Alibaba Cloud
ApsaraDB for PolarDB

User Guide for POLARDB Oracle Compatibility

Issue: 20200522

Legal disclaimer

Alibaba Cloud reminds you to carefully read and fully understand the terms and conditions of this legal disclaimer before you read or use this document. If you have read or used this document, it shall be deemed as your total acceptance of this legal disclaimer.

1. You shall download and obtain this document from the Alibaba Cloud website or other Alibaba Cloud-authorized channels, and use this document for your own legal business activities only. The content of this document is considered confidential information of Alibaba Cloud. You shall strictly abide by the confidentiality obligations. No part of this document shall be disclosed or provided to any third party for use without the prior written consent of Alibaba Cloud.
2. No part of this document shall be excerpted, translated, reproduced, transmitted, or disseminated by any organization, company, or individual in any form or by any means without the prior written consent of Alibaba Cloud.
3. The content of this document may be changed due to product version upgrades, adjustments, or other reasons. Alibaba Cloud reserves the right to modify the content of this document without notice and the updated versions of this document will be occasionally released through Alibaba Cloud-authorized channels. You shall pay attention to the version changes of this document as they occur and download and obtain the most up-to-date version of this document from Alibaba Cloud-authorized channels.
4. This document serves only as a reference guide for your use of Alibaba Cloud products and services. Alibaba Cloud provides the document in the context that Alibaba Cloud products and services are provided on an "as is", "with all faults" and "as available" basis. Alibaba Cloud makes every effort to provide relevant operational guidance based on existing technologies. However, Alibaba Cloud hereby makes a clear statement that it in no way guarantees the accuracy, integrity, applicability, and reliability of the content of this document, either explicitly or implicitly. Alibaba Cloud shall not bear any liability for any errors or financial losses incurred by any organizations, companies, or individuals arising from their download, use, or trust in this document. Alibaba Cloud shall not, under any circumstances, bear responsibility for any indirect, consequential, exemplary, incidental, special, or punitive damages, including lost profits arising from the use or trust in this document, even if Alibaba Cloud has been notified of the possibility of such a loss.

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

Document conventions

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

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

Contents

1 Overview

PolarDB is a next-generation cloud-based service developed by Alibaba Cloud for relational databases, which is compatible with MySQL, PostgreSQL, and Oracle. Based on a distributed storage architecture, PolarDB provides high-capacity, low-latency online transaction processing (OLTP) services, and cost-effective scalable services.

Basic concepts

- Cluster

  A PolarDB cluster contains one primary instance and up to 15 read-only instances (at least one read-only instance must be provided to guarantee active-active high availability support). A PolarDB cluster ID starts with pc, which stands for PolarDB cluster.

- Instance

  An instance is an independent database server in which you can create and manage multiple databases. An instance ID starts with pi, which stands for PolarDB instance.

- Database

  A database is a logical unit created in an instance. The name of each PolarDB database under the same instance must be unique.

- Region and zone

  Each region is a separate geographic area. Zones are distinct locations within a region that operate on independent power grids and networks. For more information, see Alibaba Cloud's Global Infrastructure.

Console

Alibaba Cloud offers a web-based and easy-to-use console where you can manage various products and services including PolarDB. In the console, you can create, access, and configure your PolarDB database.

For more information about the console layout, see Alibaba Cloud console.

PolarRDB console.

2 Quick start for POLARDB cluster compatible with Oracle

This topic describes how to create a POLARDB cluster compatible with Oracle, specify basic configurations, and connect to the cluster. It allows you to familiarize yourself with the entire process of purchasing and using a POLARDB cluster compatible with Oracle.

Procedure

To purchase and use a POLARDB cluster compatible with Oracle, follow these steps:

1. Create a POLARDB cluster.
2. Configure whitelists.
3. Create accounts.
4. Check endpoints.
5. Connect to the POLARDB cluster.

3 Comparison with native Oracle databases

This topic describes the differences between PolarDB databases compatible with Oracle and native Oracle databases.

Databases

- By default, each Oracle instance that uses a database platform earlier than Oracle Database 12c contains only one database. Oracle Database 12c and later versions support container databases (CDBs). Each CDB can include multiple pluggable databases (PDBs).
- An Apsara PolarDB cluster is equivalent to an Oracle instance. You can create multiple databases for each database cluster.

Users

- Similarities:

  PolarDB databases compatible with Oracle and native Oracle databases share the concept of users. Users are owners of database objects and have access to databases.
- Differences:

  A native Oracle database user can log on to a database only after being granted the CREATE SESSION privilege. By default, a PolarDB database user is granted the LOGIN privilege to log on to a database.

To create a user, you can execute the CREATE USER statement. CREATE USER has the following syntax:

- Oracle syntax:

```
CREATE USER user
  IDENTIFIED { BY password
             | EXTERNALLY [ AS 'certificate_DN' ]
             | GLOBALLY [ AS '[ directory_DN ]' ]
             }
  [ DEFAULT TABLESPACE tablespace
  | TEMPORARY TABLESPACE
      { tablespace | tablespace_group_name }
  | QUOTA size_clause
          | UNLIMITED
          }
          ON tablespace
    [ QUOTA size_clause
            | UNLIMITED
            }
            ON tablespace
    ]...
```

```
  | PROFILE profile
  | PASSWORD EXPIRE
  | ACCOUNT { LOCK | UNLOCK }
    [ DEFAULT TABLESPACE tablespace
    | TEMPORARY TABLESPACE
        { tablespace | tablespace_group_name }
    | QUOTA size_clause
          | UNLIMITED
          }
          ON tablespace
      [ QUOTA size_clause
            | UNLIMITED
            }
            ON tablespace
      ]...
    | PROFILE profile
    | PASSWORD EXPIRE
    | ACCOUNT { LOCK | UNLOCK }
    ]...
 ] ;
```

- PolarDB syntax:

```
CREATE USER|ROLE name [[WITH] option [...]] [IDENTIFIED BY password]
where option can be the following compatible clauses:
 PROFILE profile_name
| ACCOUNT {LOCK|UNLOCK}
| PASSWORD EXPIRE [AT 'timestamp']
or option can be the following non-compatible clauses:
| LOCK TIME 'timestamp'
```

PolarDB databases compatible with Oracle support the syntax of CREATE USER for PostgreSQL. For more information, visit https://www.postgresql.org/docs/11/static/sql-commands.html.

Roles

- For native Oracle databases, a role is a group of privileges and cannot be regarded as an owner of database objects. This role cannot be granted privileges of other roles and does not have access to databases.
- For PolarDB databases compatible with Oracle, a role is equivalent to a user. This role can be regarded as an owner of database objects, can be granted privileges of other roles or users, and can have access to databases. By default, a role does not have the LOGIN privilege. A role must be granted the LOGIN privilege to obtain access to databases. A user has the LOGIN privilege.

To create a role, you can execute the CREATE ROLE statement. CREATE ROLE has the following syntax:

- Oracle syntax:

```
CREATE ROLE role
  [ NOT IDENTIFIED
```

```
| IDENTIFIED { BY password
           | USING [ schema. ] package
           | EXTERNALLY
           | GLOBALLY
           }
] ;
```

- PolarDB syntax:

  The syntax of CREATE ROLE is the same as that of CREATE USER.

Schemas

- Similarities:

  A schema is a logical concept that represents a collection of database objects, such as tables, indexes, and views. These objects are also called schema objects.

- Differences:

Oracle	PolarDB
You cannot create a schema.	You can execute the CREATE SCHEMA statement to create a schema.
When you create a database user, the system automatically creates a schema with the same name as the database username.	Each database has a default schema named PUBLIC. You can use `SET SEARCH_PATH TO 'xxx ';` to modify the current default schema.

Privileges

Similar to native Oracle databases, Apsara PolarDB provides system privileges and object privileges.

- System privileges
  - Oracle

    System privileges allow you to perform specific actions, such as CREATE USER, CREATE TABLE, and CREATE TABLESPACE.

    System privileges include some administrator privileges:

    ■ SYSDBA and SYSOPER: authorized to manage almost all database objects. You are authorized to perform some standard database operations, such as starting and

shutting down databases, creating server parameter files (SPFiles), and changing database archived log.

- SYSBACKUP: backs up and restores databases.
- SYSDG: implements the Data Guard solution.
- SYSKM: manages transparent data encryption (TDE) wallets.
- SYSRAC: manages Oracle Real Application Clusters (RACs).

- PolarDB

- Object privileges

  Object privileges specify actions on objects. Database objects include tables, views , sequences, large objects, schemas, functions, and procedural language. Object privileges include SELECT, INSERT, UPDATE, DELETE, ALTER, INDEX, REFERENCES, and EXECUTE. The object privilege varies, depending on the object type.

  - Oracle

    Users are granted privileges on all objects for a schema.

  - PolarDB

    Only the object owner and superuser are authorized to modify or delete objects.

    Note:
    A superuser is a user who has the POLAR_SUPERUSER privilege.

Monitoring and O&M

- Oracle

  For more information, visit https://docs.oracle.com/cd/E11882_01/server.112/e10897/montune.htm#ADMQS1014

- PolarDB

  Metric monitoring and log monitoring are supported.

  - Metric monitoring: includes performance monitoring, alerts, and performance insights. For more information, see Performance monitoring and alert configuration and #unique_8.
  - Log monitoring: includes slow query logs and SQL Explorer. For more information, see #unique_9.

4 Migrate and synchronize the data

4.1 Migrate data from a user-created Oracle database to a PolarDB cluster compatible with Oracle

Apsara PolarDB is a next-generation relational database service developed by Alibaba Cloud. It is compatible with MySQL, PostgreSQL, and Oracle database engines. The superior performance of Apsara PolarDB in storage and computing meets diverse requirements of enterprises. This topic describes how to migrate data from a user-created Oracle database to a PolarDB cluster compatible with Oracle by using Data Transmission Service (DTS).

Prerequisites

- The version of the user-created Oracle database is 9i, 10g, 11g, 12c, 18c, or 19c.
- The ARCHIVELOG mode is enabled for the user-created Oracle database. Archived log files are accessible and a suitable retention period is set for archived log files. For more information, see Managing Archived Redo Log Files.
- Supplemental logging, including SUPPLEMENTAL_LOG_DATA_PK and SUPPLEMENTAL_LOG_DATA_UI, is enabled for the user-created Oracle database. For more information, see Supplemental Logging.
- The tables to be migrated from the user-created Oracle database contain primary keys or UNIQUE NOT NULL indexes.

Precautions

- DTS uses read and write resources of the source and destination databases during full data migration. This may increase the database load. 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 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 usage of the source and destination databases is less than 30%.
- DTS automatically resumes a failed data migration task. Before you switch your workloads to the destination database, stop or release the data migration task.

Otherwise, the data in the source database will overwrite the data in the destination database after the task is resumed.

Billing

Migration type	Instance configurations	Internet traffic
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 #unique_12.
Incremental data migration	Charged. For more information, see #unique_12.	

Migration types

Migration type	Description
Schema migration	DTS migrates the schemas of the required objects to the destination database. DTS supports schema migration for the following types of objects: table, view, synonym, trigger, stored procedure, function, package, and user-defined type.  **Note:** However, if an object contains triggers, the data between the source and destination databases will become inconsistent.
Full data migration	DTS migrates historical data of the required objects from the user-created Oracle database to the destination PolarDB cluster.  **Note:** During schema migration and full data migration, do not perform DDL operations on the objects to be migrated. Otherwise, the objects may fail to be migrated.
Incremental data migration	DTS retrieves redo log files from the user-created Oracle database. Then, DTS synchronizes incremental data from the user-created Oracle database to the destination PolarDB cluster. Incremental data migration allows you to ensure service continuity when you migrate data from the user-created Oracle database to the destination PolarDB cluster.

SQL operations that can be synchronized during incremental data migration

- INSERT, UPDATE, and DELETE operations

- CREATE TABLE operations

  Note:
  The CREATE TABLE operations to create partition tables or tables that contain functions cannot be synchronized.

- ALTER TABLE, DROP TABLE, RENAME TABLE, CREATE INDEX, and ADD INDEX operations

Preparations

Log on to the source Oracle database, create an account for data collection, and grant permissions to the account.

Note:
If you have created a database account and the account has the permissions that are listed in the following table, skip this step.

Database	Schema migration	Full data migration	Incremental data migration
User-created Oracle database	The owner permission on schemas	The owner permission on schemas	The database administrator (DBA) permission
PolarDB cluster	The owner permission on schemas	The owner permission on schemas	The owner permission on schemas

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

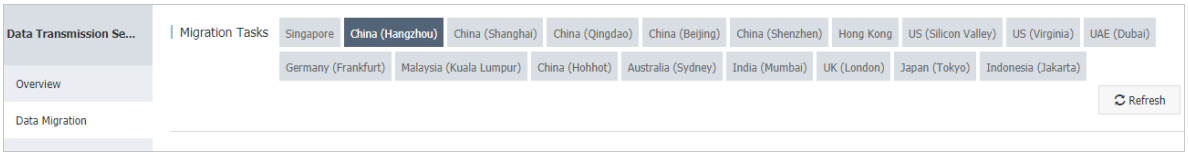
- CREATE USER and GRANT for a user-created Oracle database
- Create an account for a PolarDB cluster

Notice:
If you want to migrate incremental data from an Oracle database but the DBA permission cannot be granted to the database account, you can grant fine-grained permissions to the account. The following sample statements show you how to grant specific permissions to an Oracle database account.

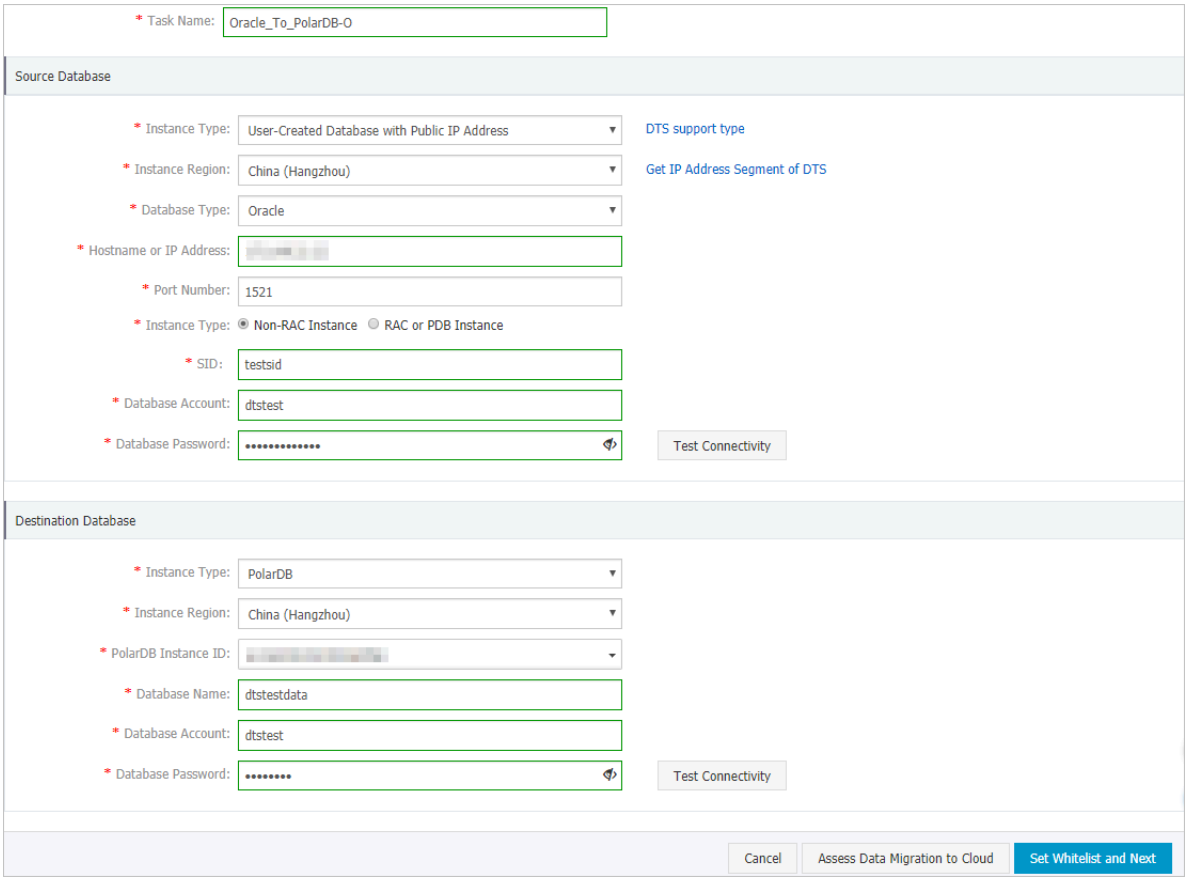
Procedure

1. Log on to the DTS console.

2. In the left-side navigation pane, click **Data Migration**.
3. At the top of **Migration Tasks** the 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 for the data migration task.



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

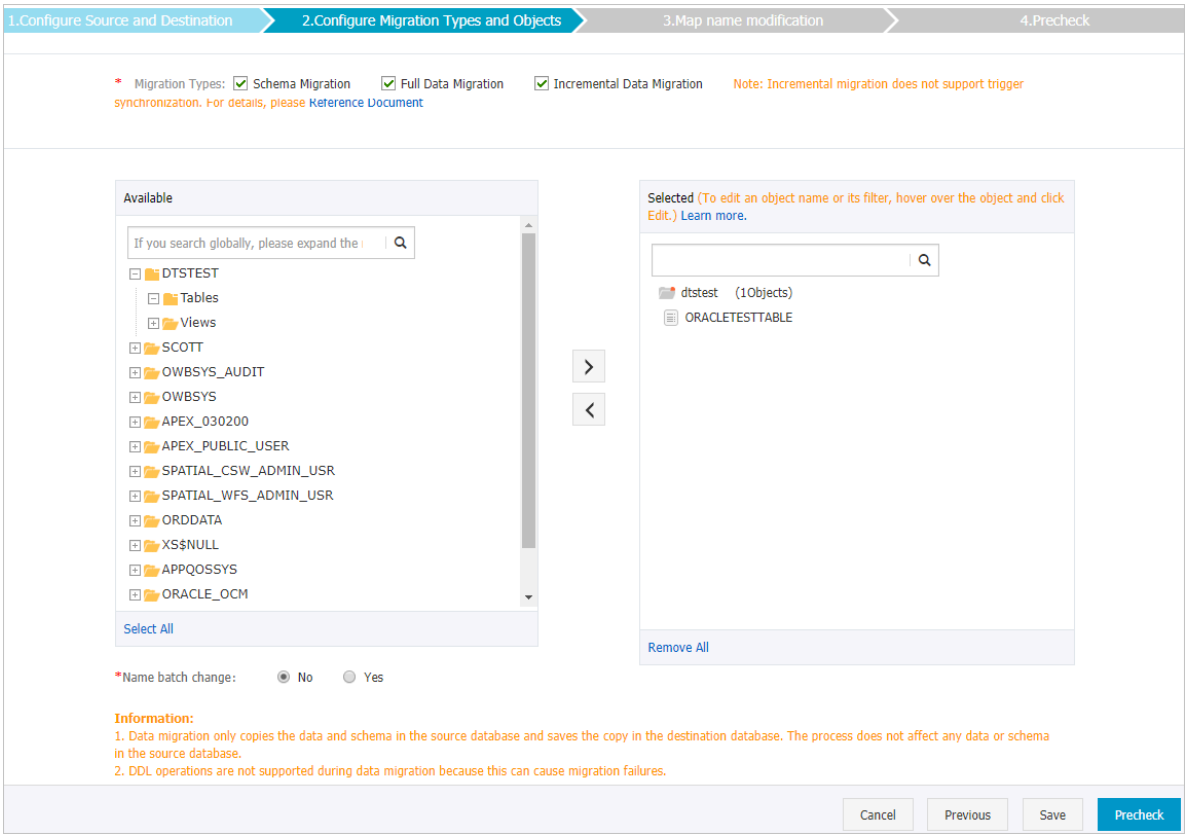
Section	Parameter	Description
Source Database	Instance Type	Select an instance type based on where the source database is deployed. The procedure in this topic uses **User-Created Database with Public IP Address** as an example. **Note:** If you select other instance types, you must prepare the environments that are required for the source database. For more information, see #unique_13.
	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**. **Note:** If a whitelist is configured for the user-created Oracle database, you must manually add the CIDR blocks of DTS servers to the whitelist of the user-created Oracle database. You can click **Get IP Address Segment of DTS** next to **Instance Region** to obtain the CIDR blocks of DTS servers.
	Database Type	Select **Oracle**.
	Hostname or IP Address	Enter the endpoint that is used to connect to the user-created Oracle database.
	Port Number	Enter the service port number of the user-created Oracle database. The default port number is **1521**. **Note:** In this example, The service port of the user-created Oracle database is accessible over the Internet.
	Instance Type	• **Non-RAC Instance**: If you select this option, you must specify the **SID**. • **RAC Instance**: If you select this option, you must specify the **Service Name**.
	Database Account	Enter the account of the user-created Oracle database. For more information about the permissions that are required for the account, see Preparations.

Section	Parameter	Description
	Database Password	Enter the password for the source 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.
Destination Database	Instance Type	Select **PolarDB**.
	Instance Region	Select the region where the destination PolarDB cluster resides.
	PolarDB Instance ID	Select the ID of the destination PolarDB cluster.
	Database Name	Enter the name of the destination database.
	Database Account	Enter the database account of the destination PolarDB cluster. For more information about the permissions that are required for the account, see Preparations.
	Database Password	Enter the password for the destination database account. **Note:** After you specify the destination database parameters, click **Test Connectivity** next to **Database Password** to verify whether the 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:
The CIDR blocks of DTS servers are automatically added to the whitelist of the destination PolarDB cluster. This ensures that DTS servers can connect to the destination PolarDB cluster.

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



Paramete	Description
Migration Types	• To perform only full data migration, select **Schema Migration** and **Full Data Migration**. • To migrate data with minimal downtime, select **Schema Migration**, **Full Data Migration**, and **Incremental Data Migration**. **Note:** If **Incremental Data Migration** is not selected, do not write data into the source database during full data migration. This ensures data consistency between the source and destination databases.

Parameter	Description
Objects	Select 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. • After an object is migrated to the destination database, the name of the object remains the same as that in the source database. You can change the names of the objects that are migrated to the destination database by using the object name mapping feature. For more information about how to use this feature, see #unique_14. • If you use the object name mapping feature on an object, other objects that are dependent on the object may fail to be migrated.

8. Click **Precheck** on the lower right of the page.

   Note:

   - A precheck is performed for a data migration task. A data migration task can be started only if it passes the precheck.
   - If the precheck fails, click corresponding to each failed item to view the details.

     Fix the problems as instructed and run the precheck again.

9. After the precheck is passed, click **Next**.
10. On the **Confirm Settings** dialog box that appears, specify **Channel Specification** and select the **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

  Do not manually stop a migration task. Otherwise, data migrated to the destination database will be incomplete. Wait until the data migration task stops when it is complete.

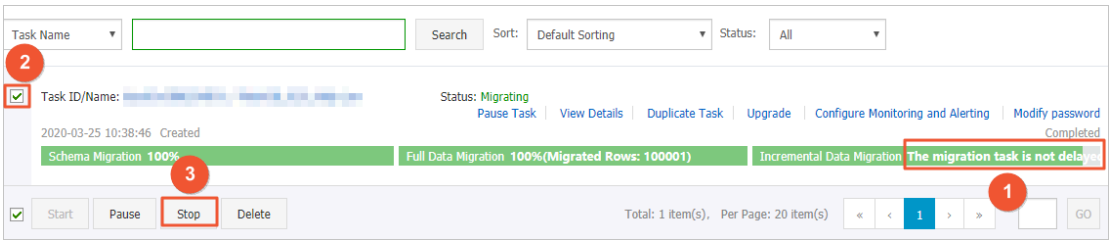
- Schema migration, full data migration, and incremental data migration

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

  > **Note:**
  > Select an appropriate time point to manually end the migration task. For example, you can end the migration task during off-peak hours or before you switch your workloads to the destination cluster.

  a. When the task progress bar switches to **Incremental Data Migration** and the message **The migration task is not delayed** appears, stop writing new data to the source database for a few minutes. Then, the progress bar will show the latency of the **incremental data migration**.

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



12. Switch your workloads to the destination cluster.

5 Pending events

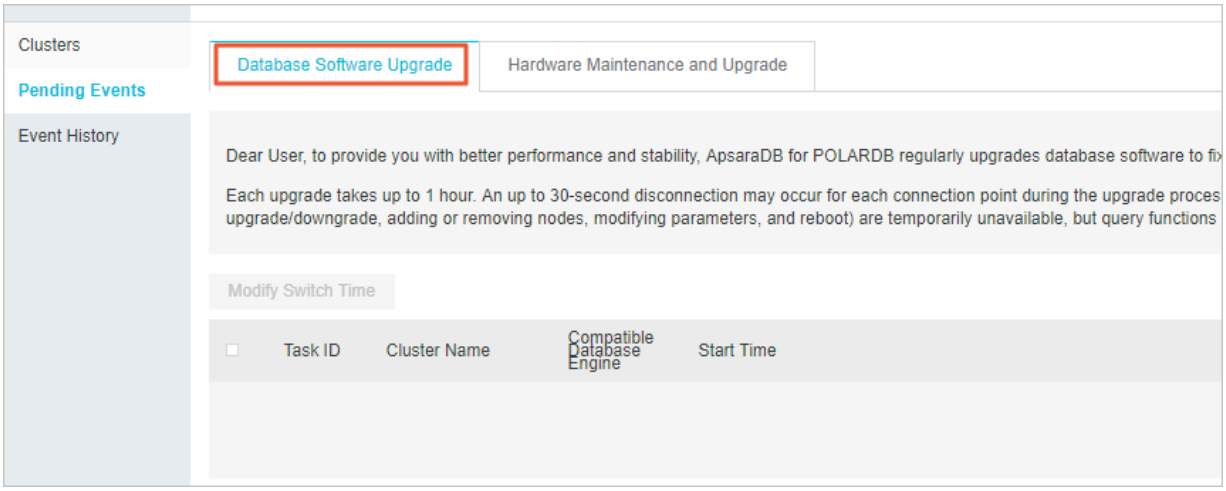
When an ApsaraDB for POLARDB event is pending for processing, you will be notified to handle the event in a timely manner in the console.

For ApsaraDB for POLARDB O&M events, including database software upgrade events and hardware maintenance and upgrade events, you are notified not only by SMS messages, phone calls, emails, or internal messages, but also in the console. You can view the details of each event, including the event type, task ID, cluster name, and switch time. You can also change the switch time.

Prerequisites

There are unprocessed O&M events.

Note:
If there are unprocessed O&M events, you can see notification badges on the **Pending Events** page.



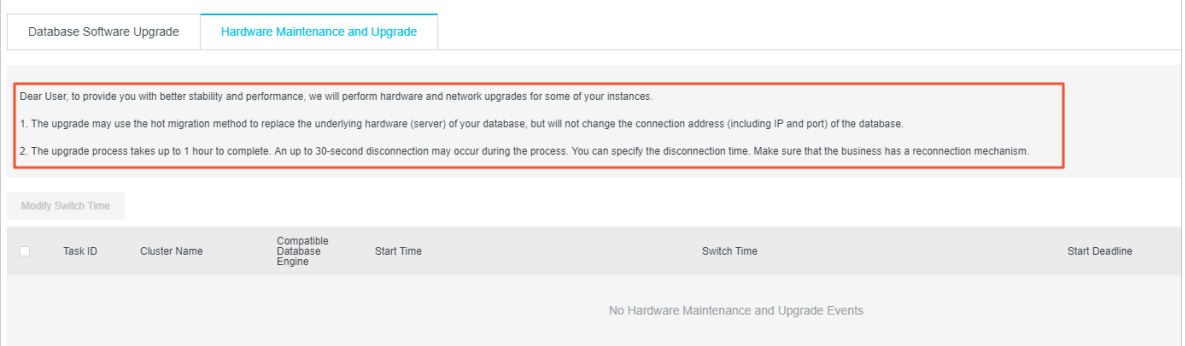
Change the switch time

1. Log on to the ApsaraDB for POLARDB console.
2. In the left-side navigation pane, click **Pending Events**.

Note:
For an O&M event for which you must reserve the switch time, a dialog box appears, asking you to complete the reservation as soon as possible.

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

> **Note:**
> Different notices are displayed on the tabs for different types of events.

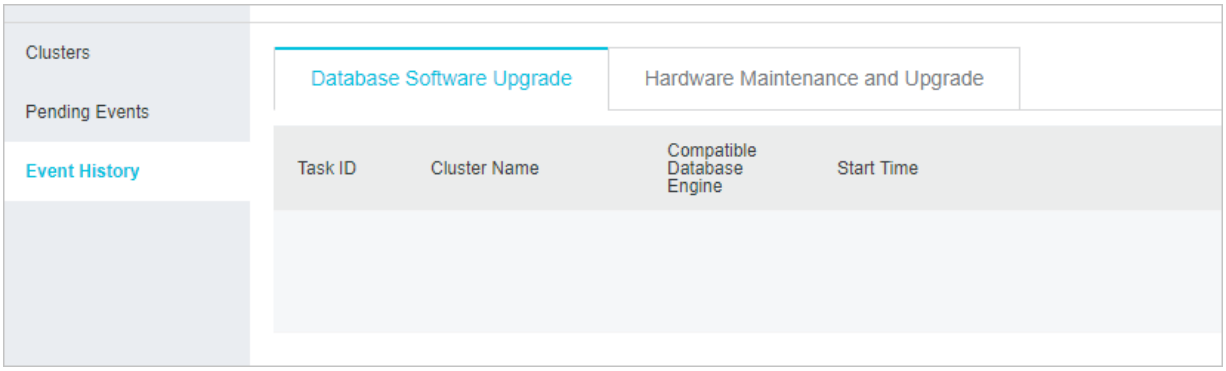


4. View event details in the event list. To change the switch time, select an event, and then click **Change Switch Time**. In the dialog box that appears, set the switch time, and then click **OK**.

> **Note:**
> The switch time cannot be later than the latest operation time allowed.

Historical events

You can view completed events on the **Event History** page.



6 Set a whitelist for a POLARDB cluster

Note:

A POLARDB cluster compatible with Oracle does not support setting a cluster whitelist.

Only instances in the same VPC can access the cluster.

7 Billing management

7.1 Change the billing method from pay-as-you-go to subscription

You can change the billing method of a cluster from pay-as-you-go to subscription based on your needs. Changing billing methods will not impact the performance of your cluster.

> **Note:**
> If a cluster uses a specification that is no longer available, you cannot change the billing method of the cluster to subscription. In this case, you need to Change the specifications of a POLARDB cluster before changing the billing method.

Precautions

You cannot change the billing method of a cluster from subscription to pay-as-you-go. Consider your resource requirements before switching the billing method to subscription to avoid resource wastage.

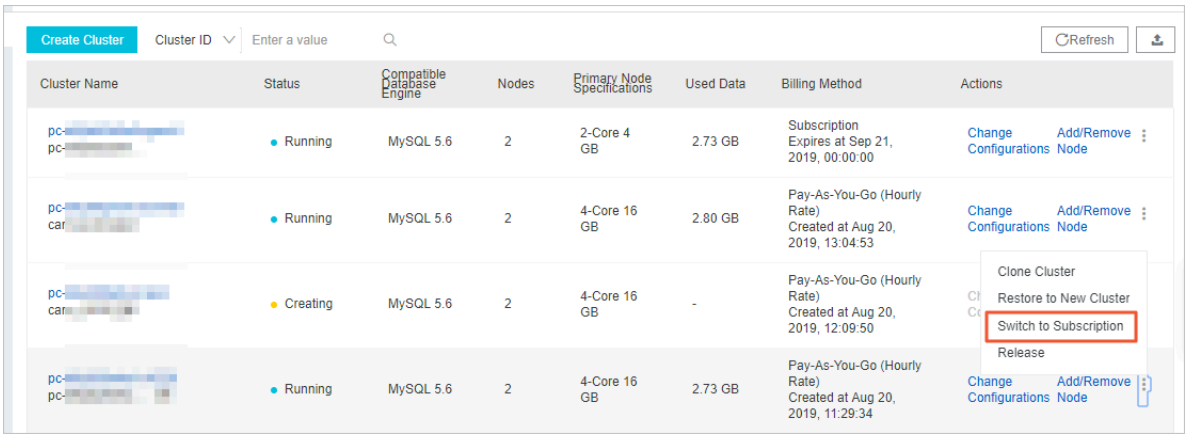
Prerequisites

- The cluster must be in the **Running** state.
- There are no pending orders for changing the billing method from pay-as-you-go to subscription. If there are any pending orders, you must complete payment for or discard them on the Orders page.

Procedure

1. Log on to the ApsaraDB for POLARDB console.
2. Select the region where the cluster resides.

3. Find the target cluster. In the **Actions** column corresponding to the cluster, choose **...** > **Switch to Subscription**.



4. Specify **Purchase Plan**, read the **ApsaraDB for POLARDB Subscription Agreement of Service**. Select the check box to indicate that you agree to it, and then click **Pay** to complete the payment.

Note:

- The new billing method will take effect after you complete the payment.
- If the order is unpaid or payment is unsuccessful, an unfinished order will be listed on the Orders page. You cannot buy a new cluster or switch the billing method of existing clusters before the unfinished order is complete. You must complete payment for or discard the order before placing a new one.

7.2 Manually renew the subscription to a cluster

You can renew your subscription to clusters in the ApsaraDB for PolarDB console or in the Renew console. In the Renew console, you can renew your subscription to multiple clusters at the same time.

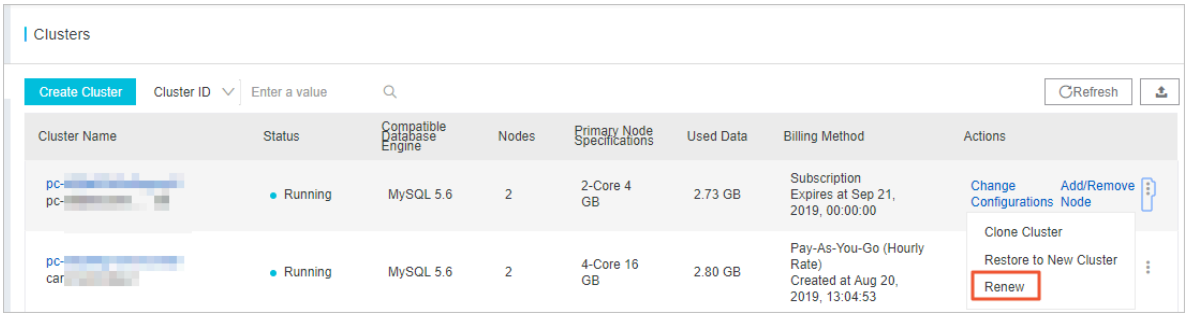
Note:

Clusters purchased through the pay-as-you-go (hourly rate) billing method do not involve expiration and renewal.

Method 1: Renew the subscription in the ApsaraDB for PolarDB console

1. Log on to the ApsaraDB for PolarDB console.
2. Select a region in the upper-left corner to view all the clusters that you deploy in this region.

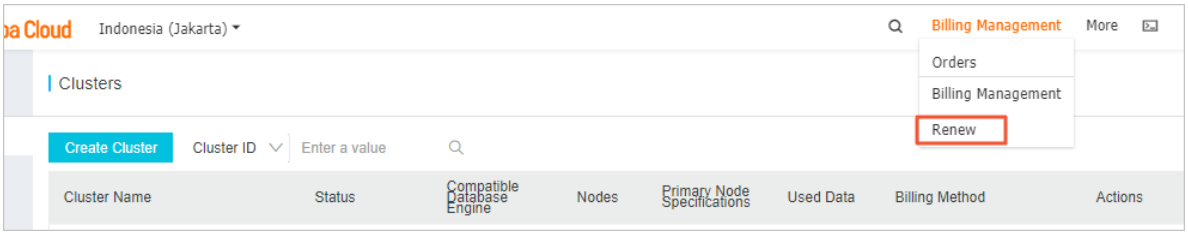
3. Find the target cluster, click the **More** icon in the **Actions** column, and choose **Renew** from the shortcut menu.



4. Specify the renewal duration, select the service agreement, and click **Pay**.

Method 2: Renew the subscription in the Renew console

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



3. In the left-side navigation pane, click **ApsaraDB for PolarDB**.
4. Click the **Manually Renew** tab. Set the filtering conditions to find the target cluster. Click **Renew** in the **Actions** column corresponding to the cluster.

> **Note:**
> To enable manual renewal for a cluster on the **Auto-Renew** or **Don't Renew** tab, click **Enable Manual Renew**, and then click **OK** in the dialog box that appears.

5. Specify the renewal duration, select the service agreement, and click **Pay**.

Enable automatic renewal

If you enable automatic renewal, you will be free from regular manual renewal operations and concerns of service interruptions. For more information, see Automatically renew the subscription to a cluster.

7.3 Automatically renew the subscription to a cluster

A subscription-based cluster has a validity period. If the cluster is not renewed in a timely manner, service interruptions or even data loss will occur after it expires. If you enable

automatic renewal, you will be free from regular manual renewal operations and concerns of service interruptions.

> **Note:**
> Clusters purchased through the pay-as-you-go (hourly rate) billing method do not involve expiration and renewal.

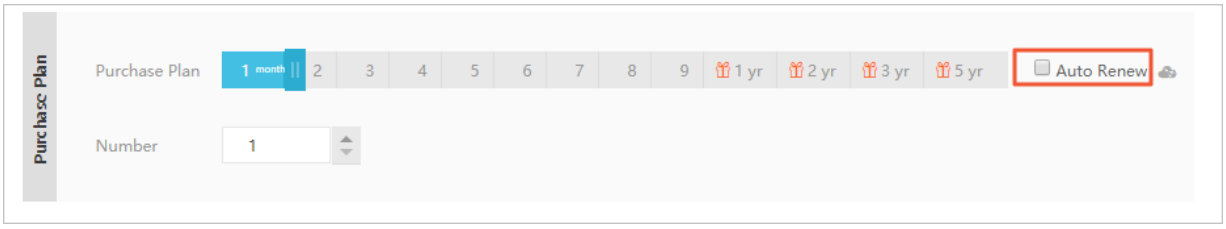
Precautions

- Automatic fee deduction will begin nine days prior to the expiration of the cluster, supporting cash and coupons. Keep your account balance adequate.
- If you manually renew the cluster before the automatic deduction, the system will automatically renew the cluster nine days prior to the next expiration.
- The automatic renewal feature takes effect the next day after it is enabled. If your cluster expires the next day, renew it manually to prevent service interruptions. For more information, see Manually renew the subscription to a cluster.

Enable automatic renewal when purchasing a cluster

> **Note:**
> After you enable automatic renewal, the system will automatically renew the subscription based on the **subscription period**. For example, if you purchase a cluster for three months and select automatic renewal, you will be charged a fee of the three-month subscription upon each automatic renewal.

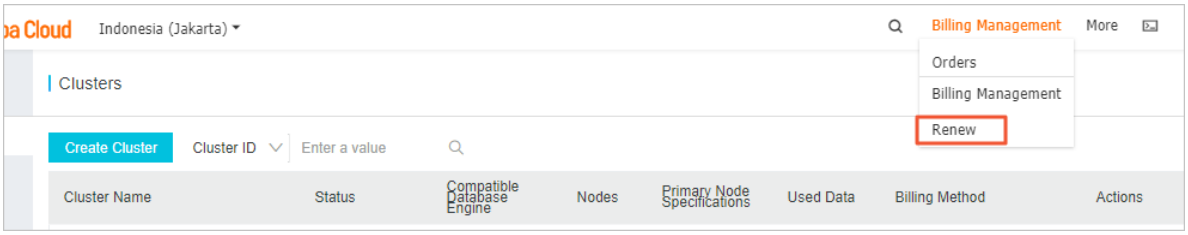
When creating a cluster, you can select **Auto Renew**.



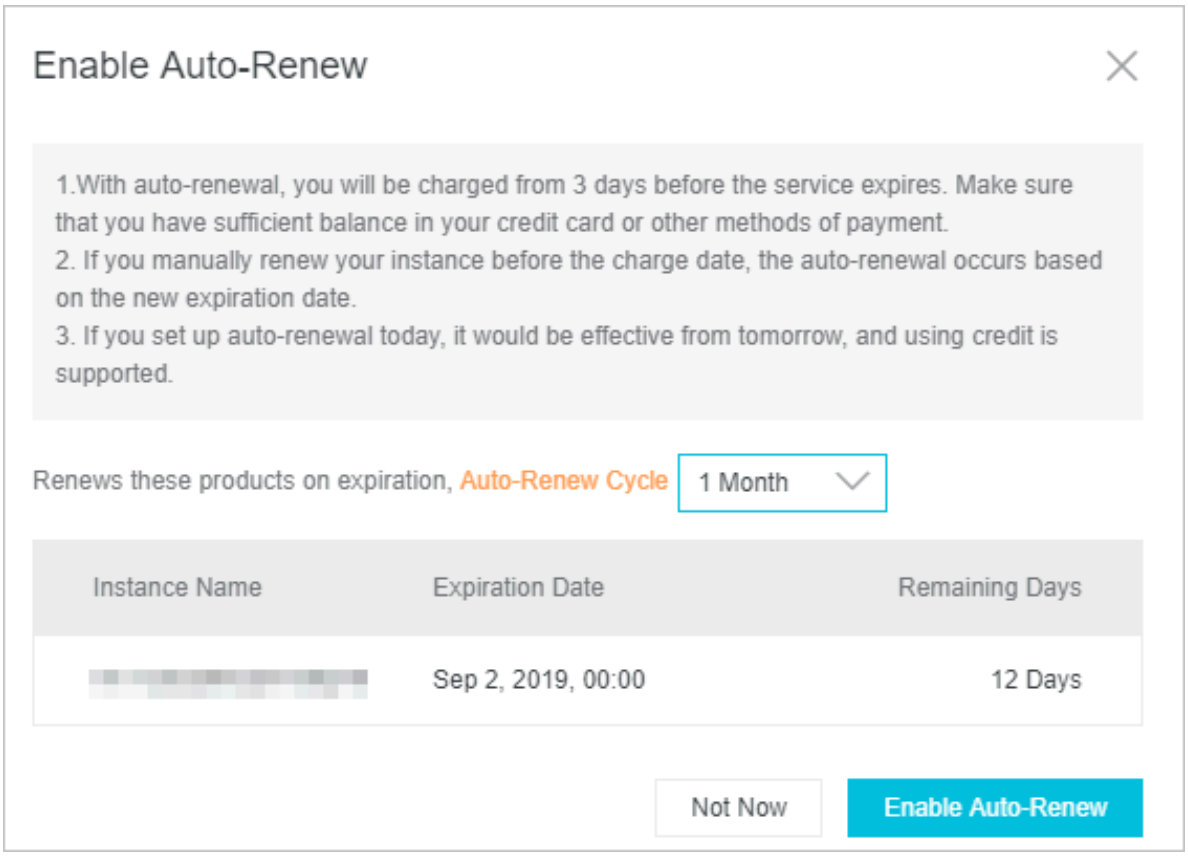
Enable automatic renewal after purchasing a cluster

> **Note:**
> After you enable automatic renewal, the system will automatically renew the subscription based on the renewal cycle you select. For example, if you select a three-month renewal cycle, you will be charged a fee of the three-month subscription upon each automatic renewal.

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



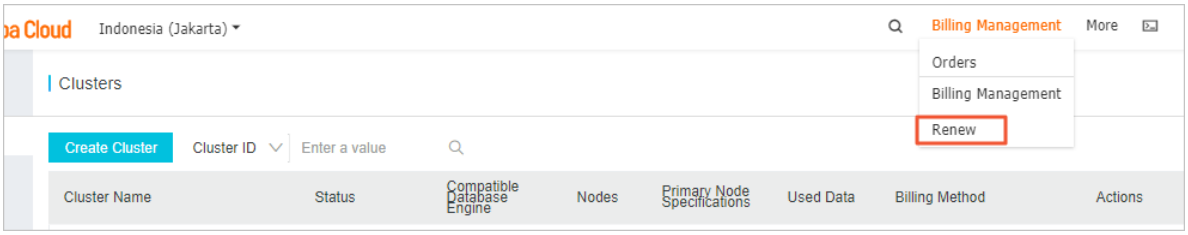
3. In the left-side navigation pane, click **ApsaraDB for POLARDB**.
4. Click the **Manually Renew** or **Don't Renew** tab in the Renew console. Set the filtering conditions to find the target cluster. Click **Enable Auto-Renew** in the **Actions** column corresponding to the cluster.
5. In the dialog box that appears, select the automatic renewal cycle, and click **Enable Auto-Renew**.



Edit the automatic renewal cycle

1. Log on to the ApsaraDB for POLARDB console.

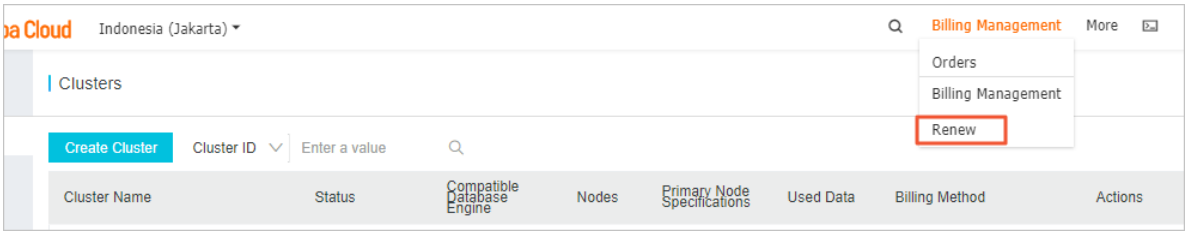
2. In the upper-right corner of the console, choose **Billing Management** > **Renew**.



3. In the left-side navigation pane, click **ApsaraDB for POLARDB**.
4. Click the **Auto-Renew** tab on the Renew console. Set the filtering conditions to find the target cluster. Click **Enable Auto-Renew** in the **Actions** column corresponding to the cluster.
5. Click the **Auto** tab. Set the filtering conditions to find the target cluster. Click **Modify Auto-Renew** in the **Actions** column corresponding to the cluster.
6. In the dialog box that appears, edit the automatic renewal cycle, and click **OK**.

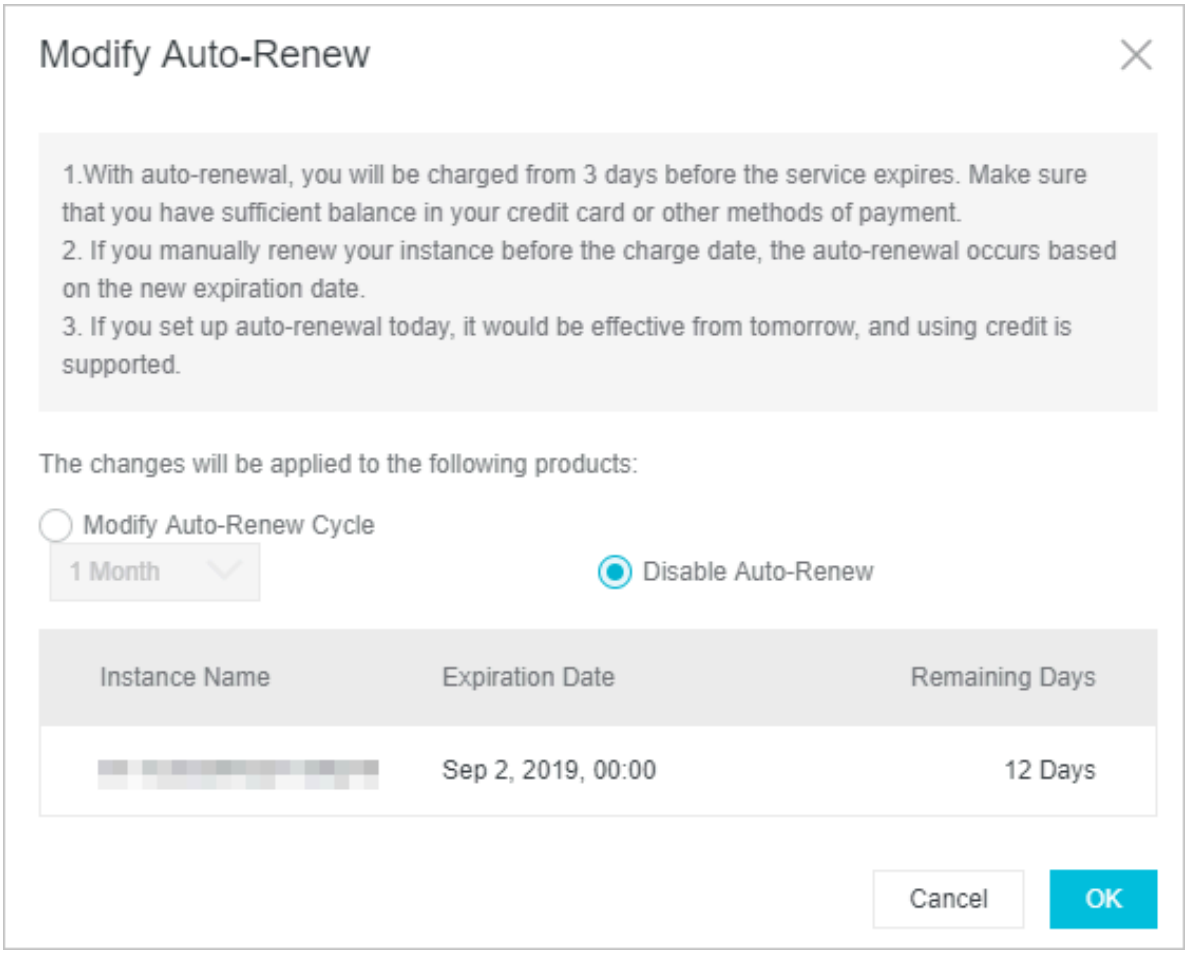
Disable automatic renewal

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



3. In the left-side navigation pane, click **ApsaraDB for POLARDB**.
4. Click the **Auto-Renew** tab in the Renew console. Set the filtering conditions to find the target cluster. Click **Modify Auto-Renew** in the **Actions** column corresponding to the cluster.

5. Select **Disable Auto-Renew** and click **OK**.



Related API operations

API operation	Description
#unique_21	Creates a POLARDB cluster. **Note:** You can enable automatic renewal when you create a cluster.
#unique_22	Enables automatic renewal for a subscription-based cluster. **Note:** You can enable automatic renewal after you create a cluster.
#unique_23	Queries the automatic renewal status of a subscription-based cluster.

8 Connect to a database cluster

8.1 View endpoints

This topic describes how to view endpoints of a POLARDB cluster compatible with Oracle and introduces primary endpoints and private endpoints.

Procedure

1. Log on to the ApsaraDB for POLARDB console.
2. Find the target cluster and click the cluster ID.
3. In the **Connection Information** section, view the endpoints.

Primary endpoints

Type	Description	Supported network type
Primary endpoint	A primary endpoint is always destined for the primary node and supports read and write operations. If the primary node fails, the primary endpoint is automatically switched to the read-only node that is promoted to the primary node.	Private network
Primary node endpoint (not recommended)	The endpoint of the primary node. We recommend that you do not connect directly to the primary node because the node becomes unavailable when it fails.	Private network
Read-only node endpoint (not recommended)	The endpoint of the read-only node. We recommend that you do not connect directly to the read-only node because the node becomes unavailable when it fails.	Private network

Private endpoints

Type	Description	Scenario
Private endpoint	• ApsaraDB for POLARDB can achieve optimal performance when it is accessed through the private endpoint. • The private endpoint cannot be released.	Examples: • If your ECS instance is located in the same VPC as the cluster, then your ECS instance can access the cluster through the private endpoint. • You can use DMS to connect to a cluster over a VPC.

Next steps

#unique_26

Related API operations

API operation	Description
#unique_27	Queries the endpoint information of a specified POLARDB cluster.
#unique_28	Creates public-facing endpoints for a specified POLARDB cluster.
#unique_29	Modifies the prefix of the default endpoint and primary endpoint for a specified POLARDB cluster.
#unique_30	Release an endpoint of a specified POLARDB cluster.

8.2 Connect to a database cluster

This topic describes how to use DMS and a client to connect to a POLARDB cluster compatible with Oracle.

Use DMS to connect to a POLARDB cluster compatible with Oracle

Data Management (DMS) provides an integrated solution for data management. DMS supports data management, schema management, access control, BI charts, trend analysis, data tracing, performance optimization, and server management. DMS supports relational databases such as MySQL, SQL Server, and PostgreSQL, as well as NoSQL databases such as MongoDB and Redis. DMS also supports the management of Linux servers.

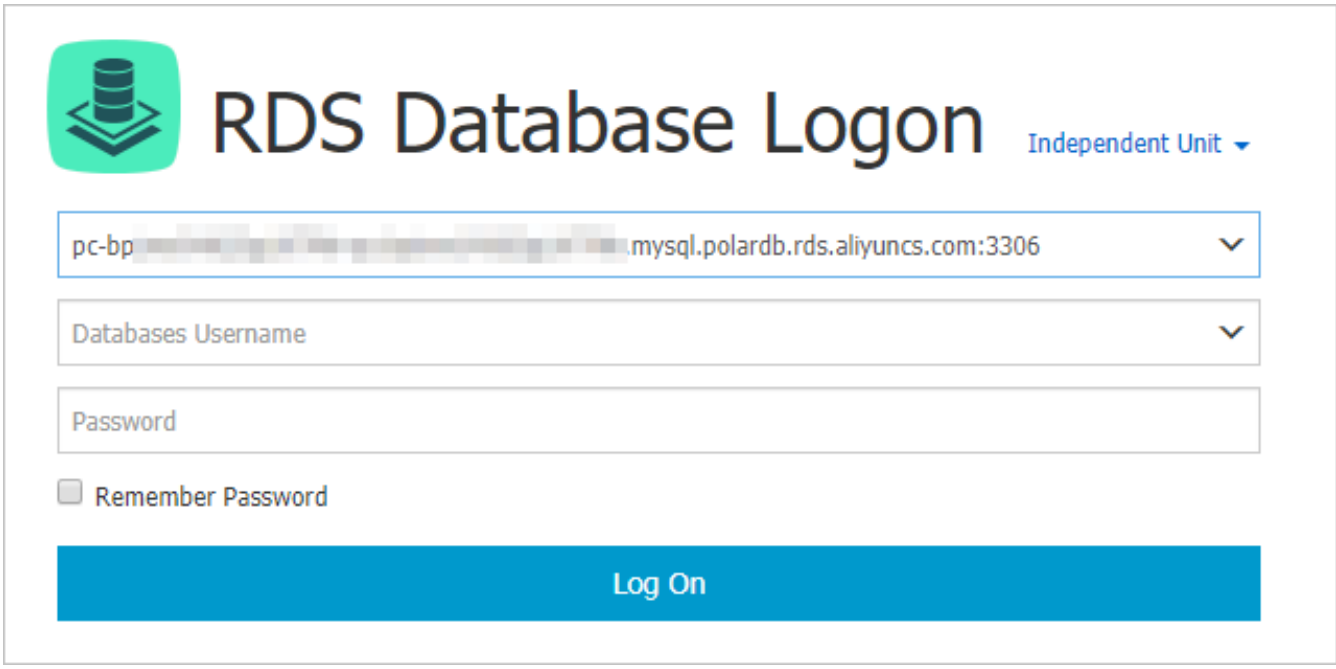
Prerequisites:

You have created a privileged account or a standard account for an existing database cluster. For more information, see Create a database account.

Procedure:

1. Log on to the ApsaraDB for POLARDB console.
2. Find the target cluster and click the cluster ID to go to the **Overview** page.

3. In the upper corner of the page, click **Log On to Database** to go to the **Database Logon** page.



4. On the Database Logon page, enter the primary endpoint and the port number, and separate them with a colon (:). Then enter the username and password of the privileged or standard account, and click **Log On**.

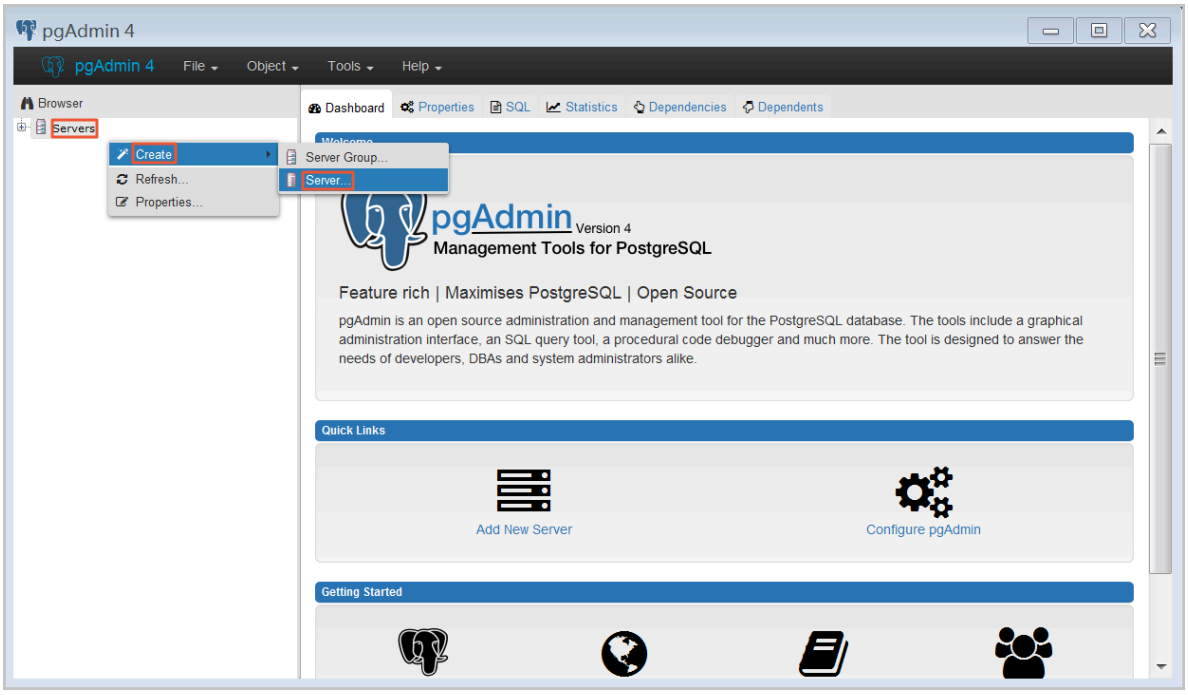
Note:
When connecting to a cluster through DMS, you can only use the primary endpoint. Cluster endpoints are not supported. For more information about how to view the endpoint, see View endpoints.

Use a client to connect to a POLARDB cluster compatible with Oracle

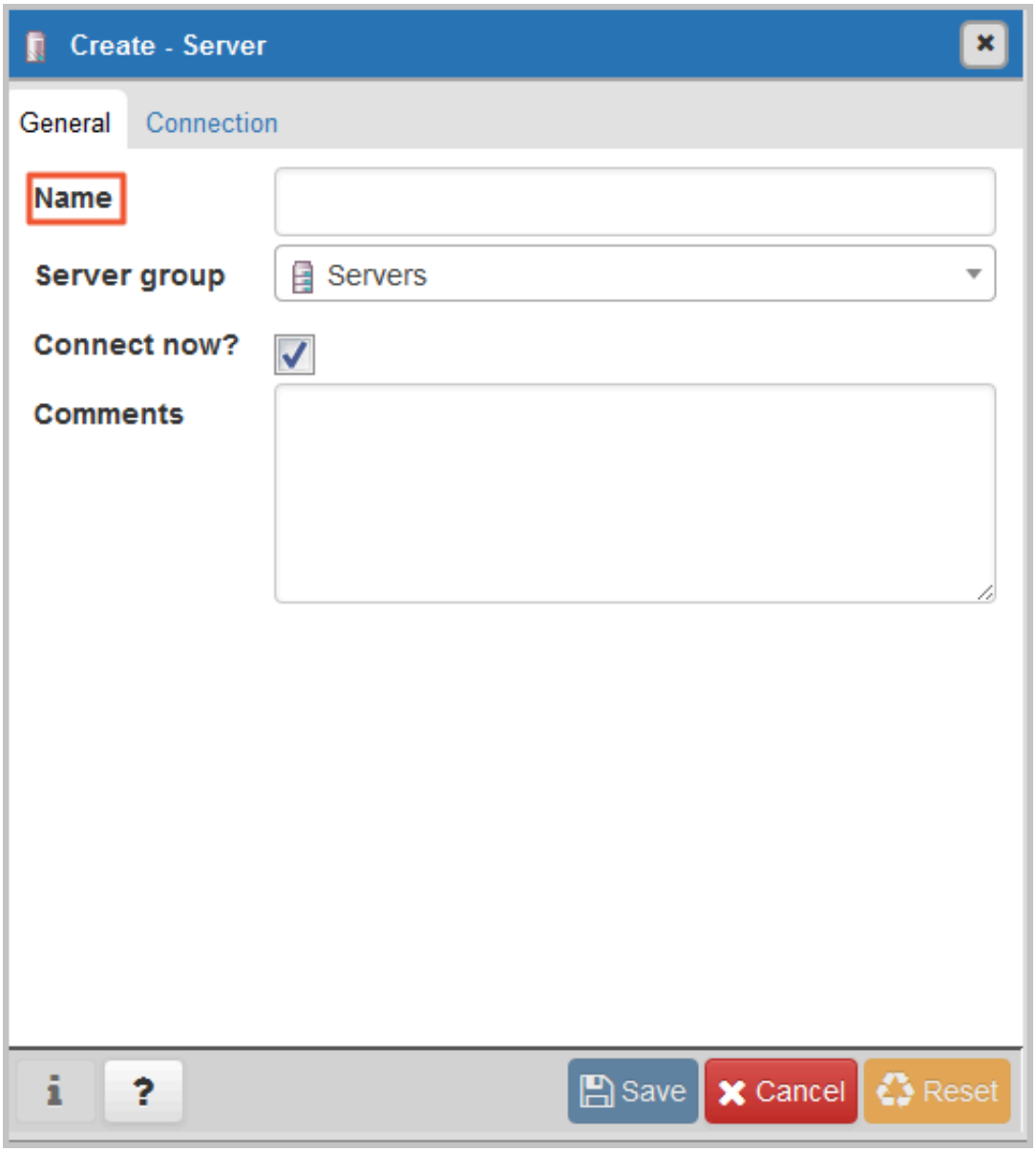
You can also use the pgAdmin 4 client to connect to a POLARDB cluster compatible with Oracle. POLARDB compatible with Oracle does not support configuring a whitelist for a

cluster. Only instances that are in the same VPC can access the cluster. Therefore, the server where the client resides and the POLARDB cluster compatible with Oracle must be in the same VPC.

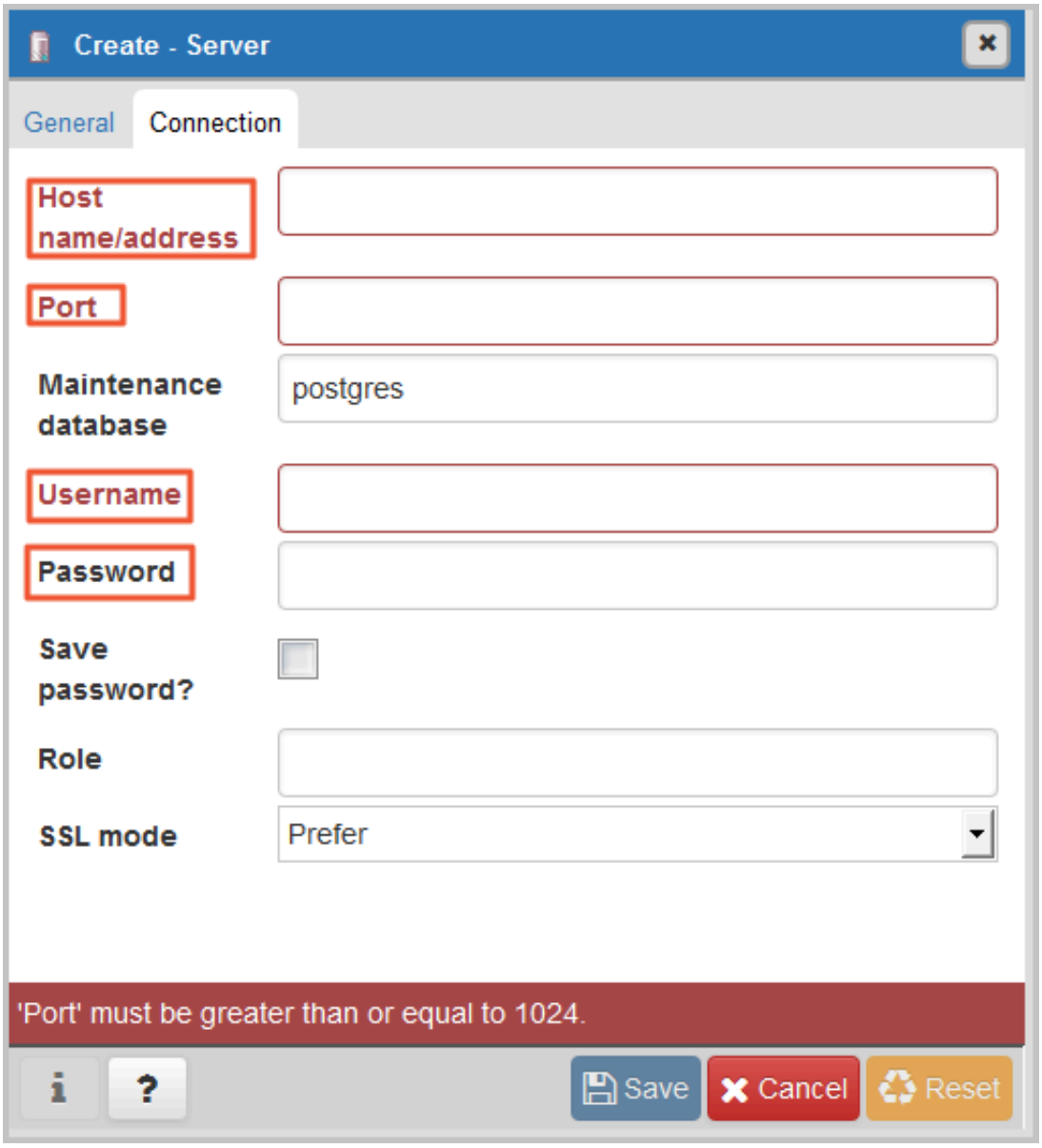
1. Start the pgAdmin 4 client.
2. Right-click **Servers** and choose **Create** > **Server** from the shortcut menu, as shown in the following figure.



3. On the **General** tab of the **Create - Server** dialog box, enter the name of the server, as shown in the following figure.



4. Click the **Connection** tab and enter the information of the destination instance, as shown in the following figure.



Parameter description:

- **Hostname/address**: the internal endpoint of the POLARDB cluster compatible with Oracle. To view the endpoint and port information of the POLARDB cluster compatible with Oracle, follow these steps:
  a. Log on to the ApsaraDB for POLARDB console.
  b. Find the target cluster and click the cluster ID.
  c. In the **Connection Information** section, view the endpoint and port information.
- **Port**: the internal port of the POLARDB cluster compatible with Oracle.

- **Username**: the name of the privileged account of the POLARDB cluster compatible with Oracle.
- **Password**: the password of the privileged account of the POLARDB cluster compatible with Oracle.

5. Click **Save**.
6. If the connection information is correct, choose **Servers** > **Server Name** > **Databases** > **postgres**. The connection is successful if the following interface is displayed.

> **Note:**
> postgres is the default system database of the POLARDB compatible with Oracle. Do not perform any operation on the database.

Use the CLI to connect to a POLARDB cluster compatible with Oracle

Additionally, you can run commands on clients to connect to a POLARDB cluster compatible with Oracle. For more information about how to download and install the clients, see #unique_33.

9 Cluster management

9.1 Create a POLARDB cluster compatible with Oracle

This topic describes how to create a POLARDB cluster compatible with Oracle in the console.

Prerequisites

You have created an Alibaba Cloud account or created a Resource Access Management (RAM) user account.

- Click here to create an Alibaba Cloud account.
- For more information about how to create and grant permissions to a RAM user account, see Create and authorize a RAM user.

Context

A cluster contains one primary node and up to 15 read-only nodes. At least one read-only node is required to implement active-active high availability architecture. A node is a virtual database server. You can create and manage multiple databases in a node.

Note:

- ApsaraDB for POLARDB supports only Virtual Private Cloud (VPC). VPC is an isolated network in the Alibaba Cloud and is more secure than a classic network.
- To achieve optimal performance, use ApsaraDB for POLARDB with Elastic Compute Service (ECS) and deploy them in the same VPC. If your ECS instance is created in a classic network, you must migrate it to a VPC.

Procedure

1. Log on to the Alibaba Cloud.
   - Click here to log on with your Alibaba Cloud account.
   - Click here to log on with your RAM user account. For more information, see Create and authorize a RAM user.
2. Click **Create Cluster** to go to the ApsaraDB for POLARDB purchase page.

3. Select **Subscription** or **Pay-As-You-Go**.
   - **Subscription**: An upfront payment is required for the compute node (a primary node and a read-only node) when you create the cluster. Storage consumed by your database is billed in GB/hour increments and the charges are deducted from your account on an hourly basis. The **Subscription** method is more cost-effective if you plan to use the new cluster for a long period of time. The longer the subscription period, the greater the discount.
   - **Pay-As-You-Go**: This method does not require any upfront payment. Compute nodes and storage consumed by your database are billed on an hourly basis and the charges are deducted from your account on an hourly basis. The **Pay-As-You-Go** method is suitable if you only want to use the new cluster for a short period of time. You can save costs by releasing clusters as needed.
4. Set the following parameters.

Console section	Parameter	Description
Basic	Region	The region where the cluster resides. You cannot change the region after you confirm your order.  **Note:** Make sure that you deploy your cluster in the same VPC as the ECS instance you want to connect to. Otherwise, the instances cannot communicate through the internal network and optimal performance cannot be achieved.
	Primary Availability Zone	• The zone of the cluster. Zones are independent physical areas in one region. There are no differences between the zones. • Your cluster and the ECS instance to be connected can be located in the same zone or in different zones.
	Network Type	• You do not need to specify this parameter. • ApsaraDB for POLARDB supports VPC only. A VPC is an isolated virtual network with higher security and better performance than a classic network.

Console section	Parameter	Description
	VPC VSwitch	Make sure that you deploy your cluster in the same VPC as the ECS instance you want to connect to. Otherwise, the instances cannot communicate through the internal network and optimal performance cannot be achieved. • If you have created a VPC that meets your network plan, select the VPC. For example, if you have created an ECS instance and the VPC where it resides meets your network plan, select this VPC. • Otherwise, use the default VPC and VSwitch. - Default VPC: ■ It is a unique VPC in the selected zone. ■ The network mask for a default VPC has 16 bits, such as 172.31.0.0/16, providing up to 65,536 internal IP addresses. ■ It is not included in the total number of VPCs that you can create. - Default VSwitch: ■ It is a unique VSwitch in your selected zone. ■ The network mask for a default VSwitch has 20 bits, such as 172.16.0.0/20, providing up to 4,096 private IP addresses. ■ The default VSwitch is not included in the total number of VSwitches that you can create in a VPC. • If the default VPC and VSwitch cannot satisfy your requirements, you can create your own VPC and VSwitch.
Instance	Database Engine	• Fully compatible with MySQL 8.0. Native concurrent queries are supported, in specific scenarios such as TPC-H tests, performance increases tenfold. For more information, see #unique_37. • Fully compatible with MySQL 5.6. • Fully compatible with PostgreSQL 11. • Compatible with Oracle (highly compatible).
	Node Specifications	Select the specifications as needed. All ApsaraDB for POLARDB nodes are dedicated, providing stable and reliable performance. For more information, see #unique_39.

Console section	Parameter	Description
	Number of Nodes	• You do not need to specify this parameter. By default, the system creates a read-only node that has the same specifications as the primary node. • If the primary node fails, the system automatically promotes the read-only node as the primary node, and creates a new read-only node. • For more information about read-only nodes, see #unique_40 .
	Storage Cost	You do not need to specify this parameter. The system charges you on an hourly basis based on the actual data usage. For more information, see #unique_39. **Note:** You do not need to select a storage capacity when you purchase a cluster. The storage capacity can automatically resize based on your data usage.

5. Specify **Purchase Plan** (only applicable to subscription clusters) and **Number**, and click **Buy Now**.

   Note:
   A maximum of 50 clusters can be created at a time, which is suitable for business scenarios such as launching multiple gaming servers at a time.

6. On the **Confirm Order** page, confirm your order information, read and confirm you agree to **ApsaraDB for POLARDB Subscription Agreement of Service** by selecting the checkbox, and then click **Pay**.

   After the payment is completed, the cluster is created in about 10 minutes. The created cluster is displayed in the cluster list.

   Note:

   - If some nodes in the cluster are still in the **Running** state, the cluster is still being created and is unavailable. The cluster is only available when the cluster is in the **Running** state.
   - Make sure that you have selected the correct region. Otherwise, you cannot view your cluster.

Next steps

Create a database account

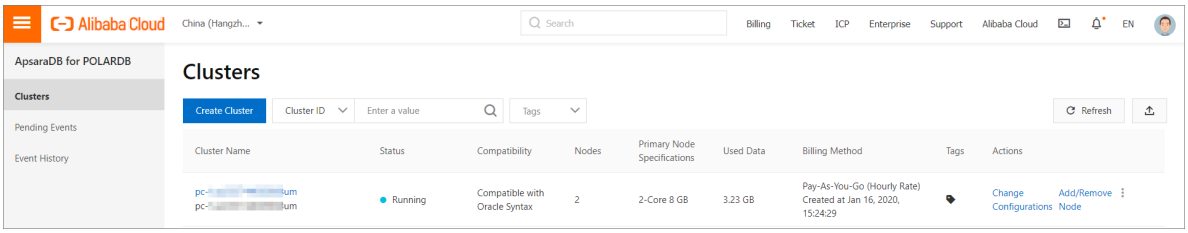
Related API operations

API operation	Description
#unique_21	Creates a POLARDB cluster.
#unique_41	Queries POLARDB clusters.
#unique_42	Queries the detailed information of a specified POLARDB cluster.
#unique_23	Queries the auto renewal details of subscription POLARDB clusters.
#unique_22	Sets the auto renewal attributes for a specified subscription POLARDB cluster.

9.2 View clusters

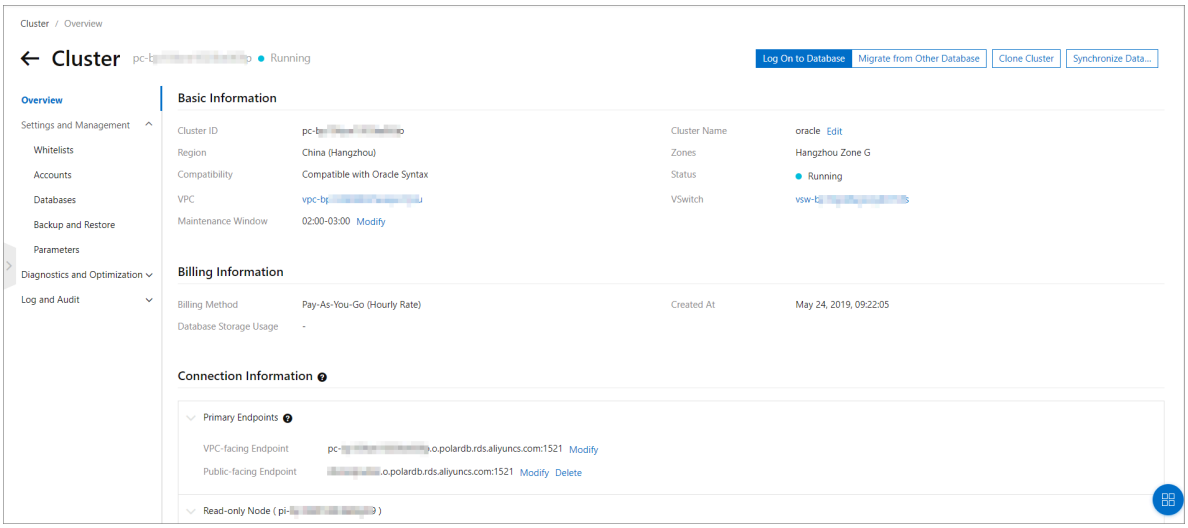
View the list of clusters and their details

1. Log on to the ApsaraDB for POLARDB console.
2. In the upper-left corner, select a region to view all the clusters that you have deployed in the region.



3. Click a cluster ID to go to the cluster details page.

   The details page displays the basic, billing, connection, and node information about the cluster.



Related API operations

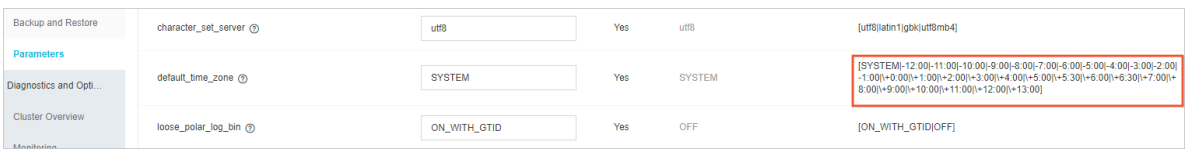
API operation	Description
#unique_21	Creates an ApsaraDB for POLARDB cluster.
#unique_41	Queries a list of ApsaraDB for POLARDB clusters.
#unique_42	Queries the detailed information of a specified ApsaraDB for POLARDB cluster.

9.3 Configure cluster parameters

This topic describes how to modify parameter values of a cluster in the ApsaraDB for POLARDB console.

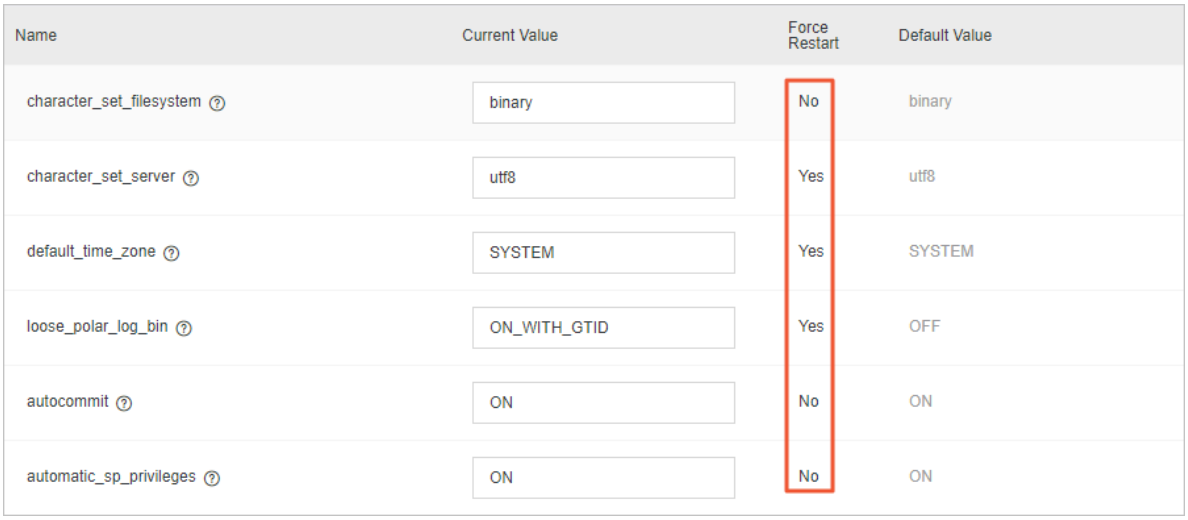
Precautions

- You must modify parameter values according to the **Value Range** column on the Parameters page.



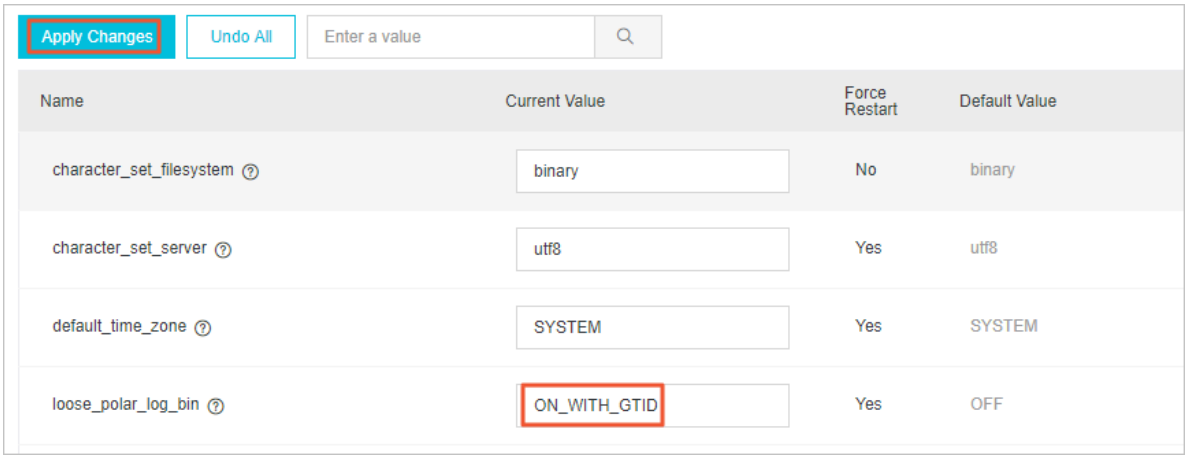
- For some parameters, you must restart all nodes after the parameter values are modified. We recommend that you make appropriate service arrangements before you

restart the nodes. Proceed with caution. You can determine whether the modification of a parameter value requires a node restart according to the value in the **Force Restart** column on the **Parameters** page.

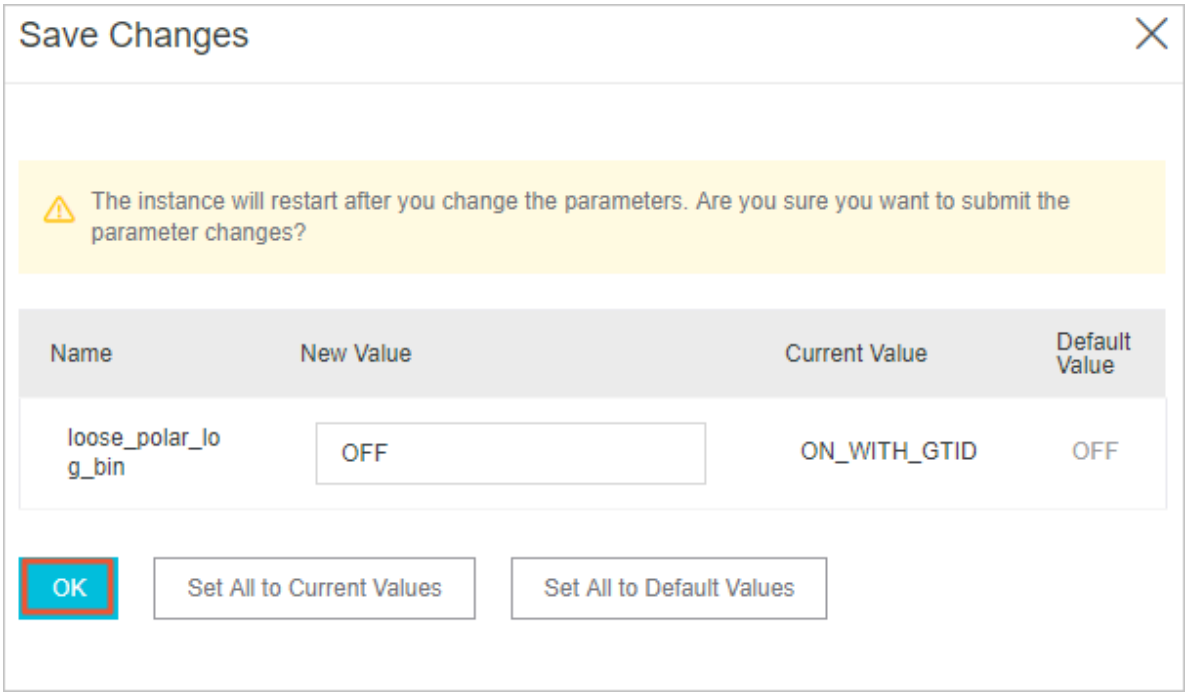


Procedure

1. Log on to the ApsaraDB for POLARDB console.
2. Select a region.
3. Find the target cluster and click the cluster ID in the **Cluster Name** column.
4. In the left-side navigation pane, choose **Settings and Management** > **Parameters**.
5. Modify the values of one or more parameters in the **Current Value** column, and click **Apply Changes**.



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



Related API operations

API operation	Description
#unique_45	Views cluster parameters.
#unique_46	Modifies the values of cluster parameters.

9.4 Change the specifications of a POLARDB cluster

This topic describes how to upgrade or downgrade the specifications of an ApsaraDB for POLARDB cluster to meet your business requirements.

ApsaraDB for POLARDB supports capacity scaling in three dimensions:

- Vertical scaling of computing power

  You can upgrade or downgrade the specifications of a cluster. This topic describes the details.
- Horizontal scaling of computing power

  You can add or delete read-only nodes. For more information, see Add or remove a read-only node.
- Horizontal scaling of storage capacity

  The storage capacity is provisioned in a serverless model. As your data increases in size, the storage is automatically expanded.

After you change the specifications of a POLARDB, it takes 5 minutes to 10 minutes for the new specifications of each node to take effect.

Specification change fees

For more information, see #unique_48.

Prerequisites

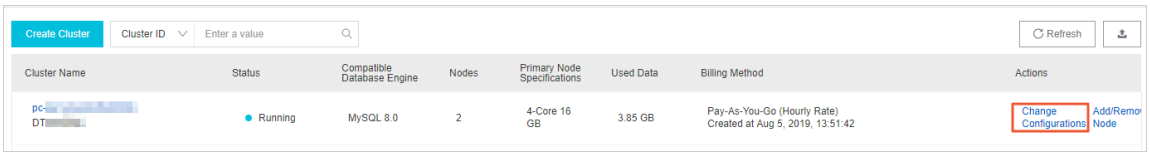
You can only change cluster specifications when the cluster does not have pending specification changes.

Precautions

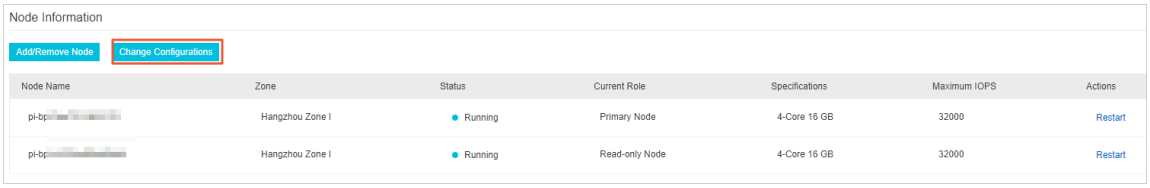
- Specification upgrades or downgrades only apply to clusters. You cannot change the specifications of a node.
- Specification upgrades or downgrades will not affect the existing data in the cluster.
- We recommend that you modify cluster specifications during off-peak periods. During a specification upgrade or downgrade, the ApsaraDB for POLARDB service will be disconnected for a few seconds and some of the functions will be disabled. You will need to reconnect from your applications once the service is disconnected.

Procedure

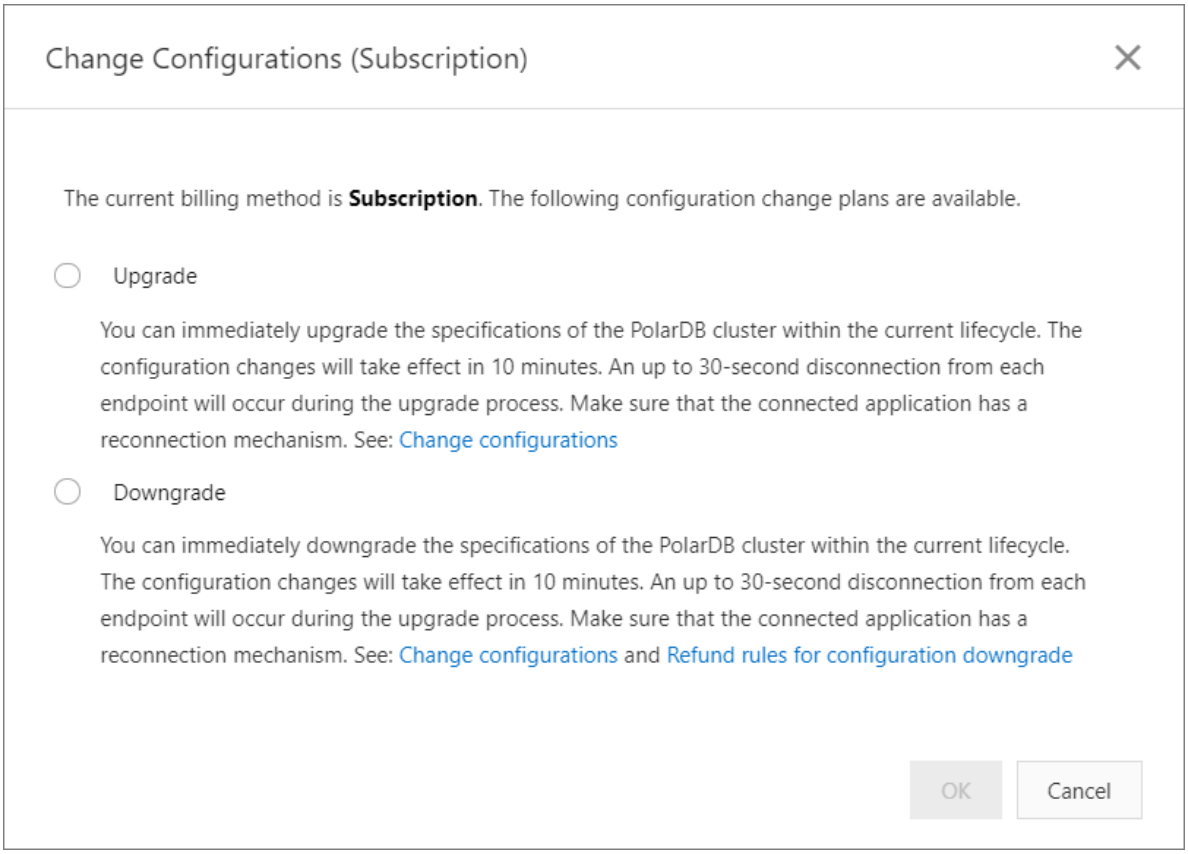
1. Log on to the ApsaraDB for POLARDB console.
2. In the upper-left corner of the page, select the region.
3. Go to the **Change Configurations** page. Perform the operation by using either of the following methods:
   - Find the target cluster and click **Change Configurations** in the **Actions** column corresponding to the target cluster.



   - Find the target cluster, click the cluster ID, and then click **Change Configurations** in the Node Information section.



4. Select **Upgrade** or **Downgrade** and click **OK**.



5. Select a specification.

> **Note:**
> All nodes in a cluster have the same specifications.

6. Read and agree to the service agreement by selecting the check box, and click **Pay** to complete the payment.

> **Note:**
> It takes about ten minutes for the new specifications to take effect.

APIs

API	Description
#unique_49	Changes the specifications of a POLARDB cluster.

9.5 Add or remove a read-only node

This topic describes how to manually add or remove a read-only node from an ApsaraDB for POLARDB cluster. An ApsaraDB for POLARDB cluster can contain up to 15 read-only

nodes, and must have at least one read-only node to ensure high availability. All nodes in a cluster have the same specifications.

You can manually add or remove read-only nodes after creating an ApsaraDB for POLARDB cluster.

Billing

The following describes the billing methods for nodes added to an existing cluster:

- If nodes are added to a subscription cluster, the nodes are billed as subscription nodes.
- If nodes are added to a pay-as-you-go cluster, the nodes are billed as pay-as-you-go nodes.

 **Note:**

- Read-only nodes that you purchase in either subscription or pay-as-you-go mode can be released at any time. After they are released, the system will Refund or stop billing.
- The added nodes are only charged based on the node specifications. For more information, see #unique_39. The storage fee is charged based on the actual data volume, regardless of the number of nodes.

Precautions

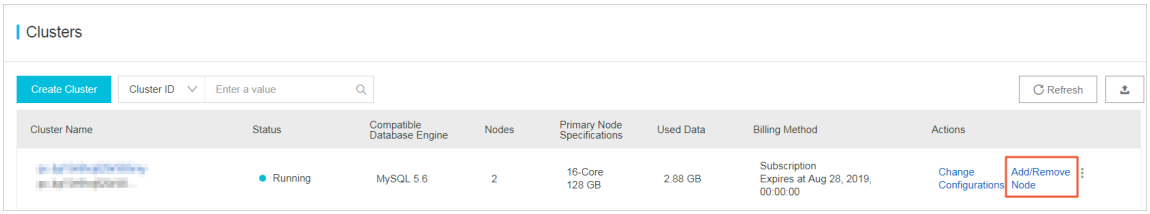
- You can only add or remove read-only nodes when the cluster does not have pending specification changes.
- To avoid misoperations, only one read-only node can be added or removed at a time. You need to perform the add or remove operation for each node.
- It takes about 5 minutes to add or remove a node.

Add a read-only node

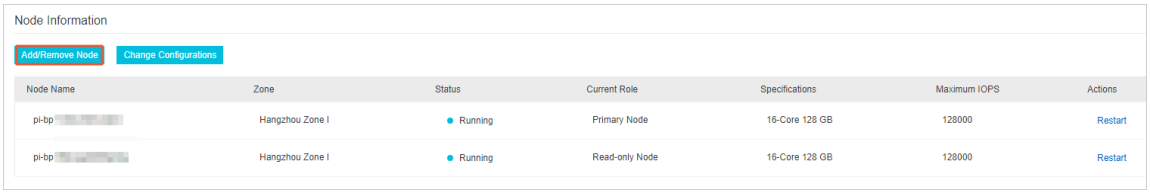
1. Log on to the ApsaraDB for POLARDB console.
2. Select a region.

3. Go to the **Add/Remove Node** page. Perform the operation by using either of the following methods:

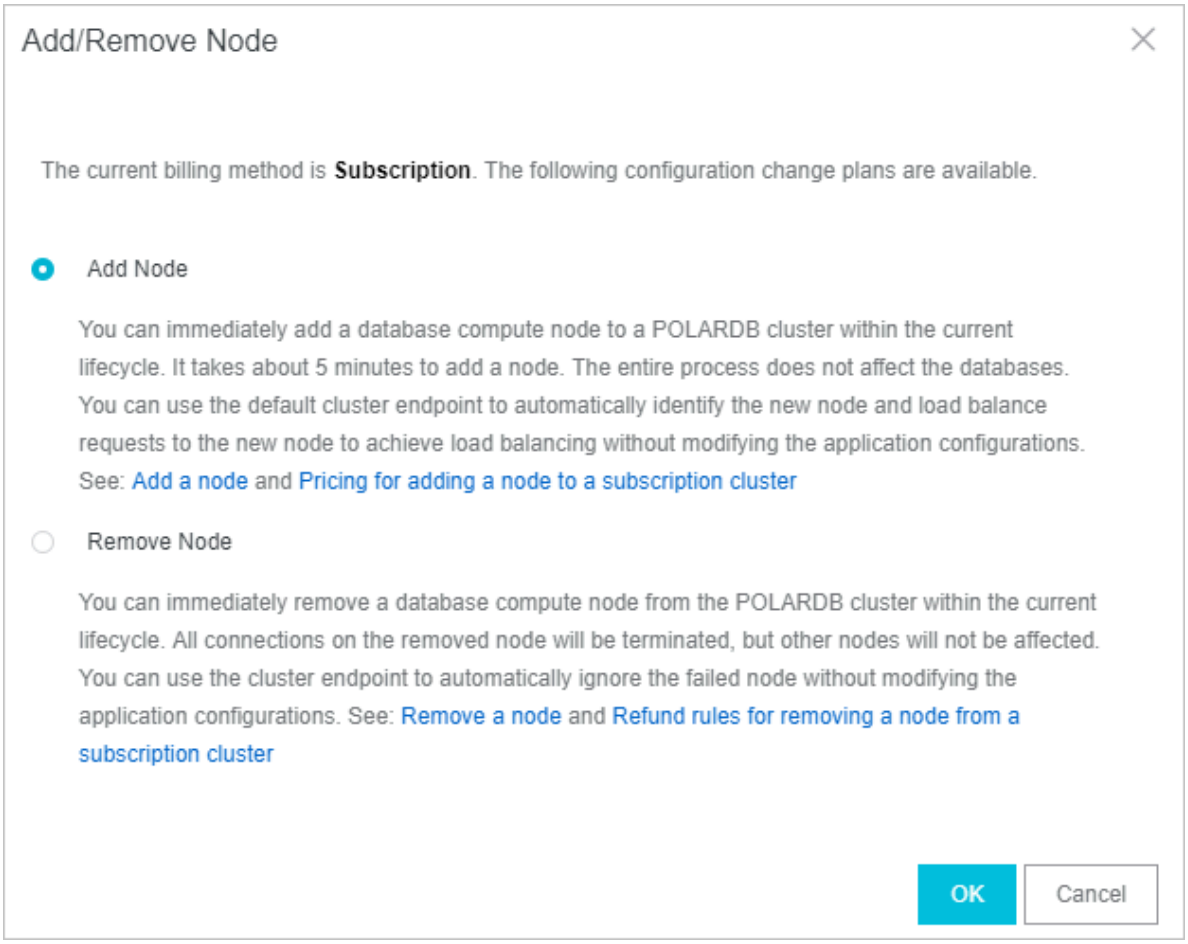
   - Find the target cluster and click **Add/Remove Node** in the **Actions** column corresponding to the cluster.



   - Find the target cluster, click the cluster ID, and then click **Add/Remove Node** in the Node Information section.



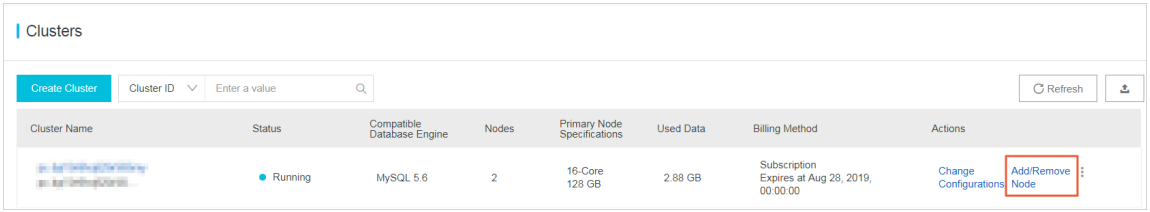
4. Select **Add Node** and click **OK**.



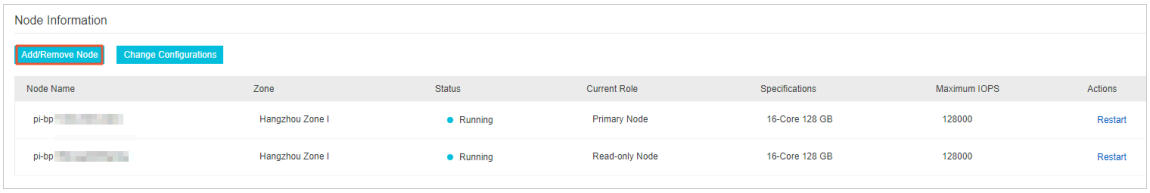
5. Click the  icon to add a read-only node. Read and agree to the service agreement by selecting the check box, and click **Pay** to complete the payment.

Remove a read-only node

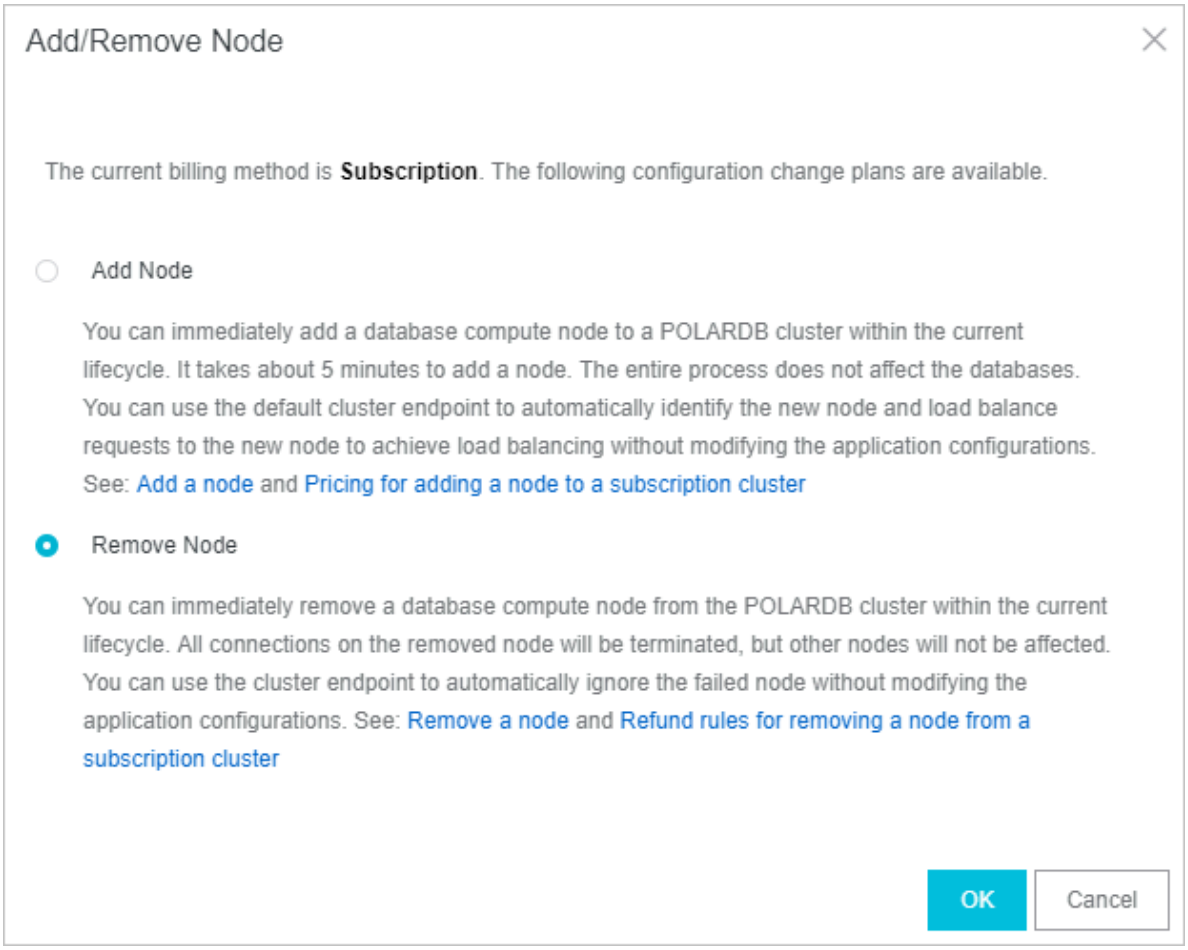
1. Log on to the ApsaraDB for POLARDB console.
2. Select a region.
3. Go to the **Add/Remove Node** page. Perform the operation by using either of the following methods:
   - Find the target cluster and click **Add/Remove Node** in the **Actions** column.



   - Find the target cluster, click the cluster ID, and then click **Add/Remove Node** in the Node Information section.



4. Select **Remove Node** and click **OK**.



5. Click the  icon next to the node that you want to remove. In the dialog box that appears, click **OK**.

> **Note:**
> You must keep at least one read-only node in the cluster to ensure high availability.

6. Read and agree to the service agreement by selecting the check box, and click **OK**.

Related API operations

API operation	Description
#unique_50	Adds a node to an ApsaraDB for PolarDB cluster.
#unique_49	Changes the specifications of a node in an ApsaraDB for PolarDB cluster.
#unique_51	Restarts a node in an ApsaraDB for PolarDB cluster.

API operation	Description
#unique_52	Removes a node from an ApsaraDB for PolarDB cluster.

9.6 Set the maintenance window

This topic describes how to set the maintenance window for an ApsaraDB for POLARDB cluster. To guarantee the stability of ApsaraDB for POLARDB, the backend system performs maintenance operations on the clusters from time to time. We recommend that you set the maintenance window within the off-peak hours of your business to minimize the impact on the business during the maintenance process.

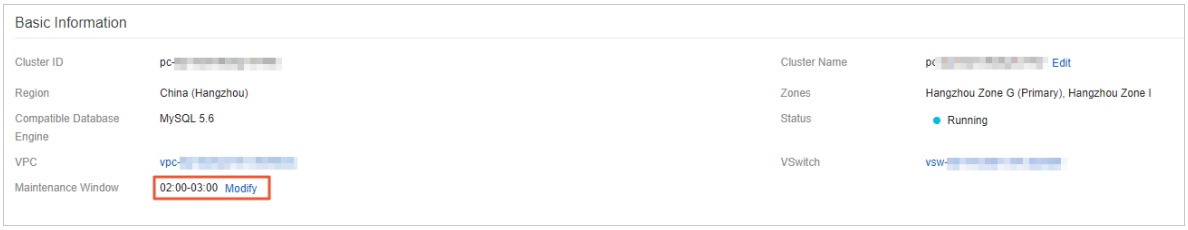
Important notes

- Before the maintenance is performed, ApsaraDB for POLARDB sends SMS messages and emails to contacts listed in your Alibaba Cloud account.
- To guarantee stability during the maintenance process, clusters first enter the **Under Maintenance** state before the preset maintenance window arrives on the day of maintenance. When a cluster is in this state, normal data access to the database is not affected. However, except for the account management, database management, and IP address whitelisting functions, other services concerning changes (such as common operations like upgrade, degrade, and restart) are unavailable in the console of this cluster. Query services such as performance monitoring are still available.
- Within the maintenance window of a cluster, the cluster may experience one or two disconnections. Make sure that your application can automatically reconnect to the cluster. The cluster restores to normal immediately after the disconnection occurs.

Procedure

1. Log on to the ApsaraDB for POLARDB console.
2. Select a region.
3. Find the target cluster and click the cluster ID in the **Cluster Name** column.

4. In the **Basic Information** section on the **Basics** page, click **Modify** next to **Maintenance Window**.



5. In the **Modify Maintenance Window** dialog box that appears, select a maintenance window for the cluster and click **Submit**.

APIs

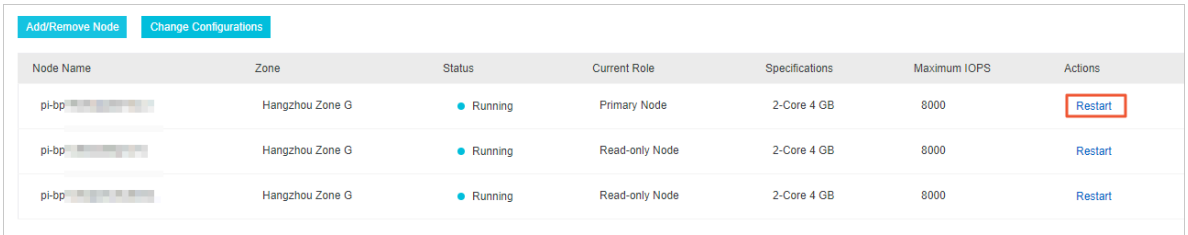
API	Description
CreateDBCluster	Creates an ApsaraDB for POLARDB cluster.
ModifyDBClusterMaintainTime	Modifies the maintenance window for an ApsaraDB for POLARDB cluster.

9.7 Restart a node

This topic describes how to manually restart a node when the number of connections exceeds the threshold or any performance issue occurs on the node. Restarting a node causes service interruptions. We recommend that you make appropriate service arrangements before you restart the nodes. Proceed with caution

Procedure

1. Log on to the ApsaraDB for PolarDB console.
2. Select a region.
3. Find the target cluster and click the cluster ID in the **Cluster Name** column.
4. In the **Node Information** section on the **Basics** page, find the node to be restarted.
5. Click **Restart** in the **Actions** column of the node.



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

Related API operations

API operation	Description
#unique_51	Restarts a database node.

9.8 Release a POLARDB cluster

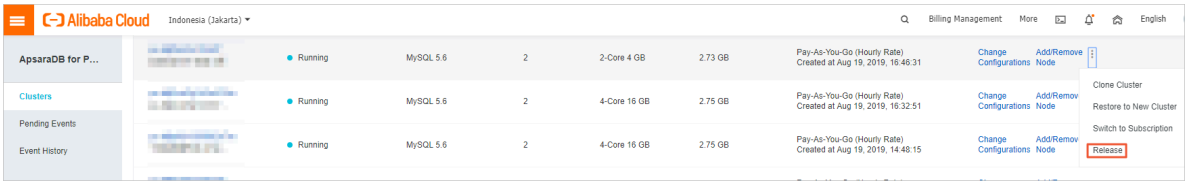
This topic describes how to manually release a pay-as-you-go POLARDB cluster according to your business requirements.

Precautions

- A subscription cluster cannot be manually released and will be automatically released once the subscription expires.
- A pay-as-you-go cluster can only be manually released when it is in the Running state.
- All the data in your cluster will be deleted when the cluster is released. Proceed with caution.
- This function is used to release a cluster, including all nodes in the specified cluster. To release one read-only node, see Add or remove a read-only node.
- You can switch the billing method of a cluster from pay-as-you-go to subscription. For more information, see Change the billing method from pay-as-you-go to subscription.

Procedure

1. Log on to the ApsaraDB for POLARDB console.
2. Select a region.
3. Find the target cluster. In the **Actions** column corresponding to the cluster, click **...** > **Release**.



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

APIs

API	Description
#unique_41	Views the list of POLARDB clusters.
#unique_57	Deletes a POLARDB cluster.

9.9 Switch workloads from writer nodes to reader nodes

An Apsara PolarDB cluster consists of one writer node and one or more reader nodes. This topic describes how to switch your workloads from a writer node to a reader node. If a failure occurs on a writer node, the system can automatically perform a failover. You may want to manually switch your workloads from the writer node to a reader node to run a disaster recovery drill and to specify a certain reader node as the writer node.

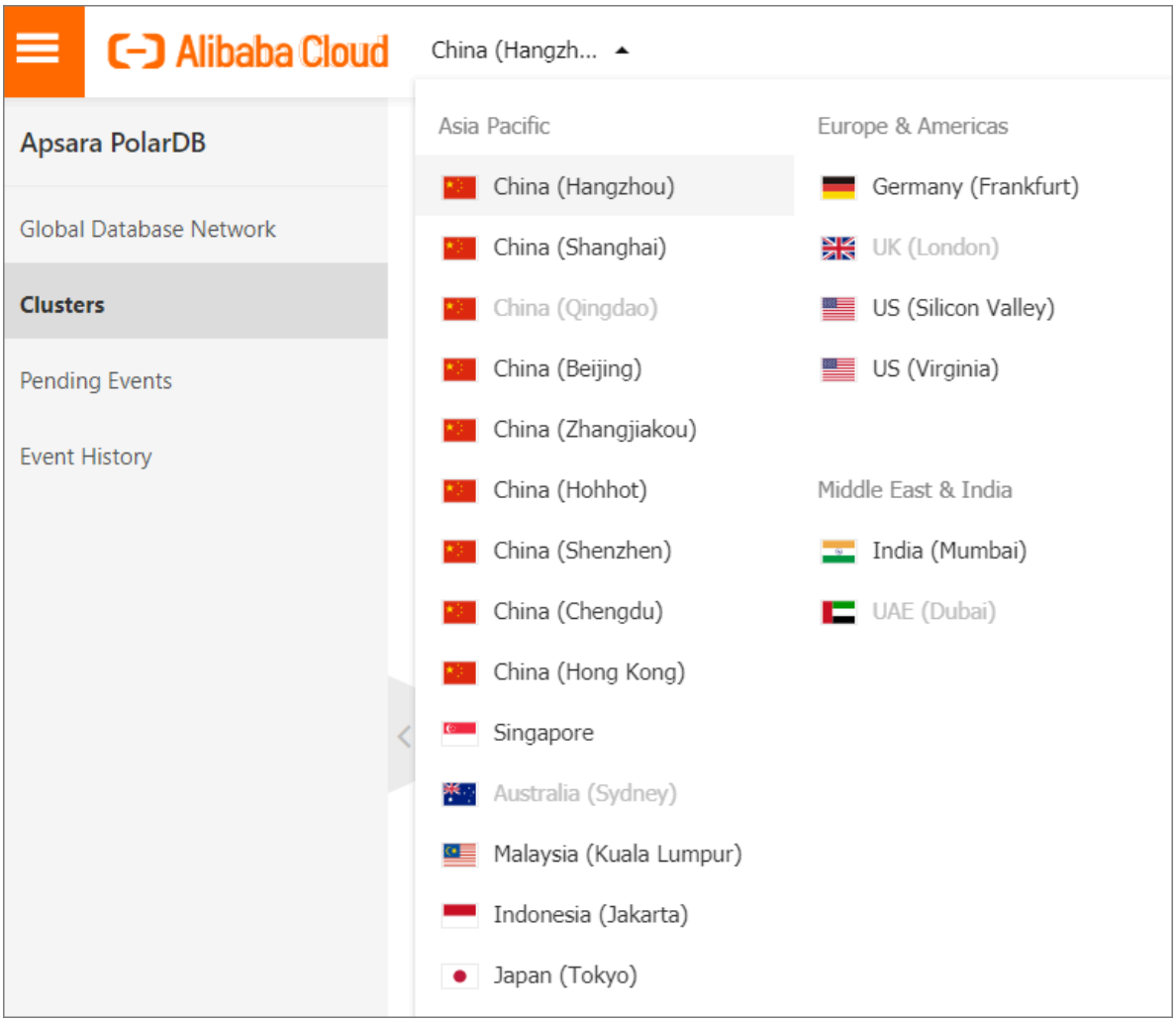
Considerations

An Apsara PolarDB cluster may be disconnected for approximately 30 seconds during switchover. We recommend that you perform the switchover during off-peak hours and make sure that your application can automatically reconnect to the Apsara PolarDB cluster.

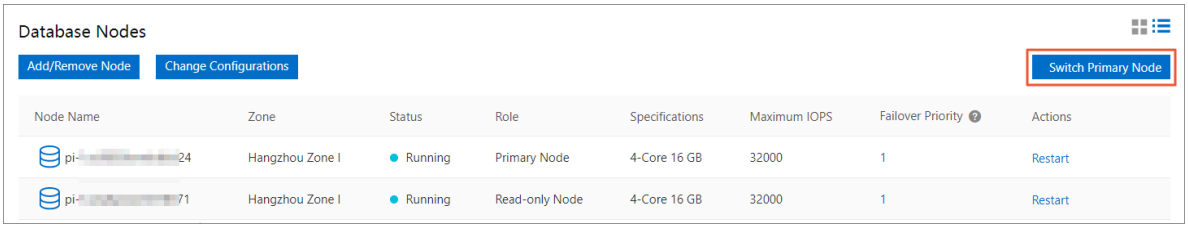
Manual switchover

1. Log on to the PolarDB console.

2. In the upper-left corner of the page, select the region where the PolarDB cluster is located.



3. Find the target cluster and click the cluster ID.
4. In the upper-right corner of the **Node Information** section, click **Switch Primary Node**.

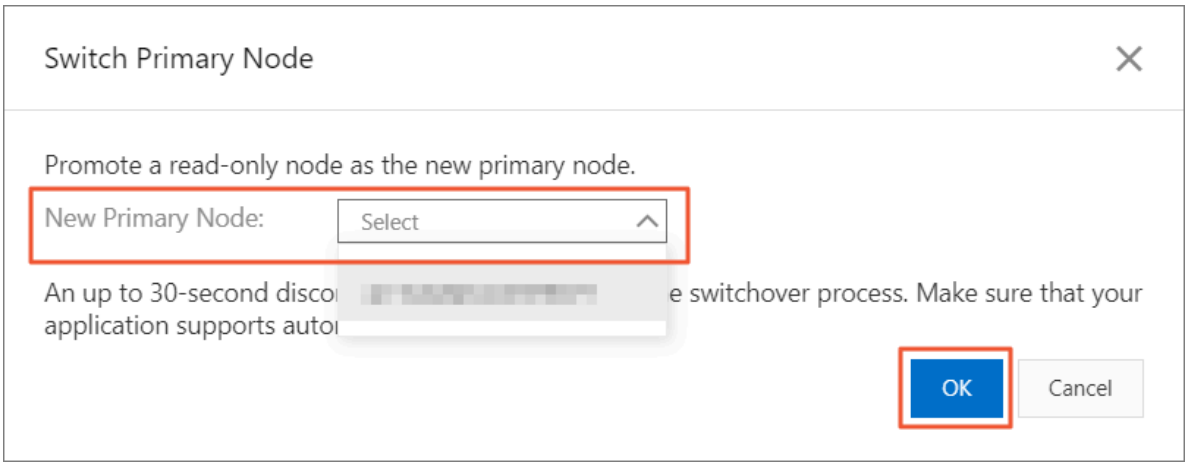


5. In the **Switch Primary Node** dialog box that appears, select a new writer node from the **New Primary Node** drop-down list, and click **OK**.

Note:

If you do not select a new writer node from the **New Primary Node** drop-down list, the system automatically promotes a reader node with the highest failover priority to the

new writer node. The cluster may be disconnected for approximately 30 seconds during switchover. Make sure that your application can automatically reconnect to the cluster.



Automatic failover between the writer node and reader nodes

An Apsara PolarDB cluster runs in an active-active high-availability architecture. This architecture allows for automatic failovers between the writer node and reader nodes.

Each node of the cluster has a failover priority. This priority determines the probability that the system promotes this node to the writer node if a failover occurs. If multiple nodes have the same failover priority, they all have the same probability of being promoted to the writer node.

The system follows these steps the to promote a reader node to the writer node:

1. Find all reader nodes that can be promoted to the writer node.
2. Select one or more reader nodes that have the highest failover priority.
3. If the failover to the first node fails due to network or replication errors, the system tries to switch the workloads to the next available node. The system continues the failover until one of the available nodes is promoted to the writer node.

Related API operations

Operation	Description
#unique_59	Manually switches your workloads from the writer node to a specified reader node in an Apsara PolarDB cluster.

9.10 Upgrade the minor version

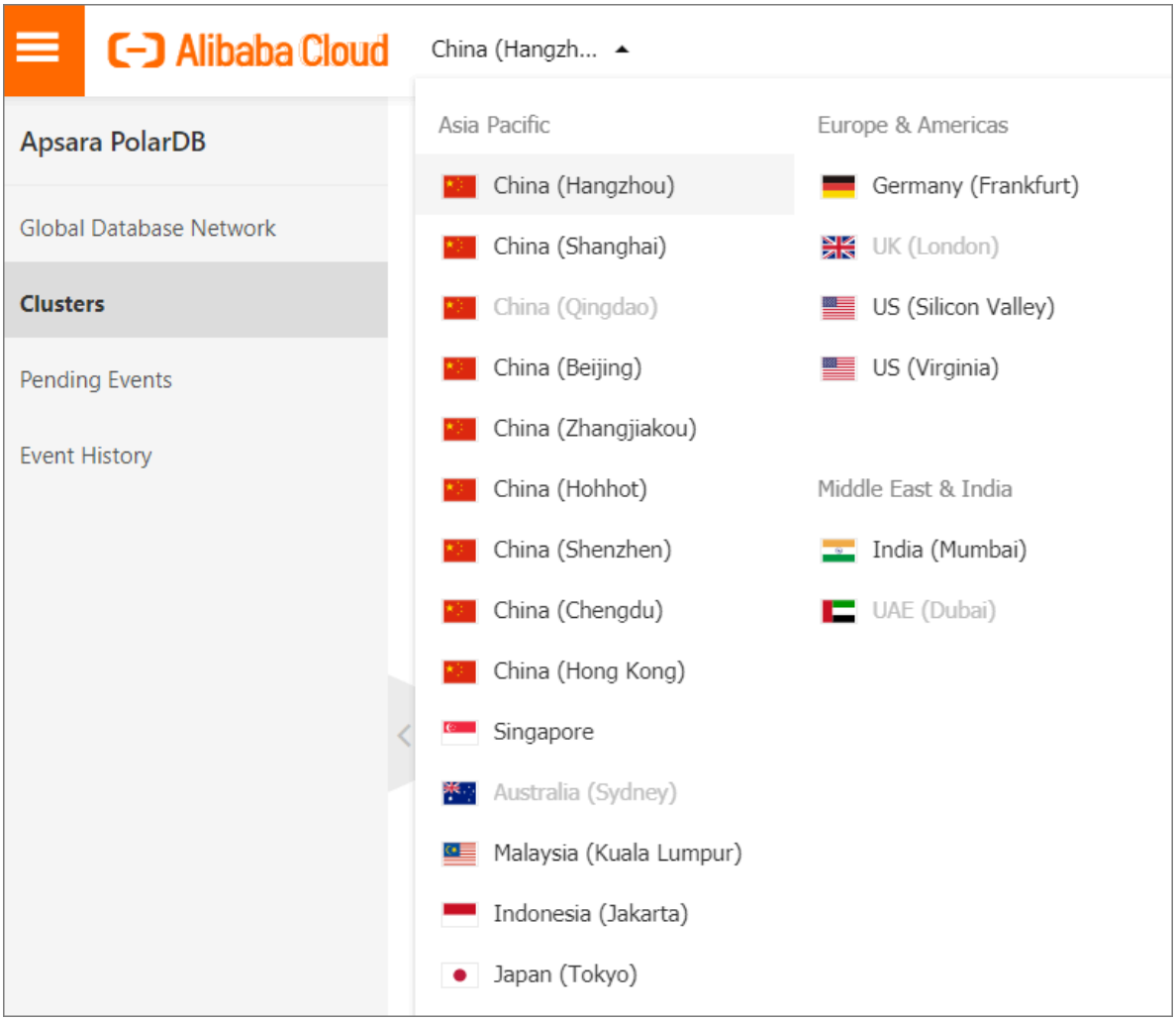
You can manually upgrade the minor kernel version of ApsaraDB for PolarDB cluster that is compatible with Oracle databases. The upgrades improve performance, provide new feature, or fix bugs.

Precautions

- Upgrading the kernel minor version will restart the instance. We recommend that you perform the upgrade during off-peak hours or make sure that your applications can automatically reconnect to the instance.
- You cannot downgrade the minor version after an upgrade.

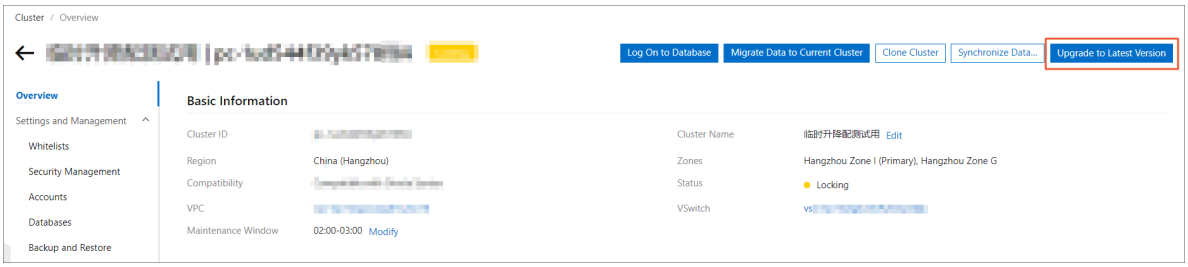
Procedure

1. Log on to the PolarDB console.
2. In the upper-left corner of the page, select the region where the PolarDB cluster is located.



3. Find the target cluster and click the cluster ID.
4. In , click .

> **Note:**
> If your cluster kernel version is already the latest, the button is not displayed.



5. In dialog box, click .

> **Note:**
> During the upgrade, services may be interrupted for about 60 seconds. Make sure that your applications can automatically reconnect to the instance.

10 Account management

10.1 Overview

Console accounts

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

- **Alibaba Cloud account**: This account allows flexible control of all your Alibaba Cloud resources and is used for billing purposes. You must create an Alibaba Cloud account before you can purchase any Alibaba Cloud services.
- **RAM user account**: optional. You can create and manage RAM user accounts in the Resource Access Management (RAM) console to share resources to multiple users. A RAM user account does not have ownership over any resources. Charges incurred are billed to the parent Alibaba Cloud account.

Database cluster accounts

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

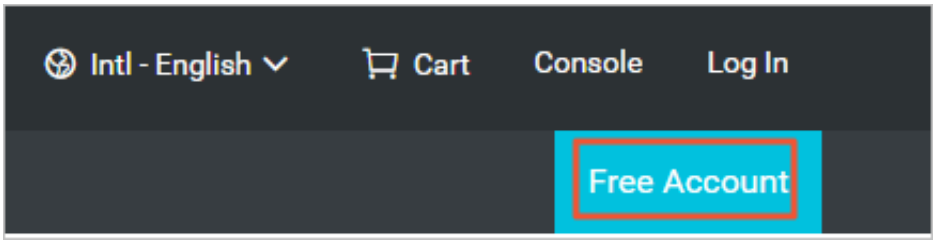
Account type	Description
Privileged account	• You can only create and manage the account in the console. • A cluster can have only one privileged account. A privileged account can manage all standard accounts and databases. • A privileged account has more permissions, which allows you to perform finer management operations. For example, you can grant query permissions on different tables to each user. • The account has all permissions on all databases in the cluster. • The account can disconnect any account from the cluster.

10.2 Register and log on to an Alibaba Cloud account

Register an Alibaba Cloud account

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

- On the Alibaba Cloud website, click **Free Account** in the upper-right corner.

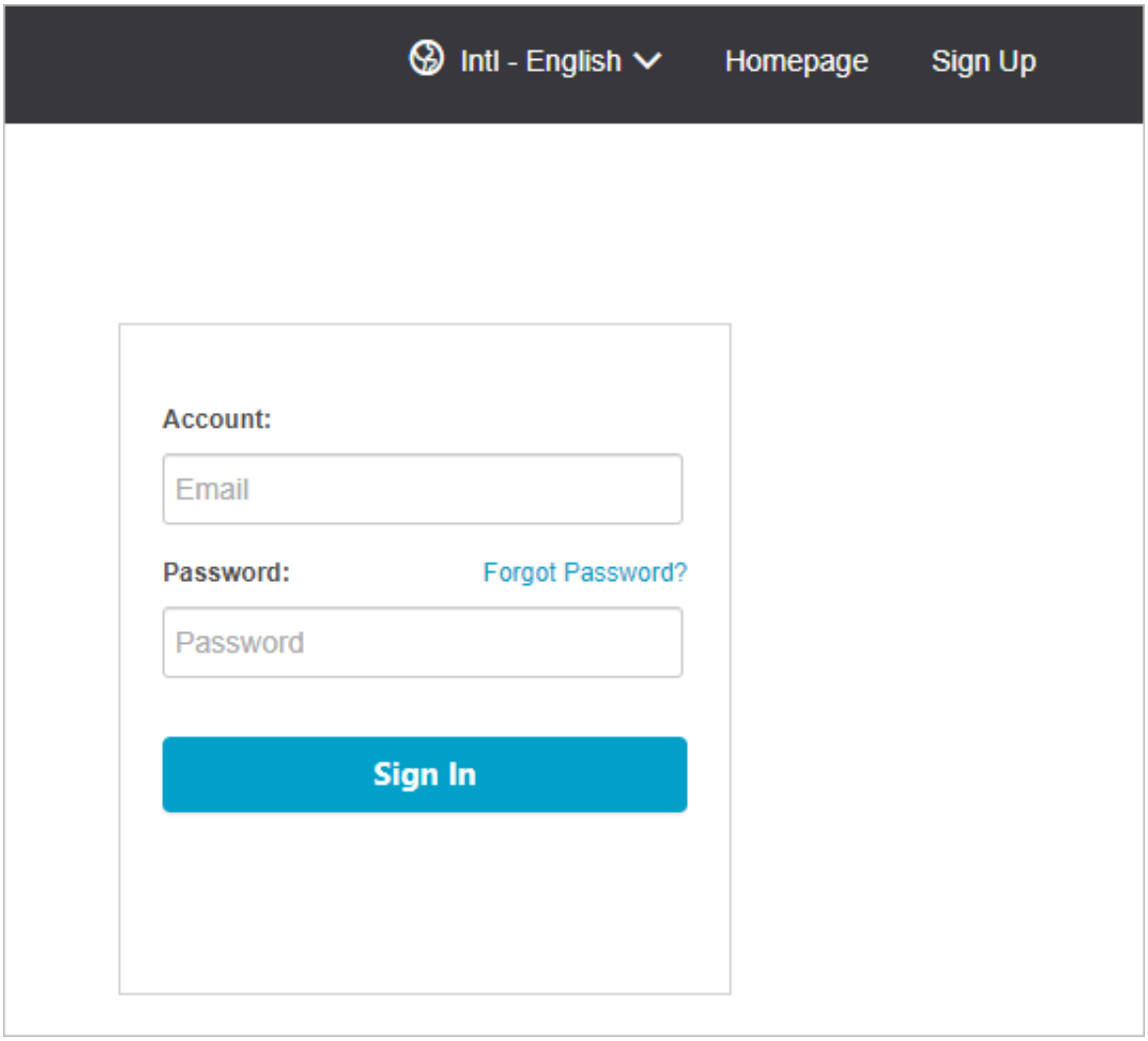


- Visit the Alibaba Cloud account registration page.

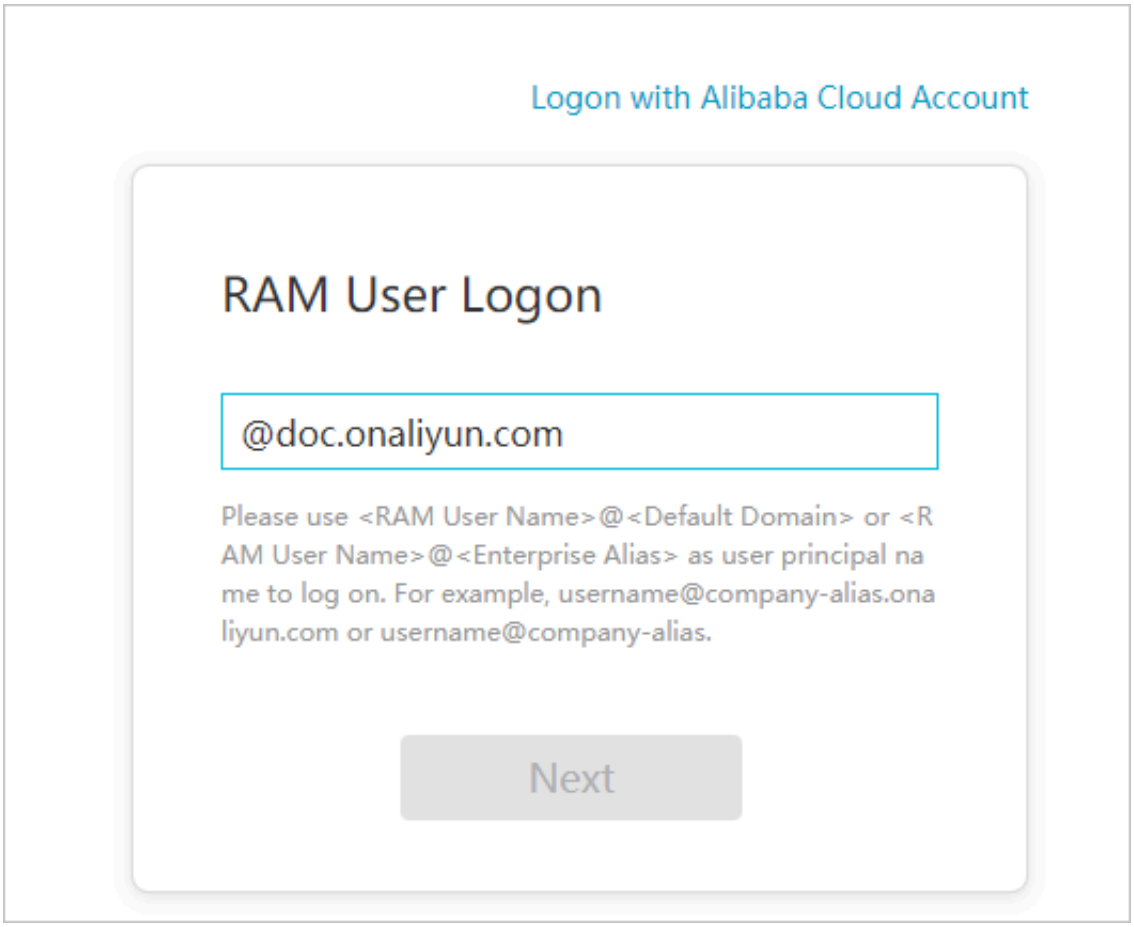
Log on to your Alibaba Cloud account.

Your Alibaba Cloud account and RAM user account have different logon pages.

- The logon page for Alibaba Cloud accounts is Alibaba Cloud accounts.



- The logon page for RAM users is RAM User Logon.



10.3 Create and authorize a RAM user

This topic describes how to create and authorize a Resource Access Management (RAM) user. You can use your Alibaba Cloud account to access your ApsaraDB for PolarDB resources. If you want to share the resources under your Alibaba Cloud account with other users, create and authorize a RAM user. The RAM user can then be used to access specified resources.

Create a RAM user

1. You can use an Alibaba Cloud account or a RAM user to create one or more RAM users. First, log on to the RAM console.
   - Click Alibaba Cloud account Logon to log on with your Alibaba Cloud account.
   - Click RAM User Logon to log on with your RAM user.

   Note:

Enter the RAM username in the format of RAM username@enterprise alias on the logon page.

2. In the left-side navigation pane, click **Users** under **Identities**.
3. Click **Create User**.

> **Note:**
> To create multiple RAM users at a time, click **Add User**.

4. Specify the **Logon Name** and **Display Name** parameters.
5. In the **Access Mode** section, select **Console Password Logon**.
6. Under **Console Password Logon**, select **Automatically Generate Default Password** or **Custom Logon Password**.
7. Under **Password Reset**, select **Required at Next Logon** or **Not Required**.
8. Under **Multi-factor Authentication**, select **Not Required**.
9. Click **OK**.

Grant permission to a RAM user on the Grants page

1. In the left-side navigation pane, click **Grants** under **Permissions**.
2. Click **Grant Permission**.
3. Under **Principal**, enter the username, and click the target RAM user.
4. In the **Policy Name** column, select the target policies by clicking the corresponding rows.

> **Note:**
> You can click **X** in the section on the right side of the page to delete the selected policy.

5. Click **OK**.
6. Click **Finished**.

Grant permission to a RAM user on the Users page

1. In the left-side navigation pane, click **Users** under **Identities**.
2. In the **User Logon Name/Display Name** column, find the target RAM user.
3. Click **Add Permissions**. On the page that appears, the principal is automatically filled in.
4. In the **Policy Name** column, select the target policies by clicking the corresponding rows.

> **Note:**
> You can click **X** in the section on the right side of the page to delete the selected policy.

5. Click **OK**.

6. Click **Finished**.

Log on as a RAM user

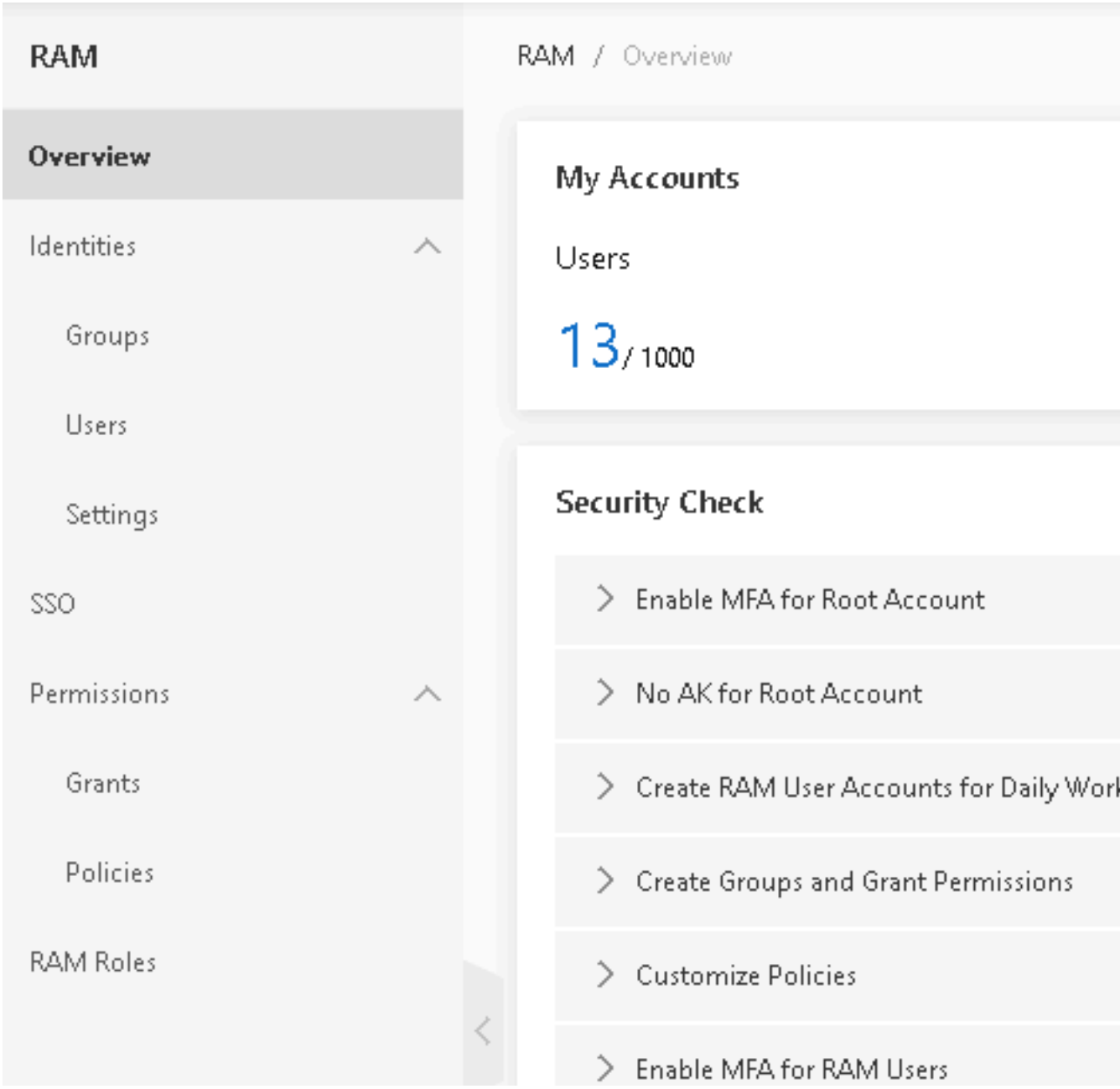
Prerequisites: You must complete the preceding authorization procedures.

You can log on as a RAM user at the following addresses:

- Universal logon address: RAM User Logon.

  If you log on at the universal logon address, you must enter the RAM username and company alias manually. The address format is RAM username@company alias.

- Dedicated logon address: You can view the logon address dedicated to your RAM users in the RAM console.



The system will enter your company alias automatically if you log on using this dedicated address. You only need to enter the RAM username.

More actions

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.

10.4 Create a database account

This topic describes how to create a database account and reset permissions of the account.

POLARDB compatible with Oracle supports only privileged accounts and allows you to manage the accounts in the console.

Account type	Description
Privileged account	• You can only create and manage the account in the console. • A cluster can have only one privileged account. A privileged account can manage all standard accounts and databases. • A privileged account has more permissions, which allows you to perform finer management operations. For example, you can grant query permissions on different tables to each user. • The account has all permissions on all databases in the cluster. • The account can disconnect any account from the cluster.

Create a privileged account

1. Log on to the ApsaraDB for POLARDB console.
2. Find the target cluster and click the cluster ID.
3. In the left-side navigation pane, click **Accounts**.
4. Click **Create Account**.
5. In the dialog box that appears, set the following parameters.

Parameter	Description
Account Name	Enter the account name. The requirements are as follows: • It must start with a lowercase letter and end with a letter or digit. • It can contain lowercase letters, digits, and underscores (_). • It must be 2 to 16 characters in length. • It cannot be a system reserved username, such as root and admin.
Account Type	Select **Privileged Account**.

Parameter	Description
Password	Enter the password of the privileged account. The requirements are as follows: • The password must contain at least three of the following character types: uppercase letters, lowercase letters, digits, and special characters. • The password must be 8 to 32 characters in length. • Special characters include ! @ # $ % ^ & * ( ) _ + - =
Confirm Password	Enter the password again.
Description	Enter information about the account to facilitate future use. The requirements are as follows: • It cannot start with http:// or https://. • The description must start with an uppercase or lowercase letter. • The description can contain uppercase or lowercase letters, digits, underscores (_), and hyphens (-). • The description must be 2 to 256 characters in length.

Next steps

View endpoints

Related API operations

API operation	Description
#unique_64	Creates a database account for a specified POLARDB cluster.
#unique_65	Queries the database accounts of a specified POLARDB cluster.
#unique_66	Modifies the description of a database account for a specified POLARDB cluster.
#unique_67	Changes the password of a database account for a specified POLARDB cluster.
#unique_68	Grants access permissions on one or more databases in a specified POLARDB cluster to a database account.
#unique_69	Revokes access permissions on one or more databases from a database account for a specified POLARDB cluster.

API operation	Description
#unique_70	Resets the permissions of a database account for a specified POLARDB cluster.

10.5 Manage a database account

A POLARDB cluster compatible with Oracle supports only privileged accounts and allows you to manage the accounts in the console.

Create a database account

For more information, see Create a database account.

Reset the password of a database account

1. Log on to the ApsaraDB for POLARDB console.
2. Find the target cluster and click the cluster ID.
3. In the left-side navigation pane, choose **Settings and Management** > **Accounts**.
4. Find the target account and click **Modify Password**.
5. In the dialog box that appears, enter a new password and click **Confirm**.

Related API operations

Operation	Description
#unique_72	Creates a database account
#unique_65	Queries the list of database accounts
#unique_66	Modifies the description of a database account
#unique_73	Changes the password of a database account
#unique_74	Deletes a database account

11 Database management

This topic provides an overview of database management, including how to create and delete databases.

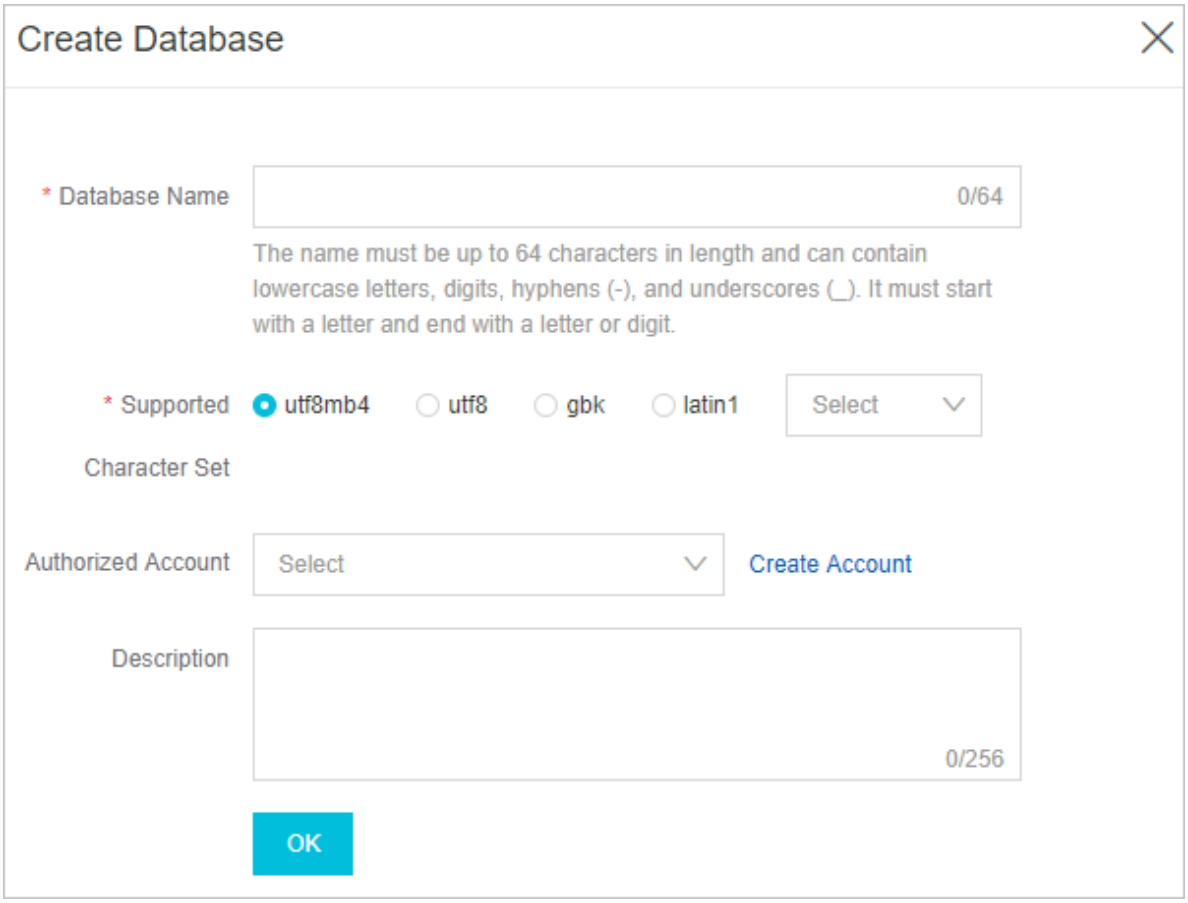
You can create and manage all databases in the ApsaraDB for PolarDB console.

Create a database

1. Log on to the ApsaraDB for PolarDB console.
2. Select a region.
3. Find the target cluster and click the cluster ID in the **Cluster Name** column.
4. In the left-side navigation pane, choose **Settings and Management** > **Databases**.
5. Click **Create Database**.
6. In the dialog box that appears, set parameters for creating a database. The following table describes the parameters.

Parameter	Description
Database Name	• It must start with a letter and end with a letter or digit. • It can contain lowercase letters, digits, underscores (_), and hyphens (-). • It must be 2 to 64 characters in length. • Each database name in an instance must be unique.
Supported Character Set	Select **utf8mb4**, **utf8**, **gbk**, or **latin1**. You can also select other required character sets from the drop-down list on the right.
Authorized Account	Select the account that you want to authorize for accessing this database. You can leave this parameter blank, and bind an account after the database is created. **Note:** Only **standard accounts** are available in the drop-down list. The privileged account has all the permissions on all databases. You do not need to authorize the privileged account to access the database that you create.
Account Permission	Select the permission that you want to grant to your account. Valid values: **Read and Write** | **Read Only**| **DML Only**.

Parameter	Description
Description	Enter the remarks of the database to facilitate subsequent database management. The requirements are as follows: • The description cannot start with http:// or https://. • The description must start with an uppercase or lowercase letter or a Chinese character. • The description can contain uppercase or lowercase letters, Chinese characters, digits, underscores (_), and hyphens (-). • The description must be 2 to 256 characters in length.



7. Click **OK**.

Delete a database

1. Log on to the ApsaraDB for PolarDB console.
2. Select a region.
3. Find the target cluster and click the cluster ID in the **Cluster Name** column.
4. In the left-side navigation pane, choose **Settings and Management** > **Databases**.
5. Find the target database and click **Delete** in the **Actions** column.
6. In the dialog box that appears, click **OK**.

Related API operations

API operation	Description
#unique_76	Creates a database.
#unique_77	Views the database list.
#unique_78	Modifies the description of a database.
#unique_79	Deletes a database.

12 Backup and restoration

12.1 Back up data

ApsaraDB for POLARDB uses a physical backup (snapshot backup), which is automatically performed once a day. You can also manually start a backup. Both the automatic backup and manual backup do not affect the normal running of the cluster. Backup files are retained for seven days.

Backup types

Backup type	Description
Automatic backup	• It is performed once a day by default. You can configure the time period and cycle for automatic backup. For more information, see Configure an automatic backup policy. • Backup files cannot be deleted.
Manual backup	• It can be started at any time. You can create a maximum of three manual backups for a cluster. For more information, see Manually create a backup. • Backup files can be deleted.

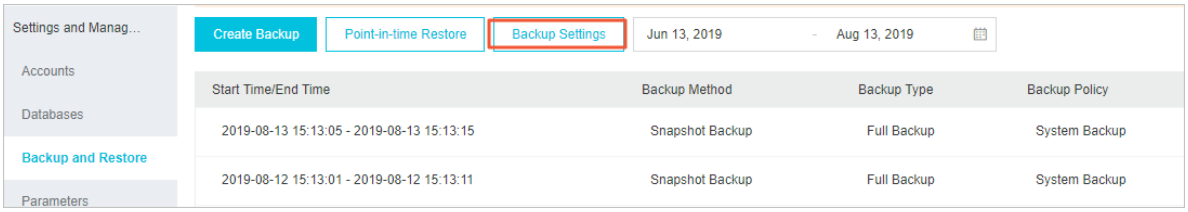
Pricing

The storage occupied by ApsaraDB for POLARDB backup files is free of charge.

Configure an automatic backup policy

1. Log on to the ApsaraDB for POLARDB console.
2. Select a region.
3. Find the target cluster and click the cluster ID in the **Cluster Name** column.
4. In the left-side navigation pane, choose **Settings and Management** > **Backup and Restore**.

5. Click **Backup Settings**.

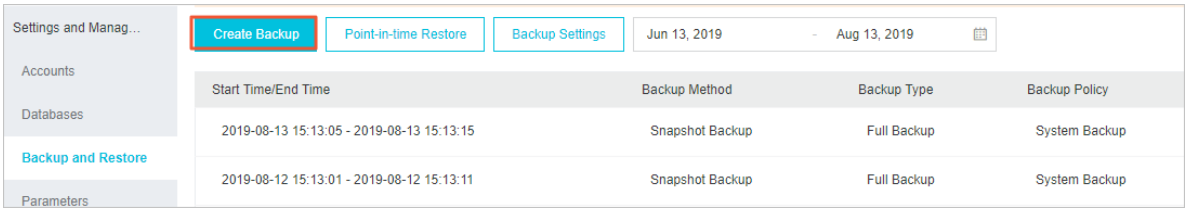


6. In the dialog box that appears, configure the time period and cycle for automatic backup.

> **Note:**
> For security reasons, automatic backup must be performed at least twice a week.

Manually create a backup

1. Log on to the ApsaraDB for POLARDB console.
2. Select a region.
3. Find the target cluster and click the cluster ID in the **Cluster Name** column.
4. In the left-side navigation pane, choose **Settings and Management** > **Backup and Restore**.
5. Click **Create Backup**.



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

> **Note:**
> You can create a maximum of three manual backups for a cluster.

Restore data

For more information, see Restore data.

API operations

Operation	Description
#unique_85	Creates a full snapshot backup for a specified ApsaraDB for POLARDB cluster.

Operation	Description
#unique_86	Queries the backup data of a specified ApsaraDB for POLARDB cluster.
#unique_87	Deletes the backup data of a specified ApsaraDB for POLARDB cluster.
#unique_88	Queries the automatic backup policy of a specified ApsaraDB for POLARDB cluster.
#unique_89	Modifies the automatic backup policy of a specified ApsaraDB for POLARDB cluster.

12.2 Restore data

A POLARDB cluster compatible with Oracle supports Restore by backup set (snapshot) to restore historical data to a new cluster.

Note:
The restored cluster contains the data and account information of the original cluster, but does not contain the parameter settings of the original cluster.

Restore data from a backup set (snapshot)

1. Log on to the ApsaraDB for POLARDB console.
2. Select the region where the cluster resides.
3. Find the target cluster and click the cluster ID.
4. In the left-side navigation pane, choose **Settings and Management** > **Backup and Restore**.
5. Find the target backup set (snapshot) and click **Restore** in the Actions column. In the dialog box that appears, click **OK**.
6. On the page that appears, select a billing method for the new cluster:
   - **Subscription**: An upfront payment must be made for each new compute cluster (contains a primary node and a read-only node by default). The storage occupied by the new cluster is billed on an hourly basis based on the actual data volume. The payment will be deducted from your Alibaba Cloud account on an hourly basis.

Subscription is more cost-effective for long term use. You can save more with longer subscription periods.

- **Pay-As-You-Go**: For the new cluster created, you do not need to pay any subscription fee for a compute cluster in advance. The compute cluster is billed on an hourly basis. The storage occupied by the new cluster is billed on an hourly basis based on the actual data volume. The payment will be deducted from your Alibaba Cloud account on an hourly basis. The **Pay-As-You-Go** method is most suitable for temporary applications, as you can release the cluster as soon as you do not need it anymore, saving costs.

7. Set the following parameters:
   - **Clone Source Type**: Select **Backup Set**.
   - **Clone Source Backup Set**: Confirm that the backup set is the one that you want to restore from.
   - **Region**: Use the default setting. It is the same as the region of the original cluster.
   - **Primary Availability Zone**: Use the default setting.
   - **Network Type**: Use the default setting.
   - **VPC** and **VSwitch**: The default settings are recommended, namely, the VPC and VSwitch of the original cluster.
   - **Database Engine**: Use the default setting.
   - **Node Specification**: Clusters with different specifications have different storage capacity and performance. For more information, see Node specifications.
   - **Number of Nodes**: Use the default setting. By default, the system will create a read-only node with the same specifications as the primary node.
   - **Cluster Name**: The system will automatically create a name for your cluster if you leave it blank. You can rename the cluster after it is created.
   - **Purchase Plan**: Set this parameter if you create a subscription cluster.
   - **Number**: The default value is 1, which cannot be modified.
8. Read the **ApsaraDB for POLARDB Agreement of Service**, select the check box to agree to it, and then complete the payment.

References

Back up data

API operations

Operation	Description
#unique_21	Creates an ApsaraDB for POLARDB cluster. **Note:** When you clone a cluster, set the value of **CreationOption** to **CloneFromPolarDB**.

13 Diagnostics and optimization

13.1 Performance monitoring and alert configuration

The ApsaraDB for POLARDB console provides a variety of performance metrics for you to monitor the status of your instances.

Performance monitoring

1. Log on to the ApsaraDB for POLARDB console.
2. In the upper-left corner of the console, select the region where the cluster is located.
3. Click the ID of the cluster.
4. In the left-side navigation pane, choose **Diagnostics and Optimization** > **Monitoring**.
5. You can view the performance information of a **Cluster** or **Node** as needed. For more information, see Metric description.
   - Cluster performance monitoring:

     Click the **Cluster** tab and set the monitoring time period. Click **OK**.
   - Node performance monitoring:

     Click the **Node** tab, select a node from the node list, and set the monitoring time period. Click **OK**.

Metric description

Category	Metric	Description
Cluster	Storage	Displays the usage of data space, log space, temporary space, and WAL log space.
	CPU	Displays the CPU utilization of each node.
	Memory	Displays the memory usage of each node.
Node	TPS	Displays the number of transactions per second of the selected node, including the number of committed transactions per second, deadlocked transactions per second, and rollback transactions per second.
	CPU	Displays the CPU utilization of the selected node.
	Memory	Displays the memory usage of the selected node.

Category	Metric	Description
	Connections	Displays the total number of current connections, active connections, and idle connections of the selected node.
	Scanned Rows	Displays the numbers of rows inserted, read, updated, deleted, and returned per second of the selected node.
	Maximum Database Age	Displays the difference between the transaction IDs of the oldest and newest transactions in the database.
	I/O Throughput	Displays the total I/O throughput, I/O read throughput, and I/O write throughput of the selected node.
	IOPS	Displays the input/output operations per second (IOPS) of the selected node, including the total IOPS, read IOPS, and write IOPS.
	Cache	Displays the cache reads per second and disk reads per second of the selected node.
	Cache Hit Ratio	Displays the cache hit ratio of the selected node.
	Temporary Files	Displays the number and total size of temporary files on the selected node.

Alert configuration

1. Log on to the CloudMonitor console.
2. In the left-side navigation pane, choose **Alerts** > **Alert Rules**.
3. On the **Alert Rules** page, click **Create Alert Rule** to go to the **Create Alert Rule** page.
4. Select **ApsaraDB for POLARDB PostgreSQL/Oracle** from the **Product** drop-down list and select a resource range from the **Resource Range** drop-down list. Set the alert rule and notification method, and click **Confirm**.

Note:

For more information about alert rules, see #unique_92.

13.2 Performance insight

ApsaraDB for POLARDB provides the performance insight feature, which focuses on monitoring the load, analyzing the load, and optimizing the performance of an ApsaraDB

for POLARDB cluster. The feature helps you easily evaluate database loads, find the causes for performance problems, and enhance database stability.

Scenarios

Performance insight can be applied in the following scenarios:

- Analyze the cluster metrics

  Performance insight helps you monitor the key metrics of an ApsaraDB for POLARDB cluster. It also allows you to check the status and trend of the loads for the cluster. You can identify the sources that generate loads and the distribution of loads within a certain period from the trend charts of key metrics.

- Evaluate database loads

  ApsaraDB for POLARDB provides the trend chart of average active sessions (AAS), which alleviates the need to analyze the complicated trend charts of various metrics. AAS trend chart shows the information of all key metrics to help you evaluate the sources that generate loads and cause performance bottlenecks. You can determine the causes for performance bottlenecks, such as high CPU usage, lock-waiting, and I/O latency, and find the corresponding SQL statement that incurs the problem.

  Note:
  AAS is the number of average active sessions of an ApsaraDB for POLARDB cluster within a certain period. The trends of AAS reflect the changes of the loads for the cluster. In the performance insight feature, AAS is a key metric used to measure the loads for an ApsaraDB for POLARDB cluster.

- Find the sources that cause performance problems

  You can analyze the trend chart of AAS and load sources in multiple dimensions to determine whether a performance problem is caused by improper cluster configurations or the database architecture. You can also find the corresponding SQL statement that incurs the performance problem.

Procedure

1. Log on to the ApsaraDB for POLARDB console.
2. Select a region.
3. Find the target cluster and click the cluster ID in the **Cluster Name** column.

4. In the left-side navigation pane, choose **Diagnostics and Optimization** > **Performance Insight**.
5. Select filtering conditions.

Description of the metrics page

- Trend charts of key metrics

  You can use the trend charts of key metrics to check the load status and resource bottlenecks of an ApsaraDB for POLARDB cluster.

  You can select a given time period or specify a custom time period to retrieve the trend charts of key metrics within the corresponding time period.

- Trend chart of AAS

  After you use the trend charts of key metrics to check the load status, you can identify the load sources.

  > **Note:**
  > max Vcores indicates the maximum number of CPU cores that can be used by an ApsaraDB for POLARDB cluster. The value determines the processing capacity of CPUs in the cluster.

  From the real-time trend chart of AAS, you can find the load sources, the time when loads occur, and the trend of loads over a period of time.

- Load sources from multiple dimensions

  You can learn the trend of the loads for an ApsaraDB for POLARDB cluster by analyzing the trend chart of AAS. You can find the specific SQL statements that cause performance bottlenecks, and the related users, hosts, and databases.

  As shown in the lower section of the preceding figure, you can find the SQL statements that affect the loads, and the usage ratio of each statement in a specified AAS.

  Performance insight supports six dimensions of AAS. You can switch dimensions by using the drop-down list of **AAS Type** in the upper-right corner of the AAS page.

Type	Description
SQL	The trends of top 10 SQL statements in your business.
Waits	The trends of wait events within the specified time period.
Users	The trends of logon users.

Type	Description
Hosts	The trends of hostnames or IP addresses of clients.
Databases	The trends of the databases where your businesses are located.
Status	The trends of active sessions within the specified time period.

14 Clone a cluster

This topic describes how to clone an ApsaraDB for PolarDB cluster. You can create an ApsaraDB for PolarDB cluster that is the same as an existing ApsaraDB for PolarDB cluster by cloning the data of the existing one. The data includes the account information, but excludes parameter settings of the cluster.

The data generated before the execution of the clone action is cloned. When cloning starts, the newly written data will not be cloned.

Procedure

1. Log on to the ApsaraDB for PolarDB console.
2. Select the region where the target cluster is located.
3. Find the cluster you want to clone. In the **Actions** column of the cluster, click the **More** icon, and then select **Restore to New Cluster**.
4. On the page that appears, set the parameters. The following table describes the parameters.

Parameter	Description
Clone Source Type	The type of the clone source. Select **Current Cluster**.
Region	The region where the cluster resides. The region of the new cluster is the same as that of the source cluster and cannot be modified.
Primary Availability Zone	• The zone of the new cluster. A zone is an independent physical area located within a region. There are no substantive differences between the zones. • You can deploy the ApsaraDB for PolarDB cluster and ECS instance in the same zone or in different zones.
Network Type	• The type of the network. Use the default setting. • ApsaraDB for PolarDB supports Virtual Private Cloud (VPC) networks only. A VPC is an isolated virtual network with higher security and performance than a classic network.

Parameter	Description
VPC Vswitch	The VPC and VSwitch of the new cluster. Select a VPC and a VSwitch from the corresponding drop-down lists, or create a VPC and a VSwitch. **Note:** Make sure that you place your ApsaraDB for PolarDB cluster and the ECS instance to be connected in the same VPC. Otherwise, they cannot intercommunicate through the internal network and achieve optimal performance.
Database Engine	The database engine of the new cluster. Use the default setting.
Node Specification	The node specification of the new cluster. Select a specification according to your needs. Clusters with different specifications have different storage capacity and performance. For more information, see #unique_39.
Number Nodes	The number of nodes in the new cluster. Use the default setting. By default, the system creates a read-only node with the same specification as the primary node.
Cluster Name	• Optional. The name of the new cluster. • The system will automatically create a name for your ApsaraDB for PolarDB cluster if you leave it blank. You can rename the cluster after it is created.
Purchase Plan	The subscription duration of the new cluster. This parameter is valid only for subscription clusters.
Number	The number of clusters. The default value 1 is used and cannot be modified.

5. Read the **ApsaraDB for PolarDB Agreement of Service**, select the check box to agree to it, and then complete the payment.

15 SQL firewall

This topic describes how to use the SQL/Protect plug-in to protect databases against SQL injection attacks.

Context

Developers are responsible for protecting databases against SQL injection attacks. They can use SQL/Protect to examine query requests and check for SQL injection. If suspicious query requests are identified, SQL/Protect immediately issues warning messages to database administrators while blocking the execution of the queries.

Types of SQL injection attacks

Attack type	What SQL/Protect can do
Unauthorized relations	While administrators can restrict access to relations (for example, tables and views), some of them do not perform this tedious task. SQL/Protect provides a learn mode that dynamically tracks the relations that a user accesses. This allows administrators to examine the workload of an application, and for SQL/Protect to learn which relations an application can be allowed to access for a given user or group of users in a role. When SQL/Protect is switched to the passive or active mode, the incoming queries are checked against the list of learned relations.
Utility commands	A common technique used in SQL injection attacks is to run utility commands, which are typically SQL Data Definition Language (DDL) statements. An example is creating a user-defined function that has the ability to access other system resources. SQL/Protect can block the running of all utility commands, which are not normally needed during standard application processing.
SQL tautology	The most frequent technique used in SQL injection attacks is issuing a tautological WHERE clause condition (that is, using a condition that is always true). The following is an example: `WHERE password = 'x' OR 'x'='x'`. Attackers usually start identifying security weaknesses by using this technique. SQL/Protect can block queries that use a tautological conditional clause.

Attack type	What SQL/Protect can do
Unbounded DML statements	A dangerous action taken during SQL injection attacks is the running of unbounded DML statements. These are UPDATE and DELETE statements with no WHERE clauses. For example, an attacker may update all users' passwords to a known value or initiate a denial of service (DoS) attack by deleting all of the data in a key table.

Protected roles

A protected role is a user or group that the database administrator has chosen to monitor by using SQL/Protect. Each protected role can be customized for the types of SQL injection attacks for which it is to be monitored, thus providing different levels of protection by role.

A role with the superuser privilege cannot be made a protected role. If a protected non-superuser role is subsequently altered to become a superuser, certain behaviors are exhibited whenever an attempt is made by that superuser to issue any command:

- A warning message is issued by SQL/Protect on every command issued by the protected superuser.
- When SQL/Protect is in active mode, all commands issued by the protected superuser are prevented from running.

A protected role that has the superuser privilege must either be altered so that it is no longer a superuser, or it must be reverted back to an unprotected role.

Additionally, every command issued by a protected role is recorded incrementally in a statistics view. This view helps to identify the start of a potential SQL injection attack against the role. The statistics are collected by type of SQL injection attack.

Note:
By default, each database supports up to 64 protected roles and up to 1024 protected tables. The maximum number of roles that can be protected is specified by the max_protected_roles parameter, and the maximum number of tables that can be protected is specified by the max_protected_relations parameter.

Configure SQL/Protect as an administrator to monitor a database

1. Edit the following parameters to start SQL/Protect:

```
edb_sql_protect.enabled = on # (This parameter is set to off by default.)
```

```
edb_sql_protect.level = passive # (SQL/Protect has three modes: learn, active, and passive. The default mode is passive.)
```

2. Execute the following statements to create a test database named targetdb and a test user named test:

```
CREATE DATABASE targetdb;
CREATE ROLE test;
GRANT ALL ON DATABASE targetdb TO test;
ALTER ROLE test LOGIN;
```

3. Log on to the test database targetdb. Then, execute the following statements to create SQL/Protect and add protected roles:

```
CREATE EXTENSION sqlprotect;
SELECT sqlprotect.protect_role('test');
```

Execute the following statements to display the list of protected roles:

```
SELECT * FROM sqlprotect.list_protected_users;
SELECT * FROM sqlprotect.edb_sql_protect;
```

4. Change the operating mode of SQL/Protect as needed.

SQL/Protect operates in three modes: learn, active, and passive. The default mode is passive. For more information, see Configure SQL/Protect operating mode.

- Switch SQL/Protect to the learn mode:

```
edb_sql_protect.level = learn; #
```

a. Log on to the targetdb database as the test user. Then, create a test table named company, and execute the SELECT and INSERT statements:

```
CREATE TABLE company(name VARCHAR(100), employee_num INT);
SELECT * FROM company;
INSERT INTO company VALUES('new', 1);
SELECT * FROM company;
```

b. Execute the following statements to display statistics about the test user accessing tables:

```
SELECT * FROM sqlprotect.edb_sql_protect_rel;
```

```
SELECT * FROM sqlprotect.list_protected_rels;
```

- Switch SQL/Protect to the passive mode:

```
edb_sql_protect.level = passive; #
```

a. Log on to the targetdb database as the test user.

b. Execute the following statements to inject SQL statements:

```
SELECT * FROM company WHERE 1 = 1;
DELETE FROM company;
```

Note:

SQL/Protect displays a message suggesting unauthorized SQL statements, but does not block the execution of the SQL statements.

- Switch SQL/Protect to the active mode:

```
edb_sql_protect.level = active; #
```

a. Log on to the targetdb database as the test user.

b. Execute the following statements to inject SQL statements:

```
SELECT * FROM company WHERE 1 = 1;
DELETE FROM company;
```

Note:

SQL/Protect displays a message suggesting unauthorized SQL statements, and blocks the execution of the SQL statements.

Configure protected roles

Protected roles are stored in the edb_sql_protect table. The database administrator can choose the users and user groups to be protected and add them to the table.

- Invoke the protect_role function to add a user to the table:

```
SELECT sqlprotect.protect_role('userA');
```

- Execute the following statements to display the protected roles in the table:

```
select * from sqlprotect.list_protected_users;
```

```
select * from sqlprotect.edb_sql_protect;
```

- Invoke the unprotect_role function to revert a protected role back to unprotected:

```
SELECT sqlprotect.unprotect_role('userA');
```

Configure SQL/Protect operating mode

The edb_sql_protect.level parameter specifies in which mode SQL/Protect operates to monitor a protected role. Three modes are available: learn, passive, and active. The default mode is passive.

Operating mode	Description
learn	SQL/Protect tracks the relations that a user accesses to learn which relations an application can be allowed to access for a given user or group of users in a role.
passive	SQL/Protect monitors every SQL statement that is to be executed. When a protected role attempts to execute an unauthorized SQL statement, SQL/Protect issues a warning message, but does not block the execution of the SQL statement.
active	SQL/Protect monitors every SQL statement that is to be executed, and blocks the execution of all unauthorized SQL statements from protected roles by using SQL firewalls when attackers start to perpetrate attacks. Additionally, SQL/Protect tracks the unauthorized SQL statements to help administrators identify database weaknesses ahead of attackers.

For example, if you want to switch SQL/Protect to the active mode, execute the following statement:

```
edb_sql_protect.level = active;  #
```

You can edit certain fields in the edb_sql_protect table to specify what need to be protected for a role.

```
targetdb=# \d sqlprotect.edb_sql_protect;
       Table "sqlprotect.edb_sql_protect"
     Column       |  Type   | Collation | Nullable | Default
------------------+---------+-----------+----------+---------
dbid              | oid     |           | not null |
roleid            | oid     |           | not null |
protect_relations | boolean |           |          |
allow_utility_cmds | boolean |          |          |
allow_tautology   | boolean |           |          |
allow_empty_dml   | boolean |           |          |
Indexes:
```

```
"edb_sql_protect_pkey" PRIMARY KEY, btree (roleid)
```

For example, if you have executed the following statement to set the **allow_utility_cmds** parameter to TRUE for a protected role named 16480, then SQL/Protect blocks the running of the utility commands issued from the protected role 16480:

```
UPDATE sqlprotect.edb_sql_protect SET allow_utility_cmds = TRUE WHERE roleid = 16480;
```

Other related operations

- To stop SQL/Protect, execute the following statements:

  ```
  edb_sql_protect.enabled = off
  edb_sql_protect.level = passive #
  ```

- To display statistics of SQL statements that are blocked by SQL/Protect, execute the following statement:

  ```
  SELECT * FROM sqlprotect.edb_sql_protect_stats;
  ```

- To delete statistics of SQL statements that are blocked by SQL/Protect for a specified user, execute the following statement:

  ```
  SELECT sqlprotect.drop_stats('username');
  ```

16 Supported extensions

This topic describes the extensions that are supported by Apsara PolarDB. Apsara PolarDB is compatible with Oracle.

Apsara PolarDB continuously updates its kernel to support new extensions or extension versions. To view the supported extensions, run the following statement:

```
show polar_supported_extensions;
```

Note:
If the kernel versions of clusters are earlier than 20191030, you can upgrade the kernels to the latest version by using the one-click upgrade feature. Then, you can use the supported extensions. The one-click upgrade feature is not available for general users. To use this feature, you must submit a ticket.

To view your kernel version, run the following statement:

```
show polar_release_date;
```

The following table describes the extensions that are supported by kernel version 20191030. The table also lists the latest extension versions.

Extension	Latest version	Description	Compatible extension	Latest version
appendchildxml	0.0.1	Appends a user-supplied value onto the target XML. The value functions as the child of the node that is indicated by an XPath expression.	N/A	N/A
btree_gin	1.3	Provides general-purpose Generalized Inverted Indexes (GINs).	N/A	N/A
btree_gist	1.5	Provides general-purpose Generalized Search Tree (GiST) indexes.	N/A	N/A
citext	1.5	Provides a case-insensitive character string type.	N/A	N/A

Extension	Latest version	Description	Compatible extension	Latest version
cube	1.4	Provides a data type that represents multidimensional cubes.	N/A	N/A
dict_int	1.0	Provides a dictionary template for full-text search. The dictionary template is designed for number indexing.	N/A	N/A
earthdistance	1.1	Provides two methods of calculating great circle distances on the surface of the Earth.	N/A	N/A
fuzzystrmatch	1.1	Provides functions that determine the similarities between strings.	N/A	N/A
hstore	1.5	Provides a data type that can be used to store sets of key-value pairs.	N/A	N/A
index_advisor	1.0	Provides index recommendations.	N/A	N/A
intagg	1.1	Provides an enumerator and integer aggregator.	N/A	N/A
intarray	1.2	Provides functions and operators that manipulate linear arrays. You can also perform indexed searches by using specific operators.	N/A	N/A
isn	1.2	Provides data types for the international product numbering standards.	N/A	N/A
ltree	1.1	Provides a data type for the data that is stored in a hierarchical tree-like structure.	N/A	N/A
pg_buffercache	1.3	Provides a method of checking the shared buffer cache in real time.	N/A	N/A

Extension	Latest version	Description	Compatible extension	Latest version
pg_prewarm	1.2	Provides an easy way to perform prewarming on table data.	N/A	N/A
pg_stat_statements	1.6	Provides a method of tracking the statistics of all SQL statements that have been executed.	N/A	N/A
pg_trgm	1.4	Provides functions and operators that determine the similarities of ASCII alphanumeric text. The similarities are determined based on trigram matching.	N/A	N/A
pg_wait_sampling	1.1	Collects the sampling data of wait events.	N/A	N/A
pgcrypto	1.3	Provides cryptographic functions.	N/A	N/A
pgrowlocks	1.2	Provides a function that returns row locking information for a specified table.	N/A	N/A
pgstattuple	1.5	Provides functions that return tuple-level statistical data.	N/A	N/A
plperl	1.0	Provides the PL/Perl procedural language that can be used to create functions.	N/A	N/A
plpgsql	1.0	Provides the PL/pgSQL procedural language that can be used to create functions.	N/A	N/A
pltcl	1.0	Provides the PL/Tcl procedural language that can be used to create functions.	N/A	N/A

Extension	Latest version	Description	Compatible extension	Latest version
polar_anydata	1.0	Provides the ANYDATA data type.	N/A	N/A
polar_gtt	1.0	Provides functions that can be used to manage global temporary tables.	N/A	N/A
sqlprotect	1.0	Provides the SQL firewall feature.	N/A	N/A
sslinfo	1.2	Provides the information about the SSL certificate that is provided by the current client.	N/A	N/A
tablefunc	1.0	Supports functions that return tables, such as the crosstab function.	N/A	N/A
unaccent	1.1	Provides a text search dictionary that removes diacritic signs from lexemes.	N/A	N/A
uuid-ossp	1.1	Provides functions that generate universally unique identifiers (UUIDs).	N/A	N/A
polar_cyclechecker_noticer	1.0	Reports an error if the result set of the CONNECT BY multi-clause contains cycle data.	N/A	N/A
ganos_geometry	2.3	Provides geography and geometry data types.	postgis	2.5.0
ganos_raster	2.3	Provides geographic raster data types.	postgis	2.5.0
ganos_geometry_sfcgal	2.3	Creates geometry 3D models.	postgis_sfcgal	2.5.0
ganos_geometry_topology	2.3	Creates a geometry topology.	postgis_topology	2.5.0
ganos_tiger_geocode	2.3	Provides the TIGER geocoding service.	postgis_tiger_geocode	2.5.0
ganos_address_standardizer	2.3	Provides the address standardization service.	address_standardizer	2.5.0

Extension	Latest version	Description	Compatible extension	Latest version
ganos_address_standardizer_data_us	2.3	Provides the address standardization service. You can use this extension to format and standardize American addresses.	address_standardizer_data_us	2.5.0
ganos_networking	2.3	Finds the optimal path in a geometric network diagram.	pgrouting	2.6.2
ganos_pointcloud	2.3	Provides data storage and computing services for the point cloud.	N/A	N/A
ganos_trajectory	2.3	Provides data types for moving objects.	N/A	N/A

If you want to use other extensions, submit a ticket.