

GaussDB(for MySQL)

Best Practices

Issue 01
Date 2023-02-28



Copyright © Huawei Technologies Co., Ltd. 2023. All rights reserved.

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

Trademarks and Permissions



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

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

Notice

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

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

Huawei Technologies Co., Ltd.

Address: Huawei Industrial Base
Bantian, Longgang
Shenzhen 518129
People's Republic of China

Website: <https://www.huawei.com>

Email: support@huawei.com

Contents

1 From ECS-hosted MySQL to GaussDB(for MySQL)	1
1.1 Overview.....	1
1.2 Resource Planning.....	3
1.3 Operation Guide.....	5
1.4 Procedure.....	6
1.4.1 ECS-hosted MySQL Server.....	6
1.4.1.1 Creating a VPC and Security Group.....	6
1.4.1.2 Creating an ECS (MySQL Server).....	9
1.4.1.3 Installing a MySQL Database (Community Edition).....	11
1.4.1.4 Creating an ECS and Installing the MySQL Client on It.....	14
1.4.2 Cloud Migration.....	14
1.4.2.1 Creating a GaussDB(for MySQL) Instance.....	14
1.4.2.2 Creating a DRS Migration Task.....	16
1.4.2.3 Checking the Migration Results.....	18
2 From Other Cloud MySQL to GaussDB(for MySQL)	21
2.1 Overview.....	21
2.2 Resource Planning.....	22
2.3 Operation Process.....	24
2.4 Creating a VPC and Security Group.....	24
2.5 Creating a GaussDB(for MySQL) Instance.....	27
2.6 Configuring a MySQL Instance on Other Clouds.....	28
2.7 Creating a DRS Migration Task.....	29
2.8 Checking Migration Results.....	31

1 From ECS-hosted MySQL to GaussDB(for MySQL)

1.1 Overview

This practice describes how to install a MySQL database (community edition) on a Huawei Cloud ECS and create a GaussDB(for MySQL) instance, and use DRS to migrate data from MySQL to GaussDB(for MySQL). With DRS, you can perform real-time migration tasks with minimal downtime. Services and databases remain operational during a migration.

Scenarios

- With the rapid increase of enterprise workloads, traditional databases have poor scalability and require distributed reconstruction.
- Building traditional databases requires purchasing and installing servers, systems, databases, and other software. Its O&M is expensive and difficult.
- Traditional databases are poor in complex queries.
- It is hard for traditional databases to smoothly migrate data without interrupting services.

Prerequisites

- You have completed real-name authentication.
- Your account balance is at least \$0 USD.

Solution Architecture

In this example, the source database is a MySQL instance on the ECS and the destination database is a GaussDB(for MySQL) instance. [Figure 1-1](#) shows the deployment architecture when the ECS-hosted MySQL and GaussDB(for MySQL) instances are in the same VPC.

If the ECS-hosted MySQL and GaussDB(for MySQL) instances are not in the same VPC, you need to configure a [VPC peering connection](#) between the two VPCs. For details about the deployment architecture, see [Figure 1-2](#).

Figure 1-1 Deployment architecture in the same VPC

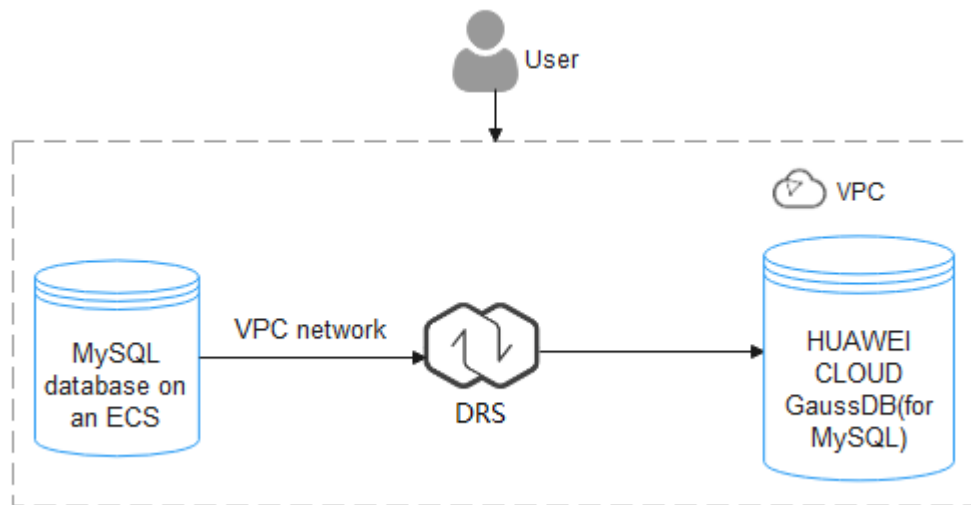
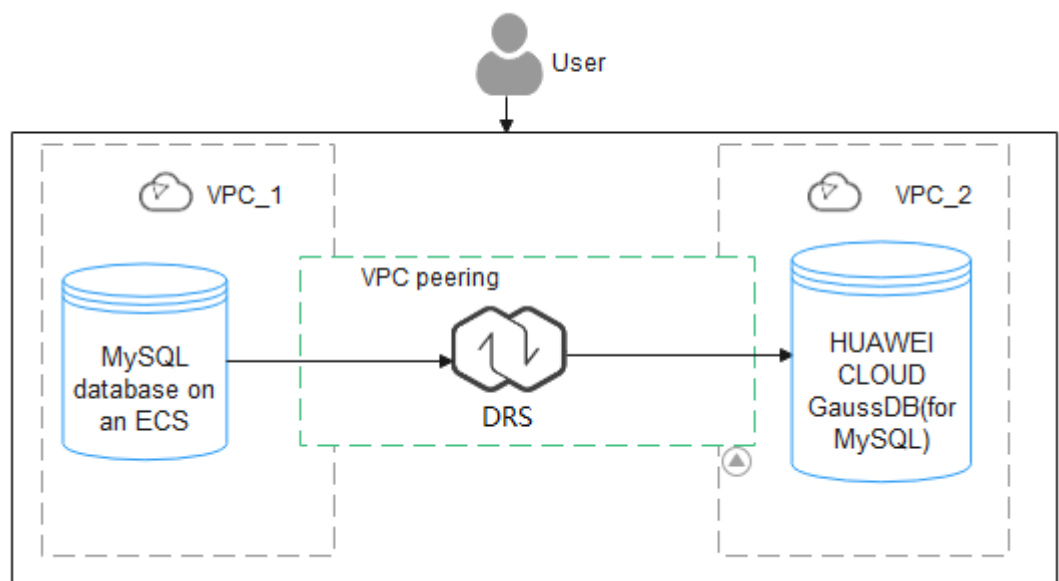


Figure 1-2 Deployment architecture in different VPCs



GaussDB(for MySQL) Advantages

- **Robust Performance:** GaussDB(for MySQL) decouples storage and compute and uses a "log as database" architecture and remote direct memory access (RDMA) to deliver seven times the performance of open-source MySQL.
- **Elastic Scaling:** In addition to a primary node, you can add up to 15 read replicas for a DB instance within minutes. You can also scale up or down CPU and memory specifications for a DB instance as needed.
- **High Reliability:** DB instances can be deployed across AZs and there are three data copies under the shared distributed storage layer. A DB instance failover can be complete within seconds with a zero RPO.

- High Security: With shared distributed storage, GaussDB(for MySQL) ensures zero data loss and service recovery within seconds. VPCs, security groups, SSL connections, and data encryption are used to strictly control secure access.
- High Compatibility: GaussDB(for MySQL) is fully compatible with MySQL. You can easily migrate your MySQL databases to GaussDB(for MySQL) without reconstructing existing applications and without sharding.
- Mass Storage: With distributed DFV storage, GaussDB(for MySQL) automatically grows storage as needed, up to 128 TB per DB instance.

Service List

- Virtual Private Cloud (VPC)
- Elastic Cloud Server (ECS)
- GaussDB(for MySQL)
- Data Replication Service (DRS)

Notes on Usage

The resources and test data in this practice are for demonstration only. Adjust them as needed.

For more information about GaussDB(for MySQL) data migration, see [From MySQL to GaussDB\(for MySQL\)](#).

1.2 Resource Planning

Table 1-1 Resource planning

Category	Subcategory	Planned Value	Remarks
VPC	VPC name	vpc-mysql	Customize a name for easy identification.
	Region	EU-Dublin	Currently, only Europe-Dublin is supported.
	AZ	AZ3	-
	Subnet	10.0.0.0/24	Select a subnet with sufficient network resources.
	Subnet name	subnet-mysql	Customize a name for easy identification.
ECS (MySQL server)	ECS name	ecs-mysql	Customize a name for easy identification.
	Specifications	s6.xlarge.2 4 vCPUs 8 GiB	Select specification based on service requirements. For details, see x86 ECS Specifications and Types
	OS	CentOS 7.6 64	-

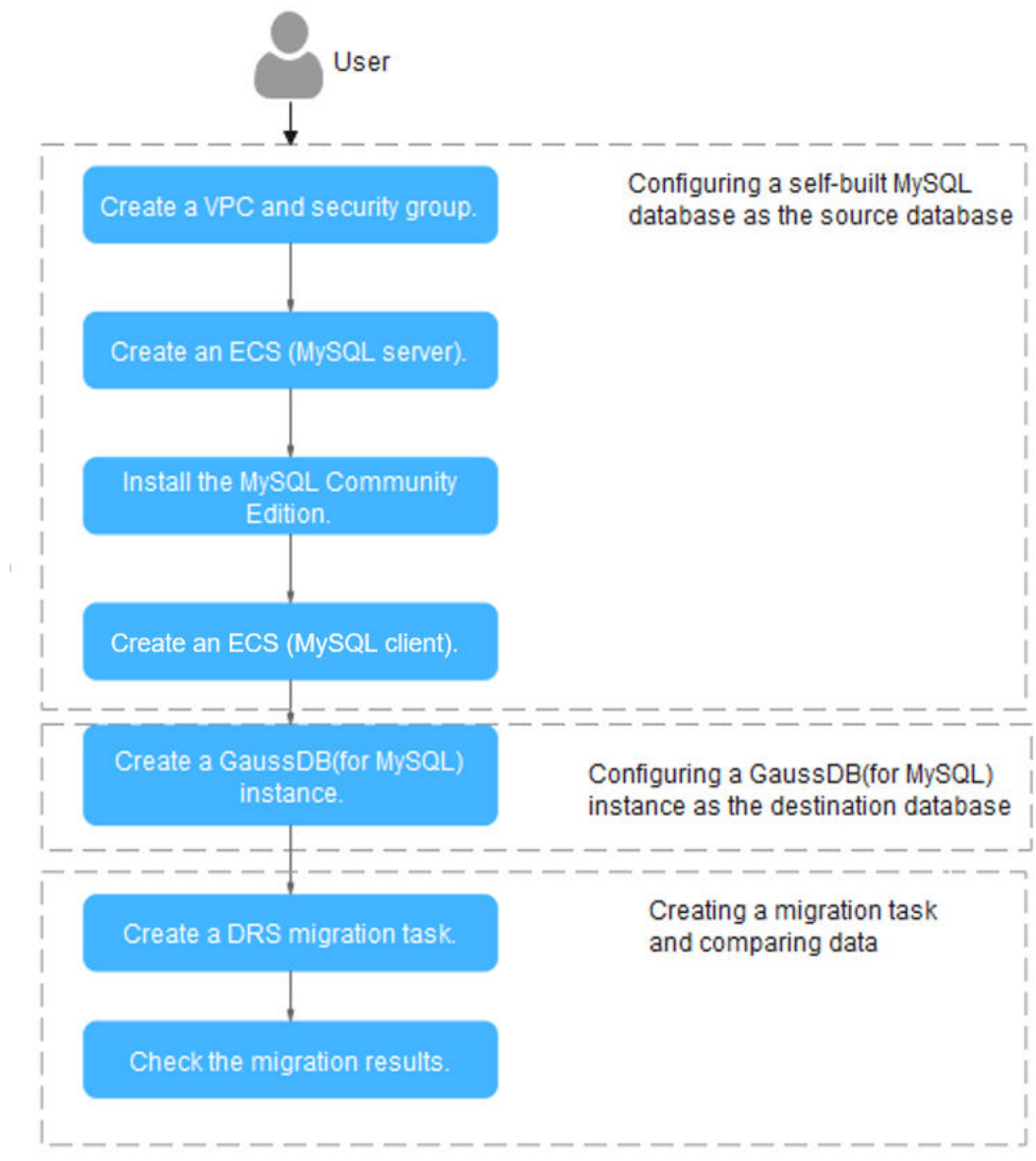
Category	Subcategory	Planned Value	Remarks
	System disk	General purpose SSD 40 GiB	-
	Data disk	Ultra-high I/O, 100 GiB	-
	EIP	Auto assign	Buy an EIP because the public network is selected for the migration task.
ECS (MySQL client)	ECS name	ecs-client	Customize a name for easy identification.
	Specifications	s6.xlarge.2 4 vCPUs 8 GiB	Select specification based on service requirements. For details, see x86 ECS Specifications and Types .
	OS	CentOS 7.6 64	-
	System disk	General purpose SSD 40 GiB	-
	Data disk	Not required	-
	EIP	Auto assign	Buy an EIP as needed. If you do not need to access the client through a public network, you do not buy an EIP.
GaussDB(f or MySQL)	Instance name	gauss-mysql	Customize a name for easy identification.
	DB engine	GaussDB(for MySQL)	-
	DB engine version	MySQL 8.0	-
	AZ type	Single-AZ	-
	AZ	AZ6	-
	Instance specifications	Dedicated Edition	-
	CPU architecture	x86 8 vCPUs 32 GB	-
DRS migration task	Task name	DRS-gaussdbformysql	Customize a name for easy identification.
	Source DB engine	MySQL	In this example, take a MySQL instance (community edition) installed on an ECS as the source database.
	Destination DB engine	GaussDB(for MySQL)	In this example, take a GaussDB(for MySQL) instance as the destination database.

Category	Subcategory	Planned Value	Remarks
	Network type	Public	In this example, select the public network.

1.3 Operation Guide

Figure 1-3 shows the process of creating a MySQL server, buying a GaussDB(for MySQL) instance, and migrating data from the MySQL server to the GaussDB(for MySQL) instance.

Figure 1-3 Flowchart



1.4 Procedure

1.4.1 ECS-hosted MySQL Server

This chapter describes how to build a MySQL server. For example, buy an ECS and install a MySQL instance on it. The tasks involved are as follows:


1. Creating a VPC and security group
2. Creating an ECS (MySQL server)
3. Installing a MySQL database (community edition)
4. Creating an ECS (MySQL client)

1.4.1.1 Creating a VPC and Security Group

This section describes how to create a VPC and security group for your MySQL server and GaussDB(for MySQL) instance.

Creating a VPC

Step 1 Log in to the [management console](#).

Step 2 Click  in the upper left corner of the management console and select **EU-Dublin**.

Step 3 Under the service list, choose **Network > Virtual Private Cloud**.

The VPC console is displayed.



Step 4 Click **Create VPC**.

Basic Information

Region

Regions are geographic areas isolated from each other. Resources are region-specific and can latency and quick resource access, select the nearest region.

Name

IPv4 CIDR Block · · · /

Recommended: 10.0.0.0/8-24 (Select) 172.16.0.0/12-24 (Select) 192.168.0.0/16-24 (Select)

Enterprise Project [Create Enterprise Project](#) ?

Advanced Settings ▾ Tag | Description

Default Subnet

AZ ?

Name

IPv4 CIDR Block · · · / ? Available IP Addresses: 251

The CIDR block cannot be modified after the subnet has been created.

IPv6 CIDR Block Enable ?

Associated Route Table ?

Advanced Settings ▾ Gateway | DNS Server Address | NTP Server Address | DHCP Lease Time | Tag | Description

Step 5 Configure parameters as needed and click **Create Now**.


Step 6 Return to the VPC list and check whether the VPC is created.

If the VPC status becomes available, the VPC has been created.

----End

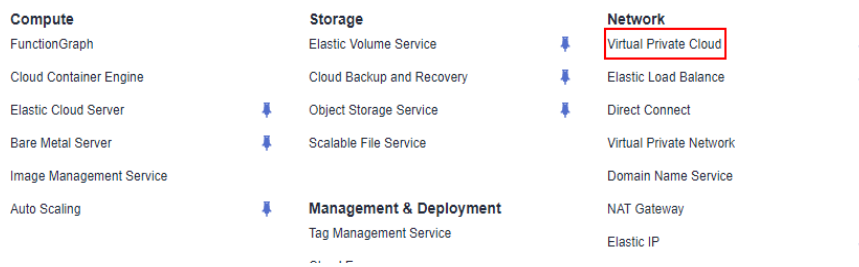
Creating a Security Group

Step 1 Log in to the [management console](#).

Step 2 Click  in the upper left corner of the management console and select **EU-Dublin**.

Step 3 Under the service list, choose **Network > Virtual Private Cloud**.

The VPC console is displayed.



Step 4 In the navigation pane on the left, choose **Access Control > Security Groups**.

Step 5 Click **Create Security Group**.

Step 6 Configure parameters as needed.

Create Security Group

* Name

* Enterprise Project [Create Enterprise Project](#) ?

* Template

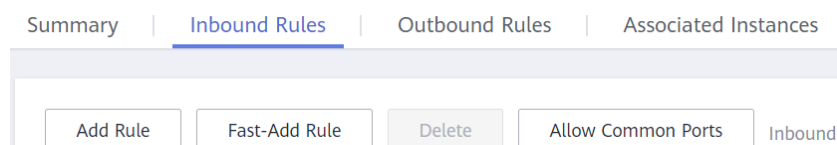
Description
0/255

[Show Default Rule](#) ▼

Step 7 Click **OK**.

Step 8 Return to the security group list and click the security group name **sg-mysql**.

Step 9 Click the **Inbound Rules** tab, and then click **Add Rule**.



Step 10 Configure an inbound rule to allow access from database port **3306**.

Add Inbound Rule [Learn more](#) about security group configuration.

i Inbound rules allow incoming traffic to instances associated with the security group.

Security Group sg-mysql
You can import multiple rules in a batch.

Priority ?	Action	Protocol & Port ?	Type	Source ?	Description	Operation
1-100	Allow	TCP 3306	IPv4	IP address 0.0.0.0/0		Operation

⊕ Add Rule

OK Cancel

Step 11 Perform [Step 9](#) to [Step 10](#) to allow access from database port **22**.

After the rules were configured, the figure similar to the following is displayed.


<input type="checkbox"/>	Priority...	Action ?	Protocol & Port ?	Type
<input type="checkbox"/>	1	Allow	TCP : 3306	IPv4
<input type="checkbox"/>	1	Allow	TCP : 22	IPv4

----End

1.4.1.2 Creating an ECS (MySQL Server)

This section describes how to buy an ECS for installing a MySQL database (community edition).

Step 1 Log in to the [management console](#).

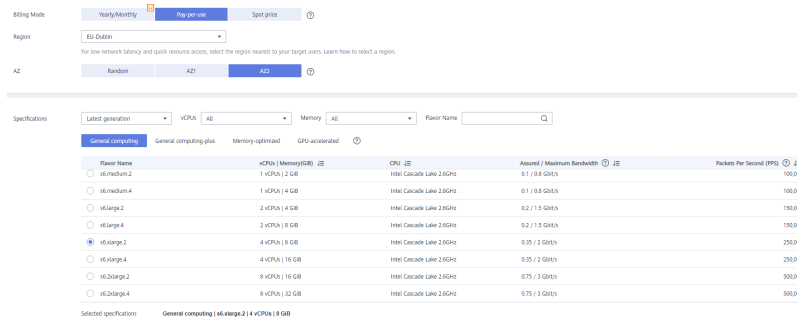
Step 2 Click  in the upper left corner of the management console and select **EU-Dublin**.

Step 3 Under the service list, choose **Compute > Elastic Cloud Server**.

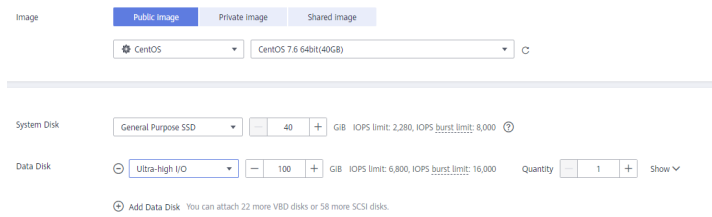
Step 4 Click **Buy ECS**.

Step 5 Configure ECS parameters.

1. Set **Specifications** to **General computing** and select **s6.xlarge.2** with 4 vCPUs and 8 GiB.

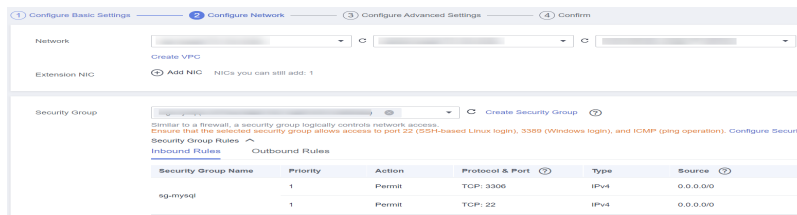


2. Select the image and disk specifications.

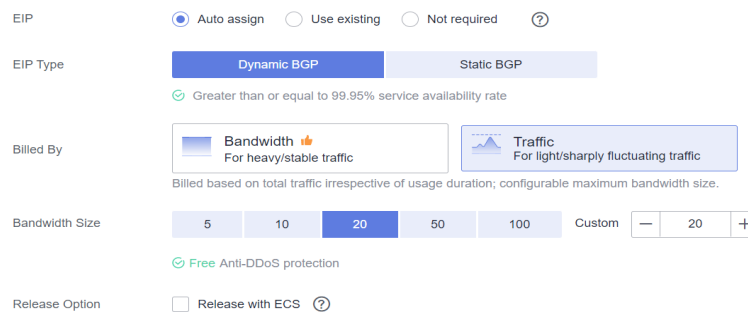


Step 6 Click **Next: Configure Network**.

1. Select the VPC and security group created in [Creating a VPC and Security Group](#).



2. Set **EIP to Auto assign**, **Billed by to Traffic**, and **Bandwidth Size to 20**. The bandwidth size can be changed as required.



Step 7 Click **Next: Configure Advanced Settings**.

Specify **ECS Name** and **Password**.

ECS Name: Allow duplicate name
If you are creating multiple ECSs at the same time, automatic naming and customizable naming are supported.

Description:

Login Mode: **Password** | Key pair | Set password later

Username: root

Password: Keep the password secure. If you forget the password, you can log in to the ECS console as a root user.

Confirm Password:

Step 8 Click **Next: Confirm**.

Step 9 Select an enterprise project and click **Submit**.

Step 10 Return to the **Elastic Cloud Server** page and view progress.

When the ECS status changes to **Running**, the ECS has been created.


----End

1.4.1.3 Installing a MySQL Database (Community Edition)

This section describes how to initialize disks and install a MySQL database (community edition).

Log In to the ECS

Step 1 Log in to the [management console](#).

Step 2 Click  in the upper left corner of the management console and select **EU-Dublin**.

Step 3 Under the service list, choose **Compute > Elastic Cloud Server**.

Step 4 Locate the ECS **ecs-mysql** and click **Remote Login** in the **Operation** column.

Step 5 Select **CloudShell-based Login**.

Logging In to a Linux ECS

 The security group of this ECS does not allow Ports21, Ports80, Ports3389, PortsICMP login to the ECS will fail. [Add security group rule](#) [View port functions](#)

CloudShell-based Login **New!**

[Failed to log in?](#)

Ensure that the port used for CloudShell-based logins (port 22 by default) is allowed to access in the security group.

CloudShell-based login allows you to copy-paste data across local and remote PCs.

Log In

Step 6 Enter the password of user **root**.

NOTE

The password is the one you specified during the ECS creation.

----End

Initializing Disks

Step 1 Create the **mysql** folder.

```
mkdir /mysql
```

Step 2 View data disk information.

```
fdisk -l
```

The command output is as follows.

```
[root@ecs-mysql ~]# fdisk -l

Disk /dev/vda: 42.9 GB, 42949672960 bytes, 83886080 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk label type: dos
Disk identifier: 0x000e3a31

   Device Boot      Start         End      Blocks   Id  System
 /dev/vda1    *          2048     83886079     41942016   83  Linux

Disk /dev/vdb: 107.4 GB, 107374182400 bytes, 209715200 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
```

Step 3 Initialize the data disk.

```
mkfs.ext4 /dev/vdb
```

Step 4 Attach the disk.

```
mount /dev/vdb /mysql
```

Step 5 Check whether the disk is attached.

```
df -h
```

If the following output is returned, the disk is attached.

```
[root@ecs-mysql ~]# df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        3.9G   0  3.9G   0% /dev
tmpfs           3.9G   0  3.9G   0% /dev/shm
tmpfs           3.9G  8.6M  3.9G   1% /run
tmpfs           3.9G   0  3.9G   0% /sys/fs/cgroup
/dev/vda1       40G  2.2G  36G   6% /
tmpfs           783M   0  783M   0% /run/user/0
/dev/vdb        99G   61M  94G   1% /mysql
```

Step 6 Create a folder and switch to the **install** folder.

```
mkdir -p /mysql/install/data
```

```
mkdir -p /mysql/install/tmp
```

```
mkdir -p /mysql/install/file
```

```
mkdir -p /mysql/install/log
```

```
cd /mysql/install
```

Step 7 Download and install [How Can I Install the MySQL Client?](#)

Step 8 Initialize the MySQL client.

```
/mysql/install/mysql-8.0.22/bin/mysqld --defaults-file= /etc/my.cnf --
initialize-insecure
```

Step 9 Start the MySQL client.

```
nohup /mysql/install/mysql-8.0.22/bin/mysqld --defaults-file= /etc/my.cnf &
```

Step 10 Connect to the MySQL client.

```
/mysql/install/mysql-8.0.22/bin/mysql
```

Step 11 Create user **root** and assign the required permissions to it.

```
grant all privileges on *.* to 'root'@'%' identified by 'xxx' with grant
option;FLUSH PRIVILEGES;
```

```
----End
```


1.4.1.4 Creating an ECS and Installing the MySQL Client on It

Step 1 This section describes how to create an ECS for the MySQL client. For details, see [Creating an ECS \(MySQL Server\)](#).

 **NOTE**

- This ECS must be in the same region, AZ, VPC, and security group as the ECS where the MySQL server is deployed.
- Data disks are not required.
- This ECS name is **ecs-client**.
- Other parameters are the same as those of the ECS where the MySQL server is deployed.

Step 2 Download and install the MySQL client. For details, see [How Can I Install the MySQL Client?](#)

----End


1.4.2 Cloud Migration

This chapter describes how to create a GaussDB(for MySQL) instance, create a DRS migration task, and migrate data from the ECS-hosted MySQL server to the GaussDB(for MySQL) instance.

1.4.2.1 Creating a GaussDB(for MySQL) Instance

This section describes how to create a GaussDB(for MySQL) instance. The GaussDB(for MySQL) instance must be in the same VPC and security group as the ECS-hosted MySQL server.

Step 1 Log in to the [management console](#).

Step 2 Click  in the upper left corner of the management console and select **EU-Dublin**.

Step 3 Under the service list, choose **Database > GaussDB(for MySQL)**.

Step 4 In the upper right corner, click **Buy DB Instance**.

Step 5 Configure the instance name and basic information.

Billing Mode Yearly/Monthly Pay-per-use

Region ▼

Regions are geographic areas isolated from each other. Resources are r resource access, select the nearest region.

DB Instance Name ?

If you buy multiple DB instances at a time, they will be named with fo be named as instance-0001, the second as instance-0002, and so on.

DB Engine GaussDB(for MySQL)

DB Engine Version MySQL 8.0

AZ Type Single-AZ

AZ az6 az1 az3

Time Zone UTC+08:00 Beijing, Chongqing, Hong Kong,... ▼

Step 6 Configure instance specifications.

Instance Specifications Dedicated Edition

CPU Architecture x86 ?

vCPU | Memory

4 vCPUs | 16 GB

8 vCPUs | 32 GB

16 vCPUs | 64 GB

32 vCPUs | 128 GB

32 vCPUs | 256 GB

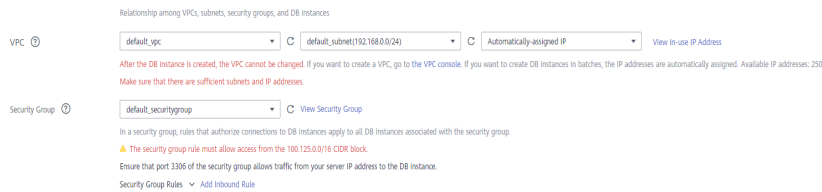
60 vCPUs | 256 GB

DB Instance Specifications Dedicated Edition

Read Replicas - + ?

Step 7 Select the VPC and security group.

The VPC and security group have been created in [Creating a VPC and Security Group](#).



Step 8 Configure the instance password.

Administrator

Administrator Password Keep your password secure. The system cannot retrieve your password.

Confirm Password

Step 9 Click **Next**.

Step 10 After confirming the settings, click **Submit**.

Step 11 Return to the instance list.


If the instance becomes **Available**, the instance has been created.

----End

1.4.2.2 Creating a DRS Migration Task

This section describes how to create a DRS migration task to migrate the **sbtest** database from the ECS-hosted MySQL server to the GaussDB(for MySQL) instance.

Step 1 Log in to the [management console](#).

Step 2 Click  in the upper left corner of the management console and select **EU-Dublin**.

Step 3 Under the service list and choose **Database > Data Replication Service**.

Step 4 In the upper right corner, click **Create Migration Task**.

Step 5 Configure parameters as needed.

1. Specify a task name.

Region

Regions are geographic areas isolated from each other. Resource access, select the nearest region.

Project

* Task Name

Description

0/256

2. Configure replication instance details as needed.
Set **Destination DB Instance** to the GaussDB(for MySQL) instance created in [Creating a GaussDB\(for MySQL\) Instance](#).

Replication Instance Details ⓘ

The following information cannot be modified after you go to the next page.

* Data Flow	<input checked="" type="radio"/> To the cloud	<input type="radio"/> Out of the cloud	
The destination database must be a database in the current cloud. If you want to migrate			
* Source DB Engine	<input checked="" type="radio"/> MySQL	<input type="radio"/> MySQL schema and logic table	<input type="radio"/> MongoDB
* Destination DB Engine	<input type="radio"/> MySQL	<input type="radio"/> DDM	<input checked="" type="radio"/> GaussDB(for MySQL) Primary/Standby Ed...
* Network Type	VPC ⓘ		
* Destination DB Instance	gauss-f693 (172.16.0.169)	View DB Instance	View I
* Replication Instance Subnet	subnet-mysql(172.16.0.0/24)	View Subnets	
* Migration Type	<input checked="" type="radio"/> Full+Incremental	<input type="radio"/> Full	This migration type allows you to migrate data with minimal downtime . After a full migratio between the source and destination databases.
* Destination DB Instance Access	<input checked="" type="radio"/> Read-only	<input type="radio"/> Read/Write	

3. Set **Enterprise Project** to **default**.

* Enterprise Project [View Project Management](#)

Tags

It is recommended that you use TMS's predefined tag function to add the same tag to different

You can add 10 more tags.

Step 6 Click **Create Now**.

It takes about 5 to 10 minutes to create a replication instance.

Step 7 Configure source and destination database information.

1. Configure source database information.
2. Click **Test Connection**.

If a successful connection message is returned, you have logged in to the source database.

Source Database

DRS migrates only some key parameters to the destination database. For the other parameters that can

Source Database Type	<input checked="" type="radio"/> self-built database	<input type="radio"/> RDS DB instance
VPC	vpc-mysql(172.16.0.0/24) ▼	
Subnet	subnet-mysql(172.16.0.0/24) ▼	
IP Address or Domain Name	172.16.0.35	
Port	3306	
Database Username	root	
Database Password	Info@123 <input type="checkbox"/>	

3. Configure the username and password for the destination database.
4. Click **Test Connection**.
If a successful connection message is returned, you have logged in to the destination database.

Destination Database

DB Instance Name	gauss-f693 (172.16.0.169)
Database Username	<input type="text" value="root"/>
Database Password	<input type="password" value="....."/>
Migrate Definer to User	<input checked="" type="radio"/> Yes ?

Step 8 Click **Next**.

Step 9 Confirm the users, snapshots, and migration objects to be migrated.
Set **Migrate Object** to **All**.

Step 10 Click **Next**.

Step 11 View pre-check results.

Step 12 If the check is complete and the check success rate is 100%, click **Next**.

Step 13 Click **Submit**.

Return to the **Online Migration Management** page and check the migration task status.

It takes several minutes to complete.

Task Name/ID	Status	Delay	Charging	Data Flow	DB Engine	Migration Type	Netwo...	Operation
DRS-test-migrate	Star...	--	No	To the cloud	MySQL-GaussDB...	Full	Public...	Stop

If the status changes to **Completed**, the migration task has been created.

----End


1.4.2.3 Checking the Migration Results

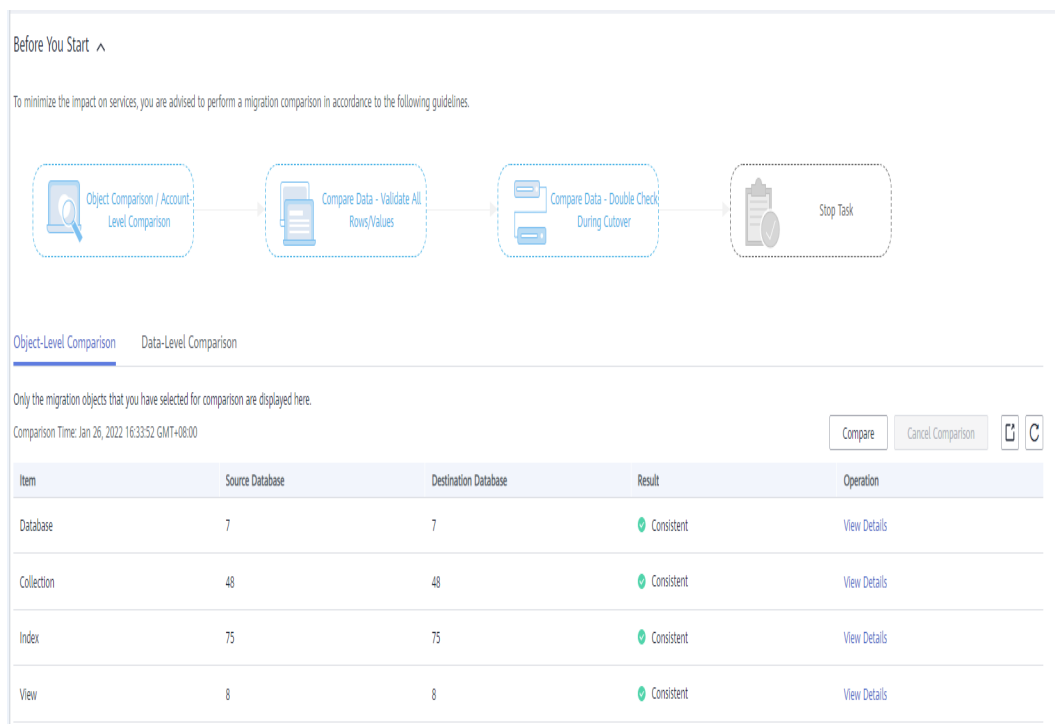
You can check migration results with either of the following methods:

Method 1: (Automatically) [View the migration results on the DRS console](#). DRS can compare migration objects, users, and data of source and destination databases and obtain the migration results.

Method 2 (manually): [View the migration results on the GaussDB\(for MySQL\) console](#). Log in to the destination database to check whether the databases, tables, and data are migrated. Manually confirm the data migration status.

Viewing Migration Results on the DRS Console

- Step 1** Log in to the [management console](#).
- Step 2** Click  in the upper left corner of the management console and select **EU-Dublin**.
- Step 3** Under the service list and choose **Database > Data Replication Service**.
- Step 4** Click the target DRS instance name.
- Step 5** Click **Migration Comparison**.



Before You Start ^

To minimize the impact on services, you are advised to perform a migration comparison in accordance to the following guidelines.

Object Comparison / Account-Level Comparison → Compare Data - Validate ALL Rows/Values → Compare Data - Double Check During Cutover → Stop Task

Object-Level Comparison | Data-Level Comparison

Only the migration objects that you have selected for comparison are displayed here.


Comparison Time: Jan 26, 2022 16:33:52 GMT+08:00

Item	Source Database	Destination Database	Result	Operation
Database	7	7	Consistent	View Details
Collection	48	48	Consistent	View Details
Index	75	75	Consistent	View Details
View	8	8	Consistent	View Details

- Step 6** Under **Compare Data - Validate ALL Rows/Values** and **Compare Data - Double Check During Cutover** tab to check whether the objects of the source database have been migrated to destination database.

----End

Viewing Migration Results on the GaussDB(for MySQL) Console

- Step 1** Log in to the [management console](#).
- Step 2** Click  in the upper left corner of the management console and select **EU-Dublin**.
- Step 3** Under the service list, choose **Database > GaussDB(for MySQL)**.
- Step 4** Click **GaussDB(for MySQL)**. Locate the target instance and choose **More > Log In** in the **Operation** column.
- Step 5** In the displayed dialog box, enter the password and click **Test Connection**.

Step 6 After the connection is successful, click **Log In**.

Step 7 Enter the password to log in to the GaussDB(for MySQL) instance.

Step 8 Check and confirm the destination database name and table name. Check whether the data migration is complete.

----End

Testing GaussDB(for MySQL) Performance

After the migration is complete, test GaussDB(for MySQL) performance by referring to [Performance White Paper](#).

2 From Other Cloud MySQL to GaussDB(for MySQL)

2.1 Overview

Description

This section includes the following content:

- Create a GaussDB(for MySQL) instance.
- Migrate data from MySQL on other clouds to GaussDB(for MySQL).

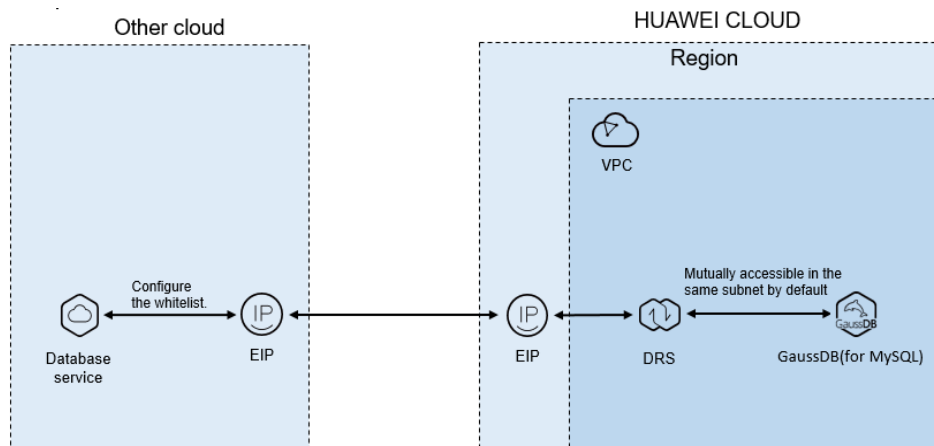
Prerequisites

- You have registered with Huawei Cloud.
- Your account balance is greater than or equal to \$0 USD.

Deployment Architecture

In this example, the source is a MySQL database on other cloud platforms and the destination is a Huawei Cloud GaussDB(for MySQL) instance. Data is migrated from the source to the destination over a public network. For details about the deployment architecture, see [Figure 2-1](#).

Figure 2-1 Deployment architecture



Service List

- Virtual Private Cloud (VPC)
- GaussDB(for MySQL)
- Data Replication Service (DRS)

Before You Start

- The resource planning in this best practice is for demonstration only. Adjust it as needed.
- All settings in this best practice are for reference only. For more information about MySQL migration, see [From MySQL to GaussDB\(for MySQL\) Primary/Standby](#).

2.2 Resource Planning

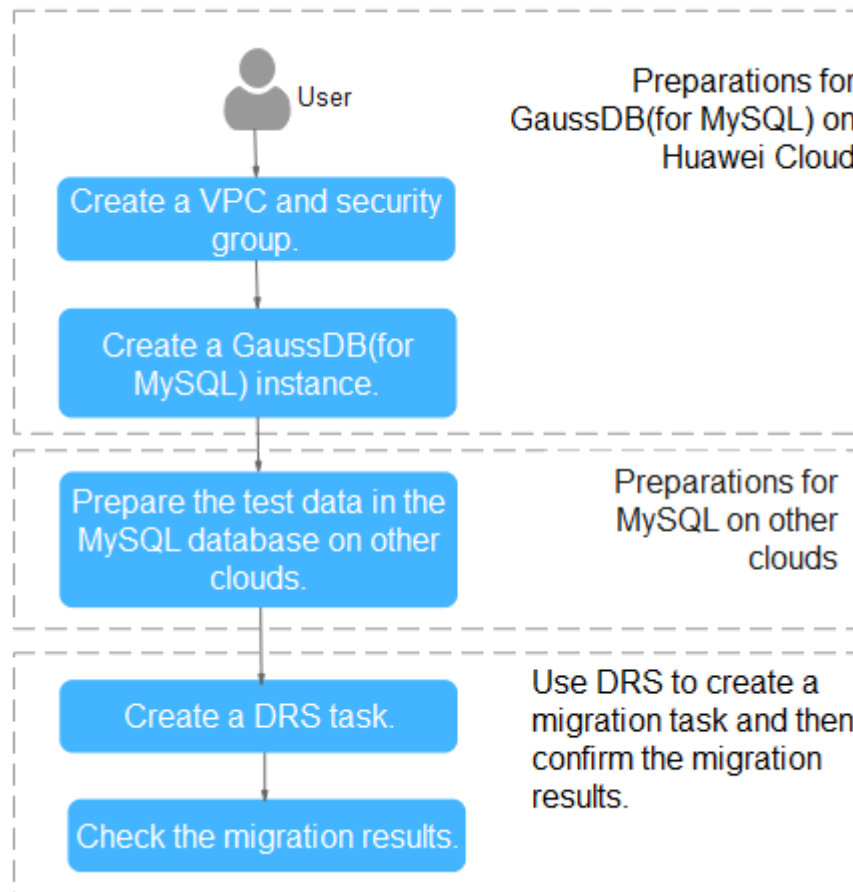
Table 2-1 Resource planning

Category	Subcategory	Plan	Description
VPC	VPC name	vpc-DRStest	Specify a name that is easy to identify.
	Region	EU-Dublin	To achieve lower network latency, select the region nearest to you.
	AZ	AZ 1	-
	Subnet	10.0.0.0/24	Select a subnet with sufficient network resources.
	Subnet name	subnet-drs01	Specify a name that is easy to identify.

Category	Subcategory	Plan	Description
Other cloud MySQL	DB engine version	MySQL 5.7	-
	IP address	10.154.217.42	Enter an IP address.
	Port	3306	-
GaussDB(for MySQL) instance	Instance name	gauss-drstar	Specify a name that is easy to identify.
	DB engine version	MySQL 8.0	-
	AZ type	Single AZ	In this example, a single AZ is used. To improve service reliability, select multiple AZs.
	AZ	AZ1	AZ1 is selected in this example. To improve service reliability, deploy the instance across multiple AZs.
	Instance class	Dedicated 4 vCPUs 16 GB	-
DRS migration task	Task name	DRS-test-migrate	Specify a name that is easy to identify.
	Source DB engine	MySQL	-
	Destination DB engine	GaussDB(for MySQL)	-
	Network type	Public network	Public network is used in this example.

2.3 Operation Process


Figure 2-2 Flowchart

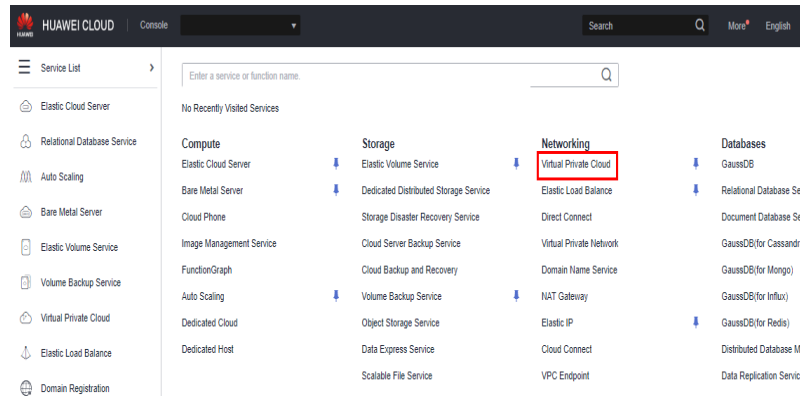


2.4 Creating a VPC and Security Group

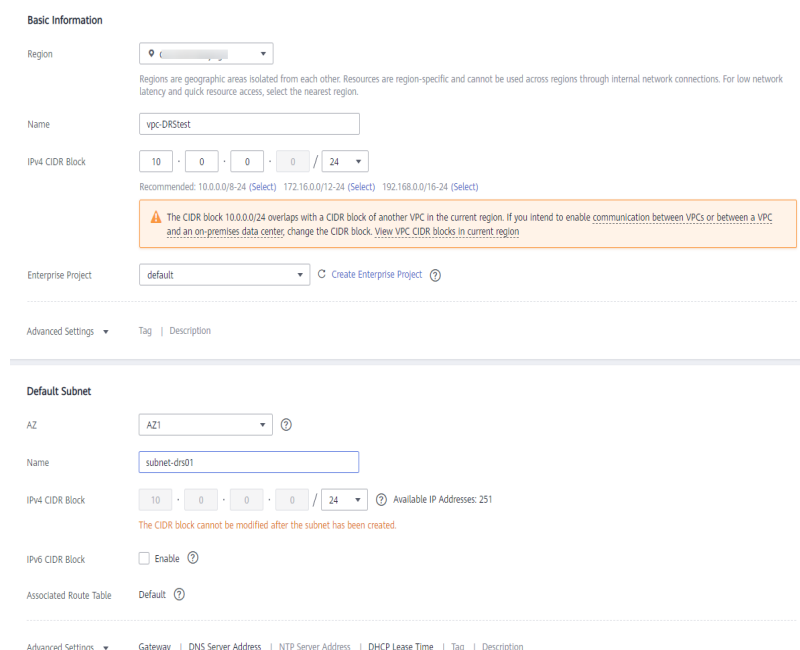
Create a VPC and security group for a GaussDB(for MySQL) instance.

Creating a VPC

- Step 1** Log in to the [management console](#).
- Step 2** Click  in the upper left corner of the management console and select region EU-Dublin.
- Step 3** Under the service list, choose **Networking > Virtual Private Cloud**.



Step 4 Click Create VPC.



Step 5 Configure parameters as needed and click **Create Now**.


Step 6 Return to the VPC list and check whether the VPC is created.

If the VPC status becomes available, the VPC has been created.

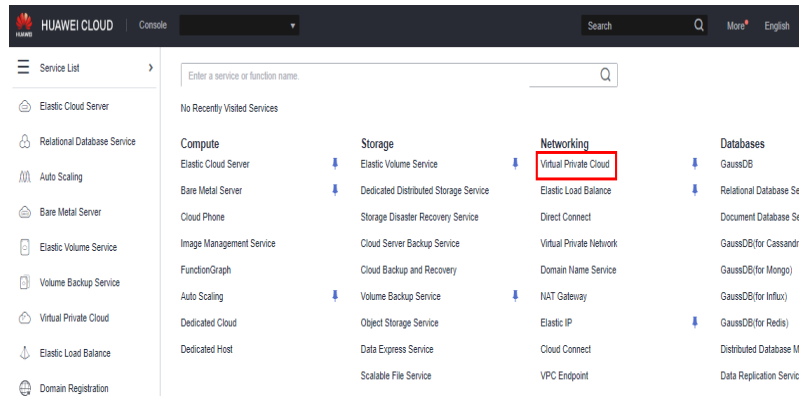
----End

Creating a Security Group

Step 1 Log in to the **management console**.

Step 2 Click  in the upper left corner of the management console and select region EU-Dublin.

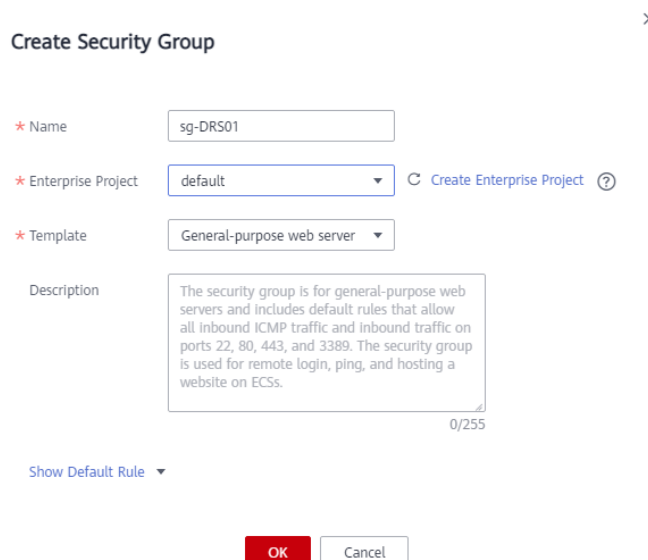
Step 3 Under the service list, choose **Networking > Virtual Private Cloud**.



Step 4 In the navigation pane, choose **Access Control > Security Groups**.

Step 5 Click **Create Security Group**.

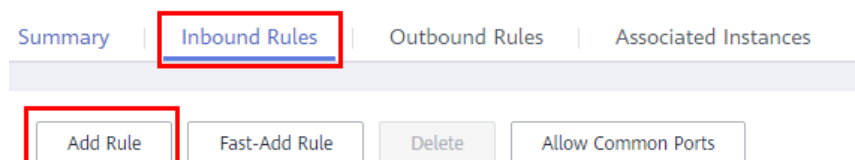
Step 6 Configure parameters as needed.



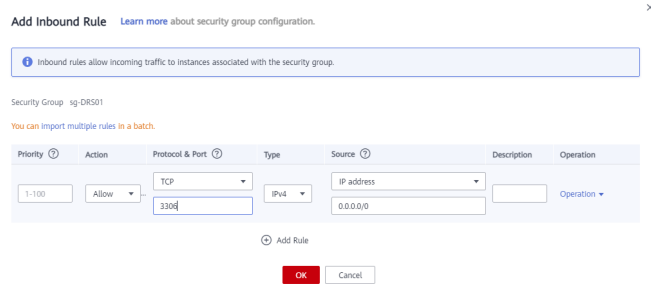
Step 7 Click **OK**.

Step 8 Return to the security group list and click the security group name (**sg-DRS01** in this example).

Step 9 Click the **Inbound Rules** tab, and then click **Add Rule**.




Step 10 Configure an inbound rule to allow access from database port **3306**.



----End

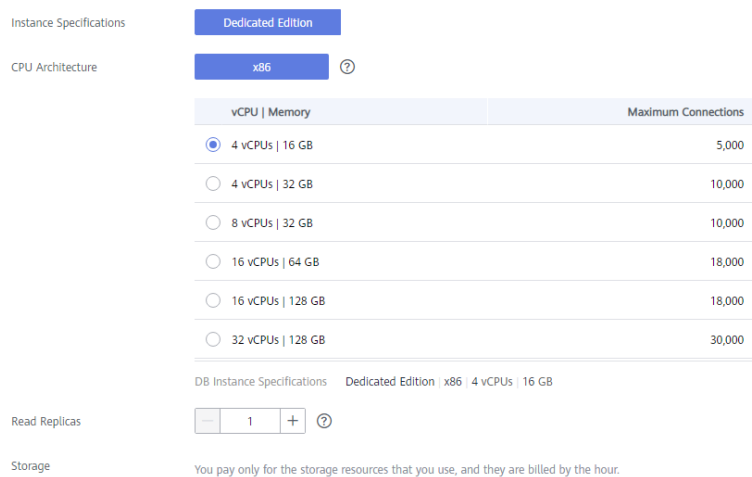
2.5 Creating a GaussDB(for MySQL) Instance

This section describes how to create a Huawei Cloud GaussDB(for MySQL) instance.

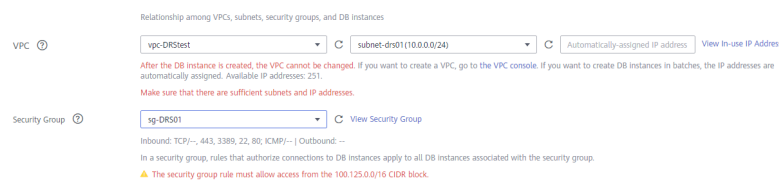
- Step 1** Log in to the [management console](#).
- Step 2** Click  in the upper left corner of the management console and select region EU-Dublin.
- Step 3** Under the service list, choose **Databases > GaussDB(for MySQL)**.
- Step 4** On the **Instances** page, click **Buy DB Instance**.
- Step 5** Configure the instance name and basic information.



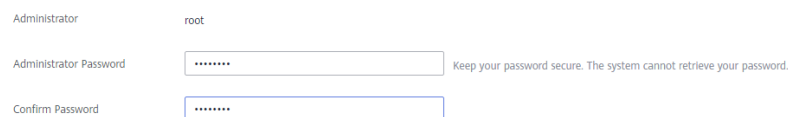
- Step 6** Configure instance specifications.



Step 7 Select a VPC and security group for the instance and configure the database port. The VPC and security group have been created in [Creating a VPC and Security Group](#).



Step 8 Configure the instance password.



Step 9 Click **Next**. If you do not need to modify your settings, click **Submit**.

Step 10 Return to the instance list. If the instance becomes **Available**, the instance has been created.

----End

2.6 Configuring a MySQL Instance on Other Clouds

Prerequisites

- You have purchased a MySQL instance on other platforms.
- The MySQL account has the migration permissions listed in [Permission Requirements](#).

Permission Requirements

To migrate data from a MySQL database on other clouds to a GaussDB(for MySQL) instance, the following permissions are required.

Table 2-2 Required permissions

Database	Full Migration Permission	Full+Incremental Migration Permission
Source DB (MySQL)	SELECT, SHOW VIEW, and EVENT	SELECT, SHOW VIEW, EVENT, LOCK TABLES, REPLICATION SLAVE, and REPLICATION CLIENT

For details about MySQL authorization operations, see [operation guide](#).

Network Configuration

Enable public accessibility for the source database. The method for enabling public accessibility depends on the cloud database vendor. For details, see the official documents of the corresponding cloud database vendor.

2.7 Creating a DRS Migration Task

This section describes how to create a DRS instance and migrate data from a MySQL database on other clouds to a GaussDB(for MySQL) instance.


Pre-migration Check

Before creating a migration task, check the migration conditions.

This section describes how to migrate data from a MySQL database to GaussDB(for MySQL). For details, see [Before You Start](#).

Creating a Migration Task

Step 1 Log in to the [management console](#).

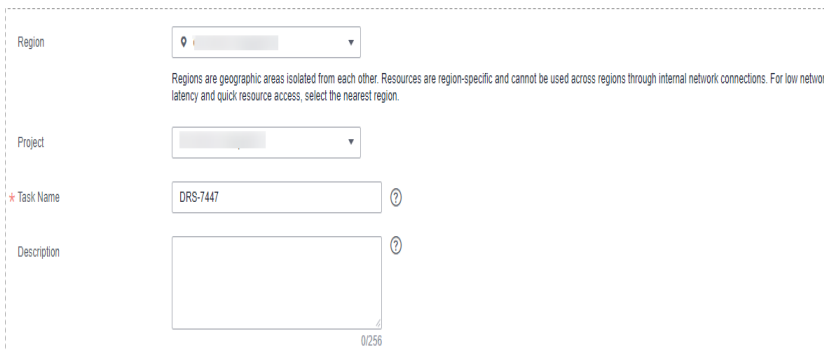
Step 2 Click  in the upper left corner of the management console and select region EU-Dublin.

Step 3 Under the service list, choose **Databases > Data Replication Service**.

Step 4 In the upper right corner, click **Create Migration Task**.

Step 5 Configure parameters as needed.

1. Specify a migration task name.



Region

Regions are geographic areas isolated from each other. Resources are region-specific and cannot be used across regions through internal network connections. For low network latency and quick resource access, select the nearest region.

Project

* Task Name ?

Description ?

0/256

2. Configure replication instance details as needed.

Select the GaussDB(for MySQL) instance created in [Creating a GaussDB\(for MySQL\) Instance](#) as the destination database.

The screenshot shows the 'Replication Instance Details' configuration page. It includes the following sections:

- Data Flow:** Buttons for 'To the cloud', 'Out of the cloud', and 'Self-built to self-built'. A note states: 'The destination database must be a database in the current cloud. If you want to migrate data between databases, select To the cloud.'
- Source DB Engine:** Buttons for 'MySQL', 'MySQL schema and log table', 'MongoDB', and 'Reds'.
- Destination DB Engine:** Buttons for 'MySQL', 'DMM', and 'GaussDB for MySQL'.
- Network Type:** A dropdown menu set to 'Public network'. A note below states: 'I understand that an EIP will be automatically bound to the replication instance and released after the replication task is complete.'
- Destination DB Instance:** A dropdown menu with 'Select an instance' and buttons for 'View DB Instance' and 'View Unselectable DB Instance'.
- Replication Instance Subnet:** A dropdown menu with 'Select the subnet' and a button for 'View Subnets'.
- Migration Type:** Buttons for 'Full-Incremental' and 'Full'. A note below states: 'This migration type is suitable for scenarios where services can be interrupted. It migrates database objects and data. In a non-system database to a destination database all at the same time. After the data migration is complete, the task automatically stops.'
- Destination DB Instance Access:** Buttons for 'Read/Write' and 'Read Only'. A note below states: 'During the migration, the destination DB instance becomes read-only to ensure the integrity and success of data migration. When the task is complete, the DB instance becomes readable and writable. This process takes a few minutes. This option is recommended.'

Step 6 Click **Create Now**.

It takes about 5 to 10 minutes to create a replication instance.

Step 7 Configure a whitelist for the source database to manage network access.

Add the EIP of the DRS replication instance to the whitelist of the source MySQL database to ensure that the source database can communicate with the DRS instance.

The method for configuring the whitelist depends on the cloud database vendor. For details, see the official documents of the corresponding cloud database vendor.

Step 8 Configure source and destination database information.

1. Configure the source database information and click **Test Connection**. If a successful test message is returned, login to the destination is successful.

The screenshot shows the 'Source Database' configuration form with the following fields and options:

- IP Address or Domain Name:** A text input field.
- Port:** A text input field containing '3306'.
- Database Username:** A text input field containing 'root'.
- Database Password:** A password input field with a visibility toggle.
- SSL Connection:** A toggle switch that is currently turned off.
- Test Connection:** A button that has been clicked, resulting in a green status indicator and the text 'Test successful'.

2. Configure destination database information and click **Test Connection**. If a successful test message is returned, login to the destination is successful.

Destination Database

The screenshot shows the 'Destination Database' configuration form with the following fields and options:

- DB Instance Name:** A text input field containing 'gauss-drstar'.
- Database Username:** A text input field containing 'root'.
- Database Password:** A password input field with a visibility toggle.
- Migrate Definer to User:** A radio button selected for 'Yes'.
- Test Connection:** A button at the bottom of the form.

Step 9 Click **Next**.

Step 10 On the **Set Task** page, select migration accounts and objects.

- Select **No** for **Migrate Account**.
- Select **All** for **Migrate Object**.

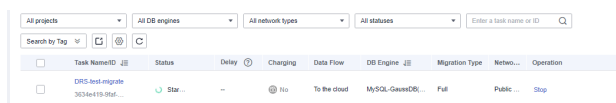
Step 11 Click **Next**. On the **Check Task** page, check the migration task.

- If any check fails, review the cause and rectify the fault. After the fault is rectified, click **Check Again**.
- If all check items are successful, click **Next**.

Step 12 Click **Submit**.

Return to the **Online Migration Management** page and check the migration task status.

It takes several minutes to complete.



The screenshot shows a table of migration tasks. The table has columns for Task Name/ID, Status, Delay, Charging, Data Flow, DB Engine, Migration Type, Network, and Operation. One task is visible with the name 'DRS-test-migrate' and ID '2023-11-19-10:10:10'. Its status is 'Star' and it is 'To be closed'. The DB Engine is 'MySQL-GaussDB' and the Migration Type is 'Full'.

If the status changes to **Completed**, the migration task has been created.

NOTE

- Currently, MySQL to GaussDB(for MySQL) migration supports two modes: full migration and full+incremental migration.
- If you create a full migration task, the task automatically stops after the full data is migrated to the destination.
- If you create a full+incremental migration task, a full migration is executed first. After the full migration is complete, an incremental migration starts.
- During the incremental migration, data is continuously migrated so the task will not automatically stop.

----End


2.8 Checking Migration Results

You can use either of the following methods to check the migration results:

1. DRS compares migration objects, users, and data and provide comparison results. For details, see [Checking the Migration Results on the DRS Console](#).
2. Log in to the destination side to check whether the databases, tables, and data are migrated. Confirm the data migration status. For details, see [Checking the Migration Results on the GaussDB\(for MySQL\) Console](#).


Checking the Migration Results on the DRS Console

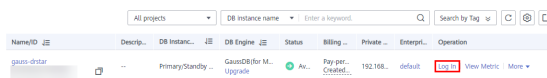
Step 1 Log in to the [management console](#).

Step 2 Click  in the upper left corner of the management console and select region EU-Dublin.

- Step 3** Under the service list, choose **Databases > Data Replication Service**.
 - Step 4** Click the DRS instance name.
 - Step 5** Click **Migration Comparison** and select **Object-Level Comparison** to check whether database objects are missing.
 - Step 6** Choose **Data-Level Comparison** and check whether the number of rows of migrated objects is consistent.
 - Step 7** Click **Account-Level Comparison** and check whether the accounts and permissions of the source and destination instances are the same.
- End

Checking the Migration Results on the GaussDB(for MySQL) Console

- Step 1** Log in to the [management console](#).
- Step 2** Click  in the upper left corner of the management console and select region EU-Dublin.
- Step 3** Under the service list, choose **Databases > GaussDB(for MySQL)**.
- Step 4** On the **Instances** page, locate the destination instance, and click **Log In** in the **Operation** column.



- Step 5** In the dialog box that is displayed, enter the password and click **Test Connection**.
 - Step 6** After the connection is successful, click **Log In**.
 - Step 7** Check whether the destination databases and tables are the same as the source instance. Check whether migration is complete.
- End