GeminiDB Cassandra

User Guide

Issue 01

Date 2025-10-24

HUAWEI CLOUD COMPUTING TECHNOLOGIES CO., LTD.

Trademarks and Permissions


and other Huawei trademarks are the property 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 Cloud 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.

Contents

1 Service Overview

1.1 What Is GeminiDB Cassandra API?

GeminiDB Cassandra API is a cloud-native NoSQL database service compatible with Cassandra, DynamoDB, and HBase. It supports Cassandra Query Language (CQL), which gives you SQL-like syntax. With robust security and reliability, GeminiDB Cassandra API offers ultra-high performance and addresses pain points of open-source Cassandra.

- High security and reliability
  - A multi-layer security system, including a VPC, subnets, security groups, SSL, and fine-grained permissions control, ensures database security and user privacy.
  - Cross-region active-active DR is supported. You can deploy an instance across three AZs and quickly back up or restore data to improve data reliability.
  - The distributed architecture provides superlative fault tolerance (*N*-1 reliability).
- Future-proof ecosystem
  - It is fully compatible with open-source Cassandra.
  - It supports a SQL-like syntax and offers you a MySQL-like user experience.
  - You can smoothly migrate data from DynamoDB tables to GeminiDB DynamoDB-Compatible instances.
- Enhanced capabilities
  - Enhanced indexing makes it easier to query massive sets of data in complex scenarios.
  - Data recovery capabilities such as second-level flashback and Point-In-Time Recovery (PITR) ensure high data reliability.
- Superior performance
  - A wide table model enables it to store petabytes of structured and semi-structured data.

  - It delivers powerful write performance, several times higher read-only performance and 2 times higher I/O performance than open-source Cassandra.
- No pain points of open-source software
  - Data keeps consistent, and there are no garbage collection (GC) issues.
  - Storage can be scaled in seconds and without affecting services.
  - Compute nodes can be added in minutes. Network jitter may last only a few seconds.

Typical Application Scenarios

- Internet

  GeminiDB Cassandra API offers superior I/O performance, high availability, dynamic scalability, and high fault tolerance. It can handle concurrent requests at low latency, ideal for Internet websites with large data volumes, for example, product catalogs, recommendations, personalized engines, and transaction records.

  Advantages

  Large-scale clusters

  Each cluster can include up to 100 nodes, helping write-intensive Internet applications process massive volumes of data.

  High availability and scalability

  The failure of one node does not affect the availability of the entire cluster. Compute resources and storage space can be quickly scaled out or up, with minimal service interruptions.

  High-concurrency writes

  Powerful write performance helps you handle a huge number of concurrent e-commerce transactions.

- Industrial data collection

  GeminiDB Cassandra API is fully compatible with Cassandra, so it can help you collect, organize, and store data from different types of terminals, and aggregate and analyze the data in real-time.

  Advantages

  Large-scale clusters

  Large-scale clusters are well suited for collecting and storing massive industrial manufacturing metrics.

  High availability and performance

  Data can be written to databases around the clock.

  Fast backup and restoration

  Storage snapshots speed up backup and recovery.

  Scaling in minutes

  Nodes can be added in minutes to effortlessly handle surges in jobs and projects.

1.2 Compatible APIs and Versions

This section describes the compatible APIs and versions supported by GeminiDB Cassandra.

Table 1-1 Compatible APIs and versions

Compatible API	Instance Type	Version
Cassandra	Cluster	3.11 and 4.0

1.3 Instance Specifications

Instances of the same type can have different memory specifications. You can select instances of different specifications based on application scenarios.

This section describes the instance specifications supported by GeminiDB Cassandra. The instance specifications depend on the selected flavor.

Table 1-2 GeminiDB Cassandra cluster instance specifications

Flavor	vCPUs	Memory (GB)	Min. Storage Space (GB)	Max. Storage Space (GB)
geminidb.cassandra.large.4	2	8	100	12,000
geminidb.cassandra.xlarge.4	4	16	100	24,000
geminidb.cassandra.2xlarge.4	8	32	100	48,000
geminidb.cassandra.4xlarge.4	16	64	100	96,000
geminidb.cassandra.8xlarge.4	32	128	100	192,000
geminidb.cassandra.large.8	2	16	100	12,000
geminidb.cassandra.xlarge.8	4	32	100	24,000
geminidb.cassandra.2xlarge.8	8	64	100	48,000
geminidb.cassandra.4xlarge.8	16	128	100	96,000

Flavor	vCPUs	Memory (GB)	Min. Storage Space (GB)	Max. Storage Space (GB)
geminidb.cassandra.6xlarge.8	24	192	100	144,000
geminidb.cassandra.8xlarge.8	32	256	100	192,000

Table 1-3 Specifications of a GeminiDB Cassandra instance with cloud native storage

Data Node Flavor	vCPUs	Memory (GB)	Min. Storage Space (GB)	Max. Storage Space (GB)
geminidb.cassandra-geminifs.large.4	2	8	10	64000
geminidb.cassandra-geminifs.xlarge.4	4	16	10	64000
geminidb.cassandra-geminifs.2xlarge.4	8	32	10	64000
geminidb.cassandra-geminifs.4xlarge.4	16	64	10	64000
geminidb.cassandra-geminifs.8xlarge.4	32	128	10	64000

1.4 Instance Statuses

The status of an instance indicates the health of the instance. You can view the status of an instance on the console.

Table 1-4 Instance statuses

Status	Description
Available	The DB instance is available.
Abnormal	The instance is abnormal.
Creating	The instance is being created.
Creation failed	DB instance creation fails.
Restarting	The instance is being restarted.

Status	Description
Resetting password	The administrator password is being reset.
Adding node	Nodes are being added to an instance.
Deleting node	Nodes are being deleted from an instance.
Scaling storage space	The storage space of an instance is being scaled up.
Changing specifications	The vCPUs and memory of an instance are being changed.
Uploading backup	The backup file is being uploaded.
Backing up	A database backup is being created.
Checking restoration	The backup of the instance is being restored to a new instance.
Changing to yearly/monthly	The billing mode is being changed from pay-per-use to yearly/monthly.
Changing to pay-per-use	The billing mode is being changed from yearly/monthly to pay-per-use.
Creating a DR cluster	A DR instance is being created.
Canceling DR relationship	A DR instance is being deleted.
Frozen	The instance is frozen because your balance drops to or below zero.
Unfreezing	Overdue payments are cleared, and the DB instance is being unfrozen.
Checking changes	The yearly/monthly instance is pending check when its billing mode is changed.

1.5 Database Rules

1.5.1 Basic Design

Design Rules

Rule 1: Do not store big data such as images and files in databases.

Rule 2: The maximum size of the key and value in a single row cannot exceed 64 KB, and the average size of rows cannot exceed 10 KB.

Rule 3: A data deletion policy must be specified for a table to prevent data from growing infinitely.

Rule 4: Partition keys can evenly distribute workloads to avoid data skew.

A partition key of a primary key determines a logical partition for storing table data. If partition keys are not evenly distributed, data and load between nodes are unbalanced, resulting in a data skew problem.

Rule 5: The design of partition keys can evenly distribute data access requests to avoid BigKey or HotKey issues.

- BigKey issue: The main cause of BigKey is that the primary key is improperly design. As a result, there are too many records or too much data in a single partition. Once a partition becomes extremely large, access to the partition increases load of a server where the partition is located, and even causes the Out of Memory (OOM) error.
- HotKey issue: This issue occurs when a key is frequently operated in a short period of time. For example, breaking news can cause a spike in traffic and large number of requests. As a result, the CPU usage and the load on the node on which the key is located increase, affecting other requests to the node and reducing the success rate of services. HotKey issues will also occur during promotion of popular products and Internet celebrity live streaming.

For details about how to handle BigKey and HotKey issues, see **How Do I Detect and Resolve BigKey and HotKey Issues?**

Rule 6: The number of rows of a single partition key cannot exceed 100,000, and the disk space of a single partition cannot exceed 100 MB.

- The number of rows of a single partition key cannot exceed 100,000.
- The size of records under a single partition key cannot exceed 100 MB.

Rule 7: Ensure strong consistency between data copies written to GeminiDB Cassandra, but do not support transactions.

Table 1-5 GeminiDB Cassandra consistency description

Consistency Model	Consistency Supported	Description
Concurrent write consistency	Yes	GeminiDB Cassandra does not support transactions, and data writing is strongly consistent.
Consistency between tables	Yes	GeminiDB Cassandra does not support transactions, and data writing is strongly consistent.
Data migration consistency	Eventual consistency	DRS migration provides the data sampling, comparison, and verification capabilities. After services are migrated, data verification is automatically performed.

Rule 8: For large-scale storage, database splitting must be considered.

Ensure that the number of nodes in the GeminiDB Cassandra cluster is less than 100. If the number of nodes exceeds 100, split the cluster vertically or horizontally.

- Vertical splitting: Data is split by functional module, for example, the order database, product database, and user database. In this mode, the table structures of multiple databases are different.
- Horizontal sharding: Data in the same table is divided into blocks and stored in different databases. The table structures in these databases are the same.

Rule 9: Avoid tombstones caused by large-scale deletion.

- Use TTL instead of Delete if possible.
- Do not delete a large amount of data. Delete data by primary key prefix.
- A maximum of 1,000 rows can be deleted at a time within a partition key.
- Avoid querying deleted data during range query.
- Do not frequently delete data of a large range in one partition.

Design Suggestion

Suggestion 1: Properly control the database scale and quantity.

- It is recommended that the number of data records in a single table be less than or equal to 100 billion.
- It is recommended that a single database contain no more than 100 tables.
- It is recommended that the maximum number of fields in a single table be 20 to 50.

Suggestion 2: Estimate how many resources that GeminiDB Cassandra servers can process.

- If it is estimated that N nodes need to be used, adding additional N/2 nodes is recommended for fault tolerance and performance consistency.
- In normal scenarios, the CPU usage of each node is limited to 50% to avoid fluctuation during peak hours.

Suggestion 3: To store large volumes of data, perform a test run based on service scenarios.

For a large number of requests and data volumes, you need to test performance in advance because the service read/write ratio, random access mode, and instance specifications vary greatly.

Suggestion 4: Split database cluster granularity properly.

- In distributed scenarios, microservices of a service can share a GeminiDB Cassandra cluster to reduce resource and maintenance costs.
- The service can be divided into different clusters based on the data importance, number of tables, and number of records in a single table.

Suggestion 5: Do not frequently update some fields in a single data record.

Suggestion 6: If there are too many nested elements such as List, Map, or Set, read and write performance will be affected. In this case, convert such elements into JSON data for storage.

1.5.2 Database Objects

Naming Rules

Rule 1: The object name cannot be duplicated with any keyword of the database.

Rule 2: Object names (including database names, table names, field names, and index names) must be in lowercase and separated by underscores (_).

Rule 3: The length of an object name (including the database name, table name, field name, and index name) cannot exceed 30 characters.

Rule 4: The table alias must be short. Generally, aliases are in lowercase letters.

Table Design Rules

Rule 1: Compatibility must be considered during table design.

Columns can be added but cannot be deleted.

Rule 2: The table name and database name cannot exceed 48 bytes.

Rule 3: By default, tables are created based on the optimal performance specifications. If the high-performance table is not required, you can set performance parameter **Z00_THROUGHPUT** to **big**, **medium**, or **small** when creating a table. By default, this parameter is set to **big**. If you use RocksDB as the storage engine, memory needs to be allocated in advance and the number of tables created in an instance is limited. For details, see **What Should I Pay Attention to When Creating a GeminiDB Cassandra Table?**.

If necessary, use denormalization and redundancy to improve the read performance.

Indexing Rules

Rule 1: Design all queries as primary-key based queries and do not rely too much on secondary indexes.

Rule 2: An index can be used for query only after it is configured.

Rule 3: Do not frequently update indexes.

Rule 4: Do not create an index column for a table that contains too many duplicate values. For example, if one table stores 100 million data records and one of its columns contains the same data or a few types data, creating an index column for this table is not recommended.

Rule 5: The **counter** column cannot be indexed.

Rule 6: Do not create an index for any column that is frequently updated or deleted.

Rule 7: Use indexes together with partition keys to minimize message forwarding between nodes and resource consumption and prevent out-of-memory or high CPU usage.

View Rules

- If a materialized view is used, ensure that the original table corresponds to no more than three views. The more views the original table corresponds to, the greater impacts on the synchronization of views.
- Do not use any frequently-updated field in the original table as the primary key of a view.

Flow Table Rules

One flow table stores 24 hours of data by default. If there is a large amount of data to be queried, return results on multiple pages. No more than 100 query results are returned each time and a retry is allowed if a query request times out.

1.5.3 Access and Connection Pools

Rule 1: A connection pool must be used to access the database to improve reliability.

Rule 2: GeminiDB Cassandra clusters use RoundRobinPolicy for load balancing.

1.5.4 Batches

Rule 1: Logged batches are not supported. Only unlogged batches are supported.

Rule 2: A maximum of 25 rows of data can be operated in a batch.

Rule 3: In a batch, a request size cannot exceed 5 KB.

Rule 4: In a batch, no more than 10 partitions are involved, and only one table is operated.

1.5.5 Queries

Using a Sort Key for Range Query

It is recommended that the sequence of the sort keys for range query be the same as that used during table creation. Otherwise, the performance deteriorates.

NOTE

If no sort key sequence is specified, the default sort key sequence is ASC during query and table creation.

Not Using ALLOW FILTERING

If a query statement does not specify all primary keys and contains **ALLOW FILTERING**, the query will scan and filter the entire table. A table with a large data volume may cause the query to time out. **ALLOW FILTERING** is forbidden in later kernel versions.

NOTE

Query timeout and excessive resource usage issues that occur when **ALLOW FILTERING** is used are not within commitments on SLAs

COUNT Query

If a database contains a very large amount of data, do not run the following statement to query the database. Otherwise, the query may fail.

```
select count(*) from "test" where sds_uid='10000000000000000006250004';
```

The following statement is recommended:

```
select sum(row_count) From system_distributed.size_estimates WHERE keyspace_name='' and
table_name='';
```

NOTE

This query is an asynchronous task in the background, so the results are not accurate and for reference only.

2 Getting Started with GeminiDB Cassandra API

2.1 Getting Started with GeminiDB Cassandra API

This section instructs you to create and connect to a GeminiDB Cassandra instance.

Connection Methods

DAS enables you to manage instances on a web-based console, simplifying database management and improving working efficiency. You can connect and manage instances through DAS. By default, you have the permission of remote login. DAS is secure and convenient for connecting to GeminiDB Cassandra instances.

Table 2-1 Connection on DAS

Method	Scenario	Remarks
DAS	You can log in to an instance on the console without using an IP address.	• Easy to use, secure, advanced, and intelligent • By default, you have the permission of remote login. DAS is secure and convenient for connecting to instances.

More Connection Operations

- See **Connecting to a GeminiDB Cassandra Instance**.

2.2 Buying and Connecting to a GeminiDB Cassandra Instance

This section instructs you to buy a GeminiDB Cassandra instance on the GeminiDB console.

GeminiDB Cassandra and DynamoDB-Compatible instances of each tenant share a quota. Each tenant can create a maximum of 50 instances by default. To request a higher quota, choose **Service Tickets > Create Service Ticket** in the upper right corner of the console and contact customer service personnel.

- **Step 1: Buying an Instance**
- **Step 2: Connecting to an Instance Through DAS**

  For details about other connection methods, see **Connecting to a GeminiDB Cassandra Instance**.

Prerequisites

- You have created a Huawei Cloud account.

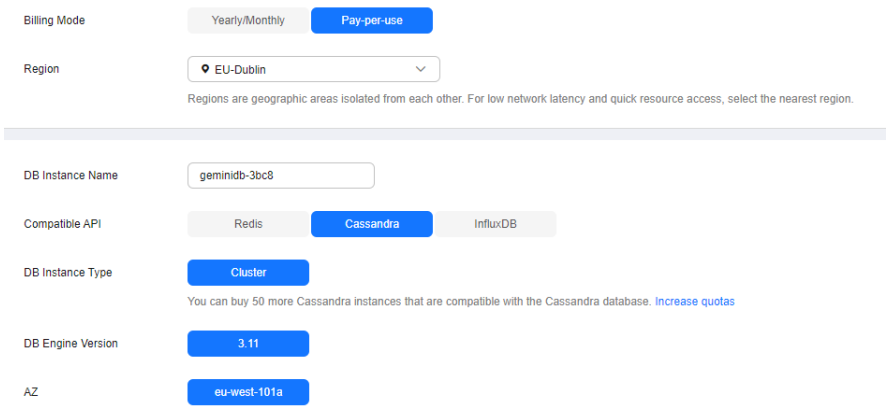
Step 1: Buying an Instance

For details, see **Buying a GeminiDB Cassandra Instance**.

1. Log in to the **GeminiDB console**.
2. On the **Instances** page, click **Buy DB Instance**.
3. On the displayed page, select a billing mode, configure instance parameters, and click **Next**.

   The following parameters are for reference only. Select proper specifications as needed. **Table 3-1** lists details about the parameters.

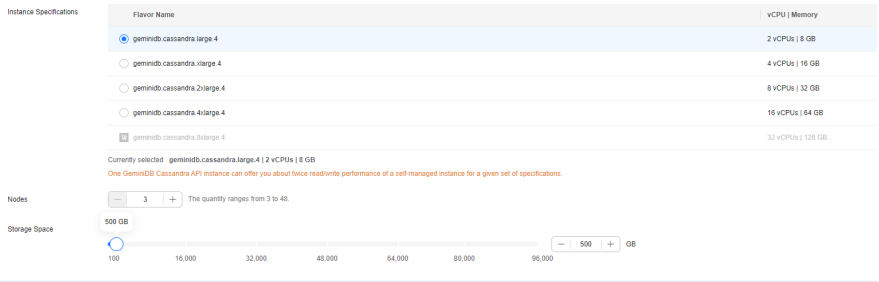
   Figure 2-1 Billing mode and basic information



Parameter	Example Value	Description
Billing Mode	Pay-per-use	Billing mode of an instance • **Yearly/Monthly**: A prepaid billing mode in which you pay for resources before using it. Bills are settled based on the subscription period. The longer the subscription term, the bigger the discount. This mode is a good option for long-term stable services. • **Pay-per-use** is a postpaid mode. You are billed based on how long you have actually used GeminiDB. Pricing is listed on a per-hour basis, and bills are calculated down to the second. This mode allows you to adjust resource usage easily. You neither need to prepare for resources in advance, nor end up with excessive or insufficient preset resources.
Region	Select EU-Dublin.	Region where a tenant is located **NOTE** To reduce network latency, select a region nearest from which you will access the instance. Instances deployed in different regions cannot communicate with each other over a private network. After you buy an instance, you cannot change its region.
DB Instance Name	User-defined	The instance name: • Can be the same as an existing instance name. • Can contain 4 to 64 characters and must start with a letter. It is case-sensitive and allows only letters, digits, hyphens (-), and underscores (_).
Compatible API	Cassandra	GeminiDB is compatible with mainstream NoSQL databases, including Redis, DynamoDB, Cassandra, and InfluxDB. You can select GeminiDB APIs by following **How Do I Select an API?**
Storage Type	Classic	• **Classic**: classic architecture with decoupled storage and compute • **Cloud native**: new, more flexible, new-gen version with support for more AZs

Parameter	Example Value	Description
DB Instance Type	Cluster	**Cluster** One cluster consists of at least three nodes. A cluster is easy to scale out to meet increasing data growth needs. A cluster is recommended when dealing with stringent availability demands, substantial data volumes, and the need for seamless scalability.
DB Engine Version	3.11	3.11
AZ	AZ 1, AZ 2, and AZ 3	Availability zone where the instance is created. An AZ is a part of a region with its own independent power supplies and networks. AZs are physically isolated but can communicate with each other over a private network.

Figure 2-2 Specifications and storage

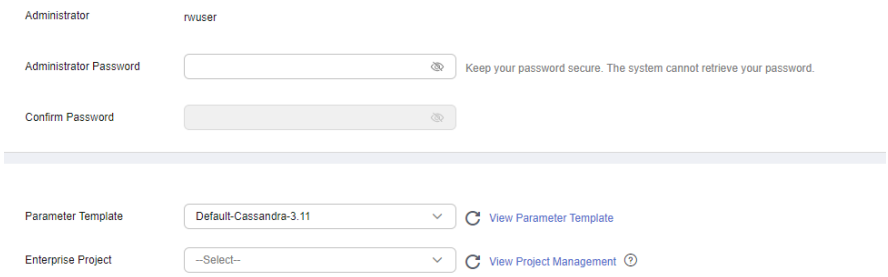


Parameter	Example Value	Description
Instance Specifications	2U8GB	Select appropriate specifications based on the CPU-memory ratio. After an instance is created, you can change its specifications. For details, see **Changing vCPUs and Memory**.

Parameter	Example Value	Description
Nodes	3	Number of nodes that the instance is deployed on. Currently, a maximum of 60 nodes are supported. To add more, choose **Service Tickets > Create Service Ticket** in the upper right corner of the console and contact the customer service. After an instance is created, you can add nodes. For details, see **Manually Adding Instance Nodes**.
Storage Space	500 GB	Instance storage space. The range depends on the instance specifications. For details, see **Instance Specifications**. After an instance is created, you can scale up its storage if necessary. For details, see **Manually Scaling Up Storage Space**.
Autoscaling	Toggled off	**Autoscaling** is toggled off by default. You can enable **Auto Scale** after an instance is created. For details, see **Automatically Scaling Up Storage Space**.
Disk Encryption	Disable	**Disable** is selected by default. If you select **Enable**, your data will be encrypted on disks and stored in ciphertext after you create an instance. When you download encrypted objects, the ciphertext will be decrypted into plain text and then sent to you. Disk encryption can improve data security and may have slight impacts on database writes and reads.

Figure 2-3 Network configuration

Figure 2-4 Database configuration



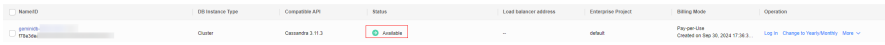
Parameter	Example Value	Description
VPC	default_vpc	Virtual private network where your instances are located. A VPC isolates networks for different services. You can select an existing VPC or create a VPC. **NOTE** • After a GeminiDB Cassandra instance is created, its VPC cannot be changed. • To connect a GeminiDB Cassandra instance to an ECS over a private network, ensure they are in the same VPC. If they are not, create a **VPC peering connection** between them.
Subnet	default_subnet	A subnet provides dedicated network resources that are logically isolated from other networks for security purposes.
Security Group	default	A security group controls access between your instance and other services. Ensure that the security group you selected allows your client to access the instance.
Administrator Password	Configured based on the password policy	Password of the administrator account. The password: • Can include 8 to 32 characters. • Can include uppercase letters, lowercase letters, digits, and any of the following special characters: ~!@#%^*-_=+? • For security reasons, set a strong password. The system will verify the password strength. Keep your password secure. The system cannot retrieve it if it is lost.

Parameter	Example Value	Description
Parameter Template	Default-Cassandra-3.11	A template of parameters for creating an instance. The template contains API configuration values that are applied to one or more instances. After an instance is created, you can modify its parameters for optimal performance. For details, see **Modifying Parameters of GeminiDB Cassandra Instances**.
Enterprise Project	default	This parameter is provided for enterprise users. An enterprise project groups cloud resources, so you can manage resources and members by project. The default project is **default**. Select an enterprise project from the drop-down list. For more information about enterprise projects, see ***Enterprise Management User Guide***.

Retain the default values for other parameters.

4. On the displayed page, confirm instance details. To modify the configurations, click **Previous**.
5. If no modification is required, read and agree to the service agreement and click **Submit**.
6. Click **Back to Instance Management** to go to the instance list.
7. On the **Instances** page, view and manage the created instance.
   - It takes about 5 to 9 minutes to create an instance. During the process, the instance status is **Creating**.
   - After the instance is created, its status becomes **Available**.

Figure 2-5 Available instance



Step 2: Connecting to an Instance Through DAS

DAS enables you to manage DB instances from a web-based console, simplifying database management and improving efficiency. You can connect and manage instances through DAS. By default, you have permissions required for remote login. DAS is recommended for connecting to your instance.

Procedure

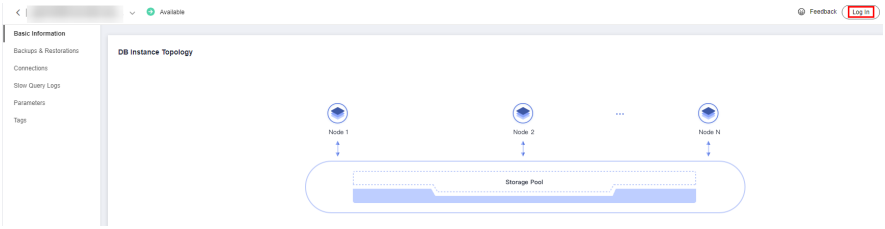
Step 1 Log in to the **GeminiDB console**.

Step 2 On the **Instance Management** page, locate the target DB instance and click **Log In** in the **Operation** column.

Figure 2-6 Logging in to the database

Alternatively, click the instance name on the **Instances** page. On the displayed **Basic Information** page, click **Log In** in the upper right corner.

Figure 2-7 Logging in to the database



Step 3 On the displayed login page, enter the administrator username and password and click **Log In**.

----End

FAQs

Question: What should I do if the DAS console cannot be redirected after I click **Log In** in the **Operation** column in the instance list or click **Log In** on the **Basic Information** page?

Solution: Set your browser to allow pop-ups and try again.

3 Working with GeminiDB Cassandra API

3.1 Using IAM to Grant Access to GeminiDB Cassandra API

3.1.1 Creating a User and Granting Permissions

This section describes how to use **IAM** to control fine-grained permissions for your GeminiDB resources. With IAM, you can:

- Create IAM users for employees based on your enterprise's organizational structure. Each IAM user will have their own security credentials for accessing GeminiDB resources.
- Grant only the permissions required for users to perform a specific task.
- Entrust a Huawei Cloud account or cloud service to perform efficient O&M on your GeminiDB resources.

If your Huawei Cloud account does not require individual IAM users, skip this section.

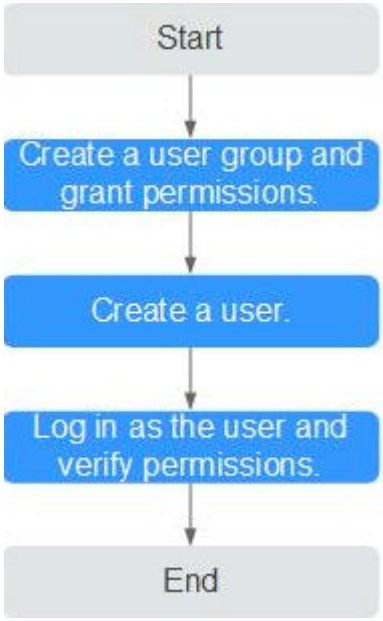
The following describes the procedure for granting permissions (see **Figure 3-1**).

Prerequisites

Learn about the permissions (see) supported by GeminiDB and choose policies or roles according to your requirements. For system permissions of other services, see **Permission Policies**.

Process Flow

Figure 3-1 Process of granting GeminiDB permissions



1. **Create a user group and assign permissions** to it.

   Create a user group on the IAM console and attach the **GeminiDB FullAccess** policy to the group.

   NOTE

   To use some interconnected services, you also need to configure permissions of such services.

   For example, when using DAS to connect to a DB instance, you need to configure the **GeminiDB FullAccess** and **DAS FullAccess** permissions.

2. **Create an IAM user** and add it to a user group.

   Create a user on the IAM console and add the user to the group created in **1**.

3. **Log in** and verify permissions.

   Log in as the created user by following **Logging In to Huawei Cloud as an IAM User**, select the authorized region, and verify permissions.

   Choose **Service List** > **GeminiDB** and click **Buy DB Instance**. If you can buy an instance, the required permission policy has taken effect.

3.1.2 Creating a Custom Policy

Custom policies can be created to supplement the system-defined policies of GeminiDB. For the actions supported for custom policies, see .

You can create custom policies in either of the following ways:

- Visual editor: Select cloud services, actions, resources, and request conditions. This does not require knowledge of policy syntax.
- JSON: Edit JSON policies from scratch or based on an existing policy.

For details, see **Creating a Custom Policy**. The following describes examples of common GeminiDB custom policies.

Example Custom Policy

- Example 1: Allowing users to create GeminiDB instances

```
{
   "Version": "1.1",
   "Statement": [
      {
         "Effect": "Allow",
         "Action": [
            "nosql:instance:create"
         ]
      }
   ]
}
```

- Example 2: Deny users the permission to delete GeminiDB instances.

  A policy with only "Deny" permissions must be used in conjunction with other policies to take effect. If the policies assigned to a user contain both Allow and Deny actions, the Deny actions take precedence over the Allow actions.

  The following method can be used if you need to assign permissions of the **GeminiDB FullAccess** policy to a user but you want to prevent the user from deleting GeminiDB instances. Create a custom policy for denying instance deletion, and attach both policies to the group to which the user belongs. Then, the user can perform all operations on GeminiDB instances except deleting GeminiDB instances. The following is an example of the deny policy:

```
{
   "Version": "1.1",
   "Statement": [
      {
        "Effect": "Deny"
        "Action": [
            "nosql:instance:delete"
         ],
       }
   ]
}
```

- Example 3: Defining permissions for multiple services in a policy

  A custom policy can contain the actions of multiple services that are of the global or project-level type. The following is an example policy containing actions of multiple services:

```
{
    "Version": "1.1",
    "Statement": [
        {
            "Action": [
                "nosql:instance:create",
                "nosql:instance:rename",
                "nosql:instance:delete",
                "vpc:publicIps:list",
                "vpc:publicIps:update"
            ],
            "Effect": "Allow"
        }
    ]
}
```


3.2 Buying a GeminiDB Cassandra Instance

This section describes how to buy a GeminiDB Cassandra instance.

GeminiDB Cassandra and DynamoDB-Compatible instances of each tenant share a quota. Each tenant can create a maximum of 50 instances by default. To request a higher quota, choose **Service Tickets > Create Service Ticket** in the upper right corner of the console and contact the customer service.

Prerequisites

- You have created a Huawei Cloud account.

Procedure

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

Step 2 On the **Instances** page, click **Buy DB Instance**.

Step 3 On the displayed page, specify instance specifications and click **Next**.

Figure 3-2 Billing mode and basic information

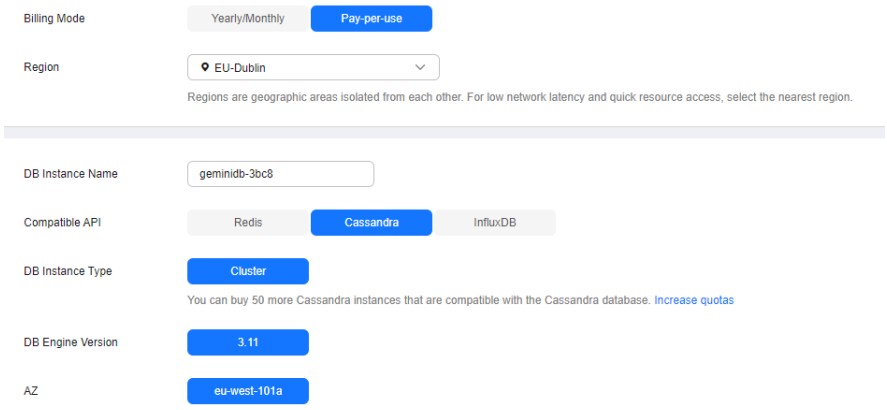


Table 3-1 Billing parameters

Parameter	Description
Billing Mode	Select **Yearly/Monthly** or **Pay-per-use**. • **Yearly/Monthly** – Specify **Required Duration**. The system deducts fees from your account based on the service price. – If you do not need such an instance any longer after it expires, change the billing mode to pay-per-use. For details, see **Changing Yearly/Monthly to Pay-per-Use**. **NOTE** ▪ Yearly/Monthly instances cannot be deleted directly. If such an instance is no longer required, unsubscribe from it. For details, see **Unsubscribing from a Yearly/Monthly Instance**. • **Pay-per-use** – If you select this billing mode, you are billed based on how much time the instance is in use. – To use an instance for a long time, change its billing mode to yearly/monthly to reduce costs. For details, see **Changing Pay-per-Use to Yearly/Monthly**.

Table 3-2 Basic information

Parameter	Description
Region	Region where a tenant is located **NOTE** To reduce network latency, select a region nearest from which you will access the instance. Instances deployed in different regions cannot communicate with each other over a private network. After you buy an instance, you cannot change its region.
DB Instance Name	The instance name: • Can be the same as an existing instance name. • Can contain 4 to 64 characters and must start with a letter. It is case-sensitive and allows only letters, digits, hyphens (-), and underscores (_). After an instance is created, you can change its name. For details, see **Changing an Instance Name**.
Compatible API	Cassandra GeminiDB is compatible with mainstream NoSQL databases, including Redis, DynamoDB, Cassandra, and InfluxDB. You can select GeminiDB APIs by following **How Do I Select an API?**

Parameter	Description
Storage Type	• **Classic**: classic architecture with decoupled storage and compute • **Cloud native**: more flexible, new-gen version with support for more AZs **NOTE** – The way you use instances with classic or cloud native storage is similar. Cloud native storage supports more AZs. If both classic and cloud native are supported, you can select any of them.
DB Instance Type	Cluster One cluster consists of at least three nodes. A cluster is easy to scale out to meet increasing data growth needs. A cluster is recommended when dealing with stringent availability demands, substantial data volumes, and the need for seamless scalability.
DB Engine Version	3.11
AZ	Availability zone where the instance is created. An AZ is a part of a region with its own independent power supplies and networks. AZs are physically isolated but can communicate with each other over a private network.

Figure 3-3 Specifications and storage

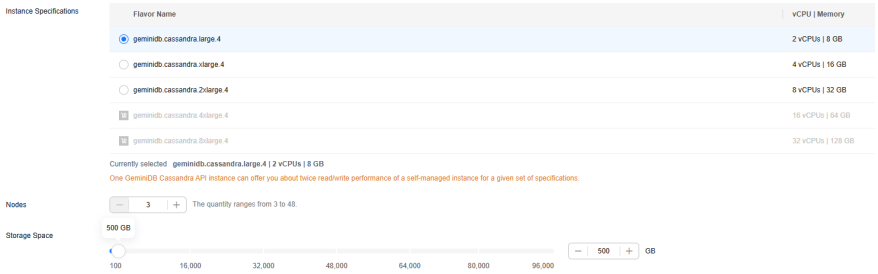


Table 3-3 Specifications and storage

Parameter	Description
Instance Specifications	Decoupled storage and compute and software-hardware synergy deliver twice or more the performance of a self-managed database with the same specifications. When you create an instance, select higher specification and specify as few nodes as possible. For example, if you need 8 vCPUs, 32 GB, and 6 nodes for an open-source instance, then for a GeminiDB Cassandra instance with 8 vCPUs and 32 GB of memory, you only need 3 nodes. Select appropriate specifications based on the CPU-memory ratio. After an instance is created, you can change its specifications. For details, see **Changing vCPUs and Memory**.
Nodes	Number of nodes that the instance is deployed on. Currently, a maximum of 48 nodes are supported. To add more, choose **Service Tickets > Create Service Ticket** in the upper right corner of the console and contact the customer service. After an instance is created, you can add nodes. For details, see **Manually Adding Instance Nodes**.

Parameter	Description
Storage Space	Instance storage space. The range depends on the instance specifications. For details, see **Instance Specifications**. You are advised to enable **Auto Scale** and set trigger conditions and storage limit. After autoscaling is triggered, the system automatically scales up the storage to ensure that the instance has sufficient storage and keeps available. Take care with the following parameters: • **Trigger If Available Storage Drops To**: storage threshold for triggering autoscaling. When the available storage usage drops to a specified threshold or the available storage drops to 10 GB, autoscaling is triggered. • **Increase By**: percentage that your instance storage will be scaled up at. If the increased storage is not a multiple of 10 GB, the system will round it up to the nearest multiple of 10 GB. At least 100 GB is added each time. • **Storage Limit**: maximum amount that the system can automatically scale up an instance's storage to. The value must be no less than the current storage of your instance and cannot exceed the maximum storage supported by your instance. After an instance is created, you can scale up its storage if necessary. For details, see **Manually Scaling Up Storage Space**. **NOTE** • Once **Auto Scale** is enabled, an agency will be created and fees will be automatically deducted. • Autoscaling is available only to users with required permissions. To use it, choose **Service Tickets > Create Service Ticket** in the upper right corner of the console and contact the customer service. • You can enable **Auto Scale** after an instance is created. For details, see **Automatically Scaling Up Storage Space**.

Figure 3-4 Network configurations

Table 3-4 Network configuration

Parameter	Description
VPC	Virtual private network where your instances are located. A VPC isolates networks for different services. You can select an existing VPC or create a VPC. If there are no VPCs available, the system automatically allocates a VPC to you. For details, see "Creating a VPC" in the *Virtual Private Cloud User Guide*. **NOTE** • After a GeminiDB Cassandra instance is created, its VPC cannot be changed. • To connect a GeminiDB Cassandra instance to an ECS over a private network, ensure they are in the same VPC. If they are not, create a **VPC peering connection** between them.
Subnet	A subnet where your instance is created. The subnet provides dedicated and isolated networks, improving network security. **NOTE** An IPv6 subnet cannot be associated with your instance. Select an IPv4 subnet.
Security Group	A security group controls access between your instance and other services. Ensure that the security group you selected allows your client to access the instance. If no security group is available, the system creates one for you.

Figure 3-5 Database configuration

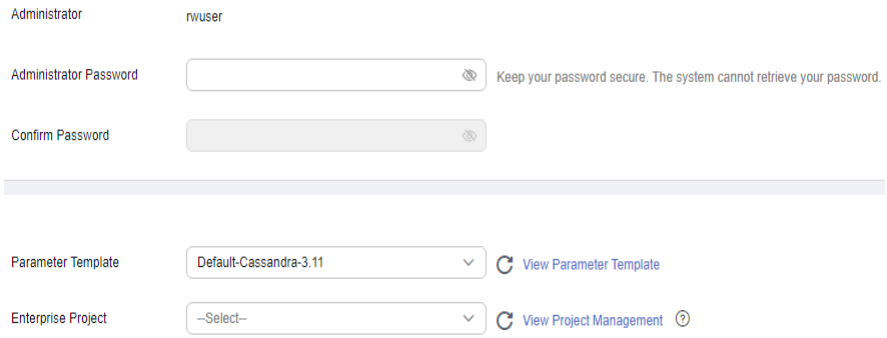


Table 3-5 Database configuration

Parameter	Description
Administrator	Username of the administrator account. The default value is **rwuser**.

Parameter	Description
Administrator Password	Password of the administrator account. The password: • Can include 8 to 32 characters. • Can include uppercase letters, lowercase letters, digits, and any of the following special characters: ~!@#%^*-_=+? • For security reasons, set a strong password. The system will verify the password strength. Keep your password secure. The system cannot retrieve it if it is lost.
Confirm Password	This password must be consistent with administrator password.
Parameter Template	A template of parameters for creating an instance. The template contains API configuration values that are applied to one or more instances. After an instance is created, you can modify its parameters for optimal performance. For details, see **Modifying Parameters of GeminiDB Cassandra Instances**.
Enterprise Project	This parameter is provided for enterprise users. An enterprise project groups cloud resources, so you can manage resources and members by project. The default project is **default**. Select an enterprise project from the drop-down list. For more information about enterprise projects, see ***Enterprise Management User Guide***.

Figure 3-6 Tag configuration

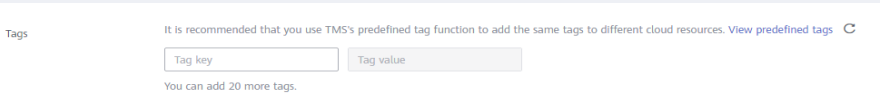


Table 3-6 Tags

Parameter	Description
Tags	This setting is optional. Adding tags helps you better identify and manage your GeminiDB Cassandra instances. A maximum of 20 tags can be added for each instance. A tag consists of a tag key and a tag value. • A tag key is mandatory if the instance will be tagged. Each tag key is unique for each instance. It can include up to 36 characters, including digits, letters, underscores (_), and hyphens (-). • A tag value is optional if the instance will be tagged. The value can be empty. The value can contain up to 43 characters, including digits, letters, underscores (_), periods (.), and hyphens (-). After an instance is created, you can view its tag details on the **Tags** tab. In addition, you can add, modify, and delete tags of an existing instance. For details, see **Managing GeminiDB Cassandra Instance Tags**.

Figure 3-7 Required duration configuration

Table 3-7 Required duration

Parameter	Description
Required Duration	The length of your subscription if you select **Yearly/Monthly** billing. Subscription lengths range from one month to three years.
Auto-renew	• This option is not selected by default. • If you select this option, the instance is automatically renewed based on the subscription duration.

Step 4 On the displayed page, confirm instance details.

- Yearly/Monthly
  - To modify the configurations, click **Previous**.
  - If no modification is required, read and agree to the service agreement, click **Pay Now**, and complete the payment.
- Pay-per-use
  - To modify the configurations, click **Previous**.
  - If no modification is required, read and agree to the service agreement and click **Submit**.

Step 5 On the **Instances** page, view and manage the created instance.

- It takes about 5 to 9 minutes to create an instance. During the process, the instance status is **Creating**.
- After the instance is created, its status becomes **Available**.

  You can click  in the upper right corner to refresh the instance status.
- An automated backup policy is enabled by default during instance creation. After the instance is created, a full backup is created.

----End

3.3 Instance Connection and Management

3.3.1 Connecting to a GeminiDB Cassandra Instance

GeminiDB Cassandra can be accessed through Data Admin Service (DAS), private networks, and public networks.

Figure 3-8 shows the process of connecting to a GeminiDB Cassandra instance.

Figure 3-8 Connection Methods

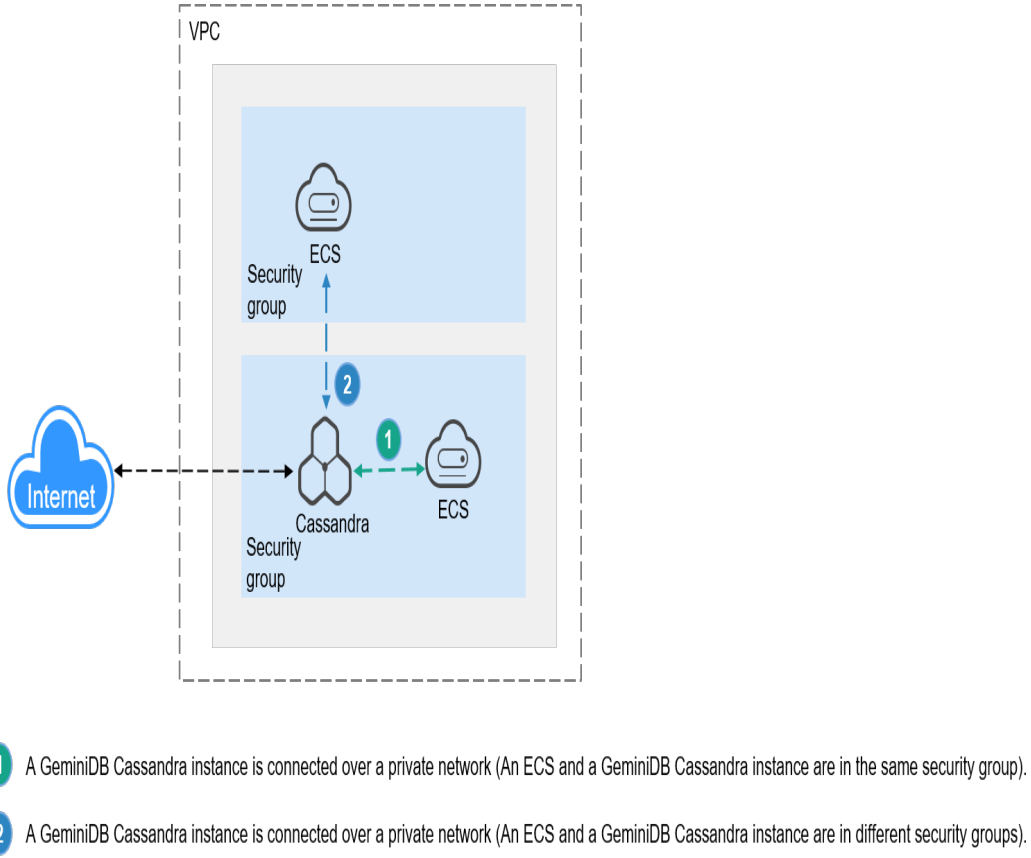


Table 3-8 Connection methods

Method	Scenario	Default Port	Description
DAS	You can log in to an instance on the console without using an IP address.	-	• Easy to use, secure, advanced, and intelligent • By default, you have the permissions required for remote login. It is recommended that you use the DAS service to log in to DB instances. DAS is secure and convenient.
Private network	Private IP addresses are provided by default. Your applications are deployed on an ECS that is in the same region and VPC as your instances.	8635	• High security and performance • If the ECS and GeminiDB Cassandra instance are in the same security group, they can communicate with each other by default. No security group rule needs to be configured. • If they are in different security groups, configure security group rules for them, separately. – Configure inbound rules of a security group for the GeminiDB Cassandra instance by following **Setting Security Group Rules for a GeminiDB Cassandra Instance**. – The default security group rule allows all outbound data packets, so you do not need to configure a security rule for the ECS. If not all access from the ECS is allowed, you need to configure an outbound rule for the ECS.
Public network	If you cannot access a DB instance through a private IP address, bind an EIP to the DB instance first and connect the ECS to the DB instance through the EIP.	8635	• Low security • For faster transmission and improved security, migrate your applications to an ECS that is in the same subnet as your instance and use a private IP address to access the instance.

Method	Scenario	Default Port	Description
Java	An example of connecting to an instance using Java is provided.	8635	-
Go	An example of connecting to an instance using Go is provided.	8635	-
Spark	An example of connecting to a GeminiDB Cassandra instance using Spark is provided.	8635	-

3.3.2 Connecting to a GeminiDB Cassandra Instance on the DAS Console

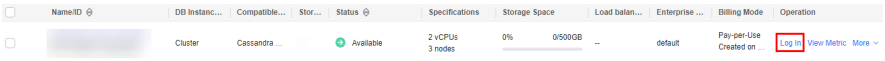
DAS enables you to manage DB instances from a web-based console, simplifying database management and improving efficiency. You can connect and manage instances through DAS. By default, you have the permission of remote login. DAS is secure and convenient for connecting to DB instances.

Procedure

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

Step 2 On the **Instance Management** page, locate the target DB instance and click **Log In** in the **Operation** column.

Figure 3-9 Logging in to the database



Alternatively, click the instance name on the **Instances** page. On the displayed **Basic Information** page, click **Log In** in the upper right corner.

Figure 3-10 Logging in to the database

Step 3 On the displayed login page, enter the administrator username and password and click **Log In**.

----End

FAQs

Question: What should I do if the DAS console cannot be redirected after I click **Log In** in the **Operation** column in the instance list or click **Log In** on the **Basic Information** page?

Solution: Set your browser to allow pop-ups and try again.

3.3.3 Connecting to a GeminiDB Cassandra Instance over a Private Network

You can install the Cassandra client on the ECS and access the instance through a private IP address.

Usage Notes

- The target instance must be in the same VPC and subnet as the ECS.
- The ECS must be in a security group that has access to the instances. For details, see **Setting Security Group Rules for a GeminiDB Cassandra Instance**.

Prerequisites

1. A GeminiDB Cassandra instance has been created and is running properly.
2. An ECS has been created. The following uses a Linux ECS as an example.

   For details, see **Purchasing an ECS** in *Getting Started with Elastic Cloud Server*.
3. Download and install the Cassandra client that matches the CPU type of the ECS.
   - If the CPU type is x86, download the **Cassandra client**.

Procedure

Step 1 Log in to ECS.

For details, see **Logging In to an ECS** in *Getting Started with Elastic Cloud Server*.

Step 2 Upload the Cassandra client installation package to the ECS.

Step 3 Run the following command to decompress the client installation package. The x86 client is used as an example.

unzip Cassandra_cqlsh_x86_64.zip

Step 4 Run the following command to grant the execute permission on all files:

chmod +x *

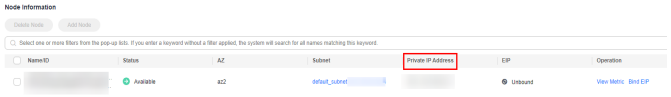
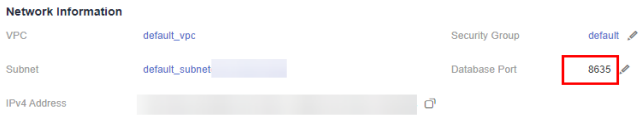
Step 5 Connect to the DB instance in the directory where the cqlsh tool is located.

./cqlsh <*DB_HOST*> <*DB_PORT*> **-u** <*DB_USER*>

Example:

./cqlsh 192.xx.xx.xx 8635 -u rwuser

Table 3-9 Description

Parameter	Description
<*DB_HOST*>	Private IP address of an instance to be connected. To obtain this IP address, go to the **Instance Management** page and click the target instance name. The IP address can be found in the **Private IP Address** field under **Node Information** on the **Basic Information** page. If the instance you purchased has multiple nodes, select the private IP address of any node. **Figure 3-11** Viewing the private IP address 
<*DB_PORT*>	Port number of the instance to be connected. The default port number is 8635. Replace it with the actual port number. Click the instance name to go to the **Basic Information** page and obtain the port number in the **Network Information** area. **Figure 3-12** Viewing the port number 
<*DB_USER*>	Database account. The default value is **rwuser**.

Step 6 Check the results. If the following information is displayed, the connection is successful.

```
rwuser@cqlsh>
```

----End

3.3.4 Connecting to a GeminiDB Cassandra Instance over a Public Network

You can use an ECS or local device to connect to a GeminiDB Cassandra instance over a public network.

This section describes how to use a Linux ECS to connect to a GeminiDB Cassandra instance over a public network.

Prerequisites

1. Bind an EIP to the GeminiDB Cassandra instance node and set security group rules. For details, see **Binding an EIP to a GeminiDB Cassandra Instance Node** and **Setting Security Group Rules for a GeminiDB Cassandra Instance**.
2. An ECS has been created. The following uses a Linux ECS as an example. For details, see **Purchasing an ECS** in *Getting Started with Elastic Cloud Server*.
3. Download and install the Cassandra client that matches the CPU type of the ECS.
   - If the CPU type is x86, download the **Cassandra client**.

Procedure

Step 1 Log in to the ECS. For details, see **Logging In to an ECS** in *Getting Started with Elastic Cloud Server*.

Step 2 Upload the Cassandra client installation package to the ECS.

Step 3 Run the following command to decompress the client installation package. The x86 client is used as an example.

unzip Cassandra_cqlsh_x86_64.zip

Step 4 Run the following command to grant the execute permission on all files:

chmod +x *

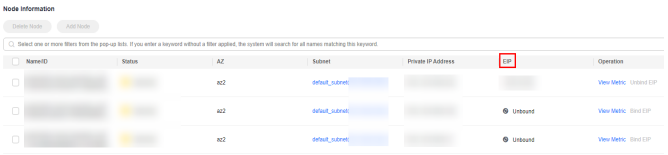
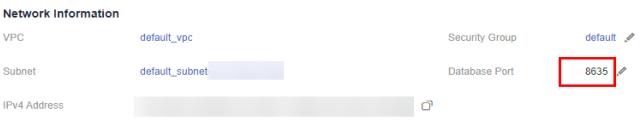
Step 5 Connect to the DB instance in the directory where the cqlsh tool is located.

./cqlsh <*DB_HOST*> <*DB_PORT*> **-u** <*DB_USER*>

Example:

./cqlsh 192.xx.xx.xx 8635 -u rwuser

Table 3-10 Description

Parameter	Description
<DB_HOST>	EIP bound to the instance to be connected. To obtain the EIP, go to the **Instances** page and click the target instance name. The EIP can be found in the **EIP** column in the **Node Information** area on the **Basic Information** page. If the instance you purchased has multiple nodes, select the EIP of any node. **Figure 3-13** Viewing the EIP  If no EIP is bound to the instance, bind an EIP to the instance by following **Binding an EIP to a GeminiDB Cassandra Instance Node** and then connect to the instance.
<DB_PORT>	Port number of the instance to be connected. The default port number is 8635. Replace it with the actual port number. Click the instance name to go to the **Basic Information** page and obtain the port number in the **Network Information** area. **Figure 3-14** Viewing the port number 
<DB_USER>	Database account. The default value is **rwuser**.

Step 6 Check the results. If the following information is displayed, the connection is successful.

```
rwuser@cqlsh>
```

----End

3.3.5 Connecting to a GeminiDB Cassandra Instance Using Java

This section describes how to use the Java to connect to a GeminiDB Cassandra instance.

Prerequisites

- A GeminiDB Cassandra instance has been created and is running properly. For details about how to create a GeminiDB Cassandra instance, see **Buying a GeminiDB Cassandra Instance**.
- For details about how to create an ECS, see **Purchasing an ECS** in *Getting Started with Elastic Cloud Server*.
- JDK has been installed on the ECS.
- DataStax 3.11.*x* is recommended. DataStax 4.*x* is not supported.

Procedure

Step 1 Obtain the private IP address and port number of the GeminiDB Cassandra instance.

For details about how to obtain the private IP address and port number, see **Viewing the IP Address and Port Number of a GeminiDB Cassandra Instance**.

Step 2 Log in to the ECS. For details, see **Logging in to an ECS** in *Elastic Cloud Server User Guide*.

Step 3 Edit the code for connecting to the GeminiDB Cassandra instance.

```
import com.datastax.driver.core.*;

Cluster cluster = null;
try {
    cluster = Cluster.builder()
            .addContactPoint("127.0.0.1")//Private IP address of the GeminiDB Cassandra instance obtained in
step 1
            .withPort(8635)         //Port number of the GeminiDB Cassandra instance obtained in step 1
            .build();
    Session session = cluster.connect();

    ResultSet rs = session.execute("select release_version from system.local");
    Row row = rs.one();
    System.out.println(row.getString("release_version"));
} finally {
    if (cluster != null) cluster.close();
}
```

Step 4 Run the sample code to check whether the result is normal.

----End

3.3.6 Connecting to a GeminiDB Cassandra Instance Using Go

This section describes how to connect to a GeminiDB Cassandra instance using Go.

Prerequisites

- A GeminiDB Cassandra instance has been created and is running normally. For details about how to create a GeminiDB Cassandra instance, see **Buying a GeminiDB Cassandra Instance**.
- For details about how to create an ECS, see **Purchasing an ECS** in *Getting Started with Elastic Cloud Server*.
- You have installed Go on the ECS. If you have not, download the **Go installation package**.

Procedure

Step 1 Obtain the private IP address and port number of the GeminiDB Cassandra instance.

For details about how to obtain the private IP address and port number, see **Viewing the IP Address and Port Number of a GeminiDB Cassandra Instance**.

Step 2 Log in to the ECS. For details, see **Logging in to an ECS** in *Elastic Cloud Server User Guide*.

Step 3 Edit the code for connecting to the GeminiDB Cassandra instance.

```
import (
    "os"
)
// Default LoadBalancingPolicy RoundRobinHostPolicy
cluster := gocql.NewCluster("192.168.1.1", "192.168.1.2", "192.168.1.3")

// There will be security risks if the username and password used for authentication are directly written into code. Store the username and password in ciphertext in the configuration file or environment variables.
// In this example, the username and password are stored in the environment variables. Before running this example, set environment variables EXAMPLE_USERNAME_ENV and EXAMPLE_PASSWORD_ENV as needed.
username = os.Getenv("EXAMPLE_USERNAME_ENV"),
password = os.Getenv("EXAMPLE_PASSWORD_ENV"),
cluster.Authenticator = gocql.PasswordAuthenticator{
        Username: username,
        Password: password
 }
cluster.Keyspace = "ks1"
// connect to the cluster
session, err := cluster.CreateSession()
if err != nil {
    log.Fatal(err)
}
defer session.Close()
```

Step 4 Run sample code to check whether the result is normal.

----End

Executing Write and Read Operations

Create a session query. Query parameters cannot be used in other statements and cannot be modified after the query starts.

Use Query.Exec if you need to read the query results after a query is executed:

```
err := session.Query(`INSERT INTO tweet (timeline, id, text) VALUES (?, ?, ?)`,
        "me", gocql.TimeUUID(), "hello world").WithContext(ctx).Exec()
```

Use Query.Scan to read one row:

```
err := session.Query(`SELECT id, text FROM tweet WHERE timeline = ? LIMIT 1`,
        "me").WithContext(ctx).Consistency(gocql.One).Scan(&id, &text)
```

Use Iter.Scanner to read multiple rows:

```
scanner := session.Query(`SELECT id, text FROM tweet WHERE timeline = ?`,
    "me").WithContext(ctx).Iter().Scanner()
 for scanner.Next() {
    var (
        id gocql.UUID
        text string
    )
    err = scanner.Scan(&id, &text)
```



```
    if err != nil {
        log.Fatal(err)
    }
    fmt.Println("Tweet:", id, text)
}
// scanner.Err() closes the iterator, so scanner nor iter should be used afterwards.
if err := scanner.Err(); err != nil {
    log.Fatal(err)
}
```

Executing Multiple Queries Concurrently

It is safe to share a session in multiple goroutines. You can execute concurrent queries using multiple worker goroutines.

```
results := make(chan error, 2)
go func() {
    results <- session.Query(`INSERT INTO tweet (timeline, id, text) VALUES (?, ?, ?)`,
        "me", gocql.TimeUUID(), "hello world 1").Exec()
}()
go func() {
    results <- session.Query(`INSERT INTO tweet (timeline, id, text) VALUES (?, ?, ?)`,
        "me", gocql.TimeUUID(), "hello world 2").Exec()
}()
```

3.3.7 Connecting to a GeminiDB Cassandra Instance Using Spark

This section describes how to use the Scala to connect to a GeminiDB Cassandra instance.

Prerequisites

- A GeminiDB Cassandra instance has been created and is running properly. For details about how to create a GeminiDB Cassandra instance, see **Buying a GeminiDB Cassandra Instance**.
- For details about how to create an ECS, see **Purchasing an ECS** in *Getting Started with Elastic Cloud Server*.
- The Spark environment has been installed on the ECS.

Procedure

Step 1 Obtain the private IP address and port number of the GeminiDB Cassandra instance.

For details about how to obtain the private IP address and port number, see **Viewing the IP Address and Port Number of a GeminiDB Cassandra Instance**.

Step 2 Log in to the ECS. For details, see **Logging in to an ECS** in *Elastic Cloud Server User Guide*.

Step 3 Edit the code for connecting to the GeminiDB Cassandra instance.

- If Spark 2.*x* is used to connect to the GeminiDB Cassandra instance, the recommended versions are as follows:

  Spark: 2.5.1

  Scala: 2.12

  spark-cassandra-connector: 2.5.1

The following is sample code:

```
  /**
   * There will be security risks if the username and password used for authentication are directly written into code. Store the username and password in ciphertext in the configuration file or environment variables.
   * In this example, the username and password are stored in the environment variables. Before running this example, set environment variables USERNAME_ENV and PASSWORD_ENV as needed.
   */
  val username: String = System.getenv().asScala.mkString("USERNAME_ENV")
  val password: String = System.getenv().asScala.mkString("PASSWORD_ENV")
  val sparkSession = SparkSession
    .builder()
    .appName("Spark Cassandra basic example")
    .master("local")
    .config("spark.cassandra.connection.host", "26.84.42.111")
    .config("spark.cassandra.connection.port", "9042")
    .config("spark.cassandra.auth.username", username)
    .config("spark.cassandra.auth.password", password)
    .getOrCreate()
```

If an error is reported during the connection, fix it by following **What Can I Do If Spark Failed to Connect to Cassandra?**.

- If Spark 3.*x* is used to connect to the GeminiDB Cassandra instance, the recommended versions include:

  Spark: 3.2.4

  Scala: 2.12.15

  Java: 1.8

  spark-cassandra-connector: 3.1.0

  a. You are advised to rewrite a CassandraConnectionFactory (change **loadBalancingPolicy** to **DefaultLoadBalancingPolicy**). The following is sample code:

```
package sample
import java.io.IOException
import java.net.{MalformedURLException, URL}
import java.nio.file.{Files, Paths}
import java.time.Duration

import com.datastax.bdp.spark.ContinuousPagingScanner
import com.datastax.dse.driver.api.core.DseProtocolVersion
import com.datastax.dse.driver.api.core.config.DseDriverOption
import com.datastax.oss.driver.api.core.CqlSession
import com.datastax.oss.driver.api.core.config.DefaultDriverOption._
import com.datastax.oss.driver.api.core.config.{DriverConfigLoader,
ProgrammaticDriverConfigLoaderBuilder => PDCLB}
import com.datastax.oss.driver.internal.core.connection.ExponentialReconnectionPolicy
import com.datastax.oss.driver.internal.core.loadbalancing.DefaultLoadBalancingPolicy
import com.datastax.oss.driver.internal.core.ssl.DefaultSslEngineFactory
import com.datastax.spark.connector.rdd.ReadConf
import com.datastax.spark.connector.util.{ConfigParameter, DeprecatedConfigParameter,
ReflectionUtil}
import org.apache.spark.{SparkConf, SparkEnv, SparkFiles}
import org.slf4j.LoggerFactory

import scala.jdk.CollectionConverters._
import com.datastax.spark.connector.cql.{CassandraConnectionFactory, CassandraConnector,
CassandraConnectorConf, CloudBasedContactInfo, DefaultScanner, IpBasedContactInfo,
LocalNodeFirstLoadBalancingPolicy, MultipleRetryPolicy, MultiplexingSchemaListener,
ProfileFileBasedContactInfo, Scanner}

class ConnectionFactory extends CassandraConnectionFactory {
  @transient
  lazy private val logger =
LoggerFactory.getLogger("com.datastax.spark.connector.cql.CassandraConnectionFactory")
```



```
 def connectorConfigBuilder(conf: CassandraConnectorConf, initBuilder: PDCLB) = {

   def basicProperties(builder: PDCLB): PDCLB = {
     val localCoreThreadCount = Math.max(1, Runtime.getRuntime.availableProcessors() - 1)
     builder
       .withInt(CONNECTION_POOL_LOCAL_SIZE,
conf.localConnectionsPerExecutor.getOrElse(localCoreThreadCount)) // moved from
CassandraConnector
       .withInt(CONNECTION_POOL_REMOTE_SIZE,
conf.remoteConnectionsPerExecutor.getOrElse(1)) // moved from CassandraConnector
       .withInt(CONNECTION_INIT_QUERY_TIMEOUT, conf.connectTimeoutMillis)
       .withDuration(CONTROL_CONNECTION_TIMEOUT,
Duration.ofMillis(conf.connectTimeoutMillis))
       .withDuration(METADATA_SCHEMA_REQUEST_TIMEOUT,
Duration.ofMillis(conf.connectTimeoutMillis))
       .withInt(REQUEST_TIMEOUT, conf.readTimeoutMillis)
       .withClass(RETRY_POLICY_CLASS, classOf[MultipleRetryPolicy])
       .withClass(RECONNECTION_POLICY_CLASS, classOf[ExponentialReconnectionPolicy])
       .withDuration(RECONNECTION_BASE_DELAY,
Duration.ofMillis(conf.minReconnectionDelayMillis))
       .withDuration(RECONNECTION_MAX_DELAY,
Duration.ofMillis(conf.maxReconnectionDelayMillis))
       .withInt(NETTY_ADMIN_SHUTDOWN_QUIET_PERIOD, conf.quietPeriodBeforeCloseMillis /
1000)
       .withInt(NETTY_ADMIN_SHUTDOWN_TIMEOUT, conf.timeoutBeforeCloseMillis / 1000)
       .withInt(NETTY_IO_SHUTDOWN_QUIET_PERIOD, conf.quietPeriodBeforeCloseMillis / 1000)
       .withInt(NETTY_IO_SHUTDOWN_TIMEOUT, conf.timeoutBeforeCloseMillis / 1000)
       .withBoolean(NETTY_DAEMON, true)
       .withBoolean(RESOLVE_CONTACT_POINTS, conf.resolveContactPoints)
       .withInt(MultipleRetryPolicy.MaxRetryCount, conf.queryRetryCount)
       .withDuration(DseDriverOption.CONTINUOUS_PAGING_TIMEOUT_FIRST_PAGE,
Duration.ofMillis(conf.readTimeoutMillis))
       .withDuration(DseDriverOption.CONTINUOUS_PAGING_TIMEOUT_OTHER_PAGES,
Duration.ofMillis(conf.readTimeoutMillis))
   }

   // compression option cannot be set to NONE (default)
   def compressionProperties(b: PDCLB): PDCLB =
     Option(conf.compression)
       .filter(_.toLowerCase != "none")
       .fold(b)(c => b.withString(PROTOCOL_COMPRESSION, c.toLowerCase))

   def localDCProperty(b: PDCLB): PDCLB =
     conf.localDC.map(b.withString(LOAD_BALANCING_LOCAL_DATACENTER, _)).getOrElse(b)

   // add ssl properties if ssl is enabled
   def ipBasedConnectionProperties(ipConf: IpBasedContactInfo) = (builder: PDCLB) => {
     builder
       .withStringList(CONTACT_POINTS, ipConf.hosts.map(h => s"${h.getHostString}:$
{h.getPort}").toList.asJava)
       .withClass(LOAD_BALANCING_POLICY_CLASS, classOf[DefaultLoadBalancingPolicy])

     def clientAuthEnabled(value: Option[String]) =
       if (ipConf.cassandraSSLConf.clientAuthEnabled) value else None

     if (ipConf.cassandraSSLConf.enabled) {
       Seq(
         SSL_TRUSTSTORE_PATH -> ipConf.cassandraSSLConf.trustStorePath,
         SSL_TRUSTSTORE_PASSWORD -> ipConf.cassandraSSLConf.trustStorePassword,
         SSL_KEYSTORE_PATH -> clientAuthEnabled(ipConf.cassandraSSLConf.keyStorePath),
         SSL_KEYSTORE_PASSWORD ->
clientAuthEnabled(ipConf.cassandraSSLConf.keyStorePassword))
         .foldLeft(builder) { case (b, (name, value)) =>
           value.map(b.withString(name, _)).getOrElse(b)
         }
         .withClass(SSL_ENGINE_FACTORY_CLASS, classOf[DefaultSslEngineFactory])
         .withStringList(SSL_CIPHER_SUITES,
ipConf.cassandraSSLConf.enabledAlgorithms.toList.asJava)
         .withBoolean(SSL_HOSTNAME_VALIDATION, false) // TODO: this needs to be
```



```
configurable by users. Set to false for our integration tests
      } else {
        builder
      }
    }

    val universalProperties: Seq[PDCLB => PDCLB] =
      Seq( basicProperties, compressionProperties, localDCProperty)

    val appliedProperties: Seq[PDCLB => PDCLB] = conf.contactInfo match {
      case ipConf: IpBasedContactInfo => universalProperties :+
ipBasedConnectionProperties(ipConf)
      case other => universalProperties
    }

    appliedProperties.foldLeft(initBuilder){ case (builder, properties) => properties(builder)}
  }

  /** Creates and configures native Cassandra connection */
  override def createSession(conf: CassandraConnectorConf): CqlSession = {
    val configLoaderBuilder = DriverConfigLoader.programmaticBuilder()
    val configLoader = connectorConfigBuilder(conf, configLoaderBuilder).build()

    val initialBuilder = CqlSession.builder()

    val builderWithContactInfo =  conf.contactInfo match {
      case ipConf: IpBasedContactInfo =>
        ipConf.authConf.authProvider.fold(initialBuilder)(initialBuilder.withAuthProvider)
          .withConfigLoader(configLoader)
      case CloudBasedContactInfo(path, authConf) =>
        authConf.authProvider.fold(initialBuilder)(initialBuilder.withAuthProvider)
          .withCloudSecureConnectBundle(maybeGetLocalFile(path))
          .withConfigLoader(configLoader)
      case ProfileFileBasedContactInfo(path) =>
        //Ignore all programmatic config for now ... //todo maybe allow programmatic config here
by changing the profile?
        logger.warn(s"Ignoring all programmatic configuration, only using configuration from
$path")
        initialBuilder.withConfigLoader(DriverConfigLoader.fromUrl(maybeGetLocalFile(path)))
    }

    val appName = Option(SparkEnv.get).map(env => env.conf.getAppId).getOrElse("NoAppID")
    builderWithContactInfo
      .withApplicationName(s"Spark-Cassandra-Connector-$appName")
      .withSchemaChangeListener(new MultiplexingSchemaListener())
      .build()
  }

  /**
   * Checks the Spark Temp work directory for the file in question, returning
   * it if exists, returning a generic URL from the string if not
   */
  def maybeGetLocalFile(path: String): URL = {
    val localPath = Paths.get(SparkFiles.get(path))
    if (Files.exists(localPath)) {
      logger.info(s"Found the $path locally at $localPath, using this local file.")
      localPath.toUri.toURL
    } else {
      try {
        new URL(path)
      } catch {
        case e: MalformedURLException =>
          throw new IOException(s"The provided path $path is not a valid URL nor an existing
locally path. Provide an " +
            s"URL accessible to all executors or a path existing on all executors (you may use
`spark.files` to " +
            s"distribute a file to each executor).", e)
      }
    }
```



```
  }

  def continuousPagingEnabled(session: CqlSession): Boolean = {
    val confEnabled =
SparkEnv.get.conf.getBoolean(CassandraConnectionFactory.continuousPagingParam.name,
CassandraConnectionFactory.continuousPagingParam.default)
    val pv = session.getContext.getProtocolVersion
    if (pv.getCode > DseProtocolVersion.DSE_V1.getCode && confEnabled) {
      logger.debug(s"Scan Method Being Set to Continuous Paging")
      true
    } else {
      logger.debug(s"Scan Mode Disabled or Connecting to Non-DSE Cassandra Cluster")
      false
    }
  }

  override def getScanner(
                          readConf: ReadConf,
                          connConf: CassandraConnectorConf,
                          columnNames: scala.IndexedSeq[String]): Scanner = {

    val isContinuousPagingEnabled =
      new CassandraConnector(connConf).withSessionDo { continuousPagingEnabled }

    if (isContinuousPagingEnabled) {
      logger.debug("Using ContinousPagingScanner")
      ContinuousPagingScanner(readConf, connConf, columnNames)
    } else {
      logger.debug("Not Connected to DSE 5.1 or Greater Falling back to Non-Continuous
Paging")
      new DefaultScanner(readConf, connConf, columnNames)
    }
  }
}
```

b. The code for connecting to the GeminiDB Cassandra instance is as follows:

```
/**
    * There will be security risks if the username and password used for authentication are
directly written into code. Store the username and password in ciphertext in the configuration
file or environment variables.
    * In this example, the username and password are stored in the environment variables.
Before running this example, set environment variables USERNAME_ENV and PASSWORD_ENV
as needed.
 */
val username: String = System.getenv().asScala.mkString("USERNAME_ENV")
val password: String = System.getenv().asScala.mkString("PASSWORD_ENV")
val sparkSession = SparkSession
  .builder()
  .appName("Spark Cassandra basic example")
  .master("local")
  .config("spark.cassandra.connection.host", host)
  .config("spark.cassandra.connection.port", port)
  .config("spark.cassandra.auth.username", username)
  .config("spark.cassandra.auth.password", password)
Set .config("spark.cassandra.connection.factory", "sample.ConnectionFactory") //Set
ConnectionFactory as needed.
  .getOrCreate()
```

Step 4 Run the sample code to check whether the instance is connected.

----End

3.3.8 Connection Information Management

3.3.8.1 Setting Security Group Rules for a GeminiDB Cassandra Instance

A security group is a collection of access control rules for ECSs and GeminiDB Cassandra instances that have the same security protection requirements and are mutually trusted in a VPC.

To ensure database security and reliability, configure security group rules to allow specific IP addresses and ports to access the GeminiDB Cassandra instances.

This section describes how to configure security group rules when you connect to a GeminiDB Cassandra instance over private and public networks.

Usage Notes

- Each account can create up to 500 security group rules by default.
- Too many security group rules will increase the first packet latency, so a maximum of 50 rules for each security group is recommended.
- One security group can be associated with only one GeminiDB Cassandra instance.
- For details about security group rules, see **Table 3-11**.

Table 3-11 Parameter description

Scenario	Description
Connecting to an instance over a private network	Check whether the ECS and GeminiDB Cassandra instance are in the same security group: • If yes, no security group rules need to be configured. • If no, configure security group rules for them, respectively. – GeminiDB Cassandra instance: Configure inbound rules for its security group. For details, see **Procedure**. – ECS: The default security group rule allows all outbound data packets, so you do not need to configure a security rule for the ECS. If not all outbound traffic is allowed in the security group, configure an outbound rule for the ECS.
Connecting to an instance over a public network	Add inbound rules for the security group associated with the GeminiDB Cassandra instance. For details, see **Procedure**.

Procedure

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

Step 2 On the **Instances** page, locate the instance that you want to configure security group rules for and click its name.

Step 3 Configure security group rules.

Method 1

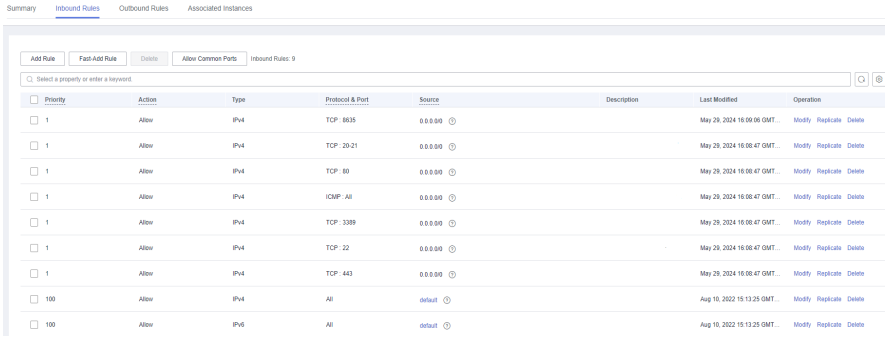
In the **Network Information** area on the **Basic Information** page, click the name of security group.

Figure 3-15 Security group

Step 4 Add an inbound rule.

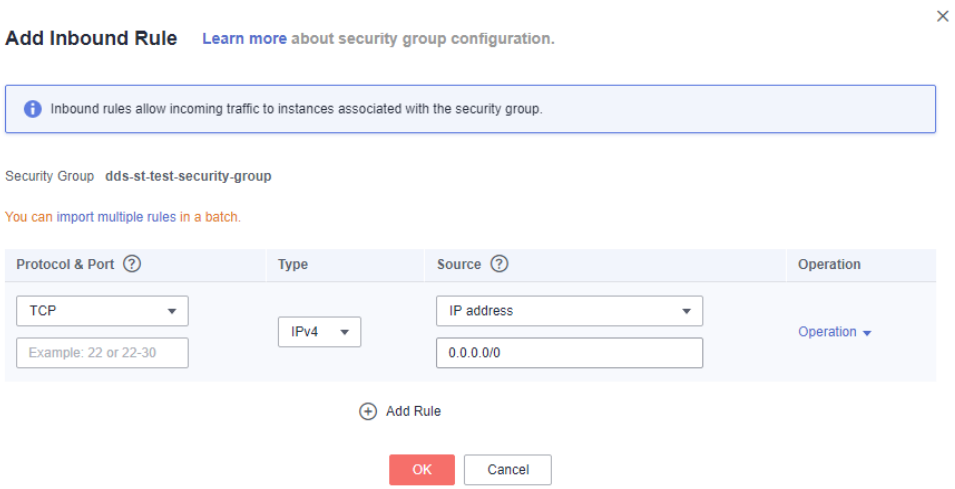
1. Click the **Inbound Rules** tab.

Figure 3-16 Inbound rules



2. Click **Add Rule**. The **Add Inbound Rule** dialog box is displayed.

Figure 3-17 Adding a rule



3. In the displayed **Add Rule** dialog box, set required parameters.

Table 3-12 Inbound rule settings

Parameter	Description	Example Value
Protocol & Port	– Network protocol. Currently, GeminiDB Cassandra instances can be accessed only over TCP. – **Port**: The port or port range that allows the access to the ECS. Range: 1 to 65535	TCP
Type	IP address type. This parameter is available only after the IPv6 function is enabled. – IPv4 – IPv6	IPv4
Source	Source address. It can be a single IP address, an IP address group, or a security group to allow access from the IP address or instances in the security group. Example: – Single IP address: xxx.xxx.xxx.xxx/32 (IPv4) – Subnet: xxx.xxx.xxx.0/24 – All IP addresses: 0.0.0.0/0 – sg-abc (security group)	0.0.0.0/0
Description	(Optional) Provides supplementary information about the security group rule. The description can contain up to 255 characters and cannot contain angle brackets (<>).	-

Step 5 Click **OK**.

----End

3.3.8.2 Binding an EIP to a GeminiDB Cassandra Instance Node

The Elastic IP service provides independent public IP addresses and bandwidth for public access. After you create a GeminiDB Cassandra instance, you can bind an EIP to its node to allow external access. If later you want to prohibit external access, you can also unbind the EIP.

Usage Notes

- This function is now in OBT. To use it, choose **Service Tickets > Create Service Ticket** in the upper right corner of the console and contact the customer service.
- To change the EIP that has been bound to a node, unbind it from the node first.

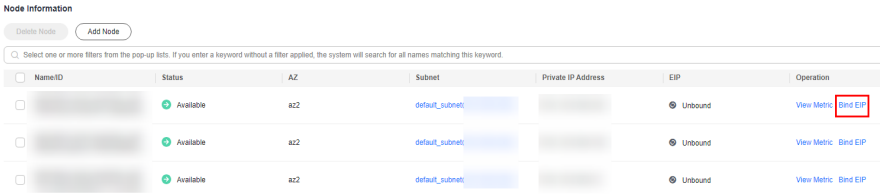
Binding an EIP

Step 1 Log in to the Huawei Cloud console.

Step 2 On the **Instances** page, locate the GeminiDB Cassandra instance that you want to bind an EIP to and click its name.

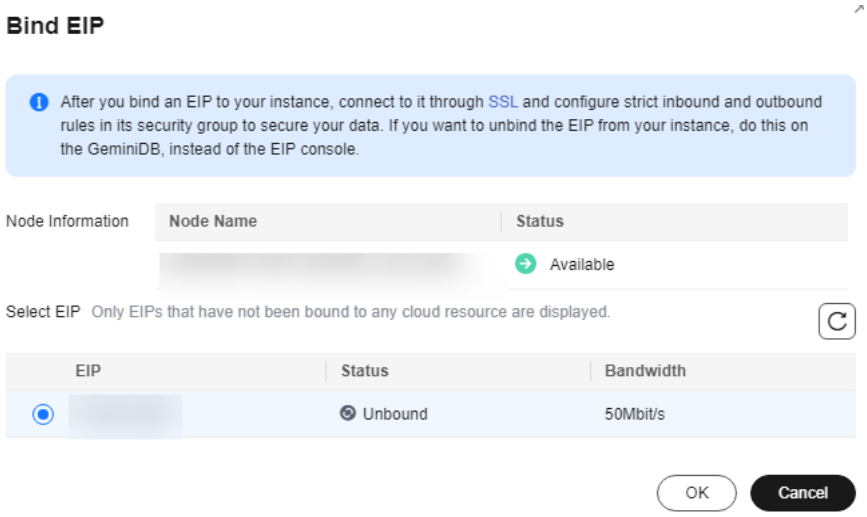
Step 3 On the **Basic Information** page, in the **Node Information** area, locate the target node and click **Bind EIP** in the **Operation** column.

Figure 3-18 Binding an EIP



Step 4 In the displayed dialog box, select the required EIP and click **Yes**. If no available EIPs are displayed, click **View EIP** and create an EIP on the VPC console.

Figure 3-19 Selecting an EIP



Step 5 In the **EIP** column, view the EIP that is successfully bound.

To unbind the EIP from the instance, see **Unbinding an EIP**.

----End

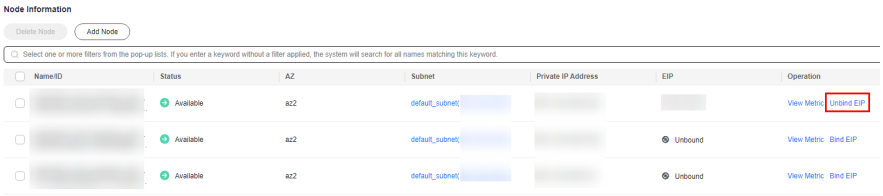
Unbinding an EIP

Step 1 Log in to the Huawei Cloud console.

Step 2 On the **Instances** page, locate the GeminiDB Cassandra instance that you want to unbind an EIP from and click its name.

Step 3 On the **Basic Information** page, in the **Node Information** area, locate the target node and click **Unbind EIP** in the **Operation** column.

Figure 3-20 Unbinding an EIP



Step 4 In the displayed dialog box, click **Yes**.

To bind an EIP to the instance again, see **Binding an EIP**.

----End

3.3.8.3 Viewing the IP Address and Port Number of a GeminiDB Cassandra Instance

This section describes how to query the IP address and port number of a GeminiDB Cassandra instance on the management console.

Procedure

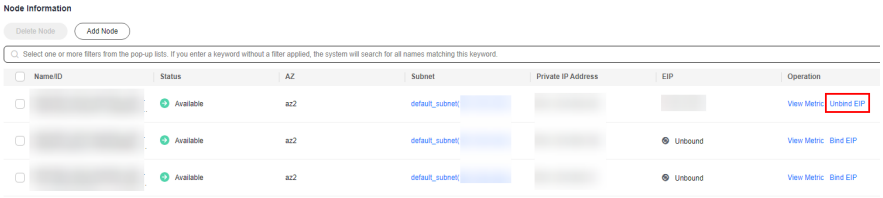
Step 1 Log in to the Huawei Cloud console.

Step 2 On the **Instances** page, locate the instance whose IP address and port you want to view and click its name.

Method 1

In the **Node Information** area on the **Basic Information** page, view the private IP address or EIP of each node in the instance.

Figure 3-21 Obtaining IP addresses



In the **Network Information** area, view the port number of the instance. The default port is 8635.

Figure 3-22 Viewing the port number

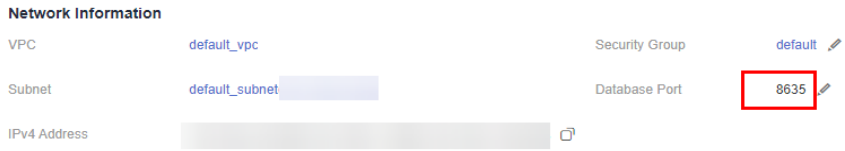
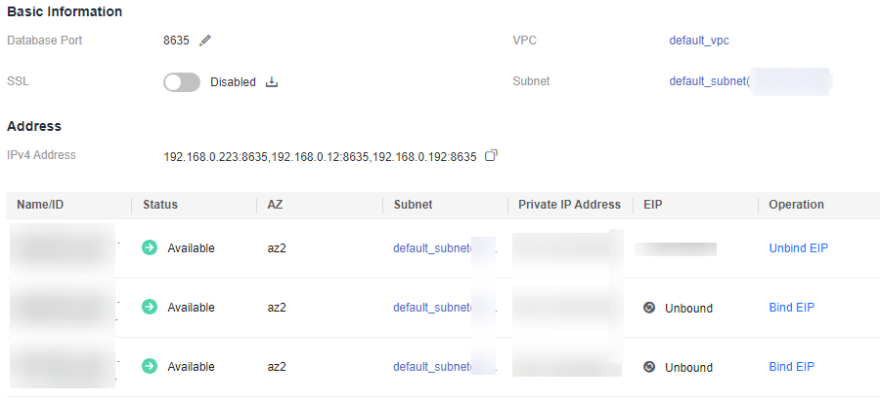


Figure 3-23 Viewing the IP addresses and port number



----End

3.4 Importing and Exporting Data by Running COPY

COPY is one of cqlsh commands. It includes **COPY TO** and **COPY FROM**. They are used to copy data to and from Cassandra.

COPY TO can export data from a table to a CSV, Parquet, or ORC file.

- If the exported file is in CSV format, it needs to be written into the target file by row, and fields are separated by delimiters.
- If no field name is specified, all fields are exported.
- To skip some fields, specify a field list.

COPY FROM allows you to import data from a CSV file to an existing table.

- The source file is imported by row.
- All rows in the dataset must contain the same number of fields, and the PRIMARY KEY field must have a value. During the import, the PRIMARY KEY field will be verified and the existing records are updated.
- If HEADER is set to **False** and no field name is specified, fields are imported in a specified order. After field names are specified, the fields are imported in sequence. The missing and empty fields are set to null.
- The source file can only have fewer fields than the target table.
- When only COPY FROM is used to import data, the number of rows in a dataset cannot exceed 2 million.

Precautions

- Import and export data during off-peak hours to minimize the impacts on your services.
- Obtain the latest binary package by following **Connecting to a GeminiDB Cassandra Instance over a Private Network**.

COPY Syntax

- **COPY TO**

  COPY *table_name* [(*column_list*)] TO *'file_name'* [, *'file2_name'*, ...] | STDOUT [WITH option = *'value'* [AND ...]]

- **COPY FROM**

  COPY *table_name* [(*column_list*)] FROM *'file_name'* [, *'file2_name'*, ...] | STDIN [WITH option = *'value'* [AND ...]]

NOTE

COPY supports one or more comma-separated file names or a list of Python glob expressions.

For some common syntax symbols in the COPY command, see **Table 3-13**.

Table 3-13 Symbol conventions

Symbol	Description
Uppercase letters	Text keyword.
Lowercase letters	A variable, which needs to be replaced with a user-defined value.
Italic	(Optional) Enclose optional command parameters in square brackets ([]). Do not enter only square brackets.
( )	Group. Parentheses (()) indicate the group to be selected. Do not input only brackets.
|	Or. Use vertical bars (|) to separate elements. You can input any element. Do not enter only vertical bars.
...	Repeatable. The ellipsis (...) indicates that you can repeat syntax elements multiple times as required.
'*Literal string*'	The single quotation marks (') must contain the character string in the CQL statement. Use single quotation marks to keep uppercase letters.
{ key : value }	The map set. Include a map set or key-value pair in braces ({}). Separate keys and values with colons.
<*datatype1*,*datatype2*>	Set, list, map, or tuple of an ordered list. Angle brackets (< >) contain data types in collections, lists, maps, or tuples. Data types are separated by commas (,).
cql_statement;	End a CQL statement. Semicolons (;) end all CQL statements.
[--]	Use two hyphens (--) to separate command line options from command arguments. This syntax is useful when parameters may be mistaken for command arguments.
' <*schema*> ... </*schema*> '	Search CQL only; single quotation marks (') enclose the entire XML schema declaration.

Symbol	Description
@xml_entity='xml_entity_type'	Search CQL only; identify entities and literal values to overwrite XML elements in schemas and solrConfig files.

COPY Usage Suggestions

Table 3-14 Description

Command	Parameter	Description	Default Value	Applicability
TO/FROM	DELIMITER	A single character used to separate fields.	English comma,	-
TO/FROM	QUOTE	A single character that contains a field value.	"	-
TO/FROM	ESCAPE	Escapes a single character using the QUOTE character.	\	-
TO/FROM	HEADER	Boolean value (true | false), indicating the name of the column in the first row. True matches the field name with the imported column name and inserts the column name into the first row of the exported data.	FALSE	-
TO/FROM	NULL	Filled value of the field whose query result is empty. You can set this parameter as required.	Empty string ()	-

Command	Parameter	Description	Default Value	Applicability
TO/FROM	DATETIMEFORMAT	Time format for reading or writing CSV time data. The timestamp is in the strftime format. If this parameter is not set, the default value is the value of **time_format** in the **cqlshrc** file. Default format: %Y-%m- %d %H: %M: %S %z.	%Y-%m-%d %H:%M:%S%z	-
TO/FROM	MAXATTEMPTS	Maximum number of retry times when an error occurs.	5	-
TO/FROM	REPORTFREQUENCY	Frequency of displaying the status, in seconds.	0.25	-
TO/FROM	DECIMALSEP	Delimiter character for decimal values.	English full stop.	-
TO/FROM	THOUSANDSSEP	Separator of a thousand array.	None	-
TO/FROM	BOOLSTYLE	Boolean values indicate True and False. The value is case-insensitive. For example, the values **yes** and **no** have the same effect as values **YES** and **NO**.	True,False	-

Command	Parameter	Description	Default Value	Applicability
TO/FROM	NUMPROCESSES	Number of working processes.	16	The default value of this parameter is the number of kernels on the computer minus one. There is no maximum value for this parameter. You can run the **dstat and dstat -lvrn 10** commands to check the CPU idle time. If the CPU idle time exists, use the default number of working processes. You can increase the number of processes while observing the CPU usage of the instance. It is recommended that the CPU usage be less than or equal to 60%. If the CPU usage of the executor is idle and the CPU usage of the instance exceeds the recommended value, expand the capacity to further improve the performance.

Command	Parameter	Description	Default Value	Applicability
TO/FROM	CONFIGFILE	Specifies a cqlshrc configuration file to set the WITH option. **NOTE** Command line options always overwrite the cqlshrc file.	None, user-defined	-
TO/FROM	RATEFILE	Prints the output statistics to this file.	None, user-defined	You are advised to add this parameter when exporting data to improve statistics efficiency.
TO/FROM	ORIGIN	Check whether the database to be imported or exported is an open-source Cassandra database. ● If the open-source Cassandra is used, the value is **True**. ● If GeminiDB Cassandra is used, the value is **False**.	False	-
FROM	CHUNKSIZE	The block size is passed to the worker process.	5000	This parameter specifies the number of rows sent from the Feeder process (reading data from files) to the worker process. Depending on the average row size of the dataset, it may be advantageous to increase the value of this parameter.

Command	Parameter	Description	Default Value	Applicability
FROM	INGESTRATE	Approximate import rate per second.	100000	**INGESTRATE** indicates the rate (in rows) at which the feeder process sends data to the worker process per second. Generally, you do not need to change the value unless the rate is too high and needs to be limited.

Command	Parameter	Description	Default Value	Applicability
FROM	MAXBATCHSIZE	Maximum size of a batch file to be imported.	20	The value of this parameter can be as large as possible but cannot exceed the upper limit. • MAXBATCHSIZE x The size of a single row < batch_size_fail_threshold_in_kb. • If the batch size is too large, an alarm will be reported and rejected. • Set the following parameters in **cassandra.yaml**: batch_size_warn_threshold_in_kb (The current value is 5.) batch_size_fail_threshold_in_kb (The current value is 50.)

Command	Parameter	Description	Default Value	Applicability
FROM	MINBATCHSIZE	Minimum size of a batch import file.	2	For each chunk, the worker process writes data in batches based on the minimum batch size. The value may need to be adjusted based on the block size, number of nodes in the cluster, and number of vnodes on each node. The larger the chunk size is, the larger the value will be.
FROM	MAXROWS	Maximum number of rows. The value **-1** indicates that there is no upper limit.	-1	-
FROM	SKIPROWS	Number of rows to skip.	0	-
FROM	SKIPCOLS	A comma-separated list of column names to skip.	None, user-defined	-
FROM	MAXPARSEERRORS	Maximum number of global parsing errors. The value -1 indicates that there is no upper limit.	-1	-
FROM	MAXINSERTERRORS	Maximum number of global insertion errors. The value -1 indicates that there is no upper limit.	-1	-

Command	Parameter	Description	Default Value	Applicability
FROM	ERRFILE	A file that stores all rows that are not imported. If no value is set, the information is stored in **import_ ks _ table .err**, where **ks** is the key space and **table** is the table name.	import_ ks _ table .err	-
FROM	TTL	The time to live is in seconds. By default, data does not expire.	3600	-
TO	ENCODING	Output character string type.	UTF-8	-
TO	PAGESIZE	Size of the page for obtaining results.	1000	Size of the result page. The value is an integer. The default value is **1000**. The larger the page size, the longer the value of pagetimeout. If the data volume in a single row is large, set this parameter to a smaller value. If the data volume in a single row is small, set this parameter to a larger value. The best effect of this value depends on the local batch write capability of the executor. If the local batch write capability is strong.

Command	Parameter	Description	Default Value	Applicability
TO	PAGETIMEOUT	The page times out to obtain the result.	10	The value is an integer, indicating the timeout interval for obtaining each page. The unit is second. The default value is 10 seconds. ● For a large page size or a large partition, increase the value of this parameter. ● If a timeout occurs, increase the value of this parameter. ● If the server times out, an exponential backoff policy is automatically initiated to prevent the server from being further overloaded, so you may notice the delay. The driver also generates a timeout. In this case, the driver does not know whether the server discards the request or returns the

Command	Parameter	Description	Default Value	Applicability
				result later. There is a low probability that data may be lost or duplicated. Increasing the value of this parameter is helpful in preventing driver build timeouts.
TO	BEGINTOKEN	Minimum token for exporting data.	None, user-defined	The value is a string, indicating the minimum token to be considered during data export. Records with smaller tokens will not be exported. The default value is empty, indicating that there is no minimum token.

Command	Parameter	Description	Default Value	Applicability
TO	ENDTOKEN	Maximum token used to export data.	None, user-defined	The value is a string, indicating the maximum number of tokens to be considered during data export. Records with larger tokens will not be exported. This parameter is left empty by default, indicating that there is no maximum token.
TO	MAXREQUESTS	Maximum number of requests that can be processed concurrently by each worker.	6	The value of this parameter is an integer, indicating the maximum number of running requests that can be processed by each working process. Total degree of parallelism during data export = Number of working processes x Value of this parameter. Default value: 6 Each request will export data for the entire token range.

Command	Parameter	Description	Default Value	Applicability
TO	MAXOUTPUTSIZE	Maximum size of an output file, in lines. After this parameter is set, the output file is split into multiple segments when the size of the output file exceeds the value of this parameter. The value -1 indicates that there is no upper limit.	-1	The value of this parameter is an integer, indicating the maximum size of an output file in the unit of lines. If the value of this parameter is exceeded, the output file is split into multiple segments. The default value is -1, indicating that there is no limit on the maximum value. Therefore, the file is the only output file. This parameter can be used together with MAXFILESIZE.

Command	Parameter	Description	Default Value	Applicability
TO	MAXFILESIZE	Maximum size of an output file, in KB. After this parameter is set, the output file is split into multiple segments when the size of the output file exceeds the value of this parameter.	None, user-defined	The value of this parameter is an integer, indicating the maximum size of an output file in bytes. The final file size is close to the value of this parameter. If the file size exceeds this value, the output file is split into multiple segments. The default value is -1, indicating that there is no limit on the maximum value. Therefore, the file is the only output file. This parameter can be used together with MAXOUTPUTSIZE.
TO	dataformats	Output file format. Currently, this parameter can only be set to json.	None, user-defined	-
TO	DATATYPE	The file format can be Parquet or ORC.	None, user-defined	-
TO	RESULTFILE	The exported file containing detailed results.	None, user-defined	You are advised to add this parameter when exporting data to improve statistics efficiency.
TO	wherecondition	Export condition specified during the export.	None, user-defined	-

Procedure

The following uses an example to describe how to preconfigure data, export data, and import data.

Step 1 Pre-configuring Data

1. Create a keyspace.
```
CREATE KEYSPACE cycling WITH replication = {'class': 'SimpleStrategy', 'replication_factor': 3};
```
2. Create a table.
```
CREATE TABLE cycling.cyclist_name (
   id UUID PRIMARY KEY,
   lastname text,
   firstname text
);
```
3. Insert a data record.
```
INSERT INTO cycling.cyclist_name (id, lastname, firstname) VALUES (5b6962dd-3f90-4c93-8f61-eabfa4a803e2, 'VOS','Marianne');
INSERT INTO cycling.cyclist_name (id, lastname, firstname) VALUES (e7cd5752-bc0d-4157-a80f-7523add8dbcd, 'VAN DER BREGGEN','Anna');
INSERT INTO cycling.cyclist_name (id, lastname, firstname) VALUES (e7ae5cf3-d358-4d99-b900-85902fda9bb0, 'FRAME','Alex');
INSERT INTO cycling.cyclist_name (id, lastname, firstname) VALUES (220844bf-4860-49d6-9a4b-6b5d3a79cbfb, 'TIRALONGO','Paolo');
INSERT INTO cycling.cyclist_name (id, lastname, firstname) VALUES (6ab09bec-e68e-48d9-a5f8-97e6fb4c9b47, 'KRUIKSWIJK','Steven');
INSERT INTO cycling.cyclist_name (id, lastname, firstname) VALUES (fb372533-eb95-4bb4-8685-6ef61e994caa, 'MATTHEWS', 'Michael');
```

Step 2 Exports data from and imports data to the **cyclist_name** table.

1. Export the **id** and **lastname** columns from the **cyclist_name** table to a CSV file.
```
COPY cycling.cyclist_name (id,lastname)  TO '../cyclist_lastname.csv' WITH HEADER = TRUE;
```


Figure 3-24 Exported successfully

```
Using 15 child processes

Starting copy of cycling.cyclist_name with columns [id, lastname].
Processed: 6 rows; Rate:      41 rows/s; Avg. rate:      41 rows/s
6 rows exported in 0.201 seconds.
Processed: 6 rows; Rate:      20 rows/s; Avg. rate:      40 rows/s
Results                 : success
Total operation         : 6
Total operation time    : 0.201 seconds
Operation rate          : 40.468307430069224 rows/s
Total   ranges          : 25
Success ranges          : 25
Failed  ranges          : 0
Num processes           : 15
Max attempts            : 5


Ranges Results:
ranges                                          result     exported rows
(-7, 683212743470724096)                        success    1
(3074457345618258593, 3757670089088982528)      success    1
(2220441416279853312, 3074457345618258593)      success    1
(-6148914691236517207, -5465701947765792768)    success    1
(-854015929338405120, -7)                       success    1
(-2391244602147534336, -1537228672809129307)    success    1
```

After the preceding command is executed successfully, the **cyclist_lastname.csv** file is created in the upper-level directory of the current directory. If the file already exists, it will be overwritten.

2. Export the **id** and **first name** columns from the **cyclist_name** table to another CSV file.

```
COPY cycling.cyclist_name (id,firstname)  TO '../cyclist_firstname.csv' WITH HEADER = TRUE;
```


Figure 3-25 Exported successfully

```
Using 15 child processes

Starting copy of cycling.cyclist_name with columns [id, firstname].
Processed: 6 rows; Rate:      67 rows/s; Avg. rate:      67 rows/s
6 rows exported in 0.134 seconds.
Processed: 6 rows; Rate:      33 rows/s; Avg. rate:      67 rows/s
Results                   : success
Total operation           : 6
Total operation time      : 0.134 seconds
Operation rate            : 66.57325993275435 rows/s
Total   ranges            : 25
Success ranges            : 25
Failed  ranges            : 0
Num processes             : 15
Max attempts              : 5


Ranges Results:
ranges                                          result     exported rows
(-854015929338405120, -7)                       success    1
(-7, 683212743470724096)                        success    1
(3074457345618258593, 3757670089088982528)      success    1
(-6148914691236517207, -5465701947765792768)    success    1
(2220441416279853312, 3074457345618258593)      success    1
(-2391244602147534336, -1537228672809129307)    success    1
```

After the preceding command is executed successfully, the **cyclist_firstname.csv** file is created in the upper-level directory of the current directory. If the file already exists, it will be overwritten.

3. Delete data from the **cyclist_name** table. To ensure data security, the TRUNCATE command is not supported.
   ```
   DELETE FROM cycling.cyclist_name WHERE id = 'fb372533-eb95-4bb4-8685-6ef61e994caa';
   ```
4. No data exists in the table.
   ```
   SELECT * FROM cycling.cyclist_name ;
   ```

Figure 3-26 Querying data

```
cqlsh> SELECT * FROM cycling.cyclist_name ;

 id | firstname | lastname
----+-----------+----------
```

5. Import the **cyclist_firstname.csv** file.
   ```
   COPY cycling.cyclist_name (id,firstname) FROM '../cyclist_firstname.csv' WITH HEADER = TRUE;
   ```

Figure 3-27 Import succeeded

```
cqlsh> COPY cycling.cyclist_name (id,firstname) FROM '../cyclist_firstname.csv' WITH HEADER = TRUE ;
Using 15 child processes

Starting copy of cycling.cyclist_name with columns [id, firstname].
Processed: 6 rows; Rate:      11 rows/s; Avg. rate:      15 rows/s
6 rows imported from 1 files in 0.387 seconds (0 skipped).
```


6. Verify the imported data.
```
SELECT * FROM cycling.cyclist_name;
```

Figure 3-28 Import succeeded

```
cqlsh> SELECT * FROM cycling.cyclist_name ;

 id                                   | firstname | lastname
--------------------------------------+-----------+----------
 e7ae5cf3-d358-4d99-b900-85902fda9bb0 |      Alex |     null
 fb372533-eb95-4bb4-8685-6ef61e994caa |   Michael |     null
 5b6962dd-3f90-4c93-8f61-eabfa4a803e2 |  Marianne |     null
 220844bf-4860-49d6-9a4b-6b5d3a79cbfb |     Paolo |     null
 6ab09bec-e68e-48d9-a5f8-97e6fb4c9b47 |    Steven |     null
 e7cd5752-bc0d-4157-a80f-7523add8dbcd |      Anna |     null
```

7. Import the **cyclist_lastname.csv** file.
```
COPY cycling.cyclist_name (id,lastname) FROM '../cyclist_lastname.csv' WITH HEADER = TRUE;
```

Figure 3-29 Importing data

```
Using 15 child processes

Starting copy of cycling.cyclist_name with columns [id, lastname].
Processed: 6 rows; Rate:      11 rows/s; Avg. rate:      16 rows/s
6 rows imported from 1 files in 0.378 seconds (0 skipped).
```

8. Check whether the data is updated.
```
SELECT * FROM cycling.cyclist_name;
```

The query result is displayed,

Figure 3-30 Import succeeded

```
cqlsh> SELECT * FROM cycling.cyclist_name ;

 id                                   | firstname | lastname
--------------------------------------+-----------+-----------------
 e7ae5cf3-d358-4d99-b900-85902fda9bb0 |      Alex |           FRAME
 fb372533-eb95-4bb4-8685-6ef61e994caa |   Michael |        MATTHEWS
 5b6962dd-3f90-4c93-8f61-eabfa4a803e2 |  Marianne |             VOS
 220844bf-4860-49d6-9a4b-6b5d3a79cbfb |     Paolo |        TIRALONGO
 6ab09bec-e68e-48d9-a5f8-97e6fb4c9b47 |    Steven |       KRUIKSWIJK
 e7cd5752-bc0d-4157-a80f-7523add8dbcd |      Anna | VAN DER BREGGEN

(6 rows)
```

----End

3.5 Instance Lifecycle Management

3.5.1 Restarting Instances

You may need to restart an instance for routine maintenance.

Usage Notes

- Only instances in states **Available**, **Abnormal**, or **Checking restoration** can be restarted.
- Restarting an instance will interrupt services, so off-peak hours are the best time. Ensure that your application can be reconnected.
- After you restart an instance, all nodes in the instance are also restarted.
- If you enable operation protection to improve the security of your account and cloud products, two-factor authentication is required for sensitive operations. For details about how to enable operation protection, see ***Identity and Access Management User Guide***.

Procedure

Step 1 Log in to the Huawei Cloud console.

Step 2 On the **Instances** page, locate the instance you want to restart and choose **More** > **Restart** in the **Operation** column.

Alternatively, click the name of the instance you want to restart, and on the displayed **Basic Information** page, click **Restart** in the upper right corner of the page.

Step 3 If you have enabled operation protection, click **Start Verification** in the **Restart DB Instance** dialog box. On the displayed page, click **Send Code**, enter the verification code, and click **Verify**. The page is closed automatically.

Step 4 In the displayed dialog box, click **Yes**.

----End

3.5.2 Deleting Instance

You can choose to delete a pay-per-use instance on the **Instances** page based on service requirements. To delete a yearly/monthly instance, unsubscribe from it. For details, see **Unsubscribing from a Yearly/Monthly Instance**.

Precautions

- Instances where operations are being performed cannot be deleted. They can be deleted only after the operations are complete.
- If a pay-per-use instance is deleted, its automated backups will also be deleted and you will no longer be billed for them. Manual backups, however, will be retained and generate additional costs.
- After an instance is deleted, all its data and all automated backups are automatically deleted as well and cannot be recovered. You are advised to create a backup before deleting an instance. For details, see **Creating a Manual Backup**.
- After you delete an instance, all of its nodes are deleted.

- Deleted instances will be retained in the recycle bin for a period of time after being released, so you can rebuild the instance from it.

Procedure

Step 1 Log in to the Huawei Cloud console.

Step 2 On the **Instances** page, locate the instance you want to delete and choose **More** > **Delete** in the **Operation** column.

Step 3 If you have enabled operation protection, click **Start Verification** in the **Delete DB Instance** dialog box. On the displayed page, click **Send Code**, enter the verification code, and click **Verify**. The page is closed automatically.

NOTE

If you enable operation protection, two-factor authentication is required for sensitive operations to secure your account and cloud products. For details about how to enable operation protection, see ***Identity and Access Management User Guide***.

Step 4 In the displayed dialog box, click **Yes**.

Deleted instances are not displayed in the instance list any longer.

----End

3.5.3 Recycling a GeminiDB Cassandra Instance

You can restore unsubscribed yearly/monthly instances or deleted pay-per-use instances from the recycle bin.

You can restore deleted pay-per-use instances from the recycle bin.

Usage Notes

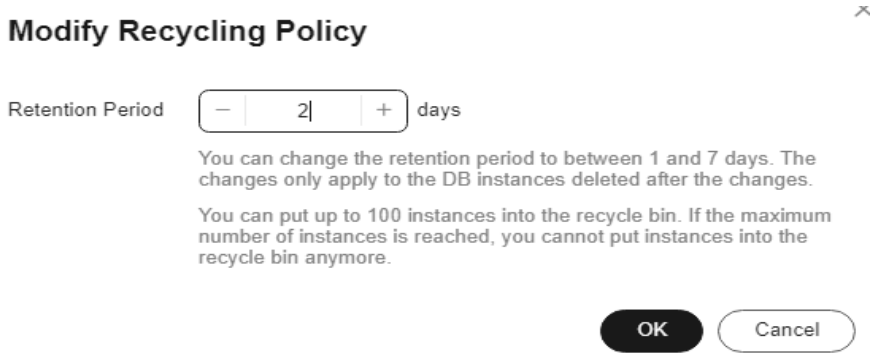
- The recycling bin is enabled by default and cannot be disabled. Instances in the recycle bin are retained for 7 days by default, and this will not incur any charges.
- Currently, you can put a maximum of 100 instances into the recycle bin.
- If you delete an instance of full storage, the deleted instance will not be moved to the recycle bin.
- You can modify the retention period, and the changes only apply to the DB instances deleted after the changes, so exercise caution when performing this operation.
- After an instance is deleted, the most recent automated full backup (if no automated full backup is available one day ago, the latest one is retained) is retained and a full backup is performed. You can select any backup file to restore the instance data.

Modifying the Recycling Policy

Step 1 Log in to the Huawei Cloud console.

Step 2 On the **Recycling Bin** page, click **Modify Recycling Policy**. In the displayed dialog box, set the retention period from 1 day to 7 days. Then, click **OK**.

Figure 3-31 Modifying a recycling policy



----End

Rebuilding an Instance

You can rebuild DB instances from the recycle bin within the retention period to restore data.

Step 1 Log in to the Huawei Cloud console.

Step 2 On the **Recycling Bin** page, locate the target instance and click **Rebuild** in the **Operation** column.

Figure 3-32 Rebuilding an instance

Step 3 On the displayed page, configure required parameters and submit the task.

----End

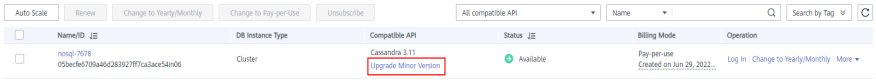
3.6 Instance Modifications

3.6.1 Upgrading a Minor Version

GeminiDB Cassandra can be upgraded by installing patches to improve performance, release new features, or fix bugs.

If a new patch is released, you can upgrade your instance by clicking the upgrade button in the **Compatible API** column on the **Instances** page.

Figure 3-33 Patch installation



If the kernel version of your instance has potential risks or major defects, has expired, or has been brought offline, the system will notify you by SMS message or email and deliver an upgrade task during maintenance.

Precautions

- Upgrade your instance once there are new patches released.
- If the database version is a risky version, the system prompts you to upgrade the database patch.
- The instance will be restarted and services may be interrupted during the upgrade. The interruption duration depends on services, quantity of nodes, and the amount of service data. Upgrade your instance during off-peak hours.
- When you upgrade a cluster, services may be interrupted a number of times equal to the number of nodes in the cluster plus one. Each interruption will last for no more than a minute and will only affect the services on that node. The upgrade duration is as follows:

  600 + (N x 60) ≤ Total upgrade duration (s) ≤ 600 + (N x 120)

  For example, if there are 9 nodes in a cluster instance, the upgrade duration is 19 to 28 minutes.

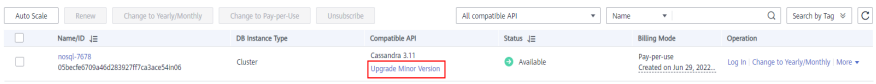
  The upgrade duration of most instances is close to 600+ (N x 60). If there are too many tokens on a single node, the upgrade duration may be increased.
- Before you upgrade a DR instance, upgrade the corresponding standby instance first and then the primary instance afterwards.

Procedure

Step 1 Log in to the Huawei Cloud console.

Step 2 On the **Instances** page, locate the instance you want to upgrade and click **Upgrade Minor Version** in the **Compatible API** column.

Figure 3-34 Patch installation

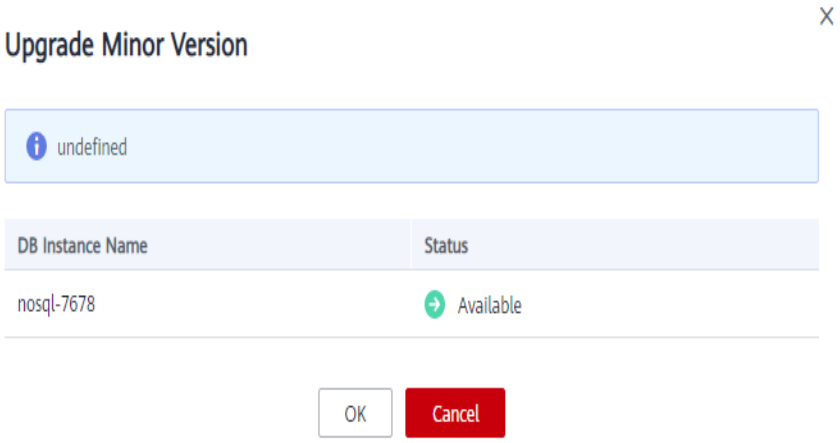


Alternatively, click the instance name to go to the **Basic Information** page. In the **DB Information** area, click **Upgrade Minor Version** in the **Compatible API** field.

Figure 3-35 Patch installation

Step 3 In the displayed dialog box, click **OK**.

Figure 3-36 Confirming dialog box



Step 4 View the upgrade result on the **Instances** page.

- When the upgrade is ongoing, the instance status is **Upgrading minor version**.
- After the upgrade is complete, the instance status changes **Available**.

----End

3.6.2 Changing an Instance Name

This section describes how to change the name of a GeminiDB Cassandra instance to identify different instances.

Method 1

Step 1 Log in to the Huawei Cloud console.

Step 2 On the **Instances** page, click next to the target instance name and change it.

- To submit the change, click **OK**.
- To cancel the change, click **Cancel**.

NOTE

The instance name:

- Can be the same as an existing instance name.
- Can include 4 to 64 bytes and must start with a letter. It is case-sensitive and allows only letters, digits, hyphens (-), and underscores (_).

Step 3 View the results on the **Instances** page.

----End

Method 2

Step 1 Log in to the Huawei Cloud console.

Step 2 On the **Instances** page, click the target instance. The **Basic Information** page is displayed.

Step 3 In the **Instance Information** area on the **Basic Information** page, click  next to **DB Instance Name** and change the instance name.

- To submit the change, click  .
- To cancel the change, click  .

NOTE

The instance name:

- Can be the same as an existing instance name.
- Can include 4 to 64 bytes and must start with a letter. It is case-sensitive and allows only letters, digits, hyphens (-), and underscores (_).

Step 4 View the results on the **Instance Management** page.

----End

3.6.3 Resetting the Administrator Password

For security reasons, change administrator passwords periodically.

Precautions

- You can reset the administrator password only when your instance is states **Available**, **Backing up**, **Checking restoration**, or **Scaling up**. You can also choose to reset the password if an instance node becomes abnormal.
- The administrator password takes effect immediately after being reset.
- If you enable operation protection to improve the security of your account and cloud products, two-factor authentication is required for sensitive operations. For details about how to enable operation protection, see ***Identity and Access Management User Guide***.

CAUTION

You are advised to change the password during off-peak hours to avoid service interruption.

Method 1

Step 1 Log in to the Huawei Cloud console.

Step 2 On the **Instances** page, locate the instance whose password you want to reset and choose **More** > **Reset Password** in the **Operation** column.

Step 3 Enter and confirm the new administrator password and click **OK**.

The password must be 8 to 32 characters in length and contain uppercase letters, lowercase letters, digits, and any of the following special characters: ~!@#%^*-_=+?

Step 4 If you have enabled operation protection, click **Start Verification** in the displayed dialog box. On the displayed page, click **Send Code**, enter the verification code, and click **Verify**. The page is closed automatically.

----End

Method 2

Step 1 Log in to the Huawei Cloud console.

Step 2 On the **Instances** page, click the instance whose password you want to reset to go to the **Basic Information** page.

Step 3 In the **DB Information** area, click **Reset Password** in the **Administrator** field.

Step 4 Enter and confirm the new administrator password and click **OK**.

The password must be 8 to 32 characters in length and contain uppercase letters, lowercase letters, digits, and any of the following special characters: ~!@#%^*-_=+?

Step 5 If you have enabled operation protection, click **Start Verification** in the displayed dialog box. On the displayed page, click **Send Code**, enter the verification code, and click **Verify**. The page is closed automatically.

----End

3.6.4 Changing vCPUs and Memory

This section describes how to change your instance vCPUs and memory to suit your service requirements.

Usage Notes

- You can increase or decrease the vCPUs and memory.
- If one instance has multiple nodes, the change will be performed on the nodes one by one. It takes about 5 to 10 minutes for each node, and the total time required depends on the number of the nodes.
- For a node whose specifications are being changed, its computing tasks are handed over to other nodes. Change specifications of nodes during off-peak hours to prevent the instance from overload.
- Do not perform DDL operations when you change the instance specifications.

  NOTE

  A data definition language (DDL) is a language for defining data structures and database objects. Common examples of DDL statements are CREATE, ALTER, and DROP. Data Definition Language (DDL) is used to create, modify, and delete database objects, such as tables, indexes, views, functions, stored procedures, and triggers.
- vCPU and memory changes are applied on all nodes in sequence. During this process, temporary I/O disruptions or increased latency may occur. You are advised to perform this operation during off-peak hours.
- If you forcibly change the specifications of an instance when the instance is abnormal, services may be affected in seconds.

Method 1

Step 1 Log in to the Huawei Cloud console.

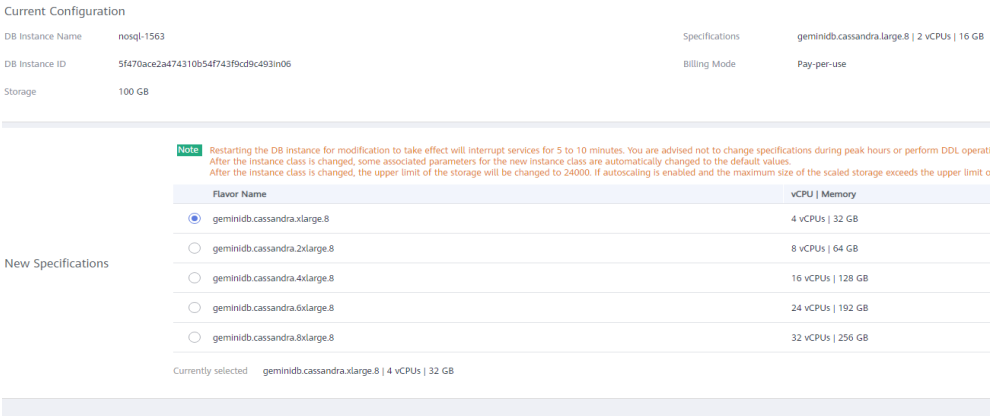
Step 2 On the **Instances** page, locate the instance whose vCPUs and memory you want to change and click its name.

Step 3 In the in **DB Information** area on the **Basic Information** page, click **Change** next to **Specifications**.

Figure 3-37 Changing specifications

Step 4 On the displayed page, select the required specifications and click **Next**.

Figure 3-38 Changing specifications



Step 5 On the displayed page, confirm the instance specifications.

- Yearly/Monthly
  - If you need to modify your settings, click **Previous**.
  - If you do not need to modify your settings, click **Submit**. If you are scaling up the instance specifications, go to the payment page, select a payment method, and complete the payment.
- Pay-per-use
  - If you need to modify your settings, click **Previous**.
  - If you do not need to modify your settings, click **Submit**.

Step 6 View the change results.

In the **DB Information** area on the **Basic Information** page, you can see the new specifications.

----End

Method 2

Step 1 Log in to the Huawei Cloud console.

Step 2 On the **Instances** page, locate the instance whose specifications you want to change and choose **More** > **Change Specifications** in the **Operation** column.

Figure 3-39 Changing specifications

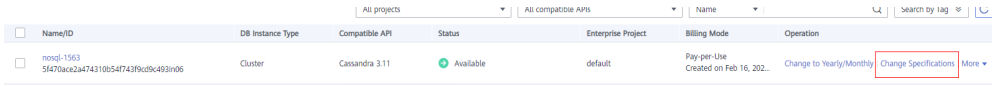
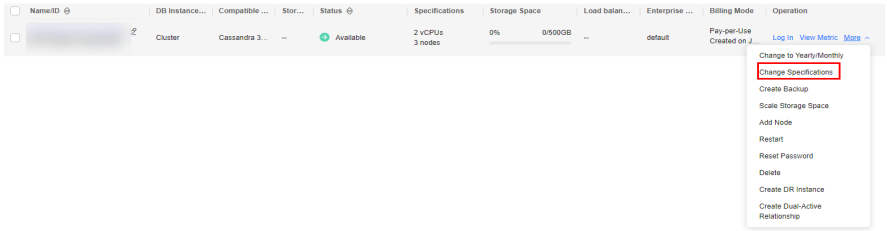
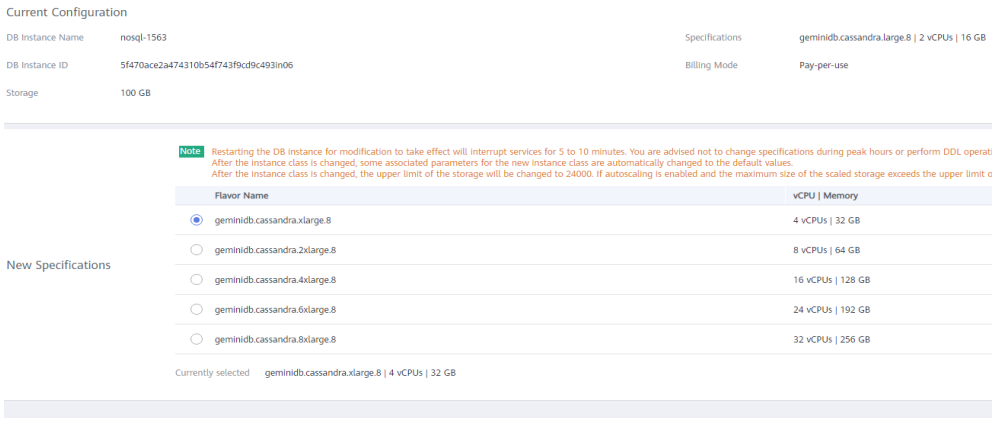


Figure 3-40 Changing specifications



Step 3 On the displayed page, select the required specifications and click **Next**.

Figure 3-41 Changing specifications



Step 4 On the displayed page, confirm the instance specifications.

- Yearly/Monthly
  - If you need to modify your settings, click **Previous**.
  - If you do not need to modify your settings, click **Submit**. If you are scaling up the instance specifications, go to the payment page, select a payment method, and complete the payment.
- Pay-per-use
  - If you need to modify your settings, click **Previous**.
  - If you do not need to modify your settings, click **Submit**.

Step 5 View the change results.

In the **DB Information** area on the **Basic Information** page, you can see the new specifications.

----End

3.6.5 Adding and Deleting Instance Nodes

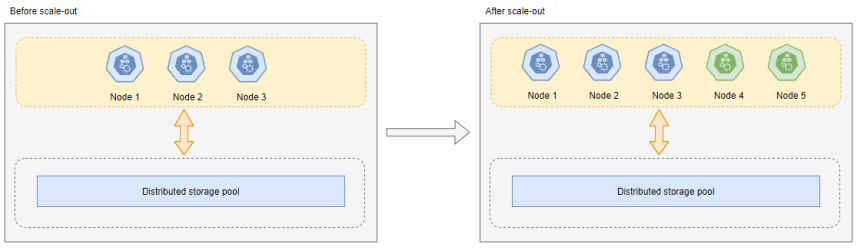
3.6.5.1 Overview

After you purchase a GeminiDB Cassandra instance, resource requirements may change along with workload volumes. You can scale your instance nodes in the following ways.

Manually Adding Instance Nodes

For example, if three nodes have been deployed and two more nodes need to be added, there will be five nodes in total. For details, see **Manually Adding Instance Nodes**.

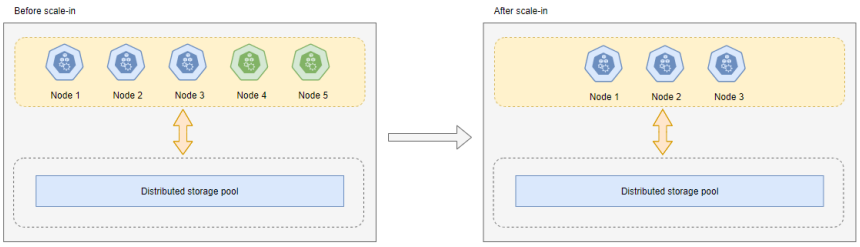
Figure 3-42 Adding instance nodes



Manually Deleting Instance Nodes

For example, if five nodes have been deployed and two of them need to be deleted, three nodes will be left. For details, see **Manually Deleting Instance Nodes**.

Figure 3-43 Deleting instance nodes



3.6.5.2 Manually Adding Instance Nodes

This section describes how to add nodes to an instance to suit your service requirements.

Usage Notes

- Adding nodes may lead to the decrease of operations per second (OPS). Perform this operation during off-peak hours.
- You can only add nodes when the instance status is **Available** or **Checking restoration**.
- An instance cannot be deleted when nodes are being added.
- You can also delete nodes as required. For details, see **Manually Deleting Instance Nodes**.
- Currently, a maximum of 60 nodes are supported. To add more, choose **Service Tickets > Create Service Ticket** in the upper right corner of the console and contact the customer service.

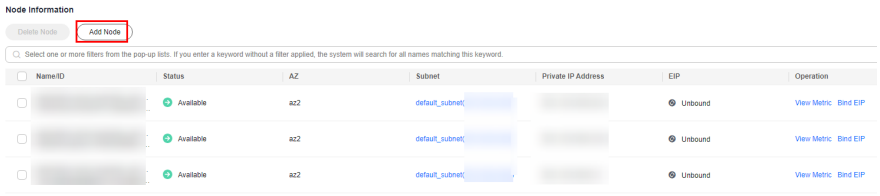
Method 1

Step 1 Log in to the Huawei Cloud console.

Step 2 On the **Instances** page, locate the instance that you want to add nodes to and click its name.

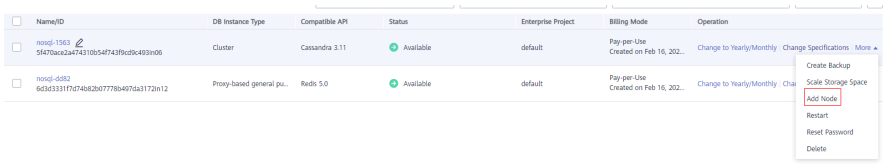
Step 3 In the **Node Information** area on the **Basic Information** page, click **Add Node**.

Figure 3-44 Node information



Step 4 Specify **Add Nodes** and click **Next**.

Figure 3-45 Adding nodes



NOTE

- New nodes are of the same specifications as existing nodes. Once a new node is added, its specifications cannot be changed.
- New nodes and the instance can be in different subnets of the same VPC.

Step 5 On the displayed page, confirm the node configurations.

- Yearly/Monthly
  - If you need to modify your settings, click **Previous**.
  - If you do not need to modify your settings, click **Next** and complete the payment.

- Pay-per-use
  - If you need to modify your settings, click **Previous**.
  - If you do not need to modify your settings, click **Submit**.

Step 6 View the results.

- When new nodes are being added, the instance status is **Adding node**.
- After the nodes are added, the instance status becomes **Available**.
- Click the instance name. In the **Node Information** area on the **Basic Information** page, view information about the new nodes.

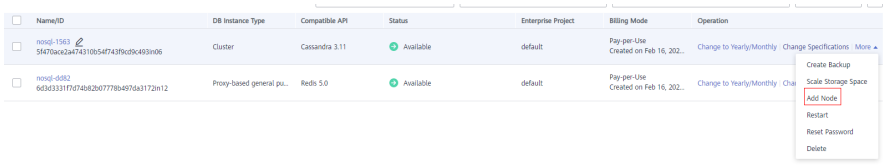
----End

Method 2

Step 1 Log in to the Huawei Cloud console.

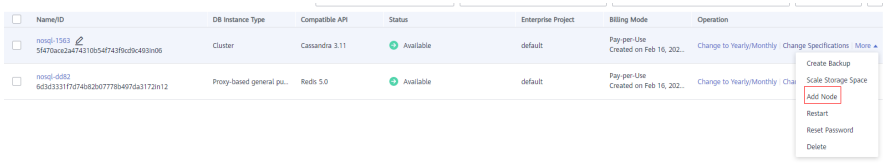
Step 2 On the **Instances** page, locate the target instance and choose **More** > **Add Node** in the **Operation** column.

Figure 3-46 Adding nodes



Step 3 Specify **Add Nodes** and click **Next**.

Figure 3-47 Adding nodes



NOTE

- New nodes are of the same specifications as existing nodes. Once a new node is added, its specifications cannot be changed.
- New nodes and the instance can be in different subnets of the same VPC.

Step 4 On the displayed page, confirm the node configurations.

- Yearly/Monthly
  - If you need to modify your settings, click **Previous**.
  - If you do not need to modify your settings, click **Next** and complete the payment.
- Pay-per-use

  - If you need to modify your settings, click **Previous**.
  - If you do not need to modify your settings, click **Submit**.

Step 5 View the results.

- When new nodes are being added, the instance status is **Adding node**.
- After the nodes are added, the instance status becomes **Available**.
- Click the instance name. In the **Node Information** area on the **Basic Information** page, view information about the new nodes.

----End

3.6.5.3 Manually Deleting Instance Nodes

You can delete nodes that are no longer used to release resources.

Usage Notes

- Deleted nodes cannot be recovered. Exercise caution when performing this operation.
- Only pay-per-use instances can be deleted.
- This function is now in OBT. To use it, choose **Service Tickets > Create Service Ticket** in the upper right corner of the console and contact the customer service.
- Deleting nodes will cause the OPS to decrease for a short period of time. Deleting nodes during off-peak hours.
- If you enable operation protection to improve the security of your account and cloud products, two-factor authentication is required for sensitive operations. For details about how to enable operation protection, see ***Identity and Access Management User Guide***.

Procedure

Step 1 Log in to the Huawei Cloud console.

Step 2 On the **Instances** page, locate the instance that you want to delete nodes from and click its name.

Step 3 In the **Node Information** area on the **Basic Information** page, locate the target node and click **Delete** in the **Operation** column.

Step 4 If you have enabled operation protection, click **Start Verification** in the **Delete Node** dialog box. On the displayed page, click **Send Code**, enter the verification code, and click **Verify**. The page is closed automatically.

Step 5 In the displayed dialog box, click **Yes**.

- When the node is being deleted, the instance status is **Deleting node**.
- After the node is deleted, the instance status becomes **Available**.

----End

3.6.6 Scaling Storage Space

3.6.6.1 Overview

As more data is added, you may start to run out of space. This section describes how to scale up storage space of your instance. **Table 3-15** lists the scaling methods supported by GeminiDB Cassandra instances.

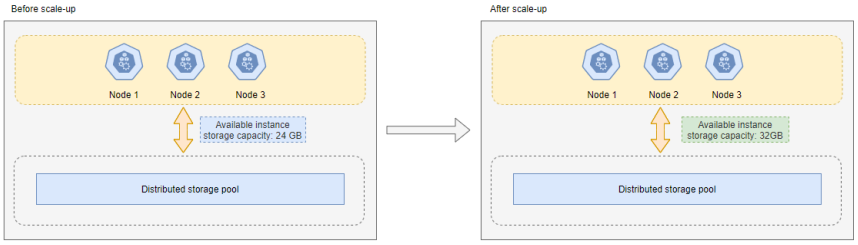
Table 3-15 Scaling methods

Method	Description
Manually Scaling Up Storage Space	You can specify how much storage space needs to be added. The added value must be a multiple of 1 (GB). The total storage space cannot exceed the upper limit defined by your instance specifications.
Automatically Scaling Up Storage Space	If storage usage exceeds the configured threshold, autoscaling will be triggered. The storage is scaled up by a percentage you specify. The added storage space is the current storage space multiplied by the scaling increment.

Manually Scaling Up Storage Space

For example, if the storage space of a cluster instance is 24 GB and is increased by 8 GB, the storage space will become 32 GB.

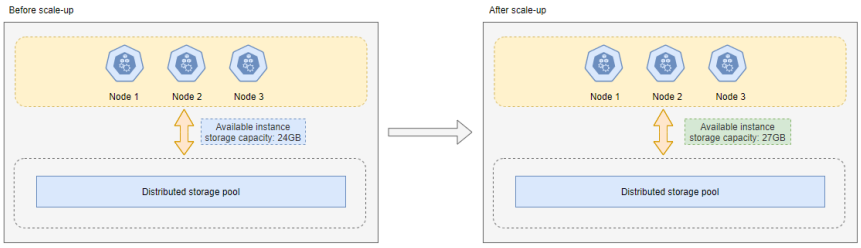
Figure 3-48 Manually scaling up storage space



Automatically Scaling Up Storage Space

For example, the storage space of a cluster instance is 24 GB before scale-up, the storage usage threshold for triggering autoscaling is set to 80%, and the total storage needs to be automatically scaled up by 10%. If the storage usage is greater than or equal to 80%, the storage space is automatically scaled up by 2.4 GB (24 x 10%), which is rounded up to 3 GB. In this case, the total storage space becomes 27 GB (24 + 3).

Figure 3-49 Automatically scaling up storage space



3.6.6.2 Manually Scaling Up Storage Space

This section describes how to scale up storage of an instance to suit your service requirements.

Usage Notes

- Storage space can only be scaled up.
- You do not need to restart the instance while scaling its storage. The change has no impact on existing data, services, or your databases.

Procedure

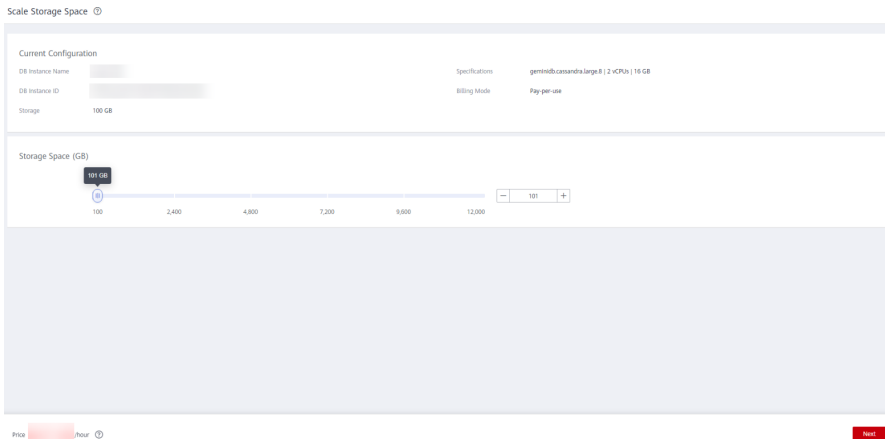
Step 1 Log in to the Huawei Cloud console.

Step 2 On the **Instances** page, locate the target instance and choose **More** > **Scale Storage Space** in the **Operation** column.

Click the instance name. In the **Storage Space** area on the **Basic Information** page, click **Scale**.

Step 3 On the displayed page, specify new storage and click **Next**.

Figure 3-50 Scaling up storage space



- To scale up classic storage, you need to add at least 1 GB each time. The value must be an integer.
- To scale up cloud native storage, you need to add at least 10 GB each time. The value must be an integer multiple of 10.

Step 4 On the displayed page, confirm the storage space.

- Yearly/Monthly
  - If you need to modify your settings, click **Previous**.
  - If you do not need to modify your settings, click **Next** and complete the payment.
- Pay-per-use
  - If you need to modify your settings, click **Previous**.
  - If you do not need to modify your settings, click **Submit**.

Step 5 Check the results.

- During the scaling process, the instance status becomes **Scaling up**.
- After the scaling process, the instance status becomes **Available**.
- Click the instance name. In the **Storage Space** area on the **Basic Information** page, check the new storage space.

----End

3.6.6.3 Automatically Scaling Up Storage Space

You can enable **Auto Scale** for GeminiDB Cassandra instances. When storage space usage reaches the upper limit, autoscaling is triggered.

You can enable **Auto Scale**:

1. When you create an instance. For details, see **Buying a GeminiDB Cassandra Instance**.
2. After you create an instance

This section describes how to configure **Auto Scale** after an instance is created.

NOTE

- If you enable **Auto Scale** using a Huawei Cloud account, no additional configuration is required.
- If you enable **Auto Scale** as an IAM user first time, you need to obtain the permission to create an agency.

Configuring Permissions

If you are using an IAM user, perform the following operations to configure GeminiDB and IAM permissions before enabling **Auto Scale**:

1. Configure the GeminiDB FullAccess permission.
2. Configure fine-grained permissions for IAM.

   For details about how to configure IAM permissions, see **Creating a Custom Policy**.

   If you use the JSON view to configure a custom policy, the policy content is as follows:

```
{
    "Version":"1.1",
    "Statement":[
        {
            "Effect":"Allow",
```



```
            "Action":[
                "iam:permissions:listRolesForAgencyOnProject",
                "iam:permissions:grantRoleToGroupOnProject",
                "iam:agencies:createAgency",
                "iam:agencies:listAgencies",
                "iam:roles:listRoles",
                "iam:roles:createRole"
            ]
        }
    ]
}
```

3. **Create a user group and assign permissions**.

   You can create a user group on the IAM console and grant it custom permissions created in **2** and the security administrator role.

4. **Create an IAM user** and add it to a user group.

   Use a Huawei Cloud account or an IAM account to locate the IAM user that the target instance belongs to. Add it to the user group created in **3**. The IAM user will inherit permissions of the user group. For details, see **Adding IAM Users to or Removing IAM Users from a User Group**.

Usage Notes

- Autoscaling is available only when your account balance is sufficient.
- Autoscaling is now in OBT. To use it, choose **Service Tickets > Create Service Ticket** in the upper right corner of the console and contact the customer service.
- The instance must be in the **Available** status.
- Once **Auto Scale** is enabled, an agency will be created and fees will be automatically deducted.
- You do not need to restart the instance while scaling its storage. The change has no impact on existing data, services, or your databases.

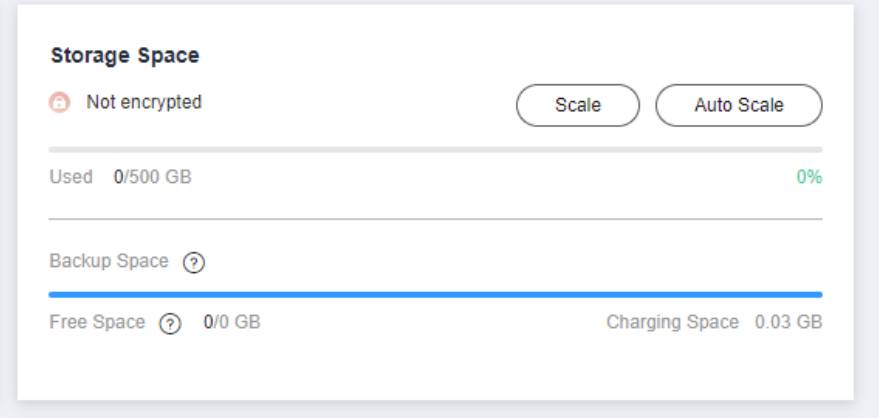
Automatically Scaling Up Storage of a Single Instance

Step 1 Log in to the Huawei Cloud console.

Step 2 On the **Instances** page, click the target instance. The **Basic Information** page is displayed.

Step 3 In the **Storage Space** area, click **Auto Scale**.

Figure 3-51 Auto Scale



Step 4 Toggle on **Auto Scale** and specify the parameters below.

Figure 3-52 Configuring autoscaling parameters

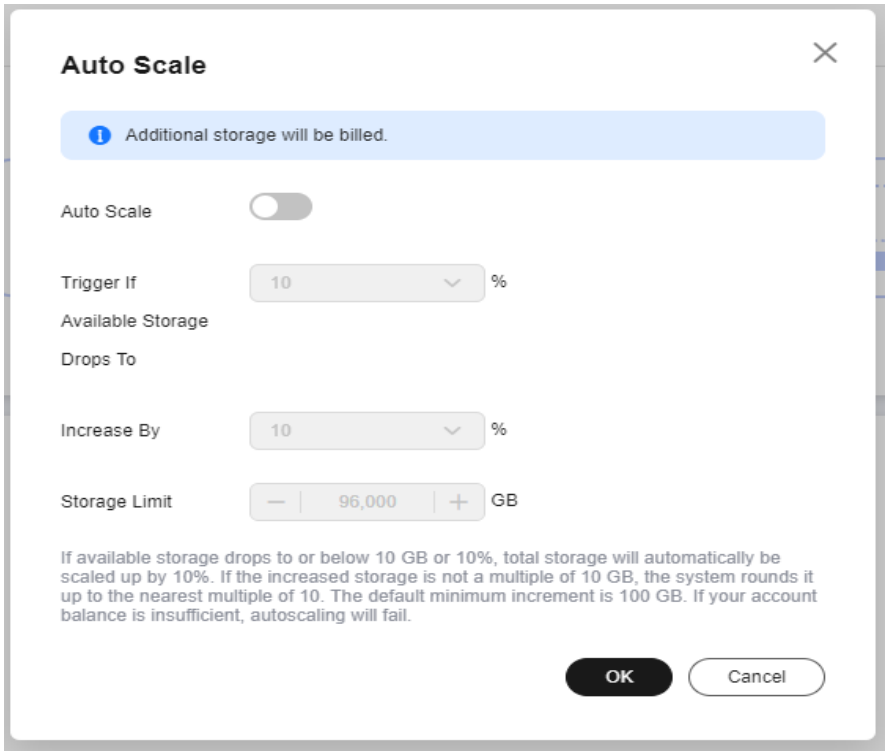


Table 3-16 Description

Parameter	Description
Auto Scale	If you toggle on this switch, autoscaling is enabled.
Trigger If Available Storage Drops To	When the available storage usage drops to a specified threshold or the available storage drops to 10 GB, autoscaling is triggered.
Increase By	Percentage that your instance storage will be scaled up at. The value can be **10%**, **15%**, or **20%**. If the value is not a multiple of 10, it is rounded up. At least 100 GB is added each time.
Storage Limit	Limit of storage (GB) that can be automatically scaled up to. • Instance storage upper limit ≥ Current storage + 100 GB • The upper limit cannot exceed the **maximum storage supported by the current specifications**.

Step 5 Click **OK**.

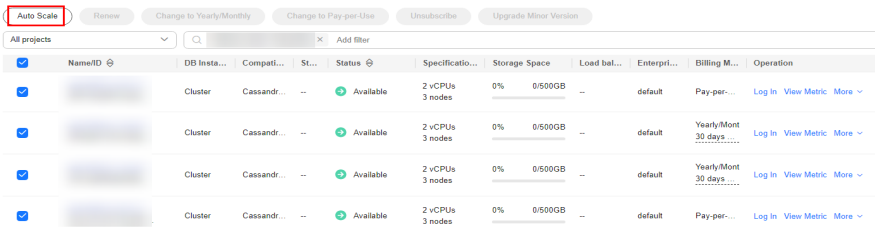
----End

Automatically Scaling Up Storage of Instances in Batches

Step 1 Log in to the Huawei Cloud console.

Step 2 Select instances and click **Auto Scale**.

Figure 3-53 Auto Scale



Step 3 Select an instance, toggle on **Auto Scale**, and specify the parameters below.

Figure 3-54 Batch Auto Scale

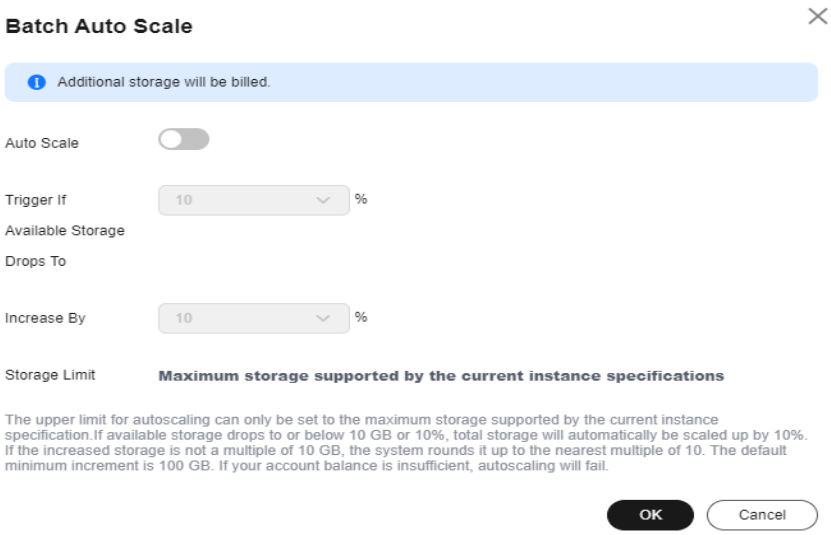


Table 3-17 Description

Parameter	Description
Auto Scale	If you toggle on this switch, autoscaling is enabled.
Trigger If Available Storage Drops To	When the available storage usage drops to a specified threshold or the available storage drops to 10 GB, autoscaling is triggered.

Parameter	Description
Increase By	Percentage that your instance storage will be scaled up at. The value can be **10%**, **15%**, or **20%**. If the value is not a multiple of 10, it is rounded up. At least 100 GB is added each time.
Storage Limit	The value cannot be specified. By default, the storage is scaled up to the maximum defined by your instance specifications.

Step 4 Click **OK**.

----End

3.7 Intra-region DR

3.7.1 Creating a DR Instance

GeminiDB instances can be deployed in HA mode. If an instance fails to be connected due to a natural disaster, its workloads can be taken over by its DR instance. You only need to modify a database connection address on applications to quickly restore the faulty instance.

Usage Notes

- A primary instance can have only one DR instance.
- This function is now in OBT. To use it, choose **Service Tickets > Create Service Ticket** in the upper right corner of the console and contact the customer service.

Prerequisites

A primary instance has been created.

Constraints

1. Currently, a DR instance can be provisioned for a GeminiDB Cassandra instance but is unavailable for GeminiDB HBase and DynamoDB-Compatible instances.

2. Currently, counter tables, TRUNCATE operations, and Lucene indexes cannot be synchronized between DR instances.

Creating a DR Instance

Step 1 Log in to the Huawei Cloud console.

Step 2 On the **Instances** page, locate the primary instance you want to create a DR instance for and choose **More** > **Create DR Instance** in the **Operation** column.

Step 3 On the displayed page, configure required parameters and click **Next**.

Table 3-18 Basic information

Parameter	Description
Billing Mode	Select **Yearly/Monthly** or **Pay-per-use**. • **Yearly/Monthly** – Specify **Required Duration**. The system deducts fees from your account based on the service price. – If you do not need such an instance any longer after it expires, change the billing mode to pay-per-use. For details, see **Changing Yearly/Monthly to Pay-per-Use**. **NOTE** Yearly/Monthly instances cannot be deleted directly. If such an instance is no longer required, unsubscribe from it. For details, see **Unsubscribing from a Yearly/Monthly Instance**. • **Pay-per-use** – If you select this billing mode, you are billed based on how much time the instance is in use. – To use an instance for a long time, change its billing mode to yearly/monthly to reduce costs. For details, see **Changing Pay-per-Use to Yearly/Monthly**.
Region	The region is the same as that of the primary instance.
DB Instance Name	The instance name: The name can include 4 to 64 bytes and must start with a letter. It is case-sensitive and allows only letters, digits, hyphens (-), and underscores (_).
Compatible API	Cassandra
DB Instance Type	Cluster
DB Engine Version	The compatible API version is the same as that of the primary instance.
CPU Type	The CPU type is the same as that of the primary instance.
AZ	Availability zone where the instance is created. An AZ is a part of a region with its own independent power supplies and networks. AZs are physically isolated but can communicate with each other over a private network. Instances can be deployed in a single AZ or three AZs. • To deploy instances in a single AZ, select one AZ. • To deploy instances across AZs for disaster recovery, select three AZs, where the instance nodes will be evenly distributed.

Table 3-19 Specifications and storage

Parameter	Description
Instance Specifications	vCPUs and memory of the instance. Performance specifications vary depending on the connections and maximum IOPS. After an instance is created, you can change its specifications. For details, see **Changing vCPUs and Memory**.
Nodes	Specify the number of nodes based on service requirements. After an instance is created, you can add nodes by referring to **Manually Adding Instance Nodes**.
Storage Space	Storage space depends on the instance specifications. The minimum storage space is 100 GB, and the storage space you set must be an integer. You can increase a minimum of 1 GB at a time. You are advised to enable **Auto Scale** and set trigger conditions and storage limit. After autoscaling is triggered, the system automatically scales up the storage to ensure that the instance has sufficient storage and keeps available. Take care with the following parameters: • **Trigger If Available Storage Drops To**: storage threshold for triggering autoscaling. When the available storage usage drops to a specified threshold or the available storage drops to 10 GB, autoscaling is triggered. • **Increase By**: percentage that your instance storage will be scaled up at. If the increased storage is not a multiple of 10 GB, the system will round it up to the nearest multiple of 10 GB. At least 100 GB is added each time. • **Storage Limit**: maximum amount that the system can automatically scale up an instance's storage to. The value must be no less than the current storage of your instance and cannot exceed the maximum storage supported by your instance. After an instance is created, you can scale up its storage if necessary. For details, see **Manually Scaling Up Storage Space**. **NOTE** • Once **Auto Scale** is enabled, an agency will be created and fees will be automatically deducted. • Autoscaling is available only to users with required permissions. To use it, choose **Service Tickets > Create Service Ticket** in the upper right corner of the console and contact the customer service. • You can enable **Auto Scale** after an instance is created. For details, see **Automatically Scaling Up Storage Space**.

Table 3-20 Network

Parameter	Description
VPC	The VPC of the DR instance remains unchanged by default.
Subnet	The subnet of the DR instance remains unchanged by default. If you select another subnet in the same VPC, ensure that the selected subnet can be connected to the subnet of the primary instance.
Security Group	The security group of the DR instance remains unchanged by default. Access from the 192.168.0.0/24 CIDR block in the security group should be allowed to ensure that DR instances can be created and work properly.

Table 3-21 Database configuration

Parameter	Description
Administrator	Username of the administrator account. The default value is **rwuser**.
Administrator Password	The password must be the same as that of the primary instance to ensure that a switchover is performed in the event of a failure.
Confirm Password	Enter the administrator password again.
Parameter Template	A parameter template contains API configuration values that can be applied to one or more instances. After an instance is created, you can modify its parameters to better meet your service requirements. For details, see **Modifying Parameters of GeminiDB Cassandra Instances**.

Table 3-22 Tags

Parameter	Description
Tags	The setting is optional. Adding tags helps you better identify and manage your instances. A maximum of 20 tags can be added for each instance. A tag consists of a tag key and a tag value. • A tag key is mandatory if the instance will be tagged. Each tag key is unique for each instance. It can include up to 36 characters, including digits, letters, underscores (_), and hyphens (-). • A tag value is optional if the instance will be tagged. The value can be empty. The value can contain up to 43 characters, including digits, letters, underscores (_), periods (.), and hyphens (-). After an instance is created, you can view its tags on the **Tags** tab and can also add, modify, and delete tags of your instance. For details, see **Managing GeminiDB Cassandra Instance Tags**.

Table 3-23 Required duration

Parameter	Description
Required duration	The length of your subscription if you select **Yearly/Monthly** billing. Subscription lengths range from one month to three years.
Auto-renew	• This option is not selected by default. • If you select this option, the instance is automatically renewed based on the subscription duration.

Step 4 On the displayed page, confirm the instance details.

- Yearly/Monthly
  - To modify the configurations, click **Previous**.
  - If you do not need to modify the settings, read and agree to the service agreement, click **Pay Now**, and complete the payment.
- Pay-per-use
  - To modify the configurations, click **Previous**.
  - If no modification is required, read and agree to the service agreement and click **Submit**.

Step 5 On the **Instances** page, click ⌄ in front of the primary instance to view and manage the DR instance.

- During DR instance creation, the status of the primary instance is **DR cluster being created**, and the status of the DR instance is **Creating**. This process takes about 5 to 9 minutes.

- After the instance is created, its status becomes **Available**.

  You can click  in the upper right corner of the page to refresh the instance status.
- An automated backup policy is enabled by default during instance creation. A full backup is automatically triggered after an instance is created.

----End

3.7.2 Deleting the DR Relationship

You can delete the primary or DR instance to delete the DR relationship.

Precautions

- When you delete an instance, all the data in it and all its automated backups are automatically deleted as well and cannot be restored.
- After you delete an instance, all nodes in the instance are also deleted.
- To delete a yearly/monthly instance, you need to unsubscribe from the order. For details, see **Unsubscribing from a Yearly/Monthly Instance**.
- If you enable operation protection to improve the security of your account and cloud products, two-factor authentication is required for sensitive operations. For details about how to enable operation protection, see ***Identity and Access Management User Guide***.

Procedure

Step 1 Log in to the Huawei Cloud console.

Step 2 On the **Instances** page, locate the primary or DR instance that you want to delete and choose **More** > **Delete** in the **Operation** column.

Step 3 If you have enabled operation protection, click **Start Verification** in the **Delete DB Instance** dialog box. On the displayed page, click **Send Code**, enter the verification code, and click **Verify**. The page is closed automatically.

Step 4 In the displayed dialog box, click **Yes**.

When the instance is being deleted, its status is **DR relationship being canceled**. After the instance is deleted, it is not displayed in the instance list.

----End

3.8 Cross-region Dual-active DR

3.8.1 Overview

GeminiDB Cassandra supports cross-region dual-active DR and bidirectional synchronization between two instances at different sites. Once an instance becomes faulty, the other instance takes over read/write traffic to ensure service continuity.

Cross-region dual-active DR allows you to deploy two GeminiDB Cassandra instances in different data centers. Both of the two instances can handle service

requests. If a data center becomes faulty, services in the faulty data center can be switched to the other data center to recover services without any interruption.

For how to configure cross-region dual-active DR, see **Creating a Dual-Active Relationship**.

3.8.2 Creating a Dual-Active Relationship

GeminiDB Cassandra API allows you to create a dual-active relationship for two instances in different regions, so that their data can be synchronized.

This section describes how to create such a dual-active relationship.

The current instance is the source instance, and you need to specify the target instance.

Usage Notes

- This function is now in OBT. To use it, choose **Service Tickets > Create Service Ticket** in the upper right corner of the console and contact the customer service.
- Before creating a dual-active relationship, create a target instance in a specific region and ensure it has the same or higher specifications and storage capacity than the source instance. To lift the specification restrictions, choose **Service Tickets > Create Service Ticket** in the upper right corner of the console and contact the customer service. The target instance specifications cannot be too smaller than the source instance specifications, or the target instance may have insufficient CPU or memory resources.
- Ensure the target instance has no additional tables before creating the dual-active relationship.
- The target instance must have the same administrator password as the source instance.
- To create a dual-active relationship again after it is removed, execute the DROP statement to clear tables in the target instance.
- The source instance transfers all of its data to the target instance.

Constraints

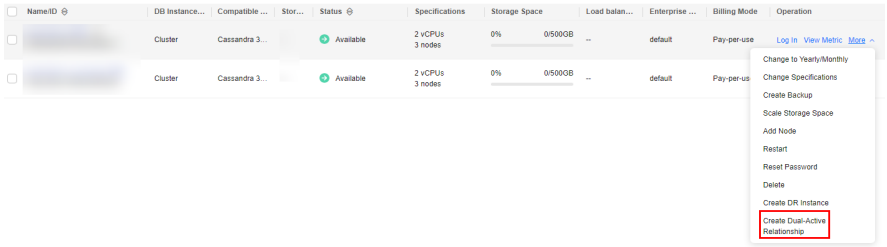
- Currently, active-active GeminiDB Cassandra instances can be created, but active-active GeminiDB DynamoDB-Compatible instances cannot.
- Currently, counter tables, TRUNCATE operations, and Lucene indexes cannot be synchronized between active-active instances.

Procedure

Step 1 Log in to the Huawei Cloud console.

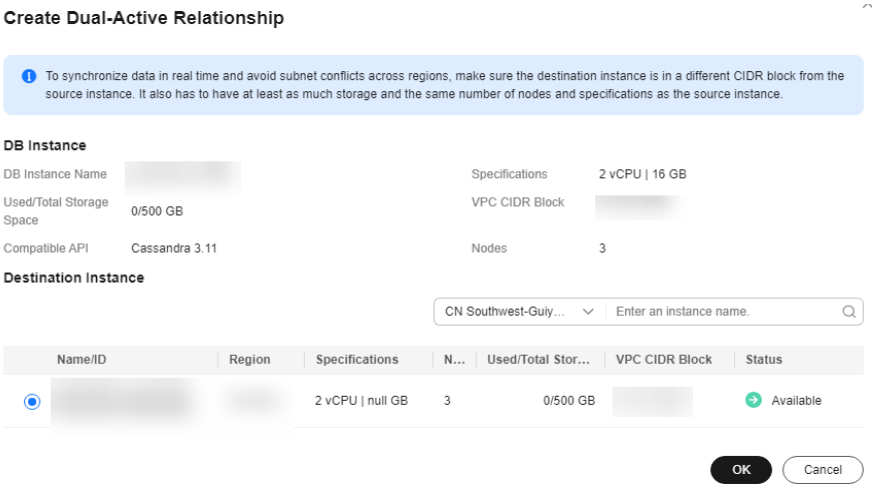
Step 2 On the **Instances** page, locate the source instance that you want to create a dual-active relationship for and choose **More** > **Create Dual-Active Relationship** in the **Operation** column.

Figure 3-55 Creating a dual-active relationship



Step 3 On the **Create Dual-Active Relationship** dialog box, locate the destination instance as the dual-active DR instance.

Figure 3-56 Selecting the destination instance



NOTE

The destination instance must be in a different CIDR block from the source instance and has the same or higher specifications and no less nodes and storage space than the source, to synchronize data in real time between them and avoid subnet conflicts across regions.

Step 4 Click **OK**.

Step 5 On the **Instances** page, click ⌄ before the source instance and view and manage its DR instance.

- When the DR instance is being created, its status is **Creating dual-active relationship**.
- After the instance is created, its status becomes **Available**.

  You can click ↻ in the upper right corner of the page to refresh the instance status.

----End

3.8.3 Deleting a Dual-active Relationship

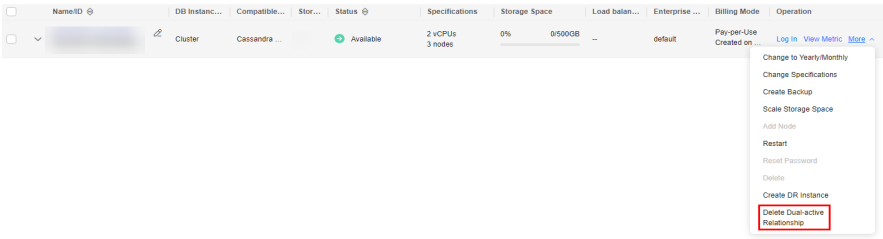
This section describes how to delete a dual-active relationship on the GeminiDB Cassandra console.

Procedure

Step 1 Log in to the Huawei Cloud console.

Step 2 On the **Instances** page, locate the instance that you want to delete a dual-active relationship from and choose **More** > **Delete Dual-active Relationship** in the **Operation** column.

Figure 3-57 Deleting a dual-active relationship



Step 3 In the displayed dialog box, click **Yes**.

When the instance is being deleted, its status is **Deleting dual-active relationship**. After the relationship is deleted, the instance status changes to **Available**.

----End

3.9 Data Backup

3.9.1 Overview

GeminiDB Cassandra API supports instance backups and restorations to ensure data reliability. After an instance is deleted, the manual backup data is retained. Automated backup data is released together with instances. Backup data cannot be downloaded or exported.

Usage Notes

Backing up data consumes a few CPUs. Uploading backup files to OBS occupies bandwidth of compute nodes, causing slight latency and jitter.

Backup Methods

GeminiDB Cassandra instances support automated backup and manual backup.

- Automated backup

  You can click **Modify Backup Policy** on the GeminiDB console, and the system will automatically back up your instance data based on the time

window and backup cycle you set in the backup policy and will store the data for the retention period you specified.

Automated backups cannot be manually deleted. You can adjust their retention period by referring to **Modifying an Automated Backup Policy**, and backups that expire will be automatically deleted.

- Manual backup

  A manual backup is a full backup of a DB instance and can be retained until you manually delete it. Manual backup can be triggered at any time to meet your service requirements.

  Regularly backing up your database is recommended. If your database becomes faulty or data is corrupted, you can restore it from backups.

Table 3-24 Comparison between automated backup and manual backup

Backup Method	Scenario
Automated backup	After you set a backup policy, the system automatically backs up your database based on the policy. You can also modify the policy based on service requirements. Either incremental or full backup is supported.
Manual backup	You can enable full backup for your instance based on service requirements.

Cross-region and table-level backups are supported based on application scenarios.

Table 3-25 Application scenarios

Method	Scenario
Cross-region backup	GeminiDB Cassandra API allows you to store backups in the destination region. Then for disaster recovery, you can restore the backups to a new instance in another region. Only an automated full backup is supported.

Full and incremental backups are created based on data volumes.

Table 3-26 Comparison between full and incremental backups

Backup Type	Full backup	Incremental backup

Description	All data in an instance is backed up.	Only data that has changed within a certain period is backed up.
Enabled by Default	Yes	Yes
Retention Duration	• You can specify how many days automated backups can be retained for. If you shorten the retention duration, the new backup policy takes effect for existing backups. • Manual backups are always retained even though a GeminiDB Cassandra instance is deleted. They can only be deleted manually.	Incremental backups will be deleted along with automated full backups.
Feature	• All data of your instance is backed up in the current point of time. • You can use a full backup to restore all data generated when its backup was created. • Full backups can be created automatically or manually.	• Incremental data in your instance is backed up since the last full backup. • When you use an incremental backup for restoration, the last full backup data and the incremental data generated since then are downloaded. • Incremental backups can be created automatically only.

How to View	Click an instance name. On the **Backups & Restorations** page, click the **Instance-level Backups** and **Table-level Backups** tabs to view the backup size.	Click an instance name. On the **Backups & Restorations** page, click the **Incremental Backup** tab to view the backup size.

How Backup Works

GeminiDB Cassandra provides a dedicated node (seed) responsible for managing backups. As shown in the following figure, a GeminiDB Cassandra cluster chooses the seed node for backing up data. The node takes snapshots of data in seconds and then stores them as compressed backups in OBS buckets, without using any store space of your instance. The CPU usage may increase 5% to 15% because uploading backups consumes CPU resources.

Backup Storage

Backups are stored in OBS buckets, providing disaster recovery and saving space.

After you purchase an instance, GeminiDB Cassandra API will provide additional backup storage of the same size as you purchased. For example, if you purchase an instance of 100 GB, you will obtain additional backup storage of 100 GB free of charge. If the size of backup data does not exceed 100 GB, the backup data is stored on OBS free of charge. If the size of the backup data exceeds 100 GB, you will be charged based on the OBS billing rules.

3.9.2 Managing Automated Backups

Automated backups can be created to ensure data reliability. If a database or table is deleted, maliciously or accidentally, backups can help restore your data.

Usage Notes

- Backup files are saved as packages in OBS buckets. Uploading backup files and reading service data both consume bandwidth, so the upload bandwidth of OBS is limited. The upload bandwidth of a single node ranges from 20 MB/s to 70 MB/s.

  For better performance, you need to specify appropriate nodes for an instance and take into account the bandwidth for uploading backups.
- The CPU usage may increase 5% to 15% because uploading backups consumes CPU resources.
- The memory usage may increase by about 300 MB during the upload of backups. The increase depends on the instance's data volume. The increased memory mainly caches data during backup upload and service read. After the backup upload is complete, the memory recovers.
- You can manually modify incremental backups of a GeminiDB Cassandra instance.

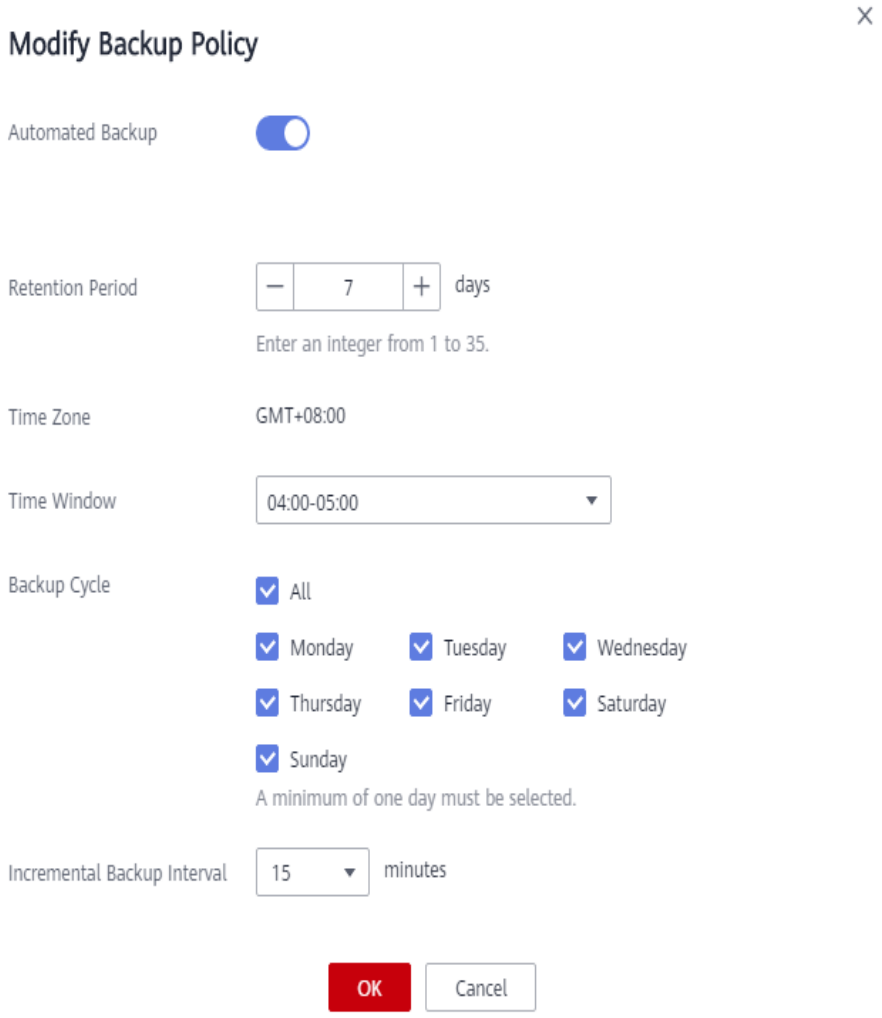
- To enable the incremental backup function, choose **Service Tickets > Create Service Ticket** in the upper right corner of the console and contact the customer service.
- After the incremental backup function is enabled, differential backup is selected by default. To enable PITR, choose **Service Tickets > Create Service Ticket** in the upper right corner of the console and contact the customer service.

Automated Backup Policy

Automated backups are generated according to a backup policy and saved as packages in OBS buckets to ensure data confidentiality and durability. You are advised to regularly back up your database, in case it becomes faulty or damaged. However, backing up data might affect the database read and write performance so it is recommended that you enable automated backups during off-peak hours.

When you create an instance, an automated backup policy is enabled by default.

Figure 3-58 Enabling the automated backup policy



- **Incremental Backup Interval**: Incremental backups are generated every 15 minutes.

- **Retention Period**: Automated backup files are saved for seven days by default. The retention period ranges from 1 to 3660 days. Full backups are retained till the retention period expires. However, even if the retention period has expired, the most recent backup will be retained.
  - Extending the retention period improves data reliability. You can extend the retention period as needed.
  - If you shorten the retention period, the new backup policy takes effect for existing backups. Any automated backups (including full and incremental backups) that have expired will be automatically deleted. Manual backups will not be automatically deleted but you can delete them manually.

  NOTE

  - If the retention period is shorter than seven days, the system automatically backs up data daily.
  - The system checks existing automated backup files and deletes the files that exceed the backup retention period you set.
  - **Time Window**: A one-hour period the backup will be scheduled within 24 hours, such as or 04:00–05:00. The backup time is in GMT format. If the DST or standard time is switched, the backup time segment changes with the time zone.

  If **Retention Period** is set to **2**, full and incremental backups that have been stored for more than two days will be automatically deleted. That is, the backup generated on Monday will be deleted on Wednesday. Similarly, the backup generated on Tuesday will be deleted on Thursday.

  Policy for automatically deleting full backups:

  To ensure data integrity, even after the retention period expires, the most recent backup will be retained, for example,

  If **Backup Cycle** was set to **Monday** and **Tuesday** and the **Retention Period** was set to **2**:

  - The full backup generated on Monday will be automatically deleted on Thursday. The reasons are as follows:

    The full backup generated on Monday expires on Wednesday, but it is the last backup, so it will be retained until a new backup expires. The next backup will be generated on Tuesday and will expire on Thursday. So the full backup generated on Monday will not be automatically deleted until Thursday.
  - The full backup generated on Tuesday will be automatically deleted on Wednesday of the following week. The reasons are as follows:

    The backup generated on Tuesday will expire on Thursday, but as it is the last backup, so it will be retained until a new backup expires. The next backup will be generated on the following Monday and will expire on the following Wednesday. So the full backup generated on Tuesday will not be automatically deleted until the following Wednesday.
- **Backup Cycle**: By default, each day of the week is selected.
  - **All**: Each day of the week is selected. The system automatically backs up data every day.
  - You can select one or more days in a week. The system automatically backs up data at the specified time.

NOTE

A full backup starts within one hour of the time you specify. The amount of time required for the backup depends on the amount of data to be backed up. The more data has to be backed up, the longer it will take.

- After an instance is created, you can set an automated backup policy. The system will back up data based on the automated backup policy.
- If **Automated Backup** is disabled, any automated backups in progress stop immediately.

Modifying an Automated Backup Policy

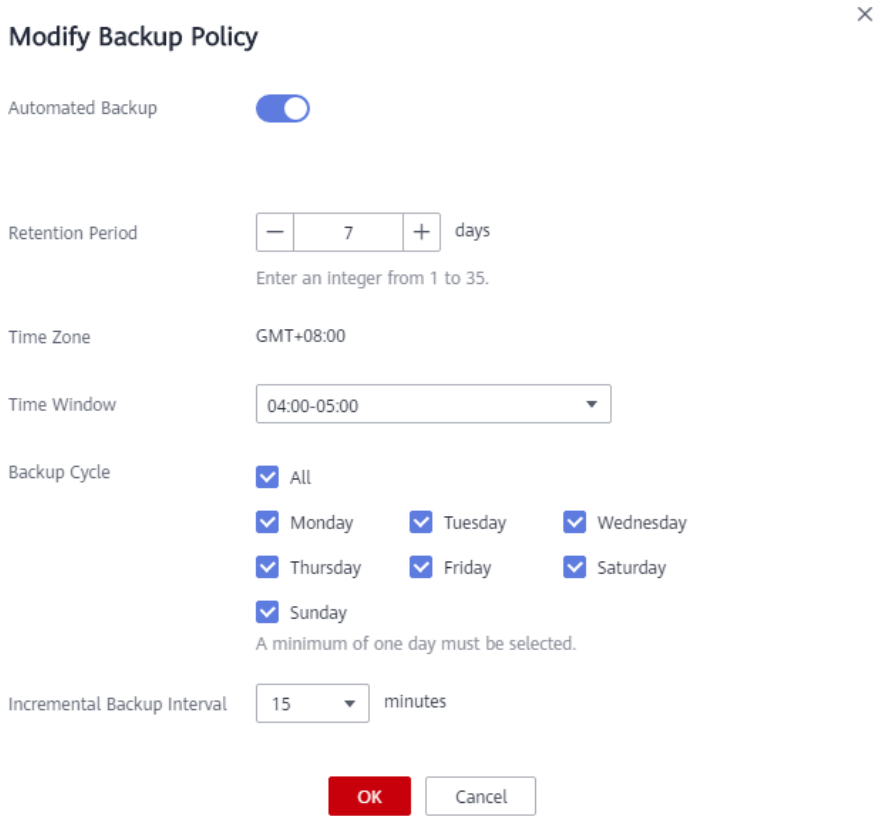
Step 1 Log in to the Huawei Cloud console.

Step 2 On the **Instances** page, click the instance whose backup policy you want to modify and click its name.

Step 3 Choose **Backups & Restorations** in the navigation pane one the left, and click **Modify Backup Policy**. In the displayed dialog box, set the backup policy and click **OK**.

For details, see **Automated Backup Policy**.

Figure 3-59 Modifying the backup policy



Step 4 Check or manage the generated backups on the **Backups** or **Backups & Restorations** page.

----End

Viewing Incremental Backups

You can view incremental backups and their size of a GeminiDB Cassandra instance.

- To view the size and records of incremental backups, choose **Service Tickets > Create Service Ticket** in the upper right corner of the console and contact the customer service.
- You can view incremental backups and their size only after you enable **Incremental Backup**, or no data is displayed.

Step 1 Log in to the Huawei Cloud console.

Step 2 On the **Instances** page, click the target instance name to access the **Basic Information** page.

Step 3 In the navigation pane, choose **Backups & Restorations**.

Step 4 On the **Backups & Restorations** page, click **Incremental Backup**.

Step 5 View incremental backups and their size.

----End

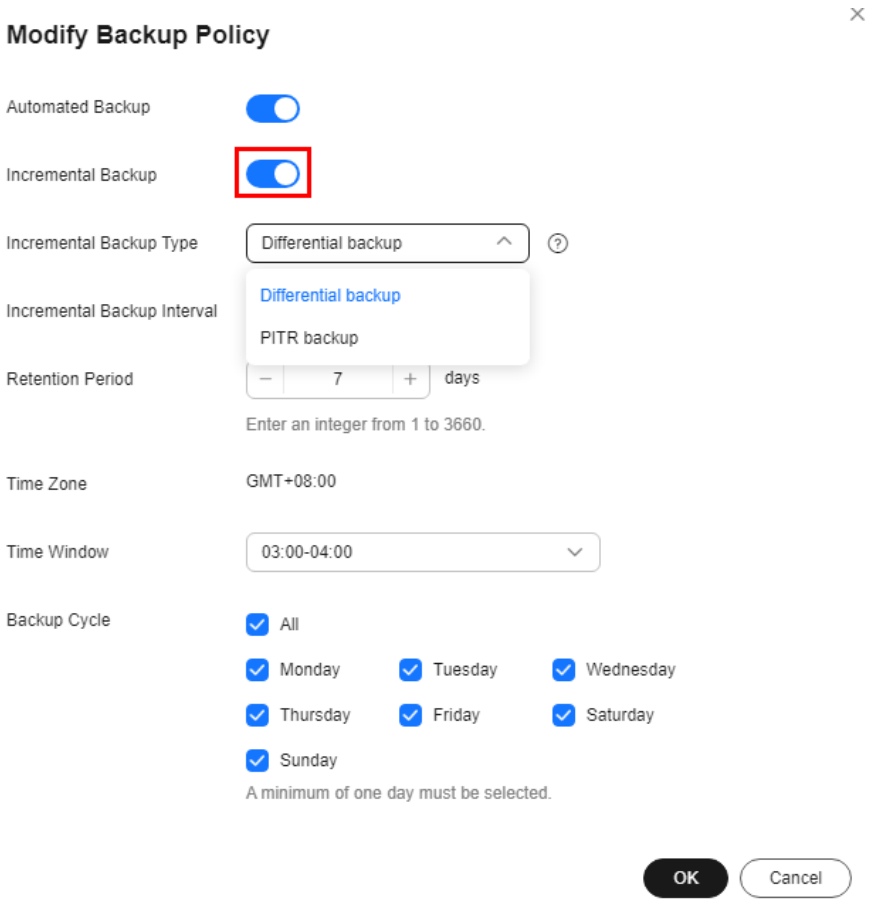
Disabling Incremental Backup

Step 1 Log in to the Huawei Cloud console.

Step 2 On the **Instances** page, click the target instance. The **Basic Information page** is displayed.

Step 3 In the navigation pane, choose **Backups & Restorations**. On the displayed page, click **Modify Backup Policy** and click  next to **Incremental Backup**.

Figure 3-60 Disabling **Incremental Backup**



----End

Disabling an Automated Backup Policy

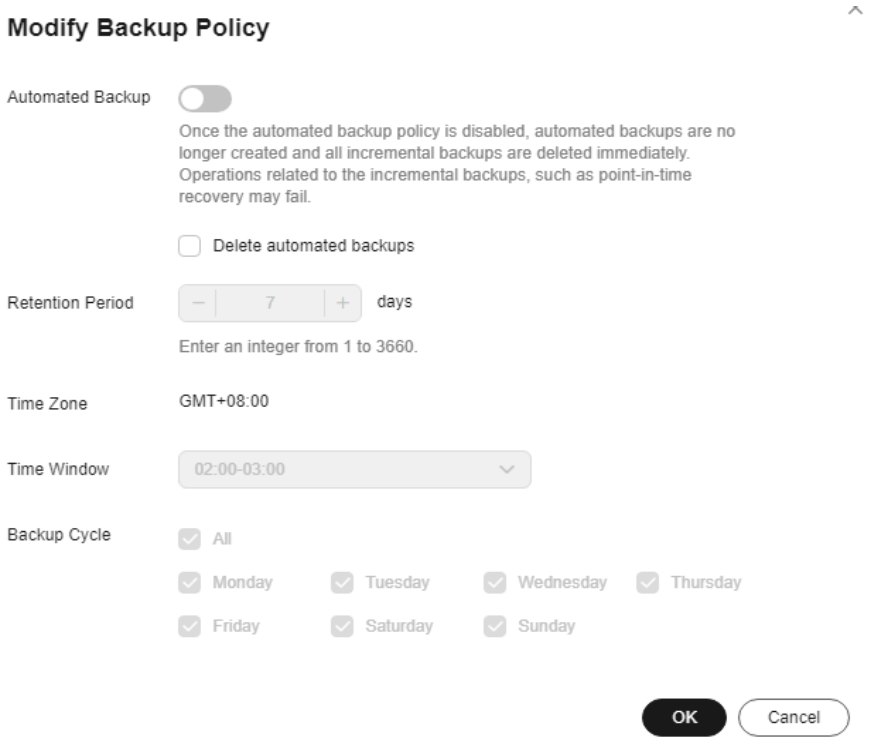
Step 1 Log in to the Huawei Cloud console.

Step 2 On the **Instances** page, locate the instance that you want to disable automated backup for and click its name.

Step 3 Choose **Backups & Restorations** in the navigation pane one the left, and click **Modify Backup Policy**.

Step 4 In the displayed dialog box, click  and click **OK**.

Figure 3-61 Disabling backup policies



When your disable automated backup, specify whether to delete the automated backups:

- If you select **Delete automated backups**, all backup files within the retention period will be deleted. There are no automated backups displayed until you enable automated backup again.
- If you do not select **Delete automated backups**, backup files within the retention period will be retained, but you can still manually delete them later if needed. For details, see **Deleting an Automated Backup**.

If **Automated Backup** is disabled, any automated backups in progress stop immediately.

----End

Deleting an Automated Backup

If automated backup is disabled, you can delete stored automated backups to free up storage space.

If automated backup is enabled, the system will delete automated backups when they expire. You cannot delete them manually.

CAUTION

Deleted backups cannot be restored.

- **Method 1**
  a. Log in to the Huawei Cloud console.
  b. On the **Instances** page, click the instance whose automated backups you want to delete and click its name.
  c. Choose **Backups & Restorations** in the navigation pane on the left, locate the backup you want to delete and click **Delete** in the **Operation** column.
  d. In the displayed dialog box, confirm the backup details and click **Yes**.
- **Method 2**
  a. Log in to the Huawei Cloud console.
  b. On the **Backups** page, locate the backup that you want to delete and click **Delete** in the **Operation** column.
  c. In the displayed dialog box, confirm the backup details and click **Yes**.

3.9.3 Managing Manual Backups

To ensure data reliability, GeminiDB Cassandra API allows you to manually back up instances whose status is **Available**. If a database or table is deleted, maliciously or accidentally, backups can help recover your data.

Usage Notes

- Manual backups are full backups.
- Backup files are saved as packages in OBS buckets. Upload of backup files and service reads both consume bandwidth, so the upload bandwidth of OBS is limited. The bandwidth of a single node ranges from 20 MB/s to 70 MB/s.

  For better performance, you need to specify appropriate nodes for an instance and take into account the bandwidth for uploading backups.
- The CPU usage may increase 5% to 15% because uploading backups consumes CPU resources.
- The memory usage may increase by about 300 MB during the upload of backups. The increase depends on the instance's data volume. The increased memory mainly caches data during backup upload and service read. After the backup upload is complete, the memory recovers.
- Manual backups are billed.

Creating a Manual Backup

Step 1 Log in to the Huawei Cloud console.

Step 2 Create a manual backup.

Method 1

On the **Instances** page, locate the instance that you want to create a backup for and choose **More** > **Create Backup** in the **Operation** column.

Method 2

1. On the **Instances** page, click the instance that you want to create a backup for and click its name.

2. Choose **Backups & Restorations** in the navigation pane on the left, click **Create Backup**.

Method 3

In the navigation pane on the left, choose **Backups** and click **Create Backup**.

Step 3 In the displayed dialog box, enter a backup name and description and click **OK**.

Figure 3-62 Creating a backup

Table 3-27 Parameter description

Parameter	Description
DB Instance Name	Must be the name of the DB instance to be backed up and cannot be modified.
Backup Name	Must be 4 to 64 characters in length and start with a letter. It is case-insensitive and contains only letters, digits, hyphens (-), and underscores (_).
Description	Contains a maximum of 256 characters and cannot include line breaks or special characters >!<"&'=

Step 4 View the backup status.

- When the backup is being created, query the backup status on the **Backups** or **Backups & Restorations** page. The backup status is **Backing up**.
- After the backup is created, the backup status is **Completed**.

----End

Deleting a Manual Backup

If you no longer need a manual backup, delete it on the **Backups** or **Backups & Restorations** page.

Deleted backups are not displayed in the backup list.

> **CAUTION**
>
> Deleted backups cannot be restored.

Method 1

1. Log in to the Huawei Cloud console.
2. On the **Instances** page, locate the instance whose backup you want to delete and click its name.
3. Choose **Backups & Restorations** in the navigation pane on the left, locate the backup you want to delete and click **Delete** in the **Operation** column.
4. In the displayed dialog box, confirm the backup details and click **Yes**.

Method 2

1. Log in to the Huawei Cloud console.
2. On the **Backups** page, locate the backup you want to delete and click **Delete** in the **Operation** column.
3. In the displayed dialog box, confirm the backup details and click **Yes**.

3.9.4 Managing Cross-Region Backups

GeminiDB Cassandra allows you to store backups in the destination region or OBS buckets. Then for disaster recovery, you can restore the backups to a new instance in another region.

After a cross-region backup policy is set for an instance, the system will synchronize backups of the instance to the destination region you specified. You can manage cross-region backup files on the **Backups** page.

Usage Notes

- To enable the cross-region backup function, choose **Service Tickets > Create Service Ticket** in the upper right corner of the console and contact the customer service.
- Before you configure a cross-region backup policy, make sure to enable automated backup first. Otherwise, the cross-region backup policy cannot take effect. For details, see **Modifying an Automated Backup Policy**.
- Only automated full backups can be created across regions.

Billing

Table 3-28 Billing

Flavor	Billing Item	Unit Price
geminidb.cassandra.crossreg.backup.space.dfv	Storage space	CNY0.0009/GB/hour

Flavor	Billing Item	Unit Price
geminidb.cassandra.crossreg.backup.flow	Cross-region backup traffic	CNY0.5/GB

Setting or Modifying a Cross-Region Backup Policy

Step 1 Log in to the Huawei Cloud console.

Step 2 On the **Instances** page, locate the instance that you want to connect to and click its name.

Step 3 In the navigation pane on the left, choose **Backups & Restorations**.

Step 4 On the displayed page, click **Set Cross-Region Backup Policy**.

Step 5 In the displayed dialog box, set required parameters.

Figure 3-63 Setting a cross-region backup policy

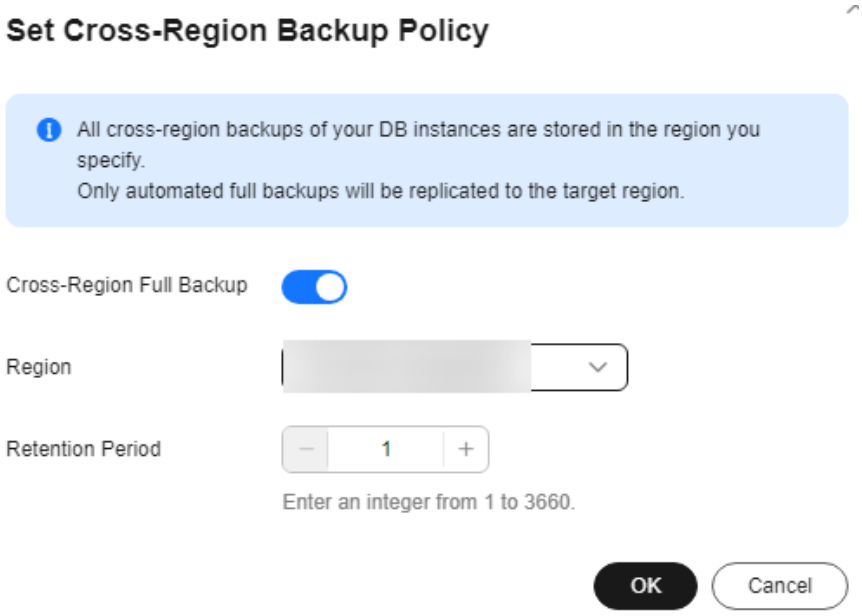


Table 3-29 Description

Parameter	Description
Cross-Region Full Backup	If you enable **Cross-Region Full Backup**, automated full backup files of the instance will be stored in the region you specify.
Region	You can select the region for storing backups based on service requirements.
Retention Period	Number of days that cross-region backups are kept. The value ranges from **1** to **1825**. You can increase the retention period to improve data reliability.

NOTE

- Only new backups generated after you set a cross-region backup policy will be stored in the region you specify.
- All cross-region backups of your DB instances are stored in the same region you specify.
- Cross-region backups are synchronized to the destination region you specify only after your instance is backed up locally.
- Only automated full backups are replicated to the destination region.

Step 6 Click **OK**.

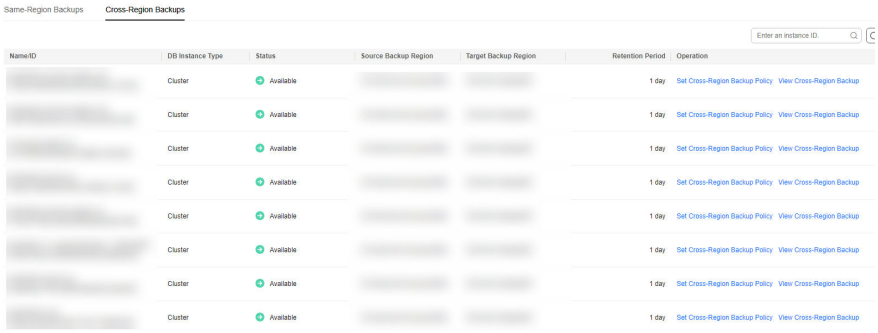
----End

Managing Cross-Region Backups

Step 1 Log in to the Huawei Cloud console.

Step 2 In the navigation pane, choose **Backups**. Click the **Cross-Region Backups** tab.

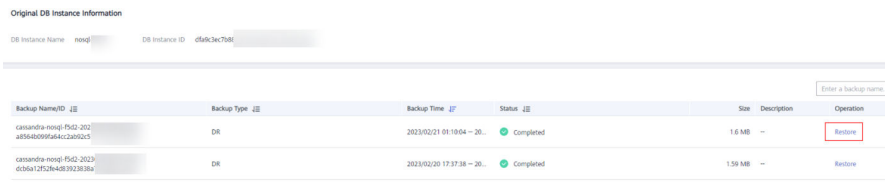
Figure 3-64 Cross-region backups



All cross-region backups are displayed by default.

- To modify the cross-region backup policy, click **Set Cross-Region Backup** in the **Operation** column.
- To view all cross-region backups, click **View Cross-Region Backup**. To restore a backup to a new instance, click **Restore** in the **Operation** column. For details, see **Restoring Data to a New Instance**.

Figure 3-65 Restoring a cross-region backup



NOTE

- Cross-region full backup replicates only automated full backups to another region for full restoration. Any of the full backups can be restored to a new instance that has no relationships with the original instance.
- The new instance uses the same parameter group as the original instance.
- During the instance restoration, backup files are downloaded from OBS buckets to the data directory of the new instance. The download bandwidth of OBS is 40 MB/s.
- If the original instance has cross-region backup disabled, the restoration may fail.
- Cross-region backup is not supported for instances with disk encryption enabled.

----End

Disabling Cross-Region Backup

Step 1 Log in to the Huawei Cloud console.

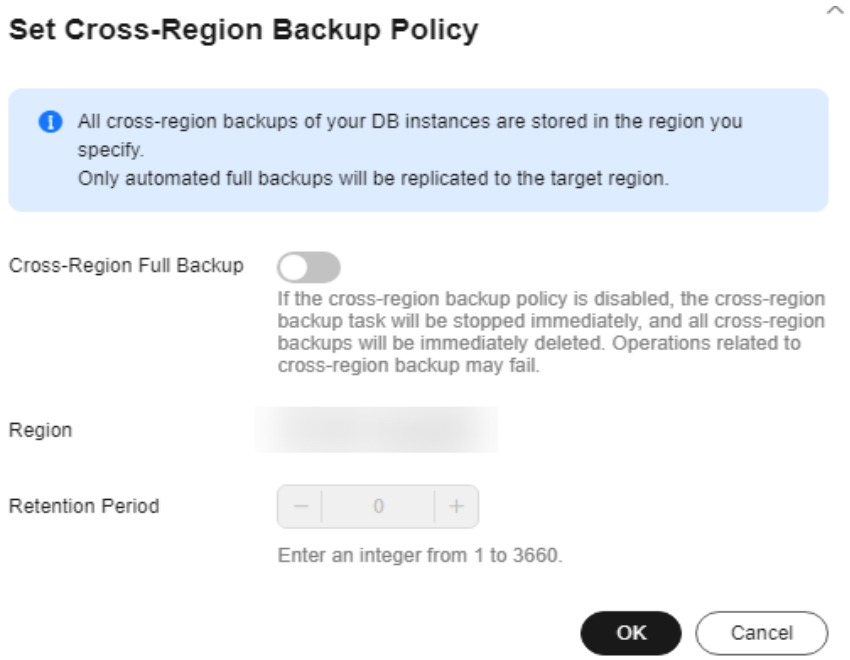
Step 2 On the **Instances** page, locate the instance that you want to connect to and click its name.

Step 3 In the navigation pane on the left, choose **Backups & Restorations**.

Step 4 On the displayed page, click **Set Cross-Region Backup Policy**.

Step 5 In the displayed dialog box, disable **Cross-Region Full Backup**.

Figure 3-66 Disabling cross-region backup



NOTE

- After cross-region backup is disabled, the cross-region backup task is stopped and all cross-region backups are deleted immediately. As a result, operations using cross-region backups will fail.
- If an instance with cross-region backup enabled is deleted, its cross-region backups will be retained. The retention period depends on settings of the cross-region backup policy.

Step 6 Click **OK**.

----End

3.10 Data Restoration

3.10.1 Restoration Methods

GeminiDB Cassandra API supports multiple forms of data restoration. You can select one based on service requirements.

Table 3-30 Restoration methods

Method	Scenario
Rebuilding an Instance	If an instance is deleted by mistake, you can rebuild it within a retention period in the recycle bin.
Restoring Data to a New Instance	You can restore an existing backup file to a new instance.
Restoring a Backup to a Specified Point in Time	You can use an automated backup to restore an instance to a specified point in time.

3.10.2 Restoring Data to a New Instance

GeminiDB Cassandra API allows you to use an existing backup to restore data to a new instance.

Precautions

- The new instances must have at least as many nodes as the original instance.
- The new instance must have at least as much storage as the original instance.
- Restoration to the current instance is not supported.
- You can scale in the memory, but the memory decrease cannot become less than the actual memory used during the backup.
- The restored instance uses the same parameter group as the original instance.
- During the instance restoration, backups are downloaded from OBS buckets to the data directory of the restored instance. The download bandwidth of OBS is 40 MB/s.

Procedure

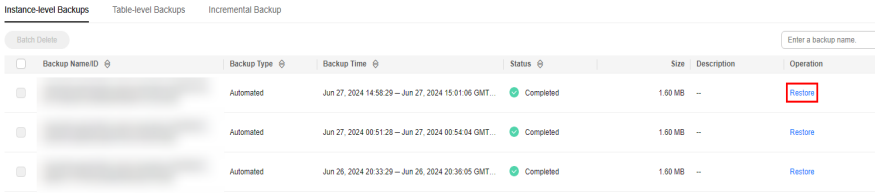
Step 1 Log in to the Huawei Cloud console.

Step 2 Restore an instance from backup.

Method 1

1. On the **Instances** page, locate the instance whose backup you want to restore and click its name.
2. Choose **Backups & Restorations** in the navigation pane on the left, locate the backup that you want to restore and click **Restore** in the **Operation** column.

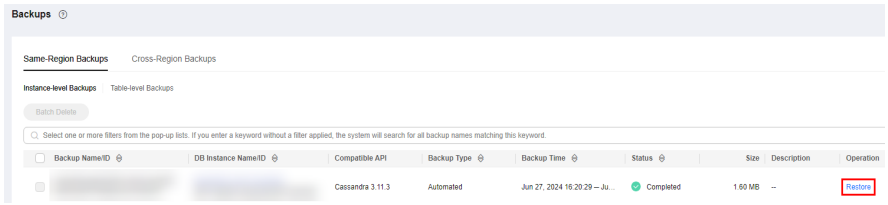
Figure 3-67 Backups and restorations



Method 2

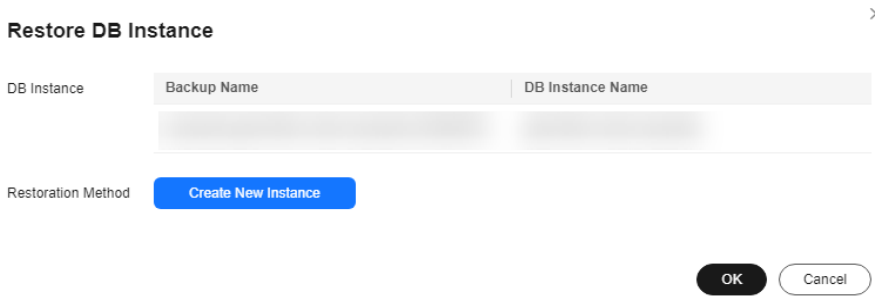
On the **Backups** page, locate the backup that you want to restore and click **Restore** in the **Operation** column.

Figure 3-68 Backup management



Step 3 In the displayed dialog box, confirm the current instance details and restoration method and click **OK**.

Figure 3-69 Restoring data to a new instance



- The default API type and DB engine version are the same as those of the original instance and cannot be changed.

- GeminiDB automatically calculates the minimum storage space required for restoration based on the size of the selected backup file. The storage capacity depends on the instance specifications, and must be an integer.
- The administrator password needs to be reset.
- To modify other parameters, see the description of buying instances of other DB engines in *Getting Started*.

Step 4 View the results.

A new instance is created using the backup data. The status of the new instance changes from **Creating** to **Available**.

After the restoration, the system will perform a full backup.

The new instance is independent from the original one.

----End

3.10.3 Restoring a Backup to a Specified Point in Time

You can restore an existing automated backup to a specified point in time.

The most recent full backup will be downloaded from OBS for restoration. After the restoration is complete, incremental backups will be replayed to the specified point in time. The time required depends on the amount of data to be restored.

Usage Notes

- This function is only available to new GeminiDB Cassandra instances.
- After automated backup is enabled, the system performs an incremental backup based on the preset incremental backup interval. The incremental backup is stored in OBS.
- Data can be restored to a specified time point only after the automated backup policy is enabled.
- During the instance restoration, backup files are downloaded from OBS buckets to the data directory of the restored instance. The download bandwidth of OBS is 40 MB/s.
- This function is now in OBT. To use it, choose **Service Tickets > Create Service Ticket** in the upper right corner of the console and contact the customer service.

Procedure

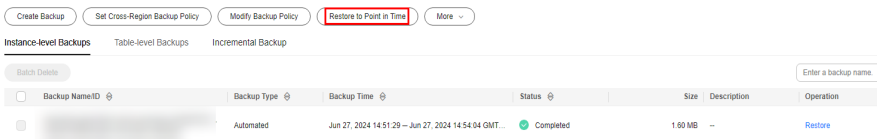
Step 1 Log in to the Huawei Cloud console.

Step 2 On the **Instances** page, click the instance that you want to perform a PITR for.

Step 3 In the navigation pane on the left, choose **Backups & Restorations**.

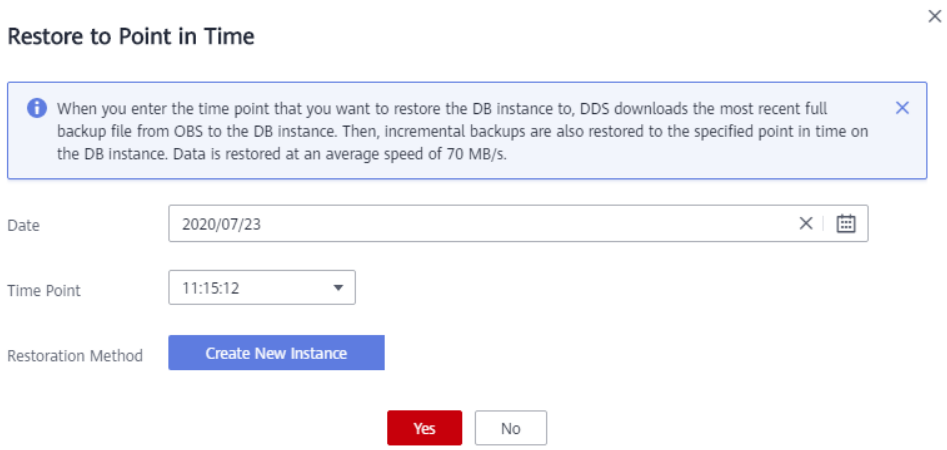
Step 4 On the **Backups & Restorations** page, click **Restore to Point in Time**.

Figure 3-70 Restoring data to a point in time



Step 5 Select a restoration date and a time point and click **OK**.

Figure 3-71 Restore to Point in Time



Step 6 On the **Create New Instance** page, create an instance of the same specifications as the instance to be restored. The new instance is independent from the original one.

- The new instance should be deployed in a different AZ to ensure that your applications will not be affected by SPOFs.
- The compatible API, instance type, instance version, and CPU type are the same as those of the original and cannot be changed.
- Other settings are the same as those of the original instance by default but can be modified. For details, see **Buying a GeminiDB Cassandra Instance**.

----End

3.11 Parameter Management

3.11.1 Modifying Parameters of GeminiDB Cassandra Instances

You can modify parameters in a custom parameter template so that your instance can deliver spectacular performance.

Usage Notes

- Note that parameter values in default parameter templates cannot be changed.
- Though parameter values in a default template cannot be changed, you can view details about a default parameter template.
- If a custom parameter template is set incorrectly, the database startup may fail. You can re-configure the custom parameter template according to the configurations of the default parameter template.

CAUTION

Exercise caution when modifying parameter values to prevent exceptions.

Modifying Custom Parameters and Applying the Modifications to an Instance

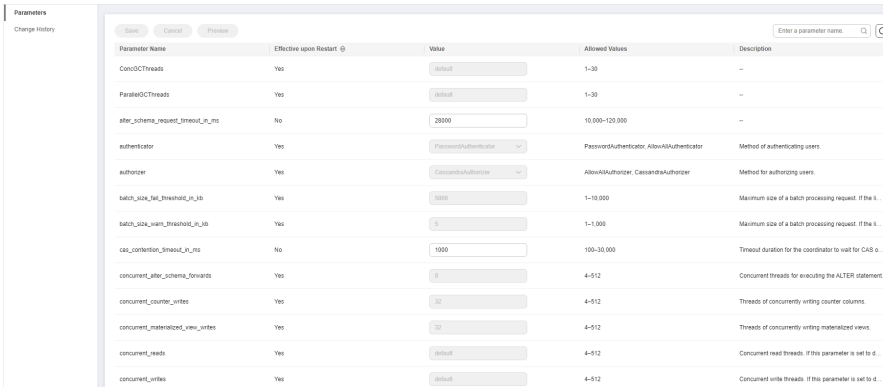
Step 1 Log in to the Huawei Cloud console.

Step 2 In the navigation pane on the left, choose **Parameter Templates**.

Step 3 Click the **Custom Templates** tab, locate the parameter template you want to modify, and click its name.

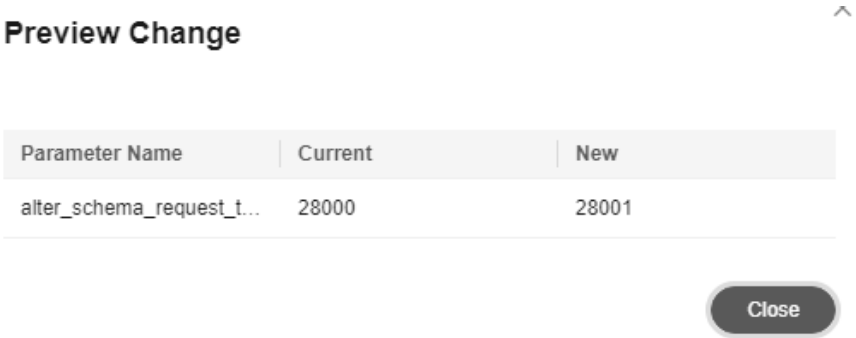
Step 4 Change parameter values as required.

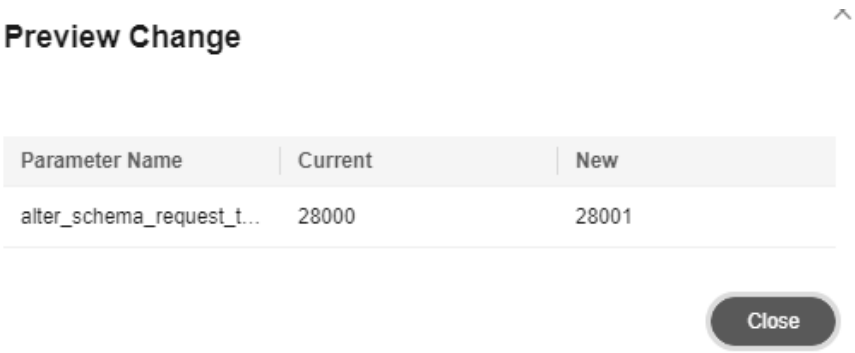
Figure 3-72 Modifying parameters in a parameter template



- To save the modifications, click **Save**.
- To cancel the modifications, click **Cancel**.
- To preview the modifications, click **Preview**.

Figure 3-73 Previewing changes





Step 5 After parameters are modified, click **Change History** to view parameter modification details.

For details about how to view parameter modification details, see **Viewing Parameter Change History**.

NOTE

- You need to manually apply the modifications to the current instance. For details, see **Applying a Parameter Template**.
- The change history page displays only modifications in the last seven days.

----End

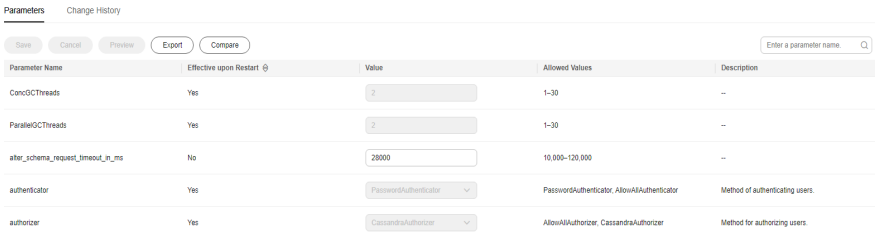
Modifying Parameters of an Instance

Step 1 Log in to the Huawei Cloud console.

Step 2 In the navigation pane on the left, choose **Instances**. On the displayed page, locate the instance whose parameters you want to modify and click its name.

Step 3 In the navigation pane on the left, choose **Parameters**. On the displayed page, modify parameters as required.

Figure 3-74 Parameters



- To save the modifications, click **Save**.
- To cancel the modifications, click **Cancel**.
- To preview the modifications, click **Preview**.

Step 4 After parameters are modified, click **Change History** to view parameter modification details.

For details about how to view parameter modification details, see **Viewing Parameter Change History**.

NOTE

The modifications are immediately applied to the current instance.

Check the value in the **Effective upon Restart** column.

- If the value is **Yes** and the instance status on the **Instances** page is **Pending restart**, restart the instance to apply the modifications.
- If the value is **No**, the modifications are applied immediately.

----End

3.11.2 Creating a Parameter Template

You can use database parameter templates to manage DB API configurations. A database parameter template acts as a container for API configuration values that can be applied to one or more DB instances.

Each user can create up to 100 parameter templates. All types of instances in the same project can share the quota.

Procedure

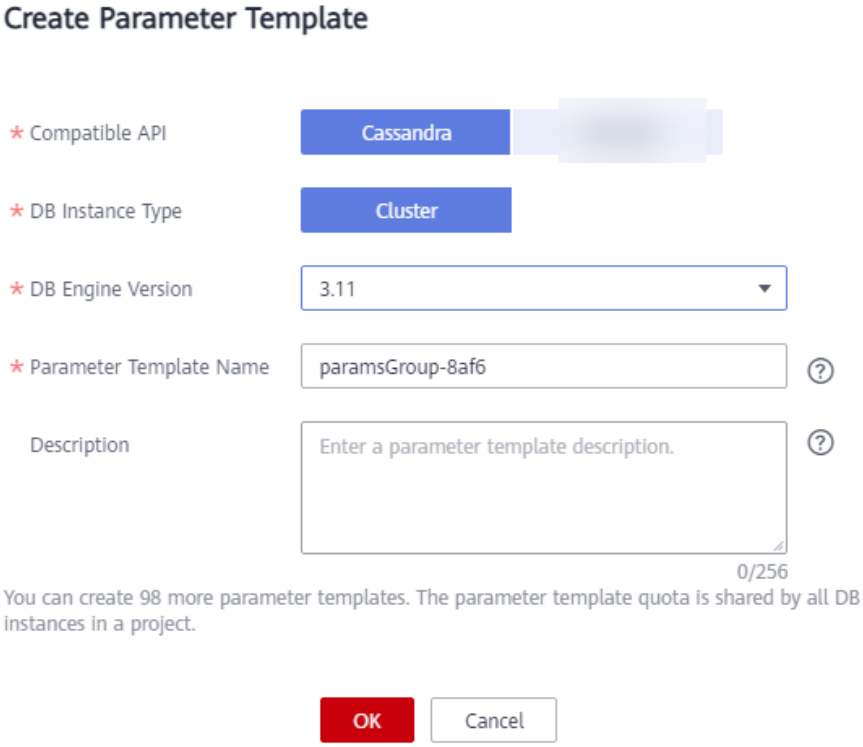
Step 1 Log in to the Huawei Cloud console.

Step 2 In the navigation pane on the left, choose **Parameter Templates**.

Step 3 On the **Parameter Templates** page, click **Create Parameter Template**.

Step 4 Select a compatible API, specify a DB engine version and a parameter group description, and click **OK**.

Figure 3-75 Creating a parameter template



- **Compatible API**: Select the API type and instance type that are compatible with your DB API parameter template.
- **DB Engine Version**: Select a DB engine version, for example, 3.11.
- **Parameter Template Name**: The template name can include 1 to 64 characters. It can contain only uppercase letters, lowercase letters, digits, hyphens (-), underscores (_), and periods (.).
- **Description**: The description contains a maximum of 256 characters and cannot include line breaks or the following special characters >!<"&'=

Step 5 On the **Parameter Templates** page, view the created parameter template.

----End

3.11.3 Viewing Parameter Change History

Scenarios

You can view parameter change history of an instance or one of its custom parameter templates based on service requirements.

Precautions

In a newly exported or created parameter template, change history is left blank.

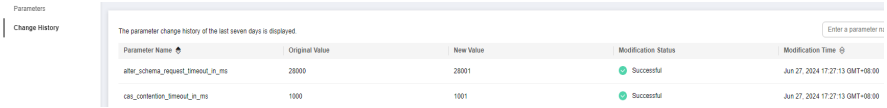
Viewing Change History of a Custom Parameter Template

Step 1 Log in to the Huawei Cloud console.

Step 2 In the navigation pane on the left, choose **Parameter Templates**. On the **Custom Templates** page, click the parameter template whose change history you want to view.

Step 3 In the navigation pane on the left, choose **Change History**. Then, view the name, original value, new value, modification status, and modification time of the target parameter.

Figure 3-76 Viewing change history of a customer parameter template



You can apply the parameter template to instances by referring to **Applying a Parameter Template**.

----End

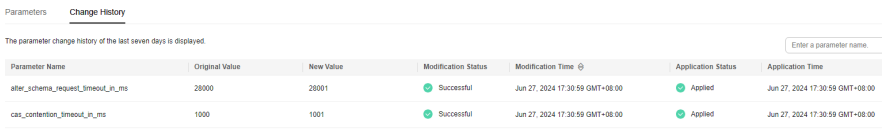
Viewing Parameter Change History of an Instance

Step 1 Log in to the Huawei Cloud console.

Step 2 On the **Instances** page, locate the instance whose parameter change history you want to view and click its name.

Step 3 In the navigation pane on the left, choose **Parameters**. On the **Change History** page, view the name, original value, new value, modification status, and modification time of the target parameter.

Figure 3-77 Viewing parameter change history of an instance



----End

3.11.4 Exporting a Parameter Template

- You can export a parameter template of a DB instance for future use. To learn how to apply the exported parameter template to a DB instance, refer to section **Applying a Parameter Template**.
- You can export the parameter template details (parameter names, values, and descriptions) of a DB instance to a CSV file for review and analysis.

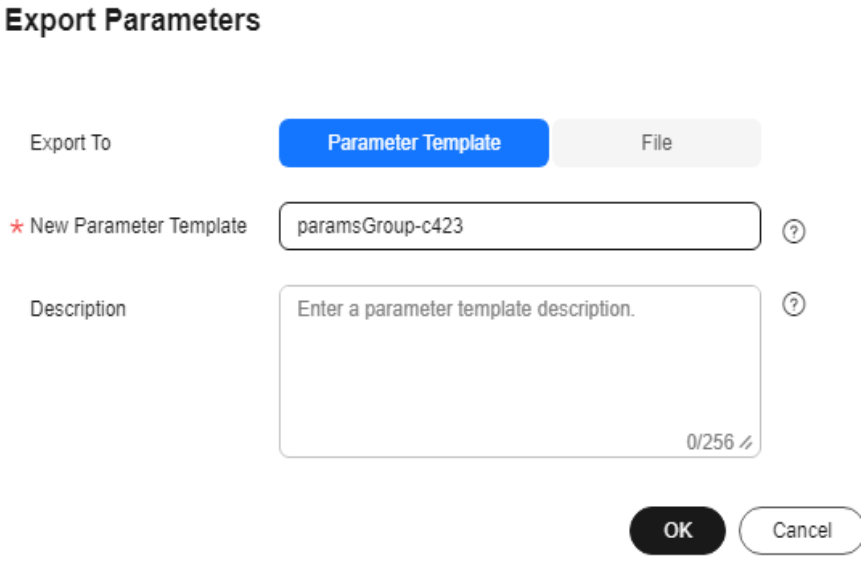
Procedure

Step 1 Log in to the Huawei Cloud console.

Step 2 In the navigation pane on the left, choose **Instances**, locate the instance whose parameters you want to export, and click its name.

Step 3 In the navigation pane on the left, choose **Parameters**. On the **Parameters** tab, above the parameter list, click **Export**.

Figure 3-78 Exporting a parameter template



- **Parameter Template**: You can export the parameters of the DB instance to a template for future use.

  In the displayed dialog box, configure required details and click **OK**.

  NOTE

  - **Parameter Template Name**: The template name can be 1 to 64 characters long. It can contain only uppercase letters, lowercase letters, digits, hyphens (-), underscores (_), and periods (.).
  - **Description**: The template description consists of a maximum of 256 characters and cannot include line breaks or the following special characters: >!<"&'=

  After the parameter template is exported, a new template is generated in the list on the **Parameter Templates** page.
- **File**: You can export the parameter template details (parameter names, values, and descriptions) of a DB instance to a CSV file for review and analysis.

  In the displayed dialog box, enter the file name and click **OK**.

  NOTE

  The file name must start with a letter and consist of 4 to 81 characters. It can contain only letters, digits, hyphens (-), and underscores (_).

----End

3.11.5 Comparing Parameter Templates

This section describes how to compare two parameter templates of the same instance type and compatible API to learn about their configurations.

Comparing Parameter Templates

Step 1 Log in to the Huawei Cloud console.

Step 2 In the navigation pane on the left, choose **Parameter Templates**.

Step 3 In the parameter template list, locate the parameter template that you created and click **Compare** in the **Operation** column.

Step 4 In the displayed dialog box, select a parameter template that is of the same instance type and compatible API as the selected template and click **OK**.

Figure 3-79 Comparing two parameter templates

- If their parameters are different, the different parameter names and values are displayed.

- If their parameters are the same, no data is displayed.

----End

Comparing Parameter Templates of a Specified Instance

Step 1 Log in to the Huawei Cloud console.

Step 2 In the navigation pane on the left, choose **Instances**.

Step 3 In the instance list, locate the instance whose parameter templates you want to compare and click its name.

Step 4 In the navigation pane on the left, choose **Parameters** and then click **Compare** above the parameter list.

Step 5 In the displayed dialog box, select a parameter template that is of the same instance type as the template of current instance and click **OK**.

Figure 3-80 Comparing two parameter templates

- If their parameters are different, the different parameter names and values are displayed.
- If their parameters are the same, no data is displayed.

----End

3.11.6 Replicating a Parameter Template

You can replicate a parameter template you have created. When you have already created a parameter template and want to include most of the custom parameters and values from that template in a new parameter template, you can replicate that parameter template. You can also export a parameter template of a DB instance for future use.

Default parameter templates cannot be replicated, but you can create parameter templates based on the default templates provided.

Procedure

Step 1 Log in to the Huawei Cloud console.

Step 2 In the navigation pane on the left, choose **Parameter Templates**.

Step 3 On the **Parameter Templates** page, click the **Custom Templates** tab. Locate the target parameter template and click **Replicate** in the **Operation** column.

Alternatively, click the target DB instance on the **Instances** page. On the **Parameters** page, click **Export** to generate a new parameter template for future use.

Step 4 In the displayed dialog box, enter the parameter template name and description and click **OK**.

Figure 3-81 Replicating a parameter template

- **New Parameter Template**: The template name can be up to 64 characters long. It can contain only uppercase letters, lowercase letters, digits, hyphens (-), underscores (_), and periods (.).
- **Description**: The description contains a maximum of 256 characters and cannot include line breaks or the following special characters >!<"&'=

After the parameter template is replicated, a new template is generated in the list on the **Parameter Templates** page.

----End

3.11.7 Resetting a Parameter Template

You can reset all parameters in a custom parameter template to their default settings.

Procedure

Step 1 Log in to the Huawei Cloud console.

Step 2 In the navigation pane on the left, choose **Parameter Templates**.

Step 3 On the **Parameter Templates** page, click the **Custom Templates** tab. Locate the target parameter template and choose **More** > **Reset** in the **Operation** column.

Step 4 Click **Yes** to reset the parameter template.

----End

3.11.8 Applying a Parameter Template

GeminiDB Cassandra API allows you to apply a parameter template. Modifications to parameters in a custom parameter template take effect only after you have applied the template to the target instance.

Procedure

Step 1 Log in to the Huawei Cloud console.

Step 2 In the navigation pane on the left, choose **Parameter Templates**.

Step 3 On the **Parameter Templates** page, perform the following operations based on the template type:

- To apply a default template, click **Default Templates**, locate the template, and in the **Operation** column, click **Apply**.
- To apply a custom template, click **Custom Templates**, locate the target parameter template, and choose **More** > **Apply** in the **Operation** column.

A parameter template can be applied to one or more instances.

Step 4 In the displayed dialog box, select one or more instances that the parameter template will be applied to and click **OK**.

After a parameter template is applied, you can **view its application records**.

----End

3.11.9 Viewing Application Records of a Parameter Template

GeminiDB Cassandra allows you to view application records of a parameter template.

Procedure

Step 1 Log in to the Huawei Cloud console.

Step 2 In the navigation pane on the left, choose **Parameter Templates**.

Step 3 On the **Parameter Templates** page, locate the parameter template whose application records you want to view and choose **More** > **View Application Record** in the **Operation** column.

You can view the name or ID of the instance that the parameter template applies to, as well as the application status, application time, and causes of any failures that have occurred.

----End

3.11.10 Modifying a Parameter Template Description

You can modify the description of a custom parameter template if needed.

Procedure

Step 1 Log in to the Huawei Cloud console.

Step 2 In the navigation pane on the left, choose **Parameter Templates**.

Step 3 On the **Parameter Templates** page, click the **Custom Templates** tab. Locate the target parameter template and click  in the **Description** column.

Step 4 Enter a new description. You can click  to submit or  to cancel the modification.

- After you submit the modification, you can view the new description in the **Description** column on the **Parameter Templates** page.
- The description can include up to 256 characters but cannot contain the following special characters: >!<"&'=

----End

3.11.11 Deleting a Parameter Template

You can delete a custom parameter template that is no longer in use.

Precautions

- Deleted templates cannot be recovered, so exercise caution when performing this operation.
- Default parameter templates cannot be deleted.

Procedure

Step 1 Log in to the Huawei Cloud console.

Step 2 In the navigation pane on the left, choose **Parameter Templates**.

Step 3 On the **Parameter Templates** page, click the **Custom Templates** tab. Locate the parameter template you want to delete and choose **More** > **Delete** in the **Operation** column.

Step 4 Click **Yes** to delete the parameter template.

----End

3.12 CTS

3.12.1 Key Operations Supported by CTS

With CTS, you can record operations on GeminiDB Cassandra instances for later queries, audit, and backtracking.

Table 3-31 GeminiDB Cassandra key operations

Operation	Resource Type	Trace Name
Creating an instance	instance	NoSQLCreateInstance
Deleting an instance	instance	NoSQLDeleteInstance
Adding nodes	instance	NoSQLEnlargeInstance
Deleting nodes	instance	NoSQLReduceInstance
Restarting an instance	instance	NoSQLRestartInstance
Restoring data to a new instance	instance	NoSQLRestoreNewInstance
Scaling up storage space of an instance	instance	NoSQLExtendInstanceVolume
Resetting the password of an instance	instance	NoSQLResetPassword
Modifying the name of an instance	instance	NoSQLRenameInstance
Changing specifications	instance	NoSQLResizeInstance
Binding an EIP	instance	NoSQLBindEIP
Unbinding an EIP	instance	NoSQLUnBindEIP
Freezing an instance	instance	NoSQLFreezeInstance
Unfreezing an instance	instance	NoSQLUnfreezeInstance
Creating a backup	backup	NoSQLCreateBackup
Deleting a backup	backup	NoSQLDeleteBackup
Modifying the backup policy of an instance	backup	NoSQLSetBackupPolicy
Adding a tag for an instance	tag	NoSQLAddTags
Modifying an instance tag	tag	NoSQLModifyInstanceTag
Deleting an instance tag	tag	NoSQLDeleteInstanceTag
Creating a parameter template	parameterGroup	NoSQLCreateConfigurations
Modifying a parameter template	parameterGroup	NoSQLUpdateConfigurations
Modifying instance parameters	parameterGroup	NoSQLUpdateInstanceConfigurations

Operation	Resource Type	Trace Name
Replicating a parameter template	parameterGroup	NoSQLCopyConfigurations
Resetting a parameter template	parameterGroup	NoSQLResetConfigurations
Applying a parameter template	parameterGroup	NoSQLApplyConfigurations
Deleting a parameter template	parameterGroup	NoSQLDeleteConfigurations
Deleting the node that fails to be added	instance	NoSQLDeleteEnlargeFail-Node
Changing the security group of an instance	instance	NoSQLModifySecurityGroup
Configuring autoscaling	instance	NoSQLModifyAutoEnlarge-Policy
Exporting parameter template information for an instance	instance	NoSQLSaveConfigurations
Modifying the recycling policy	instance	NoSQLModifyRecyclePolicy

3.12.2 Querying Traces

After CTS is enabled, CTS starts recording operations on cloud resources. The CTS console stores the last seven days of operation records.

This section describes how to query the last seven days of operation records on the CTS console.

Procedure

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

Step 2 Click in the upper left corner and select a region and project.

Step 3 In the navigation pane on the left, click **Trace List**.

Step 4 Specify filter criteria to search for the required traces. The following four filter criteria are available:

- **Trace Source**, **Resource Type**, and **Search By**

  Select filters from the drop-down list.

  When you select **Trace name** for **Search By**, you need to select a specific trace name.

When you select **Resource ID** for **Search By**, you also need to select or enter a specific resource ID.

When you select **Resource name** for **Search By**, you also need to select or enter a specific resource name.

- **Operator**: Select a specific operator (a user other than the tenant).
- **Trace Status**: Select **All trace statuses**, **Normal**, **Warning**, or **Incident**.
- **Start Date** and **End Date**: You can specify a time range to query traces.

Step 5 Locate the required trace and click ⌄ on the left of the trace to view details.

Step 6 Click **View Trace** in the **Operation** column. In the displayed dialog box, the trace structure details are displayed.

----End

3.13 Viewing Metrics and Configuring Alarms

3.13.1 Supported Metrics

This section describes GeminiDB Cassandra API metrics reported to Cloud Eye as well as their namespaces and dimensions. You can use APIs provided by Cloud Eye to query the metrics and alarms.

Namespace

SYS.NoSQL

Metrics

NOTE

You can view metrics on instance nodes by referring to **Viewing Metrics**.

Table 3-32 GeminiDB Cassandra API metrics

Metric ID	Name	Description	Value Range	Unit	Number System	Monitored Object	Monitoring Period (Raw Data)
nosql005_disk_usage	Storage Space Usage	Storage space usage of the monitored object.	0–100	%	N/A	GeminiDB Cassandra instances	1 minute

Metric ID	Name	Description	Value Range	Unit	Number System	Monitored Object	Monitoring Period (Raw Data)
nosql006_disk_total_size	Total Storage Space	Total storage space of the monitored object.	≥ 0	GB	1024(IEC)	GeminiDB Cassandra instances	1 minute
nosql007_disk_used_size	Used Storage Space	Used storage space of the monitored object.	≥ 0	GB	1024(IEC)	GeminiDB Cassandra instances	1 minute
nosql009_dfv_write_delay	Storage Write Latency	Average delay of writing data to the storage layer in a specified period	≥ 0	ms	N/A	GeminiDB Cassandra instance nodes	1 minute
nosql010_dfv_read_delay	Storage Read Latency	Average latency of reading data from the storage layer in a specified period	≥ 0	ms	N/A	GeminiDB Cassandra instance nodes	1 minute
cassandra001_cpu_usage	CPU Usage	CPU usage of an instance	0–100	%	N/A	GeminiDB Cassandra instance nodes	1 minute
cassandra002_mem_usage	Memory Usage	Memory usage of the instance	0–100	%	N/A	GeminiDB Cassandra instance nodes	1 minute
cassandra003_bytes_out	Network Output Throughput	Outgoing traffic in bytes per second	≥ 0 Bytes/s	Bytes/s	1024(IEC)	GeminiDB Cassandra instance nodes	1 minute
cassandra004_bytes_in	Network Input Throughput	Incoming traffic in bytes per second	≥ 0	Bytes/s	1024(IEC)	GeminiDB Cassandra instance nodes	1 minute

Metric ID	Name	Description	Value Range	Unit	Number System	Monitored Object	Monitoring Period (Raw Data)
cassandra014_connections	Active Node Connections	Total number of connections attempting to connect to Cassandra instance nodes	≥ 0	Counts	N/A	GeminiDB Cassandra instance nodes	1 minute
cassandra015_read_latency	Average Read Latency	Average amount of time consumed by read requests	≥ 0	ms	N/A	GeminiDB Cassandra instance nodes	1 minute
cassandra016_write_latency	Average Write Latency	Average amount of time consumed by write requests	≥ 0	ms	N/A	GeminiDB Cassandra instance nodes	1 minute
cassandra037_pending_write	Suspended Write Tasks	Number of write tasks waiting in the queue	≥ 0	Counts	N/A	GeminiDB Cassandra instance nodes	1 minute
cassandra038_pending_read	Suspended Read Tasks	Number of read tasks waiting in the queue.	≥ 0	Counts	N/A	GeminiDB Cassandra instance nodes	1 minute
cassandra044_range_slice_latency	Scan Duration	Average time consumed by scan operations	≥ 0	ms	N/A	GeminiDB Cassandra instance nodes	1 minute
cassandra049_dropped_mutation	Dropped Writes	Average number of dropped writes	≥ 0	Counts	N/A	GeminiDB Cassandra instance nodes	1 minute
cassandra052_dropped_read	Dropped Reads	Average number of dropped reads	≥ 0	Counts	N/A	GeminiDB Cassandra instance nodes	1 minute
cassandra092_load_info	Data Volume on a Node	Data volume on a node	≥ 0	Byte	1024(IEC)	GeminiDB Cassandra instance nodes	1 minute

Metric ID	Name	Description	Value Range	Unit	Number System	Monitored Object	Monitoring Period (Raw Data)
cassandra093_write_count_latency	Accumulated Write Requests	Number of write requests initiated by a node	≥ 0	Counts	N/A	GeminiDB Cassandra instance nodes	1 minute
cassandra094_write_1min_rate	Average Write Rate in the Last Minute	Average write rate in the last minute	≥ 0	Counts/s	N/A	GeminiDB Cassandra instance nodes	1 minute
cassandra095_write_p75_latency	p75 Write Latency	p75 write latency	≥ 0	ms	N/A	GeminiDB Cassandra instance nodes	1 minute
cassandra096_write_p95_latency	p95 Write Latency	p95 write latency	≥ 0	ms	N/A	GeminiDB Cassandra instance nodes	1 minute
cassandra097_write_p99_latency	p99 Write Latency	p99 write latency	≥ 0	ms	N/A	GeminiDB Cassandra instance nodes	1 minute
cassandra098_read_count_latency	Accumulated Read Requests	Number of read requests initiated by a node	≥ 0	Counts	N/A	GeminiDB Cassandra instance nodes	1 minute
cassandra099_read_1min_rate	Average Read Rate in the Last Minute	Average read rate in the last minute	≥ 0	Counts/s	N/A	GeminiDB Cassandra instance nodes	1 minute
cassandra100_read_p75_latency	p75 Read Latency	p75 read latency	≥ 0	ms	N/A	GeminiDB Cassandra instance nodes	1 minute
cassandra101_read_p95_latency	p95 Read Latency	p95 read latency	≥ 0	ms	N/A	GeminiDB Cassandra instance nodes	1 minute

Metric ID	Name	Description	Value Range	Unit	Number System	Monitored Object	Monitoring Period (Raw Data)
cassandra102_read_p99_latency	p99 Read Latency	p99 read latency	≥ 0	ms	N/A	GeminiDB Cassandra instance nodes	1 minute
cassandra103_range_slice_count_latency	Accumulated Range Read Requests	Number of range read requests	≥ 0	Counts	N/A	GeminiDB Cassandra instance nodes	1 minute
cassandra104_range_slice_1min_rate	Average Range Read Rate in the Last Minute	Average range read rate in the last minute	≥ 0	Counts/s	N/A	GeminiDB Cassandra instance nodes	1 minute
cassandra105_range_slice_p75_latency	p75 Range Read Latency	p75 range read latency	≥ 0	ms	N/A	GeminiDB Cassandra instance nodes	1 minute
cassandra106_range_slice_p95_latency	p95 Range Read Latency	p95 range read latency	≥ 0	ms	N/A	GeminiDB Cassandra instance nodes	1 minute
cassandra107_range_slice_p99_latency	p99 Range Read Latency	p99 range read latency	≥ 0	ms	N/A	GeminiDB Cassandra instance nodes	1 minute
cassandra163_write_p999_latency	p999 Write Latency	p999 write latency	≥ 0	ms	N/A	GeminiDB Cassandra instance nodes	1 minute
cassandra164_read_p999_latency	p999 Read Latency	p999 read latency	≥ 0	ms	N/A	GeminiDB Cassandra instance nodes	1 minute

Metric ID	Name	Description	Value Range	Unit	Number System	Monitored Object	Monitoring Period (Raw Data)
cassandra165_large_partition_num	Big Keys	Number of big keys on the current node	≥ 0	Counts	N/A	GeminiDB Cassandra instance nodes	1 minute
cassandra166_write_max_latency	Maximum Write Latency	Maximum write latency	≥ 0	ms	N/A	GeminiDB Cassandra instance nodes	1 minute
cassandra167_read_max_latency	Maximum Read Latency	Maximum read latency	≥ 0	ms	N/A	GeminiDB Cassandra instance nodes	1 minute
cassandra168_imbalance_table_num	Tables with Uneven Data Distribution	Number of tables whose data is not evenly distributed	≥ 0	Counts	N/A	GeminiDB Cassandra instance nodes	1 minute
cassandra169_modify_request_size_mean	Average Write Request Size	Average write request size	≥ 0	Bytes	1024(IEC)	GeminiDB Cassandra instance nodes	1 minute
cassandra170_query_response_size_mean	Average Query Response Size	Average size of query requests	≥ 0 Bytes	Bytes	1024(IEC)	GeminiDB Cassandra instance nodes	1 minute
cassandra173_limit_diff_count_mean	Mean of limit Value and Returned Rows	Mean of limit difference	≥ 0	-	N/A	GeminiDB Cassandra instance nodes	1 minute
cassandra174_tombstone_query_rate	Tombstone Query Requests per Second	Rate of tombstone query requests	≥ 0	Counts/s	N/A	GeminiDB Cassandra instance nodes	1 minute

Metric ID	Name	Description	Value Range	Unit	Number System	Monitored Object	Monitoring Period (Raw Data)
cassandra175_max_sync_delay	maximum disaster backup instance sync delay	Maximum synchronization delay in case of disaster recovery	≥ 0	ms	N/A	GeminiDB Cassandra instance nodes	1 minute
cassandra175_single_delete_rate	Row Delete Requests per Second	Rate at which a single row is deleted	≥ 0	Counts/s	N/A	GeminiDB Cassandra instance nodes	1 minute
cassandra176_range_delete_rate	Range Delete Requests per Second	Range deletion rate	≥ 0	Counts/s	N/A	GeminiDB Cassandra instance nodes	1 minute
cassandra177_large_row_count	Large Rows per Second	Number of large rows	≥ 0	Counts/s	N/A	GeminiDB Cassandra instance nodes	1 minute

Dimensions

Key	Value
cassandra_cluster_id	Cluster ID of the GeminiDB Cassandra instance
cassandra_node_id	Node ID of the GeminiDB Cassandra instance

3.13.2 Configuring Alarm Rules

Setting alarm rules allows you to customize objects to be monitored and notification policies so that you can closely monitor your instances.

Alarm rules include the alarm rule name, instance, metric, threshold, monitoring interval, and whether to send notifications. This section describes how to set alarm rules.

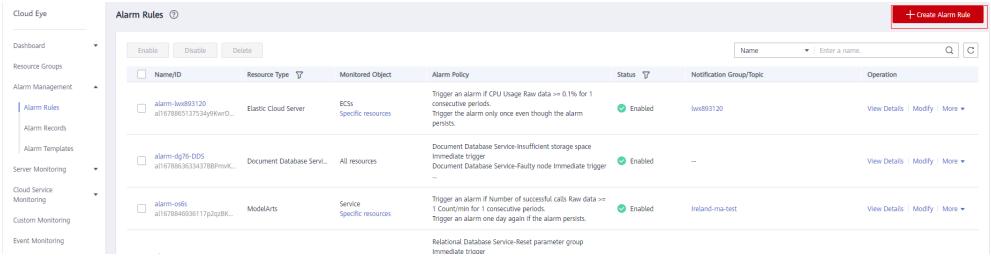
Procedure

Step 1 Log in to the **Cloud Eye console**.

Step 2 In the navigation pane on the left, choose **Alarm Management** > **Alarm Rules**.

Step 3 On the **Alarm Rules** page, click **Create Alarm Rule**.

Figure 3-82 Creating an alarm rule



Step 4 Set alarm parameters.

1. Configure basic alarm information.

Figure 3-83 Configuring basic information for an alarm rule

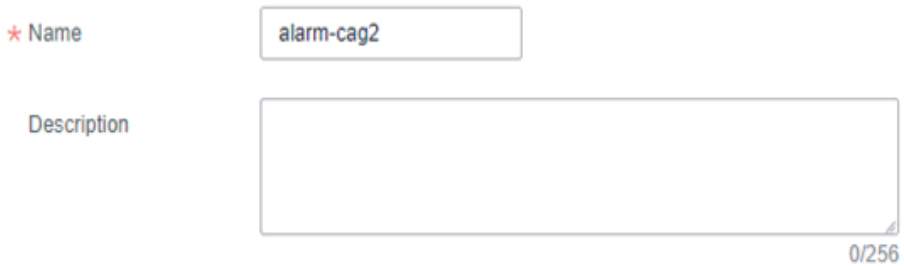


Table 3-33 Basic alarm rule information

Parameter	Description	Example Value
Name	Name of the rule. The system generates a random name and you can modify it.	alarm-cag2
Description	(Optional) Alarm rule description.	-

2. Select objects to be monitored and specify the monitoring scope.

Figure 3-84 Configuring objects to be monitored

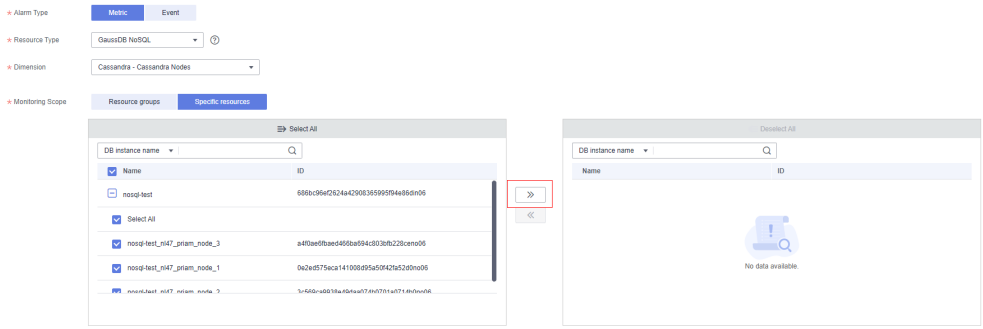


Table 3-34 Parameter description

Parameter	Description	Example Value
Alarm Type	Alarm type that the alarm rule is created for. The value can be **Metric** or **Event**.	Metric
Resource Type	Type of the resource the alarm rule is created for. Select **GeminiDB**.	-
Dimension	Metric dimension of the alarm rule. Select **Cassandra - Cassandra Nodes**.	-
Monitoring Scope	Monitoring scope the alarm rule applies to. **NOTE** – If you select **Resource groups** and any resource in the group meets the alarm policy, an alarm notification will be sent. – After you select **Specific resources**, select one or more resources and click » to add them to the box on the right.	Specified resources
Group	This parameter is mandatory when **Monitoring Scope** is set to **Resource groups**.	-

3. Configure an alarm policy.

Figure 3-85 Configuring an alarm policy

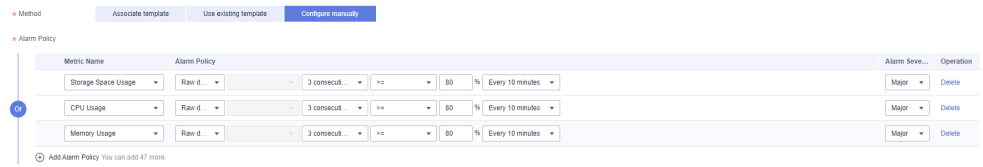


Table 3-35 Parameter description

Parameter	Description	Example Value
Method	Select **Associate template**, **Use existing template**, or **Configure manually**. **NOTE** If you set **Monitoring Scope** to **Specific resources**, you can set **Method** to **Use existing template**.	Configure manually
Template	Select the template to be used. This parameter is available only when you set **Method** to **Use existing template**.	-
Alarm Policy	Policy for triggering an alarm. You can configure the threshold, consecutive periods, alarm interval, and alarm severity based on service requirements. – **Metric Name**: metric that an alarm rule is created for The following metrics are recommended: **Storage Space Usage**: Storage usage of GeminiDB Cassandra instances. If the storage usage is greater than 80%, scale up the storage in a timely manner by referring to **Manually Scaling Up Storage Space**. **CPU Usage** and **Memory Usage**: Compute resource usage of each GeminiDB Cassandra instance node. If the CPU usage or memory usage is greater than 80%, you can **add nodes** or **increase node specifications** in a timely manner. For more metrics, see **Supported Metrics**. – **Alarm Severity**: specifies the severity of the alarm. Valid values are **Critical**, **Major**, **Minor**, and **Informational**. **NOTE** A maximum of 50 alarm policies can be added to an alarm rule. If any one of these alarm policies is met, an alarm is triggered.	Take the CPU usage as an example. The alarm policy configured in **Figure 3-85** indicates that a major alarm notification will be sent to users every 10 minutes if the original CPU usage reaches 80% or above for three consecutive periods.

4. Configure alarm notification information.

Figure 3-86 Configuring alarm notification information

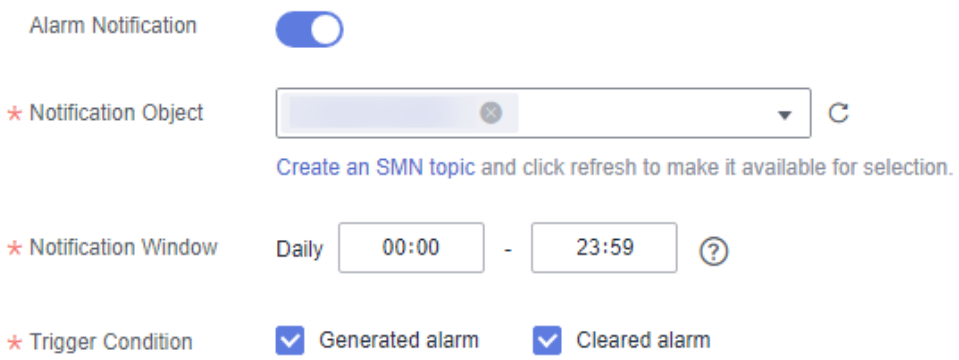


Table 3-36 Parameter description

Parameter	Description	Example Value
Alarm Notification	Whether to notify users when alarms are triggered. Notifications can be sent by email, text message, or HTTP/HTTPS message. Enabling alarm notification is recommended. When the metric data reaches the threshold set in the alarm rule, Cloud Eye immediately notifies you through SMN that an exception has occurred.	Enabled **Alarm Notification**.
Notification Object	Specifies the object that receives alarm notifications. You can select the account contact or a topic. – Account contact is the mobile phone number and email address provided for registration. – **Topic** is used to publish messages and subscribe to notifications. If the required topic is unavailable, create one first and add subscriptions to it.	-

Parameter	Description	Example Value
Notification Window	Cloud Eye sends notifications only within the notification window specified in the alarm rule. For example, if **Notification Window** is set to **00:00-8:00**, Cloud Eye sends notifications only within 00:00-08:00.	-
Trigger Condition	Condition for triggering an alarm notification. You can select **Generated alarm** (when an alarm is generated), **Cleared alarm** (when an alarm is cleared), or both.	-

5. Configure advanced settings.

Figure 3-87 Advanced settings

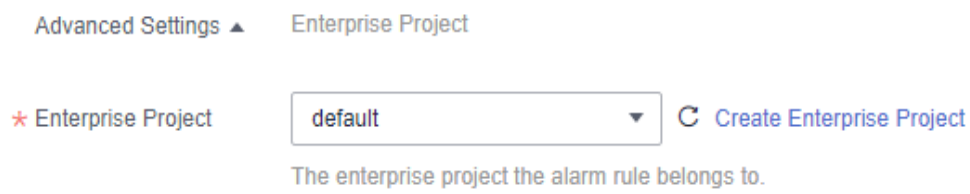


Table 3-37 Parameter description

Parameter	Description	Example Value
Enterprise Project	Enterprise project that the alarm rule belongs to. Only users with the enterprise project permissions can view and manage the alarm rule.	default

Step 5 After the configuration is complete, click **Create**.

When the metric data reaches the threshold set in the alarm rule, Cloud Eye immediately notifies you through SMN that an exception has occurred.

NOTE

For more information about alarm rules, see **Cloud Eye User Guide**.

----End

3.13.3 Viewing Metrics

Cloud Eye monitors the status of GeminiDB Cassandra instances. You can check GeminiDB Cassandra API metrics on the console.

Monitored data requires a period of time for transmission and display. The status of the monitored object displayed on the Cloud Eye page is the status obtained 5 to 10 minutes before. You can view the monitored data of a newly created DB instance 5 to 10 minutes later.

Usage Notes

- The DB instance is running properly.

  Cloud Eye does not display the metrics of a faulty or deleted DB instance. You can view the monitoring information only after the instance is restarted or recovered.
- The DB instance has been properly running for at least 10 minutes.

  The monitoring data and graphics are available for a new DB instance after the instance runs for at least 10 minutes.

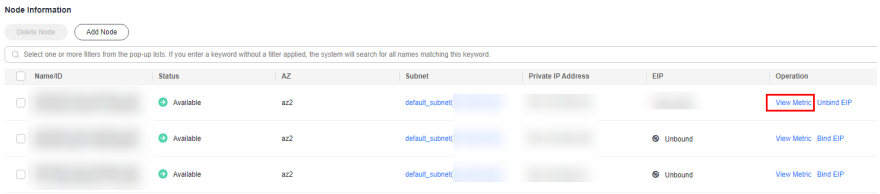
Procedure

Step 1 Log in to the Huawei Cloud console.

Step 2 On the **Instance** page, click the instance whose metrics you want to view and click its name.

Step 3 In the **Node Information** area on the **Basic Information** page, click **View Metric** in the **Operation** column.

Figure 3-88 Querying monitoring metrics



Step 4 In the monitoring area, select a time range to view monitoring data.

You can view the monitoring data in the last 1, 3, or 12 hours.

To view the monitoring curve in a longer time range, click to enlarge the graph.

----End

3.14 Enterprise Project

3.14.1 Overview

An enterprise project facilitates project-level management and grouping of cloud resources and users. The default project is **default**.

You can also customize enterprise projects to meet your service requirements. For details, see ***Enterprise Management User Guide***.

3.14.2 Quota Management

GeminiDB Cassandra API provides a quota function that allows you to manage resources by controlling the number of resources in each enterprise project to ensure that resources can be used and managed properly.

This section describes how to query used resources in each enterprise project and its resource quotas.

This function is now in OBT. To use it, choose **Service Tickets > Create Service Ticket** in the upper right corner of the console and contact the customer service.

Viewing Resource Quotas in Each Enterprise Project

Step 1 Log in to the Huawei Cloud console.

Step 2 In the navigation pane on the left, choose **My Quotas** to view quota details of the current enterprise project.

Figure 3-89 Quota management

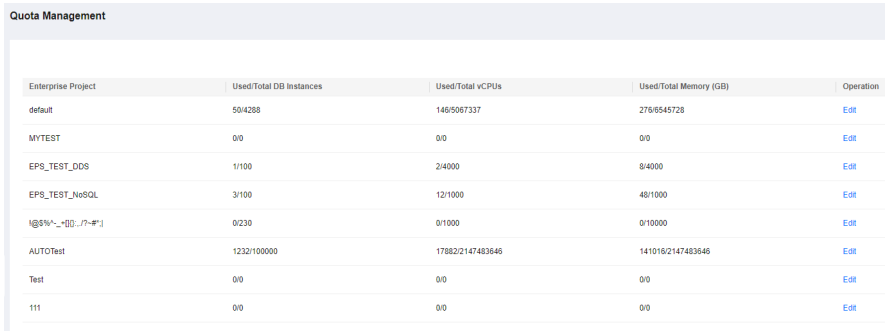


Table 3-38 Parameter description

Parameter	Description
Enterprise Project	Enterprise project that an instance belongs to.
Used/Total DB Instances	Number of used instances in the current enterprise project
Used/Total vCPUs	vCPUs of all instances in the current enterprise project
Used/Total Memory (GB)	Memory of all instances in the current enterprise project

NOTE

If there are no resources in an enterprise project, the default quota is 0. Before creating an instance, you need to set quotas first by referring to **Modifying Resource Quotas of an Enterprise Project**.

----End

Modifying Resource Quotas of an Enterprise Project

Step 1 Log in to the Huawei Cloud console.

Step 2 In the navigation pane on the left, choose **My Quotas**. In the quota list, select the enterprise project you want to set quotas for and click **Modify** in the **Operation** column.

Figure 3-90 Managing quotas

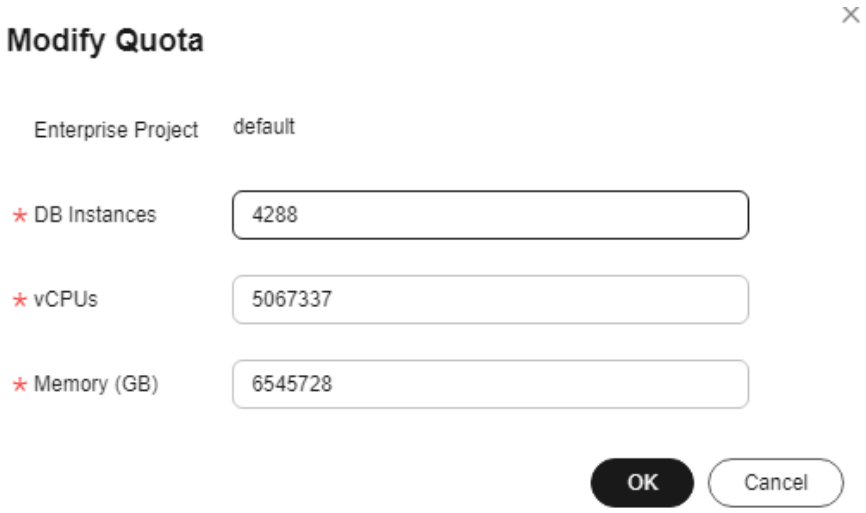


Table 3-39 Quota management

Parameter	Value Range
DB Instances	0–5,000
vCPUs	0–8,000,000
Memory (GB)	0–16,000,000

----End

3.15 Billing Management

3.15.1 Renewing Instances

This section describes how to renew your yearly/monthly GeminiDB Cassandra instances.

Precautions

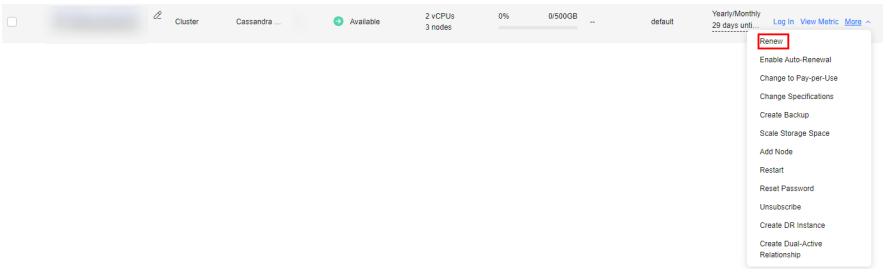
Pay-per-use instances do not support this function.

Renewing a Single Yearly/Monthly Instance

Step 1 Log in to the Huawei Cloud console.

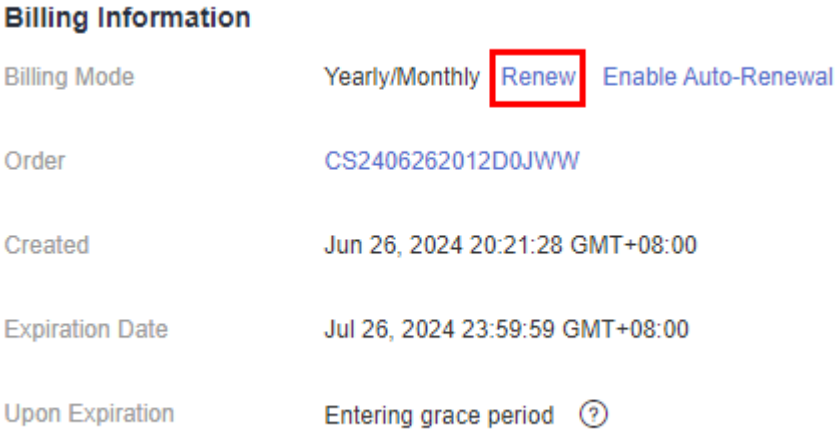
Step 2 On the **Instances** page, locate the target instance and click **Renew** in the **Operation** column.

Figure 3-91 Renewal button



Alternatively, click the instance name to go to the **Basic Information** page. In the **Billing Information** area, click **Renew** next to the **Billing Mode** field.

Figure 3-92 Renewal button



Step 3 On the displayed page, renew the instance.

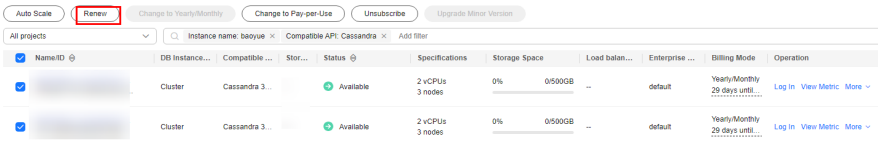
----End

Renewing Instances

Step 1 Log in to the Huawei Cloud console.

Step 2 On the **Instances** page, select the instances that you want to renew and click **Renew** above the instance list.

Figure 3-93 Renewing instances



Step 3 In the displayed dialog box, click **Yes**.

----End

3.15.2 Changing Pay-per-Use to Yearly/Monthly

This section describes how to change the billing mode of a GeminiDB Cassandra instance from pay-per-use to yearly/monthly. If you want to use a pay-per-use instance for a long time, change its billing mode to yearly/monthly to reduce costs.

Precautions

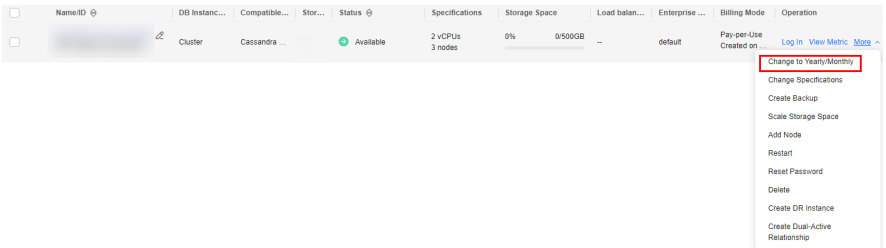
Only when the status of a pay-per-use instance is **Available**, its billing mode can be changed to yearly/monthly.

Changing the Billing Mode of a Single Pay-per-Use Instance to Yearly/Monthly

Step 1 Log in to the Huawei Cloud console.

Step 2 On the **Instances** page, locate the target instance and click **Change to Yearly/Monthly**.

Figure 3-94 Change to Yearly/Monthly



Step 3 On the displayed page, select the renewal duration in month. The minimum duration is one month.

If you do not need to modify your settings, click **Pay Now**.

Step 4 Select a payment method and click **Pay**.

Step 5 View the results on the **Instances** page.

In the upper right corner of the instance list, click to refresh the list. The instance status will become **Available** after the change is successful. The billing mode becomes to **Yearly/Monthly**.

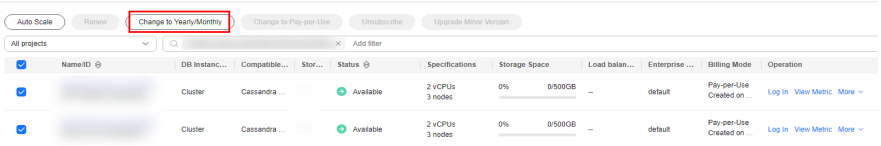
----End

Batch Changing Pay-per-Use to Yearly/Monthly

Step 1 Log in to the Huawei Cloud console.

Step 2 On the **Instances** page, select the instances whose billing mode you want to change and click **Change to Yearly/Monthly** above the instance list. In displayed dialog box, click **Yes**.

Figure 3-95 Change to Yearly/Monthly



Step 3 On the displayed page, select the renewal duration in month. The minimum duration is one month.

If you do not need to modify your settings, click **Pay Now**.

Step 4 Select a payment method and click **Pay**.

Step 5 View the results on the **Instances** page.

In the upper right corner of the instance list, click to refresh the list. The instance status will become **Available** after the change is successful. The billing mode changes to **Yearly/Monthly**.

----End

3.15.3 Changing Yearly/Monthly to Pay-per-Use

You can change a yearly/monthly GeminiDB Cassandra instance to pay-per-use if you intend to discontinue long-term use once it expires.

Precautions

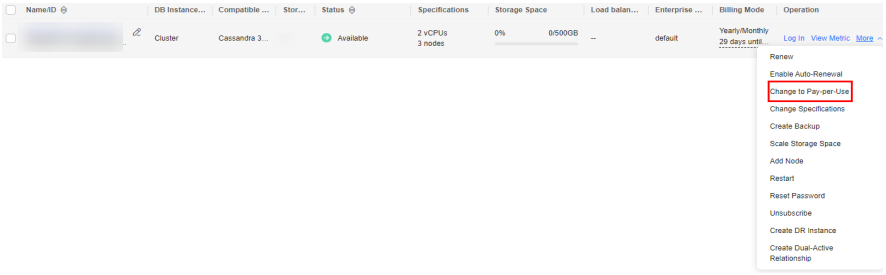
- Only a yearly/monthly instance in the **Available** state can be changed to pay-per-use.
- After the billing mode is changed, auto-renewal will be disabled.

Changing a Single Yearly/Monthly Instance to Pay-per-Use

Step 1 Log in to the Huawei Cloud console.

Step 2 On the **Instances** page, locate the instance whose billing mode you want to change and click **Change to Pay-per-Use** in the **Operation** column.

Figure 3-96 Change to Pay-per-Use



Step 3 On the displayed page, confirm the instance information and click **Change to Pay-per-Use** to submit the change. The billing mode will change to pay-per-use after the instance expires.

Step 4 After you submit the change, a message is displayed in the **Billing Mode** column of the target DB instance, indicating that the billing mode will be changed to pay-per-use after the DB instance expires.

Step 5 To cancel the change, choose **Billing** > **Renewal** to enter the Billing Center. On the **Renewals** page, locate the target DB instance and click **More** > **Cancel Change to Pay-per-Use**.

Step 6 In the displayed dialog box, click **Yes**.

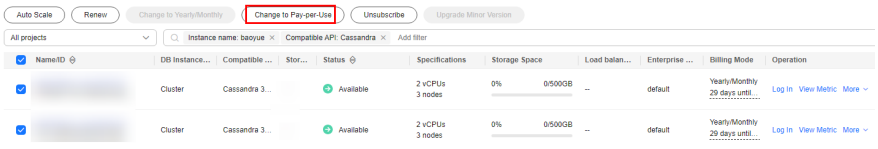
----End

Batch Changing Yearly/Monthly to Pay-per-Use

Step 1 Log in to the Huawei Cloud console.

Step 2 On the **Instances** page, select the instances whose billing mode you want to change and click **Change to Pay-per-Use** above the instance list.

Figure 3-97 Batch changing yearly/monthly to pay-per-use



Step 3 In the displayed dialog box, click **Yes**.

Step 4 On the displayed page, confirm the instance information and click **Change to Pay-per-Use**. The billing mode will change to pay-per-use after the instance expires.

NOTE

After the billing mode is changed, auto-renewal will be disabled.

Step 5 After you submit the change, check whether a message is displayed in the **Billing Mode** column, indicating that the billing mode will be changed to pay-per-use after the instance expires.

Step 6 To cancel the change, choose **Billing** > **Renewal** to enter the Billing Center. On the **Renewals** page, locate the instance and click **More** > **Cancel Change to Pay-per-Use**.

Step 7 In the displayed dialog box, click **Yes**.

----End

3.15.4 Unsubscribing from a Yearly/Monthly Instance

If you do not need a yearly/monthly instance any longer, unsubscribe from it.

Precautions

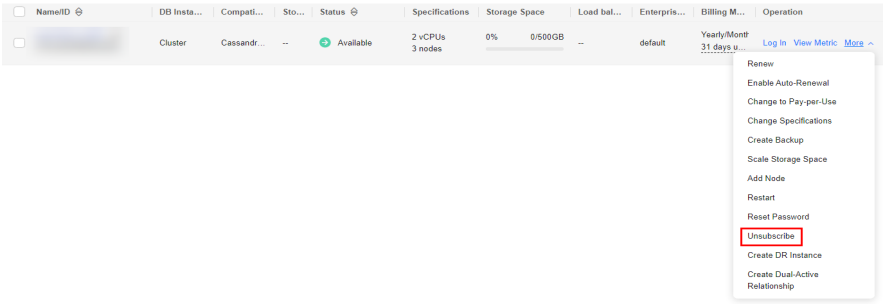
- Unsubscribed operations cannot be undone. Exercise caution when performing this operation. To retain data, create a manual backup before unsubscription. For details, see **Creating a Manual Backup**.
- After an unsubscription request is submitted, resources and data will be deleted and cannot be retrieved. To retain data, back it up before submitting the unsubscription request.

Unsubscribing from a Single Yearly/Monthly Instance

Step 1 Log in to the Huawei Cloud console.

Step 2 On the **Instances** page, locate the instance you want to unsubscribe from and choose **More** > **Unsubscribe** in the **Operation** column.

Figure 3-98 Unsubscribing from a yearly/monthly instance



Step 3 In the displayed dialog box, click **Yes**.

Step 4 On the displayed page, confirm the unsubscription and select a reason. Then, click **Confirm**.

For details, see **Unsubscription Rules**.

Step 5 In the displayed dialog box, click **Yes**.

NOTE

1. After an unsubscription request is submitted, resources and data will be deleted and cannot be retrieved.
2. To retain data, back it up before submitting the unsubscription request.

Step 6 View the unsubscription result. After the instance order is successfully unsubscribed, the instance is no longer displayed in the instance list on the **Instances** page.

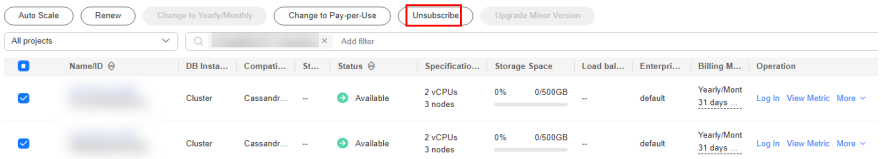
----End

Batch Unsubscribing from Yearly/Monthly Instances

Step 1 Log in to the Huawei Cloud console.

Step 2 Choose **Instances** in the navigation pane on the left, select the instances you want to unsubscribe from and click **Unsubscribe** above the instance list.

Figure 3-99 Batch unsubscribing from yearly/monthly instances



Step 3 In the displayed dialog box, click **Yes**.

Step 4 On the displayed page, confirm the unsubscription and select a reason. Then, click **Confirm**.

For details, see **Unsubscription Rules**.

Step 5 In the displayed dialog box, click **Yes**.

NOTE

1. After an unsubscription request is submitted, resources and data will be deleted and cannot be retrieved.
2. To retain data, back it up before submitting the unsubscription request.

Step 6 View the unsubscription result. After the instances are successfully unsubscribed from, they are no longer displayed in the instance list any longer on the **Instances** page.

----End

3.16 Managing GeminiDB Cassandra Instance Tags

Tag Management Service (TMS) enables you to manage resources using tags on the management console. TMS works with other cloud services to manage tags. TMS manages tags globally while other cloud services manage their own tags.

Adding tags to GeminiDB Cassandra instances helps you better identify and manage them. An instance can be tagged when or after it is created.

After an instance is tagged, you can search for the tag key or value to quickly query the instance details.

Usage Notes

- You are advised to set predefined tags on the TMS console.
- A tag consists of a key and value. You can add only one value for each key. For details about the naming rules of tag keys and tag values, see **Table 3-40**.
- A maximum of 20 tags can be added for each instance.
- The tag name must comply with the naming rules described in **Table 3-40**.

Table 3-40 Naming rules

Parameter	Requirement	Example Value
Tag key	• Cannot be left blank. • Must be unique for each instance. • Can contain a maximum of 128 characters. • Can only consist of digits, letters, underscores (_), and hyphens (-).	Organization
Tag value	• Can be left blank. • Can contain a maximum of 255 characters. • Can only consist of digits, letters, underscores (_), periods (.), and hyphens (-).	nosql_01

Adding a Tag

Step 1 Log in to the Huawei Cloud console.

Step 2 On the **Instances** page, click the target instance. The **Basic Information** page is displayed.

Step 3 In the navigation pane on the left, click **Tags**.

Step 4 On the **Tags** page, click **Add Tag**. In the displayed dialog box, enter a tag key and value, and click **OK**.

Step 5 View and manage tags on the **Tags** page.

----End

Editing a Tag

Step 1 Log in to the Huawei Cloud console.

Step 2 On the **Instances** page, click the target instance. The **Basic Information** page is displayed.

Step 3 In the navigation pane on the left, click **Tags**.

Step 4 On the **Tags** page, locate the tag that you want to edit and click **Edit** in the **Operation** column. In the displayed dialog box, change the tag value and click **OK**.

Only the tag value can be edited.

Step 5 View and manage tags on the **Tags** page.

----End

Deleting a Tag

Step 1 Log in to the Huawei Cloud console.

Step 2 On the **Instances** page, click the target instance. The **Basic Information** page is displayed.

Step 3 In the navigation pane on the left, click **Tags**.

Step 4 On the **Tags** page, locate the tag that you want to delete and click **Delete** in the **Operation** column. In the displayed dialog box, click **Yes**.

Step 5 Check whether the deleted tag is displayed on the **Tags** page.

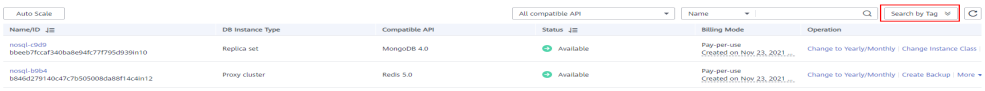
----End

Searching by tag

Step 1 Log in to the Huawei Cloud console.

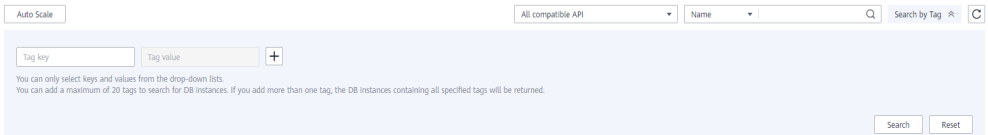
Step 2 On the **Instances** page, click **Search by Tag** in the upper right corner of the instance list.

Figure 3-100 Search by Tag



Step 3 Enter a tag key or value and click **Search** to query the instance associated with the tag.

Figure 3-101 Searching by tag key



----End

4 FAQs

4.1 Product Consulting

4.1.1 What Should I Pay Attention to When Using GeminiDB Cassandra API?

1. DB instance operating systems (OSs) are invisible to you. Your applications can access a database only through an IP address and a port.
2. The backup files stored in OBS and the system containers are invisible to you. They are visible only in the GeminiDB Cassandra API management system.
3. Precautions after purchasing DB instances:

   After purchasing DB instances, you do not need to perform basic database O&M operations, such as applying HA and security patches, but you should still note:

   a. The CPU, input/output operations per second (IOPS), and space are sufficient for the DB instances.
   b. The DB instance has performance problems and whether optimization is required.

4.1.2 What Is GeminiDB Cassandra Instance Availability?

The formula for calculating the instance availability is as follows:

DB instance availability = (1 – Failure duration/Total service duration) × 100%

The failure duration refers to the total duration of faults that occur during the running of an instance after you buy the instance. The total service duration refers to the total running time of the instance.

4.2 Billing

4.2.1 What Are the Differences Between Yearly/Monthly and Pay-per-Use Billing Modes?

Yearly/Monthly is a prepaid billing mode in which resources are billed based on the service duration. This cost-effective mode is ideal when the duration of resource usage is predictable. It is recommended for long-term users.

Pay-per-use is a postpaid mode. You are only billed for how long you have actually used your instance. This mode can be a good option when future requirements are unpredictable. Pay-per-use instances are priced by the hour, but if an instance is used for less than one hour, you will be billed based on the actual duration.

4.2.2 Can I Switch Between Yearly/Monthly and Pay-per-Use Payments?

You can change the billing mode from yearly/monthly to pay-per-use or vice versa.

- If you want to change the billing mode from yearly/monthly to pay-per-use, see **Changing Yearly/Monthly to Pay-per-Use**.
- If you want to change the billing mode from pay-per-use to yearly/monthly, see **Changing Pay-per-Use to Yearly/Monthly**.

4.3 Database Usage

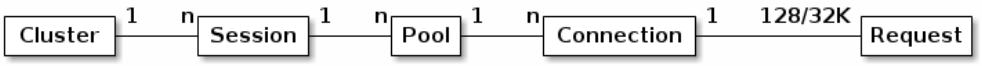
4.3.1 Why Does the Overall Instance Performance Deteriorate When QPS Increases After the Batch Size Is Decreased?

Symptom

The original **batch_size** was 100, and the size of a single row was about 400 bytes. **batch_size** was then changed to 10 because an alarm was triggered when the batch size reached 5 KB. To ensure the overall write performance, QPS was 10 times of the original QPS. However, the overall performance deteriorated after the changes.

Possible Cause

The number of concurrent clients is restricted by the Driver configuration parameters, including the number of hosts, number of sessions, **ConnectionsPerHost**, and **MaxRequestsPerConnection**.



For example, a user starts a cluster, creates a session for the cluster, and has three hosts. If **ConnectionsPerHost** is set to **2** and **MaxRequestsPerConnection** uses the default value **128**, the maximum number of concurrent requests of the session is 768, and the maximum number of requests of a single node is 256.

For details about the parameters, see the **official document**.

Solution

View **monitoring metrics** to observe the CPU usage, read/write pending, and read/write latency of a single node.

- If the load of a single node reaches the upper limit, you need to add nodes. For details, see **Manually Adding Instance Nodes**.
- If the load of a single node is low, you need to adjust the configuration of Driver.
  a. Increase the value of **ConnectionsPerHost**. Ensure that the total number of connections to the cluster does not exceed the configured alarm threshold.
  b. Increase the value of **MaxRequestsPerConnection**. Ensure that the value does not exceed the load capability of a single node. Observe the CPU usage, read/write latency, and read/write pending.

4.3.2 What Can I Do if Error "field larger than field limit (131072)" Is Reported During Data Import?

Symptom

When you import data, the size of a single column exceeds 128 KB. As a result, the Python CSV single-column restriction is triggered.

Error message:

```
field larger than field limit (131072)
```

Possible Cause

When Python CSV reads a file, **csv.field_size_limit** limits the size of a single column.

Solution

Step 1 Run the following commands in the **cqlsh** directory to find the **cqlshrc** file:

touch cqlshrc

rm -rf ~/.cassandra/cqlshrc*

Step 2 Add the following information in the **cqlshrc** file and save the file:

```
[csv]
field_size_limit = 9223372036854775807
```

Step 3 Add the following parameters when connecting to an instance using cqlsh:

–cqlshrc=cqlshrc

Command example:

cqlsh 127.0.0.1 8635 -u rwuser -p password --cqlshrc=cqlshrc

----End

4.3.3 What Should I Pay Attention to When Creating a GeminiDB Cassandra Table?

When you create tables in a GeminiDB Cassandra database, pre-allocate memory to guarantee database performance. GeminiDB Cassandra has a limit on the number of tables.

Precautions

- Half of node memory is allocated to the storage engine.
- An odd number of clusters can tolerate N/2-1 faulty nodes, and an even number of clusters can tolerate N/2 faulty nodes.
- GeminiDB Cassandra API utilizes a table-level hash ring, with the **tokens** parameter indicating the number of data shards for a table. This parameter differs from the **num_tokens** used in open-source Cassandra.

Calculating the Number of Tables

The memory required for creating tables depends on the instance specifications. Assume that an instance has 4 vCPUs and 16 GB memory and the size of a single table is 768 MB.

Maximum number of tables that can be created = Total available memory of the cluster / Memory required by a single table

- Cluster with an odd number of nodes

  Available cluster memory = Node memory/2 x (N/2 + 1)
- Cluster with an even number of nodes

  Available cluster memory = Node memory/2 x (N/2)

For example:

- Available memory of an instance with 3 nodes, 4 vCPUs, and 16 GB memory = 16/2 x (3/2 + 1) = 16 GB

  Maximum number of created tables = 16 x 1024 MB/768 MB = 21
- Available memory of an instance with 4 nodes, 4 vCPUs, and 16 GB memory = 16/2 x (4/2) = 16 GB

  Maximum number of created tables = 16 x 1024 MB/768 MB = 21
- Available memory of an instance with 5 nodes, 4 vCPUs, and 16 GB memory = 16/2 x (5/2 + 1) = 24 GB

  Maximum number of created tables = 24 x 1024 MB/768 MB = 32

For details about the mapping between the number of nodes (4 vCPUs, 16 GB) and the number of tables, see **Table 4-1**.

Table 4-1 Upper limit on the number of tables

Instance Class	Number of Nodes	Number of Tables
4 vCPUs | 16 GB	3	21
	4	21

Instance Class	Number of Nodes	Number of Tables
	5	32
	6	32
	7	42
	8	42
	9	53
	10	53
	11	64
	12	64

NOTE

- A single table occupies 768 MB memory, and the default number of table tokens is 12. If tokens are separately set, calculate the number of tables using the following formula: (768/12) x Number of tokens.
- The preceding formula is designed for common tables. If stream table is enabled, one stream table consumes resources 2.5 times more than common tables.

For details about the mapping between the number of nodes (8 vCPUs, 32 GB) and the number of tables, see **Table 4-2**.

Table 4-2 Upper limit on the number of tables

Instance Class	Number of Nodes	Number of Tables
8 vCPUs | 32 GB	3	22
	4	22
	5	34
	6	34
	7	45
	8	45
	9	56
	10	56
	11	68
	12	68

NOTE

- A single table occupies 1440 MB memory, and the default number of table tokens is 12. If tokens are set separately, calculate the number of tables using the following formula: (1440/12) x Number of tokens.
- The preceding formula is designed for common tables. If stream table is enabled, one stream table consumes resources 2.5 times more than common tables.

For details about the mapping between the number of nodes (16 vCPUs, 64 GB) and the number of tables, see **Table 4-3**.

Table 4-3 Upper limit on the number of tables

Instance Class	Number of Nodes	Number of Tables
16 vCPUs | 64 GB	3	45
	4	45
	5	68
	6	68
	7	91
	8	91
	9	113
	10	113
	11	136
	12	136

NOTE

- A single table occupies 1440 MB memory, and the default number of table tokens is 12. If tokens are set separately, calculate the number of tables using the following formula: (1440/12) x Number of tokens.
- The preceding formula is designed for common tables. If stream table is enabled, one stream table consumes resources 2.5 times more than common tables.

For details about the mapping between the number of nodes (32 vCPUs, 128 GB) and the number of tables, see **Table 4-4**.

Table 4-4 Mapping between the number of nodes (32U128GB) and the number of tables

Instance Class	Number of Nodes	Number of Tables
32 vCPUs | 128 GB	3	68
	4	68
	5	102

Instance Class	Number of Nodes	Number of Tables
	6	102
	7	136
	8	136
	9	170
	10	170
	11	204
	12	204

NOTE

- A single table occupies 1920 MB memory, and the default number of table tokens is 12. If tokens are separately set, calculate the number of tables using the following formula: (1920/12) x Number of tokens
- The preceding formula is designed for common tables. If stream table is enabled, one stream table consumes resources 2.5 times more than common tables.

Parameters for Creating a Table

1. **Z00_THROUGHPUT** (throughput parameter) is related to the upper limit of table write performance. The default value is **big**, indicating the upper limit of standard write performance.
   - Low throughput
     ```
     CREATE TABLE test1 (k int,p int,s int static,v int,PRIMARY KEY (k, p)) WITH Z00_THROUGHPUT = 'small';
     ```
   - Medium throughput
     ```
     CREATE TABLE test2 (k int,p int,s int static,v int,PRIMARY KEY (k, p)) WITH Z00_THROUGHPUT = 'medium';
     ```
   - High throughput
     ```
     CREATE TABLE test3 (k int,p int,s int static,v int,PRIMARY KEY (k, p)) WITH Z00_THROUGHPUT = 'big';
     ```
2. Number of table tokens: indicates the number of table tokens when a table is created. The number of tokens must be greater than 1.
   ```
   CREATE TABLE test4 (k int,p int,s int static,v int,PRIMARY KEY (k, p)) WITH Z01_TABLE_TOKENS = 24;
   ```
3. Table parameters: Z00_BUFFER_SIZE and Z00_BUFFER_NUMBER (not recommended).

   When creating a table, you can specify the number of memtables in the storage layer and the size of each memtable.
   - **Z00_BUFFER_SIZE** is of the map type and specifies the CF name and value. The value ranges from 2 to 32.
     ```
     CREATE TABLE test6 (k int,p int,s int static,v int,PRIMARY KEY (k, p)) WITH Z00_BUFFER_SIZE = {'default': 16};
     ```
   - **Z00_BUFFER_NUMBER** is of the map type and specifies the CF name and value. The value ranges from 2 to 8.


```
CREATE TABLE test5 (k int,p int,s int static,v int,PRIMARY KEY (k, p)) WITH
Z00_BUFFER_NUMBER = {'default': 3};
```

NOTE

If you need to adjust the table specifications after the table is created, for example, when the maximum number of the tables is reached, you can reduce the table specifications to create more tables by adjusting the following parameters.

- If you set the throughput of all created tables to medium, the number of tables can be doubled
  ```
  ALTER TABLE keyspace_name.table_name WITH Z00_THROUGHPUT = 'medium';
  ```
- If you set the throughput of all created tables to small, the number of tables can be tripled.
  ```
  ALTER TABLE keyspace_name.table_name WITH Z00_THROUGHPUT = 'small';
  ```

4.3.4 How Do I Detect and Resolve BigKey and HotKey Issues?

The Cassandra database is a highly scalable, high-performance, and distributed database. It is suitable for big data scenarios and can be used to manage a large amount of structured data. With continuous growth of service volume and data traffic, some service design defects are gradually exposed, which reduces the stability and availability of the cluster. For example, the primary key design is improper, or a single partition contains a large amount of data. As a result, the partition key is too large, the node load is unbalanced, and the cluster stability deteriorates. This type of problem is called BigKey. When the workload of access to a key exceeds the maximum workload that a server can handle, we can call it a HotKey. Generally, a BigKey is an indirect cause of a HotKey issue.

GeminiDB Cassandra is a cloud-native distributed NoSQL database with a decoupled compute and storage architecture and compatible with the Cassandra ecosystem. To solve the preceding issues, GeminiDB Cassandra provides real-time detection of BigKey and HotKey issues to help you design schemas and avoid service stability risks.

BigKey Issue

- Possible causes

  The main cause of the BigKey issue is that the primary key design is improper. As a result, a single partition contains too many records or data. Once a partition becomes extremely large, the access to this partition increases the load of the server where the partition is located, and even causes the out of memory (OOM) issue.

- Troubleshooting

  You can use either of the following methods to rectify BigKey issues:

  - Add caches and optimize the table structure.
  - Add a new partition key for hashing data. Split data to avoid too much data in a single partition.

- Check method

  You can specify a threshold based on your service requirement. If any threshold is exceeded, a BigKey is generated.

  a. The number of rows of a single partition key cannot exceed 100,000.

  b. The size of a single partition cannot exceed 100 MB.

GeminiDB Cassandra supports BigKey detection and alarms. On the Cloud Eye console, you can configure BigKey alarms for instances. For details, see **Configuring Alarm Rules**.

When a BigKey event occurs, the system sends a warning notification immediately. You can **view the event data** on the Cloud Eye page and handle the event in a timely manner to prevent service fluctuation.

Figure 4-1 Viewing events of big key alarms

```
{
  "partition_size": "15877794",
  "timestamp": "2024-06-14 02:20:55,197",
  "partition_num": "125022",
  "keyspace_name": "app2019060514",
  "node_id": "74caa0d7c7b041dd8c71b59926539e3ano06",
  "table_name": "reference",
  "table_id": "25079220-5d76-11ec-bd28-15aff8bd9cbf",
  "partition_key": "{fileid=67607ba84bbed6ae6aefa59a11a734c1761}"
}
```

The alarm is described as follows:

```
[
 {
  "partition_size": "1008293497",            //Total size of oversized partition keys
  "timestamp": "2021-09-08 07:08:18,240",    //Time when a BigKey is generated
  "partition_num": "676826",         //Total number of rows for oversized partition keys
  "keyspace_name": "ssss",     //keyspace name
  "node_id": "ae342330ded14605b6304e80e6a6efeeno06",     //node ID
  "table_name": "zzzz",       //Table name
  "table_id": "024a1070-0064-11eb-bdf3-d3fe5956183b",     //Table ID
  "partition_key": "{vin=TESTW3YWZD2021003}"      //Partition key
 }
]
```

- Common cases and solutions

  Case 1: The data volume of a cluster is too large. As a result, the cluster has large partition keys (more than 2,000 partition keys are checked), and the maximum size of a partition key has reached 38 GB. When services frequently access these large partition keys, the node load remains high, affecting the service request success rate.

  The table structure is designed as follows.

```
CREATE TABLE movie (
    movieid text,
    appid int,
    uid bigint,
    accessstring text,
    moviename text,
    access_time timestamp,
    PRIMARY KEY (movieid, appid, uid, accessstring, moviename)
)
```


Table design analysis:

The **movie** table stores information about short videos. The partition key is movieid, and stores user information (uid). If movieid is a popular short video and tens of millions or even hundreds of millions of users like this short video, the size of the partition where the short video is located is large (38 GB).

Solution:

To solve the problem, perform the following steps:

a. Optimize the table structure.

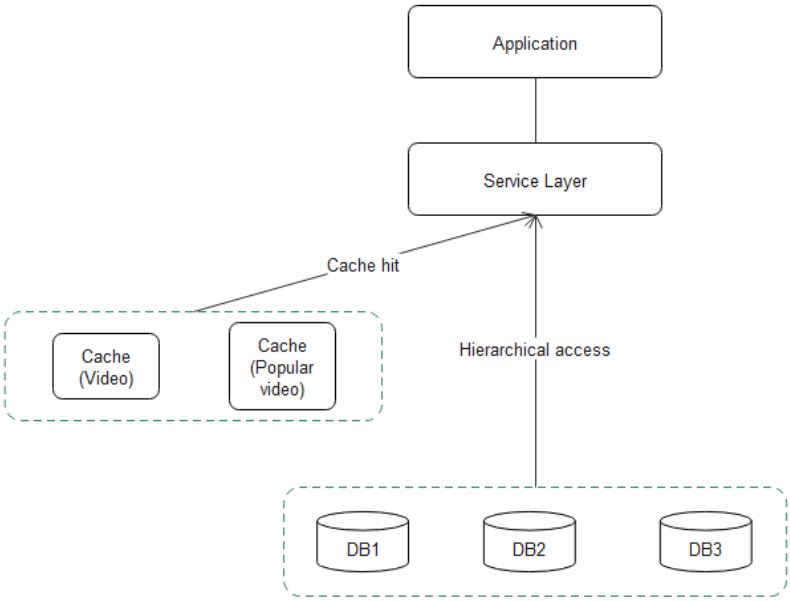
   Create a table to store the short video information. Only public short video information is retained, and user information is not included. This ensures that the table does not generate large partition keys. Write the short video information to the table.

```
CREATE TABLE hotmovieaccess (
    movieid text,
    appid int,
    accessstring text,
    access_time timestamp,
    PRIMARY KEY (movieid, appid)
)
```

b. Add caches.

   A service application first reads popular file information from the cache. If no information is found, the service application queries the database to reduce the number of database query times.

   The overall optimization logic is as follows:



   i. The service applications query the cache first. If the data to be queried already exists in the cache, the results are directly returned.

   ii. If the data is not in the cache, the popular video cache, the **hot** table, and the **hotmovieaccess** table will be accessed in sequence.

iii. If the **hotmovieaccess** table contains the results, the results are directly returned. If the **hotmovieaccess** table does not contain any record, the **movie** table is queried.

iv. Cache the query results.

Case 2: The **movie_meta** table is created by month, and each table stores only the data of the current month. The initial design can reduce or avoid large partition keys. Due to frequent service writes, a large number of popular video records are stored, generating large partitions.

```
CREATE TABLE movie_meta202110 (
    path text,
    moviename text,
    movieid text,
    create_time timestamp,
    modify_mtime timestamp,
    PRIMARY KEY (path, moviename)
)
```

Solution:

A random number (0 to 999) is added to the new partition key. The information stored in the original partition is randomly and discretely stored to 1,000 partitions. After the new partition key is used, no new partition key whose size exceeds 100 MB is formed. The old partition key data whose size exceeds 100 MB expires as time goes by.

HotKey Problem

- Hazards of HotKey:

  In daily life, when the hot news is clicked, viewed, and commented for tens of thousands of times in an application, large number of requests will be generated. In this case, the same key is frequently accessed within a short period of time. As a result, the CPU usage and load of the node where the key is located suddenly increase, affecting other requests on the node and decreasing the service success rate. Such scenarios include promotion of popular products and Internet celebrity live streaming. In these read-intensive scenarios, HotKey issues will be generated.

  The HotKey issue has the following impacts:

  a. The traffic is centralized and reaches the upper limit of the physical NICs.

  b. Too many requests may cause the cache service to break down.

  c. The database breaks down, causing service avalanche.

- Troubleshooting

  To solve the HotKey issue, perform the following steps:

  a. HotKeys must be considered in design to prevent them from being generated in a database.

  b. Add caches in the service side to reduce HotKey issues. Multi-level cache should be used to solve the HotKey issue (such as Redis + local level-2 cache).

c. Disable hotspot keys. For example, configure a whitelist for HotKeys on the service side to shield HotKeys as required.

- Check method

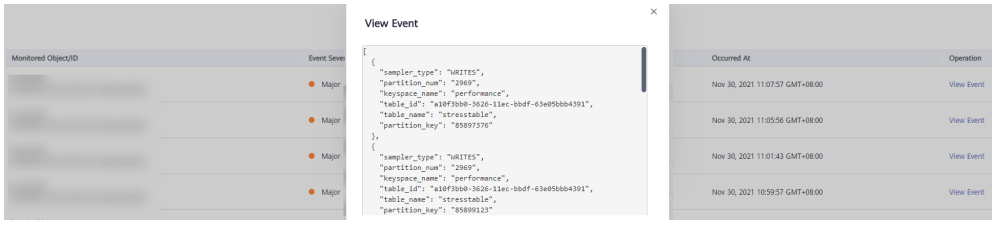
  A key whose access frequency is greater than 100,000 times per minute is defined as a HotKey.

  HotKey events are classified into the following types: One is the Writes event, indicating a write hotspot, and the other is the Reads event, indicating a read hotspot.

  GeminiDB Cassandra provides HotKey monitoring and alarms. On the Cloud Eye console, you can configure HotKey alarms for instances. For details, see **Configuring Alarm Rules**.

  When a BigKey event occurs, the system sends a warning notification immediately. You can **view the event data** on the Cloud Eye page and handle the event in a timely manner to prevent service fluctuation.

Figure 4-2 HotKey alarm



HotKey alarm description:

```
{
  "sampler_type": "WRITES",         //Sampling type. The value can be WRITES or READS. WRITES indicates write, and READS indicates read.
  "partition_num": "2969",          //Hotspot times of a partition key
  "keyspace_name": "performance",   //Keyspace name
  "table_id": "a10f3bb0-3626-11ec-bbdf-63e05bbb4391",   //Table ID
  "table_name": "stresstable",   //Table name
  "partition_key": "85897376"   //The value of the hotspot partition key.
}
```

Summary

If you use GeminiDB Cassandra API for online services, follow related rules to minimize risks in the development and design phase.

- The design of any table must consider whether HotKey or BigKey will be generated and whether load skew will occur.
- A data expiration mechanism must be established to prevent data from growing infinitely.
- In read-intensive scenarios, a cache mechanism needs to be added to handle read hotspots and improve query performance.
- A threshold must be set for each primary key and row. Otherwise, the database performance and stability will be affected. If the threshold is exceeded, optimize the settings in a timely manner.

4.3.5 How Do I Set Up a Materialized View?

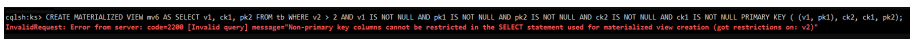
Concept

A materialized view is a standard CQL table that automatically maintains the consistency between the data that meets certain conditions and the data in the base tables.

Constraints

- The primary key of a materialized view must contain all primary keys of the base table. Static columns cannot be included in a materialized view.
- All columns that are part of the view primary key are restricted by the "IS NOT NULL" restriction, meaning that they cannot be null.
- In a materialized view, a CQL row must be mapped from the base table to another row of the view, meaning that the rows of the view and base table correspond to each other.
- The WHERE condition of the SELECT statement does not constrain non-primary key columns in a view, except the IS NOT NULL condition.

  Figure 4-3 Example value

- Static columns, counter, superColumn, and duration types are not supported.

Setting Up a Materialized View

1. Insert a record into the base table and query the result.

   Example:

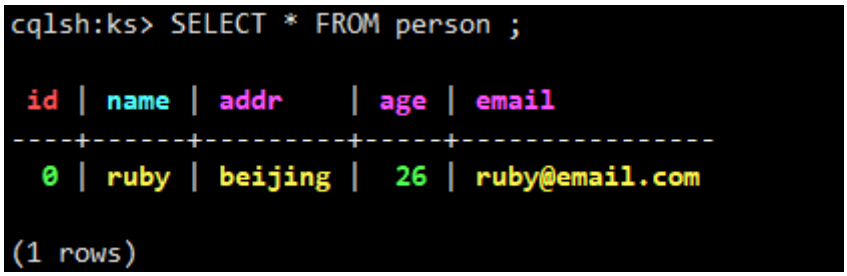
   ```
   CREATE TABLE person (
     id int,
     name text,
     addr text,
     age int,
     email text,
     PRIMARY KEY (id, name));
   ```

   Insert a record.

   insert into person(id, name, age, addr, email) values (0, 'ruby', 26, 'beijing', 'ruby@email.com');

   Query the result.

   Figure 4-4 Querying the result

2. Create a materialized view.

   CREATE MATERIALIZED VIEW person_addr AS

   SELECT * from person WHERE id IS NOT NULL AND addr IS NOT NULL AND name IS NOT NULL

   primary key (addr, id, name);

   The **system_schema.views** table records the association between views and base tables.

   Figure 4-5 Mapping between views and base tables

   The query results that do not meet the condition are not displayed, for example, IS NOT NULL.

3. Insert a record in which the **addr** value is **null**.

   insert into person(id, name, age, addr, email) values (1, 'mike', 30, null, 'mike@email.com');

   Query the data in the base table and materialized view.

   Figure 4-6 Querying the result

4. Delete the materialized view.

   DROP MATERIALIZED VIEW person_adder;

Figure 4-7 Deleting a view

```
cqlsh:ks> DROP MATERIALIZED VIEW person_addr ;
cqlsh:ks>
```

4.3.6 How Do I Use a Secondary Index?

Concept

In a GeminiDB Cassandra database, a primary key is the primary index, which can be used to query records. If you want to query records without the primary key, you can use secondary indexes.

Secondary Index Principles

A secondary index creates a hidden indexed table. The primary key becomes one of the columns in the hidden table.

Assume that there is a **playlists** table. The table structure is as follows:

```
CREATE TABLE playlists (
id int,
song_id int,
song_order int,
album text,
artist text,
title text,
PRIMARY KEY (id, song_id));
```

The query result is as follows.

Figure 4-8 Querying the result

```
id | song_id | album      | artist    | song_order | title
---+---------+------------+-----------+------------+------------
 1 |       1 | [illegible] | [illegible] |          1 | [illegible]
```

If an index is created for the **artist** field, the hidden table structure is as follows.

Figure 4-9 Querying the result

```
 artist    | id
-----------+----
 [illegible] |  1

(1 rows)
```

artist is the primary key of the index table. **id** and **song_id**, functioning as the primary key of the original table, become common columns.

In Which Scenario Is the Index Not Recommended?

- Too many duplicate values exist in a column.

  For example, if a table contains 100 million records and the values of **artist** are the same, you are not advised to index the **artist** column.
- The **counter** column cannot be indexed.
- Columns that are frequently updated or deleted.

How Do I Use an Index?

1. Creating an index

   CREATE INDEX artist_names ON playlists(artist);

   Note: If the original table contains a large amount of data, indexed data needs to be rebuilt before queries.

   You can query the **IndexInfo** table to check whether the index is recreated. If the name of the created index exists, it indicates that the indexed data has been rebuilt.

   Figure 4-10 Querying the result

   ```
   cqlsh:ks> SELECT *from system."IndexInfo";

    table_name | index_name
   ------------+--------------
             ks | artist_names

   (1 rows)
   ```

2. Query records by indexed column.

   Figure 4-11 Querying the result

   ```
   cqlsh:ks> SELECT *from playlists where artist ='[illegible]';

    id | song_id | album      | artist    | song_order | title
   ----+---------+------------+-----------+------------+------------
     1 |       1 | [illegible] | [illegible] |          1 | [illegible]

   (1 rows)
   ```

   NOTE

   Each table can have multiple indexes, but the write performance may be affected.

4.3.7 How Do I Set Paging Query with Java?

Specifying the Number of Rows Fetched in Each Page

The fetch size specifies how many rows will be fetched at once. When you create a cluster connection, you can set a fetch size for it.

```
Cluster cluster = Cluster.builder()
        .addContactPoint(contactPoint)
```



```
            .withPort(8636)
            .withQueryOptions(new QueryOptions().setFetchSize(20))
            .build();
```

After the setting is successful, for all sessions spawned with this configuration, the configured number of rows is fetched from the server at a time. When the cache (20 rows) is exhausted, the system triggers a request for fetching another 20 rows from the server and there can be a waiting period.

Obtaining the Next Page in Advance

If you need to manually fetch more rows in advance to avoid waiting and save them to the current result set, refer to the following code. When the result set has 10 rows left, submit a parallel request for fetching more rows from the server.

```
ResultSet rs = session.execute("select * from space3.table3;");
for (Row row : rs) {
   if (rs.getAvailableWithoutFetching() == 10 && !rs.isFullyFetched()){
      System.out.println("pre-fetch more rows. ");
      rs.fetchMoreResults();
   }
   System.out.println(row);
}
```

Saving and Reusing the Paging State

1. Save the current paging state.

   ```
   PagingState pagingState = resultSet.getExecutionInfo().getPagingState();
   String string = pagingState.toString();
   byte[] bytes = pagingState.toBytes();
   ```

2. Load and reuse the current paging state.

   ```
   PagingState pagingState = PagingState.fromString(string);
   Statement st = new SimpleStatement("your query");
   st.setPagingState(pagingState);
   ResultSet rs = session.execute(st);
   ```

   Note: The paging state can only be collected, stored, and reused. They cannot be modified or applied to other query statements.

NOTE

GeminiDB Cassandra API does not support offset queries, which means that you cannot skip any part of the result set and cannot fetch results within the specified index range. If you want to use offset queries, you can emulate them on the client side. You will get all results in order, but you can delete results that you do not need. For more advanced usage and introduction, see **DataStax Java Driver 3.11**.

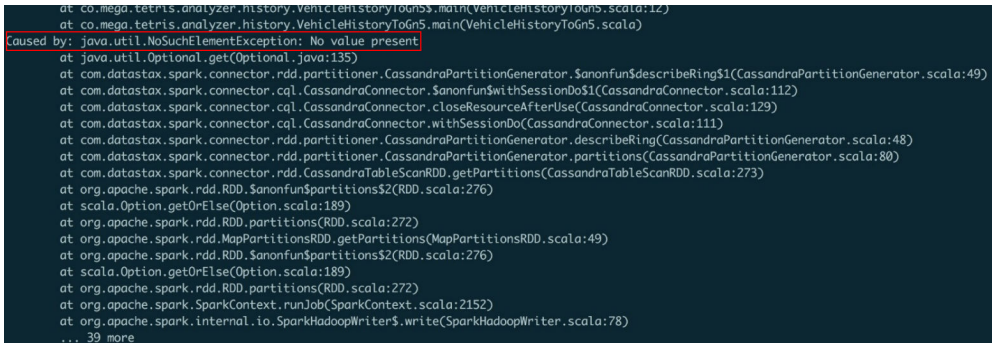
4.4 Database Connection

4.4.1 What Can I Do If Spark Failed to Connect to Cassandra?

Symptom

You used Spark to connect to the open-source Cassandra, data can be read properly, but an error was reported during the connection.

Error message is as follows.



Configuration Details

The following shows the components and account details.

- Component configuration details

  Table 4-5 Configuration details

Component	Version
spark-cassandra-connector	2.5.1
spark	2.5.1
Open-source Cassandra	3.11
scala	2.12

- User: **user1** (created by user **rwuser**)

Possible Cause

- **user1** does not have the permission to query the keyspace system.
- The Spark version is incorrect.

Solution

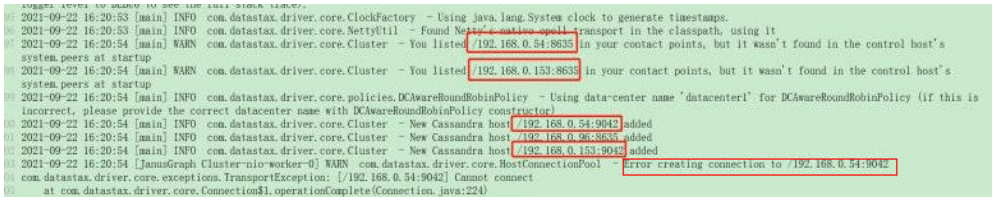
1. Grant the keyspace system query permission to **user1** as user **rwuser**.
2. Use spark-cassandra-connector 2.4.1.

4.4.2 What Can I Do If an Error Occurs When I Use Java Driver and a Mapped IP Address to Connect to a Database?

Symptom

When you use Java Driver to connect to a GeminiDB Cassandra instance, a session was established using the mapped IP address, rather than the database private IP address, over port 8635. However, an error was found in the connection log, and connection information of port 9042 was displayed.

Figure 4-12 Log information



Possible Cause

Java Driver was not used correctly, as shown in **Figure 4-13**. Do not use addContactPointsWitPorts when using Java Driver and do not map each IP address.

Figure 4-13 Incorrect usage of the Java Driver

```
public static void connectToCluster() {
    Cluster cluster = Cluster.builder()
            .addContactPointsWithPorts(new InetSocketAddress("192.168.0.96", 8635))
            .addContactPointsWithPorts(new InetSocketAddress("192.168.0.54", 8635))
            .addContactPointsWithPorts(new InetSocketAddress("192.168.0.153", 8635))

            .addContactPointsWithPorts(new InetSocketAddress("124.70.177.38", 38635))
            .addContactPointsWithPorts(new InetSocketAddress("124.70.177.38", 28635))
            .addContactPointsWithPorts(new InetSocketAddress("124.70.177.38", 18635))
            .withReconnectionPolicy(new ConstantReconnectionPolicy(100L))
            .withCredentials(USER, PASSWORD)
            .withoutJMXReporting()
            .build();

    Session session = cluster.connect();

    System.out.println                                    ;
    String queryCQL = "SELECT peer,data_center,host_id,rpc_address FROM system.peers ;";
    ResultSet rs = session.execute(queryCQL);
    List<Row> dataList = rs.all();
    System.out.println                                    ;
    System.out.println(dataList.toString());
    System.out.println                                    ;
    System.out.println                                    ;

    cluster.close();
    System.out.println("connectToCluster finished");
}
public static void main(String[] args) {
    connectToCluster();
}
```

Solution

Use the private IP address provided by the GeminiDB Cassandra database and change the port to port 8635.

The following figure shows the IP address and port.

```
Cluster cluster = Cluster.builder().addContactPoint( address: "192.168.0.96").withPort(8635).build();
```


4.4.3 How Can I Create and Connect to an ECS?

1. To create an ECS, see *Elastic Cloud Server User Guide*.
   - The ECS to be created must be in the same VPC with the GeminiDB Cassandra instance to which it connects.
   - Configure the security group rules to allow the ECS to access to the instance.
2. To connect to an ECS, see "Logging in to an ECS" *Getting Started with Elastic Cloud Server User Guide*.

4.4.4 Can I Change the VPC of a GeminiDB Cassandra Instance?

After a GeminiDB Cassandra instance is created, the VPC where the instance resides cannot be changed.

However, you can change a VPC by restoring the full backup of your instance to the VPC you want to use. For details, see **Restoring Data to a New Instance**.

4.5 Backup and Restoration

4.5.1 How Long Does GeminiDB Cassandra Store Backup Data?

Automated backup data is kept based on the backup retention period you specified. There is no limit for the manual backup retention period. You can delete manually backup files as needed.

4.6 Instance Freezing, Release, Deletion, and Unsubscription

Why Are My GeminiDB Cassandra Instances Released?

If your subscriptions have expired but not been renewed, or you are in arrears due to insufficient balance, your instances enter a grace period. If you do not renew the subscriptions or top up your account after the grace period expires, your instances will enter a retention period and become unavailable. If you still do not renew them or top up your account after the retention period ends, your instances will be released and your data stored will be deleted.

Why Are My GeminiDB Cassandra Instances Frozen?

Your instances may be frozen for a variety of reasons. The most common reason is that you are in arrears.

Can I Still Back Up Data If My Instances Are Frozen?

No. If your instances are frozen because your account is in arrears, go to top up your account to unfreeze your instances and then back up instance data.

How Do I Unfreeze My Instances?

If your instances are frozen because your account is in arrears, you can unfreeze them by renewing them or topping up your account. Frozen instances can be renewed, released, or deleted. Expired yearly/monthly instances cannot be unsubscribed from, while those that have not expired can.

What Impacts Does Instance Freezing, Unfreezing or Release Have on My Services?

- After an instance is frozen:
  - It cannot be accessed, and your services will be interrupted. For example, if a GeminiDB Cassandra instance is frozen, it cannot be connected.
  - If they are yearly/monthly resources, no changes can be made to them.
  - It can be unsubscribed from or deleted manually.
- After it is unfrozen, you can connect to it again.
- Releasing an instance means deleting it. Before the deletion, GeminiDB Cassandra API determines whether to **move the instance to the recycle bin** based on the recycling policy you specified.

How Do I Renew My Instances?

After a yearly/monthly GeminiDB Cassandra instance expires, you can renew it on the **Renewals** page. For details, see **Renewal Management**.

Can My Instances Be Recovered After They Are Released or Unsubscribed From?

If your instance is moved to the recycle bin after being deleted, you can recover it from the recycle bin by referring to **Recycling a GeminiDB Cassandra Instance**. If the recycling policy is not enabled, you cannot recover it.

When you unsubscribe from an instance, confirm the instance information carefully. If you have unsubscribed from an instance by mistake, purchase a new one.

How Do I Delete a GeminiDB Cassandra Instance?

- To delete a pay-per-use instance, see **Deleting Instance**.
- To delete a yearly/monthly instance, see **Unsubscribing from a Yearly/Monthly Instance**.

5 GeminiDB (DynamoDB API Compatible) Instance

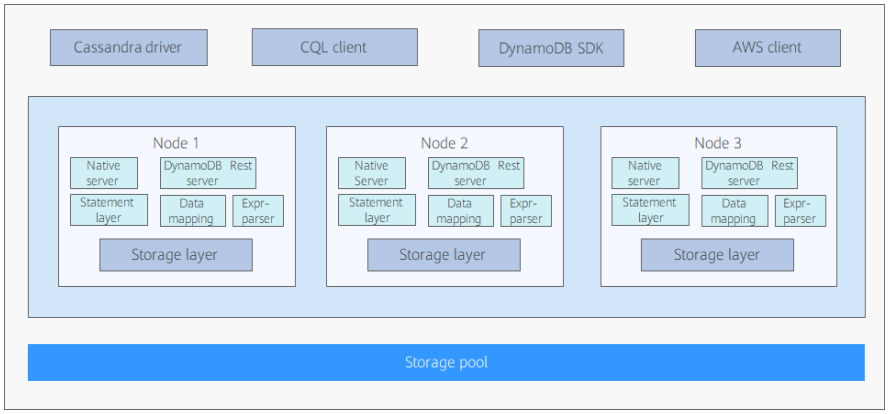
5.1 What Is GeminiDB DynamoDB API?

GeminiDB is compatible with DynamoDB APIs. You can access DynamoDB through SDKs or CLIs. GeminiDB DynamoDB API is completely compatible with Amazon DynamoDB, so you can smoothly migrate data from DynamoDB to GeminiDB without refactoring.

Product Architecture

GeminiDB DynamoDB API is embedded in kernel services of GeminiDB Cassandra API and communicates with DynamoDB over HTTPS. In addition, GeminiDB DynamoDB API supports CQL. The following figure shows the overall architecture.

Figure 5-1 GeminiDB DynamoDB API architecture



GeminiDB strictly complies with the DynamoDB syntax and data model, so DynamoDB applications can be easily migrated to GeminiDB DynamoDB instances. In addition, GeminiDB DynamoDB API provides automated management and O&M functions, such as automatic backup, fault detection, and fault tolerance.

Highlights

- Low costs

  Companies produce more data as services evolve. Accumulated read and write requests to DynamoDB databases incur higher costs. GeminiDB DynamoDB API is cost-effective. It provides higher throughput at half the cost of DynamoDB, especially when there is a large amount of data.
- Flexible deployment

  DynamoDB is a fully managed service. You need to comply with certain deployment modes and rules when using it. In contrast, GeminiDB DynamoDB API allows you to flexibly deploy services. You can deploy services on private or public clouds instead of being restricted to expensive platforms. GeminiDB DynamoDB API ensures stable and efficient services in these deployments.
- Smooth migration

  GeminiDB DynamoDB API is fully compatible with DynamoDB syntax and data models, so you can migrate data from DynamoDB to GeminiDB DynamoDB API without refactoring. The migration process is more simple, efficient, and convenient, with minimal impacts on workloads.

5.2 Connecting to a GeminiDB DynamoDB Instance

This section describes how to connect to a GeminiDB DynamoDB instance using Java or Python.

Prerequisites

- A GeminiDB DynamoDB instance has been created.
- For details about how to create an ECS, see **Purchasing an ECS** in *Getting Started with Elastic Cloud Server*.
- JDK has been installed on the ECS.

Obtaining the IP Address of a GeminiDB DynamoDB Instance

Step 1 Log in to the Huawei Cloud console.

Step 2 In the service list, choose **Databases** > **GeminiDB**.

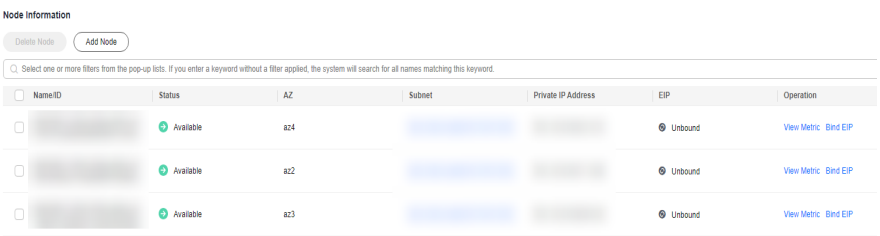
Step 3 On the **Instances** page, click the name of the target instance.

The GeminiDB DynamoDB instance uses port 8000.

Method 1

In the **Node Information** area on the **Basic Information** page, obtain the private IP address or EIP of each node.

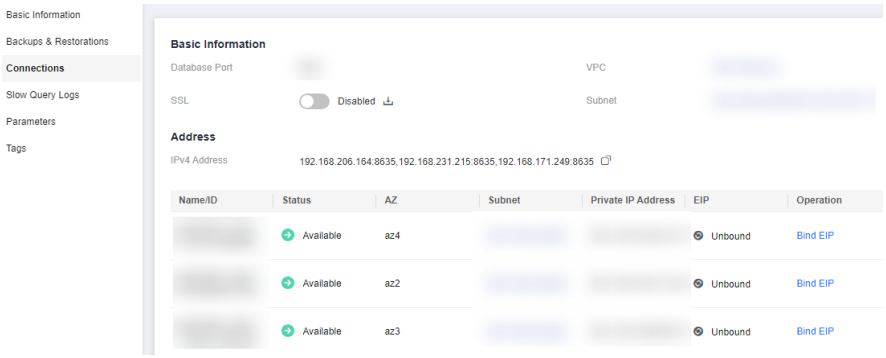
Figure 5-2 Obtaining IP addresses



Method 2

In the navigation pane, choose **Connections** to obtain the private IP address and EIP of the instance.

Figure 5-3 Obtaining IP addresses



----End

Using a Load Balancing Plug-in to Connect to a GeminiDB DynamoDB Instance

Step 1 To obtain the JAR package and plug-in code, choose **Service Tickets > Create Service Ticket** in the upper right corner of the console and contact customer service personnel.

Step 2 Replace the IP address in the following code example with the IP address queried in **Step 3**.

The following is a Java code example:

```
package com.huawei.dbs.test;

import com.huawei.dbs.RequestHandler;

import com.amazonaws.SDKGlobalConfiguration;
import com.amazonaws.auth.AWSCredentialsProvider;
import com.amazonaws.auth.AWSStaticCredentialsProvider;
import com.amazonaws.auth.BasicAWSCredentials;
import com.amazonaws.services.dynamodbv2.AmazonDynamoDB;
import com.amazonaws.services.dynamodbv2.AmazonDynamoDBClientBuilder;
import com.amazonaws.services.dynamodbv2.document.DynamoDB;
import com.amazonaws.services.dynamodbv2.document.TableCollection;

import java.net.URI;


public class V1Demo {
    public static AWSCredentialsProvider myCredentials = new AWSStaticCredentialsProvider(
        new BasicAWSCredentials("your_ak", "your_sk"));
    public static String ip = "***.***.***.***";
    public static void main(String[] args) {
        disableCertificateChecks();
        AmazonDynamoDB client = AmazonDynamoDBClientBuilder.standard()
            .withRegion("region-a")
            .withRequestHandlers(new RequestHandler(URI.create("http://" + ip + "8000")))
            .withCredentials(myCredentials)
```



```
            .build();
        DynamoDB dynamoDB = new DynamoDB(client);
        TableCollection res = dynamoDB.listTables();
        System.out.println(res);
    }

}
```

The following is a Python code example:

```
#!/usr/bin/python
import boto3
import boto3_lb

ip = '***.***.***.***'
url = boto3_lb.setup([ip], 'http', 8000, 'fake.url.com')
dynamodb = boto3.resource('dynamodb',
                          endpoint_url=url,
                          aws_access_key_id='ak',
                          aws_secret_access_key='sk',
                          region_name="region-a")

url = boto3_lb.setup_single_connection([ip], 'http', 8000, 'fake.url.com')

dynamodb = boto3.resource('dynamodb',
                          endpoint_url=url,
                          aws_access_key_id='ak',
                          aws_secret_access_key='sk',
                          region_name="region-a")
```

----End

GeminiDB DynamoDB API is completely compatible with DynamoDB. For details about common operations, see official DynamoDB documents.

5.3 API Compatibility

This section describes APIs supported by GeminiDB DynamoDB instances and precautions.

Usage Notes

- You can add, delete, and modify tables of GeminiDB DynamoDB instances only through DynamoDB APIs. Tables can only be read on the console.
- Tables created through non-DynamoDB APIs cannot be used on GeminiDB DynamoDB instances.
- In the current version, traffic statistics, traffic limit, and transaction functions (TransactWriteItems and TransactGetItems) are not supported.
- In the current version, on-demand backup APIs CreateBackup, DescribeBackup, DeleteBackup, ListBackups, and RestoreTableFromBackup are not supported.
- Hash or partition keys determine where items are stored in an internal storage device of a GeminiDB DynamoDB instance. Amazon DynamoDB uses a dedicated hash function, while GeminiDB DynamoDB API uses the Murmur3 algorithm. As a result, the sequence of data returned by a scan operation is different.
- A load balancing plug-in evenly sends requests from applications to all nodes. If this plug-in is not used, requests are sent only to a specified endpoint address. To obtain the plug-in package, choose **Service Tickets > Create**

Service Ticket in the upper right corner of the console and contact customer service personnel of GeminiDB DynamoDB API.

- The table or index name cannot contain periods (.).
- Due to different encoding methods, the size of each row of data in a table on a GeminiDB DynamoDB instance is not strictly limited to 400 KB.
- All DDL APIs are synchronous APIs.

Load Balancing

DynamoDB applications typically connect to a URL and then distribute requests within a server. To evenly distribute the requests of GeminiDB DynamoDB API to all nodes, a cost-effective and small-capacity database is attached to DynamoDB SDK. Before DynamoDB SDK sends requests, an interceptor changes the destination address in the request header to an IP address of any available node in the current cluster to achieve load balancing. The following is an example:

```
TABLE_NAME ='user'

// seed url
URL = boto3_lb.setup(['***.***.***.***'],'http',8000,'fake.domain.com')

dynamodb = boto3.resource('dynamodb', endpoint_url=URL,region_name='None',
aws_access_key_id='None', aws_secret_access_key='None')
```

Supported Expressions

ConditionExpression of GeminiDB DynamoDB API supports the following function syntax:

```
function ::=
    attribute_exists (path)
    | attribute_not_exists (path)
    | attribute_type (path, type)
    | begins_with (path, substr)
    | contains (path, operand)
```

The value of **operand** in **contains(path,operand)** can only be an attribute value (that is, the input value placeholder).

UpdateExpression of GeminiDB DynamoDB API supports the following syntax:

```
update-expression ::=
     [ SET action [, action] ... ]
     [ REMOVE action [, action] ...]
     [ ADD action [, action] ... ]
     [ DELETE action [, action] ...]

 set-action ::=
     path =value

 value ::=
     operand
     | operand '+' operand
     | operand '-' operand

 operand ::=
     path | function

 function ::=
     if_not_exists (path, value)
     | list_append (list1, list2)

 remove-action ::=
     path
```



```
add-action ::=
    path value

delete-action ::=
    path value
```

In the SET path = operand syntax, the value of **operand** cannot be set to **path**.
In the SET path = operand1 '+'|'-' operand2 syntax, the value of **operand1** must be set to **path**, and the value of **operand2** must be an attribute value.
In the SET path = if_not_exists (path, value) syntax, values of the two **path** parameters must be the same, and the values must be expression attribute values.
All values can only be placeholder expressions, for example, **:placeholder**.

Compatible APIs

Table 5-1 API list

API
CreateTable
UpdateTable
DescribeTable
ListTables
DeleteTable
PutItem
UpdateItem
GetItem
DeleteItem
BatchWriteItem
BatchGetItem
Query
Scan
UpdateTimeToLive
DescribeTimeToLive

Table 5-2 CreateTable API Compatibility

API	Parameter		Mandatory	Supported	Description
CreateTable	Request parameters	AttributeDefinitions	Yes	Yes	-
		KeySchema	Yes	Yes	-

API	Parameter		Mandatory	Supported	Description
		TableName	Yes	Yes	The value contains 3 to 48 characters. The regular expression is ([\w-]+).
		BillingMode	No	No	Billing parameter, which is not supported.
		GlobalSecondaryIndexes	No	Yes	-
		LocalSecondaryIndexes	No	Yes	-
		ProvisionedThroughput	Yes	Yes	-
		SSESpecification	No	No	This parameter is not supported in the current version and is being planned.
		StreamSpecification	No	Yes	-
		Tags	No	No	This parameter is not supported in the current version and is being planned.
	Response parameter	TableDescription	-	Yes	-

Table 5-3 UpdateTable API Compatibility

API	Parameter		Mandatory	Supported	Description
UpdateTable	Request parameters	AttributeDefinitions	No	Yes	-
		BillingMode	No	No	Billing parameter, which is not supported.

API	Parameter		Mandatory	Supported	Description
		GlobalSecondaryIndexesUpdates	Yes	Yes	CREATE and DELETE statements are supported. UPDATE statements are not supported.
		ProvisionedThroughput	No	No	Billing parameter, which is not supported.
		ReplicaUpdates	No	No	Strong consistency parameter by default
		SSESpecification	No	No	This parameter is not supported in the current version and is being planned.
		StreamSpecification	No	Yes	-
		TableName	Yes	Yes	-
	Response parameter	TableDescription	-	Yes	-

Table 5-4 DescribeTable API Compatibility

API	Parameter		Mandatory	Supported	Description
DescribeTable	Request parameter	TableName	Yes	Yes	-
	Response parameter	Table	-	Yes	The **itemCount** field is not supported.

Table 5-5 ListTables API Compatibility

API	Parameter		Mandatory	Supported	Description
ListTables	Request parameters	ExclusiveStartTable-Name	No	Yes	-
		Limit	No	Yes	-
	Response parameters	LastEvaluatedTable-Name	-	Yes	-
		TableNames	-	Yes	-

Table 5-6 DeleteTable API Compatibility

API	Parameter		Mandatory	Supported	Description
DeleteTable	Request parameter	TableName	Yes	Yes	-
	Response parameter	TableDescription	-	Yes	-

Table 5-7 PutItem API Compatibility

API	Parameter		Mandatory	Supported	Description
PutItem	Request parameters	Item	Yes	Yes	-
		TableName	Yes	Yes	-
		ConditionalOperator	No	No	Deprecated parameter. For details, see **ConditionExpression**.
		ConditionExpression	No	Yes	-
		Expected	No	No	Deprecated parameter. For details, see **ConditionExpression** in this API.

API	Parameter		Mandatory	Supported	Description
		ExpressionAttributeNames	No	Yes	-
		ExpressionAttributeValues	No	Yes	-
		ReturnConsumedCapacity	No	No	Billing parameter, which is not supported.
		ReturnItemCollectionMetrics	No	No	Statistics parameter, which is not supported.
		ReturnValues	No	Yes	-
	Response parameters	Attributes	-	Yes	-
		ConsumedCapacity	-	No	Billing parameter, which is not supported.
		ItemCollectionMetrics	-	No	Statistics parameter, which is not supported.

Table 5-8 UpdateItem API Compatibility

API	Parameter		Mandatory	Supported	Description
UpdateItem	Request parameters	Key	Yes	Yes	-
		TableName	Yes	Yes	-
		AttributeUpdates	No	No	Deprecated parameter. For details, see **UpdateExpression** in this API.
		ConditionalOperator	No	No	Deprecated parameter. For details, see **UpdateExpression** in this API.

API	Parameter		Mandatory	Supported	Description
		ConditionExpression	No	Yes	-
		Expected	No	No	Deprecated parameter. For details, see **UpdateExpression** in this API.
		ExpressionAttributeNames	No	Yes	-
		ExpressionAttributeValues	No	Yes	-
		ReturnConsumedCapacity	No	No	Billing parameter, which is not supported.
		ReturnItemCollectionMetrics	No	No	Statistics parameter, which is not supported.
		ReturnValues	No	Yes	-
		UpdateExpression	No	Yes	-
	Response parameters	Attributes	-	Yes	-
		ConsumedCapacity	-	No	Billing parameter, which is not supported.
		ItemCollectionMetrics	-	No	Statistics parameter, which is not supported.

Table 5-9 GetItem API Compatibility

API	Parameter		Mandatory	Supported	Description
GetItem	Request parameters	Key	Yes	Yes	-
		TableName	Yes	Yes	-

API	Parameter		Mandatory	Supported	Description
		AttributesToGet	No	No	Deprecated parameter. For details, see **ProjectionExpression** in this API.
		ConsistentRead	No	No	Strong consistency parameter by default
		ExpressionAttributeNames	No	Yes	-
		ProjectionExpression	No	Yes	-
		ReturnConsumedCapacity	No	No	Billing parameter, which is not supported.
	Response parameters	ConsumedCapacity	-	No	Billing parameter, which is not supported.
		Item	-	Yes	-

Table 5-10 DeleteItem API Compatibility

API	Parameter		Mandatory	Supported	Description
DeleteItem	Request parameters	Key	Yes	Yes	-
		TableName	Yes	Yes	-
		ConditionalOperator	No	No	Deprecated parameter. For details, see **ConditionExpression** in this API.
		ConditionExpression	No	Yes	-
		Expected	No	No	Deprecated parameter. For details, see **ConditionExpression** in this API.

API	Parameter		Mandatory	Supported	Description
		ExpressionAttributeNames	No	Yes	-
		ExpressionAttributeValues	No	Yes	-
		ReturnConsumedCapacity	No	No	Billing parameter, which is not supported.
		ReturnItemCollectionMetrics	No	No	Statistics parameter, which is not supported.
		ReturnValues	No	Yes	-
	Response parameters	Attributes	-	Yes	-
		ConsumedCapacity	-	No	Billing parameter, which is not supported.
		ItemCollectionMetrics	-	No	Statistics parameter, which is not supported.

Table 5-11 BatchWriteItem API Compatibility

API	Parameter		Mandatory	Supported	Description
BatchWriteItem	Request parameters	RequestItems	Yes	Yes	-
		ReturnConsumedCapacity	No	No	Billing parameter, which is not supported.
		ReturnItemCollectionMetrics	No	No	Statistics parameter, which is not supported.
	Response parameters	ConsumedCapacity	-	No	Billing parameter, which is not supported.

API	Parameter		Mandatory	Supported	Description
		ItemCollectionMetrics	-	No	Statistics parameter, which is not supported.
		UnprocessedItems	-	Yes	-

Table 5-12 BatchGetItem API Compatibility

API	Parameter		Mandatory	Supported	Description
BatchGetItem	Request parameters	RequestItems	Yes	Yes	None
		ReturnConsumedCapacity	No	No	Billing parameter, which is not supported.
	Response parameters	ConsumedCapacity	-	No	Statistics parameter, which is not supported.
		Responses	-	Yes	-
		UnprocessedKeys	-	Yes	-

Table 5-13 Query API Compatibility

API	Parameter		Mandatory	Supported	Description
Query	Request parameters	TableName	Yes	Yes	-
		AttributesToGet	No	No	Deprecated parameter. For details, see **ProjectionExpression** in this API.
		ConditionalOperator	No	No	Deprecated parameter. For details, see **ProjectionExpression** in this API.
		ConsistentRead	No	No	Strong consistency parameter by default

API	Parameter		Mandatory	Supported	Description
		ExclusiveStartKey	No	Yes	-
		ExpressionAttributeNames	No	Yes	-
		ExpressionAttributeValues	No	Yes	-
		FilterExpression	No	Yes	-
		IndexName	No	Yes	-
		KeyConditionExpression	No	Yes	-
		KeyConditions	No	No	Deprecated parameter. For details, see **KeyConditionExpression** in this API.
		Limit	No	Yes	-
		ProjectionExpression	No	Yes	-
		QueryFilter	No	No	Deprecated parameter. For details, see **FilterExpression** in this API.
		ReturnConsumedCapacity	No	No	Billing parameter, which is not supported.
		ScanIndexForward	No	Yes	-
		Select	No	Yes	SELECT COUNT is not supported.
	Response parameters	ConsumedCapacity	-	No	Billing parameter, which is not supported.
		Count	-	Yes	-
		Items	-	Yes	-
		LastEvaluatedKey	-	Yes	-
		ScannedCount	-	Yes	-

Table 5-14 Scan API Compatibility

API	Parameter		Mandatory	Supported	Description
Scan	Request parameters	TableName	Yes	Yes	-
		AttributesToGet	No	No	Deprecated parameter. For details, see **ProjectionExpression** in this API.
		ConditionalOperator	No	No	Deprecated parameter. For details, see **ConditionExpression** in this API.
		ConsistentRead	No	No	Strong consistency parameter by default
		ExclusiveStartKey	No	Yes	-
		ExpressionAttributeNames	No	Yes	-
		ExpressionAttributeValues	No	Yes	-
		FilterExpression	No	Yes	-
		IndexName	No	Yes	-
		Limit	No	Yes	-
		ProjectionExpression	No	Yes	-
		ReturnConsumedCapacity	No	No	Billing parameter, which is not supported.
		ScanFilter	No	No	Deprecated parameter. For details, see **FilterExpression** in this API.
		Segment	No	Yes	-
		Select	No	Yes	SELECT COUNT is not supported.
		TotalSegments	No	Yes	-

API	Parameter		Mandatory	Supported	Description
	Response parameters	ConsumedCapacity	No	No	Billing parameter, which is not supported.
		Count	-	Yes	-
		Items	-	Yes	-
		LastEvaluatedKey	-	Yes	-
		ScannedCount	-	Yes	-

Table 5-15 UpdateTimeToLive API Compatibility

API	Parameter		Mandatory	Supported	Description
UpdateTimeToLive	Request parameters	TableName	Yes	Yes	-
		TimeToLiveSpecification	Yes	Yes	-
	Response parameter	TimeToLiveSpecification	-	Yes	-

Table 5-16 DescribeTimeToLive API compatibility

API	Parameter		Mandatory	Supported	Remarks
DescribeTimeToLive	Request parameter	TableName	Yes	Yes	-
	Response parameter	TimeToLiveSpecification	-	Yes	-

5.4 Migration from DynamoDB to GeminiDB (DynamoDB API Compatible)

GeminiX is a Huawei-developed tool that can seamlessly migrate data from DynamoDB to GeminiDB (DynamoDB API compatible) based on DynamoDB

semantics. This section describes how to migrate data from AWS DynamoDB to GeminiDB (DynamoDB API compatible).

Migration Principles

This solution consists of two modules: full migration and incremental migration. You need to create an ECS for data forwarding. GeminiX supports distributed migration and concurrent migration of multiple service tables on multiple servers.

- During full migration, DynamoDB scan APIs read all data by segment and write the data to the GeminiDB (DynamoDB API compatible) instance.
- During incremental migration, DynamoDB stream APIs synchronize incremental data to the GeminiDB (DynamoDB API compatible) instance.

Prerequisites

- A GeminiDB (DynamoDB API compatible) instance has been created and is running properly.
- An ECS with 16 vCPUs, 64 GB of memory, and 100 or more GB of disk space has been created. For details about how to create an ECS, see **Purchasing an ECS**" in *Getting Started with Elastic Cloud Server*.
- Data is transmitted between the AWS DynamoDB instance and ECS via private connections. Ensure that the networks between the ECS and AWS DynamoDB instance and between the ECS and GeminiDB (DynamoDB API compatible) instance are normal.
- The storage space of the GeminiDB (DynamoDB API compatible) instance cannot be less than that of the AWS DynamoDB instance.
- AWS DynamoDB stream interfaces have been enabled.

Procedure

Step 1 Deploy the required migration tool.

1. Obtain GeminiX.
2. Modify the **conf.yaml** configuration file and configure the following items:

```
source:
  endpoint: 'https://dynamodb.us-west-2.amazonaws.com' #Domain name of the AWS DynamoDB instance
  access_key_id: 'xx' #Access key ID of the AWS DynamoDB instance
secret_access_key: 'xx' #Secret access key of the AWS DynamoDB instance
target:
  endpoint: ***.***.***.*** #Domain name of the GeminiDB (DynamoDB API compatible) instance
  access_key_id: 'xx' #AK of the GeminiDB (DynamoDB API compatible) instance
  secret_access_key: 'xx' #SK of the GeminiDB (DynamoDB API compatible) instance
```

Step 2 Perform a full data migration.

Run the following command:

```
nohup ./bin/migration.sh fullsync -c conf/conf.yaml >run.log 2>&1 &
```

If information similar to the following is displayed, synchronizing full data is complete.

```
finish syncing all tables and indexes
```

Step 3 Migrate incremental data.

Before running the incremental migration command, configure connection information of the DynamoDB instance and GeminiDB (DynamoDB API compatible) instance and the name of the incremental migration table in GeminiX. For details, see **Step 1.2**.

Run the following command:

```
nohup ./bin/migration.sh incresync -c conf/conf.yaml >run.log 2>&1 &
```

Step 4 After the migration is complete, switch workloads to the GeminiDB (DynamoDB API compatible) instance.

----End

