page of the cluster**. For more information about how to manage endpoints, see Apply for a cluster endpoint or a primary endpoint.

Connect to a cluster

Applications in different regions connect to the GDN by using the cluster endpoint of the cluster that is deployed in the same region as the applications. The GDN automatically performs read/write splitting. For information about how to connect to a cluster, see Connect to a cluster.

Related operations

Operation	Description
DescribeGlobalDatabaseNetworks	Queries details about the GDNs that belong to an account.
CreateDBClusterEndpoint	Creates a custom cluster endpoint for a specified cluster.
DescribeDBClusterEndpoints	Queries the information about an endpoint of a specified cluster.

Operation	Description
ModifyDBClusterEndpoint	Modifies the attributes of a specified cluster endpoint, such as the read/write mode and the consistency level. You can also specify whether newly added nodes are automatically associated with the specified cluster endpoint.
DeleteDBClusterEndpoint	Releases a custom cluster endpoint of a specified cluster.

16.HTAP

16.1. IMCIs

16.1.1. Overview

This topic describes the In-Memory Column Index (IMCI) feature of PolarDB for MySQL.

Background information

PolarDB for MySQL is oriented to online transaction processing (OLTP) scenarios that involve online business and large amounts of data. It is difficult for row store-based PolarDB for MySQL to meet the query performance requirements of all scenarios. In most cases, to implement complex analytic queries, you need to export data from PolarDB for MySQL, and then import the data to an online analytical processing (OLAP) system for analysis and queries. In this context, two database systems are required, and costs, architecture complexity, and O&M loads increase.

The PolarDB for MySQL engine releases IMCI oriented to OLAP scenarios that involve large amounts of data and complex query requirements. PolarDB for MySQL provides IMCI to implement integrated real-time transaction processing and data analysis features and provides a one-stop hybrid transaction/analytical processing (HTAP) solution. PolarDB for MySQL allows you to use only one system to meet requirements of OLTP and OLAP scenarios.

> **Note**
>
> If you have questions about the feature, we recommend that you submit a ticket to contact technical support.

Required versions

- The IMCI feature is only available for PolarDB for MySQL clusters with minor version 8.0.1 and revision version 8.0.1.1.22 or later.
- Single-node Edition, Archive Database Standalone Edition, and Archive Database Cluster Edition clusters do not support the IMCI feature.

Implementation

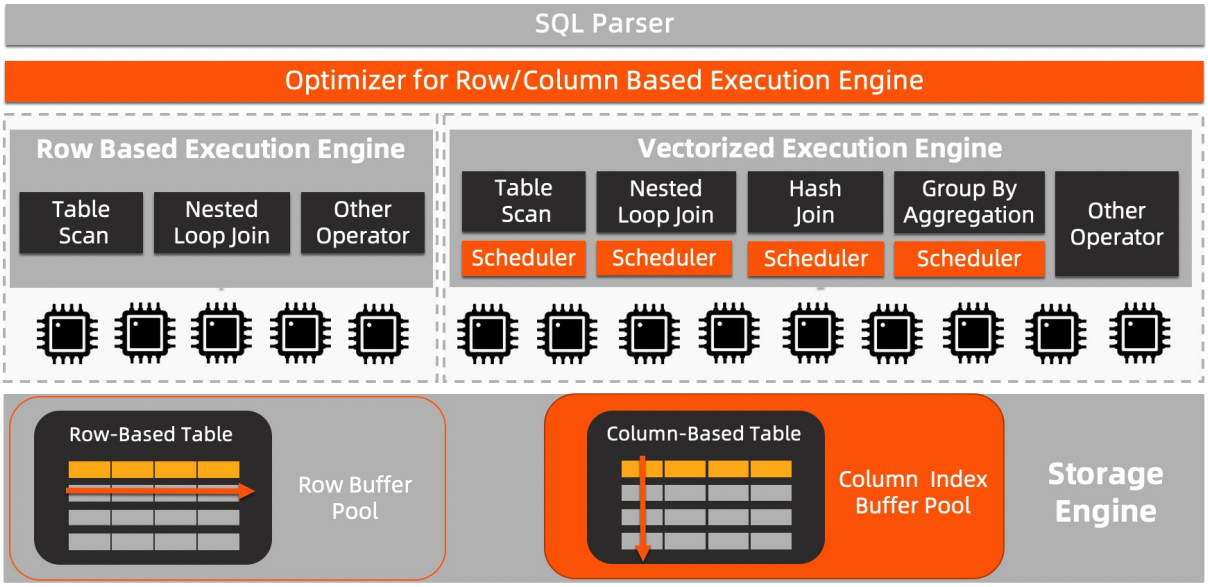
To isolate computing resources between OLAP and OLTP, you can only implement IMCIs on read-only nodes. OLAP query requests are sent only to the read-only node, not to the primary nodes. You can set request distribution rules to decide whether OLAP query requests are sent to read-only row store nodes or to read-only column store nodes.

Billing

The IMCI feature is free of charge, but you must pay for read-only column store nodes. Read-only column-store nodes are charged as common compute nodes. For more information, see Billing rules of pay-as-you-go compute nodes and Billing rules of subscription compute nodes. IMCIs also occupy billable storage space. For more information, see Storage pricing.

How it works

The following figure shows the architecture of the IMCI feature in PolarDB for MySQL.PolarDB for MySQL provides the IMCI feature designed at the storage engine, operator, and optimizer layers.



- Storage engine layer: supports the hybrid row-column storage that ensures real-time transactional consistency.
- Operator layer: uses the vectorized parallel operator that is oriented to column store. Single-table queries and multi-table queries can be implemented with minimal latency.
- SQL parser and optimizer layer: uses the cost-based optimizer (CBO) that is oriented to hybrid row-column storage. The optimizer automatically selects row store or column store based on the cost threshold to execute query requests.

This architecture helps PolarDB for MySQL accelerate queries by several orders of magnitude while PolarDB for MySQL is completely compatible with the MySQL protocol.

Benefits

Equipped with the IMCI feature, PolarDB for MySQL has the following features:

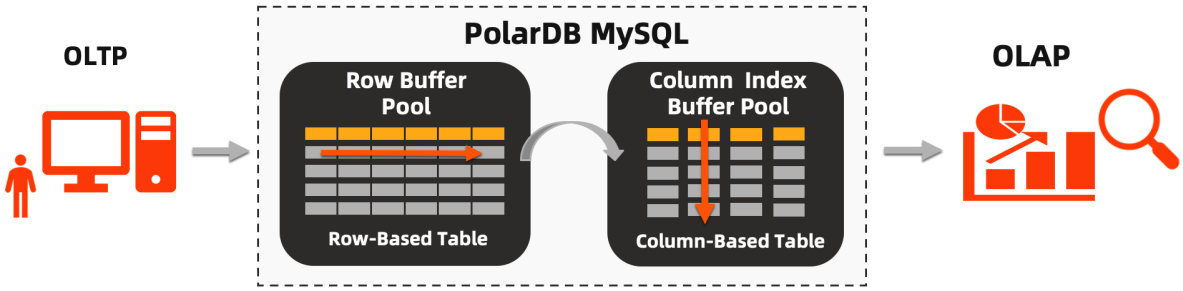
- Complete compatibility with MySQL: A system is provided for column store. This system is consistent with the system provided for row store and supports flexible type conversion.
- Ultimate HTAP performance: PolarDB provides ultimate performance in terms of OLTP. The IMCI feature provides OLAP with performance commensurate with the OLAP system.
- Hybrid row-column storage: Both row store and column store are supported, which saves costs. Moreover, transactional consistency is ensured for row store and column store. Column store also has an advantage in lower costs.

Scenarios

The IMCI feature of PolarDB for MySQL provides a one-stop HTAP experience that can be used in a variety of business scenarios.

- Scenarios where real-time analysis of online data is required, such as real-time reports.
- Data warehousing scenarios that depend on the large volume data storage capacity of PolarDB to aggregate multiple upstream data sources and use PolarDB as the dedicated data warehouse.

- Extract, transform, load (ETL)-faced accelerated data computing scenarios that depend on the powerful and flexible computing capability of IMCI provided by PolarDB to implement ETL features by using SQL syntax.



Performance improvement

The IMCI feature notably accelerates queries executed by using SQL statements by up to one hundred folds. The following section provides a query test to verify the acceleration effects. The data tables and SQL statements contained in the standard TPC-H benchmark are used in the example.

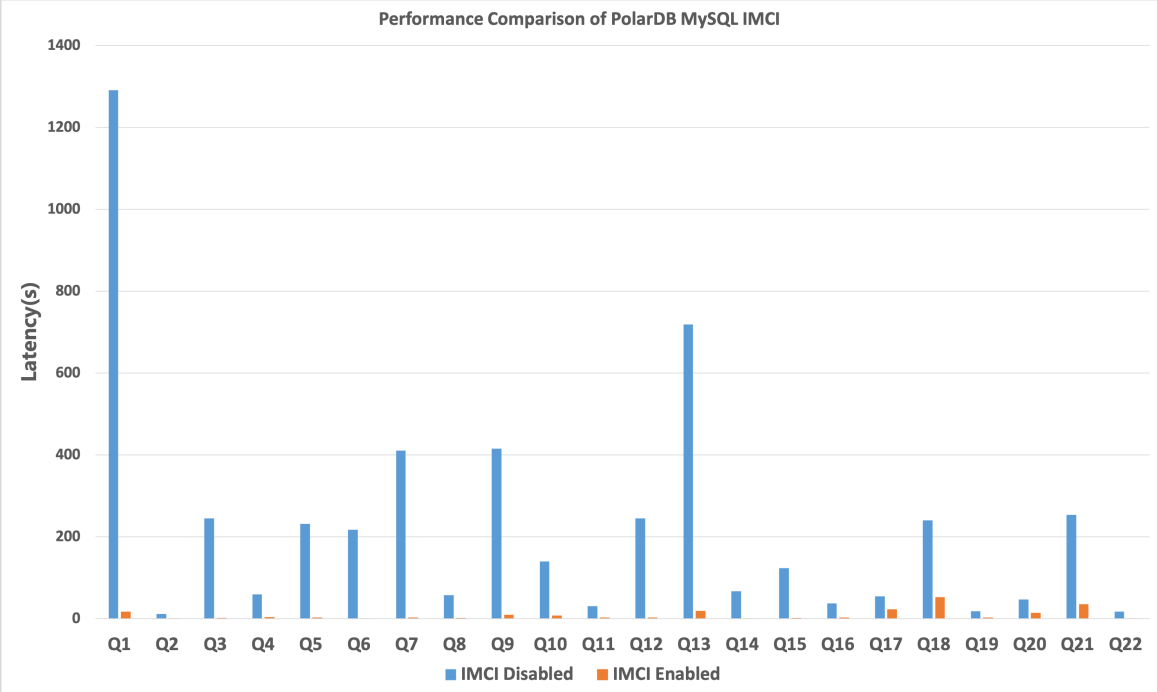
- **Test method**: TPC-H is a commonly used benchmark that is developed and released by the Transaction Processing Performance Council (TPC) to evaluate the analytic query capabilities of databases. The TPC-H benchmark contains eight tables and 22 complex SQL statements. Most of the queries contain JOIN clauses on several tables, subqueries, and GROUP BY clauses.

> **Note**
>
> In this example, a test based on the TPC-H benchmark is implemented, but it does not meet all the requirements of a TPC-H benchmark test. Therefore, the test results are incomparable with the published results of the TPC-H benchmark test.

- **Data size**: 100 GB.
- **Test results**:

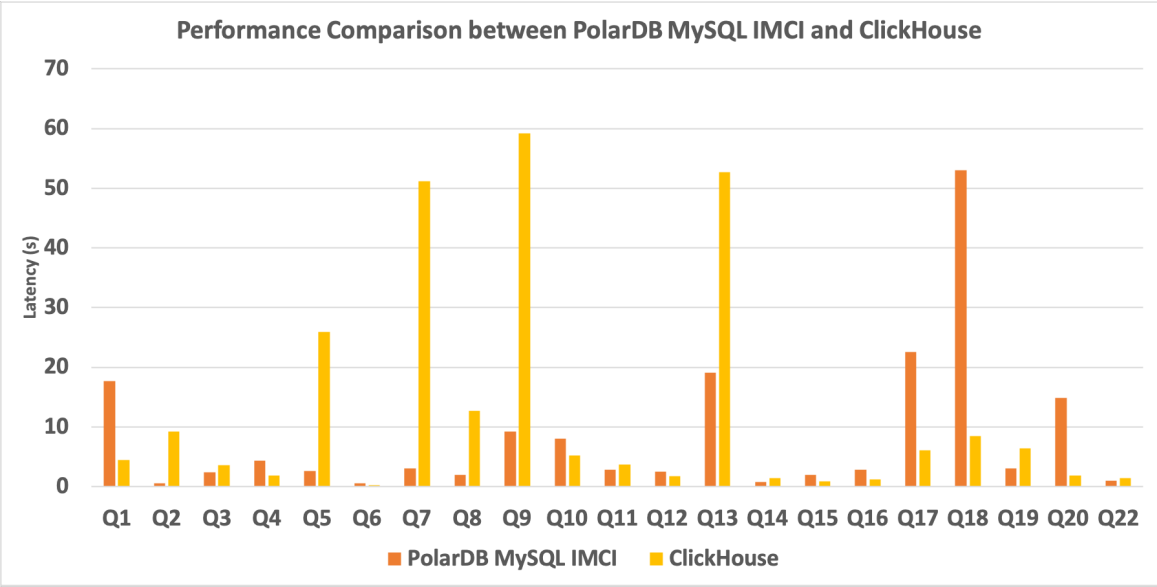
- Compare performance between IMCI-enabled and IMCI-disabled scenarios

  The following figure shows the response time difference between IMCI-enabled and IMCI-disabled scenarios when 22 complex SQL statements of the TPC-H benchmark are executed.



- Compare performance between ClickHouse and IMCI-enabled PolarDB for MySQL

  The following figure shows the response time difference between ClickHouse and IMCI-enabled PolarDB for MySQL when 21 complex SQL statements of the TPC-H benchmark are executed. The two databases have the same amount of data and the same data schema. Query Statement 21 is not executed because ClickHouse does not support Query Statement 21.



- **Conclusions**:
  - The IMCI feature notably accelerates most complex queries by up to one hundred folds.

- Each of the traditional OLAP database service ClickHouse and IMCI-enabled PolarDB for MySQL has their own advantages. PolarDB for MySQL excels in single table scan and aggregation and JOIN scenarios. In the future, the IMCI feature of PolarDB for MySQL will be tuned on an ongoing basis and make a breakthrough in terms of aggregation acceleration and window functions.

Usage

1. Add a read-only column store node.
2. Configure the cluster endpoint that contains column store nodes to implement request distribution between row store and column store nodes.
3. Use the cluster endpoint to connect to the database cluster and create an ICMI in the table based on ICMI syntax.
4. Configure a compression algorithm if necessary to reduce storage costs.

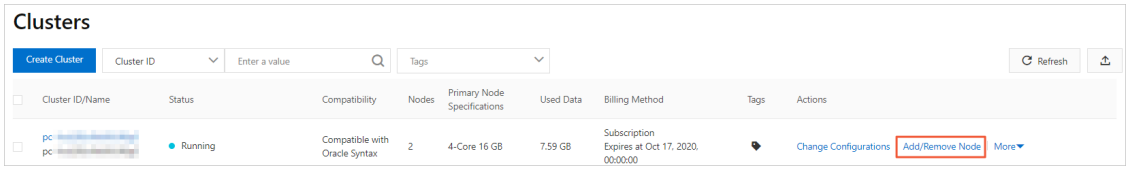
16.1.2. Add a read-only column store node

By default, a cluster contains one primary node and one read-only node. When you add a read-only node, you can select to add a read-only row store node or a read-only column store node. This topic describes how to add a read-only column store node.

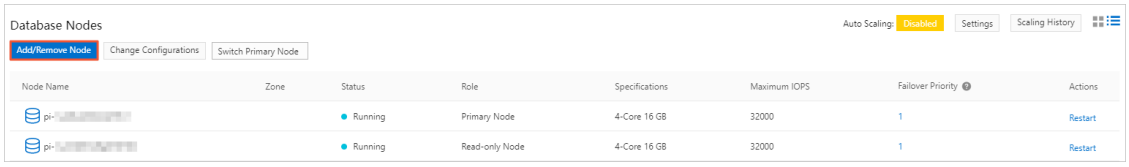
Procedure

1.
2.
3. Open the **Add/Remove Node** dialog box by using one of the following methods:
   - Open the **Add/Remove Node** dialog box on the **Clusters** page.

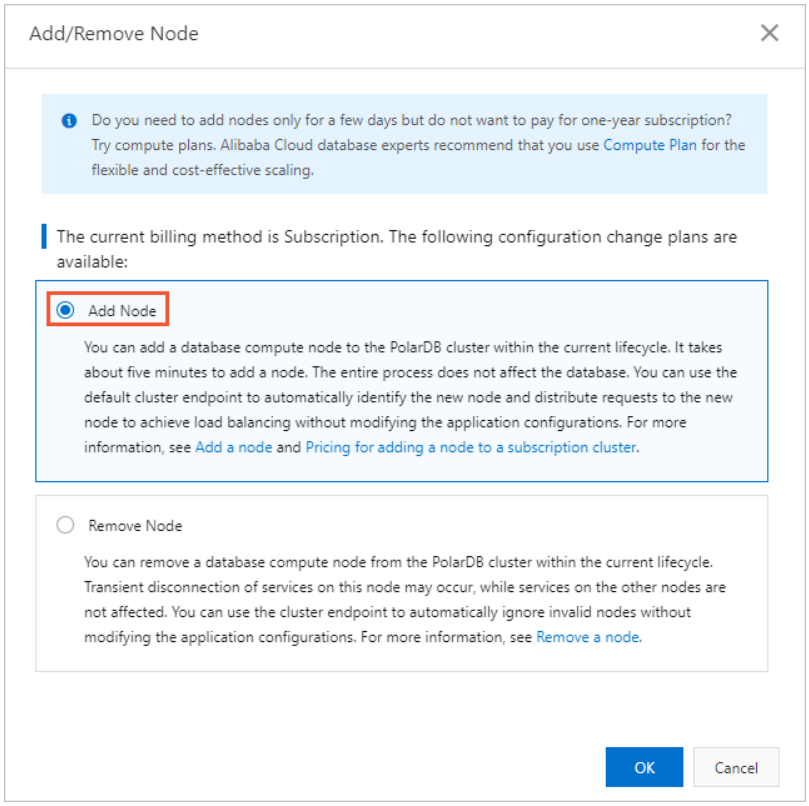
     Find the cluster that you want to manage and click **Add/Remove Node** in the **Actions** column.

   - Open the **Add/Remove Node** dialog box on the **Overview** page of the cluster.
     a. Find the cluster that you want to manage and click the cluster ID. The **Overview** page appears.
     b. In the **Database Nodes** section, click the icon to change the display mode.
     c. Click **Add/Remove Node**.

4. Select **Add Node** and click **OK**.



5. Click **+ Add a read-only node**.
6. Select **Enable** for **Columnstore Index**.
7. Set **Switching Time**, select Terms of Service, and then click **Buy Now**.

Result

After you make the payment, you are redirected to the **Basic Information** page of the cluster. After a while, the read-only column store node is added.

In the preceding figure, **Read-only Columnar Node** is the new column store node and **Read-only** is the original row store node.

16.1.3. Request distribution based on cluster endpoints

16.1.3.1. Request distribution

If your business provides both OLTP and OLAP services, you can use the cluster endpoint feature of the database proxy based on the IMCI feature to distribute OLAP requests to column store nodes and OLTP requests to row store nodes. Both manual and automatic request distribution are supported.

Background information

supports the following read-only node types:

- Read-only row store nodes: process read requests based on the row store feature. Row store nodes provide the best performance when processing OLTP read requests.

- Read-only column store nodes: process read requests based on the column store feature. Column store nodes offer much better performance than row store nodes when processing OLAP read requests, such as complex SQL queries and analytic SQL queries.

> **Note** By default, a cluster contains one primary node and one read-only row store node. When you add a read-only node, you can choose to add a row store or column store node. For more information, see Add a read-only column store node.

To maximize the performance of SQL queries, you can configure cluster endpoints to distribute OLAP requests to column store nodes and OLTP requests to row store nodes. Both manual and automatic request distribution are supported.

Automatic request distribution in hybrid scenario of OLTP and OLAP services

If both OLAP and OLTP requests use the same application to access databases, read requests of the two types can be automatically distributed to column store nodes or row store nodes based on the number of scanned rows.

Request distribution rules:

- OLTP service: generally includes read and write requests. Write requests are uniformly processed by the primary node. Read requests are processed by the read-only row store node or primary node.
- OLAP service: generally includes only read requests. Read requests are uniformly processed by the read-only column store node.

Automatic request distribution solution (**Read and Write**)

- Request distribution between the primary node and read-only column store node: Because the primary node is also in row store mode, it can also process OLTP read requests. In this solution, write requests and OLTP read requests are distributed to the primary node, and OLAP read requests are distributed to the read-only column store node.
- Request distribution between the read-only row store node and read-only column store node: In this solution, write requests are distributed to the primary node, OLTP read requests are distributed to the read-only row store node, and OLAP read requests are distributed to the read-only column store node.

Automatic request distribution rules:

- SQL requests that fall below the threshold for the number of scanned rows are distributed to the row store node or the primary node. In the case of multiple row store nodes, SLB determines the destination row store nodes.
- SQL requests that exceed the threshold for the number of scanned rows are distributed to the column store node. In the case of multiple column store nodes, SLB determines the destination column store nodes.

> **Note** We recommend that you do not set the read/write mode of the cluster endpoint to **ReadOnly** if you want to implement automatic request distribution between the row store and column store nodes. Because SLB is used in **ReadOnly** mode, OLAP service requests may occasionally not be routed to the column store node. Therefore, the request processing speed fluctuates.

For more information, see Automatic request distribution between row store and column store nodes.

Manual request distribution between row store and column store nodes for independent OLTP and OLAP services

If OLAP and OLTP requests use different applications to access databases, you can configure different cluster endpoints for the applications, and then associate row store and column store nodes to different cluster endpoints to implement request distribution.

Request distribution rules:

- OLTP service: generally includes read and write requests. Write requests are uniformly processed by the primary node. Read requests are processed by the read-only row store node or primary node.
- OLAP service: generally includes only read requests. Read requests are uniformly processed by the read-only column store node.

Manual request distribution solution (**Read and Write** or **Read Only**)

- You can associate the application for OLTP requests with a cluster endpoint that does not contain the read-only column store node, so that OLTP read requests are processed by the primary node or read-only row store node.
- You can associate the application for OLAP requests with a cluster endpoint that contains only the read-only column store node, so that OLAP read requests are processed by the read-only column store node.

For more information, see Manual request distribution between row store and column store nodes.

16.1.3.2. Automatic request distribution between row store and column store nodes

If both OLAP and OLTP requests use the same application to access databases, you must set the read/write mode of the cluster endpoint to **Read and Write** and enable the automatic request distribution feature. After the automatic request distribution feature is enabled, the database proxy implements automatic request distribution based on the number of scanned rows to maximize the performance of SQL queries. When the number of rows scanned by a SQL request exceeds the threshold, the request is automatically distributed to the column store node. When the number of rows scanned by a SQL request falls below the threshold, the request is automatically distributed to the row store node or the primary node.

Automatic request distribution solution

The database proxy of judges whether the **number of scanned rows** by SQL statements exceeds the specified threshold and then determines whether to distribute SQL requests to the row store or column store node. This can maximize the performance of the row store and column store nodes.

Request distribution rules:

- OLTP service: generally includes read and write requests. Write requests are uniformly processed by the primary node. Read requests are processed by the read-only row store node or primary node.
- OLAP service: generally includes only read requests. Read requests are uniformly processed by the read-only column store node.

Automatic request distribution solution (**Read and Write**)

- Request distribution between the primary node and read-only column store node: Because the primary node is also in row store mode, it can also process OLTP read requests. In this solution, write

requests and OLTP read requests are distributed to the primary node, and OLAP read requests are distributed to the read-only column store node.

- Request distribution between the read-only row store node and read-only column store node: In this solution, write requests are distributed to the primary node, OLTP read requests are distributed to the read-only row store node, and OLAP read requests are distributed to the read-only column store node.

Limits

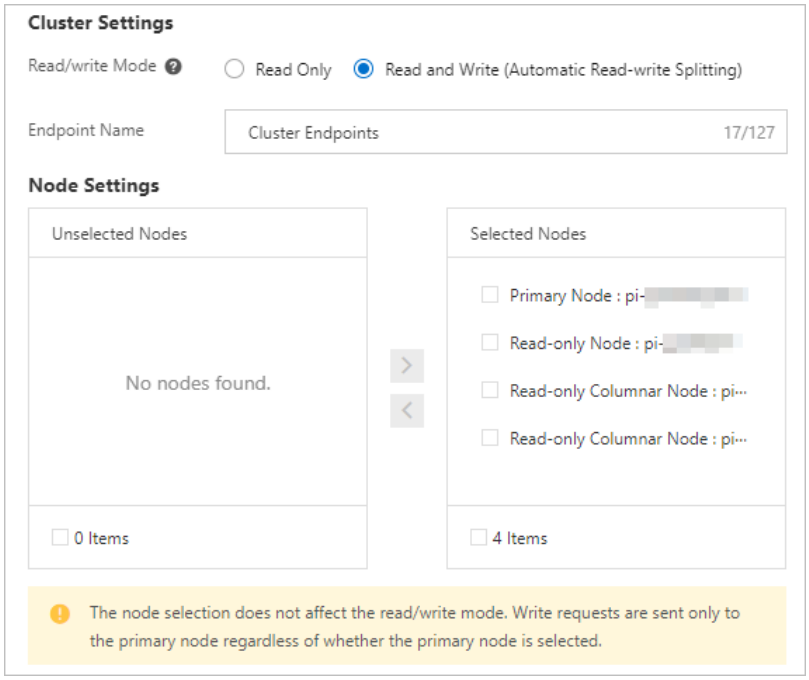
- To enable automatic request distribution between column store and row store nodes, the read/write mode of the cluster endpoint must be set to **Read and Write** and the cluster contains at least one read-only column store node.
- When the read/write mode of the cluster endpoint is **ReadOnly**, automatic request distribution between column store and row store nodes is not supported. Because SLB is used in this case, service requests may occasionally not be routed to the column store node. Therefore, if both column store and row store nodes exist, the request processing speed fluctuates.

Step 1: Enable automatic request distribution between column store and row store nodes

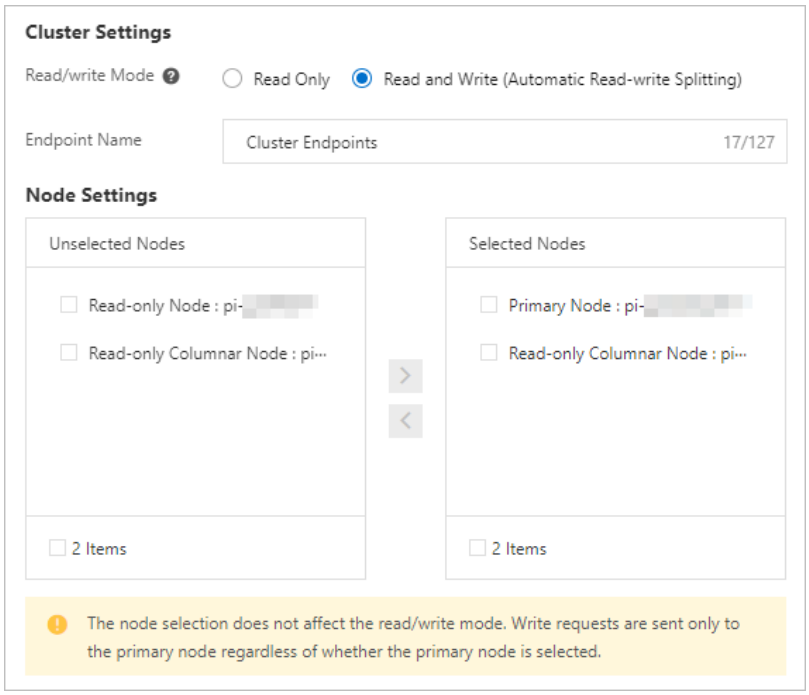
1. In the **Enterprise Edition** section on the **Overview** page, find the cluster endpoint and click **Modify** next to the cluster endpoint.
2. Set **Read/write Mode** to **Read and Write**.
3. In the **Node Settings** section, select the primary node and read-only row store and column store nodes that are used to process requests.

   > **Note** You must select at least one read-only column store node in **Node Settings**.

   Example 1: In the following figure, one primary node, one read-only row store node, and two read-only column store nodes are selected. When the automatic request distribution feature is enabled, write requests are distributed to the primary node, OLAP read requests are distributed to the read-only column store nodes, and OLTP read requests are distributed to the read-only row store node.

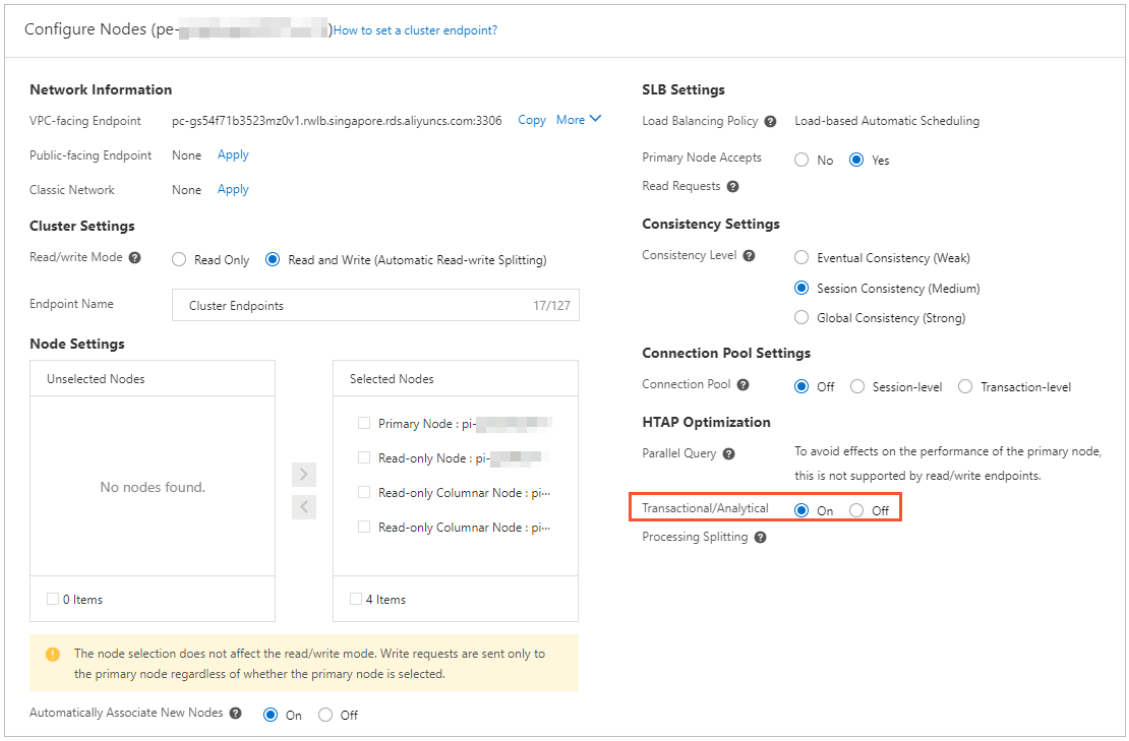


Example 2: In the following figure, one primary node and one read-only column store node are selected. When the automatic request distribution feature is enabled, write requests are distributed to the primary node, OLAP read requests are distributed to the read-only column store node, and OLTP read requests are distributed to the primary node.



> **Note** In **Read and Write** mode, all write requests are distributed only to the primary node, regardless of whether the primary node is selected in the Node Settings section.

4. In the **HTAP Optimization** section, enable **Transactional/Analytical Processing Splitting**.



5. Set other parameters for the database proxy. For more information, see Configure PolarProxy.

Step 2: Set the automatic request distribution threshold

After you enable automatic request distribution, you must set the threshold for **the number of scanned rows for an SQL statement**. After the threshold is set, the database agent judges the number of scanned rows against the threshold. When the number of rows scanned by a SQL request exceeds the threshold, the request is automatically distributed to the column store node. When the number of rows scanned by a SQL request falls below the threshold, the request is automatically distributed to the row store node or the primary node.

The maximum number of scanned rows of the SQL statement is determined by the parameters in the following table. On the **Parameters** page of the cluster, you can modify the values of the parameters based on your business requirements.

Parameter	Description
loose_imci_ap_threshold	The maximum number of scanned rows in the SQL statement for row store and column store nodes. Default value: 500000. **Note** If the value of this parameter is greater than 500000, requests are sent to column store nodes.
loose_cost_threshold_for_imci	The maximum number of scanned rows in the SQL statement for column store nodes Default value: 50000. **Note** If the value of this parameter is greater than 50000, a column store node execution plan is selected. Otherwise, a row store node execution plan is selected.

You can execute the `show status like 'Last_query_imci_cost'` statement to exactly query the execution cost of the SQL statement and therefore to determine how to modify the parameter values.

For example, you can execute the following statement to query the execution cost of the SQL statement:

```
show status like 'Last_query_imci_cost';
```

A similar result is returned:

```
+----------------------+-------+
| Variable_name        | Value |
+----------------------+-------+
| Last_query_imci_cost | 2     |
+----------------------+-------+
1 row in set (0.01 sec)
```

The query result indicates that the execution cost of the SQL statement is 2.

To implement an execution plan to distribute SQL query requests to the column store node, you must set the loose_imci_ap_threshold and loose_cost_threshold_for_imci parameters to 1.

16.1.3.3. Manual request distribution between row store and column store nodes

If OLAP and OLTP requests use different applications to access databases, you can configure different cluster endpoints for the applications, and then associate row store and column store nodes to different cluster endpoints to implement request distribution.

Manual request distribution solution

Request distribution rules:

- OLTP service: generally includes read and write requests. Write requests are uniformly processed by the primary node. Read requests are processed by the read-only row store node or primary node.
- OLAP service: generally includes only read requests. All read requests are processed by the read-only column store node.

Manual request distribution solution (**Read and Write** or **Read Only**)

- You can associate the application for OLTP requests with a cluster endpoint that does not contain the read-only column store node, so that OLTP read requests are processed by the primary node or read-only row store node.
- You can associate the application for OLAP requests with a cluster endpoint that contains only the read-only column store node, so that OLAP read requests are processed by the read-only column store node.

Procedure

You must configure different cluster endpoints for OLTP and OLAP services. For more information about how to configure cluster endpoints, see Apply for a cluster endpoint or a primary endpoint.

Take note of the following points when you configure cluster endpoints:

- Cluster endpoints for the OLTP service:
  - In **Read Only** mode, only read-only row store nodes are required.
  - In **Read and Write** mode, we recommend that you select at least one read-only row store node. If **Primary Node Accepts Read Requests** is enabled in this case, read requests are still distributed to the primary node.

    > **Note** In **Read and Write** mode, all write requests are distributed only to the primary node, regardless of whether the primary node is selected in the Node Settings section.

- Cluster endpoints for the OLAP service: The OLAP service generally contains only read requests. Therefore, we recommend that you select **Read Only** mode. In this mode, at least one read-only column store node is required.

16.1.4. IMCI syntax

16.1.4.1. Execute the CREATE TABLE statement to create an IMCI

This topic describes how to create an IMCI when you create a table.

Prerequisites

After you add a read-only column store node and configure the cluster endpoint, you can connect to the cluster by using the cluster endpoint and execute the statement to create and manage an IMCI.

- Read-only column store nodes are added. For more information, see Add a read-only column store node.
- The cluster endpoint is configured. Manual and automatic request distribution solutions can be used to distribute requests to row store and column store nodes. You can select a Request distribution method based on your business and configure the cluster endpoint.
- The database cluster is connected by using the cluster endpoint. For more information, see Connect to a cluster.

Create an IMCI

- **CREATE TABLE** syntax
  - To create an IMCI, you need only to add the **COLUMNAR=1** string to the **COMMENT** field of the **CREATE TABLE** statement when you create a table. Other parameters remain unchanged and unaffected.
  - The **COLUMNAR=1** string can be added separately to the **COMMENT** field of a column, which is valid only for the column. Alternatively, you can add the string to the **COMMENT** field at the end of the **CREATE TABLE** statement, which is valid for columns of all supported data types in the table.

    > **Note**
    > - For PolarDB for MySQL 8.0.1.1.25 and later, IMCIs support the BLOB and TEXT data types.
    > - IMCIs do not support data types such as ENUM, SET, BIT, or Geo.

Examples:

```
CREATE TABLE t1(
  col1 INT COMMENT 'COLUMNAR=1',
  col2 DATETIME COMMENT 'COLUMNAR=1',
  col3 VARCHAR(200)
) ENGINE InnoDB;
CREATE TABLE t2(
  col1 INT,
  col2 DATETIME,
  col3 VARCHAR(200)
) ENGINE InnoDB COMMENT 'COLUMNAR=1';
```

In the preceding example:

- When you create the `t1` table, the IMCI valid for the col1 and col2 columns is created.
- When you create the `t2` table, an IMCI valid for the table is created: the IMCI is valid for the col1, col2, and col3 columns.

- **CREATE TABLE LIKE** syntax: If you execute the **CREATE TABLE LIKE** statement to create a table and the source table contains an IMCI, the destination table contains the same IMCI.
- **CREATE TABLE ... SELECT** syntax: If you execute the **CREATE TABLE ... SELECT** statement to create a table, you can add the **COLUMNAR=1** string to the **COMMENT** field to create an IMCI valid for all columns of the table. However, when you execute the **CREATE TABLE ... SELECT** statement to create a table, you cannot create an IMCI valid for a specified column.

  Examples:

  ```
  CREATE TABLE t3(
    col1 INT,
    col2 DATETIME,
    col3 VARCHAR(200)
  ) ENGINE InnoDB;
  CREATE TABLE t4 COMMENT 'COLUMNAR=1' SELECT col1, col2 FROM t3;
  ```

  In the preceding example:

  The **COMMENT 'COLUMNAR=1'** string is valid for the `t4` table. An IMCI valid for all columns of the `t4` table is created, including existing columns col1 and col2, and subsequently added columns.

View the IMCI structure of a table

- Syntax: You can execute the `SHOW CREATE TABLE <tablename> FULL` statement to view the IMCI structure of a table.
- Examples:

```
SHOW CREATE TABLE test.t1\G
*************************** 1. row ***************************
       Table: t1
Create Table: CREATE TABLE `t1` (
  `id` int(11) NOT NULL,
  `col1` int(11) DEFAULT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='columnar=1'
SHOW CREATE TABLE test.t1 FULL\G
*************************** 1. row ***************************
       Table: t1
Create Table: CREATE TABLE `t1` (
  `id` int(11) NOT NULL,
  `col1` int(11) DEFAULT NULL,
  PRIMARY KEY (`id`),
  COLUMNAR INDEX (`id`,`col1`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='columnar=1'
```

In the preceding example:

- Only COMMENT information, but not the COLUMNAR INDEX definition, is displayed when you execute the `SHOW CREATE TABLE <tablename>` statement.
- To display the IMCI structure in the COLUMNAR INDEX field, you must execute the `SHOW CREATE TABLE <tablename> FULL` statement.

16.1.4.2. Use DDL statements to dynamically add and delete an IMCI

This topic describes how to execute DDL statements to dynamically create and delete an IMCI after a table is created.

Prerequisites

After you add a read-only column store node and configure the cluster endpoint, you can connect to the cluster by using the cluster endpoint and execute the statement to create and manage an IMCI.

- Read-only column store nodes are added. For more information, see Add a read-only column store node.
- The cluster endpoint is configured. Manual and automatic request distribution solutions can be used to distribute requests to row store and column store nodes. You can select a Request distribution method based on your business and configure the cluster endpoint.
- The database cluster is connected by using the cluster endpoint. For more information, see Connect to a cluster.

Create an IMCI

- Syntax:
  - Add the **COMMENT 'COLUMNAR=1'** field to the **ALTER TABLE** statement to create an IMCI that is valid for the entire table.
  - Add the **COMMENT 'COLUMNAR=1'** field to the **ALTER TABLE ... MODIFY COLUMN ...** statement to create an IMCI valid for a specified column.

> **Note**
>
> - For PolarDB for MySQL 8.0.1.1.25 and later, IMCIs support the BLOB and TEXT data types.
> - IMCIs do not support data types such as ENUM, SET, BIT, or Geo.

- Examples:

```
CREATE TABLE t5(
  col1 INT,
  col2 DATETIME,
  col3 VARCHAR(200)
) ENGINE InnoDB;
-- Create an IMCI valid for a table
ALTER TABLE t5 COMMENT 'COLUMNAR=1';
-- Create an IMCI valid for specified columns
ALTER TABLE t5 MODIFY COLUMN col1 INT COMMENT 'COLUMNAR=1',
               MODIFY COLUMN col2 DATETIME COMMENT 'COLUMNAR=1';
```

Delete an IMCI

- Syntax:
  - Add the **COMMENT 'COLUMNAR=0'** field to the **ALTER TABLE** statement to delete an IMCI that is valid for the entire table.
  - Add the **COMMENT 'COLUMNAR=0'** field to the **ALTER TABLE ... MODIFY COLUMN ...** statement to delete an IMCI that is valid for a specified column.
- Examples:

```
-- Create an IMCI valid for specified columns
CREATE TABLE t6(
  col1 INT COMMENT 'COLUMNAR=1',
  col2 DATETIME COMMENT 'COLUMNAR=1',
  col3 VARCHAR(200)
) ENGINE InnoDB;
-- Delete an IMCI valid for specified columns
ALTER TABLE t6 MODIFY COLUMN col1 INT COMMENT 'COLUMNAR=0',
               MODIFY COLUMN col2 DATETIME COMMENT 'COLUMNAR=0';
-- Create an IMCI valid for a table
CREATE TABLE t7(
  col1 INT,
  col2 DATETIME,
  col3 VARCHAR(200)
) ENGINE InnoDB COMMENT 'COLUMNAR=1';
-- Delete an IMCI valid for a table
ALTER TABLE t7 COMMENT 'COLUMNAR=0';
```

Modify IMCI definition

- Syntax:
  - Add the **COMMENT 'COLUMNAR=1'** field to the **ALTER TABLE ... MODIFY COLUMN ...** statement to add a column to the IMCI.

- Add the **COMMENT 'COLUMNAR=0'** field to the **ALTER TABLE ... MODIFY COLUMN ...** statement to delete a column for which the IMCI is valid.

> **Note**
> - For PolarDB for MySQL 8.0.1.1.25 and later, IMCIs support the BLOB and TEXT data types.
> - IMCIs do not support data types such as ENUM, SET, BIT, or Geo.

- Examples:

```
CREATE TABLE t8(
  col1 INT COMMENT 'COLUMNAR=1',
  col2 DATETIME COMMENT 'COLUMNAR=1',
  col3 VARCHAR(200)
) ENGINE InnoDB;
-- Add a column for which the IMCI is valid.
ALTER TABLE t8 MODIFY COLUMN col3 VARCHAR(200) COMMENT 'COLUMNAR=1';
-- Delete a column for which the IMCI is valid.
ALTER TABLE t8 MODIFY COLUMN col2 DATETIME COMMENT 'COLUMNAR=0';
```

Create an IMCI valid for multiple columns

Tables that contain multiple columns are often involved in the OLAP service. You can use the COMMENT field to simplify the process to create an IMCI valid for a table that contains multiple columns. By default, the IMCI is valid for all columns of supported data types in the table. You can also specify only a few columns for which the IMCI is not valid.

> **Note**
> - For PolarDB for MySQL 8.0.1.1.25 and later, IMCIs support the BLOB and TEXT data types.
> - IMCIs do not support data types such as ENUM, SET, BIT, or Geo.

For example, if you execute the following statement to create a table:

```
CREATE TABLE t9(
  col1 INT, col2 INT, col3 INT,
  col4 DATETIME, col5 TIMESTAMP,
  col6 CHAR(100), col7 VARCHAR(200),
  col8 TEXT, col9 BLOB
) ENGINE InnoDB;
```

You can execute the following statement to create an IMCI valid for the table:

```
ALTER TABLE t9 COMMENT 'COLUMNAR=1', MODIFY COLUMN col7 VARCHAR(200) COMMENT 'COLUMNAR=0';
```

A similar result is returned:

```
SHOW CREATE TABLE t9 FULL\G
*************************** 1. row ***************************
       Table: t9
Create Table: CREATE TABLE `t9` (
  `col1` int(11) DEFAULT NULL,
  `col2` int(11) DEFAULT NULL,
  `col3` int(11) DEFAULT NULL,
  `col4` datetime DEFAULT NULL,
  `col5` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  `col6` char(100) DEFAULT NULL,
  `col7` varchar(200) DEFAULT NULL COMMENT 'COLUMNAR=0',
  `col8` text,
  `col9` blob,
  COLUMNAR INDEX  (`col1`,`col2`,`col3`,`col4`,`col5`,`col6`,`col8`,`col9`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT 'COLUMNAR=1'
```

In the preceding example, the IMCI is not valid for the col7 column.

However, due to the InnoDB Online DDL implementation, the `ALTER TABLE t9 COMMENT 'COLUMNAR=1', MODIFY COLUMN col7 VARCHAR(200) COMMENT 'COLUMNAR=0';` statement in the preceding example is implemented in online rebuild mode, resulting in poor performance. You can try the following method:

```
-- Modify the COMMENT field for the column for which the IMCI is not valid.
ALTER TABLE t9 MODIFY COLUMN col7 VARCHAR(200) COMMENT 'COLUMNAR=0';
-- Modify the COMMENT field for the table to create the IMCI which is valid for the table.
ALTER TABLE t9 COMMENT 'COLUMNAR=1';
```

Create an IMCI when adding columns

When you execute the **ALTER TABLE ADD COLUMN** statement to add columns, you can add the COMMENT 'COLUMNAR=1' field to create an IMCI valid for the columns.

> **Note**
>
> - For PolarDB for MySQL 8.0.1.1.25 and later, IMCIs support the BLOB and TEXT data types.
> - IMCIs do not support data types such as ENUM, SET, BIT, or Geo.

For example, you can execute the following statement to create a table, and create an IMCI which is valid for the col1 and col2 columns:

```
CREATE TABLE t10(
  col1 INT COMMENT 'COLUMNAR=1',
  col2 DATETIME COMMENT 'COLUMNAR=1',
  col3 VARCHAR(200)
) ENGINE InnoDB;
```

You can execute the following statement to add the col4 column for which the IMCI is also valid to the `t10` table:

```
ALTER TABLE t10 ADD col4 DATETIME DEFAULT NOW() COMMENT 'COLUMNAR=1';
```

It is no longer an INSTANT DDL statement because it involves changes to the IMCI. This DDL statement deletes the old IMCI when adding the column, and creates a new IMCI that is valid for the col1, col2, and col4 columns.

View the statuses of indexes

After the IMCI feature is enabled, OLAP query requests are distributed to the read-only column store node instead of the primary node. This isolates OLAP and OLTP computing resources. Due to this isolation, the online DDL statements for creating or modifying an IMCI are optimized as **asynchronous DDL** statements. The following logic is used: After the metadata of tables and IMCIs is modified on the primary node, the modifications are synchronized to the read-only column store node by using Redo logs. The read-only column store node starts background threads to concurrently build IMCIs after the data dictionary modifications take effect.

The **asynchronous DDL** logic means that IMCIs can be queried only after they are built, although DDL statements are submitted and data dictionary modifications take effect. If you perform an OLAP query immediately after the DDL statement is executed, the read-only row store node is still used. If you perform an OLAP query after the IMCI is built, the read-only column store node is used.

You can first execute the INFORMATION_SCHEMA.IMCI_INDEXES statement on a read-only column store node to check whether the IMCI is created.

For example, if you execute the following statement to create a table:

```
CREATE TABLE t11(
  col1 INT, col2 DATETIME, col3 VARCHAR(200)
) ENGINE InnoDB;
```

You can execute the following DDL statement to create an IMCI:

```
ALTER TABLE t11 COMMENT 'COLUMNAR=1';
```

This DDL statement is similar to an INSTANT DDL in effect and executed quickly on the primary node. However, if you immediately execute the following statement to perform a query:

```
SELECT * FROM INFORMATION_SCHEMA.IMCI_INDEXES WHERE TABLE_NAME = 't11';
```

If the STATE field in the result is RECOVERING instead of COMMITTED, the IMCI is still being created.

```
+--------+-----------+----------+--------+---------+------+----------+--------+
|TABLE_ID|SCHEMA_NAME|TABLE_NAME|NUM_COLS|PACK_SIZE|ROW_ID|STATE     |MEM_SIZE|
+--------+-----------+----------+--------+---------+------+----------+--------+
|    xxxx| test      | t11      |       3|    65536|     0|RECOVERING|    0   |
+--------+-----------+----------+--------+---------+------+----------+--------+
```

If the STATE field is COMMITTED, the IMCI has been created. If you perform an OLAP query now, the read-only column store node is used.

16.1.4.3. Set a compression algorithm

To improve compression efficiency of an IMCI and reduce storage costs, you can set a compression algorithm for the IMCI. This topic describes how to set and modify the compression algorithm for an IMCI.

Supported compression algorithms

supports two IMCI compression algorithms: LZ4 and ZSTD.

- LZ4: a lossless data compression algorithm with a compression speed greater than 500 MB/s per core. For more information, see LZ4 GitHub.
- ZSTD (or Zstandard): a lossless data compression algorithm with a similar compression speed to LZ4. For more information, see Zstandard GitHub.

Set a compression algorithm when creating a table

- Syntax:

  When you execute the **CREATE TABLE** statement to create a table, you can specify the codec_opt parameter in the **COMMENT** field to set a compression algorithm.

  ```
  COMMENT 'COLUMNAR=1 codec_opt={LZ4}'
  ```

  Valid values of the codec_opt parameter: `LZ4`, `ZSTD`, and `NONE`, corresponding to LZ4 compression, ZSTD compression, and no compression.

  > **Note**
  > - The imci_default_codec parameter defines the default compression algorithm. When the codec_opt parameter in the COMMENT field uses the default compression algorithm, the compression algorithm is defined by the imci_default_codec parameter. The default value of the imci_default_codec parameter is `ZSTD`.
  > - You can execute the **SET** statement to modify the value of the imci_default_codec parameter at the session level, or submit a ticket to apply for a global modification.

  The compression algorithm specified in the **COMMENT** field of a table is applied to all IMCIs of the table. If a compression algorithm is also set in the **COMMENT** field of a column, the compression algorithm of the IMCI valid for the column takes precedence.

- Example:

  ```
  CREATE TABLE t12(
    col1 INT,
    col2 DATETIME,
    col3 VARCHAR(200)
  ) ENGINE InnoDB COMMENT 'COLUMNAR=1 codec_opt={LZ4}';
  SET imci_default_codec="{LZ4}";
  CREATE TABLE t13(
    col1 INT COMMENT 'codec_opt={NONE}',
    col2 DATETIME,
    col3 VARCHAR(200) 'codec_opt={ZSTD}',
  ) ENGINE InnoDB COMMENT 'COLUMNAR=1';
  ```

  In the preceding example:

- The default compression algorithm specified in the **COMMENT** field of the `t12` table is LZ4. Therefore, data in the col1, col2, and col3 columns uses the LZ4 compression algorithm.
- No compression algorithm is set for the `t13` table. Compression is disabled for the col1 column. The codec_opt parameter is not set for the col2 column, so the LZ4 compression algorithm specified by the imci_default_codec parameter is used. The col3 column uses the ZSTD compression algorithm because the codec_opt={ZSTD} string is added for the col3 column and the compression algorithm specified for a column has a higher priority than the compression algorithm specified by the imci_default_codec parameter.

Modify the compression algorithm of the IMCI valid for a column

You cannot modify the compression algorithm of the IMCI valid for a column. To this end, you must delete the original data of the column, and then use the table creation statement to specify a new compression algorithm of the IMCI valid for the column. Modifying the compression method of the IMCI valid for a column will be available later.

Example:

```
CREATE TABLE t14(
  col1 INT COMMENT 'COLUMNAR=1 codec_opt={ZSTD}',
  col2 DATETIME COMMENT 'COLUMNAR=1 codec_opt={ZSTD}',
  col3 VARCHAR(200)
) ENGINE InnoDB;
-- Modify the compression algorithm of the IMCI valid for a column.
ALTER TABLE t14 COMMENT 'COLUMNAR=1', MODIFY COLUMN col2 DATETIME COMMENT 'codec_opt={LZ4}'
;
```

In the preceding example, an IMCI is created for the col2 column in the `t14` table, and the compression algorithm is ZSTD. If you execute the ALTER TABLE statement to try to modify the compression algorithm to LZ4, the new compression algorithm does not take effect immediately, but when you rebuild data. The original compression algorithm is used for existing data and subsequently added data because the ALTER TABLE statement does not trigger a rebuild data operation.

17.Cluster Recycle

17.1. Pricing

The Cluster Recycle stores released clusters. You can restore a cluster in the Cluster Recycle to a new cluster, or delete a backup set of the cluster. This topic describes the pricing rules of the Cluster Recycle.

Level-1 backups are provided free of charge. Level-2 backups are paid services.

Region	Fee (USD/GB/hour)
Regions in mainland China	0.0000325
Regions outside mainland China	0.0000455

17.2. Restore a released cluster

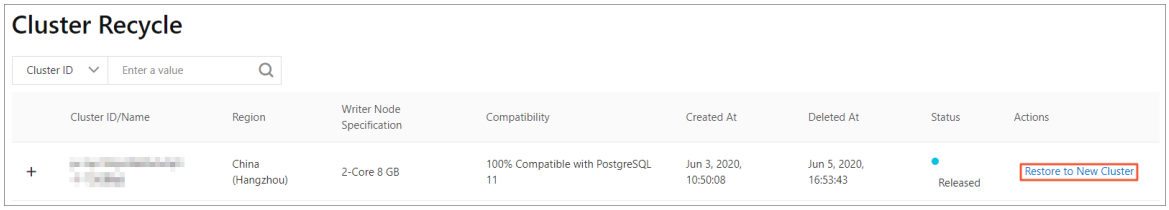
Cluster Recycle stores released clusters. You can perform actions on clusters in Cluster Recycle, such as restoring released clusters to new clusters and deleting the backup sets of released clusters. This topic uses a cluster as an example to describe how to restore clusters in Cluster Recycle.

Precautions

- Released clusters in Cluster Recycle must have at least one backup set. If all the backup sets of a cluster have been deleted, you cannot restore the released cluster.
- After a cluster is released, all level-1 backups in Cluster Recycle are asynchronously archived to level-2 backups at a rate of approximately 150 MB/s. For more information about backups, see Data backup.

Procedures

1.
2.
3. In the left-side navigation pane, click **Cluster Recycle**.
4. Find the cluster that you want to restore, and click **Restore to New Cluster** in the **Actions** column.



5. Select **Subscription** or **Pay-As-You-Go** in the **Product Type** section.
6. Configure the following parameters.

Parameter	Description

Parameter	Description
Region	The region where the cluster resides. You cannot change the region after the cluster is created. **Note** Make sure that the cluster and the ECS instance you want to connect to the cluster are deployed in the same region. Otherwise, the cluster and the ECS instance cannot communicate over the internal network, which results in decreased performance.
Create Type	Select **Recover from Recycle** to restore the released cluster from Cluster Recycle.
Source Version	Select the version of the released cluster.
Deleted Clusters	Select the ID of the released cluster.
Backup History	Select the backup set to restore. **Note** The **Backup History** drop-down list displays the timestamps of backup sets in UTC. However, the **Data Backups** tab displays timestamps of backup sets in the same time zone as the system time. Make sure that you choose the correct backup set. For example, if the timestamp of a backup set on the Backups tab is `11:19:30 on May 28, 2021` (UTC+08:00), select `2021-05-28T03:19:30Z (UTC)` in the Backup History drop-down list.
Primary Availability Zone	Select the primary zone where the cluster is deployed. **Note** In regions that have two or more availability zones, automatically replicates data to the secondary zone for disaster recovery.
Network Type	This parameter can only be set to **VPC**.

Parameter	Description
VPC	Select a **VPC** and a **vSwitch** for the cluster. We recommend that you use the same VPC and vSwitch that are used for the original cluster. **Note** Make sure that the cluster and the ECS instance you want to connect to the cluster are deployed in the same VPC. Otherwise, the cluster and the ECS instance cannot communicate over the internal network, which results in decreased performance.
vSwitch	
Compatibility	By default, the database engine version of the cluster is the same as that of the released cluster. You do not need to change this parameter value.
Edition	This parameter can only be set to .
Specification Type	has the following two types of specifications: **General Specification** and **Dedicated Specification**. ◦ Dedicated: The computing resources such as CPUs that are allocated to each cluster are exclusive to the cluster. This improves the stability and reliability. ◦ General-purpose: Idle computing resources such as CPUs are shared among clusters on the same server for cost-effectiveness. For more information about the two types of specifications, see Comparison between general-purpose and dedicated compute nodes.
Node Specification	Select a **node specification**. The maximum storage capacity and performance of clusters vary based on node specifications. For more information, see Specifications of compute nodes. **Note** We recommend that you select a **node specification** that is the same or higher than the node specification of the original cluster. This ensures that the new cluster runs as expected.
Nodes	◦ The default number of nodes of the **edition** is **2**. You do not need to change the value of this parameter. **Note** By default, a new cluster of contains one primary node and one read-only node. After a cluster is created, you can add nodes to the cluster. A cluster can contain one primary node and up to 15 read-only nodes. For more information about how to add nodes, see Add or remove read-only nodes. ◦ The default number of nodes of the and **editions** is **1**. You do not need to change this parameter value.
Storage Cost	You do not need to select the storage capacity when you purchase clusters. You are charged for the storage capacity used on an hourly basis. You can also purchase a storage plan based on your business requirements. For more information about how to purchase a storage plan, see Purchase a storage plan.

Parameter	Description
Time Zone	The time zone of the cluster. The default value is **UTC+08:00**.
Table Name Case Sensitivity	Specifies whether table names of the cluster are case-sensitive. The default value is **Not Case-sensitive (Default)**. If the databases in your instance have case-sensitive names, we recommend that you select **Case-sensitive**. **Note** This parameter value cannot be changed after the cluster is created. Proceed with caution.
Release Cluster	The backup set retention policy that is used when the cluster is deleted or released. The default value is **Retain Last Automatic Backup (Automatic Backup before Release) (Default)**. ○ **Retain Last Automatic Backup (Automatic Backup before Release) (Default)**: retains the last backup set when you release the cluster. ○ **Retain All Backups**: retains all backups when you delete the cluster. ○ **Delete All Backups (Cannot be restored)**: deletes all backups when you delete the cluster. **Note** If you retain backup sets after you delete or release the cluster, you may be billed for storage costs. You can delete the backup sets to reduce costs. For more information, see Billing rules of data backups that exceed the free quota.
Cluster Name	The name of the cluster. The name must meet the following requirements: ○ It cannot start with `http://` or `https://`. ○ It must be 2 to 256 characters in length. If you do not specify this parameter, the system automatically generates a cluster name. You can change the name after the cluster is created.
Resource Group	Select a resource group that you want to manage. For more information about how to create resource groups, see Create a resource group. **Note** A resource group is a container that contains a group of resources in an Alibaba Cloud account. You can manage these resources in a centralized manner. A resource can belong to only one resource group. For more information, see Use RAM to create and authorize resource groups.
Purchase Plan	Specify **Purchase Plan** for the cluster. **Note** This parameter is available only when the **Billing Method** parameter is set to **Subscription**.
Number	Select the **number** of clusters you want to purchase.

7. Complete the rest of the steps based on the **billing method** of the cluster.
   - **Pay-as-you-go**
     a. Click **Buy Now**.
     b. On the **Confirm Order** page, confirm your order information. Read and accept the terms of service, and then click **Buy Now**.
   - **Subscription**
     a. Click **Buy Now**.
     b. On the **Confirm Order** page, confirm your order information. Read and accept the terms of service, and then click **Buy Now**.
     c. On the **Purchase** page, confirm the information of the unpaid order and the payment method and click **Purchase**.

   After you complete the payment, it requires 10 to 15 minutes to create the cluster. Then, the newly created cluster is displayed on the **Clusters** page.

   > **Note** The amount of time required to restore data to a new cluster depends on the size of the backup set. It takes more time for the system to restore data from a larger backup set. After the cluster is created, you can return to the PolarDB console and view the new cluster on the **Clusters** page.

Related API operations

API	Description
CreateDBCluster	You can call the CreateDBCluster operation to restore the data of a cluster. **Note** You must set **CreationOption** to **CloneFromPolarDB**.

17.3. Delete a released cluster

The Cluster Recycle stores released clusters. You can restore a cluster in the Cluster Recycle to a new cluster, or delete backup sets of the cluster. This topic uses clusters as an example to describe how to delete the backup sets of released clusters.

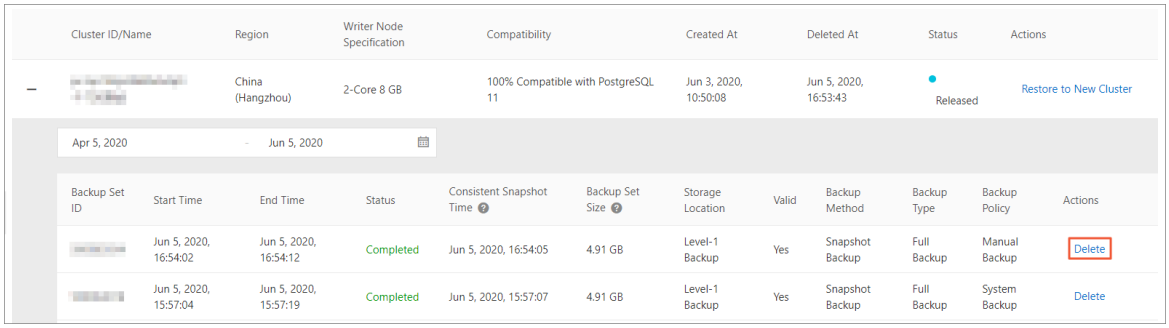
Considerations

- Released clusters in Cluster Recycle must have at least one backup set. If all the backup sets of a cluster have been deleted, you cannot restore the released cluster.
- After a cluster is released, all level-1 backups in Cluster Recycle are asynchronously archived to level-2 backups at a rate of approximately 150 MB/s. For more information about backups, see Data backup .

Delete a backup

1.
2.

3. In the left-side navigation pane, click **Cluster Recycle**.
4. Find the cluster that you want to manage, and click the + icon next to the cluster to show a list of backup sets.
5. Find the backup set that you want to delete, and click **Delete** in the **Actions** column.



6. In the pop-up dialog box, click **OK**.

> **Warning** If you delete all the backup sets of a cluster in the **Cluster Recycle**, the cluster cannot be restored. Proceed with caution.

18.Monitoring and optimization

18.1. Diagnosis

This topic describes the diagnosis feature provided by . The diagnosis feature integrates with Database Autonomy Service (DAS). This allows you to view database diagnostics and optimization results on multiple tabs. The following tabs are included: Autonomy Center, Session Management, Real-time Monitoring, Storage Analysis, Deadlock Analysis, Diagnostic Reports, and Performance Insight.

Features

- **Autonomy Center**: You can enable the autonomy service on the Autonomy Center tab. Then, if an exception occurs in a database, DAS automatically performs root cause analysis, provides optimization or stop-loss suggestions, and runs optimization or stop-loss tasks. Optimization tasks are allowed based on your authorization. For more information, see Autonomy center.
- **Session Management**: On the Session Management tab, you can view the session details and statistics for a target instance. For more information, see Session Management.
- **Real-time Monitoring**: On the Real-time Monitoring tab, you can check the information about queries per second (QPS), transactions per second (TPS), network traffic, and other related metrics for a target cluster. For more information, see Real-time Monitoring.
- **Storage Analysis**: On the Storage Analysis tab, you can check the storage overview for a target instance. For example, you can view the number of available days of storage, tablespace usage status, fragmentation percentage, and exception analysis results. For more information, see Storage analysis.
- **Deadlock Analysis**: On the Deadlock Analysis tab, you can analyze the latest deadlock and check the analysis details. For more information, see Deadlock analysis.
- **Performance Insight**: On the Performance Insight tab, you can quickly assess the database load. You can also identify the source of performance issues. This allows you to improve the performance and stability of your database. For more information, see Performance Insight.
- **Diagnostic Reports**: On the Diagnostic Reports tab, you can customize the conditions to generate and view diagnostic reports. For more information, see Diagnostic reports.

18.2. Autonomy center

The diagnostics feature of integrates with Database Autonomy Service (DAS). On the Autonomy Center tab, you can enable the autonomy service. Then, if an exception occurs in a database, DAS automatically performs root cause analysis, gives suggestions, and then performs optimizations and fixes issues. Optimizations are allowed based on your authorization.

Prerequisites

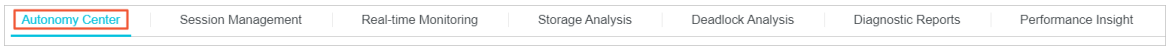
The service edition of your clusters is or . This feature is unavailable for . For more information about service editions, see Overview.

Considerations

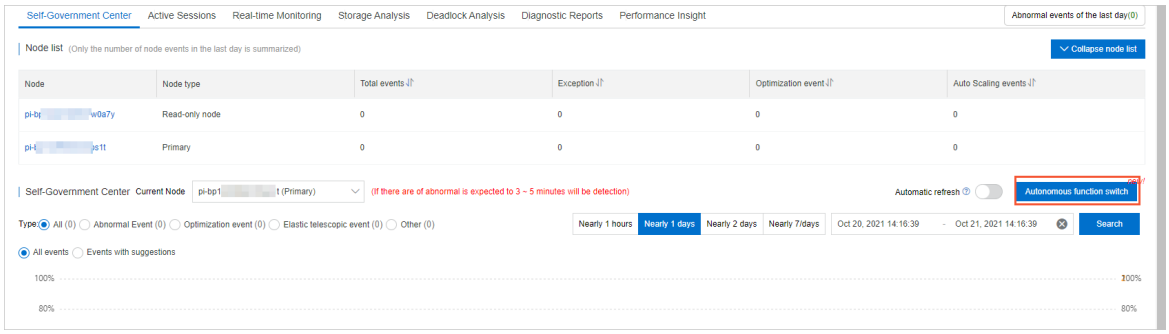
Only the clusters whose **product type** is **pay-as-you-go** support automatic scale-out and automatic scale-in. **Subscription** clusters do not support automatic scale-out or automatic scale-in. For more information, see Purchase a pay-as-you-go cluster.

Procedure

1.
2.
3. On the **Clusters** page, find the cluster for which you want to enable the autonomy service, and click the cluster ID.
4. In the left-side navigation pane, choose **Diagnostics and Optimization > Diagnosis**.
5. On the page that appears, click the **Autonomy Center** tab.

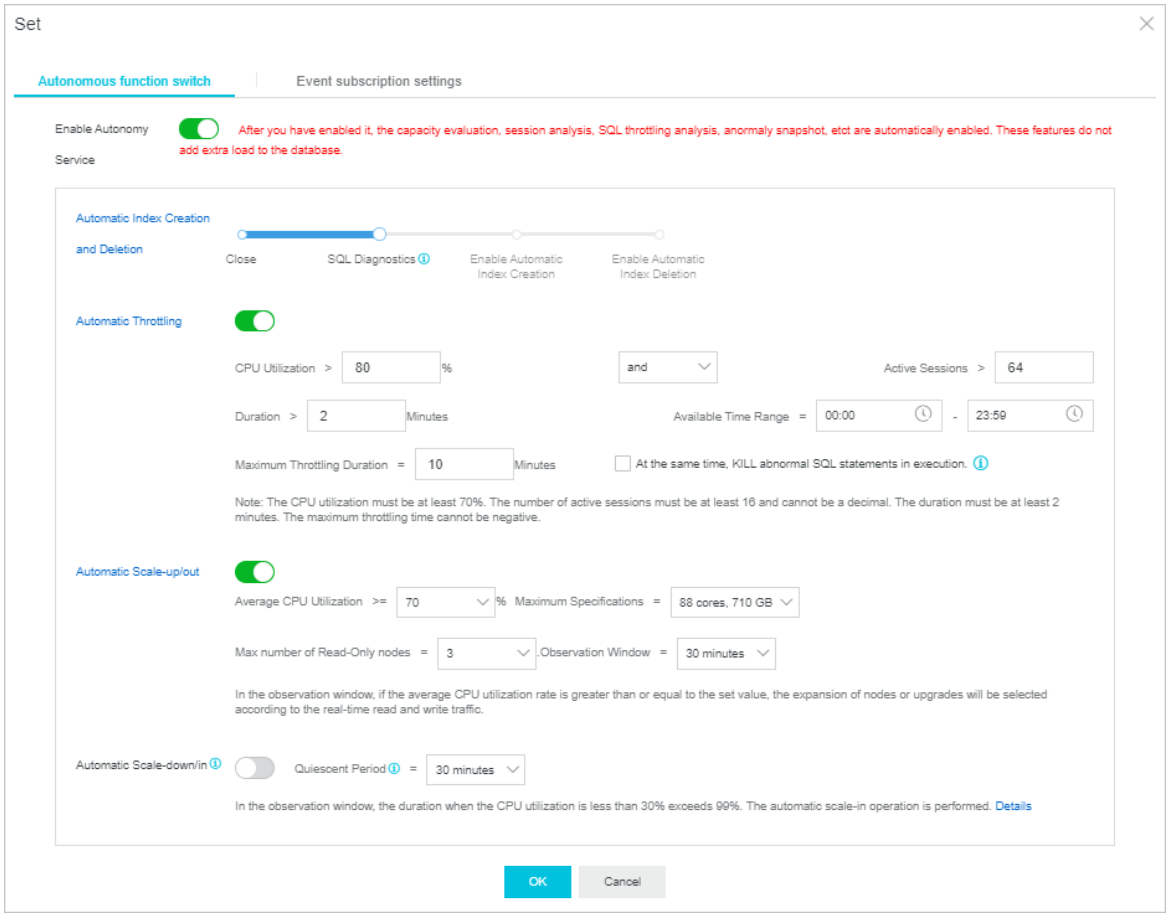


6. In the upper-right corner, click **Autonomy Service Settings**.



7. In the **Set** dialog box, click the **Autonomy Service Settings** tab and turn on **Enable Autonomy**.

> **Note** After you **enable the autonomy service**, capacity evaluation, session analytics, throttling SQL analytics, and snapshots upon exceptions are automatically performed. These features do not cause extra loads to the database.



8. Specify the following parameters: **Automatic Index Creation and Deletion**, **Automatic Throttling**, **Automatic Scale-up/out**, and **Automatic Scale-down/in**.
   - **Automatic Index Creation and Deletion**: After you turn on **Enable Autonomy**, the **SQL Diagnostics** feature is automatically enabled. You can click **Enable Automatic Index Creation** to automatically create indexes and click **Enable Automatic Index Deletion** to automatically delete indexes.
   - **Automatic Throttling**: You can specify conditions to trigger automatic SQL throttling. If the specified conditions are met, automatic SQL throttling is triggered.

     > **Note** For example, automatic throttling is triggered if the following conditions are met during the time period specified by the Current limiting period parameter (default value: 00:00 to 23:59): The CPU usage is greater than 70%, the number of active sessions is greater than 16, and the duration is at least 2 minutes. In this case, the system automatically starts to check whether the conditions are met again when the automatic throttling is triggered. If the issue is not fixed, the system automatically rolls back the throttling operation. After automatic throttling is triggered, the duration of the throttling operation does not exceed the time specified by the Maximum current limiting time parameter. For more information, see Automatic SQL throttling.

   - **Automatic Scale-up/out** and **Automatic Scale-down/in**: you can enable these features and customize conditions. Scale-in and scale-out are automatically triggered when the specified conditions are met.

Parameter	Description
Auto Scaling-out	Specifies whether to enable the auto scale-out feature.
Observation Period	If the CPU utilization is greater than or equal to the specified value during the observation period, automatically adds nodes or upgrades the specifications of your cluster after the observation period expires. This ensures that the cluster can handle the incoming read and write requests. For example, if the observation period is 5 minutes and the time required to complete a scaling activity is 10 minutes, you must wait 15 minutes before you can check the scaling result.
CPU Usage	The threshold that is used to trigger upgrades or scale-out activities. If the **CPU Usage** is greater than or equal to the specified value, an auto scale-out activity is triggered.
Maximum Specification	The highest specifications to which a cluster can be upgraded. After an upgrade is triggered, the specifications of a cluster is upgraded tier by tier until the highest specifications are applied. For example, the cluster is upgraded from 4 cores to 8 cores, and then to 16 cores.
Maximum Number of Read-only Nodes	The maximum number of read-only nodes that can be automatically added to the cluster. After a scale-out activity is triggered, read-only nodes are automatically added to a cluster one after another until the specified upper limit is reached. **Note** The nodes that are automatically added are associated with the default cluster endpoint. If the cluster uses a custom endpoint, you must specify whether to associate these nodes with the endpoint by specifying the **Automatically Associate New Nodes** parameter. For more information about **Automatically Associate New Nodes**, see Configure PolarProxy.
Auto Scaling-in	Specifies whether to enable the auto scale-in feature. **Note** After you turn on Auto Scaling-in, if the CPU utilization remains at less than 30% for more than 99% of the silent period, an automatic scale-in activity is triggered after the silent period ends. The specifications of the cluster are scaled in to the original specifications in small increments.

Parameter	Description
Quiescent Period	The minimum interval between two scaling activities. During a silent period, monitors the resource usage of a cluster but does not trigger scaling activities. If a quiescent period and an observation period end at the same time and the CPU utilization reaches the threshold value within the observation period, automatically triggers the auto scaling operation.

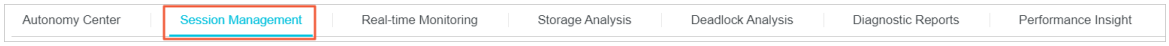
9. Click **OK**.

18.3. Session Management

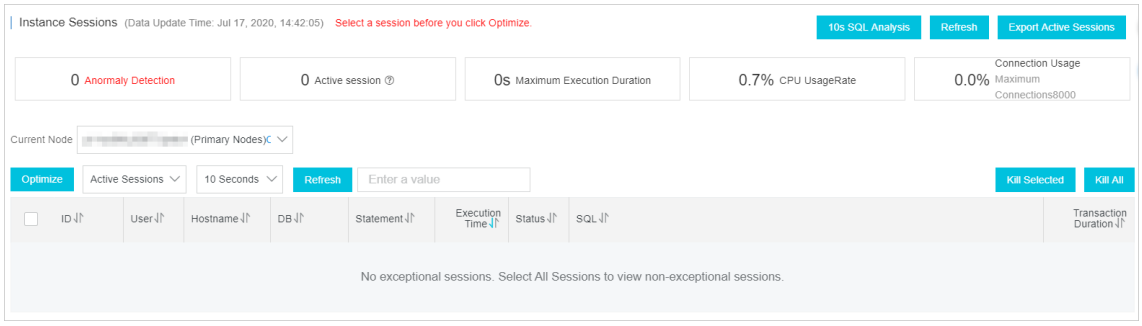
This topic describes the diagnosis feature provided by . The diagnosis feature integrates with Database Autonomy Service (DAS) and allows you to use the session management feature to view the session statistics of a target instance.

Procedure

1.
2.
3. On the **Clusters** page, click the ID of the target cluster.
4. In the left-side navigation pane, choose **Diagnostics and Optimization > Diagnosis**.
5. Click the **Session Management** tab.

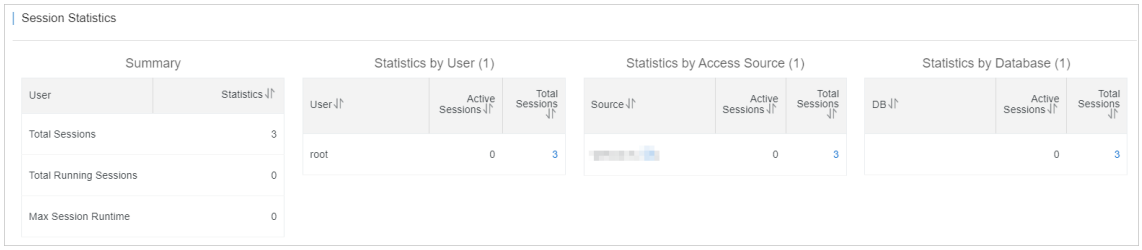


6. On the **Session Management** page, you can view the **Instance Sessions** and **Session Statistics** sections.
   - **Instance Sessions**: provides information about exceptional sessions, active sessions, and CPU usage.



> **Note**
>
> - In the upper-right corner of the Diagnosis page, you can click **10s SQL Analysis** to view the most frequently executed SQL statements and slow queries within 10 seconds. For more information, see 10-Second SQL Analysis.
> - You can click **SQL Throttling** to control the number of database requests and the number of SQL concurrent requests to ensure the availability of services. For more information, see SQL Throttling .
> - You can click **Optimize** to diagnose and optimize SQL statements. For more information, see SQL optimization.

- **Session Statistics**: provides the total number of sessions, session runtime, and other session statistics collected based on different metrics. For example, the system calculates the number of sessions initiated by different users.



18.4. Real-time Monitoring

This topic describes the diagnosis feature provided by . The diagnosis feature integrates with Database Autonomy Service (DAS) and allows you to use the real-time monitoring feature to monitor the queries per second (QPS), transactions per second (TPS), and network traffic transmitted through a cluster.

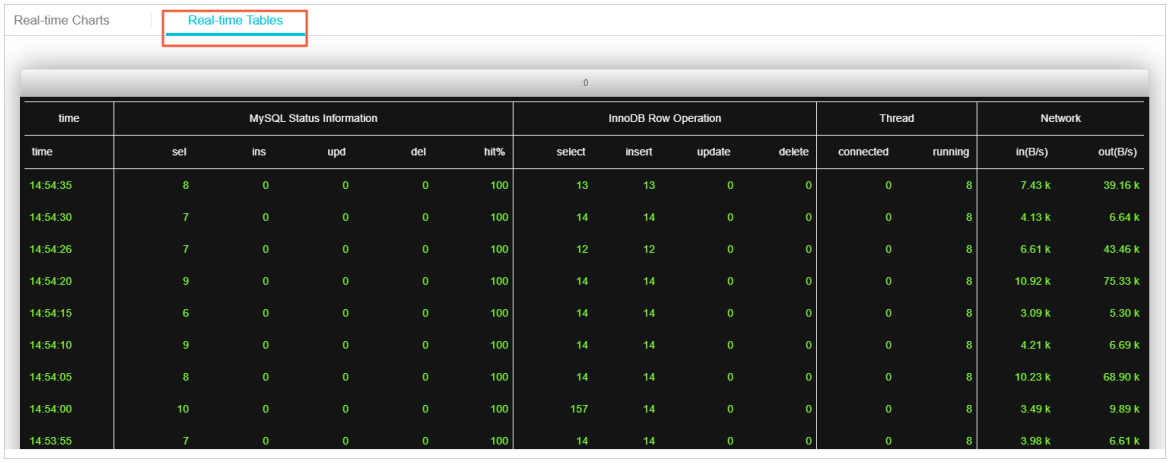
Procedure

1.
2.
3. On the **Clusters** page, click the ID of the target cluster.
4. In the left-side navigation pane, choose **Diagnostics and Optimization > Diagnosis**.
5. On the **Real-time Monitoring** page, you can view performance details, such as the server information and buffer pool information about different nodes in the cluster.

> **Note** You can click **Metric Description** to view detailed descriptions of parameters.

6. You can click the **Real-time Charts** tab to view the line charts of performance parameters. This allows you to monitor the trends of performance.

> **Note** To view specific performance values, click the **Real-Time Tables** tab.



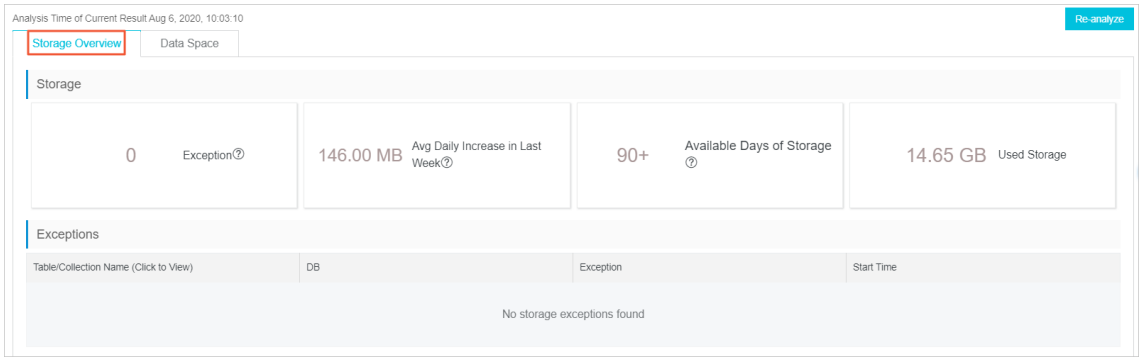
18.5. Storage analysis

The diagnosis feature of integrates with Database Autonomy Service (DAS). This allows you to view database diagnostics and optimization results on multiple tabs. On the Storage Analysis tab, you can check the storage overview for a target instance. For example, you can view the number of available days of storage, tablespace usage status, fragmentation percentage, and exception analysis results.

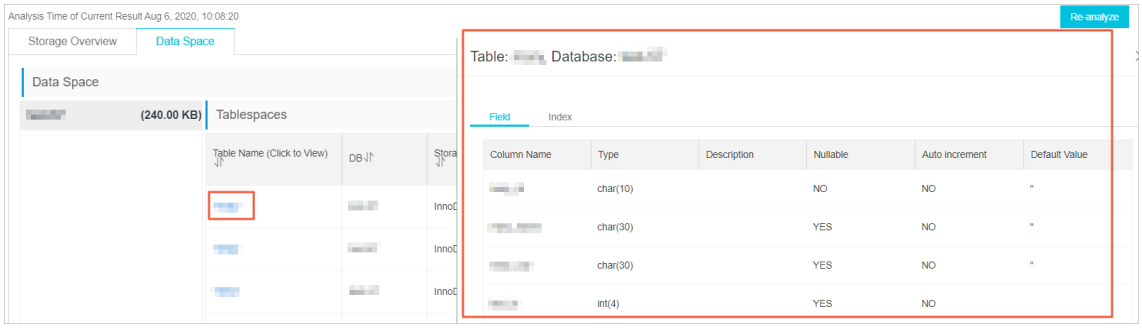
Procedure

1.
2.
3. On the **Clusters** page, find the cluster for which you want to enable the autonomy service, and click the cluster ID.
4. In the left-side navigation pane, choose **Diagnostics and Optimization > Diagnosis**.
5. Click the **Storage Analysis** tab.

6. On the **Storage Analysis** tab, you can view the following information.
   - By default, the **Space Overview** tab appears. On this tab, you can check tablespaces, storage trend, and other related storage data.



- Click the **Data Space** tab. On this tab, you can check the storage conditions for individual databases and tables. Click a table name to check the fields and indexes of the table.

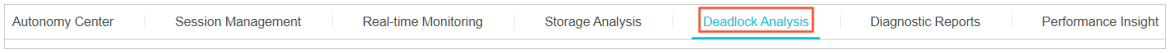


18.6. Deadlock analysis

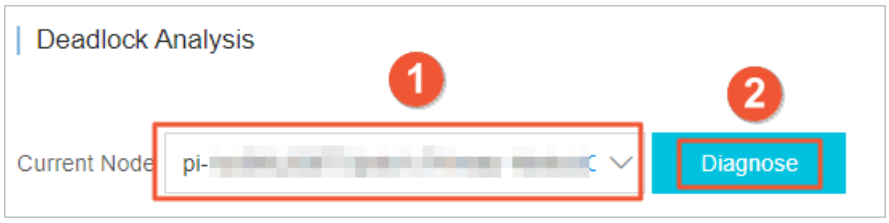
The diagnosis feature of integrates with Database Autonomy Service (DAS). This allows you to view database diagnostics and optimization results on multiple tabs. On the Deadlock Analysis tab, you can analyze the latest deadlock and check the analysis details.

Procedure

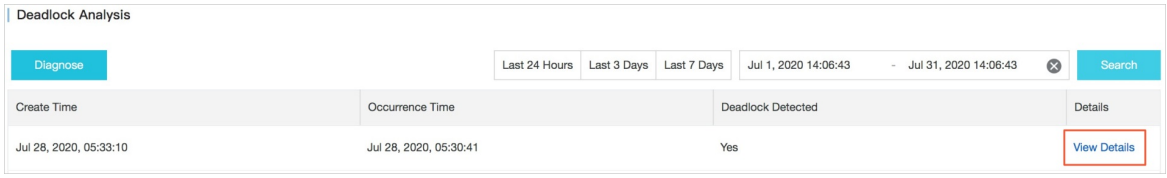
1.
2.
3. On the **Clusters** page, find the cluster for which you want to enable the autonomy service, and click the cluster ID.
4. In the left-side navigation pane, choose **Diagnostics and Optimization > Diagnosis**.
5. Click the **Deadlock Analysis** tab.



6. On the **Deadlock Analysis** tab, select a target instance ID from the **Current Node** drop-down list, and click **Diagnose**.

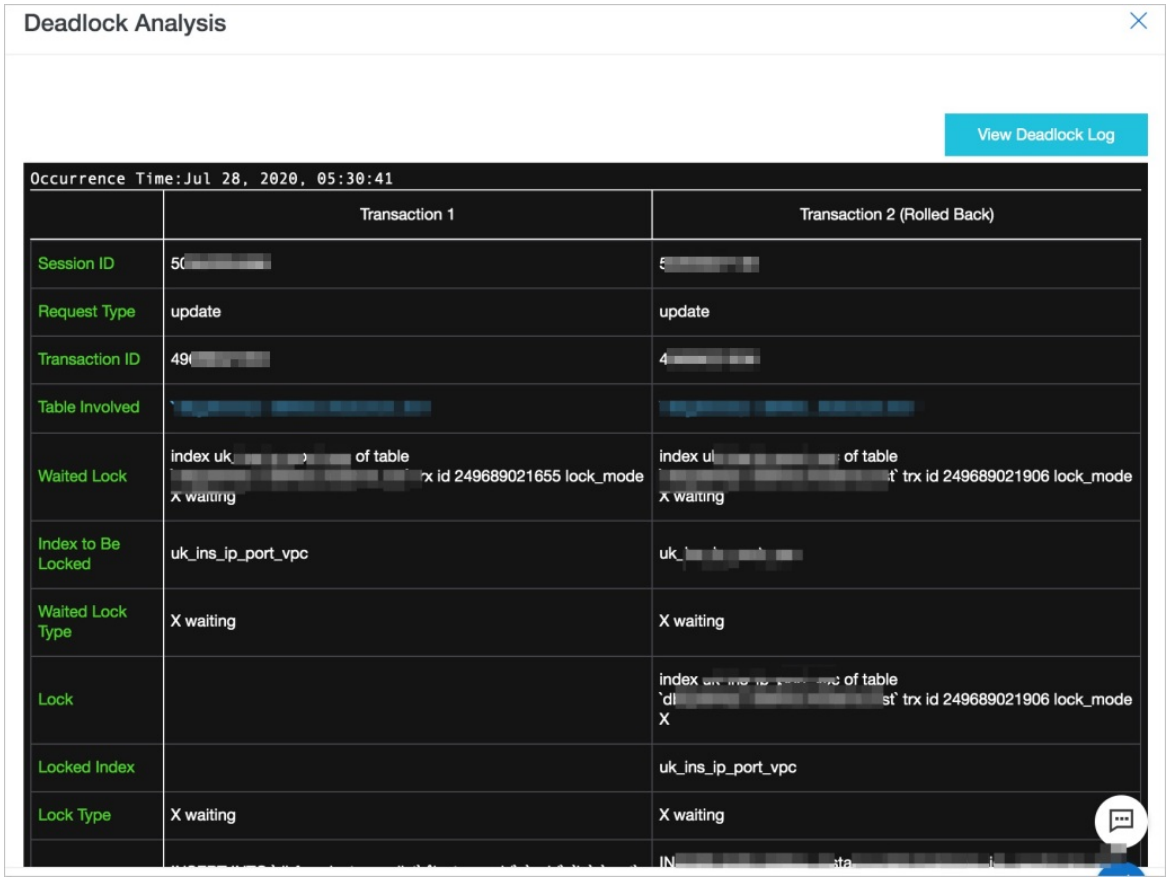


7. Click **View Details** in the **Details** column for the latest deadlock that is detected.



> **Note** The **View Details** button is available only when the **Deadlock Detected** column for the same item appears in the **Yes** state.

8. In the **Deadlock Analysis** dialog box, check the deadlock analysis results. You can click **View Deadlock Log** to view the details of the latest deadlock log.



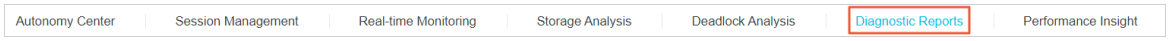
18.7. Diagnostic reports

The diagnosis feature of integrates with Database Autonomy Service (DAS). This allows you to view database diagnostics and optimization results on multiple tabs. On the Diagnostic Reports tab, you can view and customize the diagnostic reports that are generated based on the specified conditions.

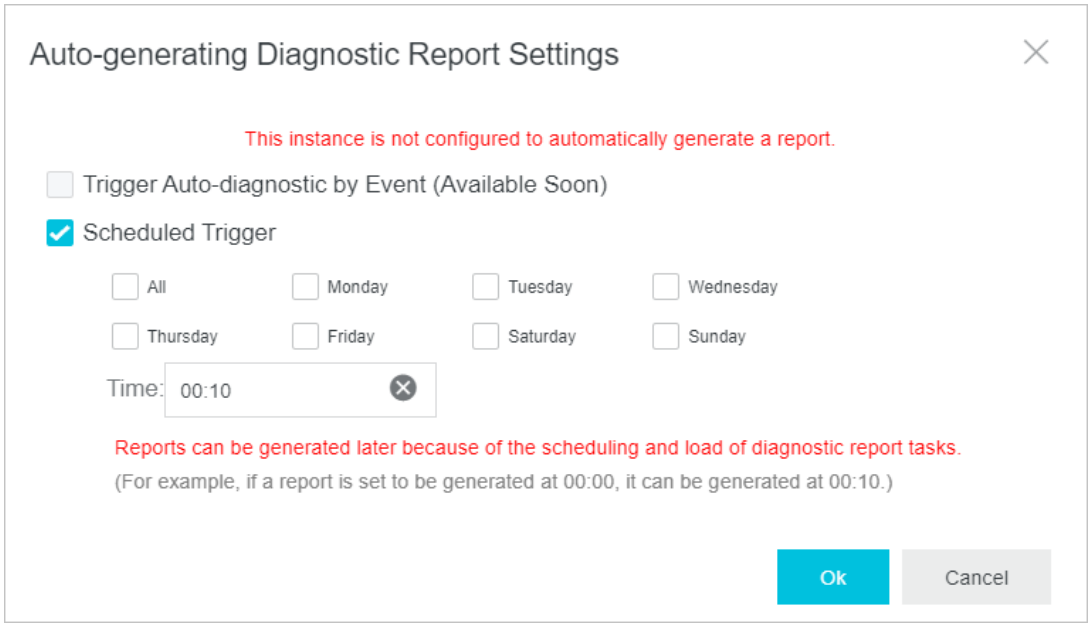
Procedure

1.
2.
3. On the **Clusters** page, find the cluster for which you want to enable the autonomy service, and click the cluster ID.
4. In the left-side navigation pane, choose **Diagnostics and Optimization > Diagnosis**.

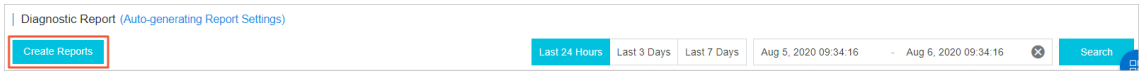
5. Click the **Diagnostic Reports** tab.



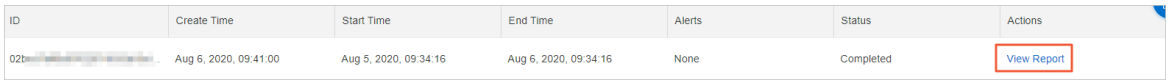
6. On the **Diagnostic Reports** tab, you can generate diagnostic reports by using the following methods:
   - **Automatically generate diagnostic reports**:
     a. Click **Auto-generating Diagnostic Report Settings** on the Diagnostic Reports tab.
     b. In the dialog box that appears, select **Scheduled Trigger**.
     c. Select **All** or other check boxes to specify one or more dates, and set the time that diagnostic reports are generated.
     d. Click **OK**.



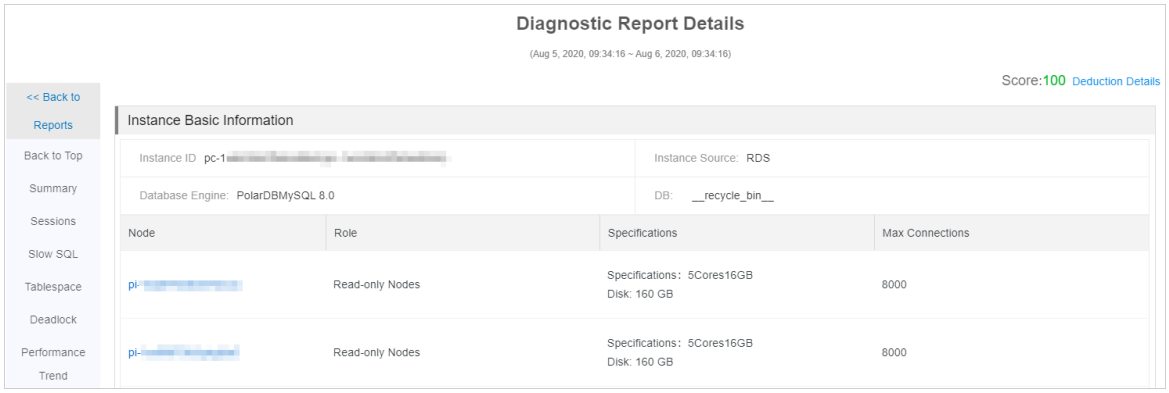
   - **Manually generate diagnostic reports**: Click **Create Reports** to manually generate diagnostic reports.



7. Find the completed diagnostic report, and click **View Report** in the **Actions** column for the report.



8. In the diagnostic report that appears, you can view the diagnostic results for the instance, including the health status, sessions, deadlocks, and performance trends.



18.8. Performance Insight

provides the diagnostics feature that integrates some features of Database Autonomy Service (DAS). You can use the Performance Insight feature to rapidly evaluate database loads and identify the root causes of performance issues. This helps you improve the database stability.

Prerequisites

The service edition of your clusters is or . This feature is unavailable for . For more information about service editions, see Overview.

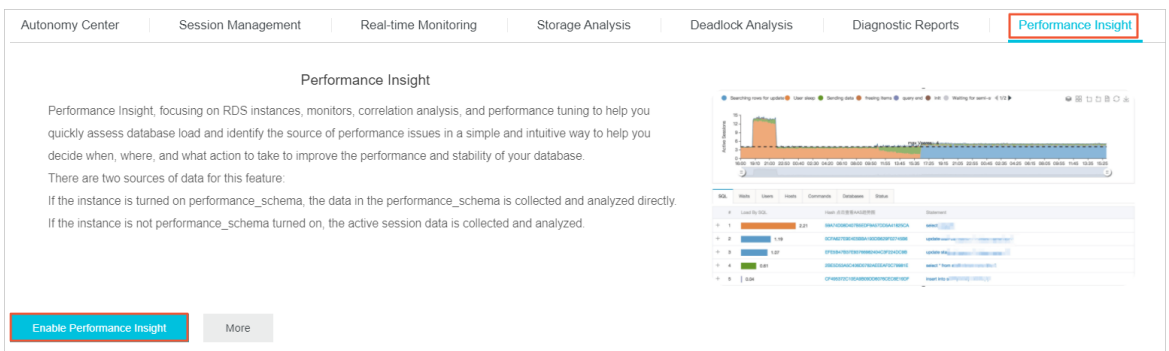
Background

The Performance Insight feature supports the following data sources:

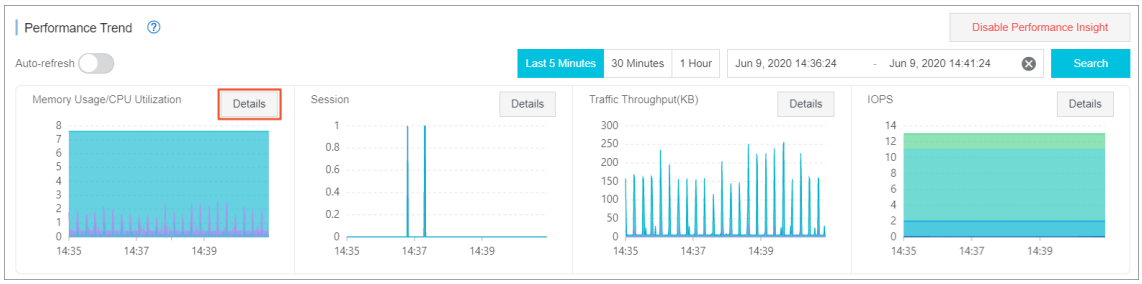
- If performance_schema is enabled for the desired instances, the Performance Insight feature directly collects and analyzes the data stored in performance_schema.
- If performance_schema is disabled for the desired instances, the Performance Insight feature collects and analyzes the data of active sessions.

Procedure

1.
2.
3. On the **Clusters** page, find the cluster for which you want to enable the autonomy service, and click the cluster ID.
4. In the left-side navigation pane, choose **Diagnostics and Optimization > Diagnosis**.
5. Click the **Performance Insight** tab.
6. Click **Enable Performance Insight**.

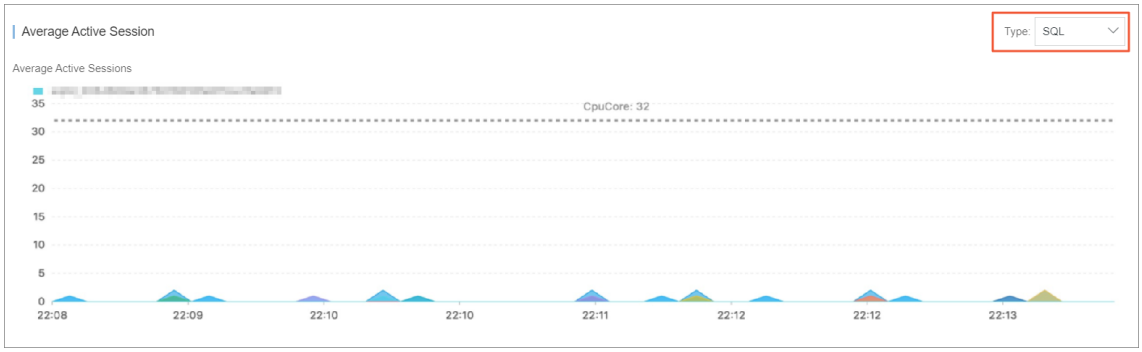


7. In the dialog box that appears, click **Confirm**.
8. On the Performance Insight tab, view and manage the following information:
   - In the **Performance Trend** section, you can specify a time range to view the performance of databases. If you need to view a specific performance metric, such as CPU usage, click **Details** next to the performance metric name.



> **Note** The duration of the specified time range cannot exceed seven days.

   - In the **Average Active Session** section, you can view the trend charts of different types of sessions, such as SQL, and the relevant multidimensional details of service loads. This helps you identify the root causes of performance issues.



18.9. Performance monitoring

18.9.1. View performance monitoring data

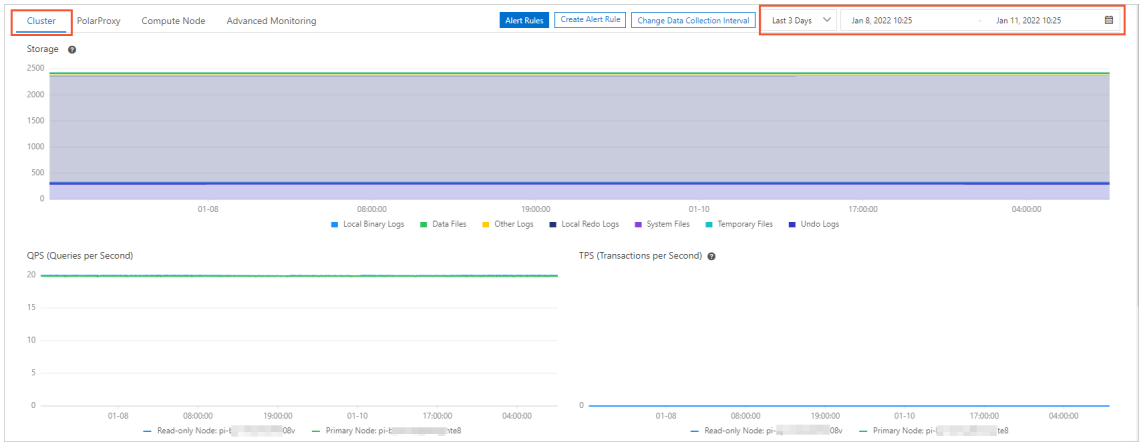
The console allows you to monitor a variety of performance metrics and view monitoring data at intervals of seconds. You can monitor the status of your clusters and locate faults based on the monitoring data.

Procedure

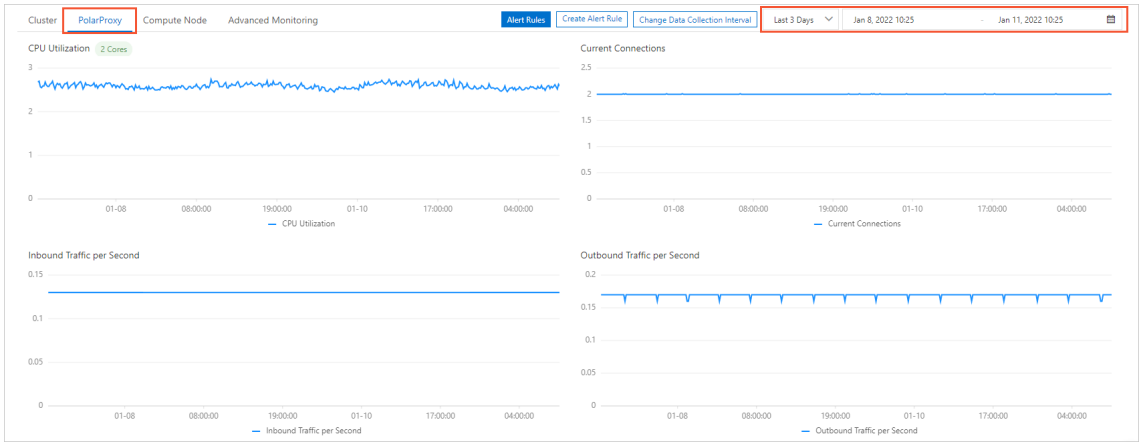
1.
2.
3.
4. In the left-side navigation pane, choose **Diagnostics and Optimization > Monitoring**.
5. View the monitoring information about the cluster on the **Cluster**, **PolarProxy**, **Compute Node**,

or **Advanced Monitoring** tab. For more information, see Metric description.

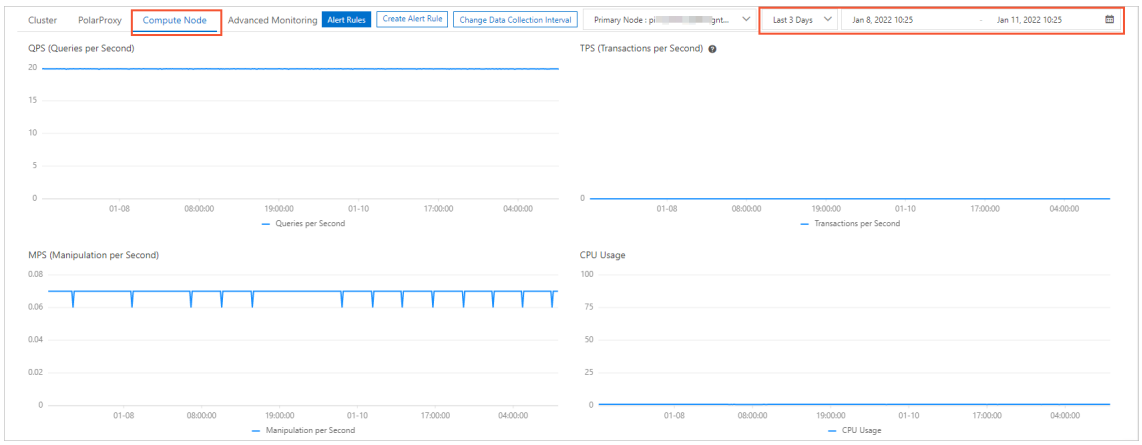
- To monitor cluster performance, click the **Cluster** tab, specify a period of time on the right side of the page, and then click **OK**.



- To monitor database proxy performance, click the **PolarProxy** tab, specify a period of time on the right side of the page, and then click **OK**.



- To monitor compute node performance, click the **Compute Node** tab, select a node from the drop-down list on the right side of the page, specify a period of time in the text box next to the drop-down list, and then click **OK**.



Note On the Real-time Monitoring Dashboard page in the Database Autonomy Service (DAS) console, you can view the monitoring data about . This can help you identify abnormal clusters for further analytics and optimization.

Metrics

Category	Metric	Description
Cluster	Storage	Displays the sizes of log files such as binary log and redo log files, and the storage used by data files, system files, and temporary files.
	QPS	Displays the queries per second (QPS) of each node. The values are sourced from the `Questions` metric in the output of the `show global status` statement.
	TPS	Displays the transactions per second (TPS) of each node. The values are sourced from the `Com_commit` and `Com_rollback` metrics in the output of the `show global status` statement.
	MPS	Displays the manipulations per second (MPS) of each node. The values are sourced from the `Com_insert`, `Com_insert_select`, `Com_update`, `Com_update_multi`, `Com_delete`, `Com_delete_multi`, `Com_replace`, and `Com_replace_select` metrics in the output of the `show global status` statement.
	CPU Usage	Displays the CPU utilization of each node.
	Memory Usage	Displays the memory usage of each node.
	Connections	Displays the number of active connections per second on each node. The values are sourced from the statistical information in the output of the `show processlist` statement.
	QPS	Displays the QPS of the selected node. The values are sourced from the `Questions` metric in the output of the `show global status` statement.
	TPS	Displays the TPS of the selected node. The values are sourced from the `Com_commit` and `Com_rollback` metrics in the output of the `show global status` statement.
	MPS	Displays the manipulations per second (MPS) of each node. The values are sourced from the `Com_insert`, `Com_insert_select`, `Com_update`, `Com_update_multi`, `Com_delete`, `Com_delete_multi`, `Com_replace`, and `Com_replace_select` metrics in the output of the `show global status` statement.
	CPU Usage	Displays the CPU utilization of the selected node.

Category	Metric	Description
Compute node	Memory Usage	Displays the memory usage of the selected node.
	Connections	Displays the total number of connections and the number of active connections on the selected node. The values are sourced from the statistical information in the output of the `show processlist` statement.
	Operations	Displays the number of operations performed per second on the selected node. The operations counted include the DELETE, INSERT, UPDATE, and REPLACE operations. The values are sourced from the metrics whose names start with `Com_` in the output of the `show global status` statement.
	Memory Buffer Pool	Displays the dirty ratio, read hit ratio, and usage of the buffer pool on the selected node. The values are sourced from `Innodb_buffer_pool_pages_dirty`, `Innodb_buffer_pool_pages_total`, `Innodb_buffer_pool_reads`, `Innodb_buffer_pool_read_requests`, and other metrics in the output of the `show global status` statement.
	I/O Throughput	Displays the total I/O throughput, read I/O throughput, and write I/O throughput of the selected node.
	IOPS	Displays the total IOPS, read IOPS, and write IOPS of the selected node.
	Network	Displays the input traffic per second and output traffic per second of the selected node. The values are sourced from the `Bytes_received` and `Bytes_sent` metrics in the output of the `show global status` statement.
	Scanned Rows	Displays the numbers of rows that are inserted, read, updated, and deleted per second on the selected node. The values are sourced from the `Innodb_rows_deleted`, `Innodb_rows_inserted`, `Innodb_rows_read`, and `Innodb_rows_updated` metrics in the output of the `show global status` statement.
	InnoDB Read and Written Data	Displays the amount of data that is read from or written into the storage engine per second on the selected node. The values are sourced from the `Innodb_data_read` and `Innodb_data_written` metrics in the output of the `show global status` statement.
	InnoDB Buffer Pool Requests	Displays the number of read operations and the number of write operations that are performed per second on the buffer pool of the selected node. The values are sourced from the `Innodb_buffer_pool_read_requests` and `Innodb_buffer_pool_write_requests` metrics in the output of the `show global status` statement.

Category	Metric	Description
	InnoDB Log Writes	Displays the number of log write requests per second and the number of times that data is synchronized to disks per second on the selected node. The values are sourced from the `Innodb_log_write_requests` and `Innodb_os_log_fsyncs` metrics in the output of the `show global status` statement.
	Temporary Tables	Displays the number of temporary tables that are created per second on the selected node. The values are sourced from the `Created_tmp_disk_tables` metric in the output of the `show global status` statement.

FAQ

- Why is the QPS that is displayed on the Monitoring page approximately 10 when the service traffic of my cluster is 0?

  The monitoring, log collection, and administration tasks that run in the system background generate approximately 10 queries per second. This has little impact on the performance of your cluster.

- How do I reduce the CPU utilization when the CPU utilization is excessively high?

  We recommend that you perform the following steps:

  - Check whether a large number of slow requests exist. If a large number of slow requests exist, we recommend that you optimize slow SQL statements first. For more information about how to view and optimize slow SQL statements, see Slow SQL query.
  - Check whether the trend of the CPU utilization curve is consistent with that of the QPS or TPS curve. If the curves are consistent, the issue is caused by high concurrency of transactions. In this case, we recommend that you upgrade the configuration of your cluster. For more information about how to upgrade the configuration of a cluster, see Manually upgrade or downgrade a PolarDB cluster.
  - If no slow requests exist and the trend of the CPU utilization curve is inconsistent with that of the QPS or TPS curve, for technical support.

- What do I do when the number of connections is much greater than the number of active connections?

  You can specify smaller values for the wait_timeout and interactive_timeout parameters to accelerate the release of idle connections. We recommend that you close the connections that are no longer in use in time to reduce the number of idle connections.

Related API operations

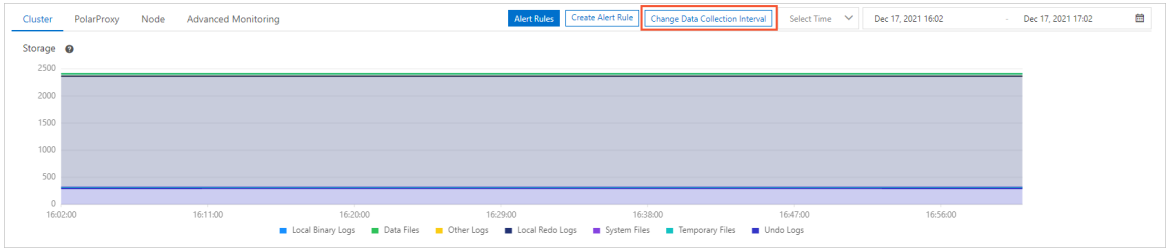
API	Description
DescribeDBClusterPerformance	Queries the performance data of a specified cluster.
DescribeDBNodePerformance	Queries the performance data of a node in a specified cluster.
DescribeDBClusterMonitor	Queries the interval at which the monitoring data of a specified cluster is collected.

18.9.2. Change the interval at which monitoring data is collected

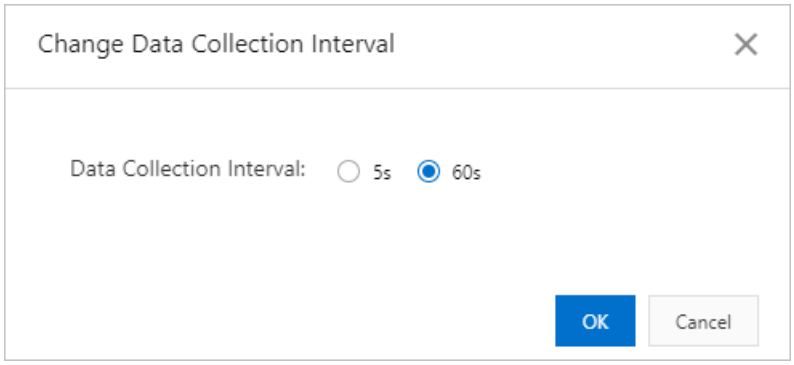
This topic describes how to change the interval at which monitoring data is collected.

Procedure

1.
2.
3.
4. In the left-side navigation pane, choose **Diagnostics and Optimization > Monitoring**.
5. Click **Change Data Collection Interval**.



6. In the **Change Data Collection Interval** dialog box, set **Data Collection Interval** to **5s** or use the default value **60s** based on your business requirements.



   - When **Data Collection Interval** is set to **5s**, the interval at which monitoring data is displayed is determined based on the following rules:
     - If the time range within which data is queried is less than or equal to 1 hour, the monitoring data is displayed at an interval of 5s.
     - If the time range within which data is queried is less than or equal to one day, the monitoring data is displayed at an interval of 1 minute.
     - If the time range within which data is queried is less than or equal to seven days, the monitoring data is displayed at an interval of 10 minutes.
     - If the time range within which data is queried is less than or equal to 30 days, the monitoring data is displayed at an interval of 1 hour.
     - If the time range within which data is queried is more than 30 days, the monitoring data is displayed at an interval of one day.
   - When **Data Collection Interval** is set to **60s**, the interval at which monitoring data is displayed

is determined based on the following rules:

- If the time range within which data is queried is less than or equal to one day, the monitoring data is displayed at an interval of 1 minute.
- If the time range within which data is queried is less than or equal to seven days, the monitoring data is displayed at an interval of 10 minutes.
- If the time range within which data is queried is less than or equal to 30 days, the monitoring data is displayed at an interval of 1 hour.
- If the time range within which data is queried is more than 30 days, the monitoring data is displayed at an interval of one day.

7. Click **OK**.

Related API operations

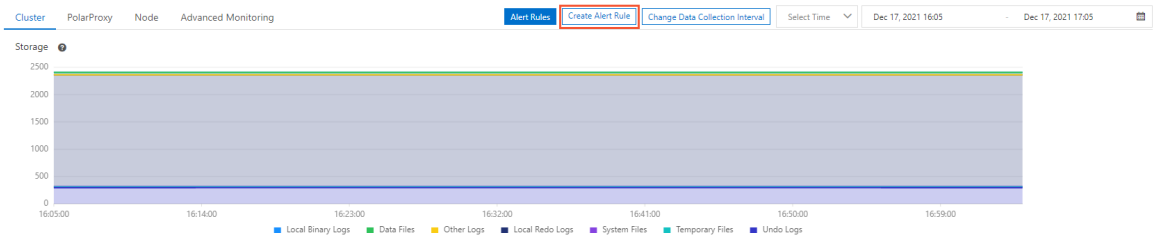
Operation	Description
ModifyDBClusterMonitor	Changes the interval at which the monitoring data of a specified cluster is collected.

18.9.3. Create an alert rule

This topic describes how to create and manage rules that can be used to trigger threshold alerts in the console. This helps you identify and handle exceptions of clusters and nodes at the earliest opportunity.

Procedure

1.
2.
3.
4. In the left-side navigation pane, choose **Diagnostics and Optimization > Monitoring**.
5. Click **Create Alert Rule**.



6. On the **Create Alert Rule** page, specify the following parameters.

Step	Parameter	Description
	Product	The service that you want to monitor. Use the default value **POLARDB MYSQL CLUSTER**.

Step	Parameter	Description
Related Resource	Resource Range	The application scope of the alert rule. Set this parameter to **All Resources** or **Cluster**. **Note** ◦ If the **Resource Range** parameter is set to **All Resources**, the system sends alert notifications if one of the clusters triggers the alert. The Rule Description parameter specifies the conditions that are used to trigger the alert. ◦ If the **Resource Range** parameter is set to **Cluster**, the system sends alert notifications only if the specified cluster triggers the alert. The Rule Description parameter specifies the conditions that are used to trigger the alert.
Set Alert Rules	Alert Rule	The name of the alert rule.
	Rule Description	The content of the alert rule. This parameter specifies the conditions that are used to trigger the alert. **Note** For more information about how to create alert rules, see Create a threshold-triggered alert rule.
	Mute for	The interval at which the system resends the alert notification if the issue that triggers the alert persists. The minimum value is 5 minutes and the maximum value is 24 hours.
	Effective Period	The validity period of the alert rule. **Note** The system sends alert notifications only within the validity period of an alert rule and records events when the validity period expires.
Notification Method	For more information about how to specify **Notification Method**, see Create a threshold-triggered alert rule.	

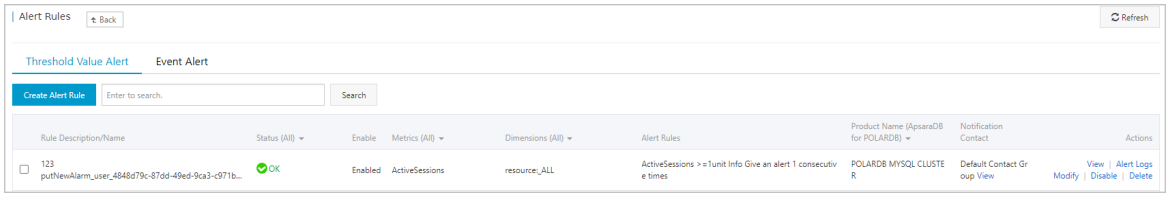
7. Click **Confirm**.

18.9.4. Manage alert rules

This topic describes how to manage alert rules that are based on threshold values in the console. The alert feature helps you detect exceptions of clusters and nodes and handle the exceptions in time.

Procedure

1.
2.
3.
4. In the left-side navigation pane, choose **Diagnostics and Optimization > Monitoring**.
5. Click **Alert Rules**. The **Alert Rules** page appears.



6. On the **Threshold Value Alert** tab, you can perform the following operations to manage the existing alert rules:
   - To view the basic information about an alert rule, click **View** in the **Actions** column of the alert rule.
   - To view the alert history associated with an alert rule, click **Alert Logs** in the **Actions** column of the alert rule.
   - To modify an alert rule, click **Modify** in the **Actions** column of the alert rule.
   - To disable an alert rule, click **Disable** in the **Actions** column of the alert rule.
   - To delete an alert rule, click **Delete** in the **Actions** column of the alert rule.
   - To view the alert contact group, alert contacts, and alert notification method for an alert rule, click **View** in the **Notification Contact** column of the alert rule.

18.10. Slow SQL query

provides the slow SQL analysis feature. This feature allows you to view slow log trends and statistics. You can also obtain the diagnostic results and suggestions on how to fix slow SQL queries.

Prerequisites

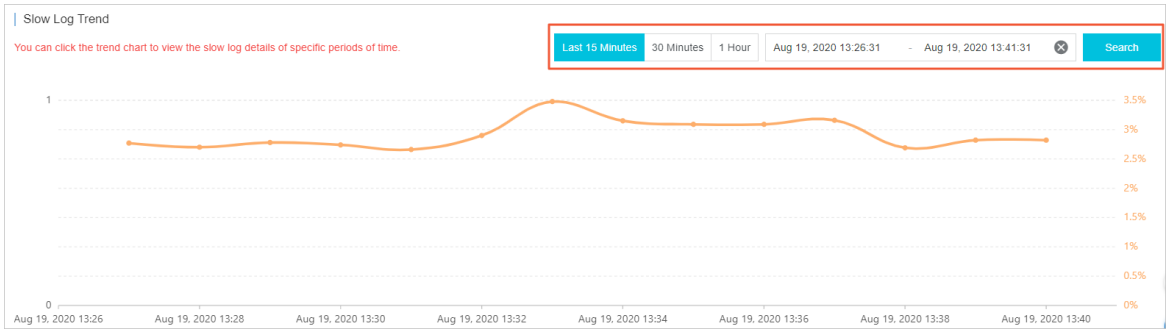
The service edition of your clusters is or . This feature is unavailable for . For more information about service editions, see Overview.

View slow log trends and statistics

1.
2.
3.
4. In the left-side navigation pane, choose **Diagnostics and Optimization > Slow SQL Query**.
5. In the **Slow Log Trend** section, use one of the following methods to view slow log trends:

- Click **Last 15 Minutes**, **30 Minutes**, or **1 Hour** to view the slow log trends within the last 15 minutes, the last 30 minutes, or the last hour.
- Specify the date range and click **Search** to view the slow log trend within at most 30 days. The interval between the end time and the start time must be within 24 hours.

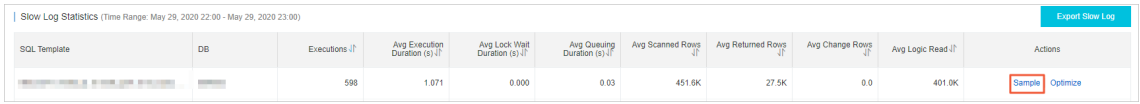
> **Note** By default, the slow log trend within the last 15 minutes is displayed.



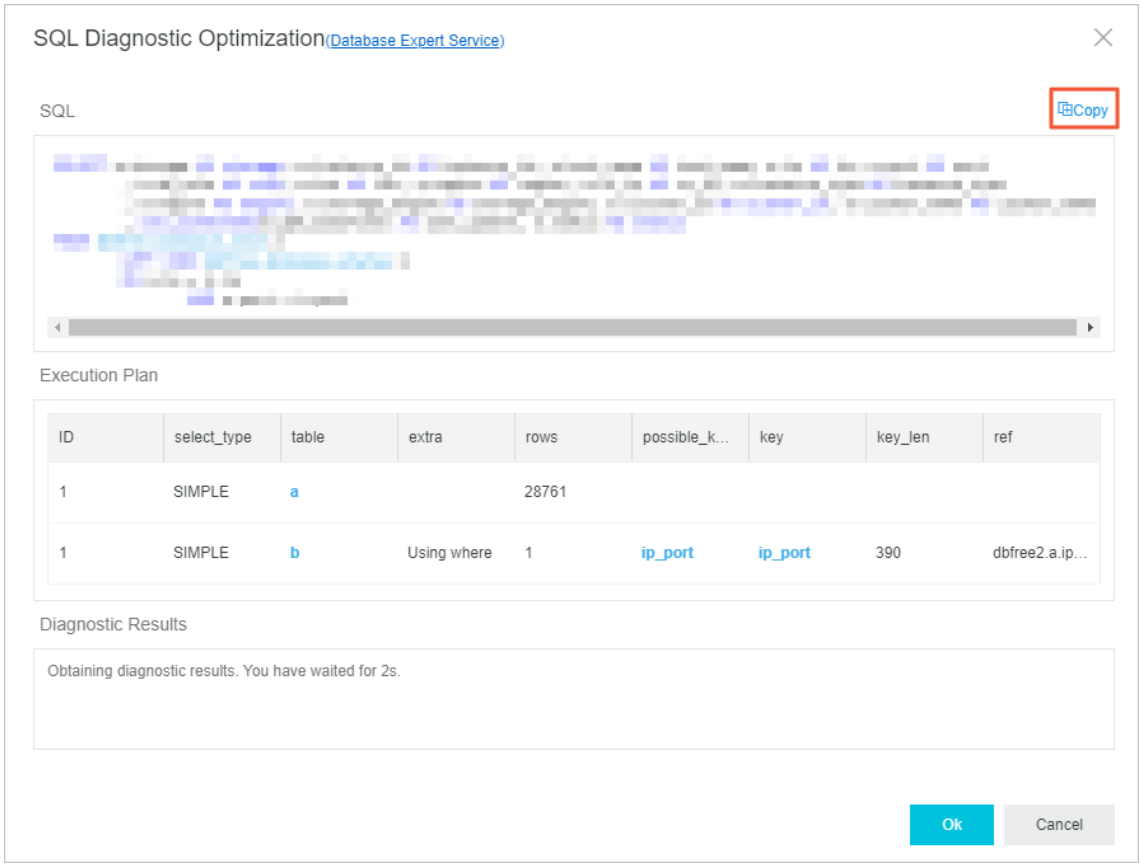
6. (Optional)To view the slow log trend of a specific node, click the ID of the node in the **Nodes** section.

7. In the **Slow Log Trend** section, click a time point in the line chart. Then, you can view the statistics of the slow logs at the point in time on the **Slow Log Statistics** tab.

8. To troubleshoot slow SQL queries, use one of the following methods:
   - Click **Sample** in the **Actions** column of the slow SQL query to view the details of the slow SQL query.



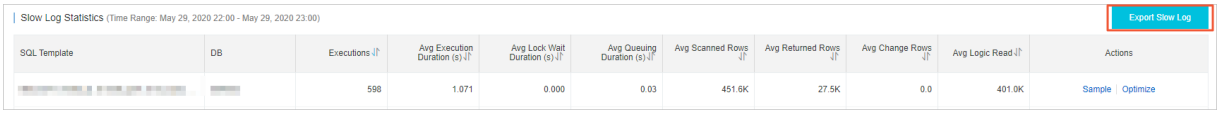
   - Click **Optimize** in the **Actions** column of the slow SQL query to view the diagnostic results and suggestions.



Note You can also click **Expert Service** to purchase the expert service. The expert service provides value-added professional database services, such as emergency solutions, health diagnostics, performance optimization, security assurance, and data migration.

Export slow logs

You can click **Export Slow Log** to export and save slow logs on your on-premises machines.



Related API operations

API	Description
DescribeSlowLogRecords	Queries the details of slow logs for a cluster.
DescribeDBClusterAuditLogCollector	Queries whether SQL data collector is enabled for a cluster. The features of SQL data collector include audit logs and SQL Explorer.
ModifyDBClusterAuditLogCollector	Enables or disables SQL data collector for a cluster. The features of the SQL data collector include audit logs and SQL Explorer.

18.11. SQL Explorer

provides the **SQL Explorer** feature that integrates the specific features of Database Autonomy Service (DAS). You can enable **SQL Explorer** to use the **search**, **SQL Explorer**, and **security audit** features of DAS Professional Edition in a convenient manner. This helps you obtain the details about SQL statements, troubleshoot performance issues, and identify the sources of high-risk SQL statements. In addition, the **traffic playback and stress testing** feature is also provided to help you verify whether your instance type needs to be scaled up or scaled out to handle traffic spikes.

Billing

You are charged for SQL Explorer based on the storage of audit logs.

- Mainland China: USD 0.0013/GB/hour
- China (Hong Kong) and regions outside China: USD 0.0019/GB/hour

> **Note** The storage of audit logs is billed based on the pay-as-you-go billing method. The subscription billing method is not supported. For more information, see Pricing of SQL Explorer (optional).

Overview

- Advanced search

  The search feature allows you to query and export SQL statements and the related information. The information includes databases, states, and execution duration. For more information, see Search.

- SQL Explorer

  The SQL Explorer feature is used to diagnose the health status of SQL statements, troubleshoot performance issues, and analyze business traffic. For more information, see SQL Explorer.

- Security auditing

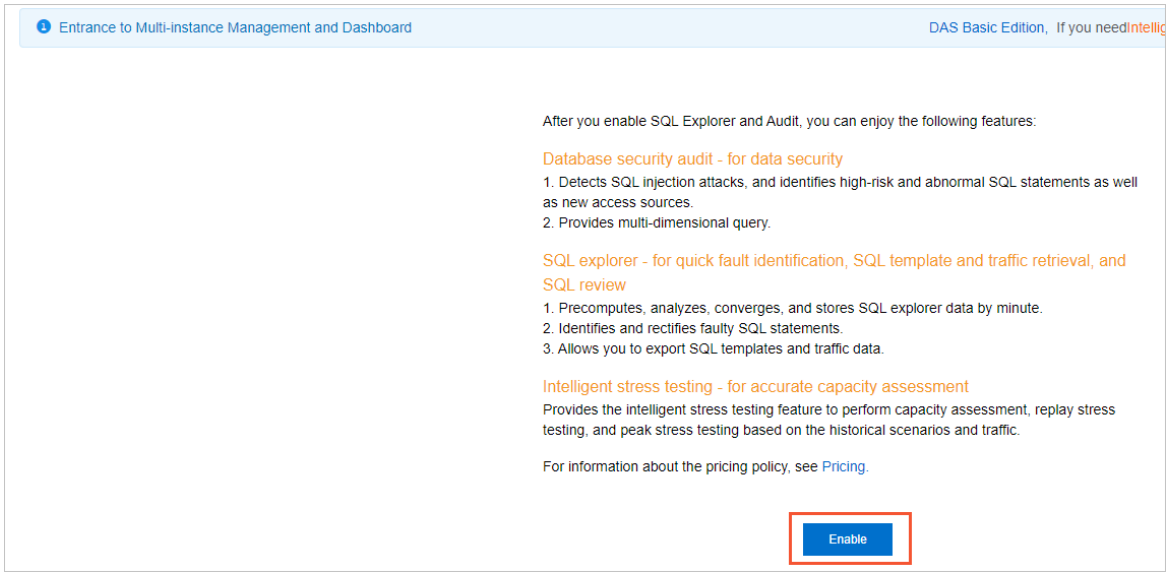
  The security audit feature is used to automatically identify risks such as high-risk SQL statements, SQL injections, and new request sources. For more information, see Security audit.

- Traffic playback and stress testing

  The traffic playback and stress test feature supports traffic playback and stress testing. You can use this feature to check whether you need to upgrade your RDS instance to handle traffic spikes during peak hours. For more information, see Traffic playback and stress testing.

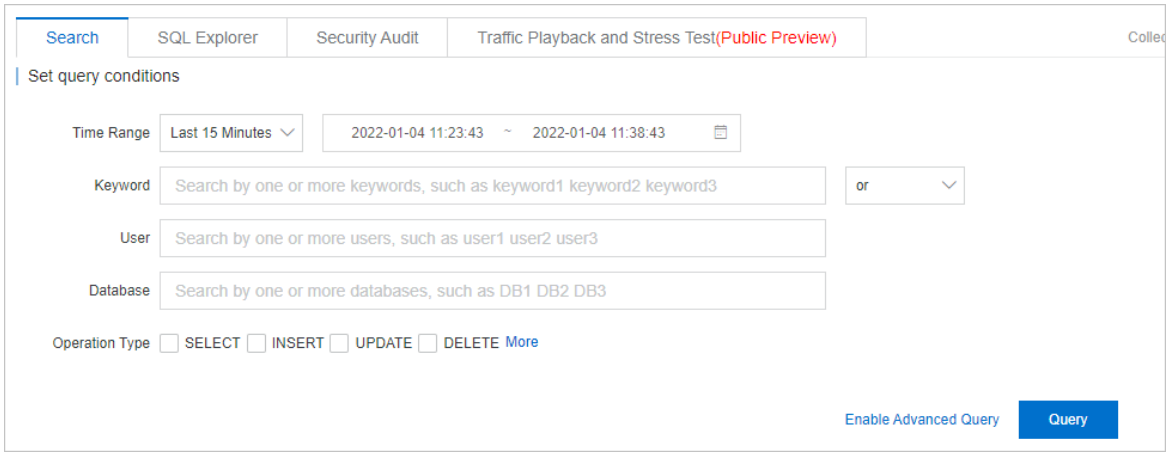
Enable the SQL Explorer feature

1.
2.
3.
4. In the left-side navigation pane, choose **Log and Audit > SQL Explorer**.
5. Click **Enable**.



- If you do not have DAS Professional Edition, click **Buy** to purchase DAS Professional Edition. Then, activate DAS Professional Edition.
- If you have DAS Professional Edition, click **Enable Professional Edition**.

After DAS Professional Edition is activated, you can use the **search**, **SQL Explorer**, **security audit**, and **traffic playback and stress testing** features of DAS Professional Edition.



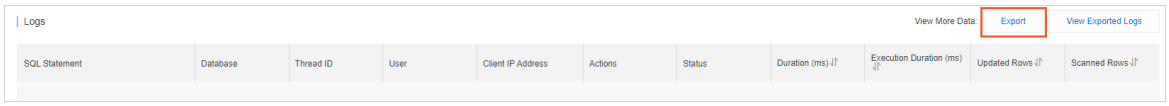
Change the retention period of SQL logs

1.
2.
3.
4. In the left-side navigation pane, choose **Log and Audit > SQL Explorer**.
5. In the upper-right corner of the SQL Explorer page, click **Service Settings**.
6. Change the retention period and click **Ok**.

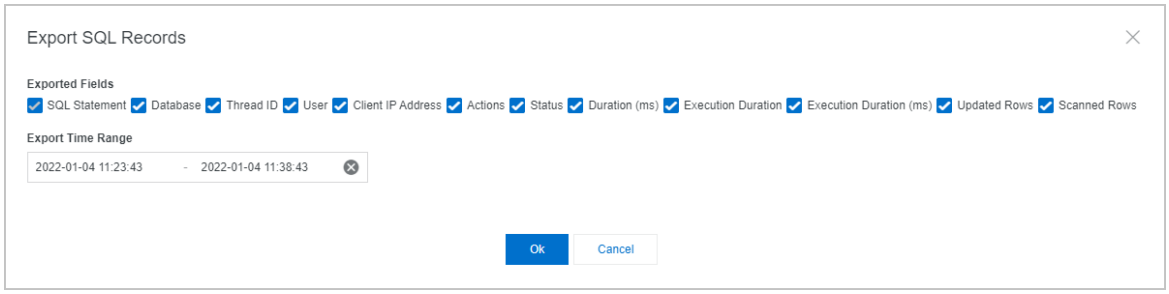
Export SQL log entries

1.
2.

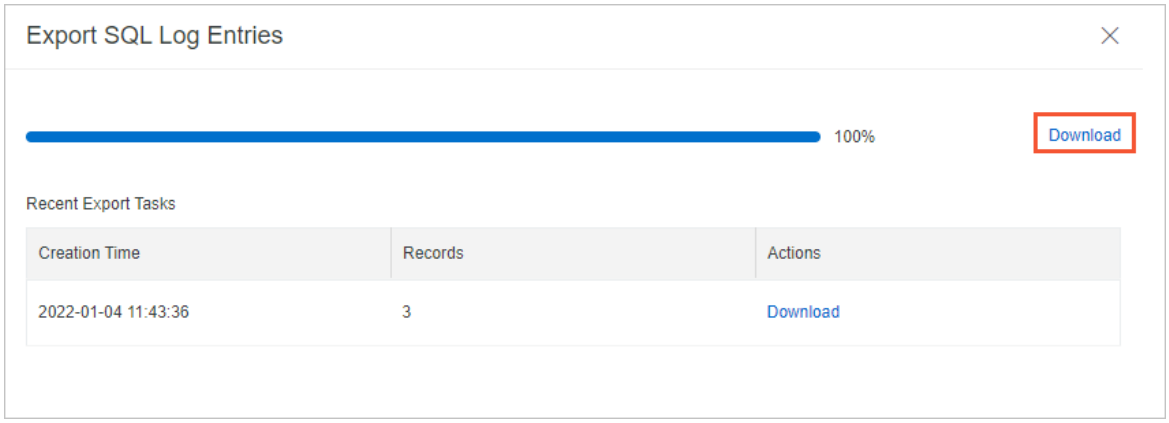
3.
4. In the left-side navigation pane, choose **Log and Audit** > **SQL Explorer**.
5. On the right side of **Logs**, click **Export**.



6. In the **Export SQL Records** dialog box, select the required fields, specify the time range, and then click **Ok**.



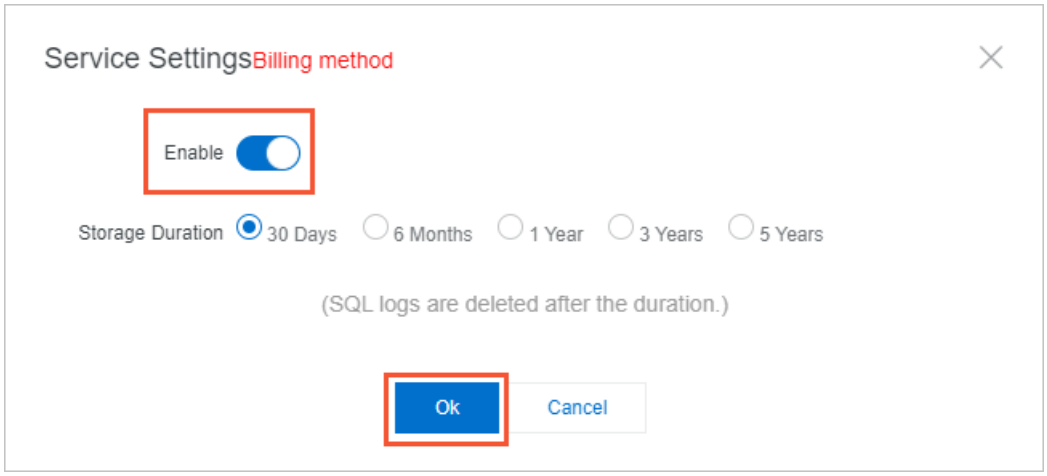
7. After log entries are exported, you can click **Download** in the **Export SQL Log Entries** panel to export SQL log entries to a `.csv` file.



8. Click **OK**.

Disable the SQL Explorer feature

1.
2.
3.
4. In the left-side navigation pane, choose **Log and Audit** > **SQL Explorer**.
5. Click **Service Settings**.
6. In the dialog box that appears, turn off the **Enable** switch and click **Ok**.



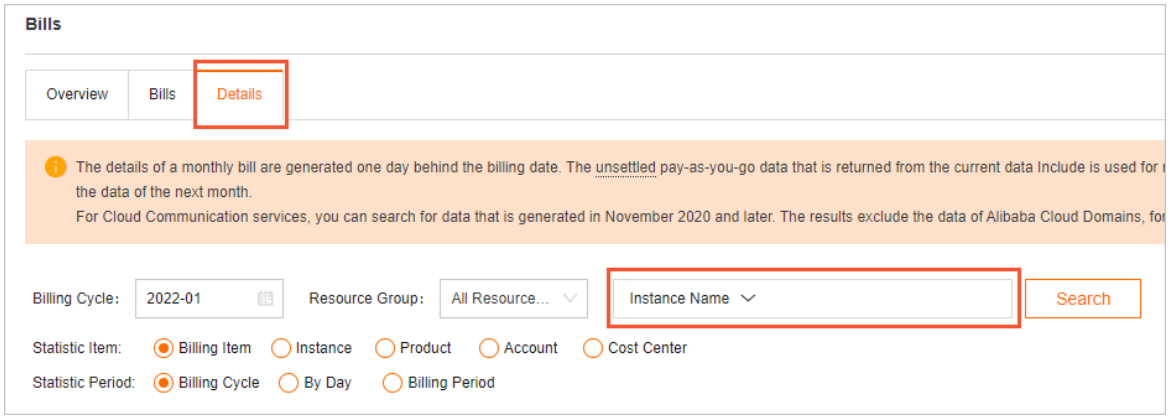
> **Note** After you disable the SQL Explorer feature, all SQL log entries are deleted. We recommend that you export SQL log entries before you disable this feature. For more information about how to export SQL log entries, see Export SQL log entries.

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

View the size and consumption details of audit logs

1. Log on to the Alibaba Cloud Management Console.
2. In the upper-right corner of the page, choose **Expenses** > **User Center**.
3. In the left-side navigation pane, choose **Spending Summary** > **Spending Summary**.
4. On the **Bills** page, click the **Details** tab. In the search bar, select **Instance ID** from the drop-down list and enter the ID of the cluster for which you want to query the details.

> **Note** To query logs generated more than 12 months ago, .



5. View the billing details in the data entries in which the value in the **Billing Item** column is **sql_explorer**.

19.Version Management

The architecture of a cluster consists of three layers: PolarProxy, the database engine, and the distributed storage. You can upgrade PolarProxy or the database engine separately or upgrade both of them at the same time.

Usage notes

- In most cases, an upgrade requires less than 30 minutes to complete. PolarProxy or the database engine is restarted during the upgrade. This can cause transient connections to occur on the database. We recommend that you perform the upgrade during off-peak hours. Make sure that your application can automatically reconnect to your database.
- During the **Upgrade PolarDB Database Proxy and Kernel** process, transient connections occur for the primary endpoint and the cluster endpoint and last for 30 to 90 seconds. Make sure that your application can automatically reconnect to your database.
- During the **Upgrade PolarDB Database Proxy Only** process, transient connections occur for the cluster endpoint and the custom endpoint and last for 30 seconds. Make sure that your application can automatically reconnect to your database.
- During the **Upgrade Kernel Only** process, clusters with PolarProxy of V2.4.7 or later can use the connection preservation technique to prevent 95% of the database connections from being interrupted.
- During the upgrade, you cannot use some features that are related to cluster changes in the console. For example, you cannot upgrade or downgrade configurations, add or delete nodes, modify parameters, or restart nodes. During this period, features related to data queries are still available, such as performance monitoring.
- You cannot downgrade PolarProxy or the database engine.

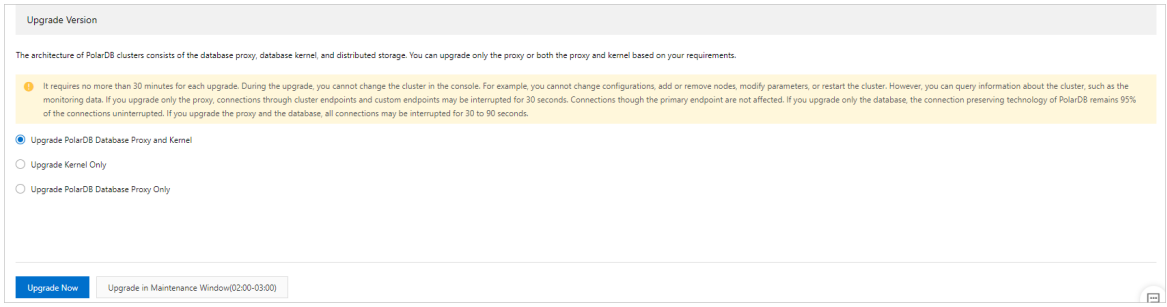
View the version information

1.
2.
3.
4. In the **Version Information** section, view the version information of PolarProxy and the database engine.

Upgrade the version

If PolarProxy or the database engine of the cluster is not of the latest version, you can upgrade the version based on your business requirements.

1. On the details page of the cluster that you want to manage, choose **Settings and Management > Version Management**. In the **Upgrade Version** section, select **Upgrade PolarDB Database Proxy and Kernel**, **Upgrade Kernel Only**, or **Upgrade PolarDB Database Proxy Only** as needed.



Note

- If PolarProxy or the database engine of the cluster is of the latest version, the **Upgrade PolarDB Database Proxy and Kernel**, **Upgrade Kernel Only**, and **Upgrade PolarDB Database Proxy Only** options are unavailable.
- If you select **Upgrade PolarDB Database Proxy Only**, only the features related to read/write splitting are upgraded, such as the consistency level, transaction splitting, and whether to offload reads from the primary node. The default consistency level is global consistency.

2. Click **Upgrade Now** or **Upgrade in Maintenance Window**.

   If you select Upgrade in Maintenance Window, you can view the details of the task or cancel the task on the **Scheduled Tasks** page. For more information, see View or cancel a scheduled task.

Notice

- During the **Upgrade PolarDB Database Proxy and Kernel** process, transient connections occur for the primary endpoint and the cluster endpoint and last for 30 to 90 seconds. Make sure that your application can automatically reconnect to your database.
- During the **Upgrade PolarDB Database Proxy Only** process, transient connections occur for the cluster endpoint and the custom endpoint and last for 30 seconds. Make sure that your application can automatically reconnect to your database.

3. In the dialog box that appears, click **OK**.

Related API operations

API	Description
DescribeDBClusterVersion	Queries the details about the database engine version of a cluster.
UpgradeDBClusterVersion	Upgrades a cluster to the latest version.

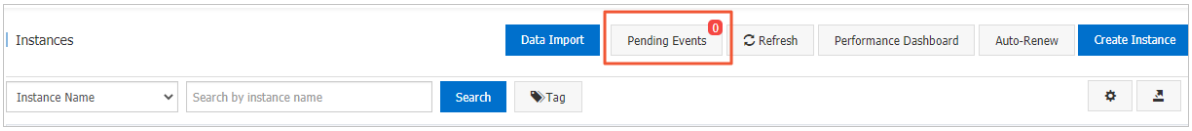
20.Scheduled O&M Events

20.1. View and manage scheduled events

For scheduled O&M events, you can be notified by text message, phone call, email, or internal message. You can also be notified in the console. Scheduled PolarDB O&M events include database software upgrade events and hardware maintenance and upgrade events. You can view the details of each scheduled event. These details include the event type, task ID, cluster name, and switching time. You can also change the switching time.

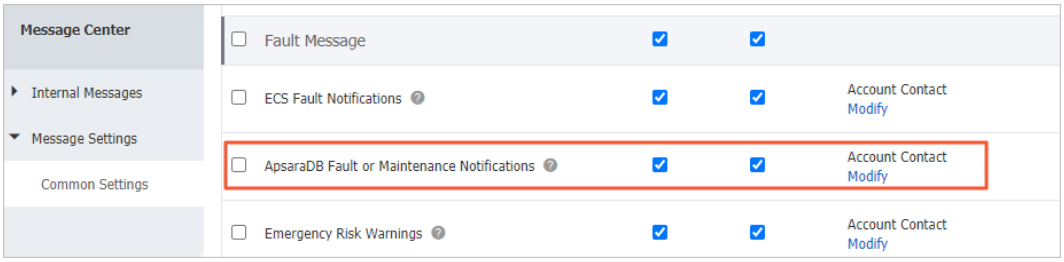
Precautions

- If you have scheduled O&M events and want to view event notifications, you can go to the left-side navigation pane in the console and choose **Event Center > Scheduled Events**.



- In most cases, you are notified of scheduled events in ApsaraDB at least three days before these events are executed. You can be notified in many ways. For example, you can be notified by phone call, email, or internal message. To use this feature, you must log on to Message Center, select **ApsaraDB Fault or Maintenance Notifications**, and then specify a contact. We recommend that you specify an O&M engineer as the contact. If you do not specify a contact, you cannot receive notifications.

  Message Center settings



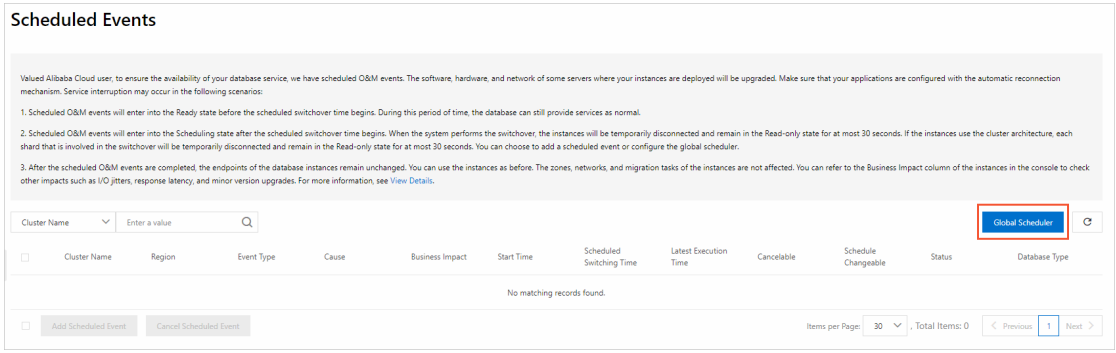
Procedure

1.
2.
3. In the left-side navigation pane, choose **Event Center > Scheduled Events**.

   > **Note** If a scheduled 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.

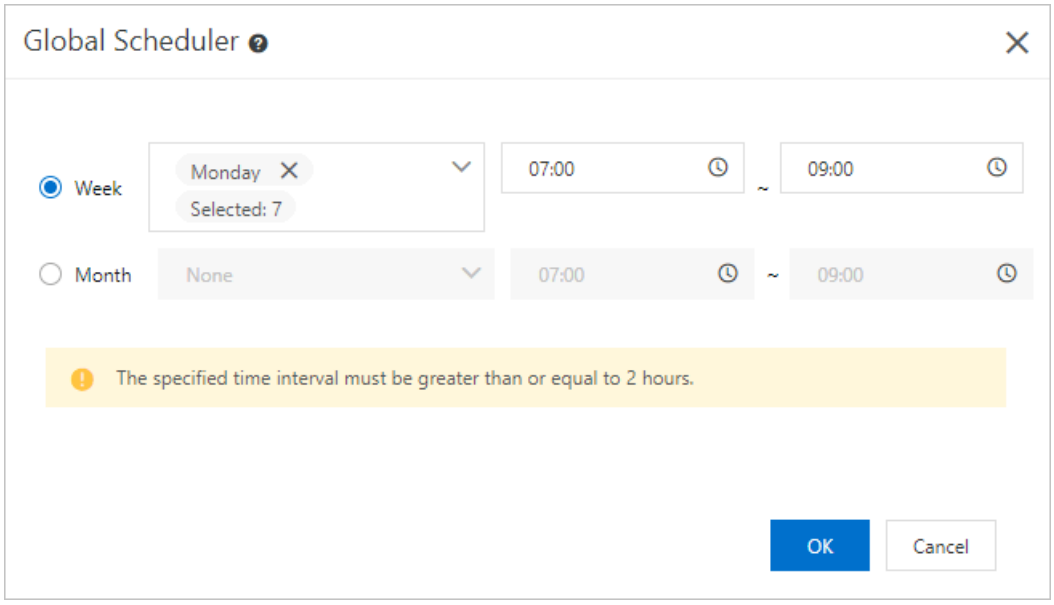
4. (Optional)On the **Scheduled Events** page, you can specify periodic switching time.

i. In the upper-right corner, click **Global Scheduler**.



> **Note** The **Global Scheduler** panel provides the global configuration items of scheduled O&M events (excluding high-risk vulnerability fix events). After the periodic switching time is set, the scheduled switching time of the new scheduled O&M event will automatically use the periodic switching time. If the periodic switching time is not set, the scheduled switching time of the new scheduled O&M event will automatically use the maintenance window of the cluster.

ii. In the Global Scheduler dialog box, configure the periodic switching time and click **OK**.



5. On the **Scheduled Events** page, you can view the details of the event. To change the switching time of the event, select the cluster that you want to manage and click **Add Scheduled Event**.
6. In the **Add Scheduled Time** dialog box, configure **Scheduled Switching Time** and click **OK**.

> **Note**
>
> - If you select **Earliest Execution Time**, the system automatically enters the earliest execution date and time. After you click **OK**, the events in the cluster are pending for processing. If you clear Set the earliest execution time, you can change the scheduled switching date and time.
> - The time that is specified by the **Scheduled Switching Time** parameter cannot be later than the time that is specified by the **Start Deadline** parameter.

Causes and impacts of events

<table>
<tr><th>Upgrade type</th><th>Cause</th><th>Impact</th><th>Description</th></tr>
<tr><td rowspan="8"></td><td>Cluster migration</td><td rowspan="6">Transient connections</td><td rowspan="6">When the switching is performed at the you may experience the following impacts:
Typically, you can use the hot upgrade mode to upgrade the minor version of a cluster. Your clusters or data shards in your clusters experience transient connections and stay in the read-only state for up to 30 seconds before all the data is synchronized. We recommend that you perform the switching during off-peak hours. Make sure that your application can be automatically reconnected to your database system.
You cannot manage your clusters by using Data Management (DMS) or Data Transmission Service (DTS). This impact is temporary.
Scheduled switching time</td></tr>
<tr><td>Switching between primary and read-only nodes</td></tr>
<tr><td>Cluster parameter adjustment</td></tr>
<tr><td>Host vulnerability fixing</td></tr>
<tr><td>SSL certificate update</td></tr>
<tr><td>Backup mode change</td></tr>
<tr><td rowspan="2">Minor version update</td><td>Transient connections</td><td>When the switching is performed at the you may experience the following impacts:
Typically, you can use the hot upgrade mode to upgrade the minor version of a cluster. Your clusters or data shards in your clusters experience transient connections and stay in the read-only state for up to 30 seconds before all the data is synchronized. We recommend that you perform the switching during off-peak hours. Make sure that your application can be automatically reconnected to your database system.
You cannot manage your clusters by using Data Management (DMS) or Data Transmission Service (DTS). This impact is temporary.</td></tr>
<tr><td>Differences between minor versions</td><td>Different minor versions have different updates. You must check the differences between the current minor version and the minor version to which your nodes are updated. For more information, see Engine release notes.</td></tr>
</table>

Upgrade type	Cause	Impact	Description
Hot update	Minor version update for proxy nodes	Transient connections	When the switching is performed at the you may experience the following impacts: • Typically, you can use the hot upgrade mode to upgrade the minor version of a cluster. Your clusters or data shards in your clusters experience transient connections and stay in the read-only state for up to 30 seconds before all the data is synchronized. We recommend that you perform the switching during off-peak hours. Make sure that your application can be automatically reconnected to your database system. • You cannot manage your clusters by using Data Management (DMS) or Data Transmission Service (DTS). This impact is temporary.
		Differences between minor versions	Different minor versions have different updates. You must check the differences between the current minor version and the minor version to which your nodes are updated.
	Network upgrade	Transient connections	When the switching is performed at the you may experience the following impacts: • Typically, you can use the hot upgrade mode to upgrade the minor version of a cluster. Your clusters or data shards in your clusters experience transient connections and stay in the read-only state for up to 30 seconds before all the data is synchronized. We recommend that you perform the switching during off-peak hours. Make sure that your application can be automatically reconnected to your database system. • You cannot manage your clusters by using Data Management (DMS) or Data Transmission Service (DTS). This impact is temporary.
		Change of virtual IP addresses	Some network upgrades may involve cross-zone migrations, which change the virtual IP address of a cluster. If a client uses a virtual IP address to connect to a cloud database, the connection is interrupted. **Note** To prevent transient connections, you must use the endpoint in the form of a domain name that is provided by your cluster and disable the DNS cache feature of the application and its server.
	Storage gateway upgrade	I/O jitter	Temporary I/O jitter may occur, and the SQL latency may increase. These impacts last no longer than three seconds.

Upgrade type	Cause	Impact	Description
Cold upgrade	Major version upgrade Public preview version upgrade	Transient connections	When the switching is performed at the you may experience the following impacts: • Your clusters or data shards in your clusters experience transient connections and stay in the read-only state for up to two minutes. If your instances contain a large number of table files, are executing large transactions before the upgrade, or encounter high CPU utilization before the upgrade, transient connections may exceed two minutes. We recommend that you upgrade the version of a cluster during off-peak hours and ensure that your application supports automatic reconnection. • If a cold upgrade is required, check the differences between versions and select an appropriate upgrade time based on your business situation. Scheduled switching time

Related API operations

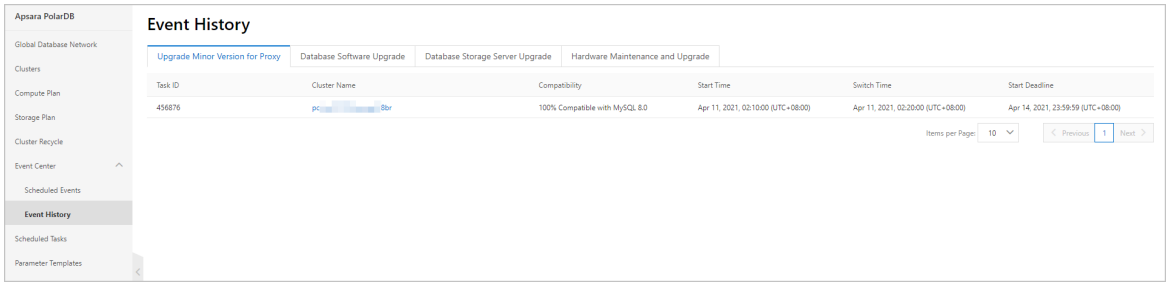
Operation	Description
DescribePendingMaintenanceActions	Queries the numbers of scheduled events for different types of tasks.
ModifyPendingMaintenanceAction	Modifies the task switching time of scheduled events.
DescribePendingMaintenanceAction	Queries the details about scheduled events.

20.2. Query historical events

You can view the scheduled O&M events that are completed in the console, such as database software upgrades, hardware maintenance, and hardware upgrades.

Procedure

1.
2.
3. In the left-side navigation pane, choose **Event Center > Event History** .



4. On the **Event History** page, click the tab of the event type that you want to view. You can view the details about each historical event, such as the cluster name and the time of the event.

21.Cluster Parameters

21.1. Specify cluster and node parameters

After you create a cluster, you can modify the parameters of the cluster and its nodes in the console and export the modified parameters as templates. You can also apply templates to clusters that are deployed in the same region for easy parameter modification. This topic describes how to modify the parameters of a cluster and its nodes, export the modified parameters as templates, and apply the templates.

Context

allows you to configure the parameters of each cluster node based on your business requirements.

- After a new node is added to a cluster, parameters of the cluster are applied to the node by default.
- When the primary node is unavailable, a read-only node takes over.
  - The original primary node becomes a read-only node, and the parameters of the node remain unchanged.
  - The original read-only node becomes the primary node and the parameters of the node remain unchanged.
- When the primary zone fails, a node in the secondary zone is promoted to become the new primary node and parameters of the cluster are applied to the new primary node.

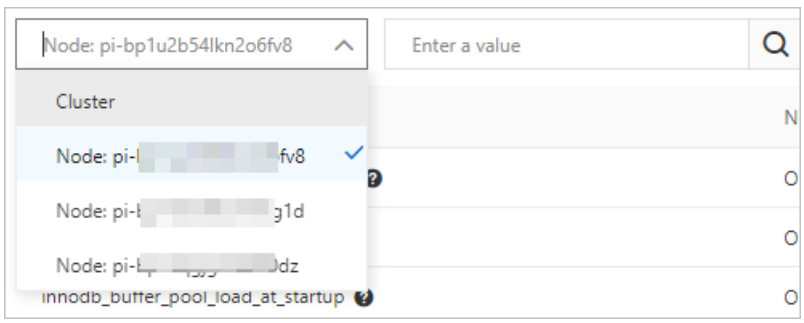
Precautions

You can only modify the `innodb_buffer_pool_size` parameter for nodes in a 8.0 cluster. To modify other node parameters, to contact technical support.

Modify parameters

The following example describes how to modify node parameters.

1.
2.
3.
4. In the left-side navigation pane, choose **Settings and Management > Parameters**.
5. Select the node whose parameters you want to modify from the drop-down list in the upper-left corner.



6. In the upper-left corner of the page, click **Modify**.
7. Find the parameter whose value you want to change, and enter the new value in the **Cluster Parameter** column.
8. In the upper-left corner of the page, click **Apply Changes**. In the **Save Changes** dialog box, click **OK**.
9. In the **Save Changes** panel, select the node to which you want the modification applies and click **OK**.

Apply a parameter template

After you specify the parameters, you can perform the following steps to apply a parameter template:

- Export the parameter settings of a cluster as a parameter template.
  i. Log on to the PolarDB console.
  ii. In the upper-left corner of the console, select the region where the cluster is deployed.
  iii. In the left-side navigation pane, click **Clusters**.
  iv. On the **Clusters** page, click the name of the cluster.
  v. In the left-side navigation pane, choose **Settings and Management > Parameters**.
  vi. Click **Export Template**.

> **Note**
> - If you want to modify parameters, we recommend that you first modify parameters and apply changes, and then click **Export Template**.
> - Parameters in the exported template include the parameters that are manually modified and the parameters that are automatically modified by the running cluster.

  vii. In the **Apply Template** dialog box, set the following parameters.

Parameter	Description
Template Name	The name of the parameter template. The name must meet the following requirements: ■ It can contain letters, digits, and underscores (_). It must start with a letter and cannot contain Chinese characters. ■ It must be 8 to 64 characters in length.
Description	The description of the parameter template. The description can be up to 200 characters in length.

  viii. Click **Next Step**.
- Apply a parameter template to a cluster
  i. Click **Apply Template**.
  ii. In the **Apply Template** dialog box, select the name of the template that you want to apply and click **OK**.

> **Note**
>
> - You can view the number of parameters, whether a restart is required, and the update time.
> - If a restart is required, we recommend that you apply the parameter template during off-peak hours and make sure that your application is configured to automatically reconnect to the cluster.
> - You can also apply a parameter template to the cluster on the Parameter Templates page. For more information, see Apply a parameter template.

Compare the parameter settings of different nodes

You can use the parameter value comparison function to compare the parameter settings of different nodes.

1.
2.
3.
4. In the left-side navigation pane, choose **Settings and Management > Parameters**.
5. In the upper-left corner of the page, click **Compare**.
6. Select the node whose parameter settings that you want to compare. Then, you can view the comparison results.

Related API operations

Operation	Description
DescribeDBClusterParameters	Queries cluster parameters.
ModifyDBClusterParameters	Modifies cluster parameters.
ModifyDBClusterAndNodesParameters	Modifies cluster parameters and applies them to specified nodes.
ModifyDBNodesParameters	Modifies the parameters of a single node and applies them to a specified node.
DescribeParameterTemplates	Queries the default parameters.

21.2. Set parameters to expressions

allows you to set the parameters of cluster specifications to expressions. Parameter values, which are expressions, dynamically change when specifications change. This ensures optimal performance and stability of a cluster.

Usage notes

- Only numeric values can be specified in an expression. String values cannot be specified in an expression.

For example, if the valid values of the innodb_sort_buffer_size parameter are numeric values 65536 to 67108864, you can set this parameter to the following expression: `{DBNodeClassMemory*3/4}`. However, the valid values of the loose_recycle_bin parameter are ON and OFF, which are not numeric values. You cannot set this parameter to an expression.

Some parameters may have numeric values that do not carry numeric meaning, such as the valid values of the ssl parameter. The valid values of this parameter are 0, which specifies that SSL encryption is disabled, and 1, which specifies that SSL encryption is enabled. In this scenario, we recommend that you do not set this parameter to an expression.

- If the specifications of a cluster change, the value of a parameter that is set to an expression may become invalid. If the value becomes invalid, the final value of the parameter is based on the valid values.

  For example, if `{DBNodeClassMemory*1/2}` is used to calculate the final value of the innodb_max_undo_log_size parameter and the final value of the parameter is within the range of valid values `10485760 to 107374182400`, the final value is determined based on the following rules:

  - If the memory of your cluster is upgraded to 256 GB or higher, the final value of `{DBNodeClassMemory*1/2}` is calculated by using the following formula: `256 GB × 1/2 = 128 GB = 137,438,953,472 bytes`. However, you cannot use this result as the final value because this result exceeds the specified range of valid values. Therefore, the final value of the innodb_max_undo_log_size parameter is 107374182400, which is the maximum valid value.
  - If the memory of your cluster is lower than 256 GB, the final value of `{DBNodeClassMemory*1/2}` is within the range of valid values. Therefore, the final value of the innodb_max_undo_log_size parameter is the actual result of the calculation based on `{DBNodeClassMemory*1/2}`.

  > **Note** To ensure that the value of a parameter remains valid, we recommend that you check the valid values of the parameter before you set it to an expression, and consider how the value changes when specifications change.

Supported expressions

The following table describes the expression syntax that is supported in .

Category	Remarks	Format

<table>
<tr><th>Category</th><th>Remarks</th><th>Format</th></tr>
<tr><td>Variables</td><td>

DBNodeClassIOPS: the input/output operations per second (IOPS) of a compute node. The value is an integer.
DBNodeClassMemory: the memory size of a compute node. The value is an integer. Default unit: bytes.
DBNodeClassCPU: the number of CPU cores of a compute node. The value is an integer.
DBNodeClassConnections: the maximum number of connections supported by an instance. The value is an integer.

Note For more information about the compute node specifications, including the IOPS, memory size, number of CPU cores, and maximum number of connections, see Specifications of compute nodes.
</td><td rowspan="2"><code>{DBNodeClassMemory*3/4}</code>: specifies that the value of the parameter must be equal to 75% of the memory size of the current compute node.</td></tr>
<tr><td>Operators</td><td>

Expression syntax: An expression is enclosed in a pair of braces (<code>{}</code>).
Division operator (/): A dividend is divided by a divisor. The quotient is an integer. If the quotient is a decimal, only the whole number is used.
Syntax:
<code>dividend / divisor</code>
Multiplication operator (*): A number multiplies another number. The product is an integer. If the product is a decimal, only the whole number is used.
Syntax:
<code>expression * expression</code>

Note The dividend, divisor, and multiplier must be integers. For example, <code>{DBNodeClassMemory*3/4}</code> is supported, but <code>{DBNodeClassMemory*0.75}</code> is not supported.
</td></tr>
</table>

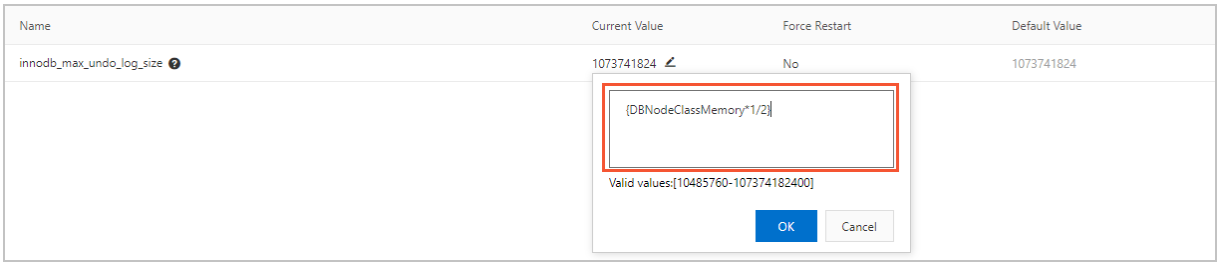
Category	Remarks	Format

Functions	• The `GREATEST()` function returns the largest value from an array of integers or the largest value from an array of values calculated by an array of expressions.
	Syntax:
	`GREATEST(argument1, argument2,...argumentn)`
	• The `LEAST()` function returns the smallest value from an array of integers or the smallest value from an array of values calculated by an array of expressions.
	Syntax:
	`LEAST(argument1, argument2,...argumentn)`
	• The `SUM()` function adds an array of integers or the values calculated by an array of expressions.
	Syntax:
	`SUM(argument1, argument2,...argumentn)`
		`LEAST({DBNodeClassMemory*1/2},10485760)`: specifies that the value of the parameter is the smaller value between 50% of the memory size of the current compute node and 10485760.

How to use expressions

The procedure that is used to set parameters to expressions is the same as the procedure that is used to configure cluster parameters. For more information about the procedure, see Specify cluster and node parameters.

You need to enter only an expression when you modify **Current Value**. For example, you can set **Current Value** to `{DBNodeClassMemory*1/2}` for innodb_sort_buffer_size.



21.3. Apply a parameter template

When you configure multiple clusters that contain one or more parameters with the same settings, you can use parameter templates to manage and quickly apply parameters to clusters. This improves the efficiency of parameter management and cluster configuration.

Prerequisites

The following versions of clusters are supported:

- MySQL 8.0
- MySQL 5.7

- MySQL 5.6

Create a parameter template

1.
2.
3. In the left-side navigation pane, click **Parameter Templates**.
4. On the **Parameter Templates** page, click **Create Parameter Template**.
5. On the page that appears, specify the following parameters.

Parameter	Description
Template Name	The name of the template. The name must meet the following requirements: ○ It can contain letters, digits, and underscores (_). It must start with a letter and cannot contain Chinese characters. ○ It must be 8 to 64 characters in length.
Database Engine	The type of database engine. Only MySQL is supported.
Version	The version of the database engine. Valid values: ○ 8.0 ○ 5.7 ○ 5.6
Description	The description of the parameter template. It must be 0 to 200 characters in length.
Add Parameter	After you click **Add Parameter**, the system adds a parameter in the list. You can specify **Template Name** and set **Current Value** based on **Description** and **Valid Value**. You can also view the information about whether a restart is required and the default parameter value on the page. **Note** ○ To add another parameter, click **Add Parameter** again. ○ To remove a parameter, click **Delete** on the right of the parameter.

Import Parameter	Click **Import Parameter** and enter parameters and values. The parameters and values must be in the `key=value` format. Separate multiple key-value pairs with line feeds. Example:
	`wait_timeout=60`
	`thread_stack=262144`

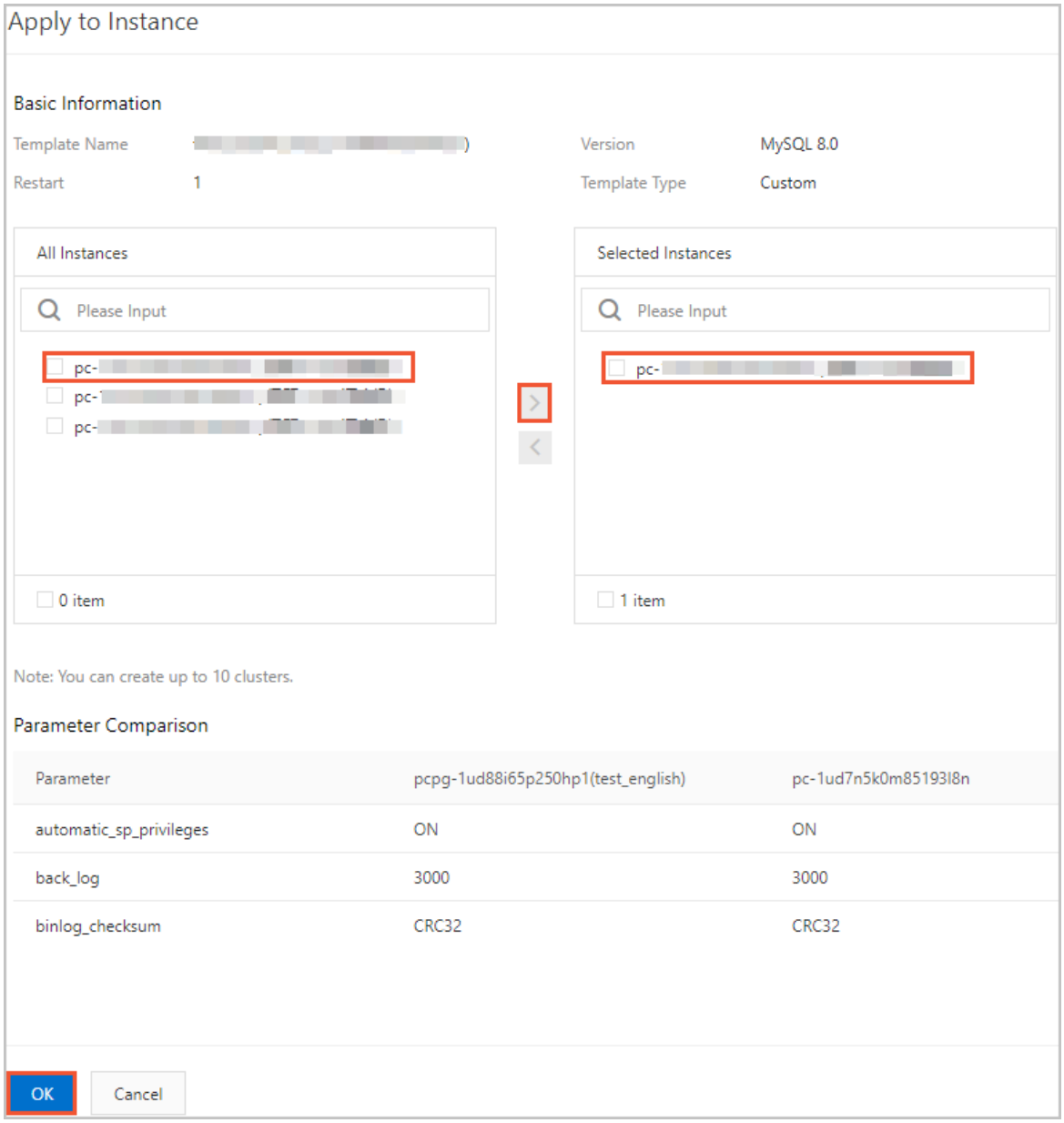
6. Click **OK**.

Apply a parameter template

> **Note**
>
> - If a restart is required, we recommend that you apply the parameter template during off-peak hours and make sure that your application is configured to automatically reconnect to the cluster.
> - After a parameter template is created, you can apply the parameter template to the cluster by using the following methods:
>   - You can apply a parameter template on the related page of a cluster. For more information, see Specify cluster parameters.
>   - You can also perform the following steps to apply a parameter template to the cluster.

1.
2.
3. In the left-side navigation pane, click **Parameter Templates**.
4. On the **Parameter Templates** page, click the **Custom Parameter Templates** tab.
5. In the parameter template list, click **Apply to Instance** in the **Actions** column of the parameter template.
6. On the **Apply to Instance** page, select the clusters to which you want to apply the parameter template and click > to move them to the right. You can view the difference between the parameter values in the template and the parameter values in the cluster in the **Parameter Comparison** section.

   > **Note** When you apply a parameter template to multiple clusters, take note of the following information:
   >
   > - Check whether the parameters are applicable to these clusters. Proceed with caution.
   > - A parameter template can be applied to up to 10 clusters at a time. To apply a parameter template to more than 10 clusters, you can apply it in batches.
   > - The parameter template must be in the same region as the cluster to which the template is applied. If no parameter template is available in the region where the cluster is deployed, create one first.



7. Click **OK**.

Copy a parameter template

1.
2.
3. In the left-side navigation pane, click **Parameter Templates**.
4. On the **Parameter Templates** page, click the **Custom Parameter Templates** tab.
5. In the parameter template list, click **Clone** in the **Actions** column of the parameter template.
6. On the page that appears, specify the following parameters.

Parameter	Description

Parameter	Description
Template Name	The name of the template. By default, the name is *the name of the original template*_clone. You can rename the template. The name must meet the following requirements: ◦ It can contain letters, digits, and underscores (_). It must start with a letter and cannot contain Chinese characters. ◦ It must be 8 to 64 characters in length.
Database Engine	The type of database engine. Only MySQL is supported.
Version	MySQL 5.6, 5.7, and 8.0 are supported. The default version is the version of the original template. You can specify the version based on your business requirements.
Description	The description of the parameter template. It must be 0 to 200 characters in length.
Add Parameter	By default, the values of parameters are the same as the parameters in the original template. You can modify the values based on your business requirements. **Note** ◦ To add a parameter, click **Add Parameter**. ◦ To modify the value of a parameter, enter the value in the **Current Value** field on the right of the parameter. ◦ To remove a parameter, click **Delete** on the right of the parameter.

Import Parameter	Click **Import Parameter** and enter parameters and values. The parameters and values must be in the `key=value` format. Separate multiple key-value pairs with line feeds. Example:
	`wait_timeout=60`
	`thread_stack=262144`

7. Click **OK**.

Delete a parameter template

1.
2.
3. In the left-side navigation pane, click **Parameter Templates**.
4. On the **Parameter Templates** page, click the **Custom Parameter Templates** tab.
5. In the parameter template list, click **Delete** in the **Actions** column of the parameter template.
6. Click **OK**.

> **Note** To delete multiple parameter templates at a time, select the parameter templates and click **Delete** at the bottom of the list.

View differences in parameter templates

1.
2.
3. In the left-side navigation pane, click **Parameter Templates**.
4. On the **Parameter Templates** page, click the **Custom Parameter Templates** tab.
5. In the custom parameter template list, select the parameter templates and click **Compare**.
6. On the **Compare** page, view the differences of parameters in different parameter templates.

Related API operations

API	Description
CreateParameterGroup	Creates a parameter template.
DescribeParameterGroups	Queries parameter templates.
DescribeParameterGroup	Queries the details of a parameter template.
ModifyDBClusterParameters	Modifies or applies a parameter template.
DeleteParameterGroup	Deletes a parameter template.

21.4. High-performance parameter template of PolarDB for MySQL 8.0

provides the high-performance parameter template feature. This topic describes the parameter settings for the high-performance parameter template of 8.0 and how to enable the high-performance parameter template feature. This topic also provides a performance comparison before and after enabling this feature.

High-performance parameter template

PolarDB has a large number of parameters, which may complicate parameter configuration when optimizing your database for specific scenarios. To help simplify database optimization, PolarDB provides the high-performance parameter template feature. You can use a template out of the box by directly applying the template to an existing cluster, or make deeper optimizations on top of the parameter template. In most cases, high-performance parameter templates help improve the performance of databases.

A high-performance parameter template of 8.0 includes the following parameters:

Parameters in the high-performance parameter template of PolarDB for MySQL 8.0

Parameter	Description	Value in the high-performance parameter template	Default value
innodb_flush_log_at_trx_commit	Specifies the disk write policy of the database. If you set this parameter to 0, data is written to and refreshed in the database once per second.	0	1
loose_innodb_lock_sys_rec_partition	The number of shards for the transaction lock. InnoDB uses shards to manage transaction locks, including row locks and table locks. This parameter reduces the competition overhead introduced by transaction lock management.	64	1
loose_query_cache_type	Specifies whether to enable the fast query cache feature. For more information, see Fast query cache. This feature improves the query performance of the database.	ON	OFF

Limits

Your cluster must be a 8.0.1 cluster whose revision version is 8.0.1.1.21 or later.

Scenarios and risks

In most cases, high-performance parameter templates help improve the performance of databases. However, after a high-performance template is applied, its pursuit for performance may pose risks to databases:

- **Database stability** may be affected. For example, log durability may be reduced.
- **Transaction durability** may be impaired. For example, when the database crashes, transaction modifications that are not flushed to the disk are lost.

Therefore, if your business can tolerate the trade-off between database performance and durability, you can apply the high-performance parameter template to improve database performance.

Notice Before you apply a high-performance parameter template, we recommend that you submit a ticket first to the Alibaba Cloud technical team for consultation.

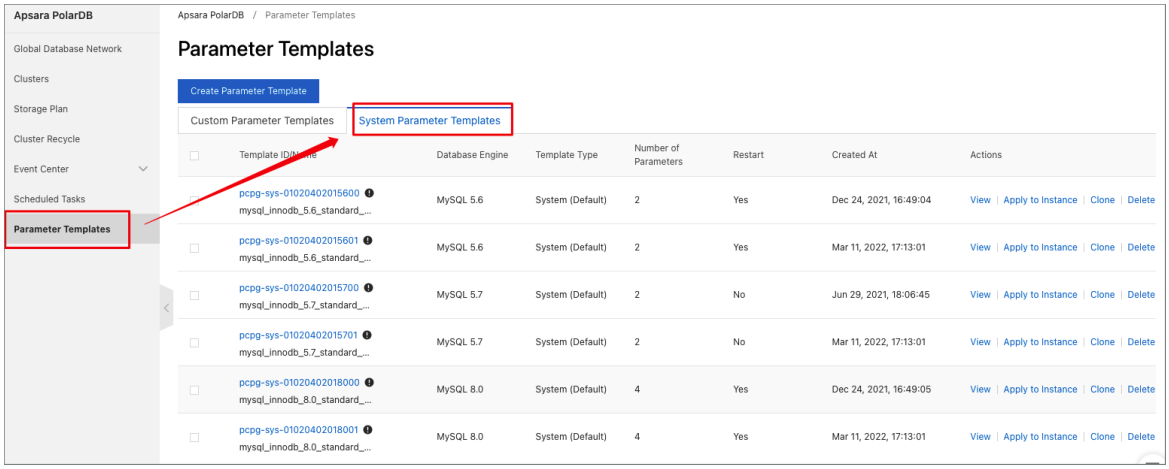
Apply a high-performance parameter template

You can use one of the following methods to apply a high-performance parameter template to your cluster.

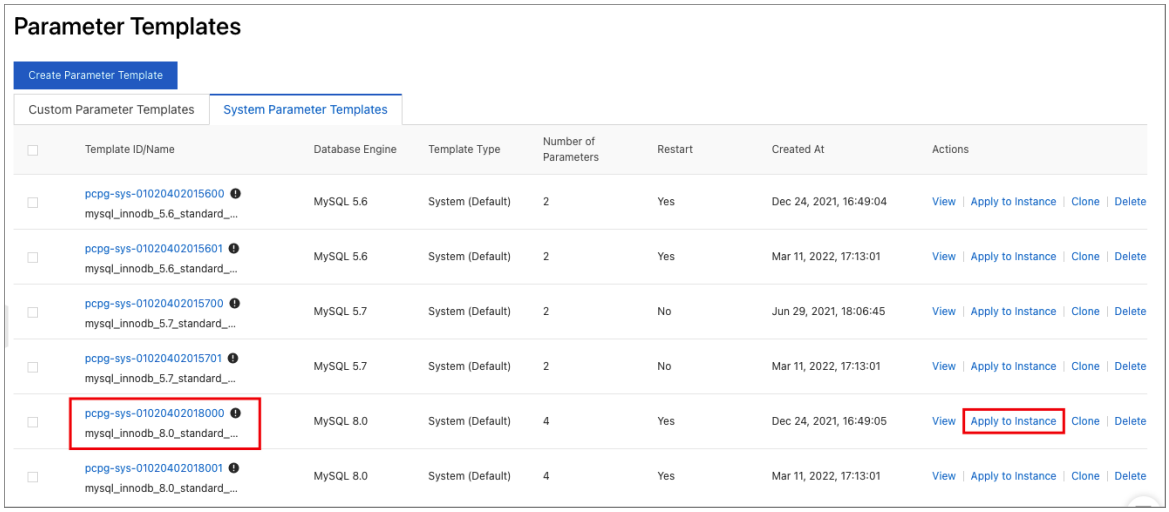
Warning After you apply the template, the database must be restarted for the template to take effect. During the restart, you may experience some intermittent service interruptions. We recommend that you modify parameters during off-peak hours and make sure that your application is configured to automatically reconnect to your cluster.

Method 1:

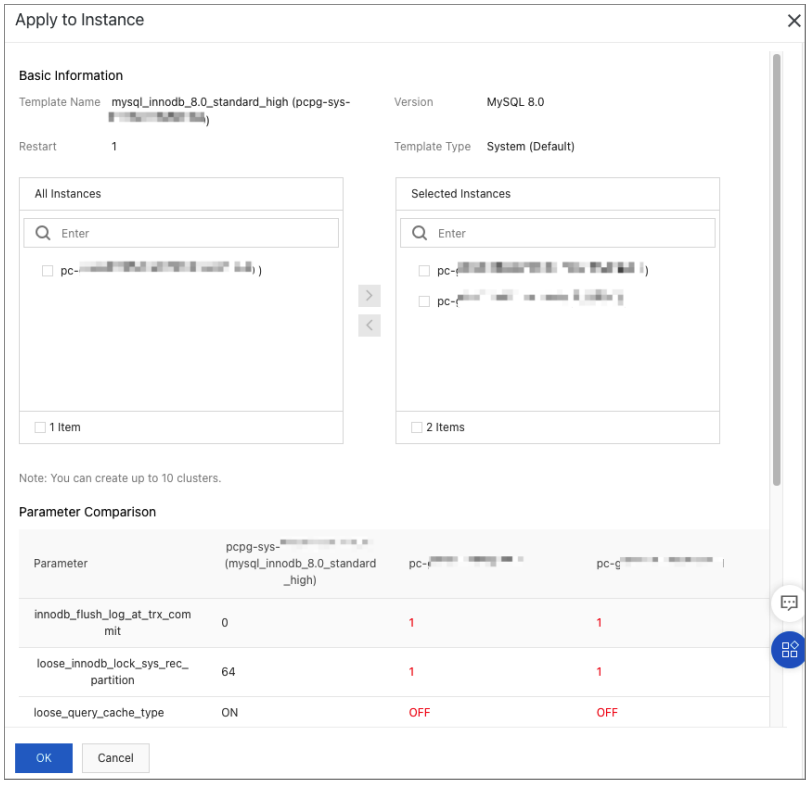
1. Log on to the PolarDB console.
2. In the upper-left corner of the console, select the region in which the cluster that you want to manage is deployed.
3. In the left-side navigation pane, click **Parameter Templates**.
4. On the **Parameter Templates** page, click **System Parameter Templates**.



5. Find the **mysql_innodb_8.0_standard_high** template and click **Apply to Instance** in the **Actions** column.



6. In the **Apply to Instance** panel, select the cluster to which you want to apply the template and click **>** to add the cluster to the **Selected Instances** list.



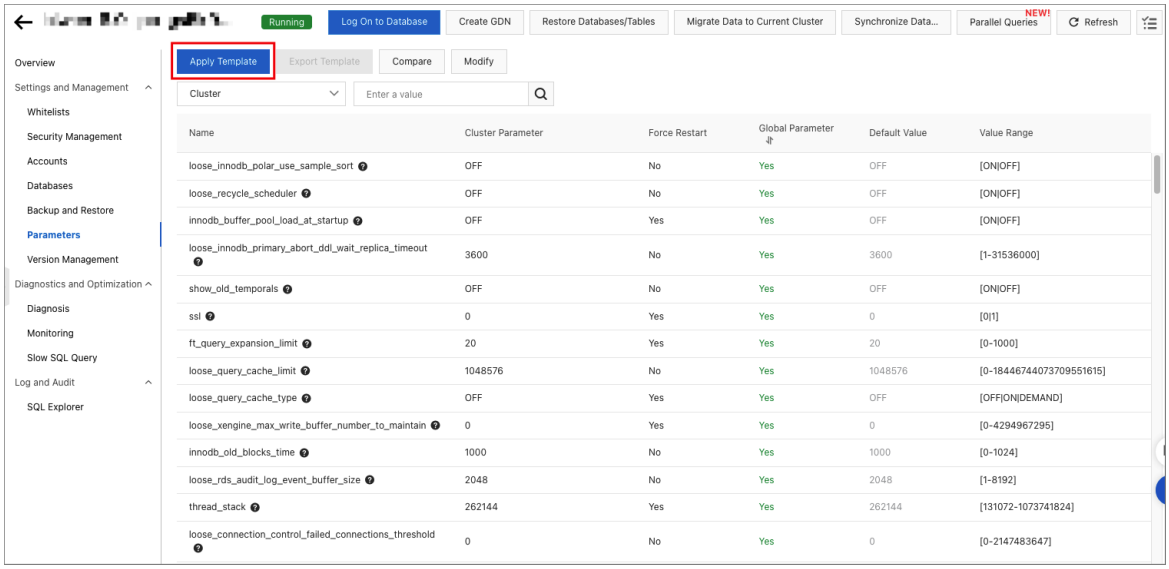
After you select a cluster, you can view the differences between the current values of the cluster parameters and those in the high-performance parameter template in the **Parameter Comparison** section.

7. Click **OK**.

> **Note**
>
> After you apply the template, the database must be restarted for the template to take effect.

Method 2:

1. Log on to the PolarDB console.
2. In the upper-left corner of the console, select the region in which the cluster that you want to manage is deployed.
3. Find the cluster and click the cluster ID.
4. In the left-side navigation pane, choose **Settings and Management > Parameters**.
5. Click **Apply Template**.



6. In the **Application Template** panel, select **mysql_innodb_8.0_standard_high** from the **Template Name** drop-down list.

After you select a template, you can view the differences between the current values of the cluster parameters and those in the high-performance parameter template in the **Parameter Comparison** section.

7. Click **OK**.

> **Note**
>
> After you apply the template, the database must be restarted for the template to take effect.

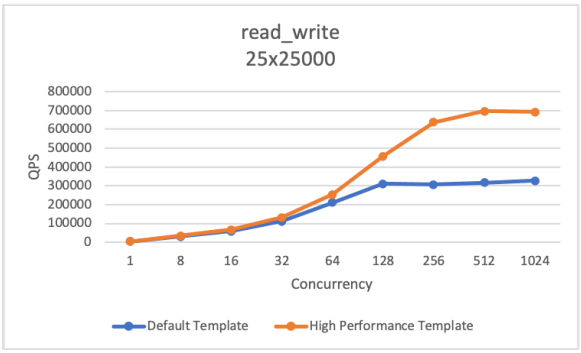
Performance comparison

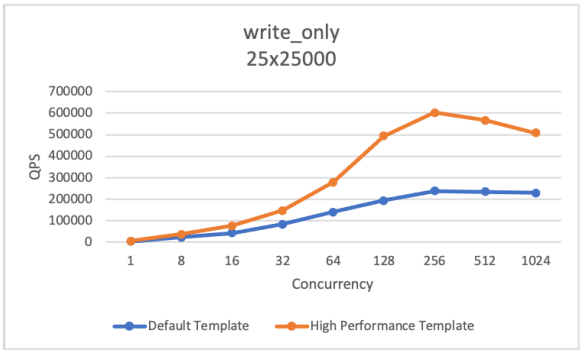
This section compares the performance between the default parameters and the high-performance parameters on a 8.0 cluster. The sysbench and TPC-C benchmark suites are used to test the cluster performance before and after the high-performance parameter template is used.

> **Note**
>
> - sysbench is a modular, cross-platform, and multi-threaded benchmark tool. sysbench is useful for evaluating the performance of load-intensive databases.
> - TPC-C is a benchmark that is widely used to evaluate the online transaction processing (OLTP) capabilities of databases. It is developed and released by Transaction Processing Performance Council (TPC).
> - The TPC-C performance tests described in this topic are implemented based on the TPC-C benchmark test but do not meet all requirements of the TPC-C benchmark test. Therefore, the test results described in this topic cannot be compared with the published results of the TPC-C benchmark test.

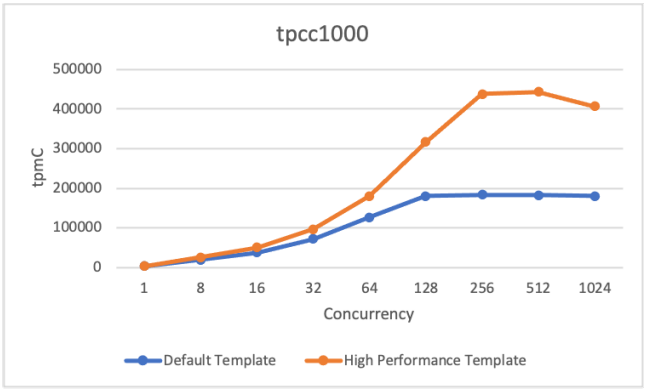
- **Test environment:**
  - cluster:
    - Cluster specification: 88 cores and 720-GB memory.
    - Revision version: 8.0.1.1.21 or later.
  - Stress test environment:
    - The latency between the ECS instance used for the stress test and the PolarDB cluster is about 1 ms.
    - The ECS instance used for the stress test has sufficient computing and network resources.
- **sysbench test:**
  - Test sets: read_write and write_only.
  - Raw data volume: 25 tables × 25,000 rows of data.
  - Performance metric - queries per second (QPS): the number of SQL statements that are executed per second in the database when 1, 8, 16, 32, 64, 128, 256, 512, and 1,024 concurrent requests are made. The SQL statements include INSERT, SELECT, UPDATE, and DELETE statements.
- **TPC-C test:**
  - Test tool: TPCC-MySQL.
  - Raw data volume: 1,000 warehouses.
  - Performance metric - transactions per minute (TPM): the number of transactions that are committed per minute in the database when 1, 8, 16, 32, 64, 128, 256, 512, and 1,024 concurrent requests are made. This test can measure the maximum qualified throughput (MQTh) of the database.
- **Test results:**

- sysbench test





- TPC-C test



- **Test conclusions:** The preceding sysbench and TPC-C test results show that the high-performance parameter template improves the performance of the cluster under intensive workloads. For 256 or more concurrent requests, the high-performance parameter template improves the performance of the cluster by twice or more.

22.More Operations

22.1. Clone a cluster

This topic describes how to create a new cluster by cloning the data of a source cluster.

Scenarios

Before you launch a service, the service is deployed in an environment that simulates real-world scenarios for testing, such as stress testing. To achieve this, you can create a new cluster by cloning the data of a source cluster. Then, you can conduct tests on the new cluster. This ensures the accuracy of the tests without affecting normal business operation.

Considerations

- The following data of the source cluster can be cloned:
  - Cluster account information.
  - The transparent data encryption (TDE) configurations can be cloned if the source cluster has TDE enabled.
- The following data of the source cluster cannot be cloned:
  - Parameter settings
  - Whitelist configurations
  - Secure sockets layer (SSL) configurations
- Only the data that exists in the source cluster before the clone operation starts is cloned.

Procedure

1.
2.
3. Find the cluster that you want to clone and choose **More > Clone Cluster** in the **Actions** column.
4. On the **Clone Instance** page, select a **billing method** for the new cluster.
5. Configure the following parameters.

Parameter	Description
Clone Source Type	By default, **Current Cluster** is selected. For this operation, do not change this setting.
Clone Source Cluster	The ID of the source cluster to clone. This setting cannot be changed.
Region	By default, the region of the new cluster is the same as that of the original cluster. This setting cannot be changed.

Parameter	Description
Primary Availability Zone	Select the primary zone where the cluster is deployed. **Note** In regions that have two or more zones, automatically replicates data to a secondary zone for disaster recovery.
Network Type	This parameter can only be set to **VPC**.
VPC	Select a VPC and a vSwitch for the cluster. We recommend that you use the same **VPC** and **VSwitch** that are used for the original cluster. **Note** Make sure that the cluster and the ECS instance you want to connect to the cluster are deployed in the same VPC. Otherwise, the cluster and the ECS instance cannot communicate over the internal network, which results in decreased performance.
VSwitch	
Compatibility	By default, the new cluster has the same compatibility as that of the source cluster. For example, if the **compatibility** of the source cluster is **MySQL 8.0**, , and , the **compatibility** of the new cluster is **MySQL 8.0**, , and . You do not need to change this parameter value.
Edition	By default, the edition of the new cluster is the same as that of the source cluster. For example, if the **edition** of the source cluster is , the **edition** of the new cluster is also . You do not need to change this parameter value.
Specification Type	has the following two types of specifications: **General Specification** and **Dedicated Specification**. For more information about the two types of specifications, see Comparison between general-purpose and dedicated compute nodes. **Note** This parameter is available only when the **edition** of the source cluster is . The and **editions** do not support this parameter.
Node Specification	Select a **node specification**. The maximum storage capacity and performance of clusters vary based on node specifications. For more information, see Specifications of compute nodes. **Note** We recommend that you select a **node specification** that is the same or higher than the node specification of the original cluster. This ensures that the new cluster runs as expected.

Parameter	Description
Nodes	○ The default number of nodes of the **edition** is 2. You do not need to change this parameter value. **Note** By default, new clusters contain one primary node and one read-only node. After a cluster is created, you can add nodes to the cluster. A cluster can contain one primary node and up to 15 read-only nodes. For more information about how to add nodes, see Add or remove read-only nodes. ○ The default number of nodes of the and **editions** is **1**. You do not need to change this parameter value.
Storage Cost	You do not need to select the storage capacity when you purchase clusters. You are charged for the storage capacity used on an hourly basis. You can also purchase a storage plan based on your business requirements. For more information about how to purchase a storage plan, see Purchase a storage plan.
Cluster Name	The name of the cluster. The name must meet the following requirements: ○ It cannot start with `http://` or `https://` . ○ It must be 2 to 256 characters in length. If you do not specify this parameter, the system automatically generates a cluster name. You can change the name after the cluster is created.
Purchase Plan	Specify the **purchase plan** for the cluster. **Note** This parameter is available only when the **Billing Method** parameter is set to **Subscription**.
Number	Select the **number** of clusters you want to purchase.

6. Read and accept the terms of service, and complete the rest of the steps based on the **billing method** of the cluster.
   - **Pay-as-you-go**

     Click **Buy Now**.
   - **Subscription**
     a. Click **Buy Now**.
     b. On the **Purchase** page, confirm the information of the unpaid order and the payment method and click **Purchase**.

Note After you complete the payment, it requires 10 to 15 minutes to create the cluster. Then, you can view the new cluster on the **Clusters** page.

22.2. Enable binary logging

This topic describes how to enable the binary logging feature for a cluster.

Context

is a cloud-native database that is compatible with MySQL. By default, PolarDB uses redo logs, which are more advanced than binary logs. However, to better integrate with the MySQL ecosystem, allows you to enable the binary logging feature. After you enable the binary logging feature, you can access data services, such as Elasticsearch and AnalyticDB. You can also replicate data from a PolarDB for MySQL cluster to an ApsaraDB for RDS instance, from an ApsaraDB for RDS instance to a PolarDB for MySQL cluster, or between PolarDB for MySQL clusters.

Limits

- If your cluster was created after April 5, 2019, you can directly enable the binary logging feature. If your cluster was created before April 5, 2019, upgrade the minor version of your cluster to the latest version before you enable the binary logging feature. For more information about how to upgrade the minor version, see Version Management.
- The binary logging feature cannot be enabled for secondary clusters in a global database network (GDN).

Billing

The storage that is used to store binary logs is a part of the cluster storage. You are charged for the actual storage resources that you use. For more information, see Storage pricing.

Precautions

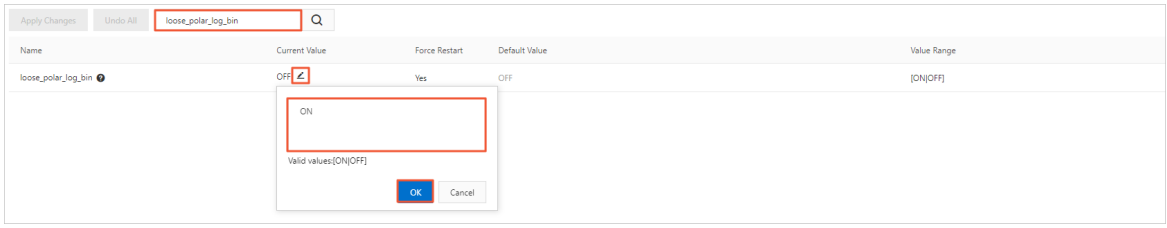
- By default, the binary logging feature is disabled. After you enable the binary logging feature, the cluster automatically restarts. A cluster restart task is typically completed in 5 minutes. During the restart, services may be interrupted for up to 40 seconds. The length of recovery time depends on the data volume and the number of tables. We recommend that you perform this operation during off-peak hours and make sure that your application is configured to automatically reconnect to the database service.
- By default, binary logs are retained for two weeks and then automatically deleted. For more information, see FAQ.
- When the binary logging feature is enabled, the write performance of your cluster is deteriorated while the read performance is not affected. Typically, your cluster may lose less than 10% of the performance after you enable the binary logging feature. For more information, see FAQ.
- We recommend that you use the **Primary Endpoint** of when you pull, subscribe to, or replicate binary logs by using services such as Data Transmission Service (DTS). This ensures compatibility and stability because the primary endpoint points to the primary node that generates binary logs. For more information about how to query the **Primary Endpoint**, see View an endpoint.
- The loose_polar_log_bin parameter described in this topic is a global parameter. To use the session-level binary logging feature, you can specify the sql_log_bin parameter to enable the feature.

  > **Note** In most cases, the sql_log_bin parameter is used to temporarily disable the session-level binary logging feature. By default, this parameter is specified to disable this feature. To enable this feature, submit a ticket to contact technical support.

Procedure

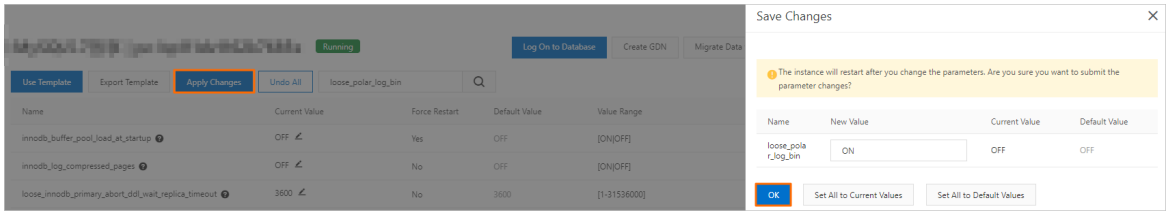
1.

2.
3.
4. In the left-side navigation pane, choose **Settings and Management > Parameters**.
5. Find the loose_polar_log_bin parameter and click the icon in the **Current Value** column. In the dialog box that appears, enter a new value and click **OK**.



> **Note**
>
> - For a cluster that runs MySQL 5.6, enter *ON_WITH_GTID*.
> - For a cluster that runs MySQL 5.7 or MySQL 8.0, enter *ON*.

6. In the upper-left corner of the page, click **Apply Changes**. In the **Save Changes** dialog box, click **OK**.



> **Note**
>
> - After you enable the binary logging feature, the cluster automatically restarts. The binary logging feature takes effect after the restart is complete. A restart task is typically completed in 5 minutes. During the restart, services may be interrupted for up to 40 seconds. The length of recovery time depends on the data volume and the number of tables. We recommend that you perform this operation during off-peak hours and make sure that your application is configured to automatically reconnect to the database service.
> - For 8.0 and 5.6 clusters, you can set the innodb_fast_startup parameter to *ON* to accelerate the restart process.
> - If the "**Custins minor version does not support current action**" message is returned, submit a ticket to enable the binary logging feature.

FAQ

- How long can binary logs be stored?

  Binary logs are stored based on the following policies:

- By default, binary logs are retained for two weeks and then automatically deleted.
  - For 5.6 clusters, you can change the value of the loose_expire_logs_hours parameter to specify the retention period of binary logs. The valid values of this parameter range from 0 to 2376. Unit of valid values: hours. The value 0 indicates that binary logs are not automatically deleted.
  - For 5.7 or 8.0 clusters, you can change the value of the binlog_expire_logs_seconds parameter to specify the retention period of binary logs. The valid values of this parameter range from 0 to 4294967295. Unit of valid values: seconds. The value 0 indicates that binary logs are not automatically deleted.
  - After you set the retention period of binary logs by modifying the loose_expire_logs_hours or binlog_expire_logs_seconds parameter, historical binary log files in the cluster are not automatically deleted. historical binary log files in the cluster will be cleared after you can restart the cluster or after a new binary log file is used when the last binary log file in the cluster reaches the max_binlog_size value.
- After the binary logging feature is disabled, existing binary logs are permanently retained.

  > **Note** To delete the binary logs, you can enable the feature and set loose_expire_logs_hours or binlog_expire_logs_seconds to a smaller value. After the retention period is reached, binary logs are automatically deleted. Then, you can disable the binary logging feature.

- Can I disable the binary logging feature?

  Yes, the binary logging feature is disabled after you change the value of the loose_polar_log_bin parameter to *OFF* and submit the modification.

  > **Note** After the binary logging feature is disabled, existing binary logs are permanently retained. To delete the binary logs, you can enable the feature and specify a short retention period. After the retention period is reached, binary logs are automatically deleted. Then, you can disable the binary logging feature.

- How can I reduce the storage space occupied by binary logs?

  You can set loose_expire_logs_hours or binlog_expire_logs_seconds to a smaller value to reduce the storage space occupied by binary logs.

- How is the performance of a cluster affected after the binary logging feature is enabled?

  After you enable the binary logging feature, only the write performance is affected, such as the performance of INSERT, UPDATE, and DELETE statements. However, the performance of SELECT statements is unaffected. Typically, your cluster may lose less than 10% of the performance after you enable the binary logging feature.

- After binary logging is enabled, the cluster automatically restarts. How long does it take to restart the cluster?

  In most cases, it takes 5 minutes to restart the cluster. During the restart, services may be interrupted for up to 40 seconds. The length of recovery time depends on the data volume and the number of tables. We recommend that you perform this operation during off-peak hours and make sure that your application is configured to automatically reconnect to the database service.

- How do I remotely retrieve and view binary logs?

For more information, see Remotely obtain and parse binary log records from a cluster of the PolarDB for MySQL.

- Why am I unable to use the DDL-based lockless change feature of Data Management Service (DMS) to manage tables of a cluster, for example, to add an index to a table?

  By default, the binary logging feature of the cluster is disabled. To use DMS to change schemas without locking tables, you must first enable the binary logging feature for the cluster. If you do not want to enable the binary logging feature, you can execute data definition language (DDL) statements to change table schemas.

22.3. Set a maintenance window

This topic describes how to set a maintenance window for a cluster so that your business is not affected during the maintenance process.

Context

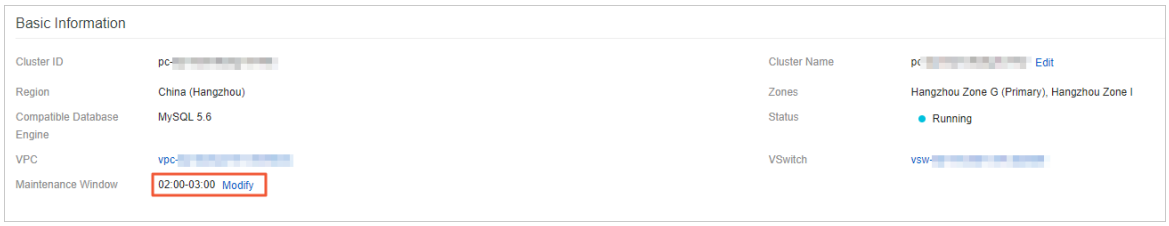
To ensure the stability of clusters, the backend system performs maintenance operations on the clusters from time to time. We recommend that you select a maintenance window within the off-peak hours of your business to minimize the impact on the business during the maintenance process.

Considerations

- Before the maintenance is performed on a cluster, sends SMS messages and emails to contacts listed in your Alibaba Cloud account.
- To ensure the stability of a cluster during the maintenance process, the cluster enters the Under Maintenance state before the specified maintenance window starts. When the cluster is in the Under Maintenance state, you can access data in the databases of the cluster. However, features that are related to configuration changes become unavailable in the console except for the account management, database management, and whitelisting features. For example, you cannot upgrade, downgrade, or restart the cluster. Query features such as performance monitoring are still available.
- Within the maintenance window of a cluster, the cluster may experience one or two transient disconnections. Make sure that the application has an automatic reconnection mechanism. The cluster recovers to the normal state immediately after the disconnection.

Procedure

1.
2.
3.
4. On the **Overview** page, click **Modify** next to **Maintenance Window**.

5. In the Modify Maintenance Window dialog box, select a maintenance window, and click **OK**.

> **Note**
>
> - To ensure the stability of clusters, the backend system performs maintenance operations on the clusters from time to time. We recommend that you select a maintenance window within the off-peak hours of your business to minimize the impact on the business during the maintenance process.
> - Within the maintenance window of a cluster, the cluster may experience one or two transient disconnections. Make sure that the application has an automatic reconnection mechanism.

Related API operations

API operation	Description
CreateDBCluster	Creates a cluster.
ModifyDBClusterMaintainTime	Modifies the maintenance window for a cluster.

22.4. Restart nodes

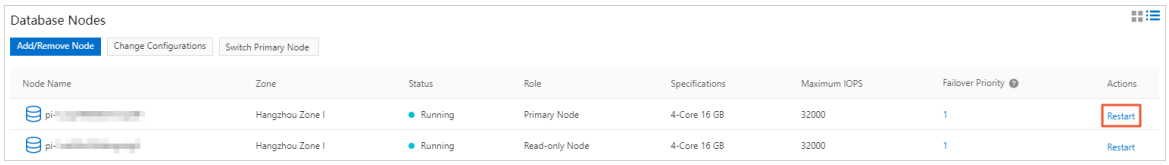
allows you to restart nodes. When the number of database connections reaches the upper limit or the database performance is compromised, you can manually restart nodes.

Usage notes

- A read/write splitting connection that is established after a read-only node is restarted forwards requests to the read-only node. If a read/write splitting connection is established before a read-only node is restarted, the connection does not forward requests to the read-only node. You can restart your application to close the read/write splitting connection and establish the connection again.
- During the restart, services may be interrupted for up to 1 minute. We recommend that you perform this operation during off-peak hours and make sure that your application is configured to automatically reconnect to the database service.
- The time required to restart a node depends on the data volume. Several hours may be required to restart a node. Proceed with caution.

Procedure

1.
2.
3.
4. In the upper-right corner of the **Database Nodes** section for the **Overview** page, click the icon to switch the display mode.
5. Find the node that you want to restart, and click **Restart** in the **Actions** column.



> **Note** If your cluster version is 8.0, you can set the innodb_fast_startup parameter to *ON* to accelerate the restart process. For more information about how to change parameter values, see Specify cluster and node parameters.

6. In the dialog box that appears, click **OK**.

Related API operations

API	Description
RestartDBNode	Restarts a node of a cluster.

22.5. View or cancel a scheduled task

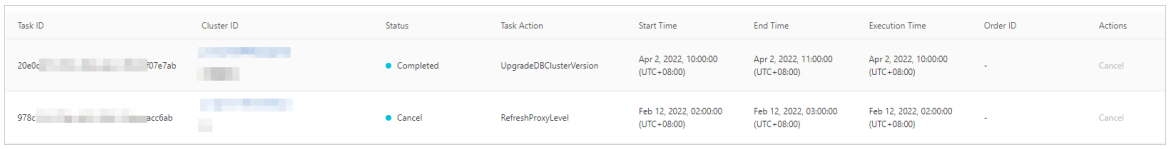
When you perform operations and management (O&M) tasks, you can customize the execution time of the tasks. For example, you can customize the execution time of tasks for upgrading a cluster, adding nodes, upgrading versions, or changing the primary zone. This topic describes how to view or cancel a scheduled task in the console after you create the task.

Precautions

- You can view the details of only the following scheduled tasks:
  - Upgrade a cluster. For more information about the procedure, see Procedure.
  - Add nodes. For more information about the procedure, see Add a read-only node.
  - Upgrade the version of a cluster. For more information, see Upgrade versions.
  - Change the primary zone. For more information about the procedure, see Change the primary zone and vSwitch of a cluster.
- You can cancel only the tasks whose **Status** is **Pending**. Scheduled tasks for downgrade operations such as node deletion and automatic or manual downgrade cannot be canceled.

View scheduled tasks

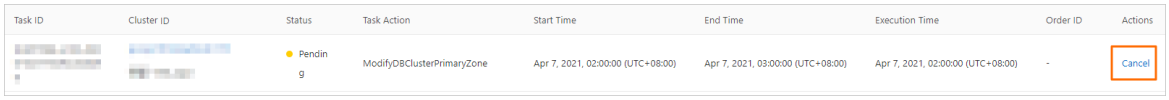
1.
2.
3. In the left-side navigation pane, click **Scheduled Tasks**.
4. On the **Scheduled Tasks** page, you can view the details about all scheduled tasks in the region, such as the **Task ID**, **Status**, **Task Action**, **Start Time**, **End Time**, and **Execution Time**.



> **Note**
>
> - The API of the task is displayed in the **Task Action** column. The following **Task Action** are supported:
>   - **ModifyDBClusterPrimaryZone**: changes the primary zone.
>   - **ModifyDBNodeClass**: upgrades a cluster.
>   - **CreateDBNodes**: adds nodes.
>   - **UpgradeDBClusterVersion**: upgrades the version of a cluster.
> - You can view the **Order ID** of the cluster only when **Task Action** is **ModifyDBNodeClass** or **CreateDBNodes**.

Cancel a scheduled task

1.
2.
3. In the left-side navigation pane, click **Scheduled Tasks**.
4. On the **Scheduled Tasks** page, find the scheduled task that you want to cancel, and click **Cancel** in the **Actions** column.



> **Note** You can cancel only the tasks whose **Status** is **Pending**. Scheduled tasks for downgrade operations such as node deletion and automatic or manual downgrade cannot be canceled.

5. In the dialog box that appears, click **OK**.

Related API operations

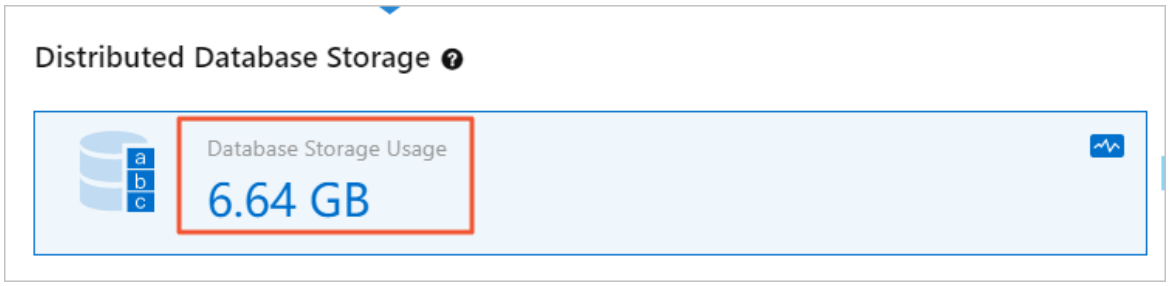
Operation	Description
DescribeScheduleTasks	Queries the details of all scheduled tasks or a specified scheduled task that belongs to the current account.
CancelScheduleTasks	Cancels a specified scheduled task.

22.6. View the database storage usage

You can view the database storage usage of a cluster in the console. This topic describes how to view the database storage usage.

Procedure

1.
2.
3.
4. On the **Overview** page, check the value of the **Database Storage Usage** in the **Distributed Database Storage** section.



> **Note** The maximum storage capacity varies based on cluster specifications. If 90% of the maximum storage capacity is used, the system sends SMS messages and emails to notify you on a daily basis. To increase the maximum storage capacity, upgrade your cluster specifications. For more information, see Manually upgrade or downgrade a PolarDB cluster.

22.7. Release a cluster

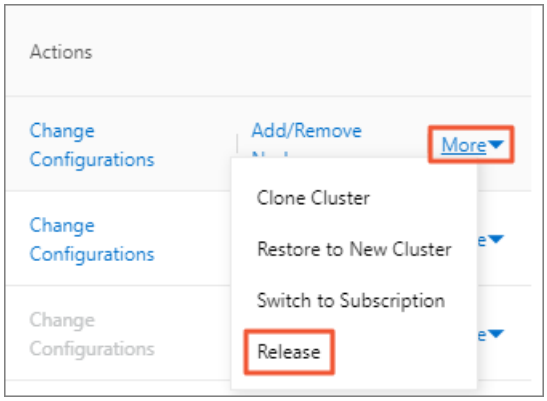
You can manually release pay-as-you-go clusters based on your business requirements. The pay-as-you-go clusters are charged on an hourly basis. This topic describes how to manually release clusters.

Considerations

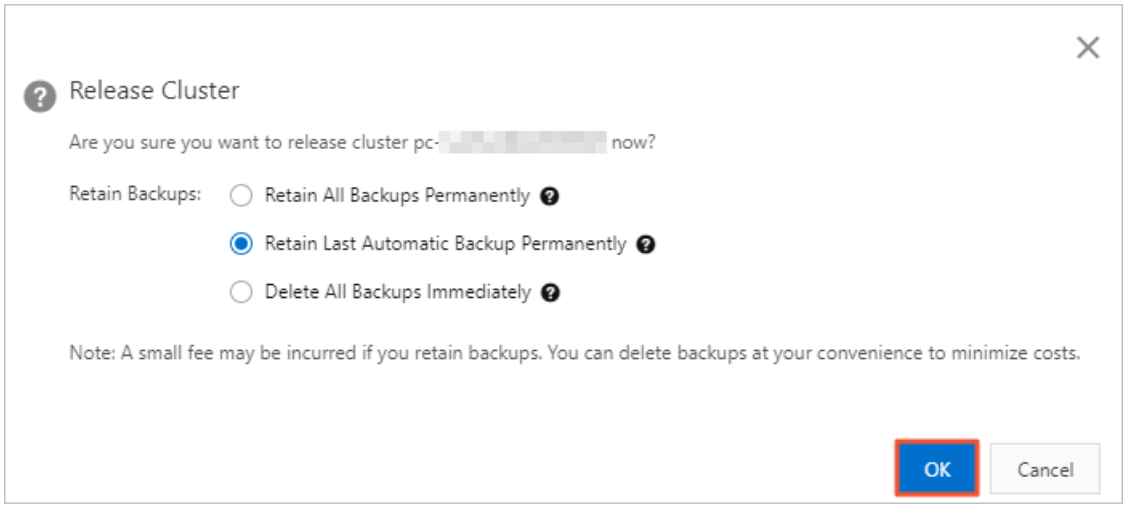
- You cannot manually release **subscription** clusters. Subscription clusters are automatically released when they expire.
- You can manually release the cluster whose **Status** is only **Running**.
- You can use this feature to release clusters. If this feature is used, all the nodes of the clusters are released. For more information about how to release a single read-only node, see Add or remove read-only nodes.

Procedure

1.
2.
3. On the **Clusters** page, find the cluster that you want to release and choose **More > Release** in the **Actions** column.



4. In the **Release Cluster** dialog box, select a backup retention policy and click **OK**.



Retain backups	Description
Permanently Retain All Backups	Retains all the backups for a cluster when you delete the cluster.
Permanently Retain Last Automatic Backup	Retains the last backup for a cluster when you delete the cluster.
Immediately Delete All Backups	Deletes all the backups for a cluster when you delete the cluster. **Warning** If you select this policy, the deleted clusters cannot be restored.

Note

- If you select the **Permanently Retain All Backups** or **Permanently Retain Last Automatic Backup** policy, the system runs an automatic backup task to retain all the data about a cluster when you delete the cluster.
- After you delete a cluster, level-1 backups are automatically transferred to level-2 backups. You can go to the **Cluster Recycle** page to view retained backups. For more information, see Restore a released cluster.

Related API operations

API	Description
DescribeDBClusters	Queries PolarDB clusters.
DeleteDBCluster	Deletes a specified PolarDB cluster.

22.8. Cluster lock feature

You can enable the cluster lock feature for your pay-as-you-go clusters to prevent potential irreversible consequences arising from accidental manual release of the clusters. This topic describes how to enable or disable the cluster lock feature.

Prerequisites

The billing method of the cluster is pay-as-you-go.

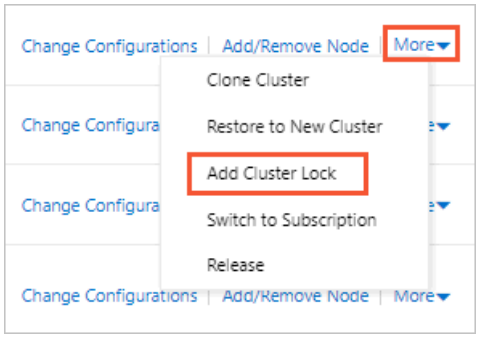
Precautions

- The billing method of clusters with the cluster lock feature enabled cannot be changed to subscription.
- The cluster lock feature cannot prevent the automatic release of clusters in normal cases such as the following ones:
  - A payment in your account is overdue for more than eight days.
  - The cluster does not comply with the applicable security compliance policies.

Enable the cluster lock feature

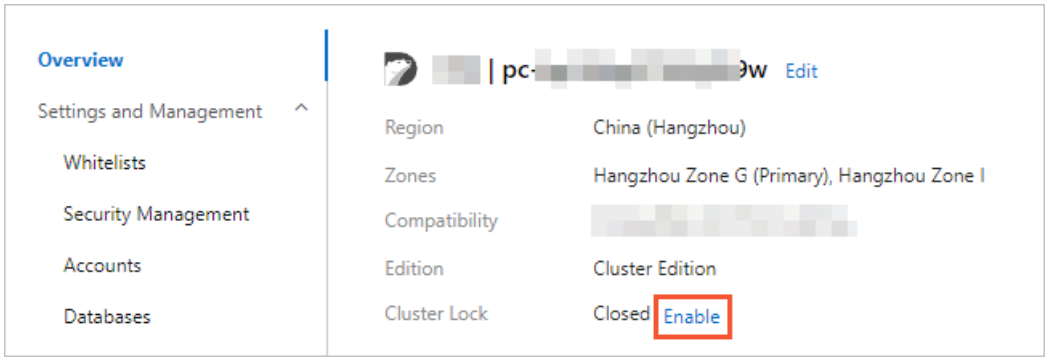
1.
2.
3. You can use one of the following methods to enable the cluster lock feature:
   - Method 1:

     On the **Clusters** page, find the cluster and choose **More > Add Cluster Lock** in the **Actions** column.

   - Method 2:
     a. On the **Clusters** page, click the cluster.

b. On the **Overview** page, click **Enable** next to **Cluster Lock**.

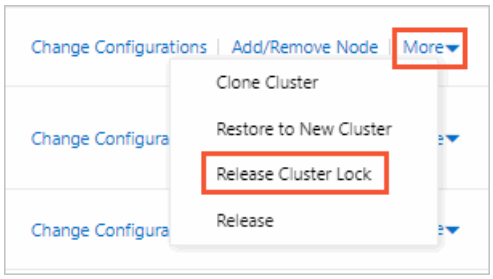


4. In the message that appears, click **OK**.

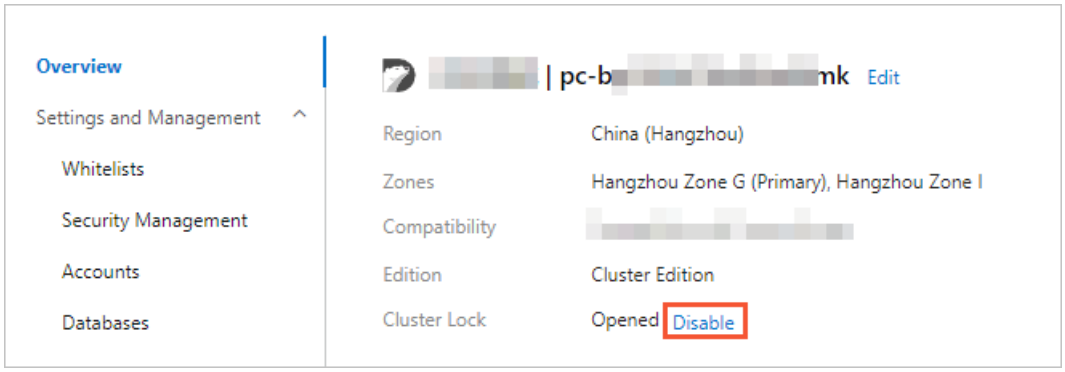
Disable the cluster lock feature

1.
2.
3. You can use one of the following methods to disable the cluster lock feature:
   - Method 1:

     On the **Clusters** page, find the cluster and choose **More > Release Cluster Lock** in the **Actions** column.



   - Method 2:
     a. On the **Clusters** page, click the cluster.
     b. On the **Overview** page, click **Disable** next to **Cluster Lock**.

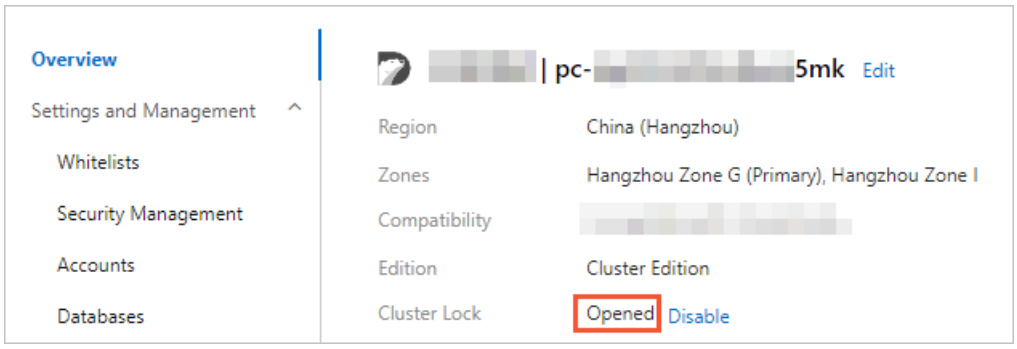


4. In the message that appears, click **OK**.

View the status of the cluster lock feature

1.

2.
3. On the **Clusters** page, click the cluster.
4. On the **Overview** page, view the status of **Cluster Lock**.



Related API operations

Operation	Description
	Enables or disables the cluster lock feature.

22.9. Tags

22.9.1. Bind a tag

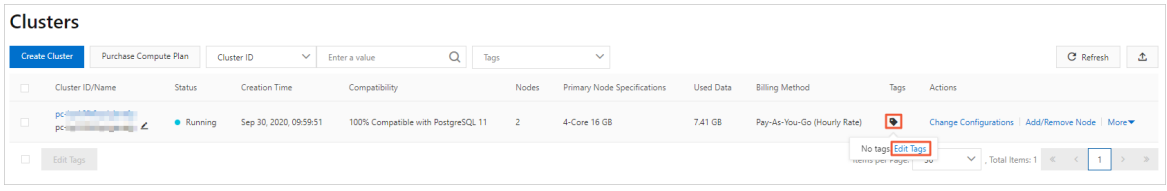
This topic describes how to bind tags to PolarDB clusters. To easily manage a large number of PolarDB clusters, you can create and bind tags to the clusters. You can also filter the clusters by tag.

Notes

- A tag consists of a key-value pair. Each key must be unique for an Alibaba Cloud account in a region. This limit does not apply to the values of keys.
- You can bind a maximum of 20 tags to a cluster. If you create a tag that has the same key as an existing tag, the existing tag is overwritten.
- The tag namespace for the clusters that are deployed in each region is unique.

Procedure

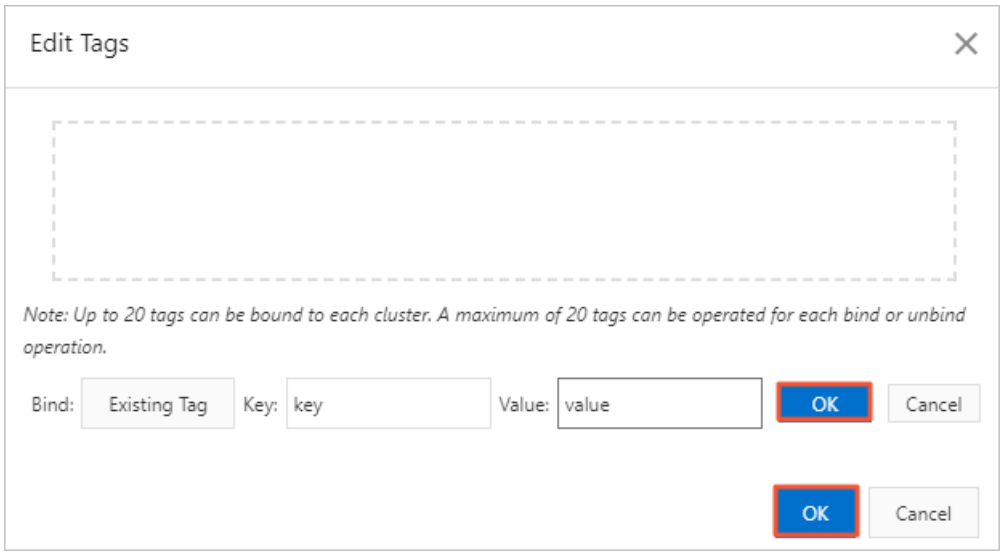
1.
2.
3. On the **Clusters** page, move the pointer over the icon in the **Tags** column of the target cluster.
4. Click **Edit Tags**.



5. In the **Edit Tags** dialog box, click **New Tag** or **Existing Tag**.

- **New Tag**:

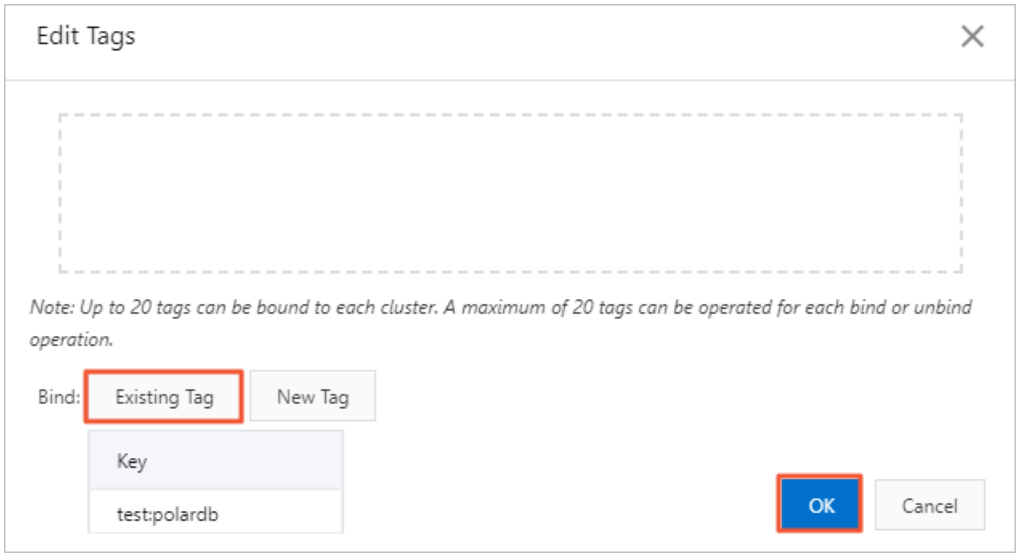
  Specify **Key** and **Value** for the tag and click **OK**.



> **Note** After the tag is created, you can bind it to other clusters.

- **Existing Tag**:

  Click the key of the target tag **Key**.



6. Repeat the preceding steps to create and bind other tags to clusters. In the lower-right corner of the dialog box, click **OK**.

Related API operations

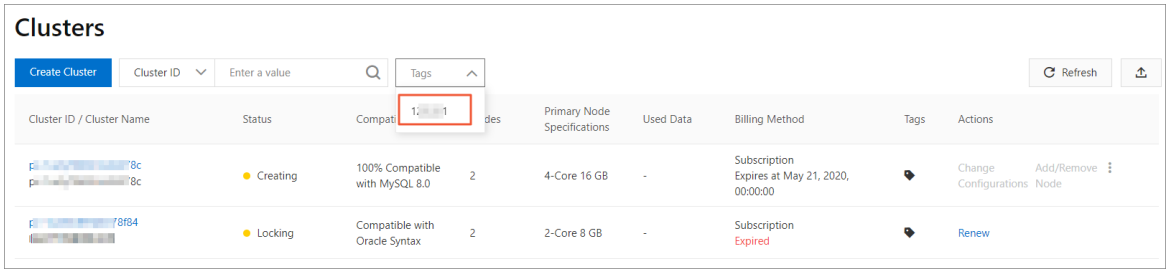
Operation	Description
TagResources	Binds tags to Apsara clusters.

22.9.2. Filter clusters by tag

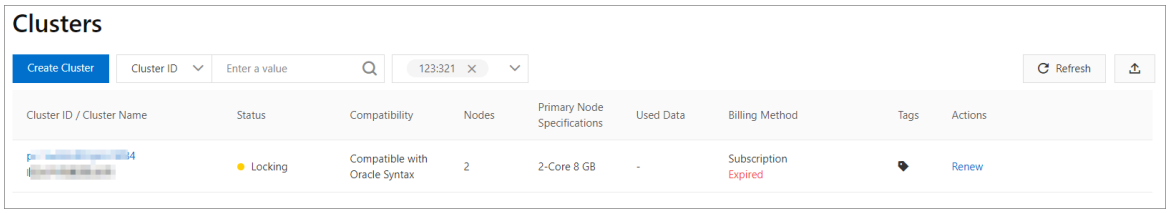
This topic describes how to filter clusters by tag. After you bind tags to clusters, you can filter clusters by tag on the Clusters page. This allows you to find the clusters that are bound to a specified tag.

Procedure

1.
2.
3. On the **Clusters** page, click **Tags** and select the target **Tags**.



4. View the clusters that are bound to the target tag. After you select the target tag, all the clusters that are bound to this tag are displayed on the **Clusters** page.



Related API operations

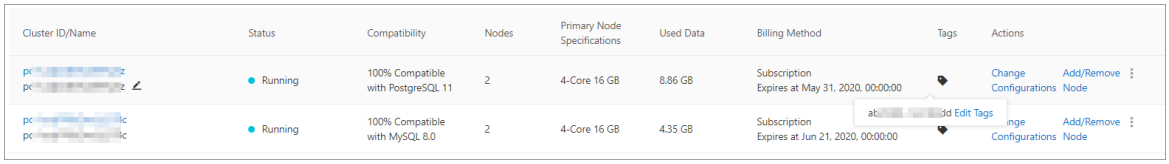
Operation	Description
ListTagResources	Queries the tags that are bound to one or more clusters, or the clusters that are bound to one or more tags.

22.9.3. View tags bound to a cluster

This topic describes how to view the tags that are bound to an cluster. You can view the tags on the Clusters page of the console.

Procedure

1.
2.
3. On the **Clusters** page, move the pointer over the icon in the **Tags** column of the target cluster.
4. View the tags that are bound to the target cluster.



Related API operations

Operation	Description
ListTagResources	Queries the tags that are bound to one or more clusters, or the clusters that are bound to one or more tags.

22.9.4. Unbind a tag

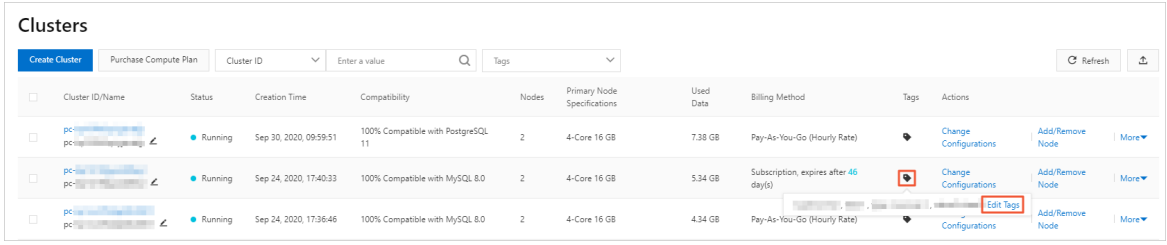
This topic describes how to unbind a tag from an cluster. You can unbind a tag from an cluster based on your business needs.

Notes

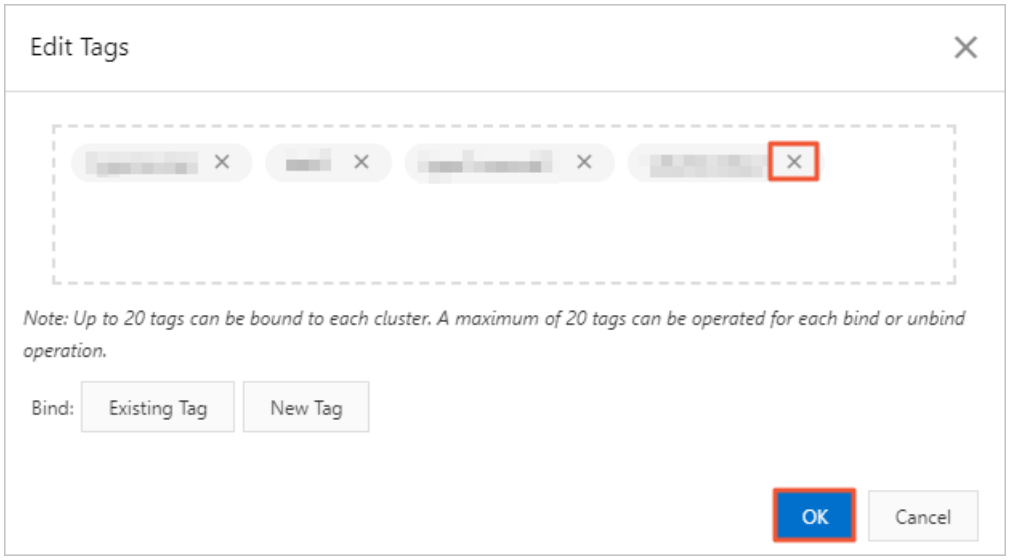
If a tag is unbound from an cluster and the tag is not bound to other clusters, the tag is automatically deleted.

Procedure

1.
2.
3. On the **Clusters** page, move the pointer over the icon in the **Tags** column of the target cluster, click **Edit Tags**.



4. In the **Edit Tags** dialog box, click the icon next to the target tag.



5. Click **OK**.

> **Note** Unbinding a tag from an cluster does not affect other clusters that are bound to this tag.

Related API operations

Operation	Description
UntagResources	Unbinds tags from clusters.