# GeminiDB Cassandra

# User Guide

**Issue** 01

**Date** 2025-09-04

HUAWEI CLOUD COMPUTING TECHNOLOGIES CO., LTD.

# Contents



# 1 Service Overview

## 1.1 What Is GeminiDB Cassandra API?

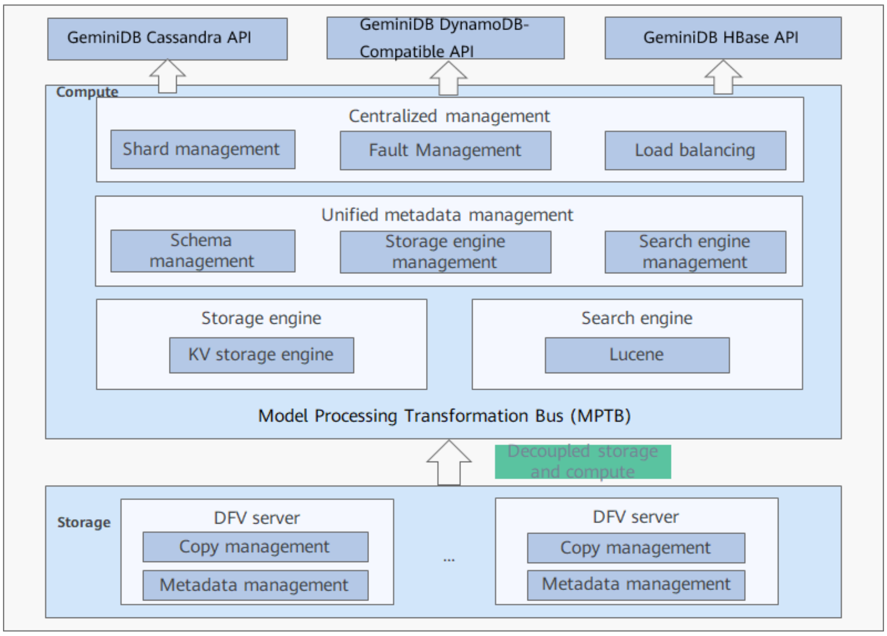
GeminiDB Cassandra API is a cloud-native NoSQL database service compatible with Cassandra, DynamoDB, and HBase. It supports Cassandra Query Language (CQL), which gives you SQL-like syntax. With robust security and reliability, GeminiDB Cassandra API offers ultra-high performance and addresses pain points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

## Architecture

The following figure shows the deployment architecture of GeminiDB Cassandra API.




## 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

| CPU Type | Flavor | vCPUs | Memory (GB) | Min. Storage Space (GB) | Max. Storage Space |
| --- | --- | --- | --- | --- | --- |
| x86 | geminidb.cassandra.large.4 | 2 | 8 | 10 | 96,000 |

| CPU Type | Flavor | vCPUs | Memory (GB) | Min. Storage Space (GB) | Max. Storage Space |
|---|---|---|---|---|---|
| | geminidb.cassandra.xlarge.4 | 4 | 16 | 10 | 96,000 |
| | geminidb.cassandra.xlarge.8 | 4 | 32 | 10 | 96,000 |
| | geminidb.cassandra.2xlarge.4 | 8 | 32 | 10 | 96,000 |
| | geminidb.cassandra.2xlarge.8 | 8 | 64 | 10 | 96,000 |
| | geminidb.cassandra.4xlarge.4 | 16 | 64 | 10 | 96,000 |
| | geminidb.cassandra.4xlarge.8 | 16 | 128 | 10 | 96,000 |
| | geminidb.cassandra.8xlarge.4 | 32 | 128 | 10 | 192,000 |
| | geminidb.cassandra.8xlarge.8 | 32 | 256 | 10 | 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. |
| 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. |
| Configuring SSL | SSL is being enabled or disabled. |
| Frozen | The instance is frozen because your balance drops to or below zero. |

| Status | Description |
| --- | --- |
| 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 Database Usage

## Mandatory Constraints

**General rules**

- If the size of a request or any configuration value in it exceeds the preset alarm threshold, the client receives a warning and Cloud Eye generates an alarm. For some requests, events are also generated.
- If the size of a request or any configuration value in it exceeds the preset failure threshold, the request fails and an event is generated.

**Specific constraints**

- Data volume of a single table row:

  - When a database reads data, it will combine multiple writes of the same primary key into one row. When the amount of data in a single row exceeds the preset threshold, an alarm and event are generated.
- A single partition:
  - The database background scanning task periodically collects partitions, rows in a partition, and total size of all rows in a partition. When the amount of data in a single row exceeds the preset threshold, an alarm and event are generated.
- Number of elements of the collection type:
  - When your database reads data (of the types like Map, List, Set, and Tuple) in the collection column, the database will count all elements in the column. An alarm and event are generated when the number of elements exceeds the preset threshold.
- Amount of data returned by a query:
  - Before a response is returned to the client, the database checks the amount of data in the response. If the amount exceeds the preset threshold, an alarm is generated or the request fails.
- Number of tombstones returned by a query:
  - After a query request is submitted, the database checks the number of tombstones scanned. When the number exceeds the preset threshold, an alarm is generated or the request fails.

**Table 1-6** Thresholds for GeminiDB Cassandra constraints

| Constraint | Alarm Threshold | Failure Threshold |
| --- | --- | --- |
| Amount of data in a single row | 100 KB | - |
| A single partition | • Rows: 100,000<br>• Size: 100 MB | - |
| Number of elements of the collection type | 500 | - |
| Amount of data returned by a query | 2 MB | 100 MB |
| Number of tombstones returned by a query | 1,000 | 100,000 |

## Optional Constraints

- Do not use ALLOW FILTERING:
  - A warning is returned when the ALLOW FILTERING statement is executed.
- Use a proper limit value for a RANGE query:
  - Using a proper limit for prefetching can accelerate RANGE queries. You can check whether a limit value is proper based on the average limit values of all monitoring items.

## 1.5.4 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.

## 1.5.5 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.

## 1.5.6 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.

# 1.6 Constraints

The following tables list the constraints designed to ensure stability and security of GeminiDB Cassandra instances.

## Specifications

**Table 1-7** Specifications

| Resource Type | Specifications | Description |
| --- | --- | --- |
| CPU and memory | GeminiDB Cassandra cluster instances are supported. | • For details about specifications of different instance types, see **Instance Specifications**.<br>• You can change the specifications to meet your service requirements by following **Changing vCPUs and Memory**. |
| Storage Space | The storage space depends on the selected **instance specifications**. | Storage capacity can be scaled up or down. For details, see **Overview**. |

## Quotas

**Table 1-8** Quotas

| Resource Type | Constraint | Description |
| --- | --- | --- |
| Tag | A maximum of 20 tags can be added for each instance. | For more information, see **Managing GeminiDB Cassandra Instance Tags**. |
| Free backup space | GeminiDB Cassandra instances provide free backup storage. | For more information, see **Backup Storage**. |
| Retention period | The default value is 7 days. The value ranges from 1 to 3660 days. | For more information, see **Automated Backup Policy**. |

## Naming Rules

**Table 1-9** Naming rules

| Item | Description |
| --- | --- |
| Instance name | ● Can contain 4 to 64 characters.<br>● Must start with a letter. Only letters (case-sensitive), digits, hyphens (-), and underscores (_) are allowed. |
| Backup name | ● Can contain 4 to 64 characters.<br>● Must start with a letter. Only letters (case sensitive), digits, hyphens (-), and underscores (_) are allowed. |
| Parameter template name | ● Can contain 1 to 64 characters.<br>● Only letters (case sensitive), digits, hyphens (-), underscores (_), and periods (.) are allowed. |

## Security

**Table 1-10** Security

| Item | Description |
| --- | --- |
| Password of database administrator **rwuser** | ● Can contain 8 to 32 characters.<br>● Can contain at least two types of the following characters: uppercase letters, lowercase letters, digits, and special characters ~!@#%^*-_=+? For more information, see **Resetting the Administrator Password**.<br>● Keep your password secure. The system cannot retrieve it if it is lost. |
| Database port | Database port number.<br>When creating a GeminiDB Cassandra instance, you cannot specify a port number. The default port number is **8635**.<br>The port number can be changed after the instance is created.<br>The database port ranges from 2100 to 9500 except 2180, 2181, 2887, 3887, 7000, 7001, 7199, 8000, 8018, 8079, 8091, 8092, 8479, 8484, 8636, and 8999. |
| VPC | After a GeminiDB Cassandra instance is created, the VPC where the instance is deployed cannot be changed. |

| Item | Description |
| --- | --- |
| Intranet security group | A security group controls access between GeminiDB Cassandra API and other services. Ensure that the security group you selected allows your client to access the instance.<br>If no security group is available, the system creates one for you. |

## Instance Operations

**Table 1-11** Instance operations

| Function | Constraint |
| --- | --- |
| Database access | • If remote access is not enabled, GeminiDB Cassandra instances and their associated ECSs must be in the same VPC subnet.<br>• The security group must allow access from the associated ECS.<br>By default, a GeminiDB Cassandra instance cannot be accessed through an ECS in a different security group. You need to add an inbound rule to the security group.<br>• The default port number of a GeminiDB Cassandra instance is 8635.<br>• The database port can be changed only after the instance is created. |
| Instance deployment | The servers where instances are deployed are not directly visible to you. You can only access the instances through IP addresses and database ports. |
| Restarting a GeminiDB Cassandra instance | • GeminiDB Cassandra instances cannot be rebooted through commands. They must be rebooted on the console.<br>• Restarting an instance will interrupt services, so off-peak hours are the best time. Ensure that your application can be reconnected. |
| Viewing GeminiDB Cassandra instance backups | GeminiDB Cassandra instance backups are stored in OBS buckets and are invisible to you. |

| Function | Constraint |
| --- | --- |
| Changing the CPU or memory of a GeminiDB Cassandra instance | ● Second-level intermittent disconnection occurs once when the specifications are changed on a single node. Therefore, the entire instance is intermittently disconnected several times. Ensure that the client can be reconnected. You are advised to change the specifications during off-peak hours.<br>● 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 instance overload. |
| Data restoration | To prevent data loss, you are advised to back up key data before data restoration. |
| Storage | If the storage space of an instance is full, data cannot be written to databases. You are advised to periodically check the storage space.<br>GeminiDB Cassandra instance storage can be automatically scaled up in case of a sudden surge in data volumes. Enable autoscaling by following **Automatically Scaling Up Storage Space**. |
| Recycle bin | ● You can move unsubscribed yearly/monthly instances and deleted pay-per-use instances to the recycle bin. You can restore an instance that was deleted up to 7 days ago from the recycle bin.<br>● The recycling policy 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.<br>● Currently, you can put a maximum of 100 instances into the recycle bin.<br>● If you delete an instance running out of storage, it will not be moved to the recycle bin. |

For details about other development and O&M specifications that can effectively evaluate and improve service system stability, see **Database Rules**.

# 2 Billing

## 2.1 Billing Overview

In this document, you will learn about how instances are billed, how you can renew subscriptions and manage costs, and what happens if your account goes into arrears.

- **Billing Modes**

  There are yearly/monthly and pay-per-use billing modes. Each one has different advantages and disadvantages.

  – Yearly/Monthly: You pay upfront for the amount of time you expect to use the instance for. You will need to make sure you have a top-up account with a sufficient balance or have a valid payment method configured first.

  – Pay-per-use: You can start using the GeminiDB instance first and then pay as you go.

  For details about the two billing modes, see **Overview**.

  You can also change the billing mode later if it no longer meets your needs. For details, see **Overview**.

- **Billing Items**

  You will be billed for instance specifications, storage space, backup space, and EIP bandwidths. For details about the billing factors and formulas for each billed item, see **Billing Items**.

  For more information about billing samples and the billing for each item, see **Billing Examples**.

- **Renewing Subscriptions**

  If you want to continue using an instance after it expires, you need to renew the instance subscription within the specified period. Otherwise, resources, such as compute and storage, will be automatically released, and data may be lost.

  You can renew your subscription manually or automatically. For details, see **Overview**.

- **Viewing Bills**

You can choose **Billing & Costs** > **Bills** to check the instance transactions and bills. For details, see **Bills**.

- **Arrears**

  If there is not a sufficient account balance to pay for your bill and there is no other payment method configured, your account will go into arrears. If you want to continue using your cloud services, you will need to top up your account in a timely manner. For details, see **Arrears**.

- **Stopping Billing**

  If you no longer need to use your GeminiDB Cassandra instance, you can unsubscribe from or delete it to stop the billing. For details, see **Billing Termination**.

- **Managing Costs**

  GeminiDB Cassandra costs include resource costs and O&M costs. You can allocate, analyze, and optimize GeminiDB costs to save more money. For details, see **Cost Management**.

# 2.2 Billing Modes

## 2.2.1 Overview

There are yearly/monthly and pay-per-use billing modes. Each one has different advantages and disadvantages.

- Yearly/Monthly is a prepaid billing mode. You pay in advance for a subscription term, and in exchange, you get a discounted rate. The longer the subscription term, the bigger the discount. Yearly/Monthly billing is a good option for long-term, stable services.
- Pay-per-use is a postpaid billing mode. You pay as you go and just pay for what you use. The instance usage is calculated by the second but billed every hour. Pay-per-use billing is a good option for scenarios where there are sudden traffic bursts, such as e-commerce promotions.

**Table 2-1** lists differences between the two billing modes.

**Table 2-1** Differences between billing modes

| Billing Mode | Yearly/Monthly | Pay-per-use |
| --- | --- | --- |
| Payment | Prepaid<br>Billed by the subscription term you purchase | Postpaid<br>Billed for what you use |
| Billing Method | Billed by the subscription term you purchase | Calculated by the second but billed every hour |
| Billed Items | Instance specifications (vCPUs and memory), storage space, backup space, and EIPs | Instance specifications (vCPUs and memory), storage space, backup space, and EIPs |

| | | |
|---|---|---|
| **Changing the Billing Mode** | Yearly/Monthly can be changed to pay-per-use. The change takes effect only after the yearly/monthly subscription expires. For details, see **Changing a Yearly/Monthly Instance to Pay-per-Use**. | Pay-per-use can be changed to yearly/monthly. For details, see **Changing a Pay-per-Use Instance to Yearly/Monthly**. |
| **Changing the Specifications** | Supported | Supported |
| **Application Scenarios** | Recommended for resources expected to be in use long term. A cost-effective option for scenarios where the resource usage duration is predictable. | Recommended when the resource demands are likely to fluctuate and you want more flexibility. |

## 2.2.2 Yearly/Monthly Billing

If you expect to use resources for a longer period, you can save money by selecting yearly/monthly billing. This section describes billing rules of yearly/monthly GeminiDB Cassandra resources.

### Application Scenarios

If you want to ensure resource stability over a certain period of time, yearly/monthly billing is a good choice for the following types of workloads:

- Long-term workloads with stable resource requirements, such as official websites, online malls, and blogs.
- Long-term projects, such as scientific research projects and large-scale events.
- Workloads with predictable traffic bursts, for example, e-commerce promotions or festivals.
- Workloads with high data security requirements.

### Billing Items

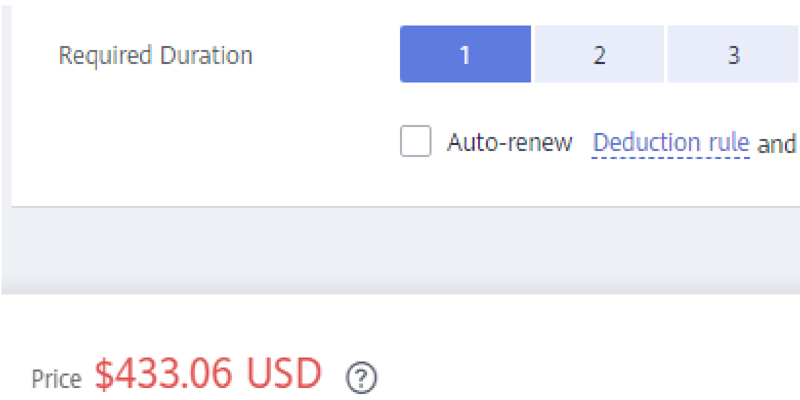
You are billed for the following items on a yearly/monthly basis.

**Table 2-2** Items billed on a yearly/monthly basis

| Billing Item | Description |
|---|---|
| Instance specifications | Instance specifications, including vCPUs and memory. |
| Storage space | If the actual storage usage exceeds your purchased storage, you will be billed for additional storage on a pay-per-use basis. |
| Backup space | GeminiDB Cassandra provides backup storage up to 100% of your provisioned database storage at no additional charge.<br>After the free backup storage is used up, additional usage will be priced by the hour based on the backup storage pricing details. If it has been used less than one hour, you will be billed based on the actual duration. |
| (Optional) EIP bandwidth | GeminiDB Cassandra instances are accessible from public networks, and you are billed for the generated public network traffic, but not for private network traffic. |

Assume that you want to buy a three-node GeminiDB Cassandra instance with 2 vCPUs, 8 GB of memory, and 100 GB of storage. At the bottom of the instance buying page, price details (excluding the backup space fee) will be displayed.

**Figure 2-1** Example price




You are billed for:

- Selected specifications for your instance
- Storage space

NOTE

The backup space fee is not included. For details about the backup price, see **Product Pricing Details**.

Backup Storage Space

| DB Instance Type | Hourly | Currency |
| --- | --- | --- |
| Cluster | 0.00004 | Price per GB |

## Billing Cycle

A yearly/monthly GeminiDB Cassandra instance is billed for the purchased duration (UTC+8). The billing starts when you activated or renewed the subscription, and ends at 23:59:59 of the expiry date.

For example, if you bought a GeminiDB Cassandra instance for one month at 15:50:04 on March 8, 2023, the billing cycle is from 15:50:04 on March 8, 2023 to 23:59:59 on April 8, 2023.

## Billing Examples

Assume that you bought a three-node GeminiDB Cassandra instance with 2 vCPUs, 8 GB of memory, 100 GB of storage, 110 GB (100 GB for free) of backup storage for one month at 15:50:04 on March 8, 2023 and renewed the subscription for one more month before it expired. The billing items include instance specifications (vCPUs, memory, and nodes), storage, backup storage, and EIP bandwidth.

- The first billing cycle is from 15:50:04 on March 8, 2023 to 23:59:59 on April 8, 2023.
- The second billing cycle is from 23:59:59 on April 8, 2023 to 23:59:59 on May 8, 2023.
  - From 23:59:59 on April 8, 2023 to 23:59:59 on May 1, 2023, 50 GB of free backup storage was used.
  - From 23:59:59 on May 1, 2023 to 23:59:59 on May 8, 2023, another 10 GB of backup storage was billed for 168 hours.

You need to pay in advance for each billing cycle. Each resource is billed separately.

**Table 2-3** Billing formulas

| Resource | Formula | Unit Price |
| --- | --- | --- |
| Instance specifications (including vCPUs and memory) | Unit price of the instance specifications x Required duration x Number of nodes | For details about the unit price, see **Cluster CPU/Memory** on **Product Pricing Details** |

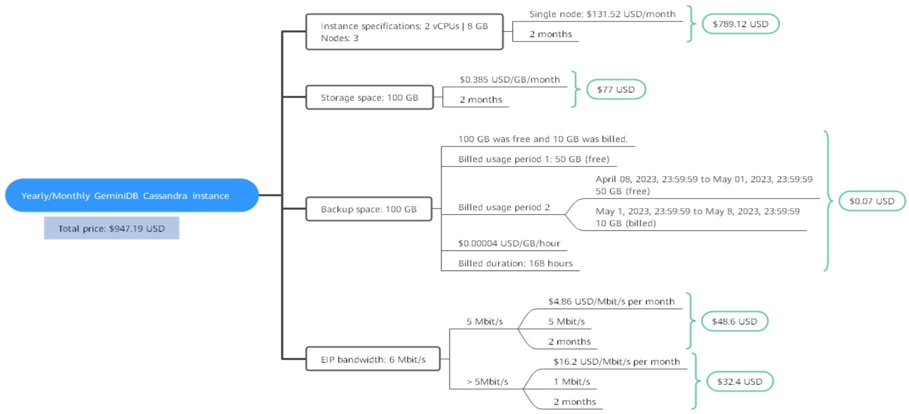
| Resource | Formula | Unit Price |
|---|---|---|
| Storage | Storage space unit price x Required duration x Storage space (GB) | See the estimated price of a cluster instance with specified storage in **GeminiDB Price Calculator**. |
| Backup storage | Backup storage unit price x Required duration x (Backup storage – Storage) (GB)<br>**NOTE**<br>The billing duration indicates how long the storage exceeding a free quota was used. | See the estimated price of an instance with specified backup storage in **GeminiDB Price Calculator**. |
| EIP bandwidth | Billed by fixed bandwidth | For details, see **Product Pricing Details**. |

**Figure 2-2** shows how the total price is calculated.

**NOTE**

The prices in the following figure are for reference only. For the actual prices, see **GeminiDB Price Calculator**.

**Figure 2-2** Total price for a yearly/monthly GeminiDB Cassandra instance




## Price Change After Specification Change

If the specifications of a yearly/monthly GeminiDB Cassandra instance no longer meet your needs, you can change the specifications on the console. The system will recalculate the price and either bill or refund you the difference.

- If you increase instance specifications, you need to pay the difference in price.
- If you decrease instance specifications, Huawei Cloud will refund you the difference.

Decreasing instance specifications will affect instance performance. You are not advised to do so. Assume that you bought a yearly/monthly three-node GeminiDB Cassandra instance with 2 vCPUs and 8 GB of memory for one month on April 8, 2023 and increased its specifications to 4 vCPUs and 16 GB of memory on April 18, 2023. The old specifications cost USD587.06/month and the new specifications USD981.62/month. The calculation formula is as follows:

**Price difference = Price of new specifications x Remaining period – Price of old specifications x Remaining period**

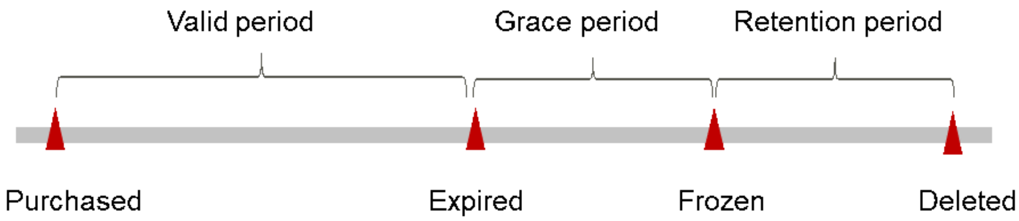
The remaining period in the formula is the remaining days of each calendar month divided by the maximum number of days in each calendar month. In this example, the remaining period is 0.6581 (12/30 + 8/31). The fee for increasing specifications is USD259.66 (981.62 x 0.6581 – 587.06 x 0.6581).

For more details, see **Pricing of a Changed Specification**.

## Impact of Expiration

**Figure 2-3** shows the statuses a yearly/monthly GeminiDB Cassandra instance can go through throughout its lifecycle. After an instance is purchased, it enters the valid period and runs normally during this period. If the instance is not renewed after it expires, before being deleted, it first enters a grace period and then a retention period.

**Figure 2-3** Lifecycle of a yearly/monthly GeminiDB Cassandra instance




### Expiration Reminder

The system will send you a reminder (by email, SMS, or in-app message) before a yearly/monthly GeminiDB Cassandra instance expires to the Huawei Cloud account creator.

- Expiration notifications will be sent 30 days, 15 days, 7 days, 3 days, and 1 day before yearly resources expire.
- Expiration notifications will be sent 15 days, 7 days, 3 days, and 1 day before monthly resources expire.

### Impact of Expiration

If your yearly/monthly GeminiDB Cassandra instance is not renewed after it expires, it changes to the **Expired** state and enters a grace period. During the grace period, you can access the GeminiDB Cassandra instance but cannot:

- Change instance specifications.
- Change the billing mode from yearly/monthly to pay-per-use.

- Unsubscribe from it.

If the yearly/monthly GeminiDB Cassandra instance is not renewed after the grace period ends, its status turns to **Frozen** and it enters a retention period. You cannot perform any operations on the GeminiDB Cassandra instance while it is in the retention period.

If the yearly/monthly GeminiDB Cassandra instance is not renewed by the time the retention period ends, it will be released and data cannot be restored.

NOTE

- For details about renewals, see **Overview**.

## 2.2.3 Pay-per-use Billing

Pay-per-use billing means you pay nothing up front and are not tied into any contract or commitment. This section describes billing rules of pay-per-use GeminiDB Cassandra instances.

### Application Scenarios

Pay-per-use billing is good for short-term, bursty, or unpredictable workloads that cannot tolerate any interruptions, such as applications for e-commerce flash sales, temporary testing, and scientific computing.

### Billing Items

You are billed for the following items on a pay-per-use basis.

**Table 2-4** Items billed on a pay-per-use basis

| Billing Item | Description |
| --- | --- |
| Instance specifications | vCPUs and memory |
| Storage | Instance storage space, which is billed hourly on a pay-per-use basis. |
| Backup storage | GeminiDB Cassandra API provides free backup storage equal to the amount of storage you purchased.<br>After the free backup storage is used up, additional usage will be priced by the hour based on the backup storage pricing details. If it has been used less than one hour, you will be billed based on the actual duration. |
| (Optional) EIP bandwidth | GeminiDB Cassandra instances are accessible from public networks, and you are billed for the generated public network traffic, but not for private network traffic. |

If you want to purchase a 3-node (specifications of each node: 2 vCPUs | 8 GB) GeminiDB Cassandra instance with 500 GB of storage space. At the bottom of the page for buying an instance, price details (excluding the backup storage fee) will be displayed.

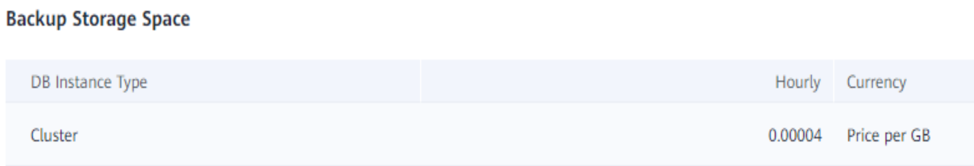
**Figure 2-4** Example price


You are billed for:

- Instance specifications (including vCPUs and memory)
- Selected storage space

NOTE

The backup space fee is not included. For details about the backup price, see **Product Pricing Details**.



Backup Storage Space

| DB Instance Type | Hourly | Currency |
| --- | --- | --- |
| Cluster | 0.00004 | Price per GB |

## Billing Cycle

Pricing is listed on a per-hour basis (GMT+08:00), and bills are calculated down to the second. The billing starts when an instance is created and ends when it is deleted.

NOTE

It takes a certain time to create an instance. The billing starts from the time when the instance is successfully created. You can view the two time points on the **Basic Information** page. You can view the time when the instance is created beside the **Created** field.

For example, if you buy a pay-per-use GeminiDB Cassandra instance at 8:45:30 and delete it at 8:55:30, you are billed for the 600 seconds from 8:45:30 to 8:55:30. The billing items include compute resources (vCPUs and nodes), storage, and backup storage.

## Billing Examples

Assume that you bought a pay-per-use 3-node instance with 2 vCPUs, 8 GB of memory, 100 GB of storage, and 110 GB of backup storage (100 GB for free) at 09:59:30 on April 18 and deleted the instance at 10:45:46 on April 18, 2023. The billing items include compute resources (vCPUs and nodes) and storage.

- Usage of 30 seconds from 9:59:30 to 10:00:00
- Usage of 2,746 seconds from 10:00:00 to 10:45:46
  - The free backup storage is used from 10:00:00 to 10:45:00.

– 10 GB of backup storage is billed for 46 seconds from 10:45:00 to 10:45:46.

The price displayed in the pricing details is per hour, so you need to divide it by 3,600 to obtain the price for each second and then multiply the per-second price by the total number of seconds. GeminiDB Cassandra instances are billed individually.

**Table 2-5** Billing formulas

| Resource | Formula | Unit Price |
| --- | --- | --- |
| Compute resources (including vCPUs and nodes) | Unit price of instance specifications x Required duration | See the estimated price of a cluster instance in **GeminiDB Price Calculator**. |
| Storage | Storage unit price x Required duration | See the estimated price of a cluster instance with specified storage in **GeminiDB Price Calculator**. |
| Backup storage | Backup storage unit price x Required duration x (Backup storage – Storage) (GB)<br>**NOTE**<br>The billing duration indicates how long the storage exceeding a free quota was used. | See the estimated price of a cluster instance with specified backup storage in **GeminiDB Price Calculator**. |
| Public network traffic | Tiered billing by fixed bandwidth<br>● 0 Mbit/s to 5 Mbit/s (included): billed at a fixed unit price per Mbit/s<br>● Greater than 5 Mbit/s: billed at a different price per Mbit/s | For details, see the estimated bandwidth price in **ECS Price Calculator** or **EIP Price Calculator**. |

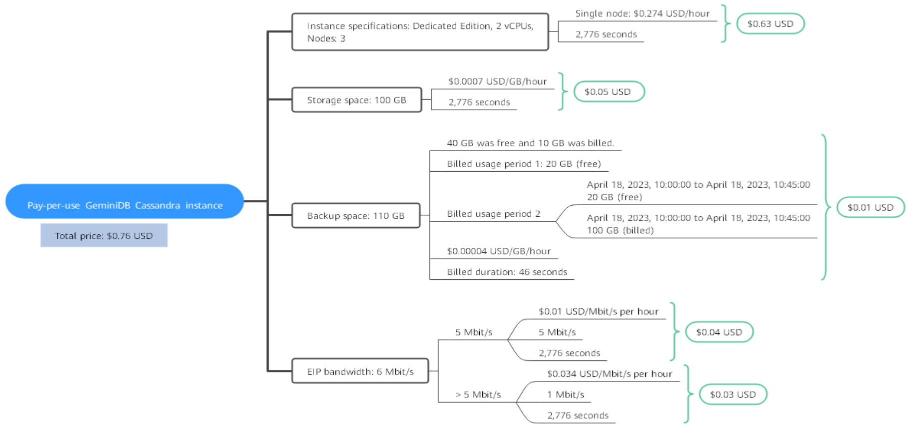
**Figure 2-5** shows how the total price is calculated.

**NOTE**

The prices in the following figure are for reference only. For the actual prices, see **GeminiDB Price Calculator**.

If the price is not an integer, it is rounded off to the nearest two decimal places. If the rounded price is less than USD0.01, USD0.01 will be displayed.

**Figure 2-5** Total price for a pay-per-use GeminiDB Cassandra instance




## Impact on Billing After Specification Changes

If you change the specifications of a pay-per-use instance, the original order will become invalid and a new order will be placed. You will be billed based on the new specifications.

If you change instance specifications within a given hour, multiple records will be generated. Different records record the billing for different specifications.

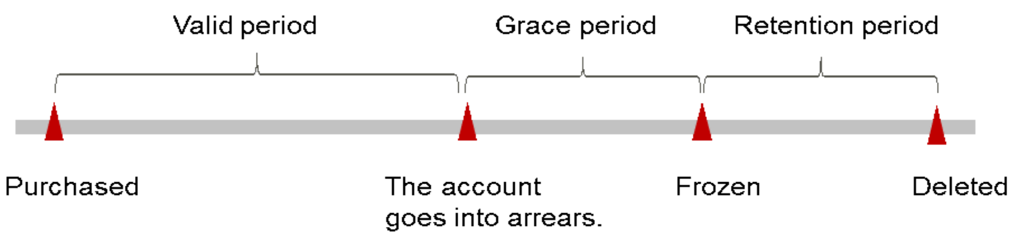
For example, if you buy a pay-per-use instance with 2 vCPUs and 8 GB of memory at 9:00:00 and increase the specifications to 4 vCPUs and 16 GB of memory at 9:30:00, two billing records will be generated between 9:00:00 and 10:00:00:

- 2 vCPUs and 8 GB of memory from 9:00:00 to 9:30:00
- 4 vCPUs and 16 GB of memory from 9:30:00 to 10:00:00

## Impact of Arrears

Figure 2-6 shows the statuses of a pay-per-use GeminiDB Cassandra instance throughout its lifecycle. After a GeminiDB Cassandra instance is purchased, it enters the valid period and runs normally during this period. If your account goes into arrears, the instance enters a grace period and then a retention period.

**Figure 2-6** Lifecycle of a pay-per-use GeminiDB Cassandra instance




### Arrears reminder

The system will bill you for pay-per-use resources after each billing cycle ends. If your account goes into arrears, the system will send an email, SMS message, or in-app message to the one who created the Huawei Cloud account.

**Impact**

If your account is in arrears due to automated deduction for pay-per-use GeminiDB Cassandra instances, the instances are not immediately stopped but given a grace period. After you top up your account, Huawei Cloud will bill you for expenditures generated during the grace period. You can view the charges on the **Billing Center** > **Overview** page.

If you do not pay the arrears within the grace period, your instance enters the retention period and its status changes to **Frozen**. You cannot perform any operations on the instance in the retention period.

If you do not pay the arrears within the retention period, your instance will be released, and data will be lost.

NOTE

- During the retention period, you cannot access or use your instance but the data stored in it can be retained. The retention period for Huawei Cloud International website is 15 days.
- During the grace period, you can access and use only some resources of your instance. The grace period for Huawei Cloud International website is 15 days.
- For details about top-up, see **Topping Up an Account**.

# 2.3 Billing Items

## Billing

You will be billed for instance specifications, storage space, backup space, and EIP bandwidths. For details, see **Table 2-6**.

NOTE

The billed items marked with asterisks (*) are mandatory.

**Table 2-6** Billing Items of a GeminiDB Cassandra Instance

| Billing Item | Description | Billing Mode | Formula |
| --- | --- | --- | --- |
| * Specifications | Billed by instance specifications, including vCPUs and memory. Computing and storage capabilities vary by the number of vCPUs and memory size. | Yearly/Monthly and pay-per-use | **Unit price x Required duration**<br>For details about the unit price, see **Cluster CPU/Memory** on **Product Pricing Details**. |

| Billing Item | Description | Billing Mode | Formula |
| --- | --- | --- | --- |
| * Storage space | Billed based on unified standards. | Yearly/ Monthly and pay-per-use | **Unit price x Storage space x Required duration**<br>For details about the unit price, see **Storage Space** on **Product Pricing Details**. |
| Backup space | Billed based on unified standards. | Pay-per-use | **Unit price x Billed backup space x Required duration**<br>For details about the unit price, see **Backup Storage Space** on **Product Pricing Details**.<br>**NOTE**<br>The billing duration indicates how long the storage exceeding a free quota was used. |
| (Optional) Cross-region backup | Billed based on unified standards. | Pay-per-use | **Storage space unit price (CNY0.0009/GB/hour) x Storage capacity x Required duration** |
| | Billed based on unified standards. | Billed by storage capacity | **Unit price of backup traffic across regions (CNY0.5/GB) x Storage capacity** |

| Billing Item | Description | Billing Mode | Formula |
|---|---|---|---|
| Public network traffic | An EIP is required if an instance needs to access the Internet.<br>Billing factors: bandwidth, traffic, and IP reservation<br>● EIP for a yearly/monthly GeminiDB Cassandra instance: billed by bandwidth.<br>● EIP for a pay-per-use instance: billed by bandwidth, traffic, or shared bandwidth. You are also billed for IP reservation. | Yearly/ Monthly and pay-per-use<br>You can purchase a bandwidth add-on package or a shared traffic package. | **Tiered pricing based on fixed bandwidth**.<br>● 0 Mbit/s to 5 Mbit/s (included): billed at a fixed unit price per Mbit/s.<br>● Greater than 5 Mbit/s: billed at a different price per Mbit/s.<br>For details about the unit price, see **Bandwidth Price** on **Product Pricing Details** or **Product Pricing Details**. |

## Billing Examples

Assume that you bought a three-node GeminiDB Cassandra instance with 2 vCPUs, 8 GB of memory, 100 GB of storage, 110 GB (100 GB for free) of backup storage for one month at 15:50:04 on March 8, 2023 and renewed the subscription for one more month before it expired. The billing items include instance specifications (vCPUs, memory, and nodes), storage, backup storage, and EIP bandwidth.

- The first billing cycle is from 15:50:04 on March 8, 2023 to 23:59:59 on April 8, 2023.
- The second billing cycle is from 23:59:59 on April 8, 2023 to 23:59:59 on May 8, 2023.
  - From 23:59:59 on April 8, 2023 to 23:59:59 on May 1, 2023, 50 GB of free backup storage was used.
  - From 23:59:59 on May 1, 2023 to 23:59:59 on May 8, 2023, another 10 GB of backup storage was billed for 168 hours.

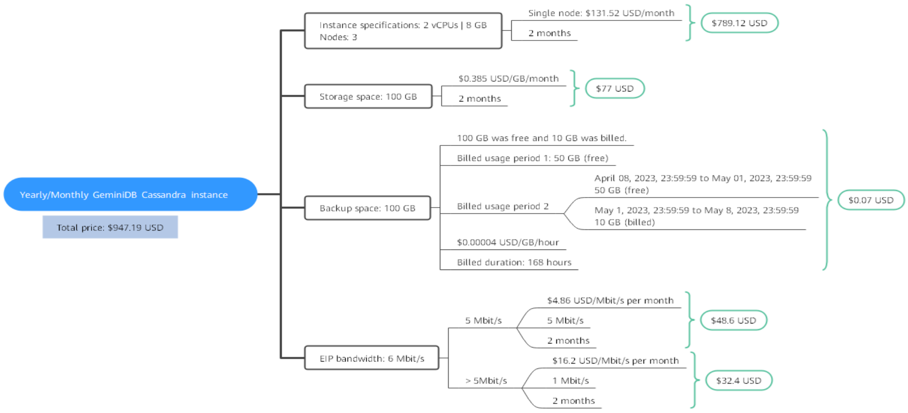
**Figure 2-7** shows how the total price is calculated.

NOTE

The prices in the following figure are for reference only. For the actual prices, see **GeminiDB Price Calculator**.

**Figure 2-7** Total price for a yearly/monthly GeminiDB Cassandra instance




For more billing examples of a pay-per-use GeminiDB Cassandra instance, see **Billing Examples**.

# 2.4 Billing Examples

## Billing Scenario

A user purchased a pay-per-use GeminiDB Cassandra instance at 15:30:00 on March 18, 2023. The instance configuration is as follows:

- Specifications: 2 vCPUs | 8 GB
- Nodes: 3
- EIP bandwidth: 6 Mbit/s

After a period, the user found that the current instance specifications could not meet service requirements and increased the specifications to 4 vCPUs and 16 GB at 9:00:00 on March 20, 2023. The user changed the billing mode to yearly/monthly at 10:30:00 on March 20, 2023 and paid the instance in advance for one month. So how much will the user be billed for this instance in March and April?

## Billing Analysis

The total price of this GeminiDB Cassandra instance involves both pay-per-use and yearly/monthly usage:

- 15:30:00 on March 18, 2023 to 10:30:00 on March 20, 2023: pay-per-use
  - 15:30:00 on March 18, 2023 to 9:00:00 on March 20, 2023
    - Instance specifications: 2 vCPUs and 8 GB of memory
    - Nodes: 3
    - Used storage space: 100 GB
    - Used backup space: 100 GB
    - EIP bandwidth: 6 Mbit/s

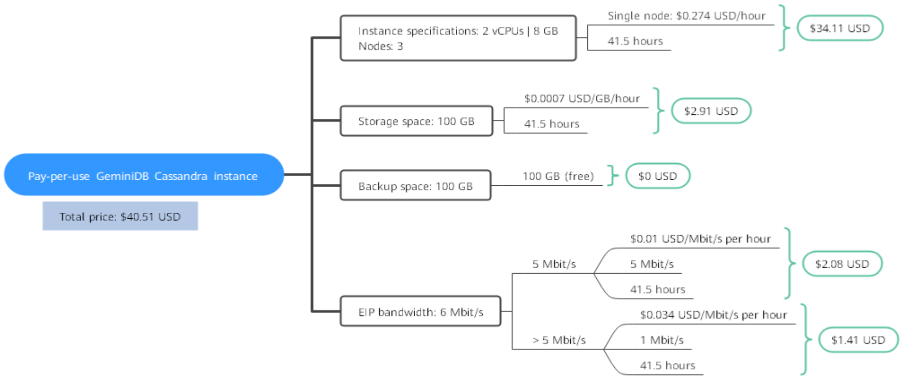
  - 9:00:00 on March 20, 2023 to 10:30:00 on March 20, 2023
    - Instance specifications: 4 vCPUs and 16 GB of memory
    - Nodes: 3
    - Storage: 200 GB
    - Backup storage: 210 GB (pay-per-use from 10:00:00 to 10:30:00 on March 20, 2023)
    - EIP bandwidth: 6 Mbit/s
- 10:30:00 on Mar 20, 2023 to 23:59:59 on Apr 20, 2023: yearly/monthly
  - Instance specifications: 4 vCPUs and 16 GB of memory
  - Nodes: 3
  - Storage: 200 GB
  - Backup storage: 300 GB (pay-per-use from 23:59:59 on April 10, 2023 to 23:59:59 on April 20, 2023)
  - EIP bandwidth: 6 Mbit/s
  - Billing duration: one month

**NOTE**

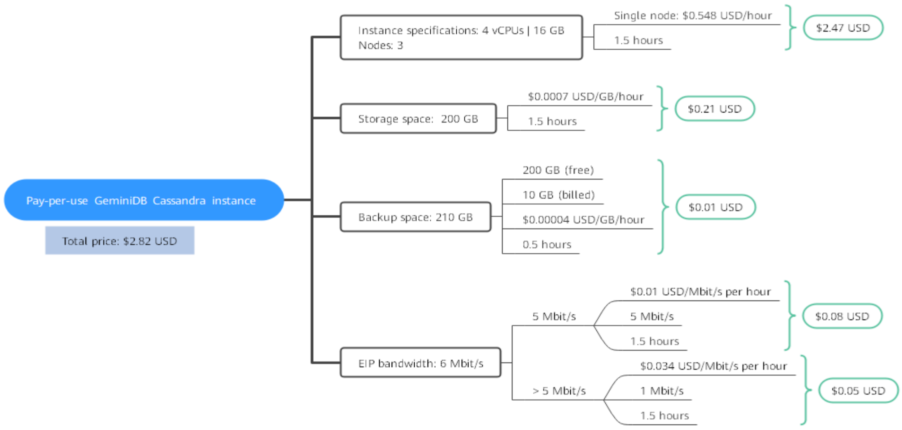
Unit prices in this example are used for reference only, and the calculated prices are only estimates. The actual unit price and cost may vary. For details, see the data released on the Huawei Cloud official website.

### Pay-per-use

From 15:30:00 on March 18, 2023 to 09:00:00 on March 20, 2023, an instance with 2 vCPUs and 8 GB of memory was used for 41.5 hours, so the price is calculated as follows.

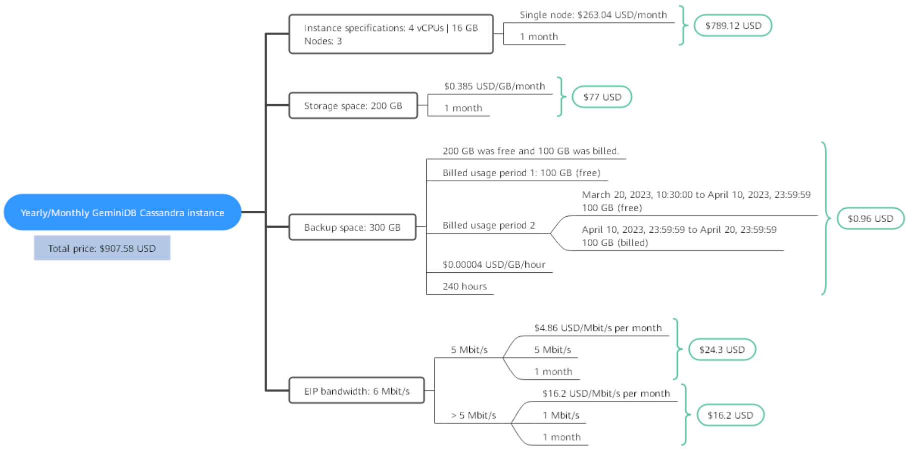



From 9:00:00 to 10:30:00 on March 20, 2023, an instance with 4 vCPUs and 16 GB of memory was used for 1.5 hours, so the price is calculated as follows.




### Yearly/Monthly

From 10:30:00 on March 20, 2023 to 23:59:59 on April 20, 2023, the yearly/monthly instance was used for one month. The price is calculated as follows.




From March to April, the total price of this instance is USD950.91 (40.51 + 2.82 + 907.58).

# 2.5 Billing Mode Changes

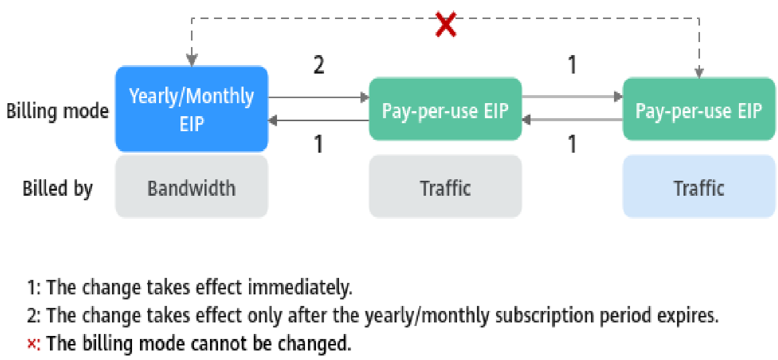
## 2.5.1 Overview

After purchasing a GeminiDB Cassandra instance, you can change the billing mode if it no longer meets your needs. **Table 2-7** lists changeable billing items of the GeminiDB Cassandra instance.

**Table 2-7** Changeable billing items of GeminiDB Cassandra instances

| Billing Item | Change Description | Reference |
| --- | --- | --- |
| Instance specifications (vCPUs and nodes) | Changing the billing mode of a GeminiDB Cassandra instance includes the changes to compute resources (vCPUs and nodes).<br>• Change from pay-per-use to yearly/monthly to enjoy lower prices.<br>• Change from yearly/monthly to pay-per-use to use the GeminiDB Cassandra instance more flexibly.<br>**NOTE**<br>Such a change takes effect only after the yearly/monthly subscription ends. | • **Changing a Pay-per-Use Instance to Yearly/Monthly**<br>• **Changing a Yearly/Monthly Instance to Pay-per-Use** |
| EIP | • A yearly/monthly EIP can be changed to a pay-per-use EIP billed by bandwidth after the yearly/monthly subscription ends.<br>• A pay-per-use EIP billed by bandwidth can be changed to a yearly/monthly EIP.<br>• Pay-per-use EIPs billed by bandwidth can be changed to pay-per-use EIPs billed by traffic, and pay-per-use EIPs billed by traffic can be changed to pay-per-use EIPs billed by bandwidth.<br>For details, see **Figure 2-8**. | • **Changing a Pay-per-Use Instance to Yearly/Monthly**<br>• **Changing a Yearly/Monthly Instance to Pay-per-Use** |

**Figure 2-8** EIP billing mode change



## 2.5.2 Changing a Pay-per-Use Instance to Yearly/Monthly

If you have a pay-per-use GeminiDB Cassandra instance that you expect to use for a long time, you can change it to yearly/monthly billing to reduce costs. Doing so will create an order. After you pay for the order, yearly/monthly billing will be applied immediately.

Suppose you bought a pay-per-use GeminiDB Cassandra instance at 15:29:16 on April 18, 2023 and changed it to yearly/monthly billing at 16:30:30 on the same day. After you paid for the order, yearly/monthly billing was applied immediately. On the **Billing Center** > **Billing** page, three line items were generated.

- Pay-per-use expenditures for 15:29:16 to 16:00:00 on April 18, 2023
- Pay-per-use expenditures for 16:00:00 to 16:30:30 on April 18, 2023
- A yearly/monthly expenditure generated at 16:30:30 on April 18, 2023

### Constraints

Resources such as EIPs that are used by an instance may not support the change with this instance. For details about their billing mode change rules and handling methods, see **Table 2-8**.

**Table 2-8** EIP billing mode change rules

| Resource | Billing Mode | Billed By | Bandwidth Type | Changed to Yearly/ Monthly Billing with the GeminiDB Cassandra Instance | Handling Measure |
| --- | --- | --- | --- | --- | --- |
| EIP | Pay-per-use | Bandwidth | Dedicated | Supported | Change the EIP to yearly/ monthly billing on the EIP console.<br>For details, see **Changing EIP Billing Mode**. |

| Resource | Billing Mode | Billed By | Bandwidth Type | Changed to Yearly/Monthly Billing with the GeminiDB Cassandra Instance | Handling Measure |
|---|---|---|---|---|---|
| EIP | Pay-per-use | Traffic | Dedicated | Not supported | An EIP that is billed by traffic on a pay-per-use basis cannot be directly changed to be billed on a yearly/monthly basis. To change this:<br>1. Change the EIP to be billed by bandwidth on a pay-per-use basis.<br>2. Change the EIP to be billed on a yearly/monthly basis.<br>For details, see **Changing EIP Billing Mode**. |

## Prerequisites

- The billing mode of the instance is pay-per-use.
- The instance status is **Available**.

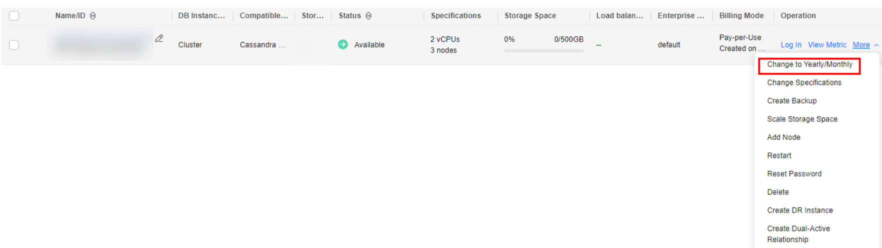
## Procedure

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

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** On the **Instances** page, locate the target instance and click **Change to Yearly/Monthly** in the **Operation** column.

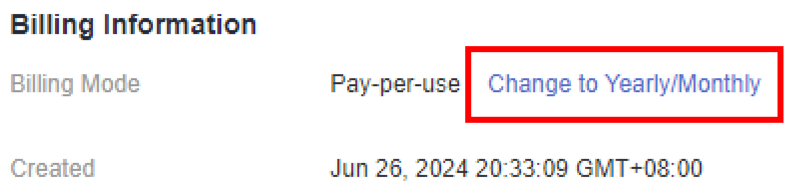
**Figure 2-9** Change to Yearly/Monthly



Alternatively, click the instance name to go to the **Basic Information** page. In the **Billing Information** area, click **Change to Yearly/Monthly** in the **Billing Mode** field.

**Figure 2-10** Change to Yearly/Monthly




NOTE

The billing mode of multiple instances can be changed in batches. Perform the following steps:

1. Select the instances whose billing mode you want to change.
2. Click **Change to Yearly/Monthly** above the instance list.

**Step 4** On the displayed page, specify a subscription duration in month. The minimum duration is one month.

If you do not need to modify your settings, click **Pay** to go to the payment page.

**Step 5** Select a payment method and click **Confirm**.

**Step 6** 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**

## 2.5.3 Changing a Yearly/Monthly Instance to Pay-per-Use

After creating a yearly/monthly GeminiDB Cassandra instance, you can change it to pay-per-use for more flexibility, and you can recoup part of what you paid for the subscription.

Suppose you bought a yearly/monthly GeminiDB Cassandra instance at 15:29:16 on April 18, 2023 and changed it to pay-per-use billing at 16:30:00 on May 18, 2023. On the **Billing Center** > **Billing** page, bill information is generated as follows:

- Yearly/Monthly expenditures for 15:29:16 on April 18 to 23:59:59 on May 18, 2023
- Pay-per-use expenditures for 23:59:59 on May 18, 2023 to the end time of pay-per-use billing. A bill was generated every hour.

NOTE

The pay-per-use billing mode will take effect only after the yearly/monthly subscription has expired. Auto-renewal will not be in effect.

## Constraints

Resources such as EIPs that are used by an instance may not support the change with this instance. For details about their billing mode change rules and handling methods, see **Table 2-9**.

**Table 2-9** EIP billing mode change rules

| Resource | Billing Mode | Billed By | Bandwidth Type | Change to Pay-per-Use Billing with the GeminiDB Cassandra Instance | Handling Measure |
| --- | --- | --- | --- | --- | --- |
| EIP | Yearly/Monthly | Bandwidth | Dedicated | Not supported | Change the EIP to yearly/monthly billing on the EIP console.<br>For details, see **Changing EIP Billing Mode**. |
| EIP | Yearly/Monthly | Traffic | Dedicated | Not supported | An EIP billed on a yearly/monthly basis cannot be directly changed to be billed by traffic on a pay-per-use basis. To change this:<br>1. Change the EIP to be billed by bandwidth on a pay-per-use basis.<br>2. Change the EIP to be billed by traffic on a pay-per-use basis.<br>For details, see **Changing EIP Billing Mode**. |

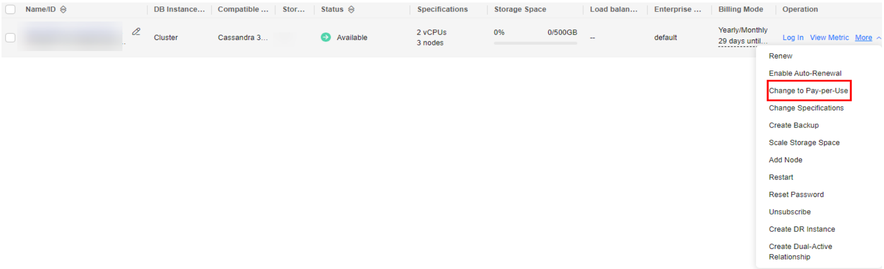
## Procedure

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

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** On the **Instances** page, locate the target instance and choose **More** > **Change to Pay-per-Use** in the **Operation** column.

**Figure 2-11** Change to Pay-per-Use




NOTE

The billing mode of multiple pay-per-use instances can be changed in batches. Perform the following steps:

1. Select the instances whose billing mode you want to change.
2. Click **More** > **Change to Pay-per-Use** in the **Operation** column

**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. 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 subscription 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**

# 2.6 Renewing Subscriptions

## 2.6.1 Overview

### When to Renew Subscriptions

If a yearly/monthly instance is about to expire but you want to continue using it, you need to renew the instance subscription within a specified period, or resources, such as vCPUs and memory, will be automatically released, and data will be lost and cannot be restored.

Only yearly/monthly instance subscriptions can be renewed. If you use pay-per-use instances, just ensure that your account has a valid payment method configured or a top-up account with a sufficient balance.

If you renew the instance before it expires, resources will be retained and you can continue using the instance. For details about statuses after instances have expired and the associated impacts, see **Impact of Expiration**.

## How to Renew Subscriptions

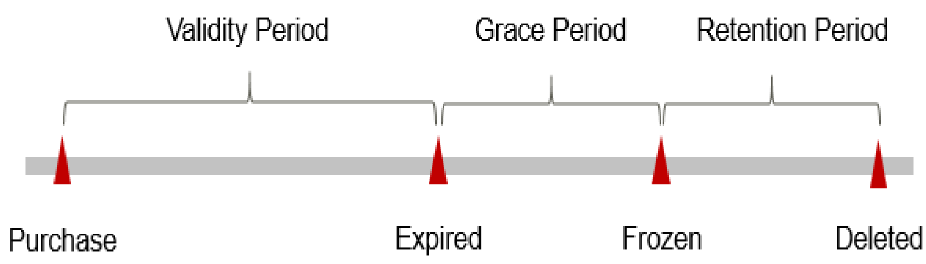
You can renew a yearly/monthly GeminiDB Cassandra instance manually or automatically.

**Table 2-10** Renewing a yearly/monthly instance

| Method | Description |
| --- | --- |
| **Manually Renewing an Instance** | You can renew a yearly/monthly instance anytime on the console before it is automatically deleted. |
| **Auto-renewing an Instance** | You can enable auto-renewal to automatically renew the instance before it expires. This prevents resources from being deleted in case you forget to renew a subscription. |

You can select a method to renew a yearly/monthly instance based on the phase the instance is currently in.

**Figure 2-12** Selecting a renewal method based on the instance's current phase




- An instance is in the **Provisioned** state after it is provisioned.
- When an instance subscription expires, the status will change from **Provisioned** to **Expired**.
- If an expired instance is not renewed, it enters a grace period. If it is not renewed by the time the grace period expires, the instance will be frozen and enter a retention period.
- If you do not renew the subscription before the retention period expires, your resources will be automatically deleted.

NOTE

- During the retention period, you cannot access or use your instance but the data stored in it can be retained. The retention period for Huawei Cloud International website is 15 days.
- During the grace period, you can access and use only some resources of your instance. The grace period for Huawei Cloud International website is 15 days.

You can enable auto-renewal any time before an instance expires. By default, the system will make the first attempt to charge your account for the renewal at 03:00, seven days before the expiry date. If this attempt fails, it will make another attempt at 03:00 every day until the subscription is renewed or expired. You can change the auto-payment date for renewal as required.

## 2.6.2 Manually Renewing an Instance

You can renew a yearly/monthly instance anytime on the console before it is automatically deleted.

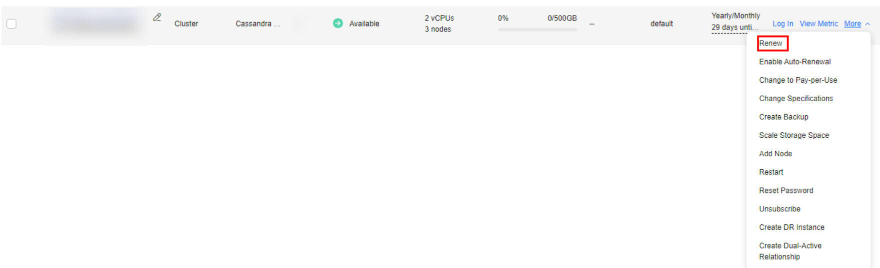
### Renewing an Instance on the Console

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

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

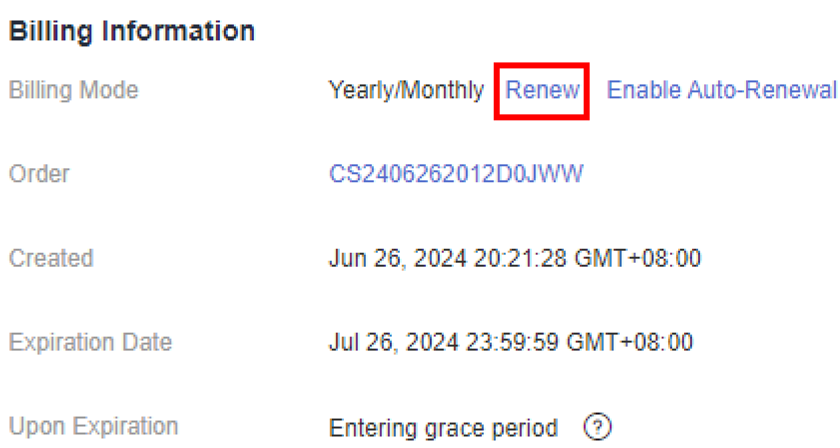
**Step 3** On the **Instances** page, locate the target instance and choose **More** > **Renew** in the **Operation** column.

**Figure 2-13** Renewing an instance




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 2-14** Renewing an instance



NOTE

You can also renew multiple instances all at once:

1. Select the yearly/monthly instances to be renewed.
2. Click **Renew** above the instance list.

**Step 4** On the displayed page, renew the instances.

**----End**

## Renewing an Instance in Billing Center

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

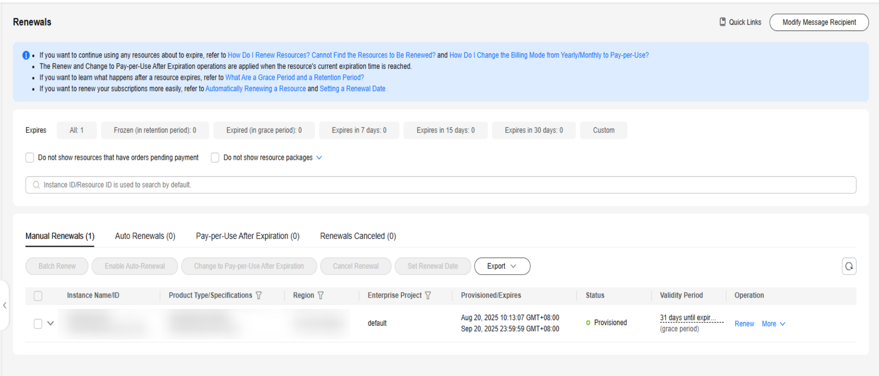
**Step 2** On the top menu bar, choose **Billing** > **Renewal**.

The **Renewals** page is displayed.

**Step 3** Select the search criteria.

On the **Manual Renewals**, **Auto Renewals**, **Pay-per-Use After Expiration**, and **Renewals Canceled** pages, you can view the instances to be renewed.

**Figure 2-15** Renewal management

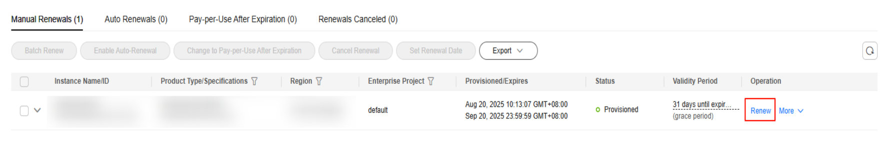


You can move all resources to be manually renewed to the **Manual Renewals** tab. For details, see Restoring to Manual Renewal.

**Step 4** Manually renew resources.

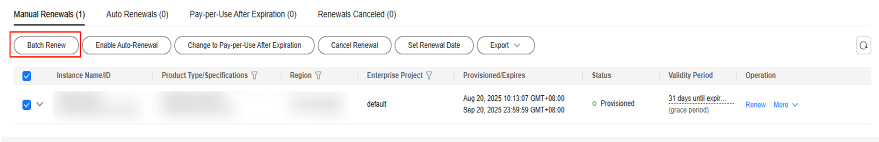
- Individual renewal: Locate an instance that you want to renew and click **Renew** in the **Operation** column.

**Figure 2-16** Individual renewal



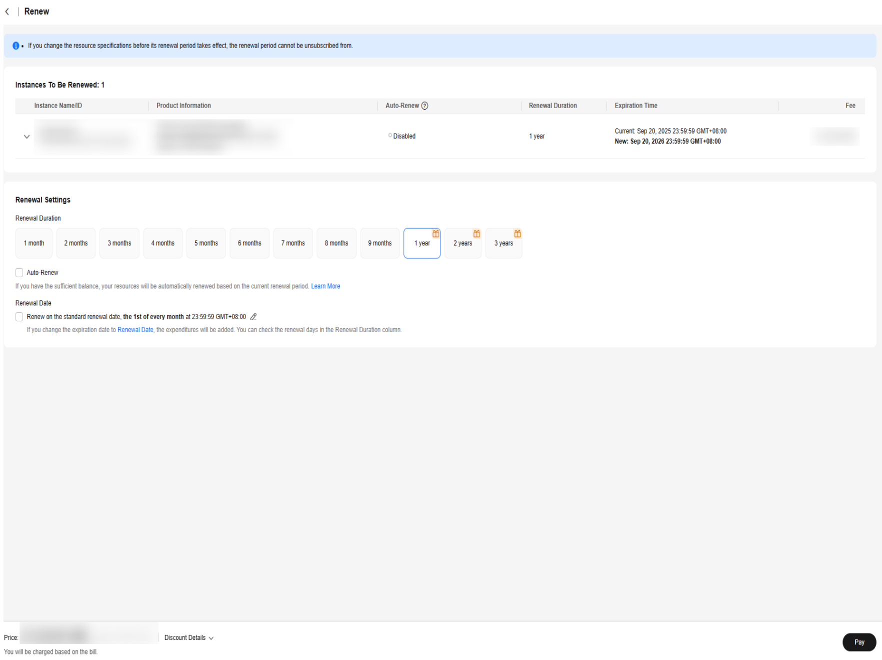
- Batch renewal: Select multiple instances that you want to renew and click **Batch Renew** in the upper left corner.

**Figure 2-17** Batch renewal




**Step 5** Select a renewal duration and optionally select **Renew on the standard renewal date**. For details, see **Setting the Same Renewal Day for Yearly/Monthly Resources**. Confirm the price and click **Pay**.

**Figure 2-18** Confirming renewal




**Step 6** Select a payment method and make your payment. Once the order is paid for, the renewal is complete.

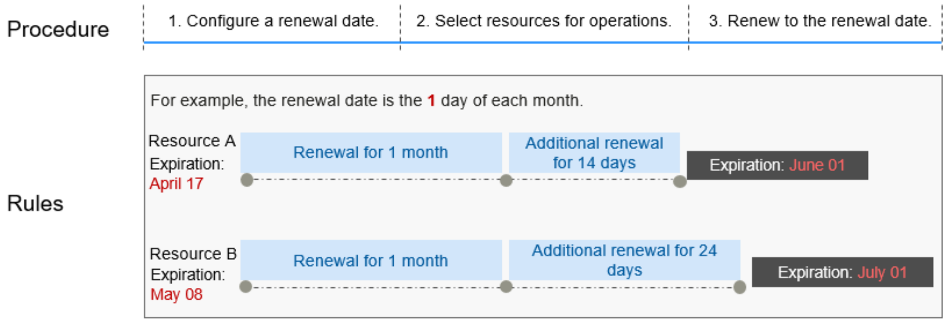
**----End**

## Setting the Same Renewal Day for Yearly/Monthly Resources

If the instances have different expiry dates, you can set the same renewal day, for example, the first day of each month, to make it easier to manage renewals.

In **Figure 2-19**, a user sets the same renewal day for two resources that will expire at different dates.

**Figure 2-19** Setting the same renewal day for resources with different expiry dates




For more details, see **Setting a Renewal Date**.

## 2.6.3 Auto-renewing an Instance

Auto-renewal can prevent instances from being automatically deleted if you forget to manually renew them. The auto-renewal rules are as follows:

- The first auto-renewal date is based on when an instance expires and the billing cycle.
- The auto-renewal period is subject to the renewal duration you select.
  - Your monthly subscription will be renewed each month.
  - Your yearly subscription will be renewed each year.
- You can enable auto-renewal anytime before an instance expires. By default, the system will make the first attempt to renew your account at 03:00 seven days before the expiry date. If this attempt fails, it will make another attempt at 03:00 every day until the subscription is renewed or expired.
- After auto-renewal is enabled, you can still renew the instance manually if you want to. After a manual renewal is complete, auto-renewal is still valid, and the renewal fee will be deducted from your account seven days before the new expiry date.
- By default, the renewal fee is deducted from your account seven days before the new expiry date. You can change this auto-renewal payment date as required.

For more information about auto-renewal rules, see **Auto-Renewal Rules**.

### Prerequisites

Your yearly/monthly instance is not expired.

### Enabling Auto-Renewal During Purchase

You can enable auto-renewal on the instance purchase page, as shown in **Figure 2-20**. For details, see **Buying an Instance**.

**Figure 2-20** Enabling auto-renewal

## Enabling Auto-Renewal on the Renewals Page

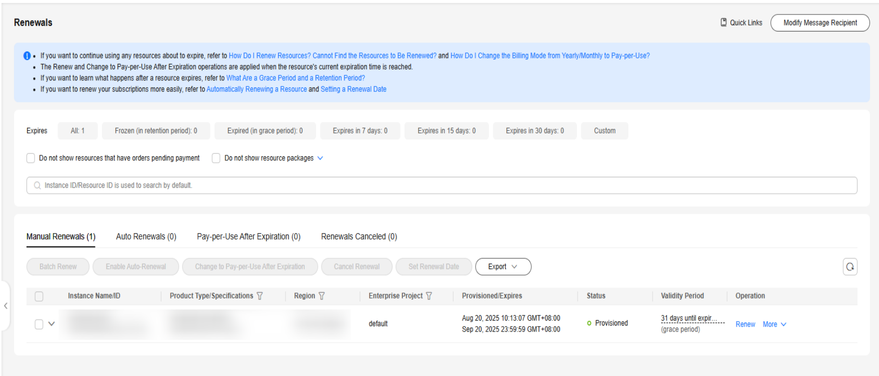
**Step 1** Log in to the Huawei Cloud console.

**Step 2** On the top navigation bar, choose **Billing** > **Renewal**.

**Step 3** Select the search criteria.

- On the **Auto Renewals** page, you can view the resources that auto-renewal has been enabled for.
- You can enable auto-renewal for resources on the **Manual Renewals**, **Pay-per-Use After Expiration**, and **Renewals Canceled** pages.

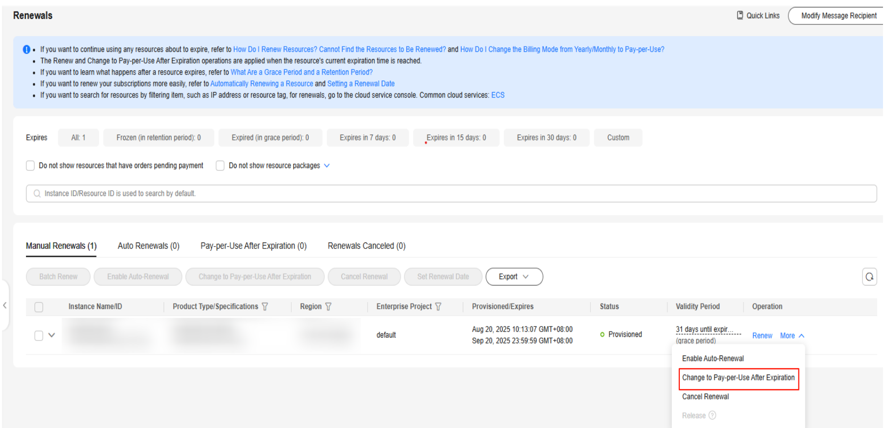
**Figure 2-21** Renewal management




**Step 4** Enable auto-renewal for yearly/monthly resources.

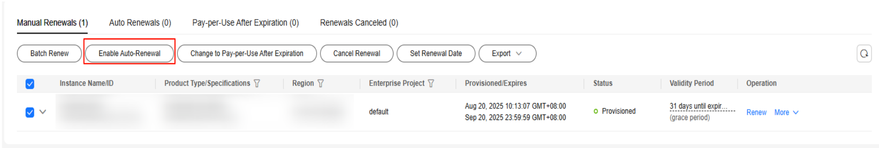
- Enabling auto-renewal for a single instance: Locate the instance that you want to enable auto-renewal for and choose **More** > **Enable Auto-Renew** in the **Operation** column.

**Figure 2-22** Enabling auto-renewal for an instance



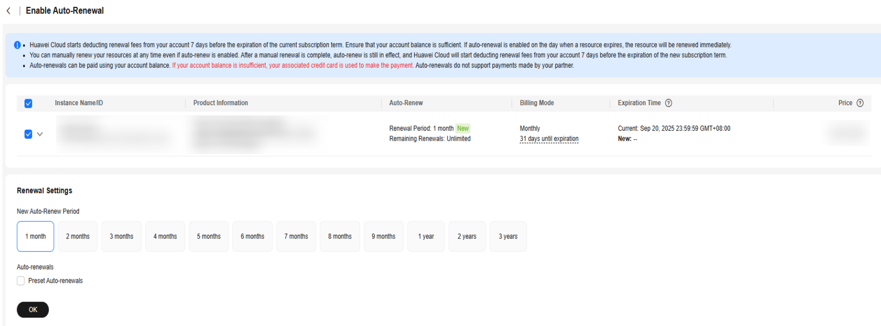

- Enabling auto-renewal for multiple instances at a time: Select the instances that you want to enable auto-renewal for and click **Enable Auto-Renew** above the list.

**Figure 2-23** Enabling auto-renewal for multiple instances




**Step 5** Select a renewal period, specify the auto-renewal times, and click **Pay**.

**Figure 2-24** Enabling auto-renewal




**----End**

# 2.7 Bills

You can view the resource usage and bills for different billing cycles on the **Bills** page in the Billing Center.

## Bill Generation

A bill is generated after a yearly/monthly instance is paid.

The usage of pay-per-use instances is reported to the billing system at a fixed interval. A pay-per-use resource is billed by the hour, day, or month, depending on the resource's usage type. The GeminiDB Cassandra instance usage is billed by the hour. For details, see **Bill Run for Pay-per-Use Resources**.

The fee deduction time of a pay-per-use instance may be later than the settlement period. For example, if an instance is deleted at 08:30, the fees generated from 08:00 to 09:00 are usually deducted at about 10:00. In Billing Center, choose **Billing** > **Transactions and Detailed Bills** > **Transaction Bills**. **Expenditure Time** in the bill indicates the time when the pay-per-use resource is used.

## Viewing Bills of a Specific Resource

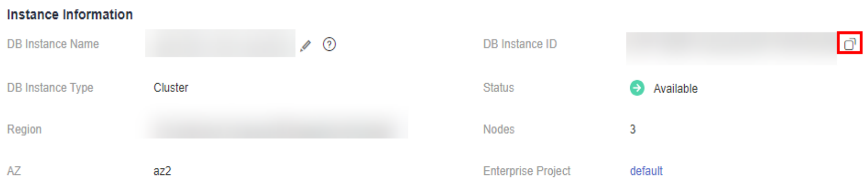
[Method 1: Use the instance ID to search for a bill.]

**Step 1** **Log in to the Huawei Cloud console**. Choose **Databases** > **GeminiDB Cassandra API**.

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

**Step 3** Click the icon shown in the figure below to copy the instance ID.

**Figure 2-25** Copying the instance ID

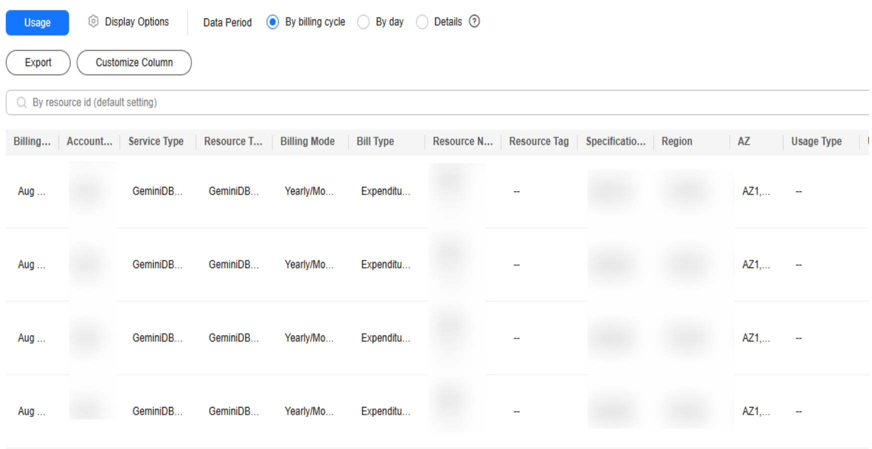



**Step 4** On the top menu bar, choose **Billing** > **Bills**.

The **Bills** page is displayed.

**Step 5** In the navigation pane, choose **Billing** > **Expenditure Details**. Select **Resource ID** as the filter, enter the resource ID, and click to search for the resource bills.

**Figure 2-26** Querying resource bills




By default, the bill details are displayed by usage and billing cycle. You can choose other display options. For details, see **Bills**.

**----End**

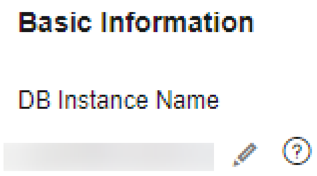
[Method 2: Use the resource name to search for a bill.]

**Step 1** **Log in to the Huawei Cloud console**. Choose **Databases** > **GeminiDB Cassandra API**.

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

**Step 3** On the **Basic Information** > **Instance Information** page, obtain the instance name.

**Figure 2-27** Obtaining an instance name

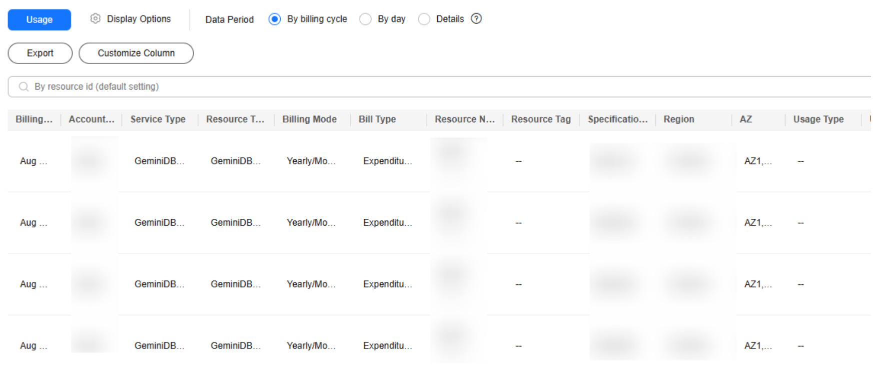



**Step 4** On the top menu bar, choose **Billing** > **Bills**.

The **Bills** page is displayed.

**Step 5** In the navigation pane, choose **Billing** > **Expenditure Details**. Select **Resource Name** as the filter, enter the resource name, and click the search icon to search for the resource bills.

**Figure 2-28** Querying resource bills




By default, the bill details are displayed by usage and billing cycle. You can choose other display options. For details, see **Bills**.

**----End**

## Scenario Example: Checking the Consistency of the Actual Usage and Billed Usage

Assume that you purchased a pay-per-use GeminiDB Cassandra instance at 10:09:06 on April 8, 2023 and deleted it later that day, at 12:09:06.

- Transaction Records

  Pricing is listed on a per-hour basis, and bills are calculated down to the second. You can check the transaction records against the actual usage. The resources are billed separately. **Table 2-11** uses storage as an example.

**Table 2-11** Transaction records

| | |
|---|---|
| **Product Type** | GeminiDB Cassandra |
| **Resource Type** | Storage |
| **Billing Mode** | Pay-per-use |
| **Expenditure Time** | For the period of time from 10:09:06 to 12:09:06 on April 08, 2023, 6 transaction records would be generated for the resource usage in the following periods:<br>● 10:09:06 – 11:00:00<br>● 11:00:00 – 12:00:00<br>● 12:00:00 – 12:09:06 |
| **List Price** | List price on the official website = Usage x Unit price x Capacity<br>The instance was used for 3,054 seconds in the first period. You can check its unit price in **GeminiDB Price Calculator**. The list price in the first period is USD0.02375333 = (3054/3600) × 0.0007 × 40. You can also calculate the list price in the other periods. |
| **Discounted Amount** | You can enjoy discounts on cloud services, such as business, partner-authorized, and promotional discounts. The discounts are calculated based on the list price. |
| **Truncated Amount** | Billing of Huawei Cloud is calculated to the 8th decimal place. However, the amount due is truncated to the 2nd decimal place. The third and later decimal places are referred to as the truncated amounts.<br>For example, in the first billing cycle, the truncated amount is USD0.00375333. |
| **Amount Due** | Amount due = List price – Discount amount – Truncated amount<br>Take the first period as an example. If the discount amount is 0, the amount due is $0.02 USD (0.02375333 – 0 – 0.00375333). |

- Bill details of the GeminiDB Cassandra instance

  Bill details can be displayed in multiple ways. By default, the bill details are displayed by usage and billing cycle. You can check the information listed in **Table 2-12** against the actual usage.

**Table 2-12** Bill details

| | |
|---|---|
| **Product Type** | GeminiDB Cassandra |

| | |
|---|---|
| **Resource Type** | Storage |
| **Billing Mode** | Pay-per-use |
| **Resource Name/ID** | Name and ID<br>Example: nosql-b388 and 21e8811a64bf4de88bc2e2556da17983in12 |
| **Specifications** | Storage |
| **Usage Type** | Duration of a pay-per-use GeminiDB Cassandra instance |
| **Unit Price** | When pay-per-use billing is used, the unit price is only provided if the amount is equal to the usage multiplied by the unit price. No unit price is provided in other pricing modes, for example, tiered pricing.<br>You can check the unit price of a pay-per-use instance in **GeminiDB Price Calculator**. |
| **Unit** | USD/GB/Hour in **GeminiDB Price Calculator** |
| **Usage** | Depends on the unit of the unit price, which is USD/GB/hour. Storage usage is billed by the hour. In this example, the total duration is 2 hours. |
| **Usage Unit** | Hour |
| **List Price** | List price on the official website = Usage x Unit price x Capacity<br>The instance has been used for 2 hours. Its unit price is displayed in **GeminiDB Price Calculator**. The list price is USD0.056 (2 × 0.0007 × 40). |
| **Discounted Amount** | You can enjoy discounts on cloud services, such as business, partner-authorized, and promotional discounts. The discounts are calculated based on the list price. |
| **Amount Due** | Amount that should be paid for used cloud services after discounts are applied. |

## 2.8 Arrears

If the available account balance is less than the amount to be settled, the account will be in arrears. To continue using your instances, you need to top up your account in a timely manner.

## Arrears Reason

If you do not have yearly/monthly instances, your account falls into arrears any time your configured payment method is unable to pay for the used resources on the pay-per-use basis.

## Arrears Impact

- Yearly/Monthly

  This is a pre-paid billing mode, so you can continue using yearly/monthly GeminiDB Cassandra resources even if your account is in arrears. However, you cannot perform operations such as purchasing GeminiDB Cassandra instances, upgrading instance specifications, and renewing subscriptions, because they will generate new expenditures.

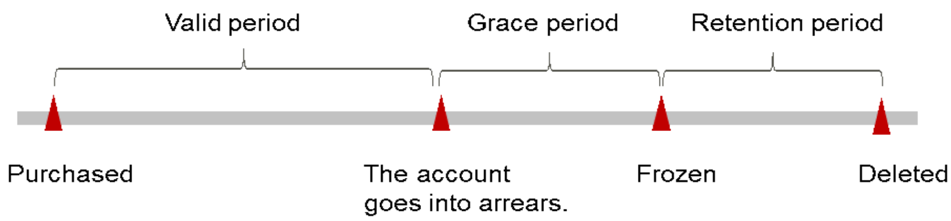
- Pay-per-use

  If your configured payment method is unable to pay a bill for pay-per-use resources, the resources enter a grace period. After you top up your account, Huawei Cloud will bill you for expenditures generated by the resources during the grace period. You can view the expenditures on the **Overview** page of the Billing Center.

  If your account is still in arrears after the grace period ends, the resources enter the retention period and their status turns to **Frozen**. You cannot perform any operations on these resources.

  After the retention period ends, the compute resources (vCPUs and memory) and EIPs will be released and cannot be restored.

**Figure 2-29** Lifecycle of a pay-per-use instance




NOTE

The grace period and retention period are both 15 days.

## Avoiding and Handling Arrears

Make sure you have a valid payment method configured as soon as possible after your account is in arrears. For details, see **Topping Up an Account**.

If a GeminiDB Cassandra instance is no longer used, you can delete it to avoid generating further expenditures.

To help make sure your account never falls into arrears, you can configure the **Balance Alert** on the **Overview** page of the Billing Center. Then, any time an expenditure quota drops to below the threshold you specify, Huawei Cloud automatically notifies you by SMS or email.

# 2.9 Billing Termination

## Yearly/Monthly Resources

When you purchase a yearly/monthly resource, such as a yearly/monthly GeminiDB Cassandra instance, you make a one-time up-front payment. By default, the billing automatically stops when the purchased subscription expires.

- You can unsubscribe from a yearly/monthly resource before it expires. Depending on whether coupons or discounts were used, Huawei Cloud may issue you a refund. For details about unsubscription rules, see **Unsubscriptions**.
- If you have enabled auto-renewal but no longer wish to automatically renew the subscription, disable it before the auto-renewal date (7 days before the expiration date by default) to avoid unexpected expenditures.

## Pay-per-Use Resources

If pay-per-use resources, such as pay-per-use GeminiDB Cassandra instances, are no longer required, delete them in a timely manner.

## Searching for Resources from Bills and Stopping Billing

To ensure that all related resources are deleted, you can search the billing records by resource ID, and then delete the resources you identify in this way.

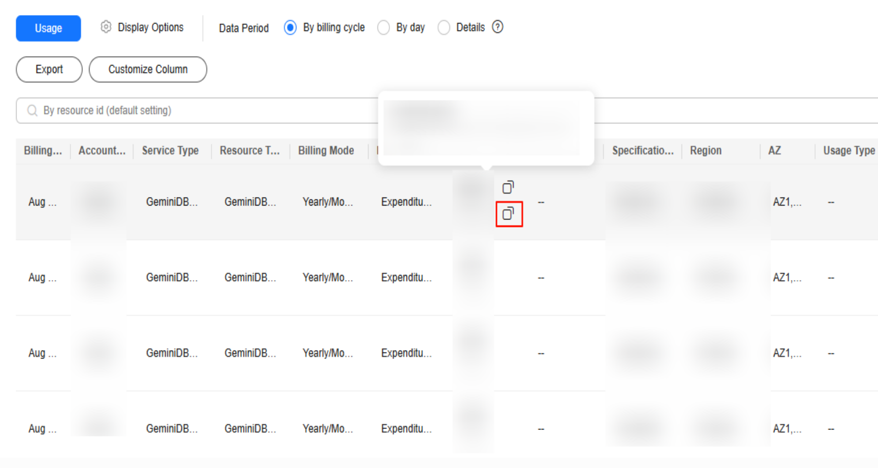
[Method 1: Use the resource ID in the bill to search for the resource.]

**Step 1** **Log in to the Huawei Cloud console**. On the top menu bar, choose **Billing** > **Bills**.

The **Bills** page is displayed.

**Step 2** In the navigation pane, choose **Billing** > **Expenditure Details**. Click the icon shown in the following figure to copy the resource ID.

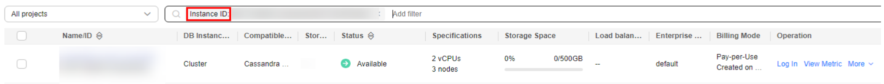
**Figure 2-30** Copying the resource ID



**Step 3** Log in to the Huawei Cloud console. Choose **Databases** > **GeminiDB Cassandra API**.

**Step 4** Select the region where the resource is located. Select **Instance ID**, enter the resource ID copied from **Step 2**, and click 🔍 to search for the resource.

**Figure 2-31** Searching for resources




**Step 5** Locate the instance you want to delete and click **More** > **Delete** in the **Operation** column. Ensure that the resource is not found in the list.

> **NOTE**
>
> You are billed one hour after the resource usage is calculated, so a bill may still be generated after the pay-per-use resource is deleted. For example, if you delete an instance (which is billed on an hourly basis) at 08:30, the expenditures for that hour from 08:00 to 09:00 are usually not billed until about 10:00.

**----End**

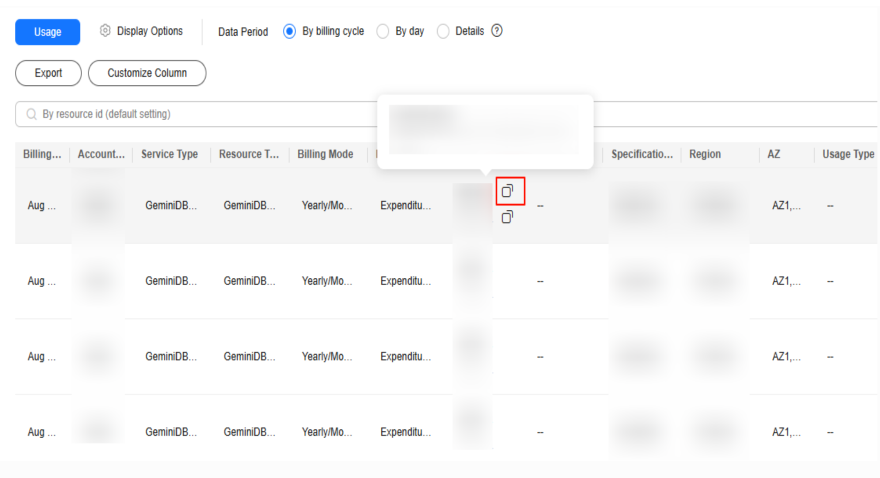
[Method 2: Use the resource name in the bill to search for the resource.]

**Step 1** Log in to the Huawei Cloud console. On the top menu bar, choose **Billing** > **Bills**.

The **Bills** page is displayed.

**Step 2** In the navigation pane, choose **Billing** > **Expenditure Details**. Click the icon shown in the following figure to copy the resource name.

**Figure 2-32** Copying the resource name

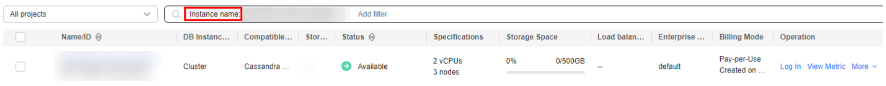



**Step 3** Log in to the Huawei Cloud console. Choose **Databases** > **GeminiDB Cassandra API**.

**Step 4** Enter the instance name copied from **Step 2** in the search box and click .

**Figure 2-33** Searching for resources



**Step 5** Locate the instance you want to delete and click **More** > **Delete** in the **Operation** column. Ensure that the resource is not found in the list.

NOTE

You are billed one hour after the resource usage is calculated, so a bill may still be generated after the pay-per-use resource is deleted. For example, if you delete an instance (which is billed on an hourly basis) at 08:30, the expenditures for that hour from 08:00 to 09:00 are usually not billed until about 10:00.

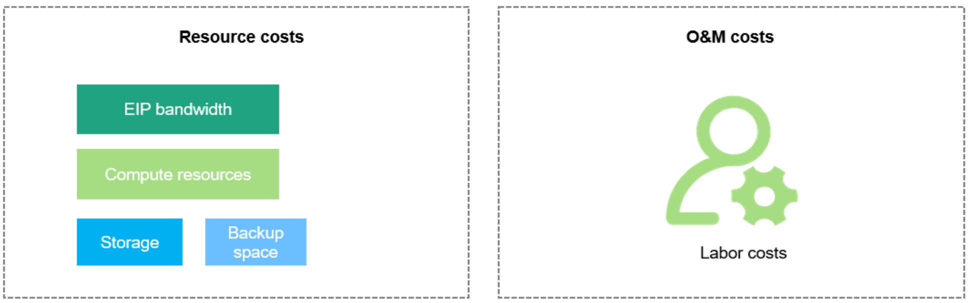
**----End**

# 2.10 Cost Management

## 2.10.1 Cost Composition

GeminiDB Cassandra costs consist of two parts:

- Resource costs: costs of compute and storage resources. For details, see **Billing Modes**.
- O&M costs: labor costs incurred during the use of GeminiDB Cassandra.




## 2.10.2 Cost Allocation

A good cost accountability system is a prerequisite for cost management. It ensures that departments, business teams, and owners are accountable for their respective cloud costs. An enterprise can allocate cloud costs to different teams or projects so as to have a clear picture of their respective costs.

Huawei Cloud **Cost Center** provides various tools for you to group costs in different ways. You can experiment with these tools and find a way that works best for you.

- **By linked account**

The enterprise master account can manage costs by grouping the costs of its member accounts by linked account. For details, see **Viewing Costs by Linked Account**.

- **By enterprise project**

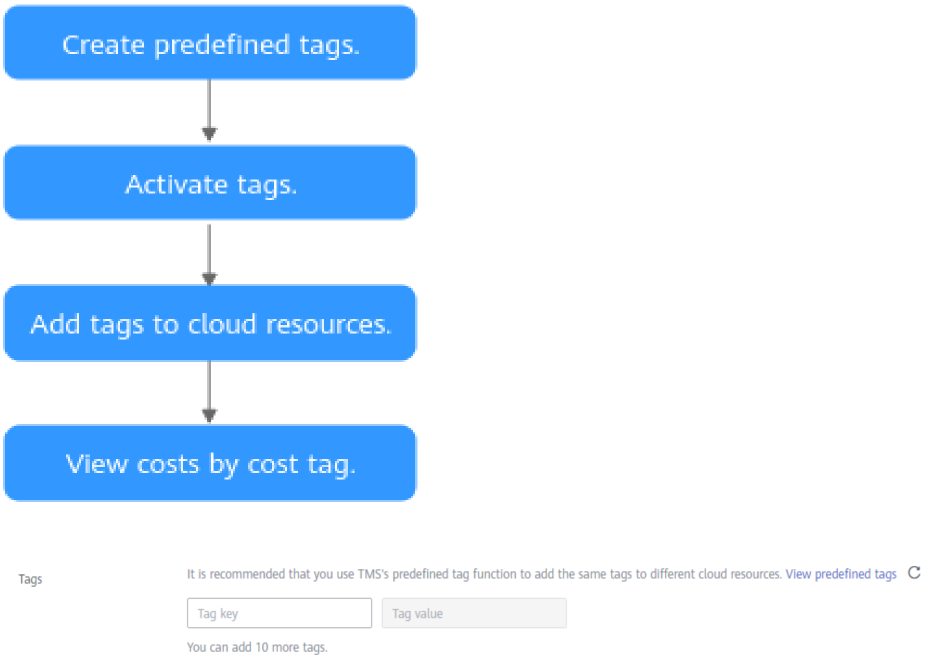
  Before allocating costs, enable Enterprise Project Management Service (EPS) and plan your enterprise projects based on your organizational structure or service needs. When purchasing cloud resources, select an enterprise project so that the costs of resources will be allocated to the selected enterprise project. For details, see **Viewing Costs by Enterprise Project**.

  **Figure 2-34** Selecting an enterprise project


- **By cost tag**

  You use tags to sort your Huawei Cloud resources in a variety of different ways, for example, by purpose, owner, or environment. The following is the process of managing costs by predefined tags (recommended).

  **Figure 2-35** Adding a tag




  For details, see **Viewing Costs by Cost Tag**.

- **By cost category**

  You can use cost categories provided by **Cost Center** to split shared costs. Shared costs are the costs of resources (compute, network, storage, or resource packages) shared across multiple departments or the costs that cannot be directly split by cost tag or enterprise project. These costs are not directly attributable to a singular owner, and they cannot be categorized into a singular cost type. In this case, you can define cost splitting rules to fairly allocate these costs among teams or business units. For more information, see **Allocating Costs By Cost Category**.

## 2.10.3 Cost Analysis

To precisely control and optimize your costs, you need a clear understanding of what parts of your enterprise incurred different costs. **Cost Center** visualizes your original costs and amortized costs using various dimensions and display filters for cost analysis so that you can analyze the trends and drivers of your service usage and costs from a variety of perspectives or within different defined scopes.

You can also use cost anomaly detection provided by **Cost Center** to detect unexpected expenses in a timely manner. In this way, costs can be monitored, analyzed, and traced.

For details, see **Performing Cost Analysis to Explore Costs and Usage** and **Enabling Cost Anomaly Detection to Identify Anomalies**.

## 2.10.4 Cost Optimization

You can identify resources with high costs based on the analysis results in the cost center, determine the causes of high costs, and take optimization measures accordingly.

### Resource rightsizing

- View GeminiDB Cassandra monitoring metrics on Cloud Eye, such as the CPU, memory, and disk usage. If the current configuration is too high, you can reduce the configuration by changing specifications.
- Monitor idle GeminiDB Cassandra resources and delete idle instances in a timely manner.

### Billing mode selection

Different types of services have different requirements on resource usage periods, so the most economical billing mode for one resource may not be the best option for another resource.

- For mature services that tend to be stable for the long term, select yearly/monthly billing.
- For short-term, unpredictable services that experience traffic bursts and cannot afford to be interrupted, select pay-per-use billing.
- Monitor the lifecycle of instances and renew yearly/monthly resources that are about to expire in a timely manner.

# 2.11 Billing FAQs

## 2.11.1 What Are the Differences Between Yearly/Monthly and Pay-per-Use Billing?

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 billing is a postpaid payment mode. This billing mode allows you to make or cancel subscriptions at any time. Pricing is listed on a per-hour basis, but bills are calculated based on the actual usage duration.

## 2.11.2 Can I Switch Between Yearly/Monthly and Pay-per-Use Billing?

You can change the billing mode of your instance from yearly/monthly to pay-per-use or vice versa.

- For details about how to change the billing mode from yearly/monthly to a pay-per-use, see **Changing a Yearly/Monthly Instance to Pay-per-Use**.
- For details about how to change the billing mode from pay-per-use to yearly/monthly, see **Changing a Pay-per-Use Instance to Yearly/Monthly**.

## 2.11.3 How Do I Renew a Single or Multiple Yearly/Monthly Instances?

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

### Precautions

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

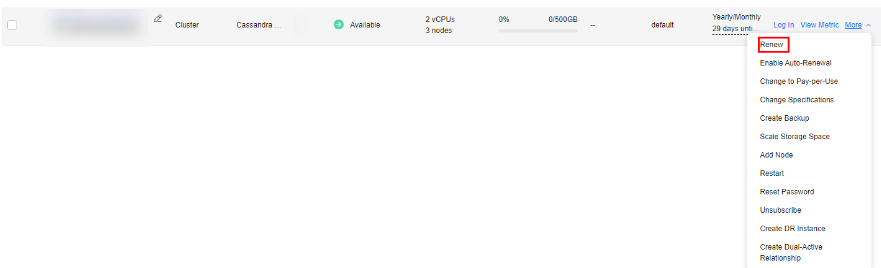
### Renewing a Single Yearly/Monthly Instance

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

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

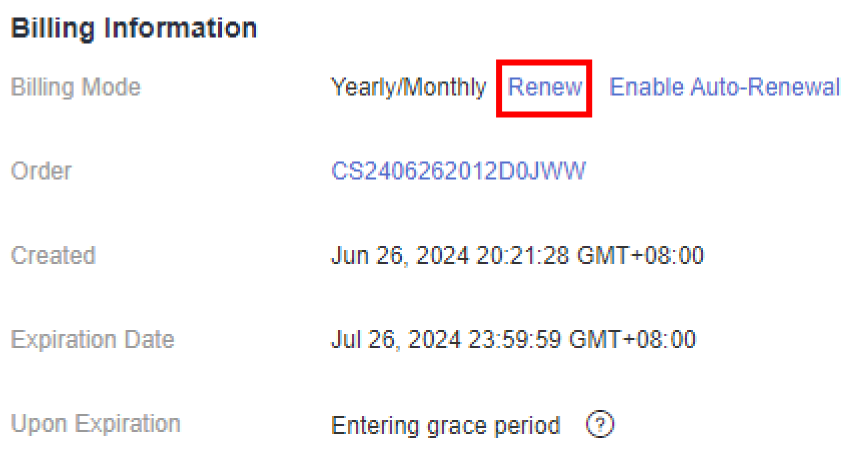
**Step 3** On the **Instances** page, locate the target instance and choose **More** > **Renew** in the **Operation** column.

**Figure 2-36** Renewal




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 2-37** Renewal




**Step 4** On the displayed page, renew the instance.

**----End**

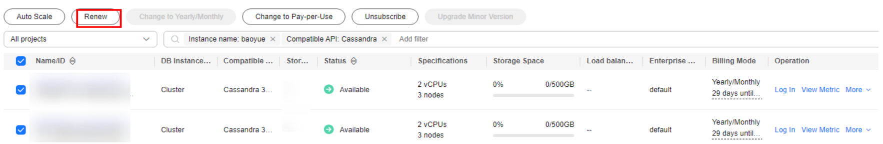
## Renewing Instances in Batches

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

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** On the **Instances** page, select the instances that you want to renew and click **Renew** above the instance list.

**Figure 2-38** Renewing instances in batches



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

**----End**

# 2.11.4 How Do I Unsubscribe from a Yearly/Monthly Instance?

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

## Precautions

- The unsubscription action cannot be undone. 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.

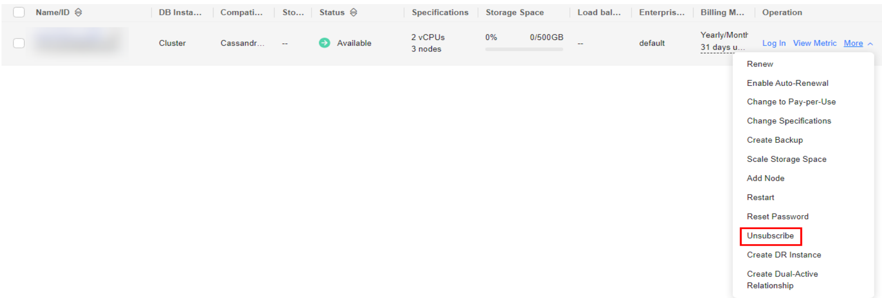
## Unsubscribing a Single Yearly/Monthly Instance

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

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** On the **Instances** page, locate the instance you want to unsubscribe from and choose **More** > **Unsubscribe** in the **Operation** column.

**Figure 2-39** Unsubscribe




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

**Step 5** On the displayed page, confirm the order to be unsubscribed and select a reason. Then, click **Confirm**.

For details, see **Unsubscription Rules**.

**Step 6** 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 7** View the unsubscription result. After you unsubscribe from the instance order, the instance is no longer displayed in the instance list on the **Instances** page.

**----End**

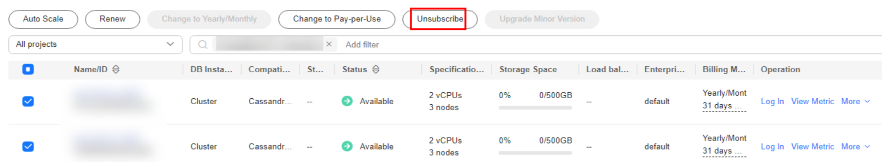
## Batch Unsubscribing from Yearly/Monthly Instances

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

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** 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 2-40** Unsubscribe



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

**Step 5** On the displayed page, confirm the order to be unsubscribed and select a reason. Then, click **Confirm**.

For details, see **Unsubscription Rules**.

**Step 6** 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 7** View the unsubscription result. After you unsubscribe from the instance order, the instance is no longer displayed in the instance list on the **Instances** page.

**----End**

# 3 Getting Started with GeminiDB Cassandra API

## 3.1 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 3-1** 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<br>• 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**.

# 3.2 Buying and Connecting to a GeminiDB Cassandra Instance

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

GeminiDB Cassandra, DynamoDB, and HBase 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 Huawei Cloud console**.
2. In the service list, choose **Databases** > **GeminiDB**.
3. On the **Instances** page, click **Buy DB Instance**.
4. 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 4-1** lists details about the parameters.

**Figure 3-1** Billing mode and basic information (classic storage)

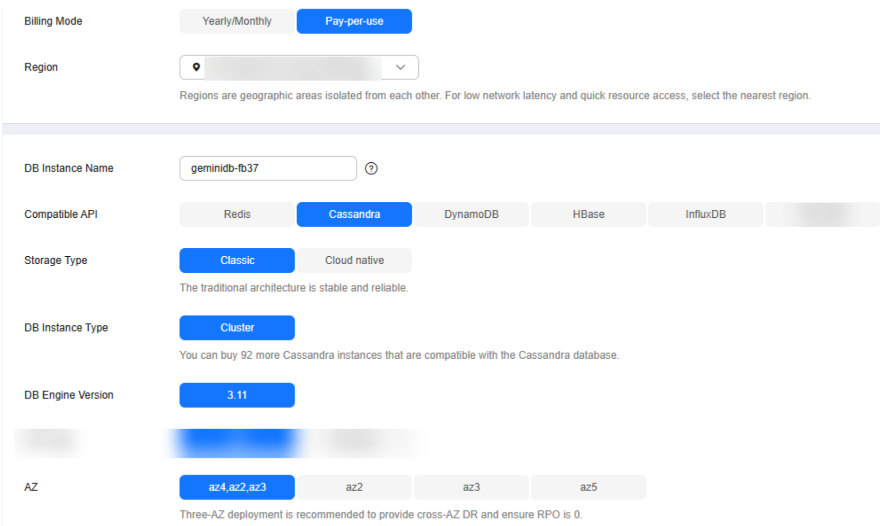


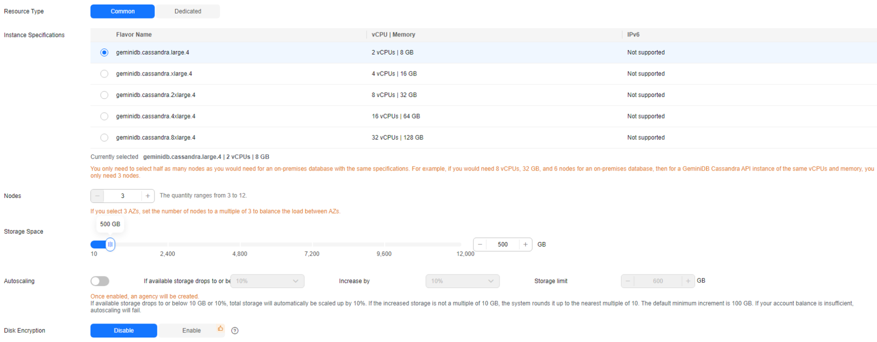
**Figure 3-2** Billing mode and basic information (cloud native storage)


| Parameter | Example Value | Description |
|---|---|---|
| Billing Mode | Pay-per-use | Billing mode of an instance<br>● **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.<br>● **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 **CN-Hong Kong**. | Region where a tenant is located<br>**NOTE**<br>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. |

| Parameter | Example Value | Description |
|---|---|---|
| DB Instance Name | User-defined | The instance name:<br>● Can be the same as an existing instance name.<br>● 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, HBase, MongoDB, 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<br>● **Cloud native**: new, more flexible, new-gen version with support for more AZs |
| DB Instance Type | Cluster | **Cluster**<br>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.<br>Instances can be deployed in a single AZ or three AZs.<br>● To deploy instances in a single AZ, select one AZ.<br>● To deploy instances across AZs for disaster recovery, select three AZs, where the instance nodes will be evenly distributed. |

**Figure 3-3** Specifications and storage




| Parameter | Example Value | Description |
|---|---|---|
| Instance Specifications | 2U8GB | Select appropriate specifications based on the CPU-memory ratio.<br>After an instance is created, you can change its specifications. For details, see **Changing vCPUs and Memory**. |
| Nodes | 3 | Number of nodes that the instance is deployed on.<br>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.<br>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**.<br>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.<br>You can enable **Auto Scale** after an instance is created. For details, see **Automatically Scaling Up Storage Space**. |

| Parameter | Example Value | Description |
|---|---|---|
| Disk Encryption | Disable | **Disable** is selected by default.<br>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 3-4** Network configuration

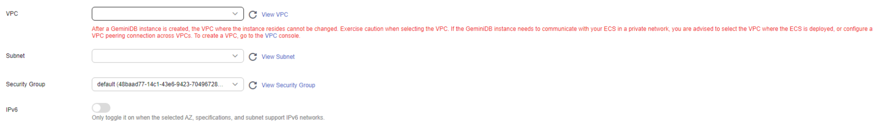
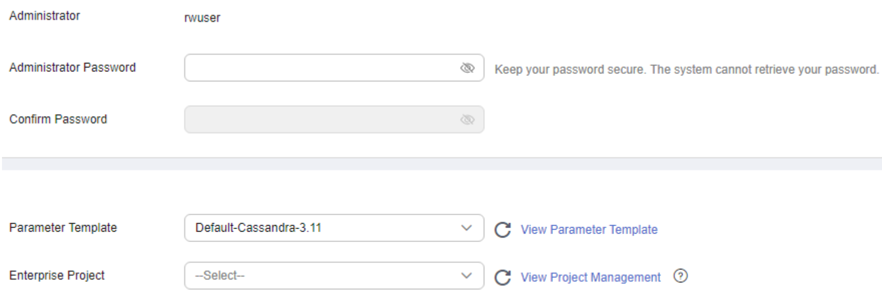



**Figure 3-5** Database configuration




| 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.<br>**NOTE**<br>• After a GeminiDB Cassandra instance is created, its VPC cannot be changed.<br>• If you want to connect to an instance using an ECS over a private network, ensure that the GeminiDB Cassandra instance and the ECS are in the same VPC. If they are not, create a **VPC peering connection** between them. |

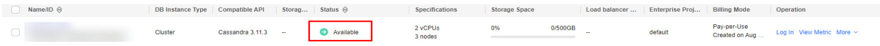
| Parameter | Example Value | Description |
| --- | --- | --- |
| Subnet | default_subnet | A subnet provides dedicated network resources that are logically isolated from other networks for security purposes. |
| Security Group | default | A security group controls access between your instance and other 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:<br>● Can include 8 to 32 characters.<br>● Can include uppercase letters, lowercase letters, digits, and any of the following special characters: ~!@#%^*-_=+?<br>● For security reasons, set a strong password. The system will verify the password strength.<br>Keep your password secure. The system cannot retrieve it if it is lost. |
| 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.<br>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.<br>An enterprise project groups cloud resources, so you can manage resources and members by project. The default project is **default**.<br>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.

5. On the displayed page, confirm instance details. To modify the configurations, click **Previous**.

6. If no modification is required, read and agree to the service agreement and click **Submit**.
7. Click **Back to Instance Management** to go to the instance list.
8. 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 3-6** 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.

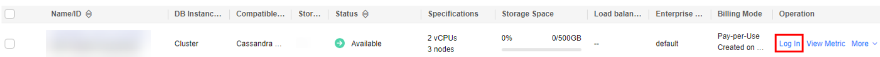
## Procedure

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

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

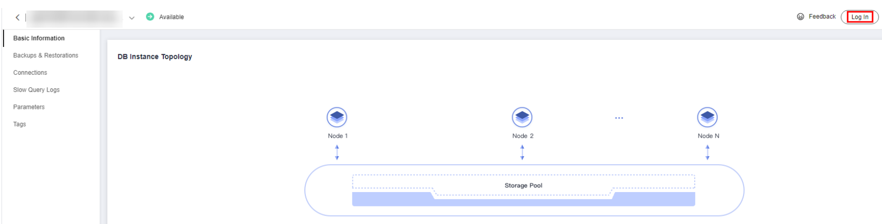
**Step 3** On the **Instance Management** page, locate the target DB instance and click **Log In** in the **Operation** column.

**Figure 3-7** 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 3-8** Logging in to the database



**Step 4** On the displayed login page, enter the administrator username and password and click **Log In**.

For details about how to manage databases through DAS, see **GeminiDB Cassandra Data Management**.

**----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 Getting Started with Common Practices

After purchasing and connecting to a GeminiDB Cassandra DB instance, you can view common practices to better use it.

**Table 3-2** Common practices

| Practice | | Description |
|---|---|---|
| Database rules | **Basic Design** | Describes basic design specifications of GeminiDB Cassandra. |
| | **Database Objects** | Describes rules of using database objects. |
| | **Database Usage** | Describes constraints of using databases. |
| | **Queries** | Describes rules of querying GeminiDB Cassandra. |
| Instance modifications | **Changing an Instance Name** | Describes how to change the name of a GeminiDB Cassandra instance to identify different instances. |
| | **Resetting the Administrator Password** | Describes how to change your administrator password. For security reasons, change it periodically. |
| | **Changing vCPUs and Memory** | Describes how to change the CPU or memory of your instance to suit your service requirements. |
| Data backup | **Managing Automated Backups** | Describes how GeminiDB Cassandra API automatically creates backups for a DB instance during a backup window and saves the backups based on the configured retention period. |
| | **Managing Manual Backups** | Describes how to create manual backups for a DB instance. These backups can be used to restore data for improved reliability. |
| | **Managing Cross-Region Backups** | Describes how to set a cross-region backup policy for a DB instance. Then for disaster recovery, you can restore backups to a new instance in another region. |
| | **Managing Table-level Backups** | Describes how to create a table-level backup for a DB instance. If a database or table is deleted, maliciously or accidentally, backups can help restore your data. |

| Practice | | Description |
|---|---|---|
| Data restoration | **Restoring Data to a New Instance** | Describes how to restore an existing automated or manual backup to a new instance. The restored data is the same as the backup data. |
| | **Restoring a Backup to a Specified Point in Time** | Describes how to use an automated backup to restore instance data to a specified point in time. |
| Log management | **Viewing and Exporting Slow Query Logs** | Describes how to view slow query logs of a GeminiDB Cassandra database. The unit of the execution time is ms. You can identify the SQL statements that take a long time to execute and tune them based on slow query logs. |

# 4 Working with GeminiDB Cassandra API

## 4.1 Using IAM to Grant Access to GeminiDB Cassandra API

### 4.1.1 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.

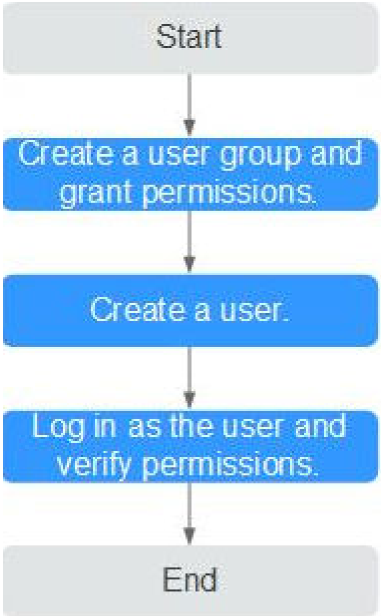
The following describes the procedure for granting permissions (see **Figure 4-1**).

#### Prerequisites

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

## Process Flow

**Figure 4-1** 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 to the management console using the created user, and verify that the user only has read permissions.

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

# 4.1.2 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 **Permissions Policies and Supported Actions**.

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"
                }
        ]
}
```

# 4.2 Buying a GeminiDB Cassandra Instance

This section describes how to buy a GeminiDB Cassandra instance.

GeminiDB Cassandra, DynamoDB-Compatible, and HBase 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 Huawei Cloud console**.

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** On the **Instances** page, click **Buy DB Instance**.

**Step 4** On the displayed page, specify a billing mode and instance specifications and click **Next**.

**Figure 4-2** Billing mode and basic information (classic storage)

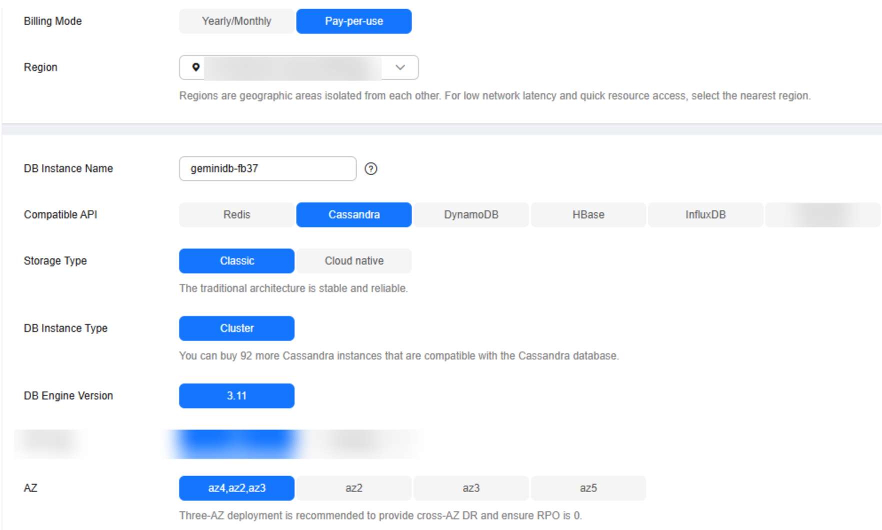


**Figure 4-3** Billing mode and basic information (cloud native storage)

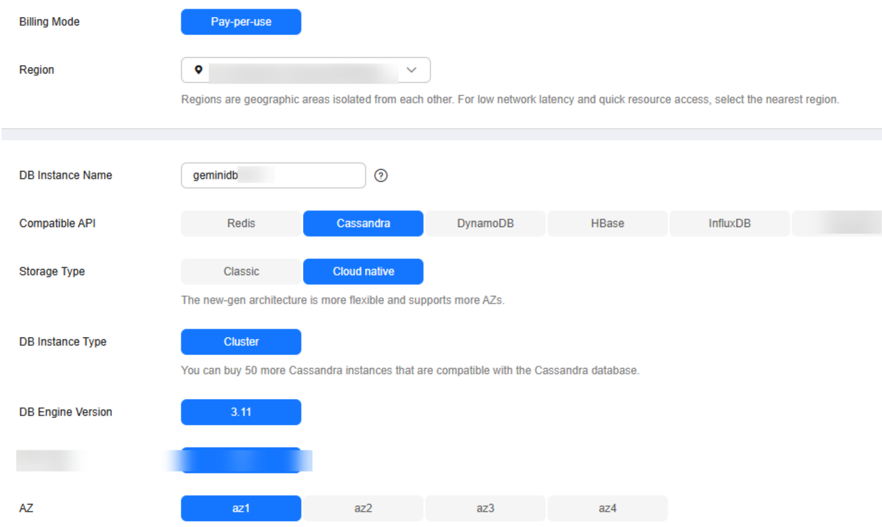



**Table 4-1** Billing parameters

| Parameter | Description |
| --- | --- |
| Billing Mode | Select **Yearly/Monthly** or **Pay-per-use**.<br>● **Yearly/Monthly**<br>– Specify **Required Duration**. The system deducts fees from your account based on the service price.<br>– 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 a Yearly/Monthly Instance to Pay-per-Use**.<br>**NOTE**<br>■ Yearly/Monthly instances cannot be deleted directly. If such an instance is no longer required, unsubscribe from it. For details, see **How Do I Unsubscribe from a Yearly/Monthly Instance?**.<br>■ Yearly/Monthly instances with cloud native storage are now in OBT. To use such an instance, choose **Service Tickets > Create Service Ticket** in the upper right corner of the console and contact the customer service.<br>● **Pay-per-use**<br>– If you select this billing mode, you are billed based on how much time the instance is in use.<br>– To use an instance for a long time, change its billing mode to yearly/monthly to reduce costs. For details, see **Changing a Pay-per-Use Instance to Yearly/Monthly**. |

**Table 4-2** Basic information

| Parameter | Description |
| --- | --- |
| Region | Region where a tenant is located<br>**NOTE**<br>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:<br>● Can be the same as an existing instance name.<br>● Can contain 4 to 64 characters and must start with a letter. It is case-sensitive and allows only letters, digits, hyphens (-), and underscores (_).<br>After an instance is created, you can change its name. For details, see **Changing an Instance Name**. |
| Compatible API | Cassandra<br>GeminiDB is compatible with mainstream NoSQL databases, including Redis, DynamoDB, Cassandra, HBase, MongoDB, and InfluxDB. You can select GeminiDB APIs by following **How Do I Select an API?** |
| Storage Type | ● **Classic**: classic architecture with decoupled storage and compute<br>● **Cloud native**: more flexible, new-gen version with support for more AZs<br>**NOTE**<br>– 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<br>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.<br>Instances can be deployed in a single AZ or three AZs.<br>● To deploy instances in a single AZ, select one AZ.<br>● To deploy instances across AZs for disaster recovery, select three AZs, where the instance nodes will be evenly distributed. |

**Figure 4-4** Specifications and storage

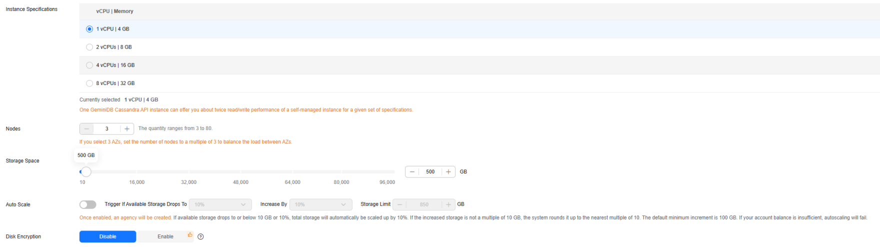


**Table 4-3** Specifications and storage

| 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.<br>Select appropriate specifications based on the CPU-memory ratio.<br>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.<br>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.<br>After an instance is created, you can add nodes. For details, see **Manually Adding Instance Nodes**. |

<table>
<tr><th>Parameter</th><th>Description</th></tr>
<tr><td>Storage Space</td><td>Instance storage space. The range depends on the instance specifications. For details, see Instance Specifications.<br><br>To scale up classic storage, you need to add at least 1 GB each time. To scale up cloud native storage, you need to add at least 10 GB each time. The value must be an integer.<br><br>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:<br>• 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.<br>• 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.<br>• 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.<br><br>After an instance is created, you can scale up its storage if necessary. For details, see Manually Scaling Up Storage Space.<br><br>NOTE<br>• Once Auto Scale is enabled, an agency will be created and fees will be automatically deducted.<br>• 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.<br>• You can enable Auto Scale after an instance is created. For details, see Automatically Scaling Up Storage Space.</td></tr>
</table>

<table>
<tr><th>Parameter</th><th>Description</th></tr>
<tr><td>Disk Encryption</td><td>You can select to enable disk encryption based on service requirements.<ul><li>Disable: Data is not encrypted.</li><li>Enable: If you select this option, 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.<ul><li>Key Name: Select an existing key or create one.</li><li>To use a shared key, ensure that you have created an agency. For details, see Creating an Agency (by a Delegating Party). Select another account from the drop-down list to share the key of the current account. VPC owners can share the keys with one or multiple accounts through Resource Access Manager (RAM). For details, see Creating a Resource Share.</li><li>Enter a key ID. The key must be in the current region.</li></ul>NOTE<ul><li>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.</li><li>An agency will be created after disk encryption is enabled.</li><li>After an instance is created, the disk encryption status and the key cannot be changed.</li><li>The key cannot be disabled, deleted, or frozen when being used. Otherwise, the database becomes unavailable.</li><li>For details about how to create a key, see "Creating a CMK" in Data Encryption Workshop User Guide.</li></ul></li></ul></td></tr>
</table>

**Figure 4-5** Network configuration

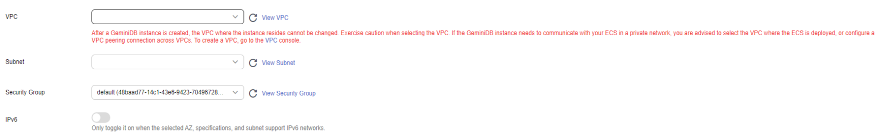


**Table 4-4** Network configuration

| 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.<br>If there are no VPCs available, the system automatically allocates a VPC to you.<br>For details, see "Creating a VPC" in the *Virtual Private Cloud User Guide*.<br>**NOTE**<br>● After a GeminiDB Cassandra instance is created, its VPC cannot be changed.<br>● 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.<br>**NOTE**<br>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.<br>If no security group is available, the system creates one for you. |
| SSL | A security protocol. Secure Sockets Layer (SSL) certificates set up encrypted connections between clients and servers, preventing data from being tampered with or stolen during transmission.<br>You can enable SSL to improve data security. After an instance is created, connect to the instance through SSL. For details, see **SSL**.<br>**NOTE**<br>After SSL is enabled, it cannot be disabled. |

**Figure 4-6** Database configuration

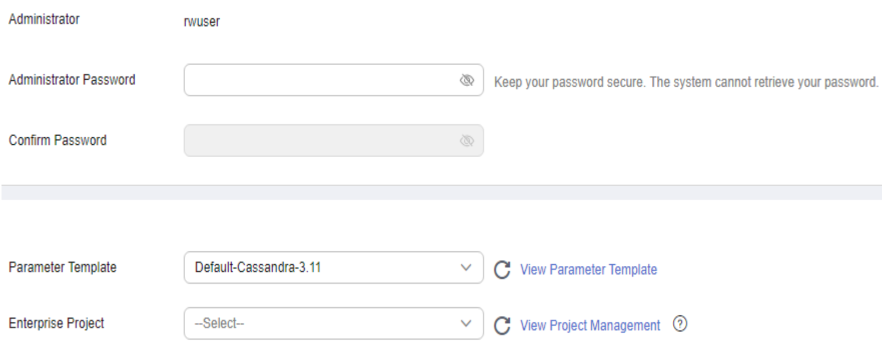


**Table 4-5** Database configuration

| Parameter | Description |
| --- | --- |
| Administrator | Username of the administrator account. The default value is **rwuser**. |
| Administrator Password | Password of the administrator account. The password:<br>● Can include 8 to 32 characters.<br>● Can include uppercase letters, lowercase letters, digits, and any of the following special characters: ~!@#%^*-_=+?<br>● For security reasons, set a strong password. The system will verify the password strength.<br>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.<br>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.<br>An enterprise project groups cloud resources, so you can manage resources and members by project. The default project is **default**.<br>Select an enterprise project from the drop-down list. For more information about enterprise projects, see ***Enterprise Management User Guide***. |

**Figure 4-7** Tag configuration

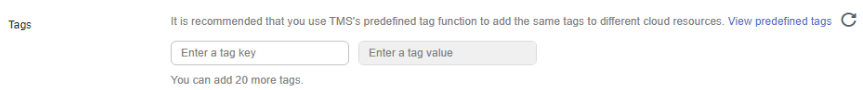


**Table 4-6** Tags

| Parameter | Description |
| --- | --- |
| Tags | This setting is optional. Adding tags helps you better identify and manage your GeminiDB Cassandra instances.<br>A maximum of 20 tags can be added for each instance.<br>If your organization has configured tag policies for GeminiDB Cassandra, you need to add tags to instances based on the tag policies. If a tag does not comply with the policies, an instance may fail to be created. Contact your organization administrator to learn more about tag policies.<br>A tag consists of a tag key and a tag value.<br>● A tag key is mandatory if the instance will be tagged. Each tag key is unique for each instance. It can contain 1 to 128 characters, cannot start with **_sys_**, and cannot start or end with a space. Only letters, digits, spaces, and the following special characters are allowed: -_@.:/+=<br>● A tag value is optional if the instance will be tagged. The value can be empty.<br>The value can contain a maximum of 255 characters. Only letters, digits, spaces, and the following special characters are allowed: _.-:+=@/<br>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 4-8** Required duration configuration


**Table 4-7** 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.<br>● If you select this option, the instance is automatically renewed based on the subscription duration. |

**Step 5** 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 6** 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**

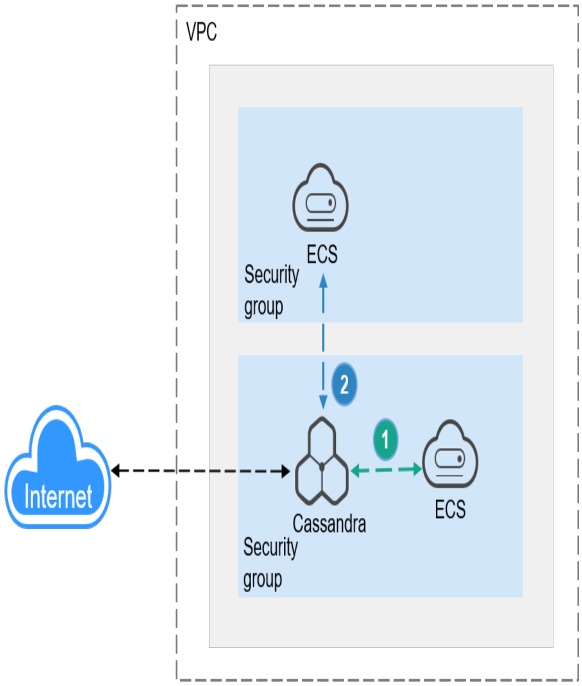
# 4.3 Instance Connection and Management

## 4.3.1 Connecting to a GeminiDB Cassandra Instance

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

**Figure 4-9** shows the process of connecting to a GeminiDB Cassandra instance.

**Figure 4-9** Connection Methods




1 A GeminiDB Cassandra instance is connected over a private network (An ECS and a GeminiDB Cassandra instance are in the same security group).

2 A GeminiDB Cassandra instance is connected over a private network (An ECS and a GeminiDB Cassandra instance are in different security groups).

**Table 4-8** 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<br>• 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. |

<table>
<tr><th>Method</th><th>Scenario</th><th>Default Port</th><th>Description</th></tr>
<tr><td>Private network</td><td>Private IP addresses are provided by default.<br>Your applications are deployed on an ECS that is in the same region and VPC as your instances.</td><td>8635</td><td><ul><li>High security and performance</li><li>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.</li><li>If they are in different security groups, configure security group rules for them, separately.<ul><li>Configure inbound rules of a security group for the GeminiDB Cassandra instance by following Setting Security Group Rules for a GeminiDB Cassandra Instance.</li><li>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.</li></ul></li></ul></td></tr>
<tr><td>Public network</td><td>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.</td><td>8635</td><td><ul><li>Low security</li><li>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.</li><li>For EIP pricing details, see EIP billing details.</li></ul></td></tr>
<tr><td>Java</td><td>An example of connecting to an instance using Java is provided.</td><td>8635</td><td>-</td></tr>
<tr><td>Go</td><td>An example of connecting to an instance using Go is provided.</td><td>8635</td><td>-</td></tr>
<tr><td>Spark</td><td>An example of connecting to an instance using Spark is provided.</td><td>8635</td><td>-</td></tr>
</table>

## 4.3.2 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.

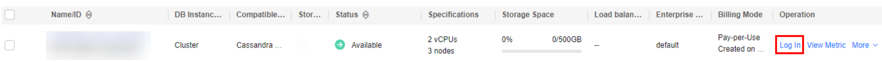
### Procedure

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

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

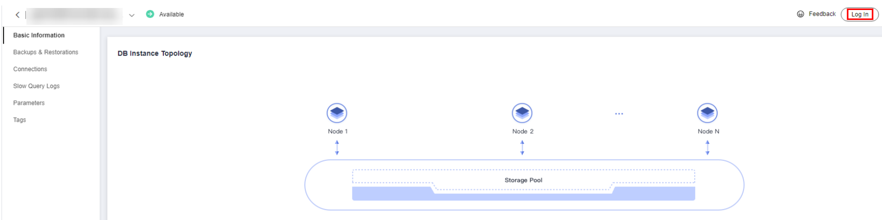
**Step 3** On the **Instance Management** page, locate the target DB instance and click **Log In** in the **Operation** column.

**Figure 4-10** 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 4-11** Logging in to the database



**Step 4** On the displayed login page, enter the administrator username and password and click **Log In**.

For details about how to manage databases through DAS, see **GeminiDB Cassandra Data Management**.

**----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.

# 4.3.3 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.

GeminiDB Cassandra API allows you to connect to an instance over SSL or non-SSL connections. SSL encrypts data and is more secure.

## 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**.
   - If the CPU type is Kunpeng, download the **Cassandra client**.
4. Before connecting to an instance over SSL, obtain an SSL certificate. For details, see **Downloading the SSL Certificate**.

## Non-SSL Connection

**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 ***

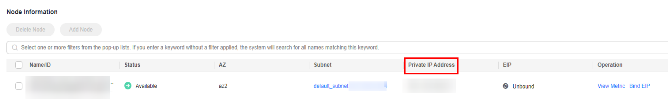
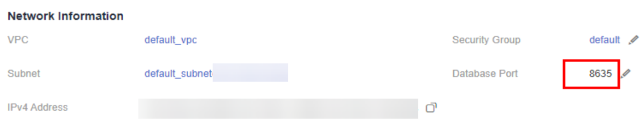
**Step 5** Connect to the DB instance in the directory where the cqlsh tool is located.

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

Example:

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

**Table 4-9** Description

<table>
<tr><th>Parameter</th><th>Description</th></tr>
<tr><td><DB_HOST></td><td>
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 4-12 Viewing the private IP address

</td></tr>
<tr><td><DB_PORT></td><td>
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 4-13 Viewing the port number


</td></tr>
<tr><td><DB_USER></td><td>Database account. The default value is rwuser.</td></tr>
</table>

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

```
rwuser@cqlsh>
```

**----End**

## SSL

**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** Upload the SSL certificate to the ECS.

**Step 4** 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 5** Run the following command to grant the execute permission on all files:

**chmod +x ***

**Step 6** Connect to the DB instance in the directory where the cqlsh tool is located.

export SSL_CERTFILE=/*<PATH_OF_SSL_CERT_FILE>*

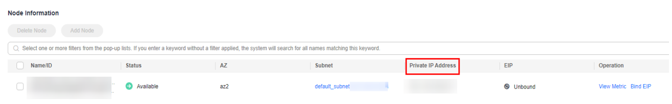
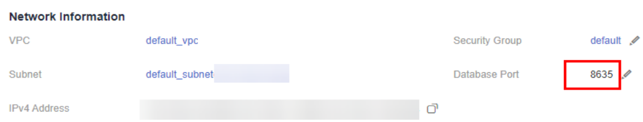
export SSL_VERSION=TLSv1_2

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

Example:

**./cqlsh 192.168.1.8 8635 --ssl -u rwuser**

**Table 4-10** Description

<table>
<tr><th>Parameter</th><th>Description</th></tr>
<tr><td><PATH_OF_SSL_CERT_FILE></td><td>SSL file path.</td></tr>
<tr><td><DB_HOST></td><td>Private IP address of an instance to be connected.<br>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.<br>If the instance you purchased has multiple nodes, select the private IP address of any node.<br>Figure 4-14 Viewing the private IP address<br>
</td></tr>
<tr><td><DB_PORT></td><td>Port number of the instance to be connected. The default port number is 8635. Replace it with the actual port number.<br>Click the instance name to go to the Basic Information page and obtain the port number in the Network Information area.<br>Figure 4-15 Viewing the port number<br>
</td></tr>
<tr><td><DB_USER></td><td>Database account. The default value is rwuser.</td></tr>
</table>

**Step 7** Check the results. If the following information is displayed, the connection is successful.

```
rwuser@cqlsh>
```

**----End**

## Follow-up Operations

After logging in to the instance, you can create keyspaces, databases, or tables. For details, see **Buying and Connecting to a GeminiDB Cassandra Instance**.

# 4.3.4 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.

You can also establish a common or an SSL connection.

## 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**.
   - If the CPU type is Kunpeng, download the **Cassandra client**.
4. Before connecting to an instance over SSL, obtain an SSL certificate. For details, see **Downloading the SSL Certificate**.

## Non-SSL Connection

**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 ***

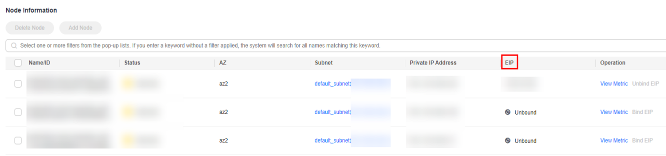
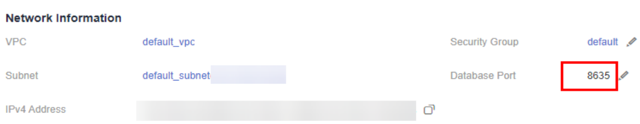
**Step 5** Connect to the DB instance in the directory where the cqlsh tool is located.

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

Example:

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

**Table 4-11** Description

| Parameter | Description |
|---|---|
| *<DB_HOST>* | EIP bound to the instance to be connected.<br>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.<br>If the instance you purchased has multiple nodes, select the EIP of any node.<br>**Figure 4-16** Viewing the EIP<br><br>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.<br>Click the instance name to go to the **Basic Information** page and obtain the port number in the **Network Information** area.<br>**Figure 4-17** Viewing the port number<br> |
| *<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**

## SSL Connection

**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** Upload the SSL certificate to the ECS.

**Step 4** 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 5** Run the following command to grant the execute permission on all files:

**chmod +x ***

**Step 6** Connect to the DB instance in the directory where the cqlsh tool is located.

export SSL_CERTFILE=/<*PATH_OF_SSL_CERT_FILE*>

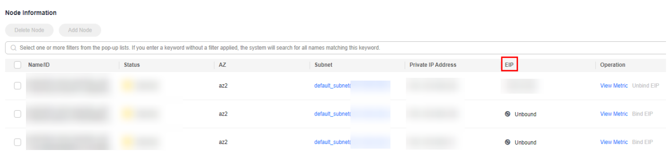
export SSL_VERSION=TLSv1_2

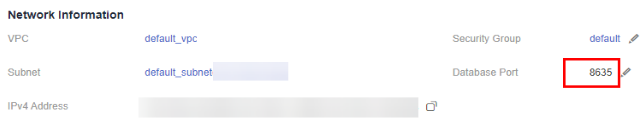
**./cqlsh** <*DB_HOST*> <*DB_PORT*> **--ssl -u** <*DB_USER*>

Example:

**./cqlsh 192.168.1.8 8635 --ssl -u rwuser**

**Table 4-12** Description

| Parameter | Description |
| --- | --- |
| <*PATH_OF_SSL_CERT_FILE*> | SSL file path |
| <*DB_HOST*> | EIP bound to the instance to be connected.<br>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.<br>If the instance you purchased has multiple nodes, select the EIP of any node.<br>**Figure 4-18** Viewing the EIP<br><br>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. |

| Parameter | Description |
| --- | --- |
| *<DB_PORT>* | Port number of the instance to be connected. The default port number is 8635. Replace it with the actual port number.<br>Click the instance name to go to the **Basic Information** page and obtain the port number in the **Network Information** area.<br>**Figure 4-19** Viewing the port number<br> |
| *<DB_USER>* | Database account. The default value is **rwuser**. |

**Step 7** Check the results. If the following information is displayed, the connection is successful.

```
rwuser@cqlsh>
```

**----End**

## Follow-up Operations

After logging in to the instance, you can create keyspaces, databases, or tables. For details, see **Buying and Connecting to a GeminiDB Cassandra Instance**.

# 4.3.5 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 *Getting Started with Elastic Cloud Server*.

**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**

## 4.3.6 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 *Getting Started with Elastic Cloud Server*.

**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()
}()
```

## 4.3.7 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 *Getting Started with Elastic Cloud Server*.

**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**

## 4.3.8 Connection Information Management

### 4.3.8.1 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 4-13**.

**Table 4-13** 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:<br>● If yes, no security group rules need to be configured.<br>● If no, configure security group rules for them, respectively.<br>– GeminiDB Cassandra instance: Configure inbound rules for its security group. For details, see **Procedure**.<br>– 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 Huawei Cloud console.

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** On the **Instances** page, locate the instance that you want to configure security group rules for and click its name.

**Step 4** Configure security group rules.

**Method 1**

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

**Figure 4-20** Security group

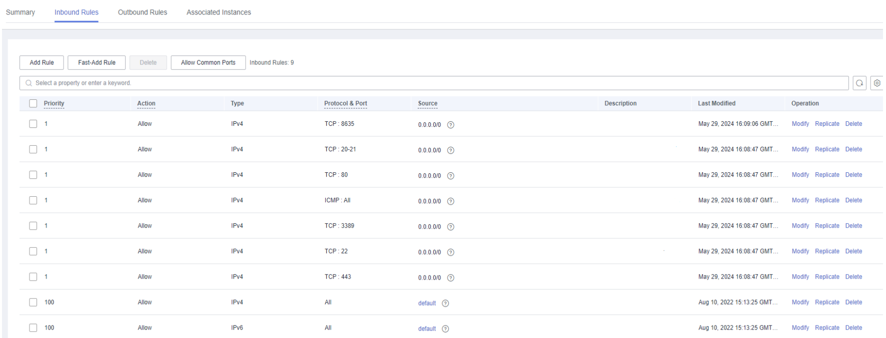
**Method 2**

On the **Basic Information** page, choose **Connections** in the navigation pane on the left. In the **Security Group** area on the right, click the name of the security group. The **Security Group** page is displayed.


**Step 5** Add an inbound rule.

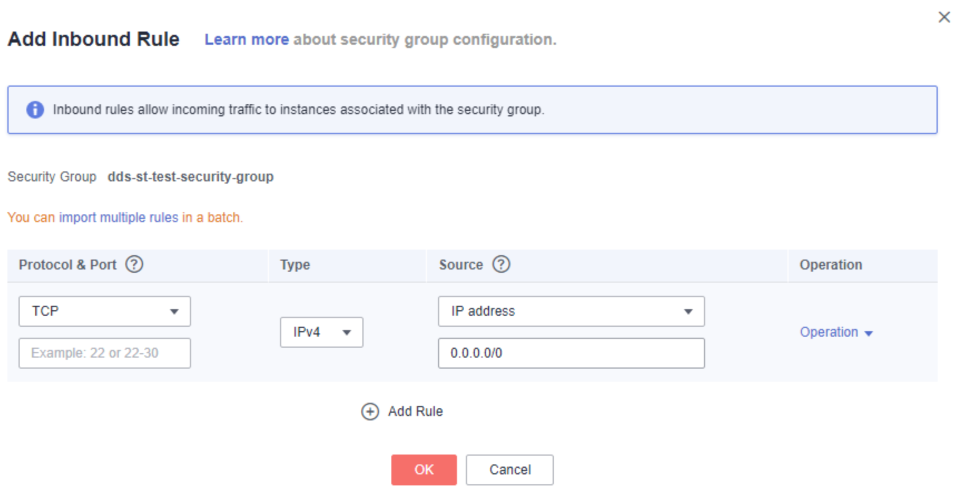
1. Click the **Inbound Rules** tab.

**Figure 4-21** Inbound rules



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

**Figure 4-22** Adding a rule




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

**Table 4-14** Inbound rule settings

| Parameter | Description | Example Value |
| --- | --- | --- |
| Protocol & Port | – Network protocol. Currently, GeminiDB Cassandra instances can be accessed only over TCP.<br>– **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.<br>– IPv4<br>– 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:<br>– Single IP address: xxx.xxx.xxx.xxx/32 (IPv4)<br>– Subnet: xxx.xxx.xxx.0/24<br>– All IP addresses: 0.0.0.0/0<br>– sg-abc (security group) | 0.0.0.0/0 |
| Description | (Optional) Provides supplementary information about the security group rule.<br>The description can contain up to 255 characters and cannot contain angle brackets (<>). | - |

**Step 6** Click **OK**.

**----End**

## 4.3.8.2 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

- To change the EIP that has been bound to a node, unbind it from the node first.

### Binding an EIP

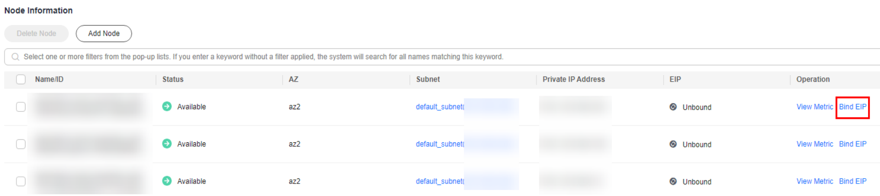
**Step 1** Log in to the Huawei Cloud console.

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** On the **Instances** page, locate the GeminiDB Cassandra instance that you want to bind an EIP to and click its name.

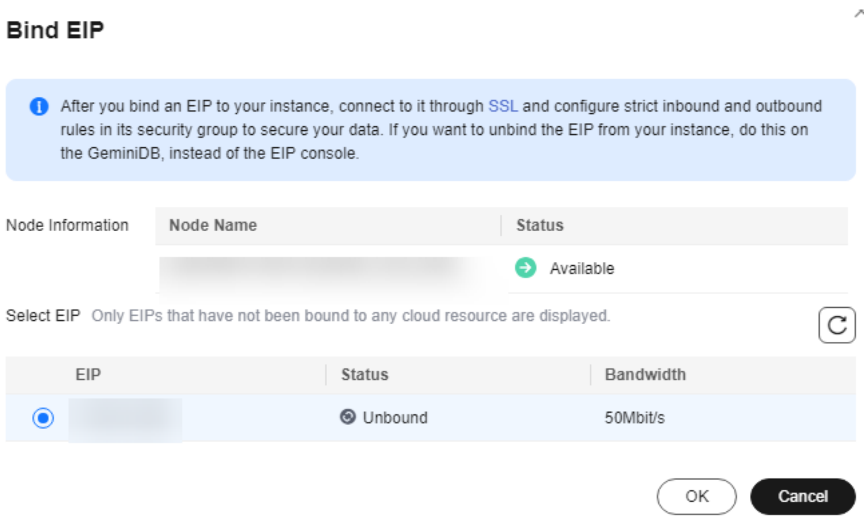
**Step 4** On the **Basic Information** page, in the **Node Information** area, locate the target node and click **Bind EIP** in the **Operation** column.

**Figure 4-23** Binding an EIP




**Step 5** 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 4-24** Selecting an EIP




**Step 6** 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

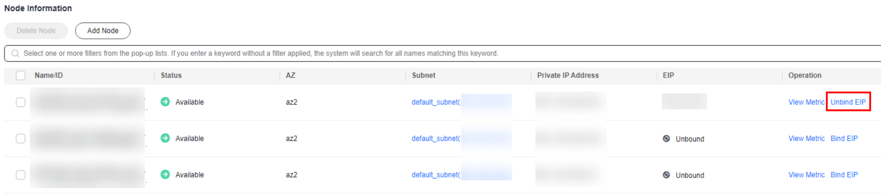
**Step 1** Log in to the Huawei Cloud console.

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** On the **Instances** page, locate the GeminiDB Cassandra instance that you want to unbind an EIP from and click its name.

**Step 4** On the **Basic Information** page, in the **Node Information** area, locate the target node and click **Unbind EIP** in the **Operation** column.

**Figure 4-25** Unbinding an EIP




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

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

**----End**

## 4.3.8.3 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.

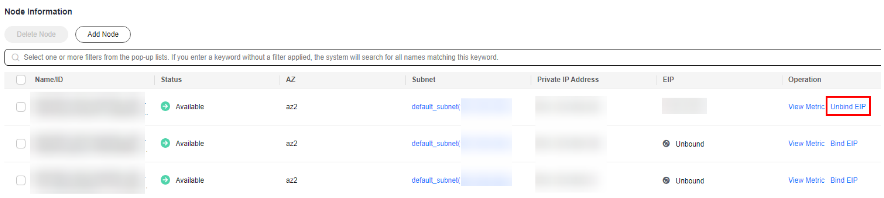
**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** On the **Instances** page, locate the instance whose IP address and port you want to view and click its name.

**Method 1**

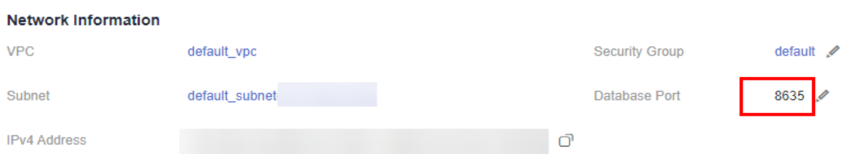
In the **Node Information** area on the **Basic Information** page, view the private IP address or EIP of each node in the instance.

**Figure 4-26** Obtaining IP addresses




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

**Figure 4-27** Viewing the port number



**Method 2**

In the navigation pane on the left, click **Connections** to view private IP addresses, EIPs, and port number of the instance.

**Figure 4-28** Viewing the IP addresses and port number


**----End**

## 4.3.8.4 Changing the Port of a GeminiDB Cassandra Instance

### Scenarios

GeminiDB Cassandra API allows you to change the database port of an instance to ensure security.

The database port cannot be changed when the instance is in any of the following statuses:

- Frozen
- Restarting
- Adding nodes
- Changing specifications
- Scaling storage space
- Deleting nodes

### Procedure

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

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** On the **Instances** page, locate the instance whose database port you want to change and click its name.

**Step 4** In the navigation pane on the left, choose **Connections**.

**Step 5** In the **Basic Information** area, click to the right of the **Database Port** field.

The database port number can range from 2100 to 9500 but cannot be 2180, 2887, 3887, 7000, 7001, 7199, 8018, 8079, 8091, 8092, 8479, 8484, 8636, and 8999.

- To submit the change, click ✓. This operation takes about 1 to 5 minutes.
- To cancel the change, click ✗.

**Step 6** View the change result at the **Basic Information** area.

**----End**

### 4.3.8.5 Changing the Security Group of a GeminiDB Cassandra Instance

You can change the security group of your GeminiDB Cassandra instance.

## Usage Notes

If you are adding nodes to an instance, the security group of the instance cannot be changed.

## Procedure

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

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** On the **Instances** page, locate the instance whose security group you want to change and click its name.

**Step 4** In the navigation pane on the left, choose **Connections**.

**Step 5** In the **Security Group** area, click ✎ beside the security group name and select the required security group.

- To submit the change, click ✓. This process takes about 1 to 3 minutes.
- To cancel the change, click ✗.

**Step 6** View the change result.

**----End**

### 4.3.8.6 Encrypting Data over SSL for a GeminiDB Cassandra Instance

Secure Socket Layer (SSL) is an encryption-based Internet security protocol for establishing an encrypted link between a server and a client. It provides privacy, authentication, and integrity to Internet communications.

- Authenticates users and servers, ensuring that data is sent to the correct clients and servers.
- Encrypts data to prevent it from being intercepted during transfer.
- Ensures data integrity during transmission.

After SSL is enabled, you can establish an encrypted connection between your client and the instance you want to access to improve data security.

## Usage Notes

- After you enable or disable SSL, the established connection is interrupted. Restart the instance to apply the change.
- Enabling SSL will prolong network connection response and increase CPU usage. So, evaluate impacts on service performance before enabling SSL.
- The SSL function provided by GeminiDB Cassandra supports only TLS 1.3 or later.

## Enabling SSL

**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 target instance.

**Step 4** In the **DB Information** area, click to enable SSL.

**Figure 4-29** Enabling SSL


Alternatively, choose **Connections** in the navigation pane on the left. In the **Basic Information** area, click in the **SSL** field to enable SSL.

**Figure 4-30** Enabling SSL


After SSL is enabled, you can connect to the GeminiDB Cassandra instance through SSL. For details, see **SSL**.

**----End**

## Disabling SSL

**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 target instance.

**Step 4** In the **DB Information** area, click to disable SSL.

**Figure 4-31** Disabling SSL


Alternatively, choose **Connections** in the navigation pane on the left. In the **Basic Information** area, click  in the **SSL** field to disable SSL.

**Figure 4-32** Disabling SSL


After SSL is disabled, you can connect to the GeminiDB Cassandra instance over a non-SSL connection. For details, see **Non-SSL Connection**.

**----End**

## 4.3.8.7 Downloading the SSL Certificate

Secure Sockets Layer (SSL) certificates set up encrypted connections between clients and servers, preventing data from being tampered with or stolen during transmission.

To improve data security, you can enable SSL when creating an instance. After the instance is created, you can connect to it using an SSL certificate.

This section describes how to obtain an SSL certificate.

## Procedure

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

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** On the **Instances** page, locate the instance that you want to connect to and click its name.

**Step 4** In the **DB Information** area, click  in the **SSL** field to download the SSL certificate.

**Figure 4-33** Downloading the SSL certificate

**----End**

# 4.4 Data Migration

## 4.4.1 Migration Solution

This section describes how to migrate services to a GeminiDB Cassandra instance. If you have any questions about the migration, choose **Service Tickets > Create Service Ticket** in the upper right corner of the console to get technical support.

### Migration Tool

- DRS is used for full and incremental data migration while ensuring data security. For details, see **Migration Overview**.

### Permissions

Ensure that the database port is enabled in the security group of the GeminiDB Cassandra instance.

### Migration Scenarios

**Table 4-15** Migration scenarios

| No. | Source | Destination | Reference |
| --- | --- | --- | --- |
| 1 | Open-Source Cassandra | GeminiDB Cassandra | **Using DRS to Migrate Data from Open-Source Cassandra to GeminiDB Cassandra API** |
| 2 | DynamoDB (web service) on other clouds | GeminiDB Cassandra | **Using DRS to Migrate Data from DynamoDB (Web Service) on Other Clouds to GeminiDB Cassandra API** |

## 4.4.2 Using DRS to Migrate Data from Open-Source Cassandra to GeminiDB Cassandra API

DRS is used for full and incremental data migration while ensuring data security. For details, see **Migration Overview**.

For details about how to use DRS to migrate data from open-source Cassandra to GeminiDB Cassandra API, see **From Open-Source Cassandra to GeminiDB Cassandra API**.

## 4.4.3 Using DRS to Migrate Data from DynamoDB (Web Service) on Other Clouds to GeminiDB Cassandra API

DRS is used for full and incremental data migration while ensuring data security. For details, see **Migration Overview**.

For details about how to use DRS to migrate data from DynamoDB (web service) on other clouds to GeminiDB Cassandra API, see **From DynamoDB to GeminiDB Cassandra API**.

## 4.4.4 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 an Instance over a Private Network**.

### COPY Syntax

- **COPY TO**

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

- **COPY FROM**

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

**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 4-16**.

**Table 4-16** 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. |
| @*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 4-17** 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.<br>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 () | - |
| TO/FROM | DATETIMEFORMAT | Time format for reading or writing CSV time data.<br>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.<br>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 | - |

| Command | Parameter | Description | Default Value | Applicability |
| --- | --- | --- | --- | --- |
| 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.<br>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.<br>**NOTE**<br>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.<br>● If the open-source Cassandra is used, the value is **True**.<br>● 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.<br>● MAXBATCHSIZE x The size of a single row < batch_size_fail_threshold_in_kb.<br>● If the batch size is too large, an alarm will be reported and rejected.<br>● Set the following parameters in **cassandra.yaml**: batch_size_warn_threshold_in_kb (The current value is 5.)<br>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.<br>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 | - |

| Command | Parameter | Description | Default Value | Applicability |
| --- | --- | --- | --- | --- |
| 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**.<br>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 (for example, Huawei Cloud obsfs is used), you can increase the value. |

| 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.<br>● For a large page size or a large partition, increase the value of this parameter.<br>● If a timeout occurs, increase the value of this parameter.<br>● 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.<br>Records with smaller tokens will not be exported.<br>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.<br>Records with larger tokens will not be exported.<br>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.<br>Total degree of parallelism during data export = Number of working processes x Value of this parameter.<br>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.<br>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.<br>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 4-34** 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 4-35** 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 4-36** 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 4-37** 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.

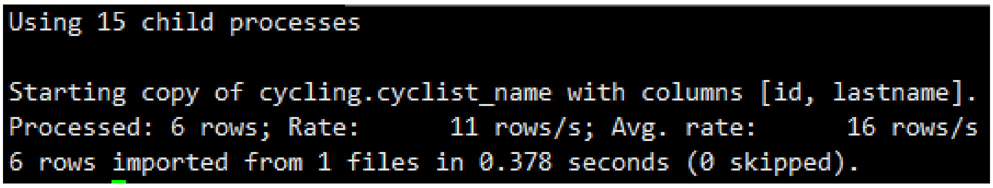
```
SELECT * FROM cycling.cyclist_name;
```

**Figure 4-38** Import succeeded


7. Import the **cyclist_lastname.csv** file.

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

**Figure 4-39** Importing data




8. Check whether the data is updated.

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

The query result is displayed,

**Figure 4-40** Import succeeded


**----End**

## Helpful Links

**What Can I Do if Error "field larger than field limit (131072)" Is Reported During Data Import?**

# 4.5 Instance Lifecycle Management

## 4.5.1 Restarting GeminiDB Cassandra 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** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** 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 4** 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 5** In the displayed dialog box, click **Yes**.

**----End**

## 4.5.2 Exporting Instance Information

### Scenarios

You can export information about all or selected instances to view and analyze instance information.

### Exporting Information About All Instances

**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 in the upper right corner. By default, information about all instances are exported. In the displayed dialog box, you can select the items to be exported and click **Export**.

**Step 4** After the export task is complete, check an XLS file on your local PC.

**----End**

## Exporting Information About Selected Instances

**Step 1** On the **Instances** page, select the target instances or search for required instances by project, compatible API, name, ID, or tag and click in the upper right corner. In the displayed dialog box, select the items to be exported and click **Export**.

**Step 2** After the export task is complete, check an XLS file on your local PC.

**----End**

# 4.5.3 Deleting a Pay-per-Use 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 **How Do I Unsubscribe 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** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** On the **Instances** page, locate the instance you want to delete and choose **More** > **Delete** in the **Operation** column.

**Step 4** 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 5** In the displayed dialog box, click **Yes**.

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

**----End**

## 4.5.4 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.

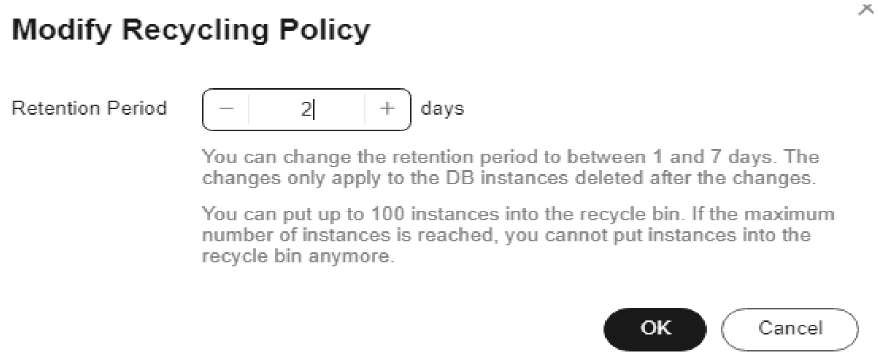
### Modifying the Recycling Policy

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

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** 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 4-41** Modifying a recycling policy



**----End**

## Rebuilding an Instance

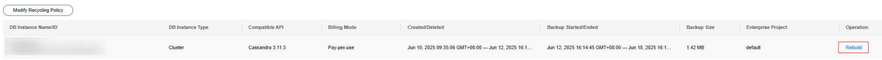
You can rebuild DB instances from the recycle bin within the retention period to restore data.

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

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** On the **Recycling Bin** page, locate the target instance and click **Rebuild** in the **Operation** column.

**Figure 4-42** Rebuilding an instance



**Step 4** On the displayed page, configure required parameters and submit the task.

**----End**

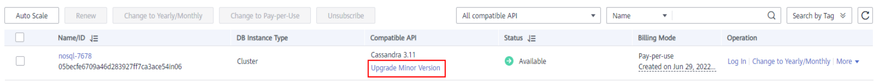
# 4.6 Instance Modifications

## 4.6.1 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 4-43** 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.

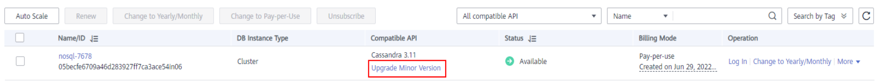
## Procedure

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

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** On the **Instances** page, locate the instance you want to upgrade and click **Upgrade Minor Version** in the **Compatible API** column.

**Figure 4-44** Patch installation

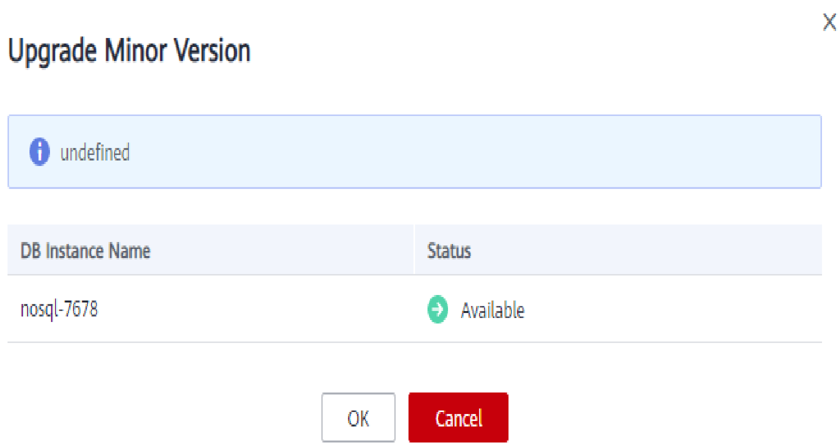


Alternatively, click the instance name to go to the **Basic Information** page. In the **DB Information** area, click **Upgrade Minor Version** in the **Compatible API** field.

**Figure 4-45** Patch installation


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

**Figure 4-46** Confirming dialog box



**Step 5** 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**

## 4.6.2 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** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** 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 4** View the results on the **Instances** page.

**----End**

### Method 2

**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 target instance. The **Basic Information** page is displayed.

**Step 4** 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 5** View the results on the **Instance Management** page.

**----End**

## 4.6.3 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.
- For two instances with an intra-region DR or cross-region dual-active relationship, make sure that they have the same administrator passwords.
- 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** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** On the **Instances** page, locate the instance whose password you want to reset and choose **More** > **Reset Password** in the **Operation** column.

**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**

## Method 2

**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 instance whose password you want to reset to go to the **Basic Information** page.

**Step 4** In the **DB Information** area, click **Reset Password** in the **Administrator** field.

**Step 5** 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 6** 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**

# 4.6.4 Changing vCPUs and Memory

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

## Usage Notes

- Instances can be scaled up or down by changing their specifications.
- 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** In the service list, choose **Databases** > **GeminiDB**.

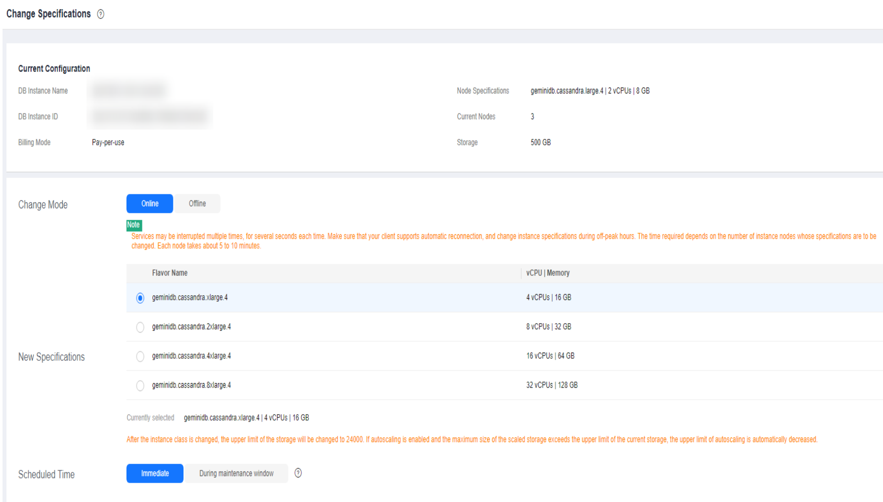
**Step 3** On the **Instances** page, locate the instance whose vCPUs and memory you want to change and click its name.

**Step 4** In the in **DB Information** area on the **Basic Information** page, click **Change** next to **Specifications**.

**Figure 4-47** Changing specifications


**Step 5** On the displayed page, select the required specifications and click **Next**.

**Figure 4-48** Changing specifications




**Step 6** 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 7** View the change results.

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

**----End**

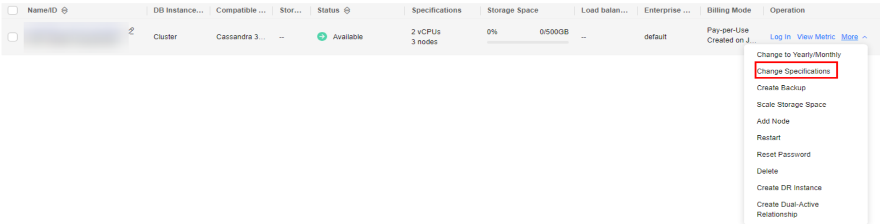
## Method 2

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

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

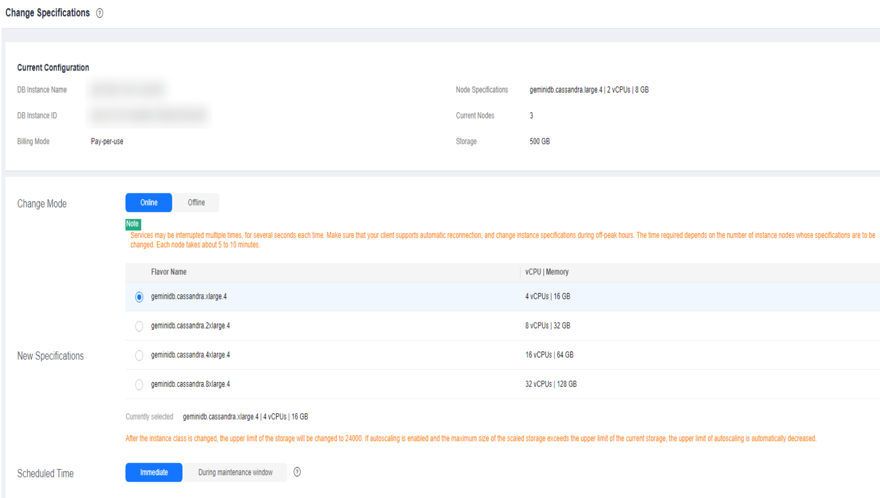
**Step 3** On the **Instances** page, locate the instance whose specifications you want to change and choose **More** > **Change Specifications** in the **Operation** column.

**Figure 4-49** Changing specifications




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

**Figure 4-50** 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**

## 4.6.5 Setting a Maintenance Window

The default maintenance window is 10:00–14:00 (GMT+08:00) but you can change it if needed. To prevent service interruption, set the maintenance window to off-peak hours. Before calling this API:

### Usage Notes

- You can configure a maintenance window only for restarting a DB instance, changing an instance class, or upgrading the minor version of a DB instance.
- The specification change and patch upgrade that have been performed during the maintenance period cannot be performed immediately. The instance can be restarted immediately.
- You can cancel a task to be executed.
- Changing the maintenance window will not affect the timing that has already been scheduled.
- The maintenance window cannot overlap the time window configured for backups. Otherwise, scheduled tasks may fail.
- During the maintenance window, the scheduled task is scanned and executed every 10 minutes. If the task is delivered near the end of the maintenance period, the task may fail to be scanned and the execution is canceled.

### Setting a Maintenance Window

**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 target instance. The **Basic Information** page is displayed.

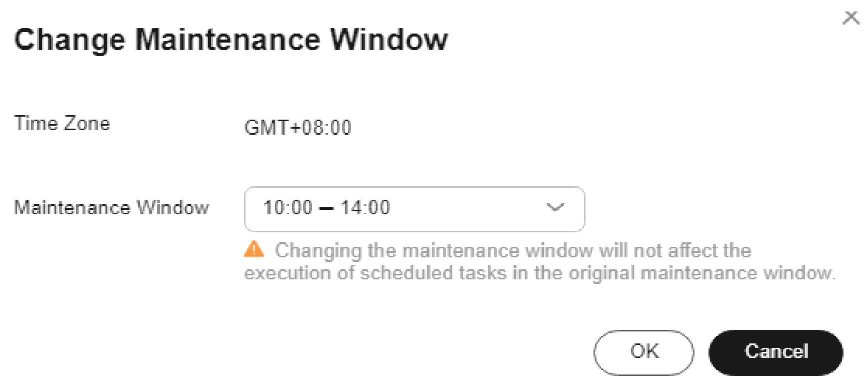
**Step 4** In the **DB Information** area, locate **Maintenance Window** and click **Change**.

**Figure 4-51** The change button


**Step 5** On the **Change Maintainable Window** page, select the maintenance time period as needed, and then click **OK**.

Supported time periods: 02:00-06:00, 06:00-10:00, 10:00-14:00, 14:00-18:00, 18:00-22:00, and 22:00-02:00

**Figure 4-52** Changing a maintenance window




**Step 6** Check the result.

On the **Basic Information** page, you can view the changed maintenance window.

**----End**

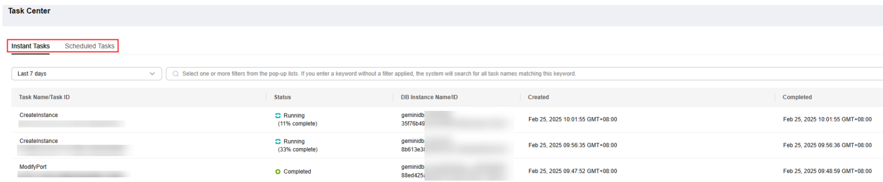
## Querying an Executed Task

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

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** On the **Task Center** page, click the **Instant Tasks** or **Scheduled Tasks** tab to view a task.

**Figure 4-53** Querying a task



**----End**

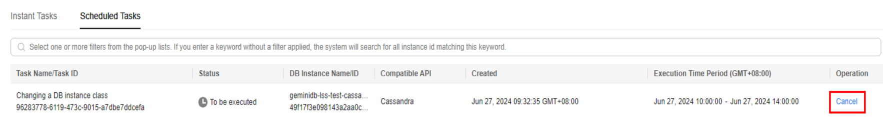
## Canceling a Scheduled Task

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

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** On the **Task Center** page, locate a scheduled task, and click **Cancel** in the **Operation** column.

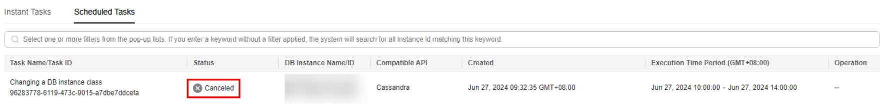
**Figure 4-54** Canceling a task



**Step 4** Check the result.

On the **Task Center** page, you can view the result. After the task is cancelled, its status changes to **Cancelled**.

**Figure 4-55** Checking cancelled tasks




**----End**

# 4.6.6 Adding and Deleting Instance Nodes

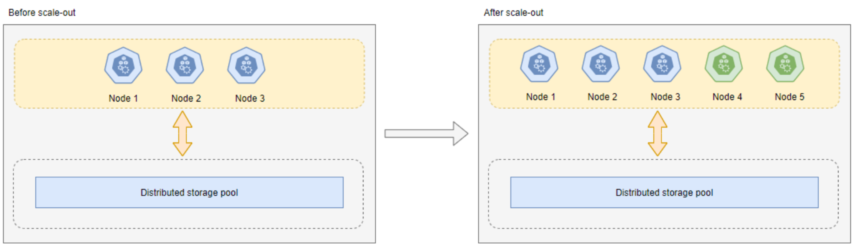
## 4.6.6.1 Overview

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**.

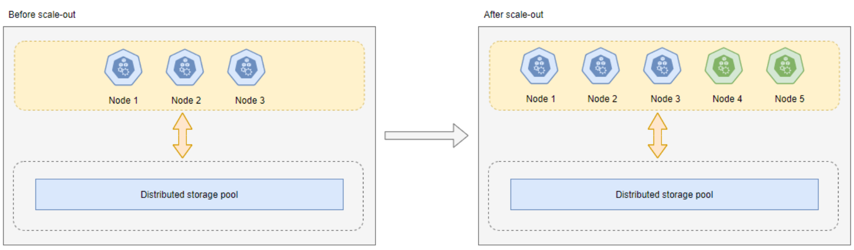
**Figure 4-56** Adding instance nodes




### Automatically 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 **Automatically Adding Instance Nodes**.

**Figure 4-57** Adding instance nodes

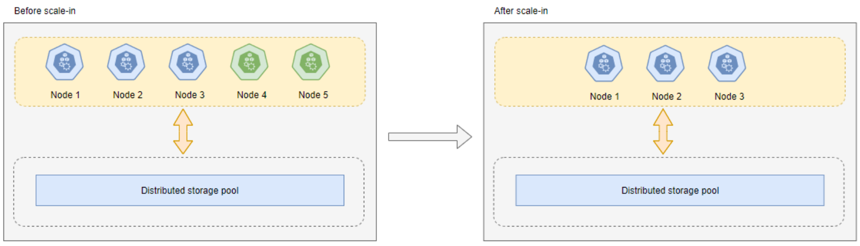


## Manually Deleting Instance Nodes

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 4-58** Deleting instance shards




### 4.6.6.2 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

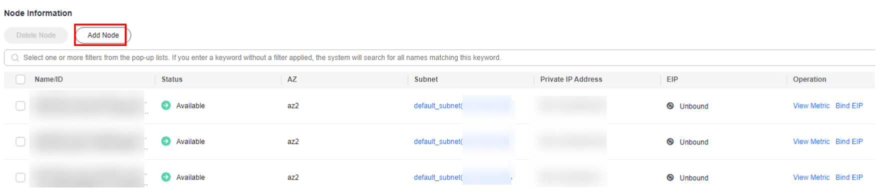
**Step 1** Log in to the Huawei Cloud console.

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** On the **Instances** page, locate the instance that you want to add nodes to and click its name.

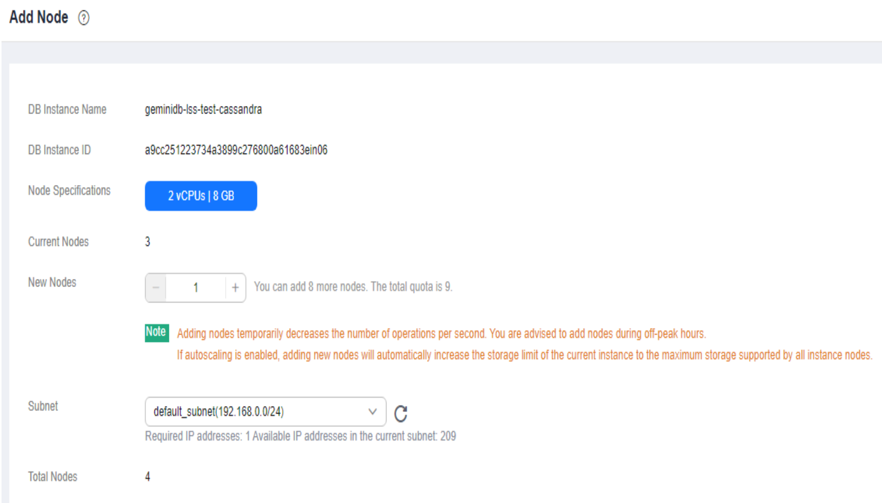
**Step 4** In the **Node Information** area on the **Basic Information** page, click **Add Node**.

**Figure 4-59** Node information



**Step 5** Specify **Add Nodes** and click **Next**.

**Figure 4-60** 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 6** 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 7** 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**

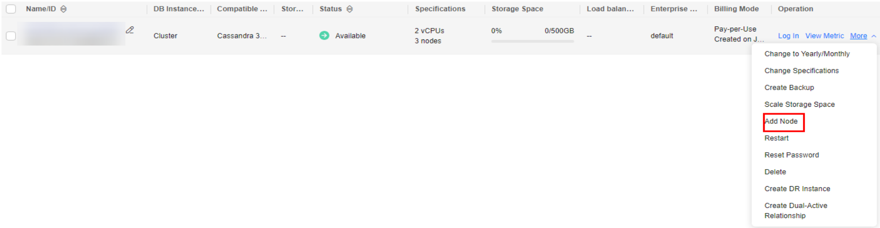
## Method 2

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

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

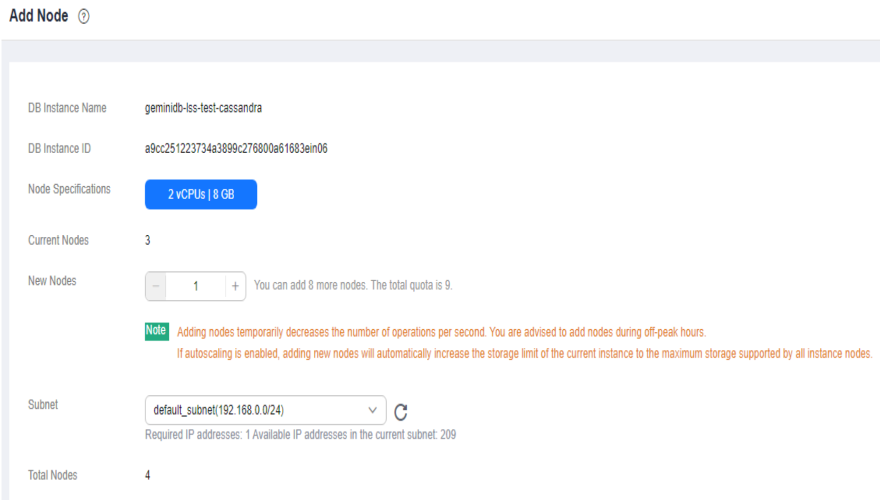
**Step 3** On the **Instances** page, locate the target instance and choose **More** > **Add Node** in the **Operation** column.

**Figure 4-61** Adding nodes



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

**Figure 4-62** 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**

### 4.6.6.3 Automatically Adding Instance Nodes

When an autoscaling threshold is met, GeminiDB Cassandra instance nodes can be automatically added to reduce the server load or I/O overhead.

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 you enable storage autoscaling:

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**.

   Custom policy in JSON format:
```
{
    "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.

   Log in to the IAM console using a Huawei Cloud account or as an IAM user, locate the IAM user that the target instance belongs to, and add it to the user group created in **3**. The IAM user will inherit permissions of the user group.

## 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.

- If resources in the current region are insufficient, nodes may fail to be added.
- Autoscaling is available only when your account balance is sufficient.
- The instance must be in the **Available** status.
- Once autoscaling is enabled, an agency will be created and fees will be automatically deducted.

## Procedure

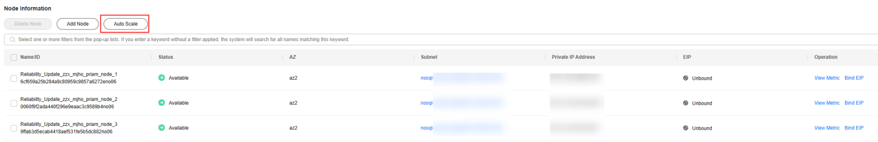
**Step 1** Log in to the Huawei Cloud console.

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** On the **Instances** page, click the target instance. The **Basic Information** page is displayed.

**Step 4** In the **Node Information** area on the **Basic Information** page, click **Auto Scale**.

**Figure 4-63** Auto Scale



**Step 5** Set autoscaling triggers and thresholds listed in **Table 4-18**.

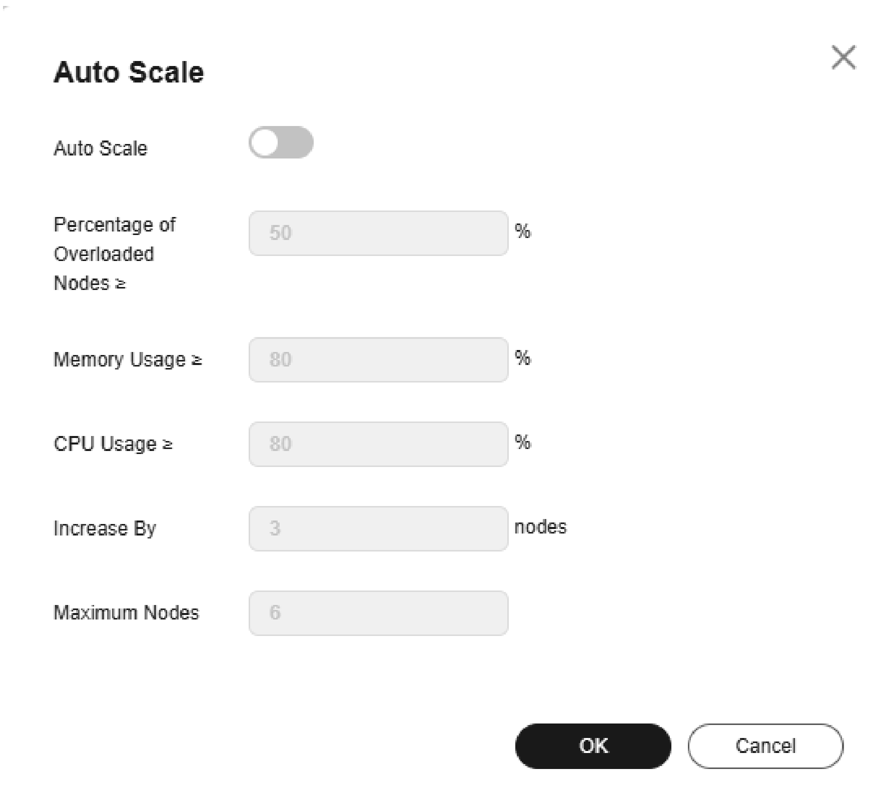


**Table 4-18** Description

| Parameter | Description |
| --- | --- |
| Auto Scale | Whether autoscaling is enabled |
| Percentage of Overloaded Nodes ≥ | Percentage of overloaded nodes that need to be scaled out |
| Memory Usage ≥ | Memory usage of nodes for which autoscaling is triggered |
| CPU Usage ≥ | CPU usage of nodes for which autoscaling is triggered |
| Increase By | Number of nodes to be added each time |
| Maximum Nodes | Maximum number of nodes that can be automatically added |

**----End**

### 4.6.6.4 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.
- 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** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** On the **Instances** page, locate the instance that you want to delete nodes from and click its name.

**Step 4** In the **Node Information** area on the **Basic Information** page, locate the target node and click **Delete** in the **Operation** column.

**Step 5** 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 6** 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**

# 4.6.7 Scaling Storage Space

## 4.6.7.1 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. As data volumes decrease, you can scale down storage space to avoid low database node utilization and resource waste. **Table 4-19** lists the scaling methods supported by GeminiDB Cassandra instances.

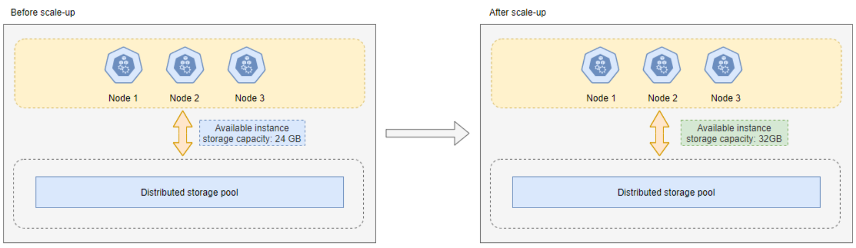
**Table 4-19** Scaling methods

| Method | Description |
| --- | --- |
| **Manually Scaling Up Storage Space** | You can specify how much storage space needs to be added.<br>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.<br>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 Down Storage Space** | You can specify how much storage space needs to be reduced.<br>The storage space to be reduced must be an integer multiple of 1 GB and greater than or equal to 125% of the used storage space. The value is rounded up. |

## 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.

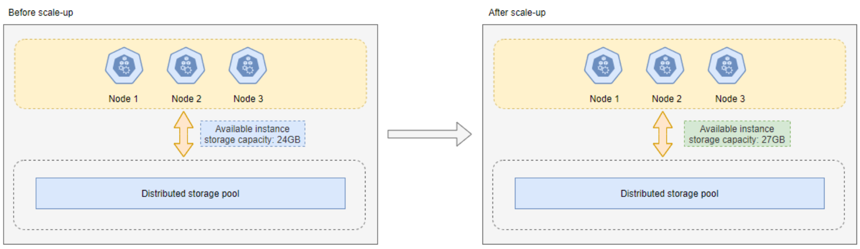
**Figure 4-64** Manually scaling up storage space



## 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 4-65** Automatically scaling up storage space




## Manually Scaling Down Storage Space

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

**Figure 4-66** Manually scaling down storage space


## 4.6.7.2 Manually Scaling Up Storage Space

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

## Usage Notes

- **Scaling up storage does not interrupt your services. After storage scale-up is complete, you do not need to restart your instance.**
- If your yearly/monthly instance is running out of storage, additional usage will be billed on a pay-per-use basis. To avoid these extra costs and maintain the benefits of your yearly/monthly subscription, you are advised to scale up storage.

## Procedure

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

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** 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 4** On the displayed page, specify new storage and click **Next**.

**Figure 4-67** 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 5** 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 6** Check the results.

- When the scaling task is ongoing, the instance status is **Scaling storage space**.
- 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**

### 4.6.7.3 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.

   Log in to the IAM console using a Huawei Cloud account or as an IAM user, locate the IAM user that the target instance belongs to, and add it to the user group created in **3**. The IAM user will inherit permissions of the user group.

## Usage Notes

- Autoscaling is available only when your account balance is sufficient.
- The instance must be in the **Available** status.
- Once **Auto Scale** is enabled, an agency will be created and fees will be automatically deducted.

## Automatically Scaling Up Storage of a Single Instance

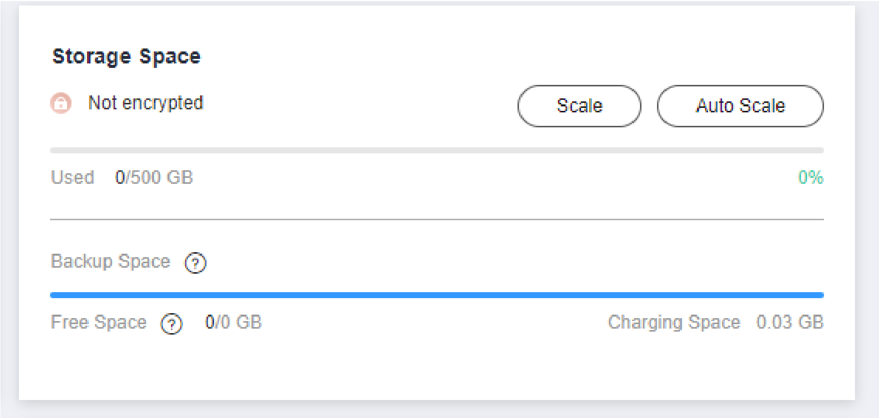
**Step 1** Log in to the Huawei Cloud console.

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** On the **Instances** page, click the target instance. The **Basic Information** page is displayed.

**Step 4** In the **Storage Space** area, click **Auto Scale**.

**Figure 4-68** Auto Scale




**Step 5** Toggle on **Auto Scale** and specify the parameters below.

**Figure 4-69** Configuring autoscaling parameters

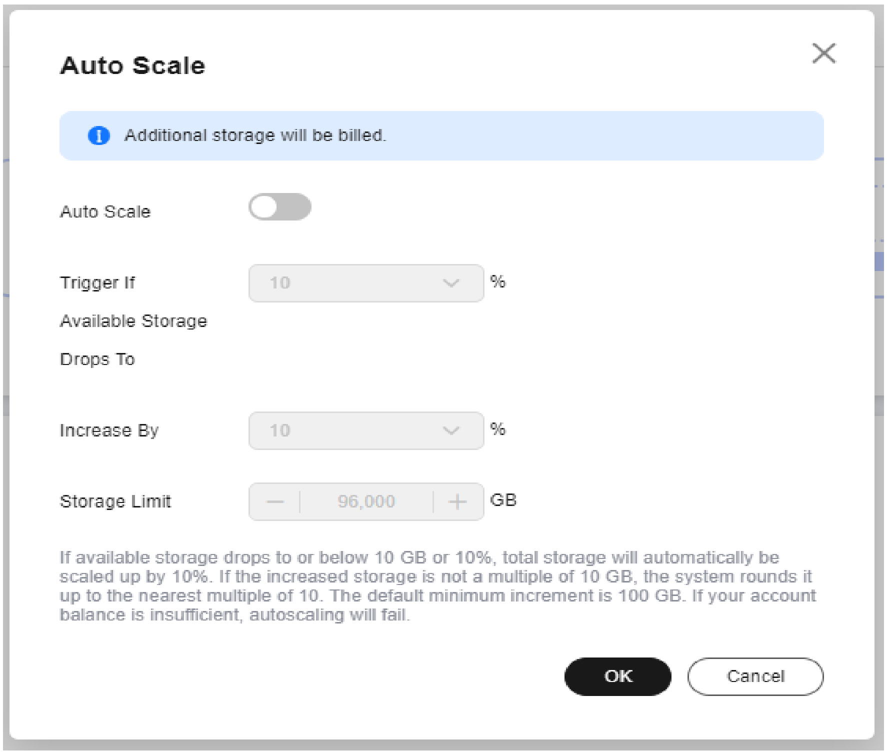


**Table 4-20** Description

| 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.<br>● Instance storage upper limit ≥ Current storage + 100 GB<br>● The instance storage upper limit cannot exceed the **maximum storage supported by the current specifications**. |

**Step 6** Click **OK**.

**----End**

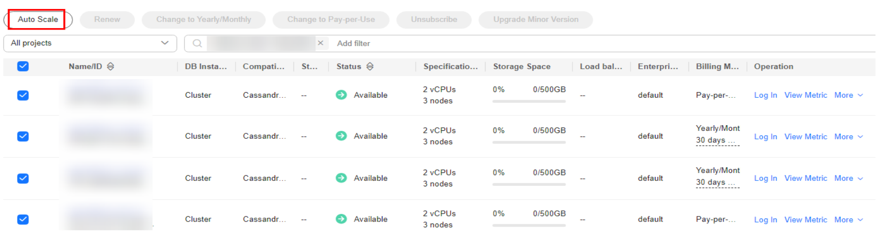
## Automatically Scaling Up Storage of Instances in Batches

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

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** Select instances and click **Auto Scale**.

**Figure 4-70** Auto Scale




**Step 4** Select an instance, toggle on **Auto Scale**, and specify the parameters below.

**Figure 4-71** Batch Auto Scale

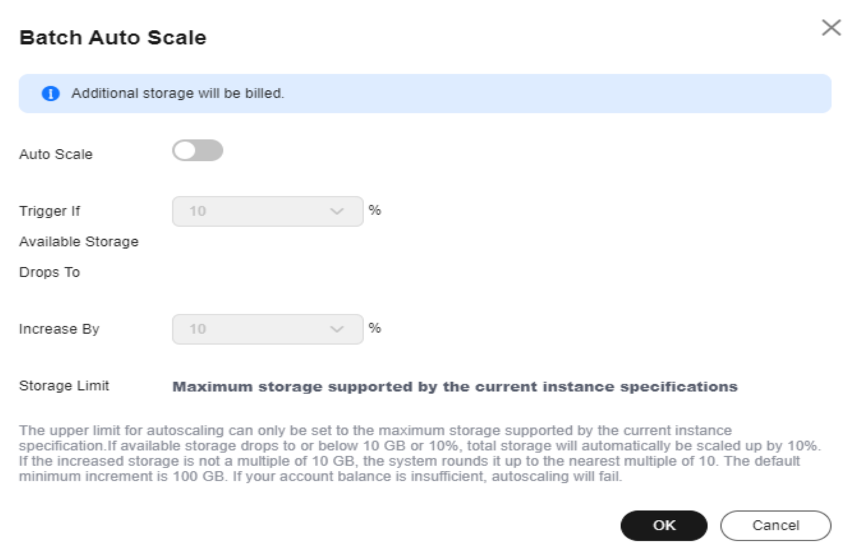



**Table 4-21** Description

| 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 | The value cannot be specified. By default, the storage is scaled up to the maximum defined by your instance specifications. |

**Step 5** Click **OK**.

**----End**

### 4.6.7.4 Manually Scaling Down Storage Space

As data volumes decrease, you can scale down storage space to avoid low database node utilization and resource waste.

## Usage Notes

- To scale down storage, ensure the new storage space is at least 1.25 times more than the used space and rounded up.

- Scaling down storage does not interrupt your services, and you do not need to restart your instance.
- If your yearly/monthly instance is running out of storage, additional usage will be billed on a pay-per-use basis. To avoid these extra costs and maintain the benefits of your yearly/monthly subscription, you are advised to scale up storage.
- Only classic storage can be scaled in.

## Procedure

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

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** 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 4** On the displayed page, specify new storage and click **Next**.

**Figure 4-72** Scaling down storage space


Select at least 1 GB each time, and the value must be an integer.

**Step 5** On the displayed page, confirm the storage space.

- Yearly/Monthly
  - To modify your settings, click **Previous** to go back to the page where you specify details.
  - If you do not need to modify your settings, click **Next** and complete the payment.
- Pay-per-use
  - To modify your settings, click **Previous** to go back to the page where you specify details.
  - If you do not need to modify your settings, click **Submit**.

**Step 6** Check the results.

- During the scale-down process, the instance status becomes **Scaling storage space**.
- 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**

# 4.7 Intra-region DR

## 4.7.1 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** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** 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 4** On the displayed page, configure required parameters and click **Next**.

**Table 4-22** Basic information

| Parameter | Description |
| --- | --- |
| Billing Mode | Select **Yearly/Monthly** or **Pay-per-use**.<br>● **Yearly/Monthly**<br>– Specify **Required Duration**. The system deducts fees from your account based on the service price.<br>– 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 a Yearly/Monthly Instance to Pay-per-Use**.<br>NOTE<br>Yearly/Monthly instances cannot be deleted directly. If such an instance is no longer required, unsubscribe from it. For details, see **How Do I Unsubscribe from a Yearly/Monthly Instance?**.<br>● **Pay-per-use**<br>– If you select this billing mode, you are billed based on how much time the instance is in use.<br>– To use an instance for a long time, change its billing mode to yearly/monthly to reduce costs. For details, see **Changing a Pay-per-Use Instance to Yearly/Monthly**. |
| Region | The region is the same as that of the primary instance. |
| DB Instance Name | The instance name:<br>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.<br>Instances can be deployed in a single AZ or three AZs.<br>● To deploy instances in a single AZ, select one AZ.<br>● To deploy instances across AZs for disaster recovery, select three AZs, where the instance nodes will be evenly distributed. |

**Table 4-23** Specifications and storage

| Parameter | Description |
| --- | --- |
| Instance Specifications | vCPUs and memory of the instance.<br>Performance specifications vary depending on the connections and maximum IOPS.<br>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.<br>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.<br>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:<br>• **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.<br>• **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.<br>• **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.<br>After an instance is created, you can scale up its storage if necessary. For details, see **Manually Scaling Up Storage Space**.<br>**NOTE**<br>• Once **Auto Scale** is enabled, an agency will be created and fees will be automatically deducted.<br>• 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.<br>• You can enable **Auto Scale** after an instance is created. For details, see **Automatically Scaling Up Storage Space**. |

**Table 4-24** 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. |
| SSL | A security protocol. Secure Sockets Layer (SSL) certificates set up encrypted connections between clients and servers, preventing data from being tampered with or stolen during transmission.<br>You can enable SSL to improve data security. After an instance is created, connect to it through SSL. |

**Table 4-25** 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.<br>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 4-26** 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.<br>A tag consists of a tag key and a tag value.<br>● A tag key is mandatory if the instance will be tagged. Each tag key is unique for each instance. It can contain 1 to 128 characters, cannot start with **_sys_**, and cannot start or end with a space. Only letters, digits, spaces, and the following special characters are allowed: -_@.:/+=<br>● A tag value is optional if the instance will be tagged. The value can be empty.<br>The value can contain a maximum of 255 characters. Only letters, digits, spaces, and the following special characters are allowed: _.-:+=@/<br>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 4-27** 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.<br>● If you select this option, the instance is automatically renewed based on the subscription duration. |

**Step 5** 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 6** 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**

## 4.7.2 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 **How Do I Unsubscribe 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** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** On the **Instances** page, locate the primary or DR instance that you want to delete and choose **More** > **Delete** in the **Operation** column.

**Step 4** 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 5** 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**

## 4.7.3 Redundancy Switchover Configuration

The GeminiDB Cassandra supports redundancy switchover configuration. You can configure DR switchover for an instance as required. If a natural disaster occurs on the primary DB instance and the node fault of the primary DB instance reaches the DR switchover threshold, the DR instance is switched to the primary DB instance. After the database link address is changed on the application side, service access of the application can be quickly restored.

## Prerequisites

A primary DB instance and a DR instance have been created.

## Procedure

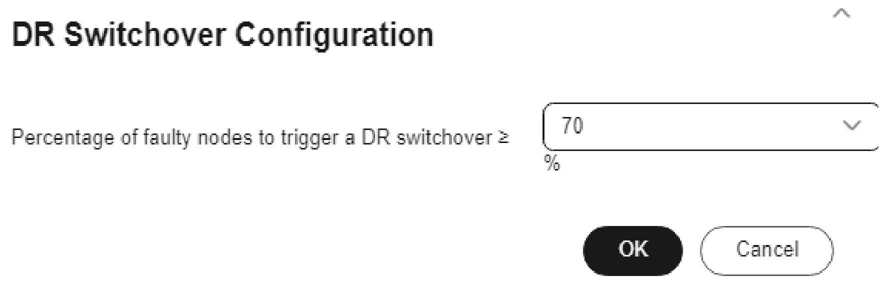
**Step 1** Log in to the Huawei Cloud console.

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** On the **Instance Management** page, locate the target primary DB instance and choose **More** > **DR Switchover Configuration** in the Operation column.

**Step 4** In the **DR Switchover Configuration** area, select the percentage of faulty nodes for instance DR switchover and click **to confirm the**. You can select 50, 60, 70, 80, 90, or 100.

**Figure 4-73** Redundancy Switchover Configuration




**----End**

# 4.8 Cross-region Dual-active DR

## 4.8.1 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**.

# 4.8.2 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 HBase and DynamoDB-Compatible instances cannot.
- Currently, counter tables, TRUNCATE operations, and Lucene indexes cannot be synchronized between active-active instances.

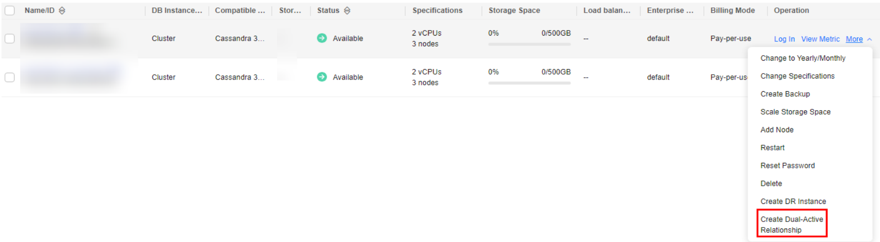
## Procedure

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

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

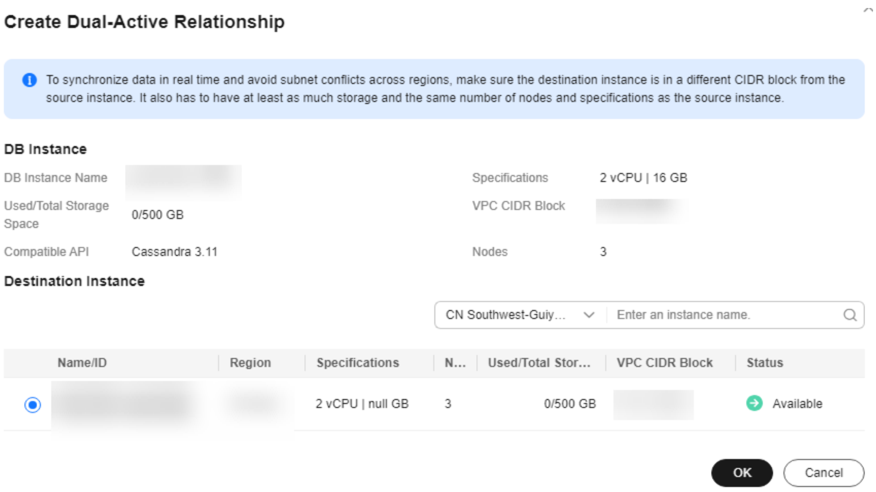
**Step 3** On the **Instances** page, locate the source instance that you want to create a dual-active relationship for and choose **More** > **Create Dual-Active Relationship** in the **Operation** column.

**Figure 4-74** Creating a dual-active relationship




**Step 4** On the **Create Dual-Active Relationship** dialog box, locate the destination instance as the dual-active DR instance.

**Figure 4-75** 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 5** Click **OK**.

**Step 6** 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**

## 4.8.3 Deleting a Dual-active Relationship

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

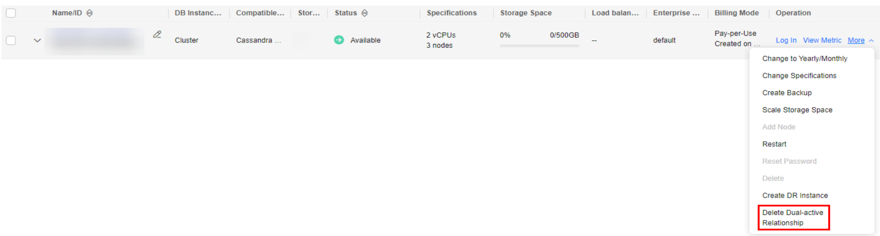
### Procedure

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

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** 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 4-76** Deleting a dual-active relationship




**Step 4** If you have enabled operation protection, in the displayed dialog box, 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 5** 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**

# 4.9 Data Backup

## 4.9.1 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 4-28** 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 4-29** 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. |
| **Managing Table-level Backups** | If a database or table is deleted maliciously or accidentally, you can use backups to restore data. Manual and automated backups are supported. |

Full and incremental backups are created based on data volumes.

**Table 4-30** 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.<br>• 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.<br>• You can use a full backup to restore all data generated when its backup was created.<br>• Full backups can be created automatically or manually. | • Incremental data in your instance is backed up since the last full backup.<br>• When you use an incremental backup for restoration, the last full backup data and the incremental data generated since then are downloaded.<br>• Incremental backups can be created automatically only. |

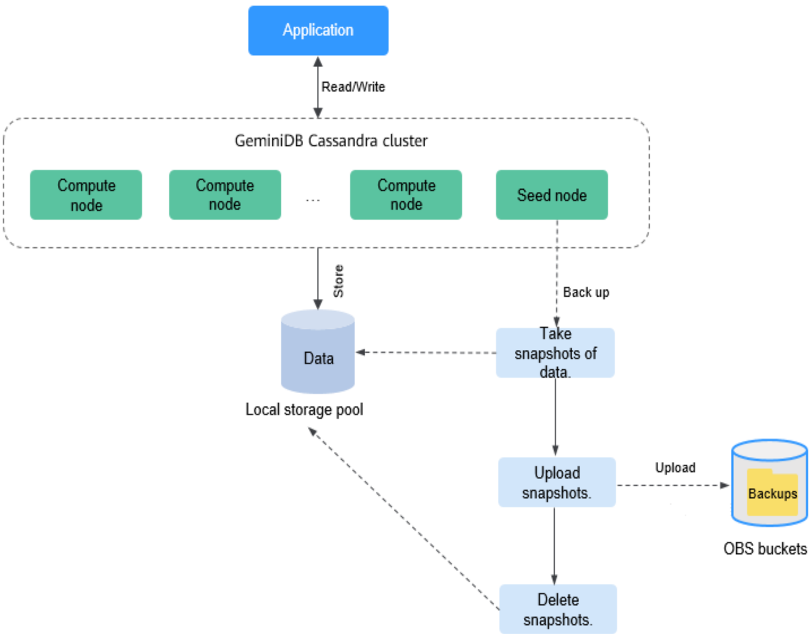
  

| | | |
|---|---|---|
| **How to View** | Click an instance name. On the **Backups & Restorations** page, click the **Instance-level Backups** and **Table-level Backups** tabs to view the backup size. | Click an instance name. On the **Backups & Restorations** page, click the **Incremental Backup** tab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

**Figure 4-77** Backup process




## 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.

## 4.9.2 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.

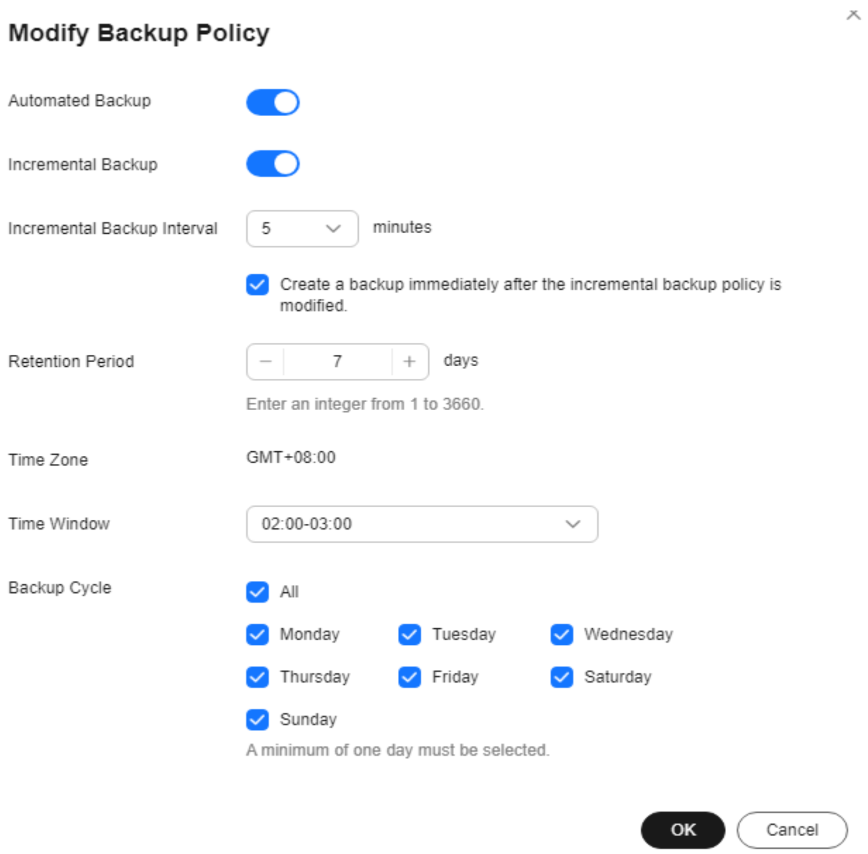
### Automated Backup Policy

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.

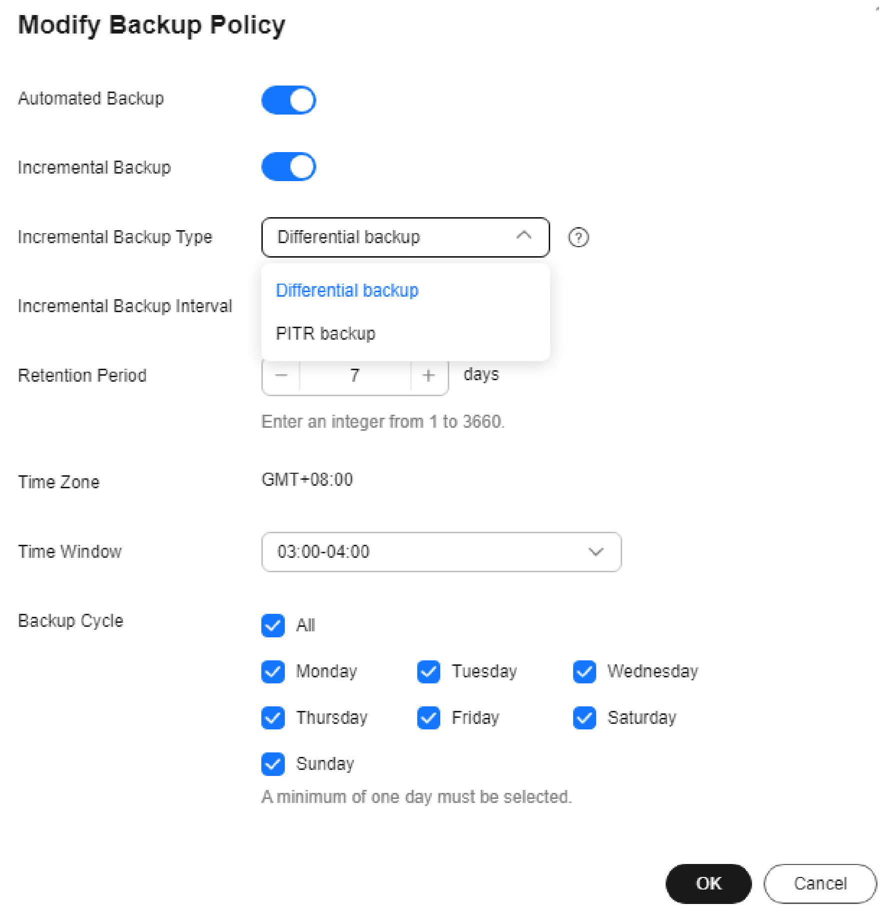
  

**Figure 4-78** Enabling the automated backup policy




- **Incremental Backup**: Incremental backup is enabled by default. You can click  to manually enable or disable **Incremental Backup**. After incremental backup is enabled, the system stores backup data in OBS. Select an incremental backup type. **Differential backup** is selected by default.
  - Differential backup: Data can be restored to a specified point in time.
  - PITR backup: Data can be restored to any point in time.

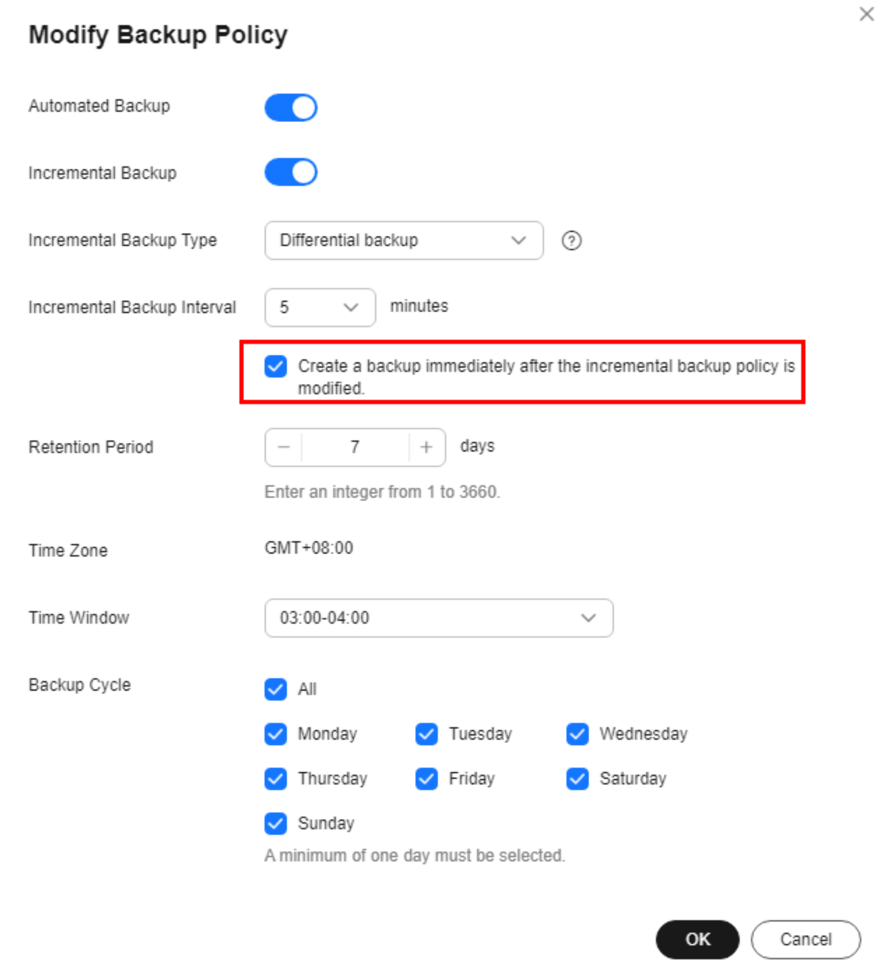
**Figure 4-79** Selecting an incremental backup type




Enabling **Incremental Backup** will take effect in the next full backup. You are advised to select **Create a backup immediately after the incremental backup policy is modified**.

- If you select it, the full backup request is delivered immediately, and the incremental backup takes effect.
- If you do not select it, the incremental backup will take effect in the next full backup.

**Figure 4-80** Selecting **Create a backup immediately after the incremental backup policy is modified**




- **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 00:00–01:00. 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.

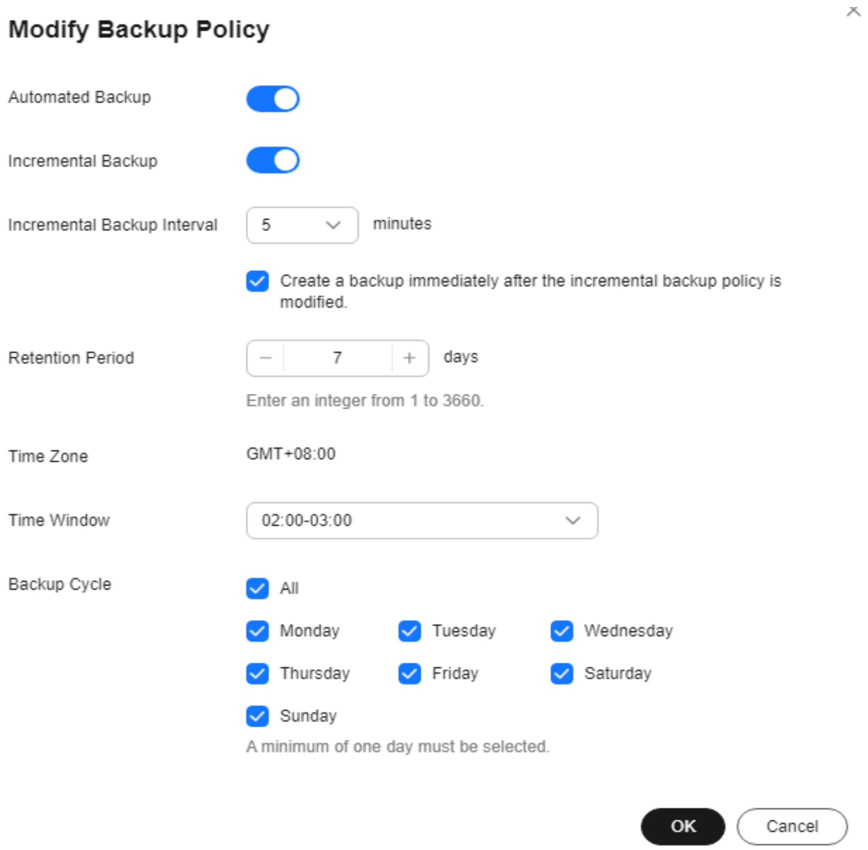
**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** On the **Instances** page, click the instance whose backup policy you want to modify and click its name.

**Step 4** 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 4-81** Modifying a backup policy




**Step 5** 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.

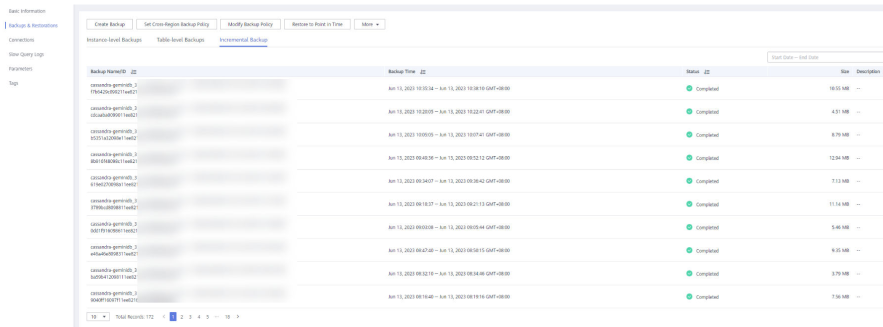
**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** On the **Instances** page, click the target instance name to access the **Basic Information** page.

**Step 4** In the navigation pane, choose **Backups & Restorations**.

**Step 5** On the **Backups & Restorations** page, click **Incremental Backup**.

**Figure 4-82** Incremental backup



**Step 6** View incremental backups and their size.

**----End**

## Disabling Incremental Backup

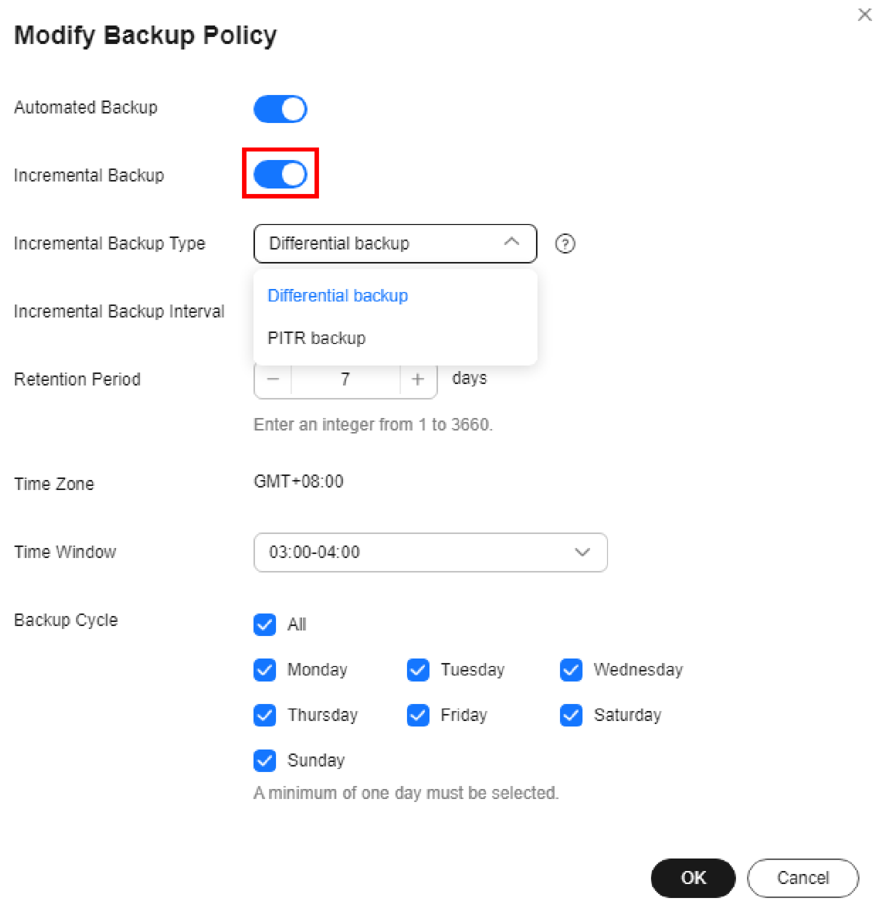
**Step 1** Log in to the Huawei Cloud console.

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** On the **Instances** page, click the target instance. The **Basic Information page** is displayed.

**Step 4** In the navigation pane, choose **Backups & Restorations**. On the displayed page, click **Modify Backup Policy** and click  next to **Incremental Backup**.

**Figure 4-83** Disabling **Incremental Backup**




**----End**

## Disabling an Automated Backup Policy

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

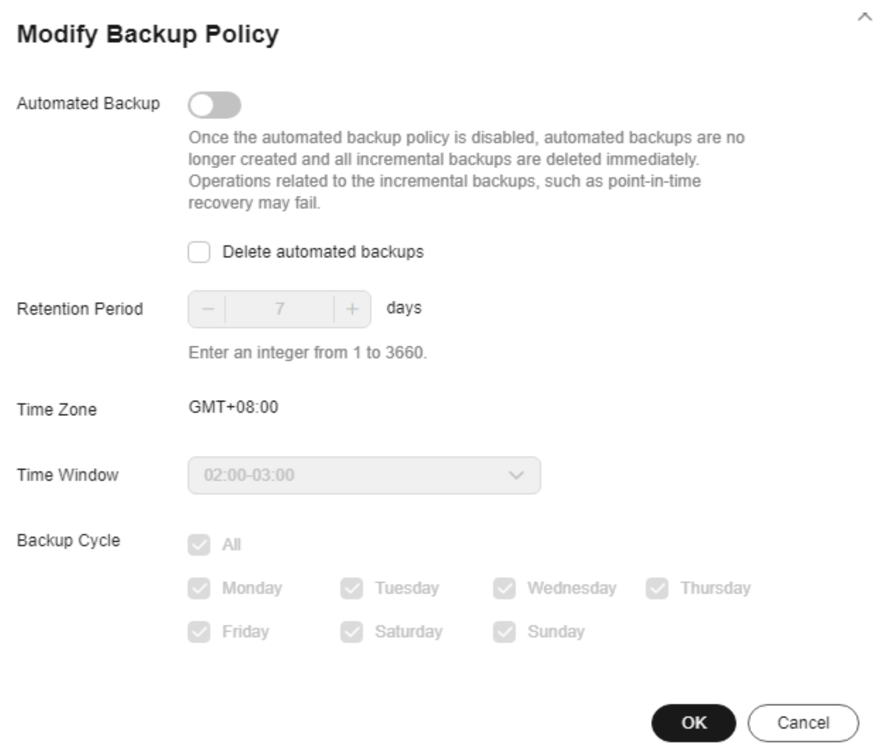
**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** On the **Instances** page, locate the instance that you want to disable automated backup for and click its name.

**Step 4** Choose **Backups & Restorations** in the navigation pane one the left, and click **Modify Backup Policy**.

**Step 5** In the displayed dialog box, click  and click **OK**.

**Figure 4-84** 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. In the service list, choose **Databases** > **GeminiDB**.
  c. On the **Instances** page, click the instance whose automated backups you want to delete and click its name.
  d. Choose **Backups & Restorations** in the navigation pane on the left, locate the backup you want to delete and click **Delete** in the **Operation** column.
  e. In the displayed dialog box, confirm the backup details and click **Yes**.
- **Method 2**
  a. Log in to the Huawei Cloud console.
  b. In the service list, choose **Databases** > **GeminiDB**.
  c. On the **Backups** page, locate the backup that you want to delete and click **Delete** in the **Operation** column.
  d. In the displayed dialog box, confirm the backup details and click **Yes**.

## 4.9.3 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 charged for instances with cloud native storage during OBT.

### Creating a Manual Backup

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

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** 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 4** In the displayed dialog box, enter a backup name and description and click **OK**.

**Figure 4-85** Creating a backup


**Table 4-31** 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 5** 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**.

You can create manual table-level backups by following **Creating and Managing Table-level Backups**.

**----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. In the service list, choose **Databases** > **GeminiDB**.
3. On the **Instances** page, locate the instance whose backup you want to delete and click its name.
4. Choose **Backups & Restorations** in the navigation pane on the left, locate the backup you want to delete and click **Delete** in the **Operation** column.
5. In the displayed dialog box, confirm the backup details and click **Yes**.

Method 2

1. Log in to the Huawei Cloud console.
2. In the service list, choose **Databases** > **GeminiDB**.
3. On the **Backups** page, 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**.

# 4.9.4 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 4-32** Billing

| Flavor | Billing Item | Unit Price |
|---|---|---|
| geminidb.cassandra.crossreg.backup.space.dfv | Storage space | CNY0.0009/GB/hour |
| 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** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** On the **Instances** page, locate the instance that you want to connect to and click its name.

**Step 4** In the navigation pane on the left, choose **Backups & Restorations**.

**Step 5** On the displayed page, click **Set Cross-Region Backup Policy**.

**Step 6** In the displayed dialog box, set required parameters.

**Figure 4-86** Setting a cross-region backup policy

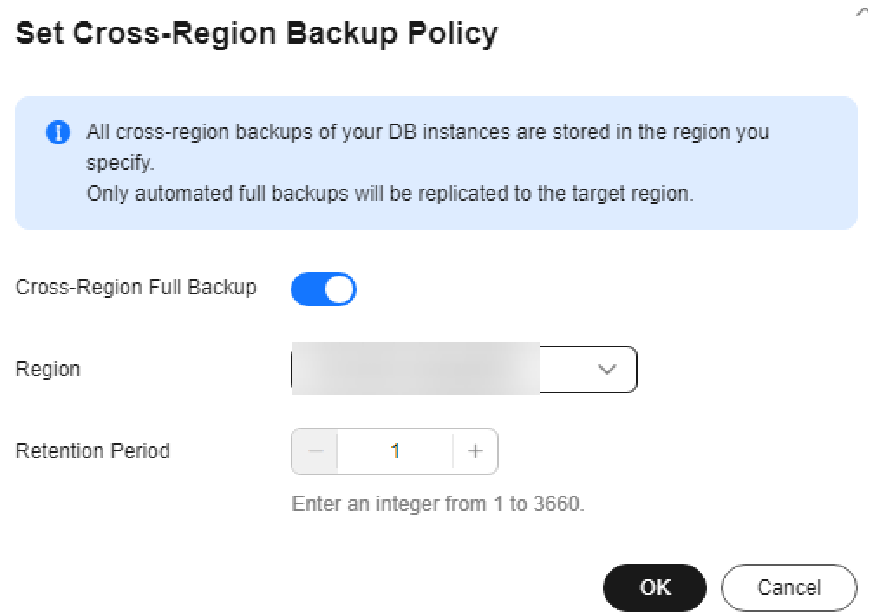


**Table 4-33** Description

| 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 7** Click **OK**.

**----End**

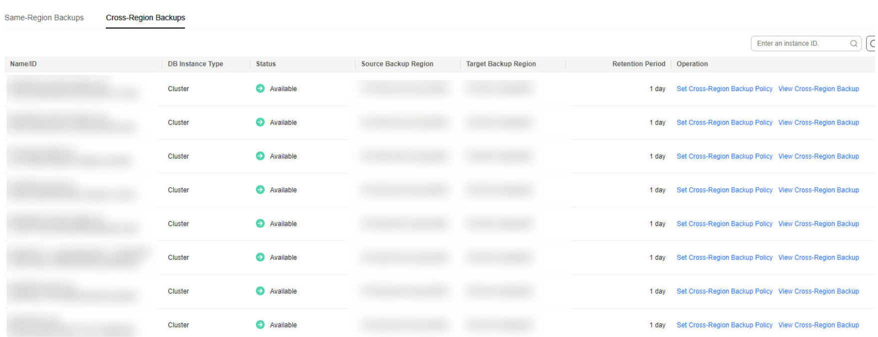
## Managing Cross-Region Backups

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

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** In the navigation pane, choose **Backups**. Click the **Cross-Region Backups** tab.

**Figure 4-87** Cross-region backups

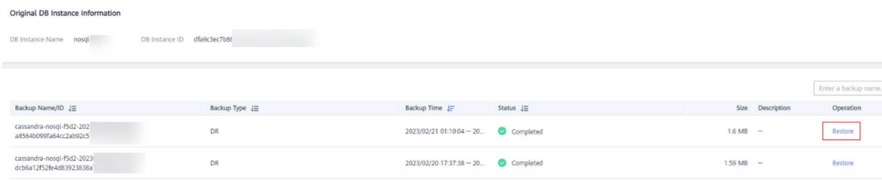



All cross-region backups are displayed by default.

- 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 4-88** 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 that has disk encryption enabled.

**----End**

## Disabling Cross-Region Backup

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

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

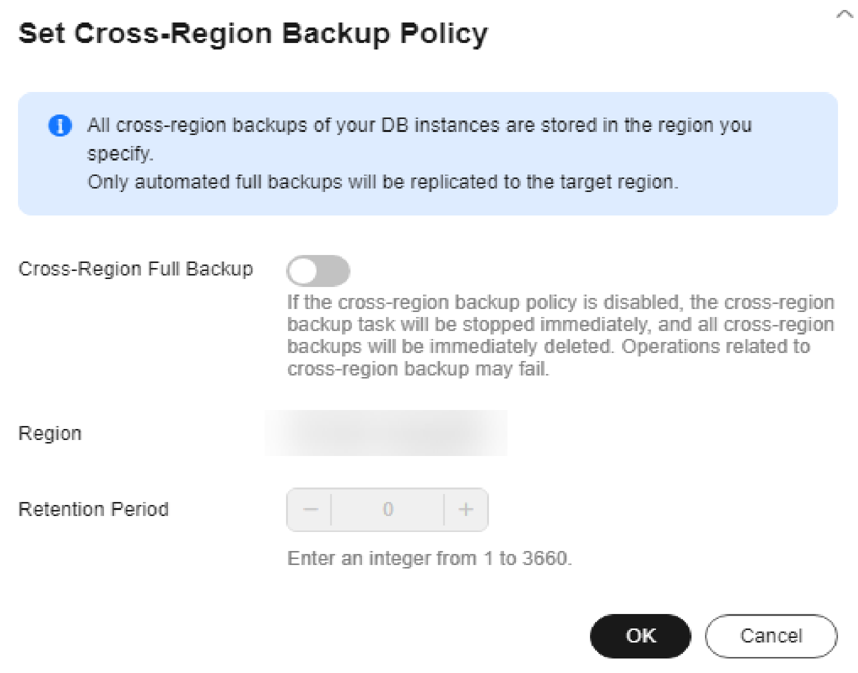
**Step 3** On the **Instances** page, locate the instance that you want to connect to and click its name.

**Step 4** In the navigation pane on the left, choose **Backups & Restorations**.

**Step 5** On the displayed page, click **Set Cross-Region Backup Policy**.

**Step 6** In the displayed dialog box, disable **Cross-Region Full Backup**.

**Figure 4-89** 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 7** Click **OK**.

**----End**

## 4.9.5 Managing Table-level Backups

GeminiDB Cassandra API allows you to create table-level backup for your instance. If a database or table is deleted maliciously or accidentally, you can use backups to restore data.

### Usage Notes

- To enable the table-level backup, choose **Service Tickets > Create Service Ticket** in the upper right corner of the console and contact the customer service.
- Table-level backups can be created automatically or manually.

### Enabling or Modifying a Table-level Backup Policy

**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 target instance name to access the **Basic Information** page.

**Step 4** In the navigation pane, choose **Backups & Restorations**.

**Step 5** On the displayed page, click **Configure Table-level Backup Policy**.

**Step 6** On the displayed page, configure related parameters.

**Figure 4-90** Setting a table-level backup policy

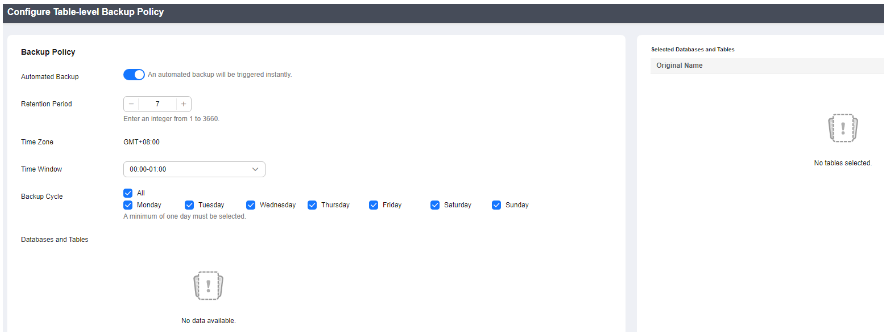



**Table 4-34** Parameters

| Parameter | Description |
| --- | --- |
| Automated Backup | After this function is enabled, database table data is backed up based on the backup policy. |
| Retention Period | Automated backup files are stored for 7 days by default. The retention period ranges from 1 to 732 days.<br>● If the retention period is shorter than seven days, the system automatically backs up data daily.<br>● The system automatically checks existing backup files and deletes files that exceed the retention period you set. |
| Time Window | A one-hour period the backup will be scheduled within 24 hours, such as 01:00-02:00 or 12:00-13:00. The backup time is in GMT format. If the DST or standard time is switched, the time window changes with the time zone. |
| Backup Cycle | ● **All**: Each day of the week is selected. This option is selected by default. The system automatically backs up data every day.<br>● You can select one or more days in a week. The system automatically backs up data on the specified days. |
| Databases and Tables | Select tables to be backed up. |

NOTE

A full backup starts within one hour of the time you specify. How long the backup takes depends on the data volume.

**Step 7** Click **OK**.

**----End**

## Creating and Managing Table-level Backups

Creating a Manual Table-level Backup

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

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** On the **Instances** page, locate the instance that you want to create a table-level backup for and click its name.

**Step 4** In the navigation pane on the left, choose **Backups & Restorations**.

**Step 5** On the displayed page, click **Create Table-level Backup**.

**Step 6** On the displayed page, configure related parameters.

**Figure 4-91** Creating a table-level backup

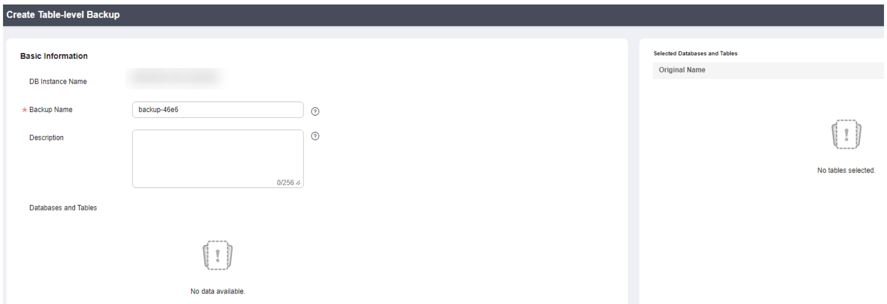



**Table 4-35** Parameters

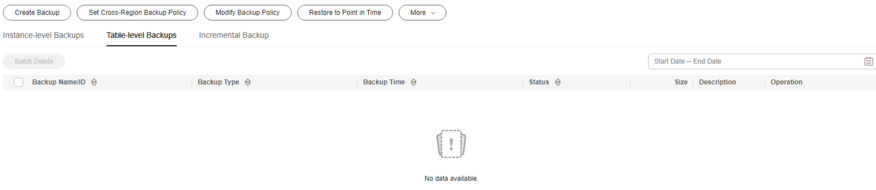
| Parameter | Description |
| --- | --- |
| Backup Name | The backup name can contain 4 to 64 characters and must start with a letter. The name is case-sensitive and can contain only letters, digits, hyphens (-), or underscores (_). |
| Description | The description can contain a maximum of 256 characters and cannot contain line breaks or special characters >!<"&'= |
| Databases and Tables | You can select the databases and specify tables therein that you want to back up. |

**Step 7** Click **OK**.

**Step 8** Choose **Backups and Restorations** > **Table-level Backup** and manage the created backup.

Alternatively, click **Backups** in the navigation pane on the left, choose **Intra-region Backups** > **Table-level Backups**, and manage the created backup.

**Figure 4-92** Managing the created table-level backup




- Click **View Tables** to view tables contained in the backup file.
- Click **Restore** in the **Operation** column to restore the backup to a new instance. For details, see **Restoring Data to a New Instance**.
- Click **Delete** in the **Operation** column to delete the created backup.

  NOTE

  Deleted backups cannot be recovered.

**----End**

## Disabling a Table-level Backup Policy

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

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

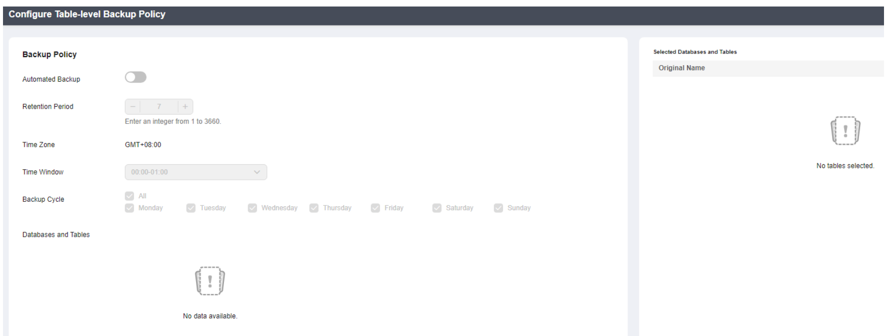
**Step 3** On the **Instances** page, click the target instance name to access the **Basic Information** page.

**Step 4** In the navigation pane, choose **Backups & Restorations**.

**Step 5** On the displayed page, click **Configure Table-level Backup Policy**.

**Step 6** On the displayed page, configure related parameters.

**Figure 4-93** Disabling table-level backup policy




NOTE

After the table-level backup policy is disabled, any table-level backup task in progress stops immediately, and all table-level backups of the instance are retained. The retention duration depends on **Retention Period** specified when you enabled the table-level backup policy.

**Step 7** Click **OK**.

**----End**

# 4.10 Data Restoration

## 4.10.1 Restoration Methods

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

**Table 4-36** 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. |

## 4.10.2 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.
- Incremental backup and PITR are not supported.
- 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.

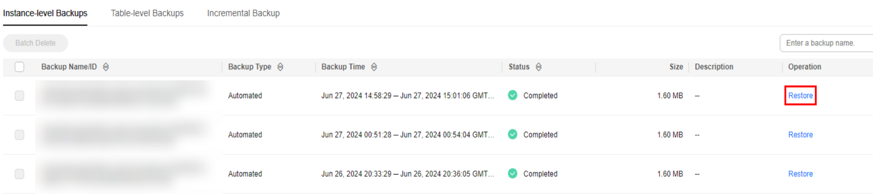
**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** 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.

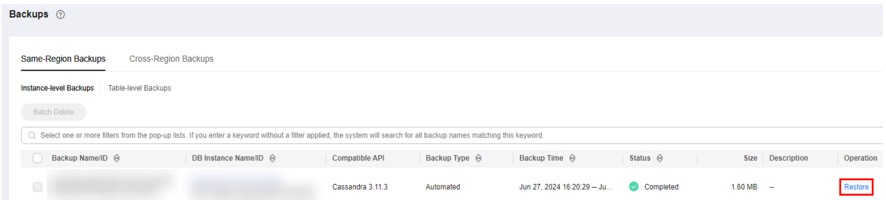
**Figure 4-94** Backups and restorations




Method 2

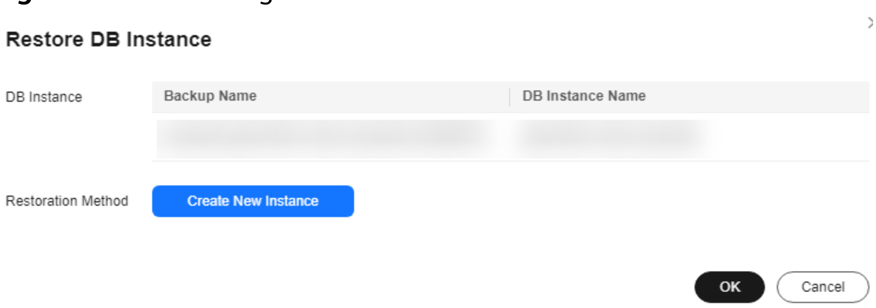
On the **Backups** page, locate the backup that you want to restore and click **Restore** in the **Operation** column.

**Figure 4-95** Backup management




**Step 4** In the displayed dialog box, confirm the current instance details and restoration method and click **OK**.

**Figure 4-96** 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 5** 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**

## 4.10.3 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.
- Keep your account balance above zero so that backup data can be restored to a new instance.
- 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.

### Procedure

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

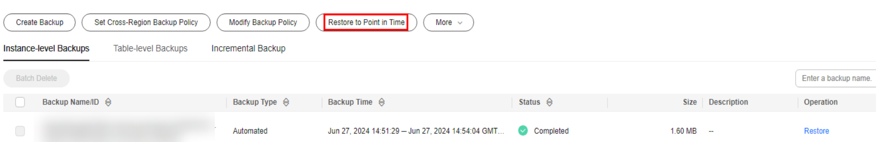
**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** On the **Instances** page, click the instance that you want to perform a PITR for.

**Step 4** In the navigation pane on the left, choose **Backups & Restorations**.

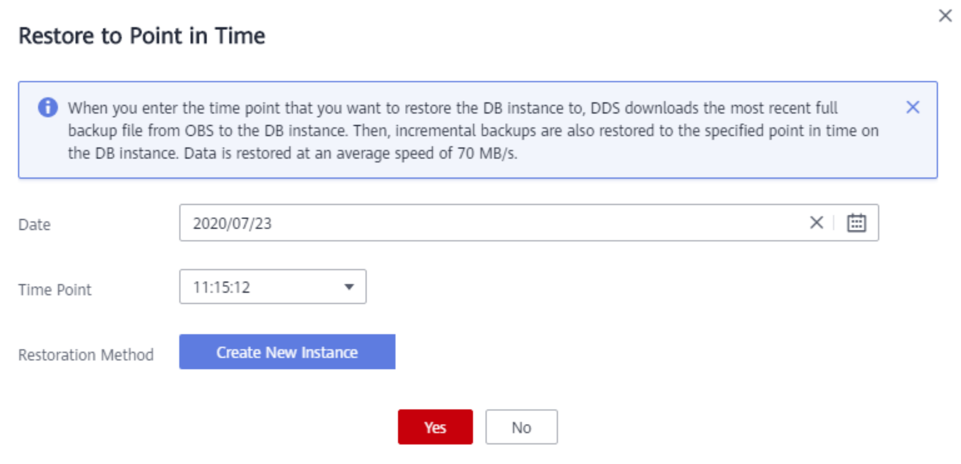
**Step 5** On the **Backups & Restorations** page, click **Restore to Point in Time**.

**Figure 4-97** Restoring data to a point in time



**Step 6** Select a restoration date and a time point and click **OK**.

**Figure 4-98** Restore to Point in Time




**Step 7** 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**

# 4.11 Parameter Management

## 4.11.1 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.

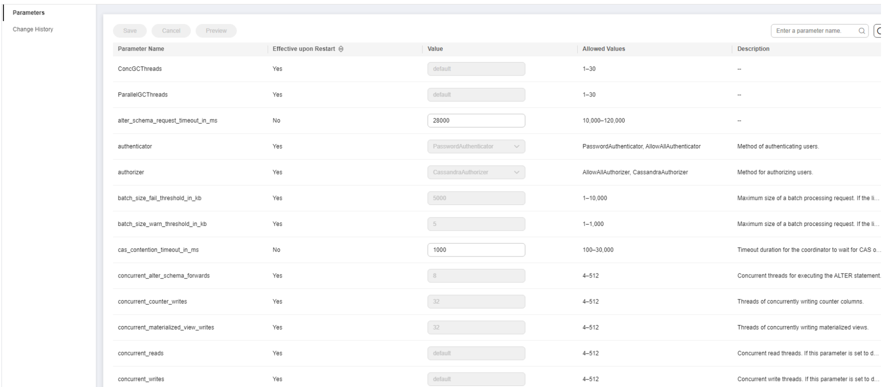
**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** In the navigation pane on the left, choose **Parameter Templates**.

**Step 4** Click the **Custom Templates** tab, locate the parameter template you want to modify, and click its name.

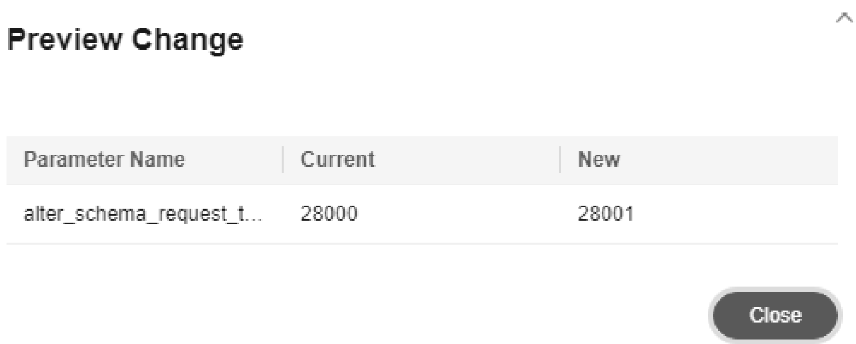
**Step 5** Change parameter values as required.

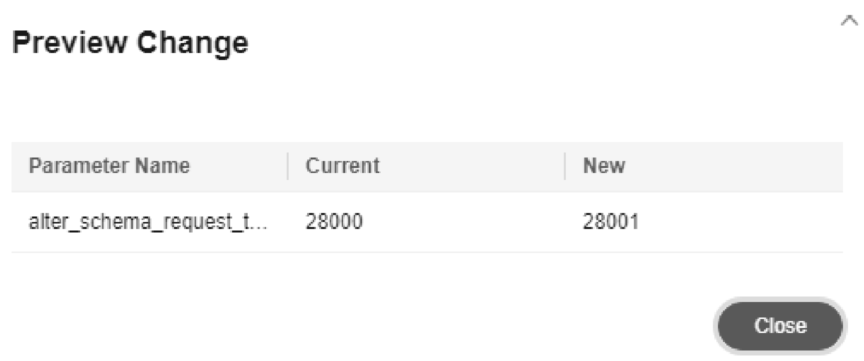
**Figure 4-99** Modifying parameters in a parameter template



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

**Figure 4-100** Previewing changes






**Step 6** 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

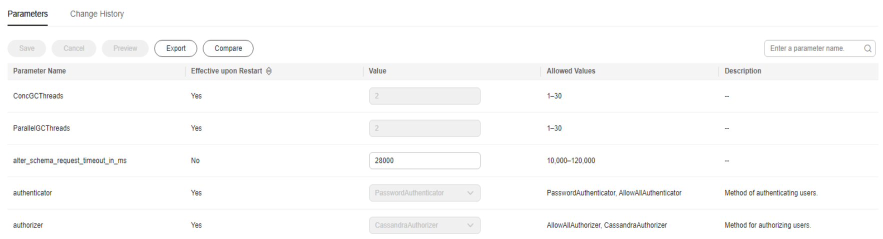
**Step 1** Log in to the Huawei Cloud console.

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** 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 4** In the navigation pane on the left, choose **Parameters**. On the displayed page, modify parameters as required.

**Figure 4-101** Parameters




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

**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

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**

## 4.11.2 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.

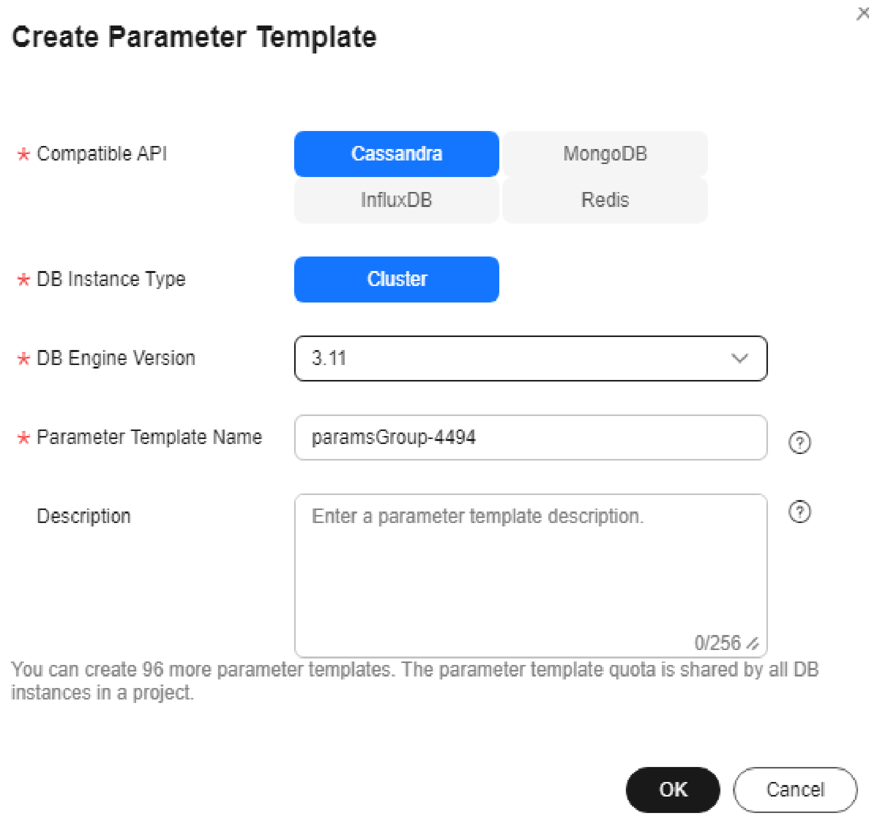
**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** In the navigation pane on the left, choose **Parameter Templates**.

**Step 4** On the **Parameter Templates** page, click **Create Parameter Template**.

**Step 5** Select a compatible API, specify a DB engine version and a parameter group description, and click **OK**.

**Figure 4-102** 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 6** On the **Parameter Templates** page, view the created parameter template.

**----End**

# 4.11.3 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

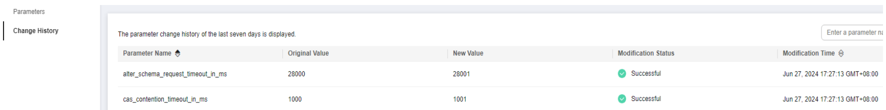
**Step 1** Log in to the Huawei Cloud console.

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** 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 4** 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 4-103** 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

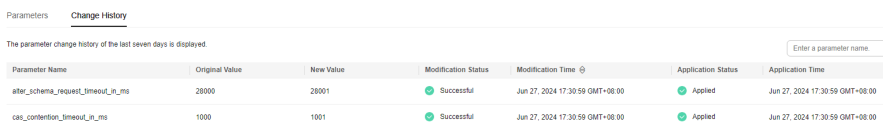
**Step 1** Log in to the Huawei Cloud console.

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** On the **Instances** page, locate the instance whose parameter change history you want to view and click its name.

**Step 4** 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 4-104** Viewing parameter change history of an instance




**----End**

# 4.11.4 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

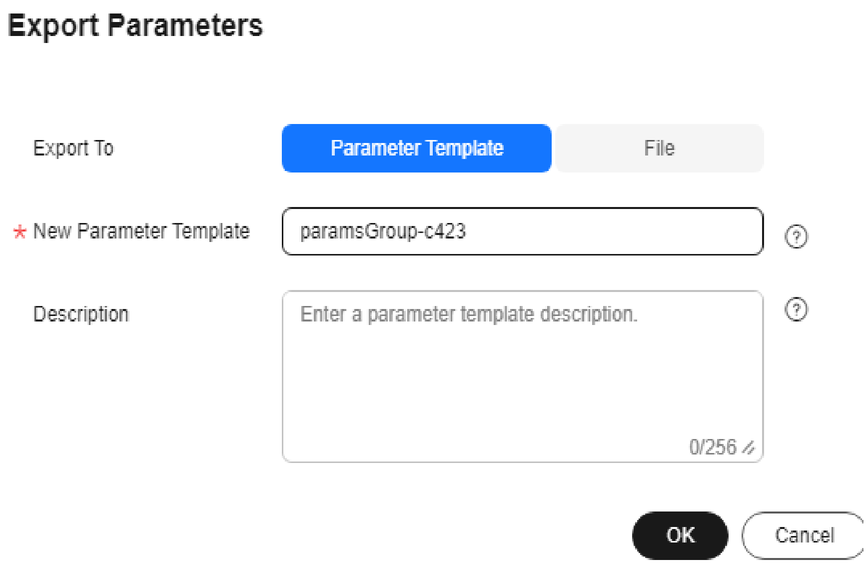
**Step 1** Log in to the Huawei Cloud console.

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** In the navigation pane on the left, choose **Instances**, locate the instance whose parameters you want to export, and click its name.

**Step 4** In the navigation pane on the left, choose **Parameters**. On the **Parameters** tab, above the parameter list, click **Export**.

**Figure 4-105** 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**

# 4.11.5 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 service list, choose **Databases** > **GeminiDB**.

**Step 3** In the navigation pane on the left, choose **Parameter Templates**.

**Step 4** In the parameter template list, locate the parameter template that you created and click **Compare** in the **Operation** column.

**Step 5** 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 4-106** 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 service list, choose **Databases** > **GeminiDB**.

**Step 3** In the navigation pane on the left, choose **Instances**.

**Step 4** In the instance list, locate the instance whose parameter templates you want to compare and click its name.

**Step 5** In the navigation pane on the left, choose **Parameters** and then click **Compare** above the parameter list.

**Step 6** 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 4-107** 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**

## 4.11.6 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 service list, choose **Databases** > **GeminiDB**.

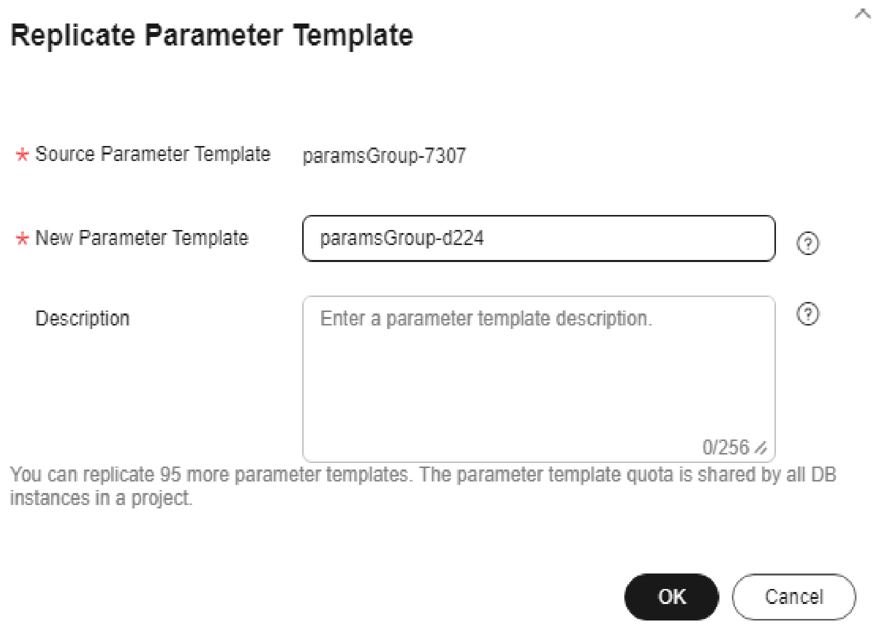
**Step 3** In the navigation pane on the left, choose **Parameter Templates**.

**Step 4** 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 5** In the displayed dialog box, enter the parameter template name and description and click **OK**.

**Figure 4-108** 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**

## 4.11.7 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 service list, choose **Databases** > **GeminiDB**.

**Step 3** In the navigation pane on the left, choose **Parameter Templates**.

**Step 4** On the **Parameter Templates** page, click the **Custom Templates** tab. Locate the target parameter template and choose **More** > **Reset** in the **Operation** column.

**Step 5** Click **Yes** to reset the parameter template.

**----End**

## 4.11.8 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 service list, choose **Databases** > **GeminiDB**.

**Step 3** In the navigation pane on the left, choose **Parameter Templates**.

**Step 4** 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 5** 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**

## 4.11.9 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 service list, choose **Databases** > **GeminiDB**.

**Step 3** In the navigation pane on the left, choose **Parameter Templates**.

**Step 4** 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**

## 4.11.10 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 service list, choose **Databases** > **GeminiDB**.

**Step 3** In the navigation pane on the left, choose **Parameter Templates**.

**Step 4** On the **Parameter Templates** page, click the **Custom Templates** tab. Locate the target parameter template and click  in the **Description** column.

**Step 5** 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**

## 4.11.11 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 service list, choose **Databases** > **GeminiDB**.

**Step 3** In the navigation pane on the left, choose **Parameter Templates**.

**Step 4** 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 5** Click **Yes** to delete the parameter template.

**----End**

# 4.12 Logs and Audit

## 4.12.1 Viewing and Exporting Slow Query Logs

GeminiDB Cassandra API allows you to view slow query logs of databases. The unit of the execution time is ms. You can identify the SQL statements that take a long time to execute and tune them based on slow query logs.

## Viewing and Exporting Log Details

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

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** On the **Instances** page, locate the instance whose log details you want to view and click its name.

**Step 4** In the navigation pane on the left, choose **Slow Query Logs**.

**Step 5** On the **Slow Query Logs** page, set search criteria and click **Search** to view log information.

- Select **All nodes** and view slow query logs of all nodes. Alternatively, select a specific node to view its slow query logs.
- View slow query logs of a specific node in different time ranges.
- View slow query logs of the following types of SQL statements:
  - SELECT

**Step 6** On the **Log Details** page, click in the upper right corner of the log list to export log details.

- You can view the exported CSV file to your local PC.
- Up to 2,000 logs can be exported at a time.

**----End**

# 4.12.2 CTS

## 4.12.2.1 Key Operations Supported by CTS

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

**Table 4-37** 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 |

| Operation | Resource Type | Trace Name |
|---|---|---|
| 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 |
| 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 | NoSQLDeleteEnlargeFailNode |
| Enabling SSL | instance | NoSQLSwitchSSL |

| Operation | Resource Type | Trace Name |
| --- | --- | --- |
| Changing the security group of an instance | instance | NoSQLModifySecurityGroup |
| Configuring autoscaling | instance | NoSQLModifyAutoEnlarge-Policy |
| Creating a dual-active relationship | instance | NoSQLBuildBiactiveInstance |
| Exporting parameter template information for an instance | instance | NoSQLSaveConfigurations |
| Modifying the recycling policy | instance | NoSQLModifyRecyclePolicy |

### 4.12.2.2 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 Huawei Cloud console.

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

**Step 3** Click **Service List**. Under **Management & Governance**, click **Cloud Trace Service**.

**Step 4** In the navigation pane on the left, click **Trace List**.

**Step 5** 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 6** Locate the required trace and click on the left of the trace to view details.

**Step 7** Click **View Trace** in the **Operation** column. In the displayed dialog box, the trace structure details are displayed.

**----End**

# 4.13 Viewing Metrics and Configuring Alarms

## 4.13.1 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 4-38** 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 usage of the current instance. | 0–100 | % | N/A | GeminiDB Cassandra instances | 1 minute |
| nosql006_disk_total_size | Total Storage Space | Total storage space of the current instance. | ≥ 0 | GB | 1024(IEC) | GeminiDB Cassandra instances | 1 minute |
| nosql007_disk_used_size | Storage Space Usage | Storage space usage of the current instance. | ≥ 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 |

| Metric ID | Name | Description | Value Range | Unit | Number System | Monitored Object | Monitoring Period (Raw Data) |
|---|---|---|---|---|---|---|---|
| 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 |
| 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 |

| Metric ID | Name | Description | Value Range | Unit | Number System | Monitored Object | Monitoring Period (Raw Data) |
|---|---|---|---|---|---|---|---|
| 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 |
| 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 |

| Metric ID | Name | Description | Value Range | Unit | Number System | Monitored Object | Monitoring Period (Raw Data) |
|---|---|---|---|---|---|---|---|
| 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 |
| 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 |

| Metric ID | Name | Description | Value Range | Unit | Number System | Monitored Object | Monitoring Period (Raw Data) |
|---|---|---|---|---|---|---|---|
| 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 |
| 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 |

| Metric ID | Name | Description | Value Range | Unit | Number System | Monitored Object | Monitoring Period (Raw Data) |
|---|---|---|---|---|---|---|---|
| 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 |
| 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 |

| Metric ID | Name | Description | Value Range | Unit | Number System | Monitored Object | Monitoring Period (Raw Data) |
|---|---|---|---|---|---|---|---|
| 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 |

# 4.13.2 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

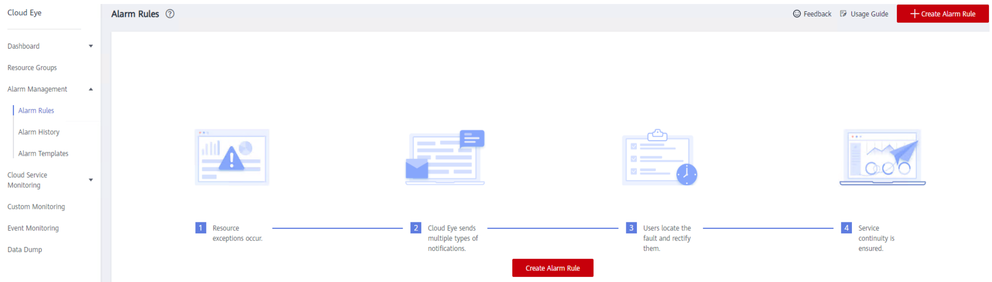
**Step 1** Log in to the Huawei Cloud console.

**Step 2** Click **Service List**. Under **Management & Governance**, click **Cloud Eye**.

**Step 3** In the navigation pane on the left, choose **Alarm Management** > **Alarm Rules**.

**Step 4** On the **Alarm Rules** page, click **Create Alarm Rule**.

**Figure 4-109** Creating an alarm rule




**Step 5** Set alarm parameters.

1. Configure basic alarm information.

**Figure 4-110** Configuring basic information for an alarm rule

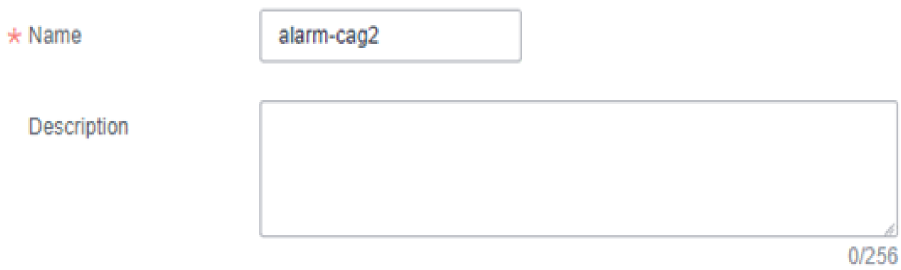



**Table 4-39** Basic alarm rule information

| 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 4-111** Configuring objects to be monitored

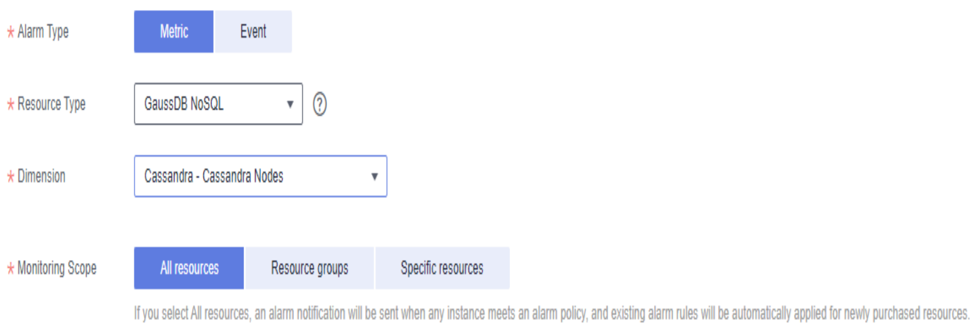


**Table 4-40** Parameter description

| 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.<br>Select **GeminiDB**. | - |
| Dimension | Metric dimension of the alarm rule.<br>Select **Cassandra - Cassandra Nodes**. | - |
| Monitoring Scope | Monitoring scope the alarm rule applies to.<br>**NOTE**<br>– If you select **All resources**, an alarm notification will be sent when any instance meets an alarm policy, and existing alarm rules will be automatically applied for newly purchased resources.<br>– If you select **Resource groups** and any resource in the group meets the alarm policy, an alarm notification will be sent.<br>– To specify **Specific resources**, click **Select Specified Resources**, select one or more resources, and click **OK**. | All resources |
| Group | This parameter is mandatory when **Monitoring Scope** is set to **Resource groups**. | - |

3. Configure an alarm policy.

**Figure 4-112** Configuring an alarm policy

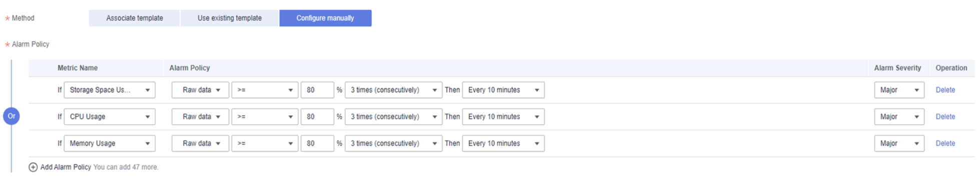


**Table 4-41** Parameter description

| Parameter | Description | Example Value |
|---|---|---|
| Method | Select **Associate template**, **Use existing template**, or **Configure manually**.<br>**NOTE**<br>If you set **Monitoring Scope** to **Specific resources**, you can set **Method** to **Use existing template**. | Configure manually |

| Parameter | Description | Example Value |
| --- | --- | --- |
| Template | Select the template to be used.<br>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.<br>– **Metric Name**: metric that an alarm rule is created for<br>The following metrics are recommended:<br>**Storage Space Usage**:<br>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**.<br>**CPU Usage** and **Memory Usage**:<br>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.<br>For more metrics, see **Supported Metrics**.<br>– **Alarm Severity**: specifies the severity of the alarm. Valid values are **Critical**, **Major**, **Minor**, and **Informational**.<br>NOTE<br>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 4-112** 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 4-113** Configuring alarm notification information

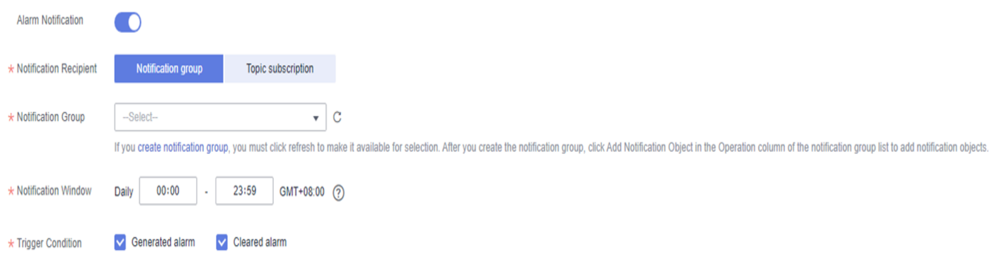


**Table 4-42** Parameter description

| 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.<br>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 Recipient | Select **Notification group** or **Topic subscription**. | - |
| Notification Group | Notification group the alarm notification is to be sent to. | - |
| Notification Object | Specifies the object that receives alarm notifications. You can select the account contact or a topic.<br>– Account contact is the mobile phone number and email address provided for registration.<br>– **Topic** is used to publish messages and subscribe to notifications. If the required topic is unavailable, create one first and add subscriptions to it. For details, see **Creating a Topic** and **Adding Subscriptions**. | - |
| Notification Window | Cloud Eye sends notifications only within the notification window specified in the alarm rule.<br>For example, if **Notification Window** is set to **00:00-8:00**, Cloud Eye sends notifications only within 00:00-08:00. | - |

| Parameter | Description | Example Value |
| --- | --- | --- |
| 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 4-114** Advanced settings

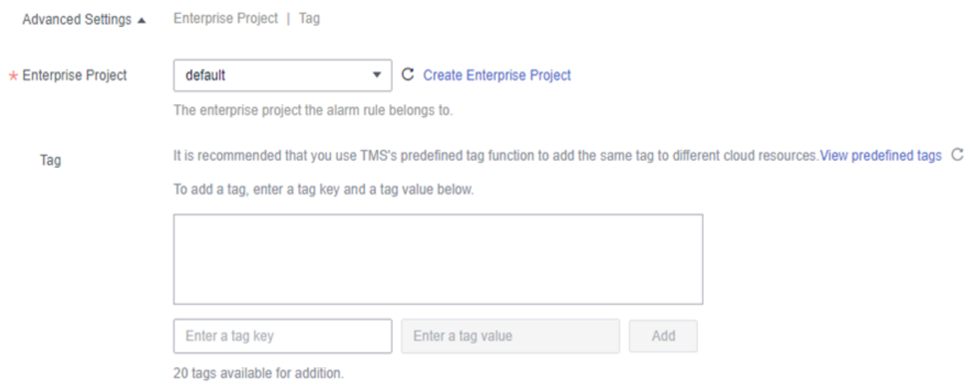



**Table 4-43** Parameter description

| 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. For details about how to create an enterprise project, see **Creating an Enterprise Project**. | default |
| Tag | A tag is a key-value pair. Tags identify cloud resources so that you can easily categorize and search for your resources. You are advised to create predefined tags on TMS. For details about how to create predefined tags, see **Creating Predefined Tags**.<br>– A key can contain a maximum of 128 characters, and a value can contain a maximum of 255 characters.<br>– A maximum of 20 tags can be added. | - |

**Step 6** 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**

## 4.13.3 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

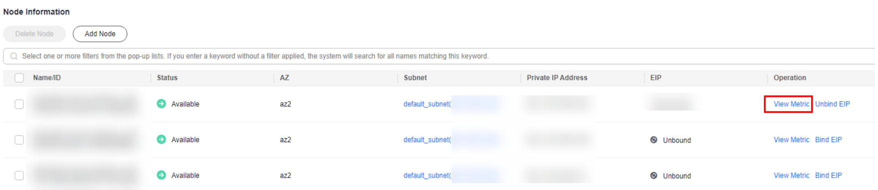
**Step 1** Log in to the Huawei Cloud console.

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** On the **Instance** page, click the instance whose metrics you want to view and click its name.

**Step 4** In the **Node Information** area on the **Basic Information** page, click **View Metric** in the **Operation** column.

**Figure 4-115** Querying monitoring metrics




**Step 5** 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**

## 4.13.4 Event Monitoring

### 4.13.4.1 Introduction to Event Monitoring

Event monitoring provides event data reporting, query, and alarm reporting. You can create alarm rules for both system and custom events. When a specific event occurs, Cloud Eye generates and sends an alarm for you.

Key operations on GeminiDB Cassandra resources are monitored and recorded by Cloud Eye as events. Events include operations performed by specific users on specific resources, such as changing instance names and specifications.

Event monitoring provides an API for reporting custom events, which helps you collect and report abnormal events or important change events generated by services to Cloud Eye.

Event monitoring is enabled by default and allows you to view monitoring details of system events and custom events. For details about system events, see **Events Supported by Event Monitoring**.

NOTE

If you do not create an alarm rule, no alarm will be sent by default.

### 4.13.4.2 Viewing Event Monitoring Data

#### Scenarios

Event monitoring provides event data reporting, query, and alarm reporting. You can create alarm rules for both system and custom events. When a specific event occurs, Cloud Eye generates and sends an alarm for you.

Event monitoring is enabled by default. You can view monitoring details about system events and custom events.

This topic describes how to view the event monitoring data.

#### Procedure

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

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** On the **Instances** page, locate the instance whose event monitoring data you want to view. In the **Node Information** area on the **Basic Information** page, click **View Metric** in the **Operation** column.

**Step 4** Click to return to the Cloud Eye console.

**Step 5** In the navigation pane on the left, choose **Event Monitoring**.

On the displayed **Event Monitoring** page, all system events generated in the last 24 hours are displayed by default.

You can also click **1h**, **3h**, **12h**, **1d**, **7d**, or **30d** to view events generated in different time periods.

**Step 6** Locate an event and click **View Event** in the **Operation** column to view its details.

**----End**

### 4.13.4.3 Creating an Alarm Rule for Event Monitoring

#### Scenarios

This topic describes how to create an alarm rule for event monitoring.

#### Usage Notes

If you do not create an alarm rule, no alarm will be sent by default.

#### Procedure

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

**Step 2** Click ☰ in the upper left corner of the page. Under **Management & Governance**, click **Cloud Eye**.

**Step 3** In the navigation pane on the left, choose **Event Monitoring**.

**Step 4** On the event list page, click **Create Alarm Rule** in the upper right corner.

**Step 5** On the **Create Alarm Rule** page, configure the parameters.

**Table 4-44** Parameter description

| Parameter | Description |
|---|---|
| Name | Specifies the name of the alarm rule. The system generates a random name, but you can change it if needed. |
| Description | (Optional) Provides supplementary information about the alarm rule. |
| Enterprise Project | You can select an existing enterprise project or click **Create Enterprise Project** to create one. |
| Alarm Type | Specifies the alarm type corresponding to the alarm rule. |
| Event Type | Specifies the event type of the metric corresponding to the alarm rule. |
| Event Source | Specifies the service the event is generated for.<br>Select GeminiDB. |

| Parameter | Description |
| --- | --- |
| Monitoring Scope | Specifies the monitoring scope for event monitoring. |
| Method | Specifies the event creation method. |
| Alarm Policy | **Event Name** indicates the instantaneous operations users performed on system resources, such as login and logout.<br>For details about events supported by Event Monitoring, see **Events Supported by Event Monitoring**.<br>You can select a trigger mode and alarm severity as needed. |

Click  to enable alarm notification. The validity period is 24 hours by default. If the topics you require are not displayed in the drop-down list, click **Create an SMN topic**.

**Table 4-45** Alarm notification parameters

| Parameter | Description |
| --- | --- |
| Alarm Notification | Whether to notify users when alarms are triggered. Notifications can be sent by email, text message, or HTTP/ HTTPS message. |
| Notification Object | Object an alarm notification is to be sent to. You can select the account contact or a topic.<br>● Account contact is the mobile phone number and email address provided for registration.<br>● Topic is used to publish messages and subscribe to notifications. If the required topic is unavailable, create one first and add subscriptions to it. For details, see **Creating a Topic** and **Adding Subscriptions**. |
| Validity Period | Notification window which Cloud Eye only sends notifications within.<br>If you set **Validity Period** to **08:00-20:00**, Cloud Eye sends notifications only within 08:00-20:00. |
| Trigger Condition | Condition for triggering the alarm notification. |

**Step 6** After the configuration is complete, click **Create**.

**----End**

### 4.13.4.4 Events Supported by Event Monitoring

**Table 4-46** Events Supported by Event Monitoring for GeminiDB

| Event Source | Event Name | Event ID | Event Severity | Description | Solution | Impact |
| --- | --- | --- | --- | --- | --- | --- |
| NoSQL | Instance creation failure | NoSQLCreateInstanceFailed | Major | The instance quota or underlying resources are insufficient. | Release unnecessary instances and try again. You can also choose **Service Tickets > Create Service Ticket** in the upper right corner of the console and submit a service ticket to adjust the quota. | Instances fail to be created. |
| | Specifications change failure | NoSQLResizeInstanceFailed | Major | The underlying resources are insufficient. | Choose **Service Tickets > Create Service Ticket** in the upper right corner of the console. Submit a service ticket to the O&M personnel to coordinate resources in the background and change the specifications again. | Services are interrupted. |

| Event Source | Event Name | Event ID | Event Severity | Description | Solution | Impact |
| --- | --- | --- | --- | --- | --- | --- |
| | Node adding failure | NoSQLAddNodesFailed | Major | The underlying resources are insufficient. | Choose **Service Tickets > Create Service Ticket** in the upper right corner of the console. Submit a service ticket to O&M personnel to coordinate resources in the background, delete nodes that failed to be added, and add the nodes again. | None |
| | Node deletion failure | NoSQLDeleteNodesFailed | Major | Releasing underlying resources failed. | Delete the node again. | None |
| | Storage space scale-up failure | NoSQLScaleUpStorageFailed | Major | The underlying resources are insufficient. | Choose **Service Tickets > Create Service Ticket** in the upper right corner of the console. Submit a service ticket to O&M personnel to coordinate resources in the background and scale up storage again. | Services may be interrupted. |

| Event Source | Event Name | Event ID | Event Severity | Description | Solution | Impact |
|---|---|---|---|---|---|---|
| | Password resetting failure | NoSQLResetPasswordFailed | Major | Resetting the password times out. | Reset the password again. | None |
| | Parameter template change failure | NoSQLUpdateInstanceParamGroupFailed | Major | Changing a parameter template times out. | Change the parameter template again. | None |
| | Backup policy configuration failure | NoSQLSetBackupPolicyFailed | Major | The database connection is abnormal. | Configure the backup policy again. | None |
| | Manual backup creation failure | NoSQLCreateManualBackupFailed | Major | The backup files fail to be exported or uploaded. | Choose **Service Tickets > Create Service Ticket** in the upper right corner of the console and submit a service ticket to O&M personnel. | Data cannot be backed up. |
| | Automated backup creation failure | NoSQLCreateAutomatedBackupFailed | Major | The backup files fail to be exported or uploaded. | Choose **Service Tickets > Create Service Ticket** in the upper right corner of the console and submit a service ticket to O&M personnel. | Data cannot be backed up. |

| Event Source | Event Name | Event ID | Event Severity | Description | Solution | Impact |
|---|---|---|---|---|---|---|
| | Instance status abnormal | NoSQLFaultyDBInstance | Major | This event is a key alarm event and is reported when an instance is faulty due to a disaster or a server failure. | Choose **Service Tickets > Create Service Ticket** in the upper right corner of the console and submit a service ticket to O&M personnel. | The database service may be unavailable. |
| | Instance status recovery | NoSQLDBInstanceRecovered | Major | If a disaster occurs, NoSQL provides an HA tool to automatically or manually rectify the fault. After the fault is rectified, this event is reported. | No further action is required. | None |
| | Node status abnormal | NoSQLFaultyDBNode | Major | This event is a key alarm event and is reported when a database node is faulty due to a disaster or a server failure. | Check whether the database service is functional. Choose **Service Tickets > Create Service Ticket** in the upper right corner of the console and submit a service ticket to O&M personnel. | The database service may be unavailable. |

| Event Source | Event Name | Event ID | Event Severity | Description | Solution | Impact |
| --- | --- | --- | --- | --- | --- | --- |
| | Node status recovery | NoSQLDBNodeRecovered | Major | If a disaster occurs, NoSQL provides an HA tool to automatically or manually rectify the fault. After the fault is rectified, this event is reported. | No further action is required. | None |
| | Primary/standby switchover or failover | NoSQLPrimaryStandbySwitched | Major | This event is reported when a primary/secondary switchover or a failover is triggered. | No further action is required. | None |
| | Occurrence of hotspot partitioning keys | HotKeyOccurs | Major | Hotspot data is stored in one partition because the primary key is improper. Improper application design causes frequent read and write operations on a key. | 1. Choose a proper partition key.<br>2. Add service cache so that service applications read hotspot data from the cache first. | The service request success rate is affected, and the cluster performance and stability deteriorates. |

| Event Source | Event Name | Event ID | Event Severity | Description | Solution | Impact |
| --- | --- | --- | --- | --- | --- | --- |
| | BigKey occurrence | BigKey Occurs | Major | The primary key design is improper. There are too many records or too much data in a single partition, causing load imbalance on nodes. | 1. Choose a proper partition key.<br>2. Add a new partition key for hashing data. | As more and more data is stored in the partition, cluster stability deteriorates. |
| | Insufficient storage space | NoSQLRiskyDataDiskUsage | Major | The storage space is insufficient. | Scale up storage space. For details, see section "Scaling Up Storage Space" in the user guide of GeminiDB. | The instance is set to read-only and data cannot be written to the instance. |
| | Data disk expanded and being writable | NoSQLDataDiskUsageRecovered | Major | The data disk has been expanded and becomes writable. | No further action is required. | None |

| Event Source | Event Name | Event ID | Event Severity | Description | Solution | Impact |
| --- | --- | --- | --- | --- | --- | --- |
| | Index creation failure | NoSQLCreateIndexFailed | Major | The service load exceeds what the instance specifications can take. In this case, creating indexes consumes more instance resources. As a result, the response is slow or even frame freezing occurs, and the creation times out. | 1. Select matched instance specifications based on the service loads.<br>Create indexes during off-peak hours.<br>Create indexes in the background.<br>Select indexes as required. | The index fails to be created or is incomplete. Delete the index and create a new one. |
| | Write speed decrease | NoSQLStallingOccurs | Major | The write speed is close to the maximum write speed allowed by the cluster scale and instance specifications. As a result, the database flow control mechanism is triggered, and requests may fail. | 1. Adjust the cluster scale or node specifications based on the maximum write rate of services.<br>2. Measure the maximum write request rate of services and distribute the peak write rate of services. | The success rate of service requests is affected. |

| Event Source | Event Name | Event ID | Event Severity | Description | Solution | Impact |
| --- | --- | --- | --- | --- | --- | --- |
| | Data write stopped | NoSQLStoppingOccurs | Major | The data write is too fast, reaching the maximum write capability allowed by the cluster scale and instance specifications. As a result, the database flow control mechanism is triggered, and requests may fail. | 1. Adjust the cluster scale or node specifications based on the maximum write rate of services.<br>2. Measure the maximum write request rate of services and distribute the peak write rate of services. | The success rate of service requests is affected. |
| | Database restart failure | NoSQLRestartDBFailed | Major | The instance status is abnormal. | Choose **Service Tickets > Create Service Ticket** in the upper right corner of the console and submit a service ticket to O&M personnel. | The instance status may be abnormal. |

| Event Source | Event Name | Event ID | Event Severity | Description | Solution | Impact |
|---|---|---|---|---|---|---|
| | Restoration to new instance failure | NoSQLRestoreToNewInstanceFailed | Major | The underlying resources are insufficient. | Choose **Service Tickets > Create Service Ticket** in the upper right corner of the console. Submit a service ticket to O&M personnel to coordinate resources in the background and add nodes again. | Data cannot be restored to a new instance. |
| | Restoration to existing instance failure | NoSQLRestoreToExistInstanceFailed | Major | The backup file fails to be downloaded or restored. | Choose **Service Tickets > Create Service Ticket** in the upper right corner of the console and submit a service ticket to O&M personnel. | The current instance may be unavailable. |
| | Backup file deletion failure | NoSQLDeleteBackupFailed | Major | The backup files fail to be deleted from OBS. | Delete the backup files again. | None |

| Event Source | Event Name | Event ID | Event Severity | Description | Solution | Impact |
| --- | --- | --- | --- | --- | --- | --- |
| | Failure to display slow query logs in plaintext | NoSQLSwitchSlowlogPlainTextFailed | Major | The DB API does not support this function. | Refer to *GeminiDB User Guide* to ensure that the API supports slow query logs in plaintext. Choose **Service Tickets > Create Service Ticket** in the upper right corner of the console and submit a service ticket to O&M personnel. | None |
| | EIP binding failure | NoSQLBindEipFailed | Major | The node status is abnormal, an EIP has been bound to the node, or the EIP to be bound is invalid. | Check whether the node is normal and whether the EIP is valid. | The instance cannot be accessed from a public network. |
| | EIP unbinding failure | NoSQLUnbindEipFailed | Major | The node status is abnormal or the EIP has been unbound from the node. | Check whether the node and EIP status are normal. | None |

| Event Source | Event Name | Event ID | Event Severity | Description | Solution | Impact |
|---|---|---|---|---|---|---|
| | Parameter modification failure | NoSQLModifyParameterFailed | Major | The parameter value is invalid. | Check whether the parameter value is valid. Choose **Service Tickets > Create Service Ticket** in the upper right corner of the console and submit a service ticket to O&M personnel. | None |
| | Parameter template application failure | NoSQLApplyParameterGroupFailed | Major | The instance status is abnormal. So, the parameter template cannot be applied. | Choose **Service Tickets > Create Service Ticket** in the upper right corner of the console and submit a service ticket to O&M personnel. | None |

| Event Source | Event Name | Event ID | Event Severity | Description | Solution | Impact |
|---|---|---|---|---|---|---|
| | Enabling or disabling SSL failure | NoSQLSwitchSSLFailed | Major | Enabling or disabling SSL times out. | Try again or choose **Service Tickets > Create Service Ticket** in the upper right corner of the console and submit a service ticket. Retain the SSL connection mode configured before the event occurred. | The SSL connection mode cannot be changed. |
| | Too much data in a single row | LargeRowOccurs | Major | If there is too much data in a single row, queries may time out, causing faults like OOM error. | 1. Limit the write length of each column and row so that the key and value length of each row does not exceed the preset threshold.<br>2. Check whether there are abnormal writes or coding, causing large rows. | If there are too many records in a single row, cluster stability will deteriorate as the data volume increases. |

| Event Source | Event Name | Event ID | Event Severity | Description | Solution | Impact |
| --- | --- | --- | --- | --- | --- | --- |
| | Schedule for deleting a KMS key | planDeleteKmsKey | Major | The user plans to delete a KMS key. | Check whether the GeminiDB instance associated with the key has been deleted or is no longer used. Deleting the key will affect the instance services. | The key will be automatically deleted after it expires. Deleting the key will affect the instance services. |
| | Too many tombstones | TooManyQueryTombstones | Major | Querying too many tombstones may time out. | Use a proper query and deletion method to avoid batch range queries. | The query may time out. |
| | Ultra-large collection column | TooLargeCollectionColumn | Major | If there are too many elements in the collection column, the query will fail. | Set a threshold for the number of elements in the collection column.<br>Check whether there is an error while data is written and encoded. | The query on the collection column will fail. |

# 4.14 Enterprise Project

## 4.14.1 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***.

## 4.14.2 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.

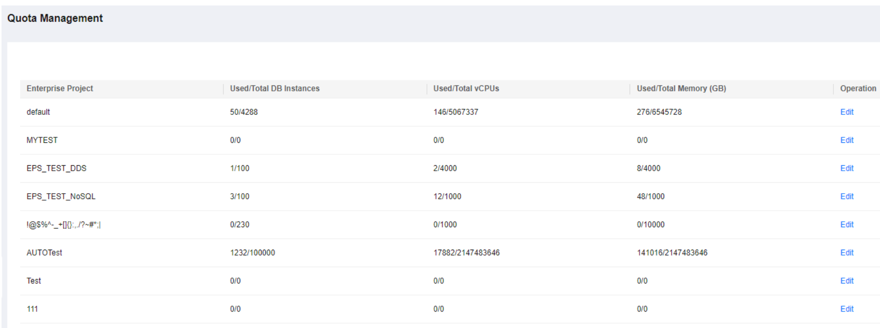
### Viewing Resource Quotas in Each Enterprise Project

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

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** In the navigation pane on the left, choose **My Quotas** to view quota details of the current enterprise project.

**Figure 4-116** Quota management



Quota Management

| Enterprise Project | Used/Total DB Instances | Used/Total vCPUs | Used/Total Memory (GB) | Operation |
|---|---|---|---|---|
| default | 50/4288 | 146/5067337 | 276/6545728 | Edit |
| MYTEST | 0/0 | 0/0 | 0/0 | Edit |
| EPS_TEST_DDS | 1/100 | 2/4000 | 8/4000 | Edit |
| EPS_TEST_NoSQL | 3/100 | 12/1000 | 48/1000 | Edit |
| !@$%^-_+[]{}:,./?~#*;\| | 0/230 | 0/1000 | 0/10000 | Edit |
| AUTOTest | 1232/100000 | 17882/2147483646 | 141016/2147483646 | Edit |
| Test | 0/0 | 0/0 | 0/0 | Edit |
| 111 | 0/0 | 0/0 | 0/0 | Edit |

**Table 4-47** Parameter description

| Parameter | Description |
|---|---|
| Enterprise Project | Enterprise project that an instance belongs to. |

| Parameter | Description |
| --- | --- |
| 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 service list, choose **Databases** > **GeminiDB**.

**Step 3** 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 4-117** Managing quotas

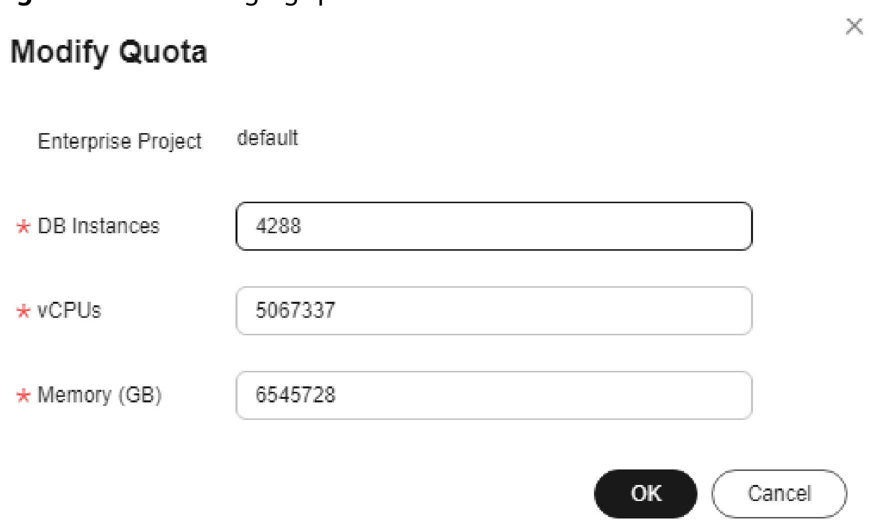


**Table 4-48** Quota management

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

**----End**

# 4.15 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 4-49**.
- A maximum of 20 tags can be added for each instance.
- The tag name must comply with the naming rules described in **Table 4-49**.

**Table 4-49** Naming rules

| Parameter | Requirement | Example Value |
| --- | --- | --- |
| Tag key | • Cannot be left blank.<br>• Must be unique for each instance.<br>• Can contain a maximum of 128 characters.<br>• Cannot start with **_sys_** and cannot start or end with a space. Only letters, digits, spaces, and the following special characters are allowed: -_@.:/+= | Organization |

| Parameter | Requirement | Example Value |
| --- | --- | --- |
| Tag value | • Can be left blank.<br>• Can contain a maximum of 255 characters.<br>• Only letters, digits, spaces, and the following special characters are allowed: -_@.:/+= | nosql_01 |

## Adding a Tag

**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 target instance. The **Basic Information** page is displayed.

**Step 4** In the navigation pane on the left, click **Tags**.

**Step 5** On the **Tags** page, click **Add Tag**. In the displayed dialog box, enter a tag key and value, and click **OK**.

**Step 6** View and manage tags on the **Tags** page.

**----End**

## Editing a Tag

**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 target instance. The **Basic Information** page is displayed.

**Step 4** In the navigation pane on the left, click **Tags**.

**Step 5** 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 6** View and manage tags on the **Tags** page.

**----End**

## Deleting a Tag

**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 target instance. The **Basic Information** page is displayed.

**Step 4** In the navigation pane on the left, click **Tags**.

**Step 5** 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 6** Check whether the deleted tag is displayed on the **Tags** page.

**----End**

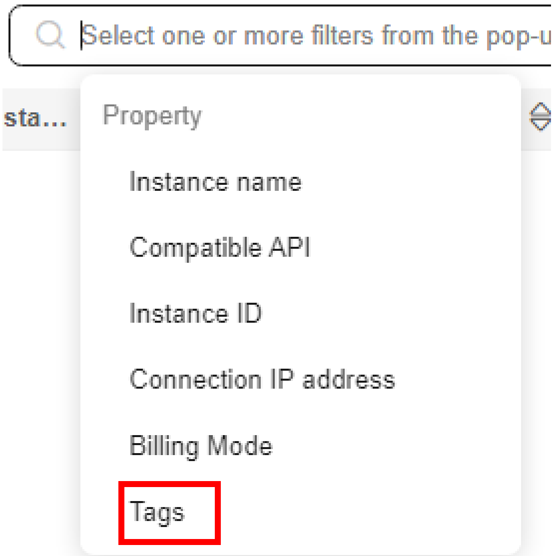
## Searching by tag

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

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

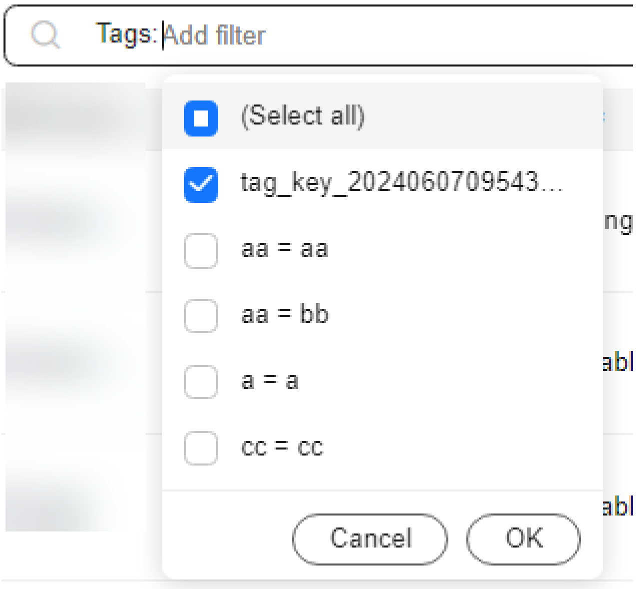
**Step 3** On the **Instances** page, select **Tags** in the search box.

**Figure 4-118** Selecting tags




**Step 4** Select the tag to be queried and click **OK** to query information about instances associated with the tag.

**Figure 4-119** Searching by tag




**----End**

# 4.16 User Resource Quotas

## Scenarios

Quotas are enforced for service resources on the platform to prevent unforeseen spikes in resource usage. Quotas limit the number or amount of resources available to users, for example, the maximum number of GeminiDB instances that you can create.

If a quota cannot meet your needs, apply for a higher quota.

## Checking Quotas

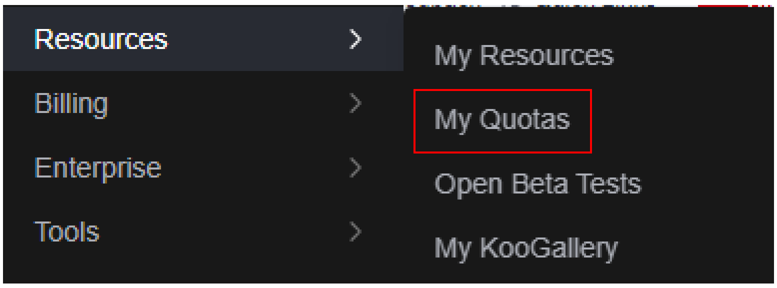
**Step 1** **Log in to the Huawei Cloud console.**

**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** Click in the upper left corner and select a region and project.

**Step 4** In the upper right corner, choose **Resources** > **My Quotas**.

**Figure 4-120** My quotas



**Step 5** On the displayed page, check the used and total quotas of each type of GeminiDB instance resources.

**----End**

## Increasing Quotas

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

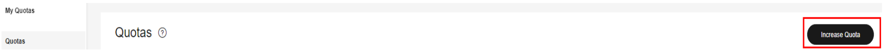
**Step 2** In the service list, choose **Databases** > **GeminiDB**.

**Step 3** Click in the upper left corner and select a region and project.

**Step 4** In the upper right corner, choose **Resources** > **My Quotas**.

**Step 5** In the upper right corner of the page, click **Increase Quota**.

**Figure 4-121** Increasing quotas




**Step 6** On the **Create Service Ticket** page, configure parameters.

In the **Problem Description** area, describe why you need the adjustment.

**Step 7** After all mandatory parameters are configured, read and agree to the agreement and click **Submit**.

**----End**

# 5 Best Practices

## 5.1 Performance Comparison Between GeminiDB Cassandra API and Open-Source Cassandra

This section describes how the performance of an open-source Cassandra cluster compares to a GeminiDB Cassandra cluster. The test environment, test model, and test steps will all be described.

### Test Environment

- Open-source Cassandra test environment

**Table 5-1** Test environment description

| | |
|---|---|
| **Name** | Open-Source Cassandra cluster |
| **Version** | 3.11.5 |
| **Nodes** | 3 |
| **OS** | CentOS 7.4 |
| **ECS Specifications** | • General computing-plus 4 vCPUs \| 16 GB<br>• General computing-plus 8 vCPUs \| 32 GB<br>• General computing-plus 16 vCPUs \| 64 GB<br>• General computing-plus 32 vCPUs \| 128 GB |

- GeminiDB Cassandra test environment

**Table 5-2** Test environment description

| | |
|---|---|
| **Name** | GeminiDB Cassandra Cluster |
| **Region** | CN-Hong Kong |

| | |
|---|---|
| **Nodes** | 3 |
| **AZ** | AZ 3 |
| **Version** | 3.11 |
| **Instance Specifications** | • 4 vCPUs \| 16 GB<br>• 8 vCPUs \| 32 GB<br>• 16 vCPUs \| 64 GB<br>• 32 vCPUs \| 128 GB |

## Load Test Tool Environment

- Load test tool specifications

**Table 5-3** Specifications

| | |
|---|---|
| **Name** | Test client ECS |
| **vCPUs** | 16 |
| **Memory** | 64 GB |
| **OS** | CentOS 7.4 |

- Load test tool information

**Table 5-4** Load test tool information

| | |
|---|---|
| **Test Tool** | YCSB |
| **Version** | 0.12.0 |
| **Download Address** | **https://github.com/brianfrankcooper/YCSB**<br>**curl -O --location https://github.com/brianfrankcooper/YCSB/releases/download/0.12.0/ycsb-0.12.0.tar.gz** |

## Testing Models

**Table 5-5** Testing models

| Service Model | Description |
|---|---|
| _read95_update5 | 95% read and 5% update |
| _update50_read50 | 50% update and 50% read |
| _read65_update25_insert10 | 65% read, 25% update, and 10% write |

| Service Model | Description |
|---|---|
| _insert90_read10 | 90% write and 10% read |

## Test Procedure

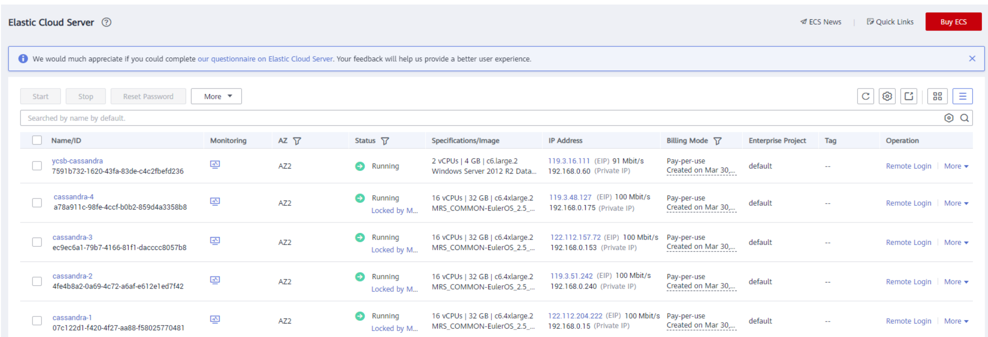
Testing Open-Source Cassandra

**Step 1** Buy an ECS.

1. Log in to the Huawei Cloud console.
2. Choose **Computing** > **Elastic Cloud Server**.
3. Click **Buy ECS** in the upper right corner of the page and configure related parameters as follows:
   - **Region**: CN-Hong Kong
   - **AZ**: **AZ3**
   - **Specifications**: **General computing-plus | c6.xlarge.4**
   - **Image**: **Public image** and **CentOS 7.6 64bit(40 GB)**
   - **Data Disk**: **Ultra-high I/O** and **200 GB**
   - **Network**: Select a VPC and subnet.
   - Other parameters: Set other parameters as needed. You can ignore optional parameters.
4. Repeat the preceding steps to create five ECSs named **Cassandra-1** (192.168.0.15), **Cassandra-2** (192.168.0.240), **Cassandra-3** (192.168.0.153), **Cassandra-4** (192.168.0.175) and **ycsb-Cassandra** (192.168.0.60).

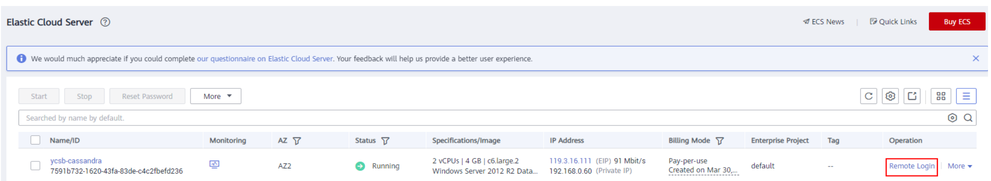
   ECSs **Cassandra-1**, **Cassandra-2**, and **Cassandra-3** are for initializing Cassandra clusters. ECS **Cassandra-4** is for capacity expansion. ECS **ycsb-Cassandra** serves as the load test server.

   **Figure 5-1** ECS details

5. After those ECSs are created, log in to them using the remote login option provided on the management console.

   **Figure 5-2** Logging in to an ECS

6. Install Java Runtime Environment:

   **yum install jre**

7. Install the Cassandra service and create a data directory.

   a. Download the Cassandra installation package:

      **wget https://archive.apache.org/dist/cassandra/3.11.5/apache-cassandra-3.11.5-bin.tar.gz**

   b. Decompress the installation package:

      **tar -zxvf apache-Cassandra-3.11.5-bin.tar.gz -C /root/**

   c. Change the installation directory:

      **mv /root/apache-Cassandra-3.11.5 /usr/local/Cassandra**

   d. Configure environment variables:

      **echo "export PATH=/usr/local/Cassandra/bin:$PATH" >> /etc/profile**

   e. Apply the variables:

      **source /etc/profile**

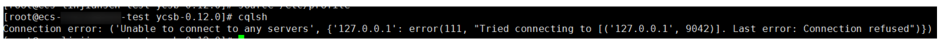
   f. Create a data directory:

      **mkdir /data**

   g. Confirm that the installation was successful.

      **cqlsh**

      **Figure 5-3** Successful installation

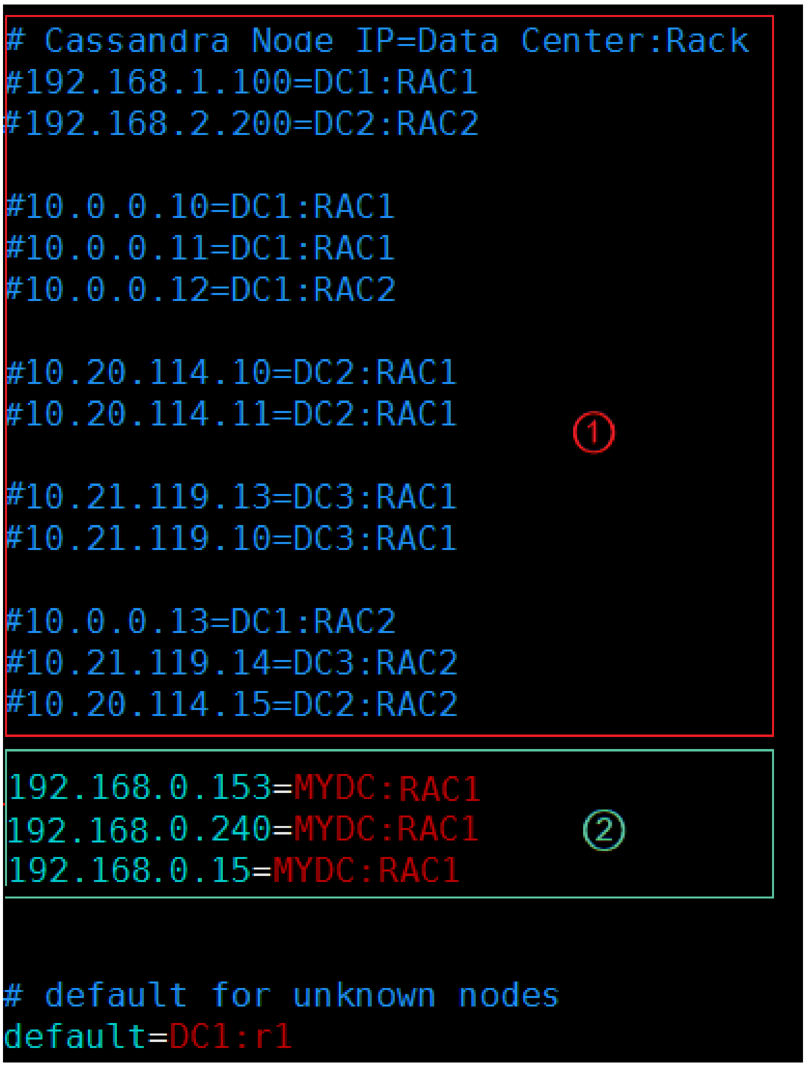



**Step 2** Configure an open-source Cassandra cluster.

1. Log in to ECSs **Cassandra-1**, **Cassandra-2**, and **Cassandra-3**.
2. Go to the **/usr/local/Cassandra/conf** directory and modify the **Cassandra-topology.properties** file as follows:
   - Comment out the content in the area marked by No.1 in **Figure 5-4**.
   - Add the content in the area marked by No.2 in **Figure 5-4**.

**Figure 5-4** Modifying the configuration file




**NOTE**

The **Cassandra-topology.properties** configuration files of **Cassandra-1**, **Cassandra-2**, and **Cassandra-3** must be the same.

3. Modify the **Cassandra.yaml** file as follows:

```
data_file_directories:
- /data
commitlog_directory: /usr/local/Cassandra/commitlog
saved_caches_directory: /usr/local/Cassandra/saved_caches
seed_provider:
# Addresses of hosts that are deemed contact points.
# Cassandra nodes use this list of hosts to find each other and learn
# the topology of the ring.  You must change this if you are running
# multiple nodes!
- class_name: org.apache.Cassandra.locator.SimpleSeedProvider
parameters:
# seeds is actually a comma-delimited list of addresses.
# Ex: "<ip1>,<ip2>,<ip3>"
- seeds: "192.168.0.153,192.168.0.240,192.168.0.15" ##Enter IP addresses of the three nodes in the cluster.
listen_address: 192.168.0.153       #IP address of each node
rpc_address: 192.168.0.153          #IP address of each node
```

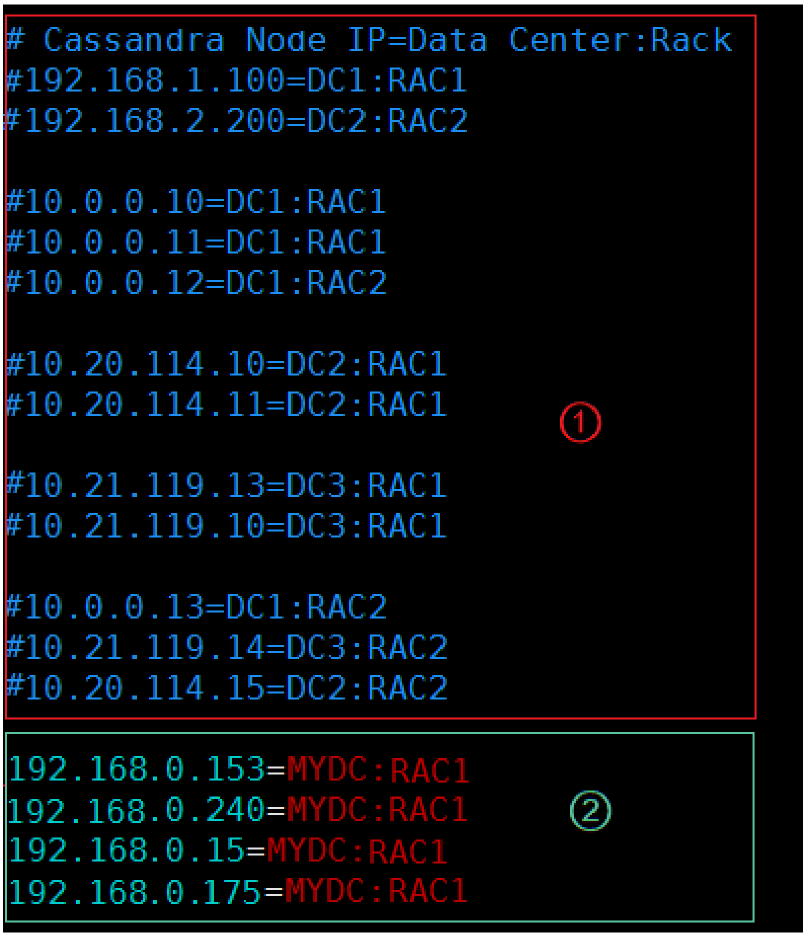
4. Run the following command on **Cassandra-1**, **Cassandra-2**, and **Cassandra-3** to start the Cassandra cluster:

   **Cassandra –R &**

**Step 3** Add nodes to the open-source Cassandra cluster.

1. Log in to **Cassandra-4**.
2. Go to the **/usr/local/cassandra/conf** directory and edit the **Cassandra-topology.properties** file as follows:
   - Comment out the content in the area marked by No.1 in **Figure 5-5**.
   - Add the content in the area marked by No.2 in **Figure 5-5**.

**Figure 5-5** Editing the configuration file




3. Modify the **Cassandra.yaml** file as follows:

```
data_file_directories:
- /data
commitlog_directory: /usr/local/Cassandra/commitlog
saved_caches_directory: /usr/local/Cassandra/saved_caches
seed_provider:
# Addresses of hosts that are deemed contact points.
# Cassandra nodes use this list of hosts to find each other and learn
# the topology of the ring.  You must change this if you are running
# multiple nodes!
- class_name: org.apache.Cassandra.locator.SimpleSeedProvider
parameters:
# seeds is actually a comma-delimited list of addresses.
# Ex: "<ip1>,<ip2>,<ip3>"
```

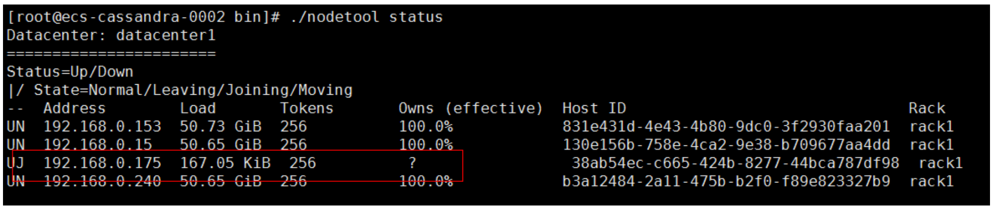
```
- seeds: "192.168.0.153,192.168.0.240,192.168.0.15" ## Enter IP addresses of the three seed nodes in the cluster, which must be the same as the values entered in step 1.
listen_address: 192.168.0.175      #IP address of each node
rpc_address: 192.168.0.175         #IP address of each node
```

4. Log in to **Cassandra-1**.
5. Stop compaction on all nodes:
   **nodetool disableautocompaction**
6. Stop the ongoing compaction task:
   **nodetool stop COMPACTION**
7. Limit migration traffic of the node:
   **nodetool setstreamthroughput 32**

   **NOTE**

   In the preceding command, the value of **nodetool setstreamthroughput 32** is set to **32 MB/s** to reduce the impact of migration on services.
8. Log in to **Cassandra-4**.
9. Start the Cassandra service:
   **Cassandra –R &**
10. Log in to **Cassandra-1**.
11. During the scaling, run the following command every 30 seconds:
    **nodetool status**
    If the status of **Cassandra-4** is **UJ**, data is being migrated. The migration is complete when the status changes to **UN**.

**Figure 5-6** Node statuses



```
[root@ecs-cassandra-0002 bin]# ./nodetool status
Datacenter: datacenter1
=======================
Status=Up/Down
|/ State=Normal/Leaving/Joining/Moving
--  Address        Load        Tokens       Owns (effective)  Host ID                               Rack
UN  192.168.0.153  50.73 GiB  256          100.0%            831e431d-4e43-4b80-9dc0-3f2930faa201  rack1
UN  192.168.0.15   50.65 GiB  256          100.0%            130e156b-758e-4ca2-9e38-b709677aa4dd  rack1
UJ  192.168.0.175  167.05 KiB  256          ?                38ab54ec-c665-424b-8277-44bca787df98  rack1
UN  192.168.0.240  50.65 GiB  256          100.0%            b3a12484-2a11-475b-b2f0-f89e823327b9  rack1
```

**----End**

Testing a GeminiDB Cassandra Instance

**Step 1** Buy a GeminiDB Cassandra cluster instance.

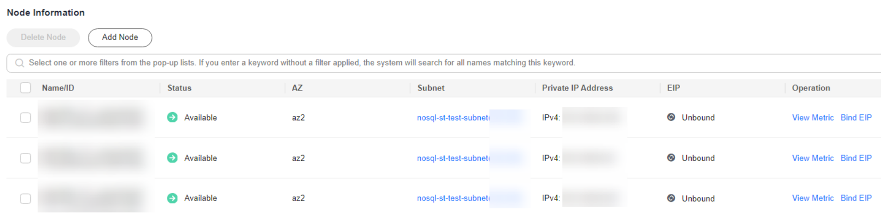
1. **Log in to the Huawei Cloud console**.
2. Choose **Databases** > **GeminiDB**.
3. Click **Buy DB Instance** in the upper right corner of the page and set required parameters as follows:
   – **Region**: CN-Hong Kong
   – **Compatible API**: **Cassandra**
   – **Specifications**: **4 vCPUs | 16 GB**
   – **Storage Space**: **200 GB**
   – **Nodes**: Enter **3**.

- **VPC**: The same as that of the purchased ECS.
- **Security Group**: The same as that of the purchased ECS.

**Step 2** Add nodes to the GeminiDB Cassandra cluster.

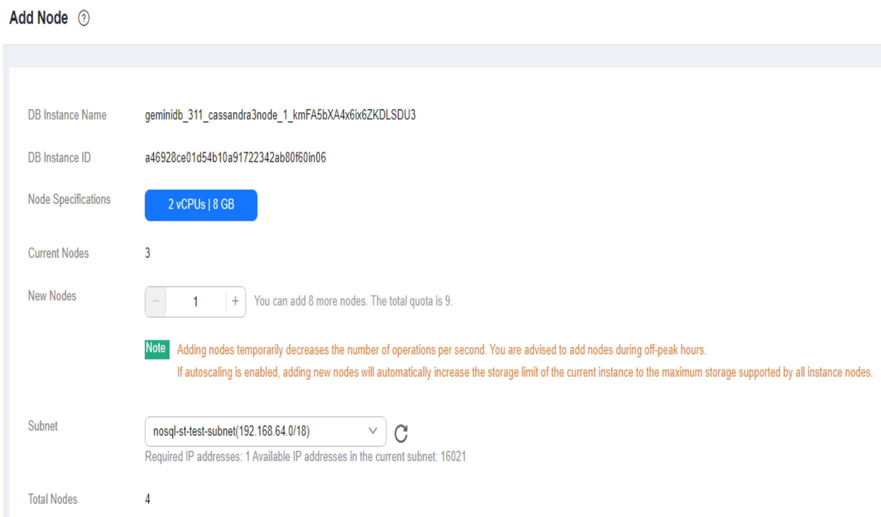
1. **Log in to the Huawei Cloud console**.
2. Choose **Databases** > **GeminiDB**.
3. Select an existing GeminiDB Cassandra instance.
4. Click the instance name to enter the **Basic Information** page.
5. In the **Node Information** area on the **Basic Information** page, click **Add Node**.

**Figure 5-7** Node information



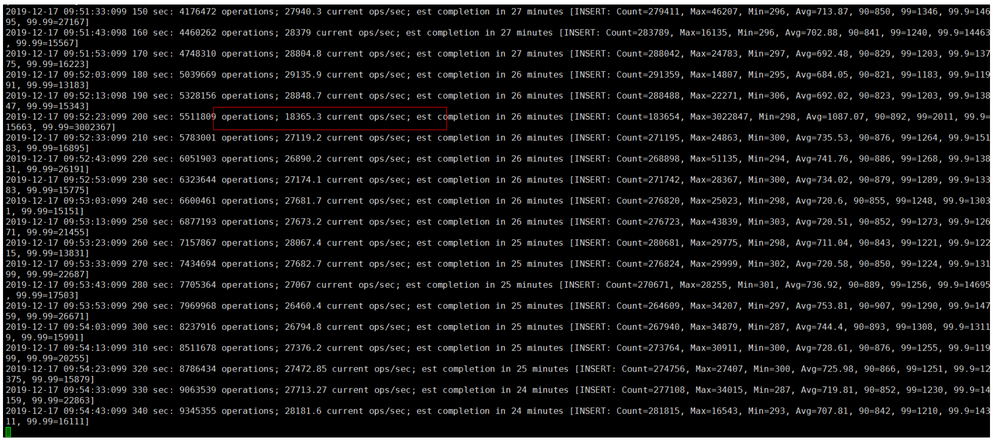

6. On the displayed page, click **+** on the right of field **Add Nodes** .

**Figure 5-8** Adding nodes




7. Wait until the nodes are added.
8. View the change of QPS during the scale-out process.

**Figure 5-9** QPS changes



```
2019-12-17 09:51:33:099 150 sec: 4176472 operations; 27940.3 current ops/sec; est completion in 27 minutes [INSERT: Count=279411, Max=46207, Min=296, Avg=713.87, 90=850, 99=1346, 99.9=14695, 99.99=27167]
2019-12-17 09:51:43:098 160 sec: 4460262 operations; 28379 current ops/sec; est completion in 27 minutes [INSERT: Count=283789, Max=16135, Min=296, Avg=702.88, 90=841, 99=1240, 99.9=14463, 99.99=15567]
2019-12-17 09:51:53:099 170 sec: 4748310 operations; 28804.8 current ops/sec; est completion in 27 minutes [INSERT: Count=288042, Max=24783, Min=297, Avg=692.48, 90=829, 99=1203, 99.9=13775, 99.99=16223]
2019-12-17 09:52:03:099 180 sec: 5039669 operations; 29135.9 current ops/sec; est completion in 26 minutes [INSERT: Count=291359, Max=14807, Min=295, Avg=684.05, 90=821, 99=1183, 99.9=11991, 99.99=13183]
2019-12-17 09:52:13:098 190 sec: 5328156 operations; 28848.7 current ops/sec; est completion in 26 minutes [INSERT: Count=288488, Max=22271, Min=306, Avg=692.02, 90=823, 99=1203, 99.9=13847, 99.99=15343]
2019-12-17 09:52:23:099 200 sec: 5511809 operations; 18365.3 current ops/sec; est completion in 26 minutes [INSERT: Count=183654, Max=3022847, Min=298, Avg=1087.07, 90=892, 99=2011, 99.9=15663, 99.99=3002367]
2019-12-17 09:52:33:099 210 sec: 5783001 operations; 27119.2 current ops/sec; est completion in 26 minutes [INSERT: Count=271195, Max=24863, Min=300, Avg=735.53, 90=876, 99=1264, 99.9=15183, 99.99=16895]
2019-12-17 09:52:43:099 220 sec: 6051903 operations; 26890.2 current ops/sec; est completion in 26 minutes [INSERT: Count=268898, Max=51135, Min=294, Avg=741.76, 90=886, 99=1268, 99.9=13831, 99.99=26191]
2019-12-17 09:52:53:099 230 sec: 6323644 operations; 27174.1 current ops/sec; est completion in 26 minutes [INSERT: Count=271742, Max=28367, Min=300, Avg=734.02, 90=879, 99=1289, 99.9=13383, 99.99=15775]
2019-12-17 09:53:03:099 240 sec: 6600461 operations; 27681.7 current ops/sec; est completion in 26 minutes [INSERT: Count=276820, Max=25023, Min=298, Avg=720.6, 90=855, 99=1248, 99.9=13031, 99.99=15151]
2019-12-17 09:53:13:099 250 sec: 6877193 operations; 27673.2 current ops/sec; est completion in 26 minutes [INSERT: Count=276723, Max=43839, Min=303, Avg=720.51, 90=852, 99=1273, 99.9=12671, 99.99=21455]
2019-12-17 09:53:23:099 260 sec: 7157867 operations; 28067.4 current ops/sec; est completion in 25 minutes [INSERT: Count=280681, Max=29775, Min=298, Avg=711.04, 90=843, 99=1221, 99.9=12215, 99.99=13831]
2019-12-17 09:53:33:099 270 sec: 7434694 operations; 27682.7 current ops/sec; est completion in 25 minutes [INSERT: Count=276824, Max=29999, Min=302, Avg=720.58, 90=850, 99=1224, 99.9=13199, 99.99=22687]
2019-12-17 09:53:43:099 280 sec: 7705364 operations; 27067 current ops/sec; est completion in 25 minutes [INSERT: Count=270671, Max=28255, Min=301, Avg=736.92, 90=889, 99=1256, 99.9=14695, 99.99=17503]
2019-12-17 09:53:53:099 290 sec: 7969968 operations; 26460.4 current ops/sec; est completion in 25 minutes [INSERT: Count=264609, Max=34207, Min=297, Avg=753.81, 90=907, 99=1290, 99.9=14759, 99.99=26671]
2019-12-17 09:54:03:099 300 sec: 8237916 operations; 26794.8 current ops/sec; est completion in 25 minutes [INSERT: Count=267940, Max=34879, Min=287, Avg=744.4, 90=893, 99=1308, 99.9=13119, 99.99=15991]
2019-12-17 09:54:13:099 310 sec: 8511678 operations; 27376.2 current ops/sec; est completion in 25 minutes [INSERT: Count=273764, Max=30911, Min=300, Avg=728.61, 90=876, 99=1255, 99.9=11999, 99.99=20255]
2019-12-17 09:54:23:099 320 sec: 8786434 operations; 27472.85 current ops/sec; est completion in 25 minutes [INSERT: Count=274756, Max=27407, Min=300, Avg=725.98, 90=866, 99=1251, 99.9=12375, 99.99=15879]
2019-12-17 09:54:33:099 330 sec: 9063539 operations; 27713.27 current ops/sec; est completion in 24 minutes [INSERT: Count=277108, Max=34015, Min=287, Avg=719.81, 90=852, 99=1230, 99.9=14159, 99.99=22863]
2019-12-17 09:54:43:099 340 sec: 9345355 operations; 28181.6 current ops/sec; est completion in 24 minutes [INSERT: Count=281815, Max=16543, Min=293, Avg=707.81, 90=842, 99=1210, 99.9=14311, 99.99=16111]
```

During the scale-out process, **the QPS of the GeminiDB Cassandra instance decreases slightly for about 10 seconds, which almost has no effect on services**. The whole scaling process takes about 10 minutes.

After the scale-out is complete, you can analyze test data.

**----End**

## Test Results

- Performance results

**Table 5-6** Performance data

| qps_avg Statistics | Node Class | Concurrent Threads of the Client | Data Volume to Be Prepared | _read95_update5 | _update50_read50 | _read65_update25_insert10 | _insert90_read10 |
|---|---|---|---|---|---|---|---|
| Open-source Cassandra cluster | 4 vCPUs \| 6 GB | 32 | 50 | 2884 | 5068 | 8484 | 10694 |
| | 8 vCPUs \| 32 GB | 64 | 100 | 2796 | 2904 | 5180 | 7854 |
| | 16 vCPUs \| 64 GB | 128 | 200 | 5896 | 14776 | 14304 | 15707 |

| qps_avg Statistics | Node Class | Concurrent Threads of the Client | Data Volume to Be Prepared | _read95_update5 | _update50_read50 | _read65_update25_insert10 | _insert90_read10 |
|---|---|---|---|---|---|---|---|
| | 32 vCPUs \| 128 GB | 256 | 400 | 8964 | 22284 | 19592 | 22344 |
| GeminiDB Cassandra cluster performance data | 4 vCPUs \| 6 GB | 32 | 50 | 8439 | 10565 | 9468 | 23830 |
| | 8 vCPUs \| 32 GB | 64 | 100 | 24090 | 24970 | 21716 | 44548 |
| | 16 vCPUs \| 64 GB | 128 | 200 | 48985 | 51335 | 43557 | 67290 |
| | 32 vCPUs \| 128 GB | 256 | 400 | 91280 | 85748 | 74313 | 111540 |
| Performance comparison between GeminiDB Cassandra API and open-source Cassandra | 4 vCPUs \| 6 GB | 32 | 50 | 2.93 | 2.08 | 1.12 | 2.23 |
| | 8 vCPUs \| 32 GB | 64 | 100 | 8.62 | 8.60 | 4.19 | 5.67 |
| | 16 vCPUs \| 64 GB | 128 | 200 | 8.31 | 3.47 | 3.05 | 4.28 |

| qps_avg Statistics | Node Class | Concurrent Threads of the Client | Data Volume to Be Prepared | _read95_update5 | _update50_read50 | _read65_update25_insert10 | _insert90_read10 |
|---|---|---|---|---|---|---|---|
| | 32 vCPUs \| 128 GB | 256 | 400 | 10.18 | 3.85 | 3.79 | 4.99 |

- Test Conclusion
  a. The GeminiDB Cassandra cluster performs ten times better than the open-source Cassandra cluster in terms of read latency.
  b. GeminiDB Cassandra cluster gives you basically the same write performance as the open-source cluster.
  c. Adding nodes slightly affects both the GeminiDB Cassandra and open-source clusters.
     - The scale-out for GeminiDB Cassandra is fast and only affects services briefly (10s). You do not need to change parameters, and the scale-out process takes 10 minutes.
     - For an open-source Cassandra cluster, the time needed for adding nodes depends on the data volume and parameter settings, and the impact on performance varies. In this test, the scale-out took more than 30 minutes when the preset data size was 50 GB.
     - Calculation formula: Highest migration speed = (**nodetool setstreamthroughput 32** value, 200 Mbit/s by default) x Original nodes

       In this test, the highest migration speed = 32 Mbit/s x 3 = 12 MB/s = 720 MB/min = 0.703 GB/min. So, the time needed for migrating 50 GB of data in this scenario was 71.1 minutes (50/0.703).

# 5.2 Buying and Connecting to a GeminiDB Cassandra Instance

This section describes how to buy a GeminiDB Cassandra instance and uses a Linux ECS as an example to describe how to connect to the instance over a private network.

- **Buying a GeminiDB Cassandra Instance**
- **Buying an ECS**
- **Connecting to the GeminiDB Cassandra Instance**

## Buying a GeminiDB Cassandra 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 **Buy DB Instance**.

**Step 4** Click **Buy DB Instance**, select a billing mode, and configure instance parameters. Then, click **Next** and complete subsequent operations.

**Figure 5-10** Basic information

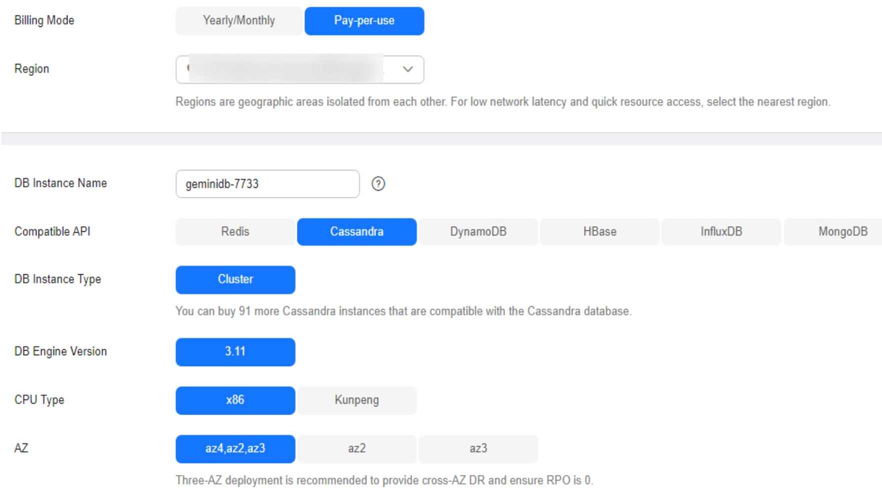
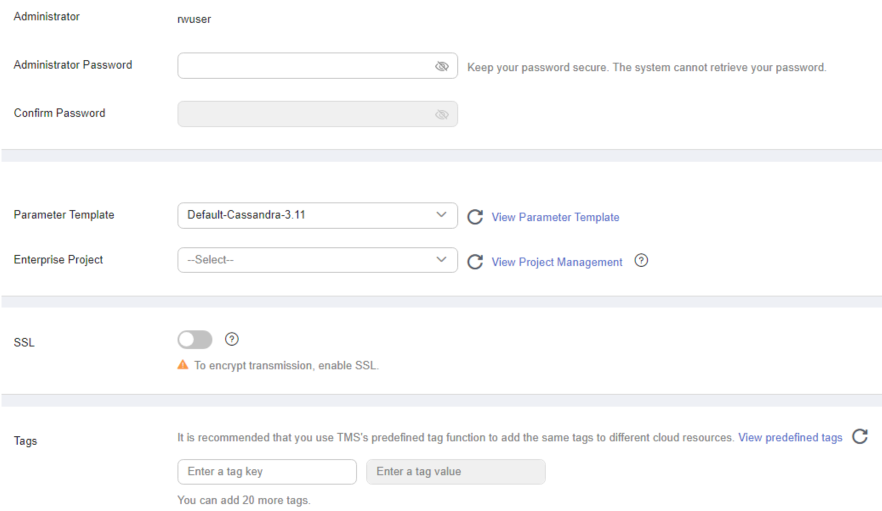
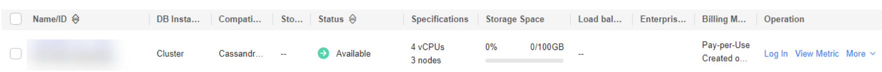



**Figure 5-11** Setting a password




**Step 5** View the purchased GeminiDB Cassandra instance.

**Figure 5-12** Available instance



**----End**

## Buying an ECS

**Step 1** [Log in to the Huawei Cloud console.]

**Step 2** In the service list, choose **Compute** > **Elastic Cloud Server**. On the Elastic Cloud Server console, click **Buy ECS**.

**Step 3** Configure basic settings and click **Next: Configure Network**. Make sure that the ECS is in the same region, AZ, VPC, and security group as the GeminiDB Cassandra instance you created.

**Figure 5-13** Basic settings

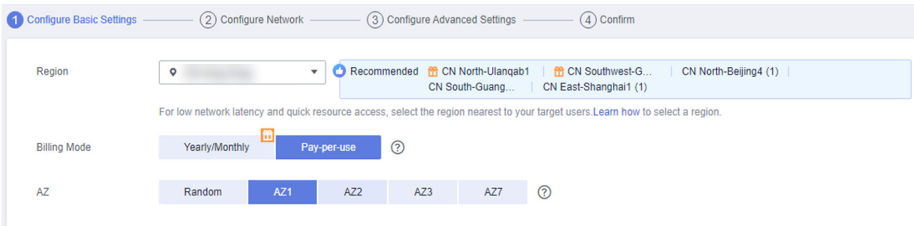



**Figure 5-14** Selecting specifications

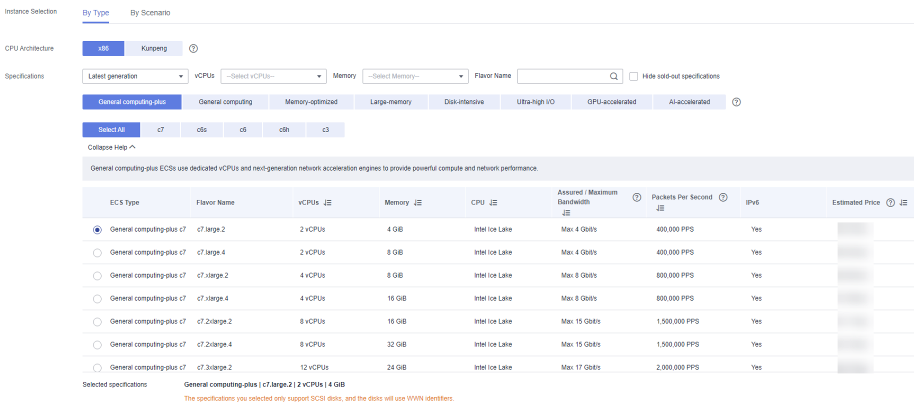
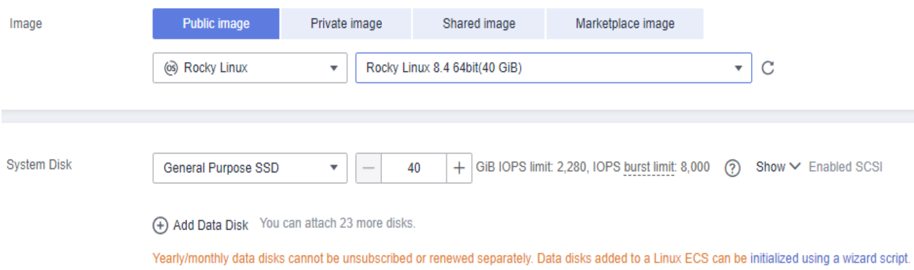


**Figure 5-15** Selecting an image




**Step 4** Configure the network and click **Next: Configure Advanced Settings**. Make sure that the ECS is in the same VPC and security group as the GeminiDB Cassandra instance.

- If security group rules allow access from the ECS, you can connect to the instance using the ECS.

- If the security group rules do not allow access from the ECS, add an inbound rule to the security group.

**Figure 5-16** Network settings

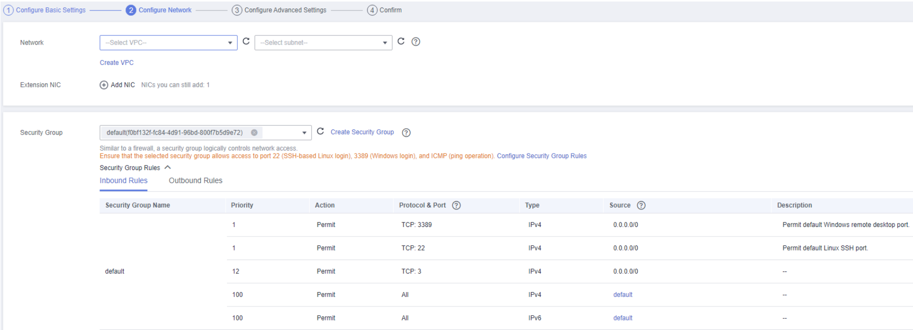
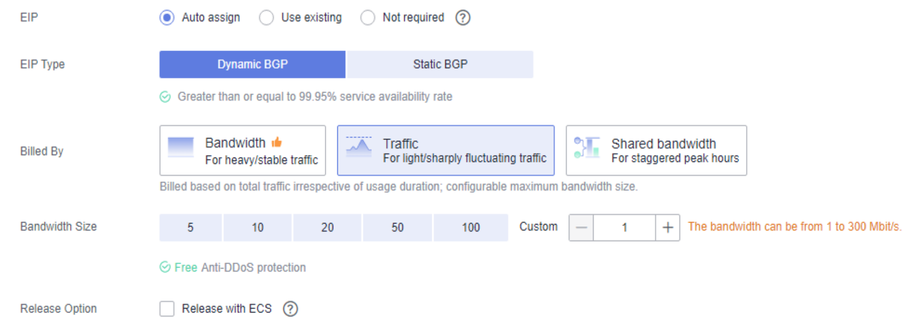
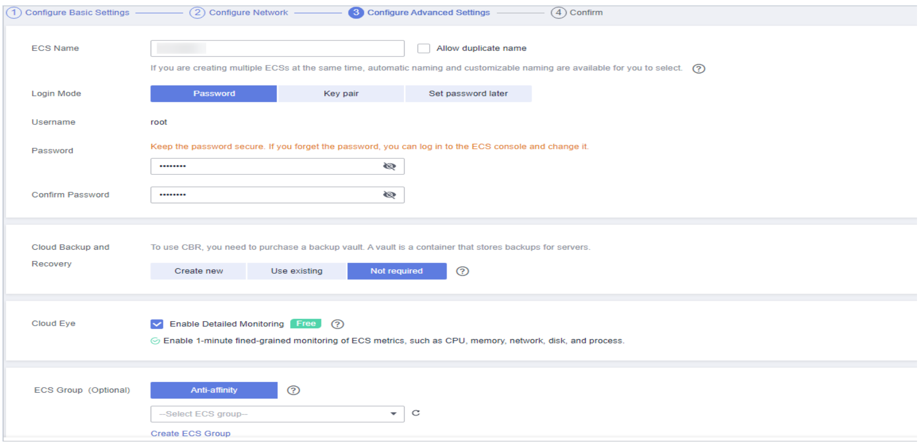



**Figure 5-17** Selecting an EIP



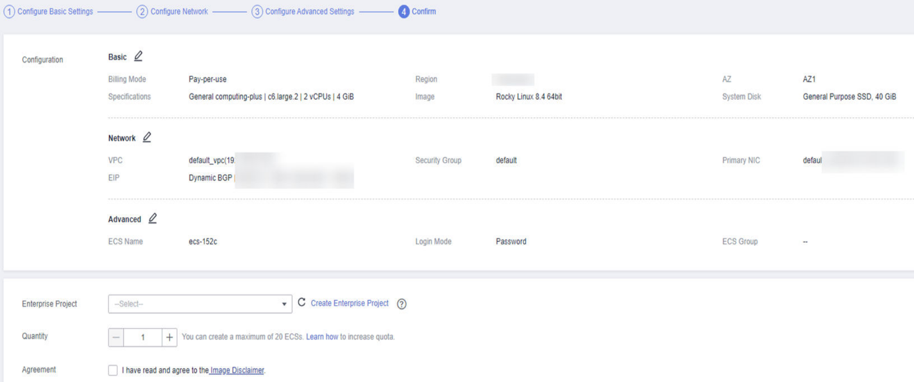

**Step 5** Configure a password for the ECS and click **Next: Confirm**.

**Figure 5-18** Advanced settings




**Step 6** Confirm the configurations and click **Submit**.

**Figure 5-19** Confirming the configurations



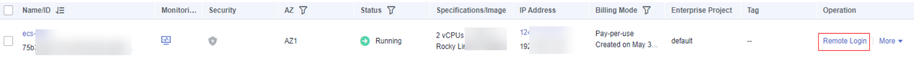

**Step 7** View the purchased ECS.

**----End**

## Connecting to the GeminiDB Cassandra Instance

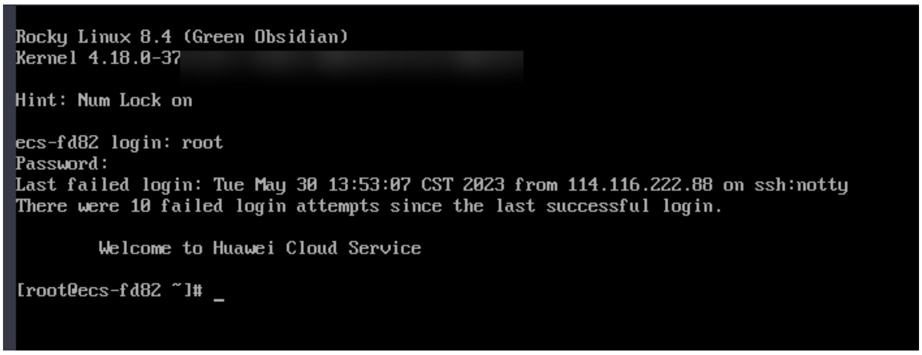
**Step 1** On the ECS console, log in to the ECS using the remote login option.

**Figure 5-20** Remote login




**Step 2** Enter the username and password of the ECS.

**Figure 5-21** Entering the username and password




**Step 3** Download the Cassandra installation package and upload it to the ECS.

Method 1:

```
wget https://dbs-download.obs.cn-north-1.myhuaweicloud.com/nosql/Cassandra_cqlsh_x86_64.zip
```

Method 2:

Download the **Cassandra client** installation package using your browser and upload it to the ECS.

**Step 4** Decompress the client package.

**unzip Cassandra_cqlsh_x86_64.zip**

**Step 5** Make the files executable:

**chmod +x ***

**Step 6** Connect to the instance in the directory where cqlsh is located.

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

Example:

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

**Table 5-7** Required description

| Parameter | Description |
| --- | --- |
| *<DB_HOST>* | The private IP address of the instance to be accessed.<br>To obtain this IP address, go to the **Instances** page, locate the instance, and click its name. The IP address can be found in the **Private IP Address** field under **Node Information** on the **Basic Information** page.<br>If the GeminiDB Cassandra instance you purchased has multiple nodes, select the private IP address of any node. |
| *<DB_PORT>* | The port used to access the instance. The default port number is 8635. Set this parameter based on service requirements.<br>Click the instance name to go to the **Basic Information** page and obtain the port number in the **Network Information** area. |
| *<DB_USER>* | Username of the instance administrator. The default value is **rwuser**. |

**Step 7** If information similar to the following is displayed, the connection was successful.

```
rwuser@cqlsh>
```

**----End**

# 5.3 Modeling Data of GeminiDB Cassandra Instances

This section describes concepts and suggestion of modeling data on GeminiDB Cassandra instances.

GeminiDB Cassandra API is a distributed, decentralized, and highly available wide-column store, a special type of NoSQL databases.

Data is evenly distributed to nodes in a GeminiDB Cassandra cluster using the consistent hashing algorithm. Each node functions as a proxy to receive requests from clients. Based on the cluster keyspace replica and snitch policies, GeminiDB Cassandra API replicates data within the specified primary key range of each node to other nodes in the cluster to improve data reliability and service availability in the distributed system.

An adjustable consistency level (such as ONE and QUORUM) is defined for each read and write, so GeminiDB Cassandra API ensures service availability and data consistency of a single request.

## Concepts

**Key**

There are multiple keys, for example:

```
CREATE TABLE mytable1 ( name text PRIMARY KEY , age int , address text , person_id text );
CREATE TABLE mytable2 ( name text , age int , address text , person_id text, PRIMARY KEY (name, age) );
CREATE TABLE mytable3 ( name text , age int , address text , person_id text, PRIMARY KEY ((name, age), person_id) ) WITH CLUSTERING ORDER BY (person_id DESC );
```

- *PRIMARY KEY*: A primary key is a unique identifier for each record in a table and consists of multiple data types. In the preceding example, **name**, **(name, age)**, and **((name, age), person_id)** indicate primary keys of **mytable1**, **mytable2**, and **mytable3**.
- *PARTITION KEY*: A partition key is the first column of a primary key and determines which node will store hashed data of GeminiDB Cassandra instances. In the preceding example, **name**, **name**, and **(name, age)** indicate partition keys of **mytable1**, **mytable2**, and **mytable3**, respectively. Data that shares partition keys is distributed to the same partition.
- *CLUSTERING KEY*: Each primary key column after a partition key is considered a clustering key, which is used to sort data within a partition. In the preceding example, **mytable1** does not have a clustering key. **age** and **person_id** indicate clustering keys of **mytable2** and **mytable3**, respectively.

To improve performance of a GeminiDB Cassandra cluster, ensure data is evenly distributed on each node in the cluster. Factors affecting the performance include the partition size, data redundancy, and disk space usage. No more than 100,000 records and no more than 100 MB of data are recommended in each partition.

**Secondary index**

Example:

```
CREATE INDEX mytable_idx_age ON mytable2 (age);
```

Create a native secondary index in the **age** column of **mytable2**. The native secondary index stores index data in a new table on a GeminiDB Cassandra instance. Values in the index column are used as keys while the original index keys are used as values. The final structure of the index table may be:

```
CREATE TABLE mytable_index_age (age int, name text , address text , person_id text, PRIMARY KEY(age, name));
```

*PARTITION KEY* cannot be used to find the node storing the index table based on **age**, because index and native data in the index table are stored on the same node and the local data placement policy is used.

You are advised to add a partition constraint of the original table when using the native secondary index. If the partition key is not specified, full table scan will be performed for a query. The following modes are recommended:

```
SELECT * FROM mytable2 WHERE age = 11 AND name = 'name';
SELECT * FROM mytable2 WHERE age >= 11 AND name IN ('name1', 'name2') ;
SELECT * FROM mytable2 WHERE age = 11 AND TOKEN (name)> xxxxx AND TOKEN(name) < yyyyy;
```

## Suggestions and Principles for Modeling Data

Before performing operations on GeminiDB Cassandra instances, you need to create a service model, organize data (design primary keys), read and write data based on application features.

- No joins: GeminiDB Cassandra API does not support a JOIN query, so you need to execute it on a client or create a table if necessary.
- No referential integrity: Cross-table referential integrity is not supported. Data in a table cannot be referenced from another table using foreign keys.
- Denormalization: Redundant data is added to an otherwise normalized relational database to improve read performance.
- Query-first: Different from a relational database management system (RDBMS), the query-first approach focuses on how to search for information first and then set up a database based on those searches or queries.
- Designing for optimal storage: How to store relational database tables is transparent to users. To model data of GeminiDB Cassandra instances, you need to consider data storage rules on disks and minimize data partitions.
- Sorting is a design decision: The sort order available on queries is fixed during table creation.

## Static Column

Generally, a table is used to store basic user information (like email addresses and passwords) and user status updates. Generally, basic information of a user seldom changes, but the user status changes frequently. If the basic information of the user is added to each status update, a large amount of storage space is wasted. To solve this problem, static columns are introduced to GeminiDB Cassandra instances. In the same partition key, a static column has only one value. That is, only one replica is stored.

You can easily define a static column by adding STATIC to the end of the column, for example:

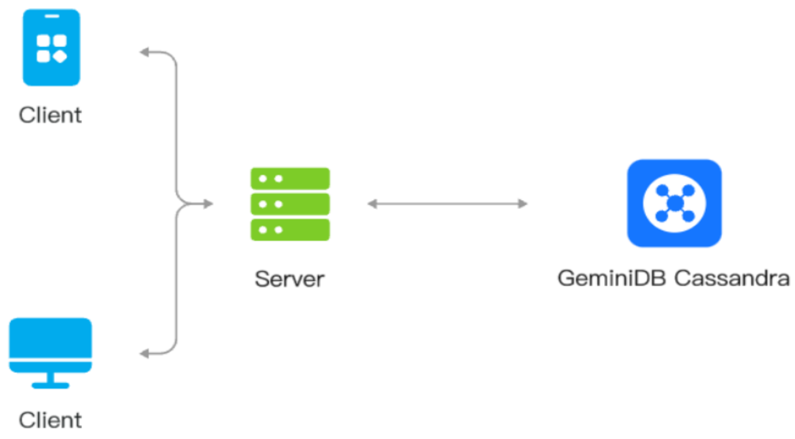
```
CREATE TABLE test (username text, id timeuuid, email text STATIC, PRIMARY KEY (username, id));
```

**email** is defined as static, so only one email address is available for a specified username.

# 5.4 Scenarios

GeminiDB Cassandra API supports high-concurrency and low-latency access. It features high availability and autoscaling and is suitable for online applications on the Internet, such as messages, orders, and websites, with a large amount of data to be dealt with.

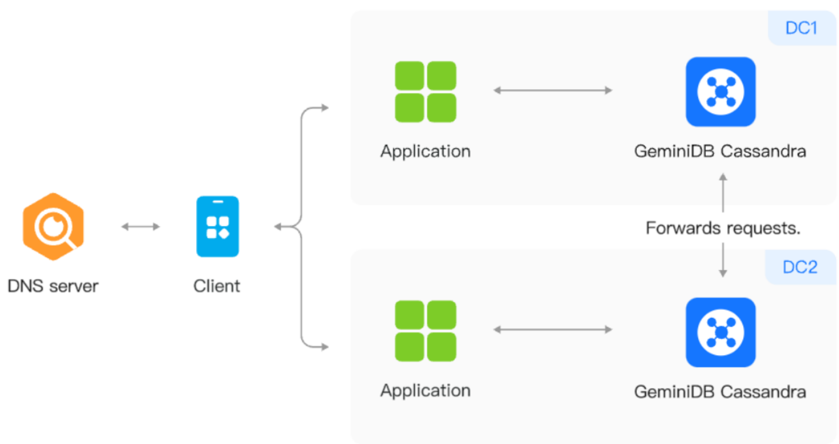
**Figure 5-22** Online applications




## Advantages

- High availability: A single point of failure does not affect services.
- Low latency: It takes milliseconds for a data packet to go from its source to its destination.
- Autoscaling: Compute and storage resources can be flexibly scaled out.
- Multi-active deployment: GeminiDB Cassandra instances can be deployed in multiple DCs to provide high availability and DR capabilities.

**Figure 5-23** Multi-DC deployment




# 5.5 Designing Primary Keys for a Wide Table

GeminiDB Cassandra API is a distributed database engine in which data is distributed based on primary keys. If the primary key of a table contains multiple columns, GeminiDB Cassandra API uses the columns in a left-to-right order to query data. A primary key that is improperly designed cannot be effectively used in queries. In this case, a large number of queries may be performed on a small amount of hot spot data, which degrades the query performance. Therefore, the design of primary keys plays an important role in data queries. This topic describes the considerations for primary key design and provides examples.

## Are Primary Key Values Unique?

Different versions of a row use the same primary key value. By default, the latest version is returned when a query is performed. In most cases, primary keys must be unique.

Best practice: A primary key is a column or a set of columns. Each primary key value corresponds to a record.

- [userid]: Only one column is specified as the primary key. Only one record is generated for each user.
- [userid][orderid]: Two columns are specified as the primary key. Multiple records are generated for each user.

## How Do I Design the Primary Key in Different Scenarios?

The primary key design restricts data query methods. GeminiDB Cassandra API supports SELECT statements that use the following methods:

- The primary key is used to query data, for example:
```
SELECT * FROM table WHERE userid='abc' AND orderid=123;
```

**NOTE**

To use this method, you need to specify all primary key columns. The values in all primary key columns must be explicit.

- Data is queried based on the primary key range, for example:
```
SELECT * FROM table WHERE userid='abc' AND 123<orderid<456;
```

**NOTE**

To use this method, you need to specify the range that you want to scan in the first primary key column. If you do not specify the range, queries may time out or fail.

Best practice: How do I submit complex queries using the preceding query methods?

- Create an index table.
- Specify columns that you want to scan other than the primary key columns in the query conditions. Irrelevant data is automatically filtered out.
- Use secondary indexes.
- Execute the ORDER BY statement to sort data in descending order. This way, new records are sorted to top rows of the table. For example:
```
SELECT * FROM table WHERE userid='abc' AND 123<orderid<456 ORDER BY orderid DESC;
```

**NOTE**

When most queries are submitted to retrieve the up-to-date data, you can design the primary key as [userid][orderid DESC] to sort the data in descending order.

## Factors To Be Considered During Primary Key Design

The following factors need to be considered:

- Length of values in primary key columns: Values in primary key columns should be short in length. Columns that store fixed-length values, such as long integers, are recommended as the primary key columns. If the length is not fixed, you are advised to limit it within 2 KB to reduce storage costs and improve write performance.
- Number of primary key columns: Fewer primary key columns can improve write performance and reduce storage costs. One to three primary key columns are recommended.

## What Should I Avoid When Designing Primary Keys?

GeminiDB Cassandra instance data is distributed based on primary keys. If the primary key of a table contains multiple columns, data is distributed based the columns in a left-to-right order. To avoid a large number of write operations from being performed on a small amount of hot spot data, note the following items:

- Values in the first primary key column must be dispersed.
- Do not specify a column that contains auto-incremental data or a column in which values have the same prefix, such as the timestamp column, as the first primary key column or the index column.
- Do not specify a column that contains enumerated data, such as order types, or a column in which values have obvious prefixes as the first primary key column.

If you have to specify a column of the preceding type as the first primary key column, use the hash method to distribute data in the column.

For example, if you have to specify the column pk that contains auto-incremental strings as the first primary key column, you can create a column named pk1 based on the pk column using the following algorithm: pk1 = hash(pk).substring(0,4)+pk. The pk1 column is concatenated by the pk column and a prefix that is the first four digits of the result returned by the hash method based on the pk column.

## Will Stacked Hot Spots Occur for Fully Distributed Data?

The hash method is used to distribute data to different partitions. This prevents a server from being terminated by hot spots and the other servers from being idle. This way, the distributed architecture and concurrent processing are utilized in an efficient manner.

Best practice:

- Design an MD5 hash algorithm. The primary key is [md5(userid).subStr(0,4)][userId][orderid].
- Design a reverse index. The primary key is [reverse(userid)][orderid].
- Design the modulo operation. The primary key is [bucket][timestamp][hostname][log-event]; long bucket = timestamp % numBuckets.
- Add random numbers. The primary key is [userId][orderid][random(100)].

## Can a Primary Key Be Simplified?

You can reduce the number of primary key columns to decrease the amount of data that is scanned and improve the efficiency of queries and insert operations.

Best practice:

- Replace the STRING data type with the LONG or INT data type, for example, '2015122410' => Long(2015122410).
- Replace names with codes, for example,'mobile phone'=>'sj'.

## Common Design Examples

Primary key designs for log data and time series data

- To query the data of a metric that is generated over a period of time, design the primary key as [hostname][log-event][timestamp].
- To query the most recent records of a metric, design the primary key as [hostname][log-event][timestamp DESC].
- To query data that contains only the time dimension or query data whose volume is large in a specific dimension, design the primary key as long bucket = timestamp % numBuckets; [bucket][timestamp][hostname][log-event].

Primary key designs for transaction data

- To query the transaction records of a seller within a specific period of time, design the primary key as [seller_id][timestamp][order_number].
- To query the transaction records of a buyer within a specific period of time, design the primary key as [buyer_id][timestamp][order_number].
- To query data based on order IDs, design the primary key as [order_number].
- To join three tables to perform queries, design the primary key of the table that stores buyer data as [buyer_id][timestamp][order_number], primary key of the table that stores seller data as [seller_id][timestamp][order_number], and primary key of the table that stores order IDs as [order_number].

# 5.6 Pre-partitioning Tables

Tables compatible with GeminiDB HBase can be pre-partitioned to properly design a row key and prevent data hotspots.

## Method

Run the following statement to partition the test table compatible with GeminiDB HBase.

```
create 'test', {NAME => 'cf1'}, SPLITS => ['1111', '2222', '3333']
```

**Table 5-8** Fields in the test table

| Field | Description |
| --- | --- |
| NAME | Column family name of a table |
| SPLITS | Pre-partition boundary. Data is distributed in preset partitions based on the byte order of row keys. |

# 5.7 Suggestions on Alarm Rules of GeminiDB Cassandra Instance Metrics

After setting alarm rules on the Cloud Eye console, for example, specifying monitored objects and notification policies, you can stay ahead of your instance status. For details, see **Configuring Alarm Rules**.

This section describes recommended alarm rules of GeminiDB Cassandra instances.

**Table 5-9** Suggestions on alarm rules of GeminiDB Cassandra instance metrics

| Metric ID | Metric Name | Dimension | Threshold (Raw Value) in Best Practices | Alarm Severity in Best Practices | Alarm Handling Suggestion |
| --- | --- | --- | --- | --- | --- |
| nosql005_disk_usage | Storage Space Usage | Instance | > 80% for 3 consecutive periods | Major | • Evaluate how much storage needs to be added based on data growth. For details, see **Manually Scaling Up Storage Space**.<br>• Enable autoscaling. For details, see **Automatically Scaling Up Storage Space**. |
| cassandra001_cpu_usage | CPU Usage | Node | > 80% for 3 consecutive periods | Major | Upgrade CPU specifications. For details, see **Changing vCPUs and Memory**. |
| cassandra002_mem_usage | Memory Usage | Node | > 80% for 3 consecutive periods | Major | Upgrade memory specifications. For details, see **Changing vCPUs and Memory**. |
| cassandra015_read_latency | Average Read Latency | Node | > 900 ms for 3 consecutive periods | Major | Check whether the service traffic increases sharply and whether the database is normal. For details, see **Viewing Metrics**. |

| Metric ID | Metric Name | Dimension | Threshold (Raw Value) in Best Practices | Alarm Severity in Best Practices | Alarm Handling Suggestion |
|---|---|---|---|---|---|
| cassandra016_write_latency | Average Write Latency | Node | > 900 ms for 3 consecutive periods | Major | Check whether the service traffic increases sharply and whether the database is normal. For details, see **Viewing Metrics**. |
| cassandra037_pending_write | Suspended Write Tasks | Node | > 3,000 for 3 consecutive periods | Major | Check whether the service traffic increases sharply and whether the database is normal. For details, see **Viewing Metrics**. |
| cassandra038_pending_read | Suspended Read Tasks | Node | > 3,000 for 3 consecutive periods | Major | Check whether the service traffic increases sharply and whether the database is normal. For details, see **Viewing Metrics**. |

# 5.8 How Do I Sort a Large Result Set?

This section describes how to improve query performance of GeminiDB Cassandra instances when a large result set is returned.

## Application Scenarios

Common scenarios for sorting result sets of GeminiDB Cassandra instances:

- In a small result set, ORDER BY can be used for efficient in-memory calculations without any constraints.
- In a large result set, using ORDER BY may consume extra compute resources and make the query statement execution time out.

## Solution

By default, query results of GeminiDB Cassandra instances are sorted based on the sorting rule specified during table creation. That is, when creating a table, you can determine which columns need to be sorted and how they will be sorted. In this way, the query performance can be improved. For example:

```
CREATE TABLE test(
   pk1 text,
```

```
    pk2 text,
    ck1 text,
    PRIMARY KEY (pk1, pk2)
) WITH CLUSTERING ORDER BY (pk1 DESC, pk2 ASC);
```

In the **test** table, the **pk1** column is sorted in descending order and **pk2** in ascending order. WITH CLUSTERING ORDER BY specifies the sorting order, which can be **ASC** (ascending) or **DESC** (descending), of clustering keys. If no order is specified, data is sorted in ascending order (**ASC**) by default.

## ORDER BY Example

A small result set can be sorted using ORDER BY, for example:

- Sort the **ck1** column in ascending order.

```
SELECT * FROM test WHERE pk1=? ORDER BY ck1 asc;
```

- Sorts the **ck1** column in descending order.

```
SELECT * FROM test WHERE pk1=? ORDER BY ck1 desc;
```

# 5.9 Basic Syntax Examples of GeminiDB Cassandra Instances

This section describes the basic syntax of GeminiDB Cassandra instances.

- Keyspace syntax
  - Create a keyspace.

    Example:

    **CREATE KEYSPACE IF NOT EXISTS nosql WITH replication = {'class': 'SimpleStrategy', 'replication_factor': '3'};**

    In this example, the keyspace name is set to **nosql** and **class** to **SimpleStrategy**. **replication_factor** indicates the number of copies. A GeminiDB Cassandra instance provides strong consistency, three-copy storage by default.
  - Run **DESC** *<keyspace_name>* to verify the creation results.

    **Figure 5-24** Verifying the creation results

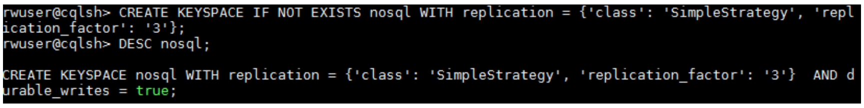
    
    
  - Run **use** *<keyspace_name>* to switch to the created keyspace.

    **Figure 5-25** Switching the keyspace

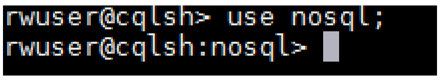
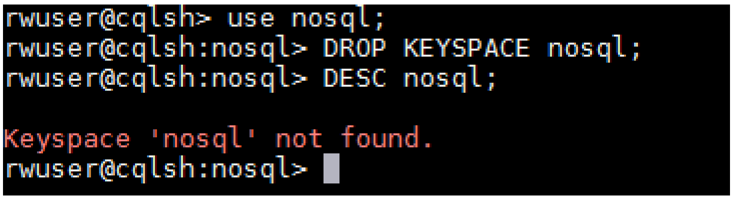
    
    
  - Run **DROP KEYSPACE** *<keyspace_name>* to delete the created keyspace.

**Figure 5-26** Deleting the keyspace



```
rwuser@cqlsh> use nosql;
rwuser@cqlsh:nosql> DROP KEYSPACE nosql;
rwuser@cqlsh:nosql> DESC nosql;

Keyspace 'nosql' not found.
rwuser@cqlsh:nosql>
```

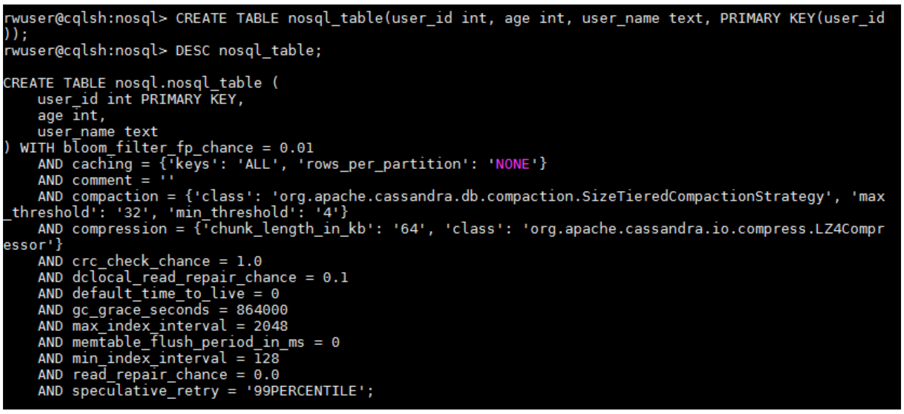
- Table syntax
  - Create a table.

    Example:

    **CREATE TABLE nosql_table(user_id int, age int, user_name text, PRIMARY KEY(user_id));**

    **nosql_table** is a table name defined by the following three columns: **user_id**, **age**, and **user_name**. **user_id** indicates a user ID of the INT data type. **age** indicates user age of the INT data type. **user_name** indicates a username of the TEXT data type. The primary key is **user_id**.
  - Run **DESC** *<table_name>* to verify the creation results.

**Figure 5-27** Verifying the creation results



```
rwuser@cqlsh:nosql> CREATE TABLE nosql_table(user_id int, age int, user_name text, PRIMARY KEY(user_id
));
rwuser@cqlsh:nosql> DESC nosql_table;

CREATE TABLE nosql.nosql_table (
    user_id int PRIMARY KEY,
    age int,
    user_name text
) WITH bloom_filter_fp_chance = 0.01
    AND caching = {'keys': 'ALL', 'rows_per_partition': 'NONE'}
    AND comment = ''
    AND compaction = {'class': 'org.apache.cassandra.db.compaction.SizeTieredCompactionStrategy', 'max
_threshold': '32', 'min_threshold': '4'}
    AND compression = {'chunk_length_in_kb': '64', 'class': 'org.apache.cassandra.io.compress.LZ4Compr
essor'}
    AND crc_check_chance = 1.0
    AND dclocal_read_repair_chance = 0.1
    AND default_time_to_live = 0
    AND gc_grace_seconds = 864000
    AND max_index_interval = 2048
    AND memtable_flush_period_in_ms = 0
    AND min_index_interval = 128
    AND read_repair_chance = 0.0
    AND speculative_retry = '99PERCENTILE';
```

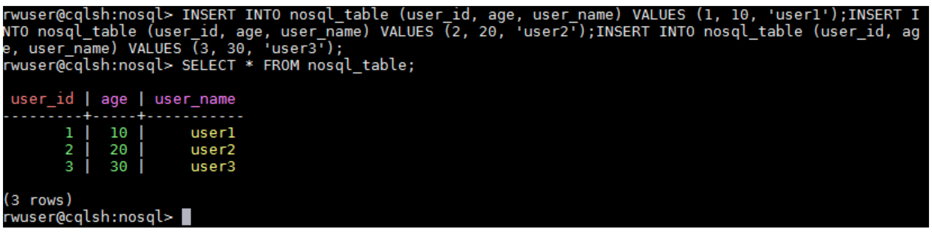
  - Insert data into the table, for example,

    **INSERT INTO nosql_table (user_id, age, user_name) VALUES (1, 10, 'user1');**

    **INSERT INTO nosql_table (user_id, age, user_name) VALUES (2, 20, 'user2');**

    **INSERT INTO nosql_table (user_id, age, user_name) VALUES (3, 30, 'user3');**
  - Run **SELECT * FROM** *<table_name>* to query table data.

**Figure 5-28** Querying table data



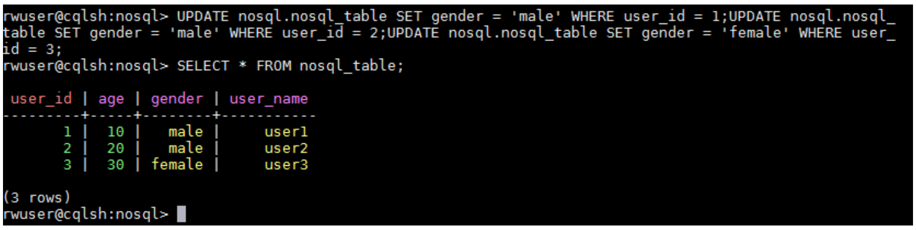
```
rwuser@cqlsh:nosql> INSERT INTO nosql_table (user_id, age, user_name) VALUES (1, 10, 'user1');INSERT I
NTO nosql_table (user_id, age, user_name) VALUES (2, 20, 'user2');INSERT INTO nosql_table (user_id, ag
e, user_name) VALUES (3, 30, 'user3');
rwuser@cqlsh:nosql> SELECT * FROM nosql_table;

 user_id | age | user_name
---------+-----+-----------
       1 |  10 |     user1
       2 |  20 |     user2
       3 |  30 |     user3

(3 rows)
rwuser@cqlsh:nosql>
```

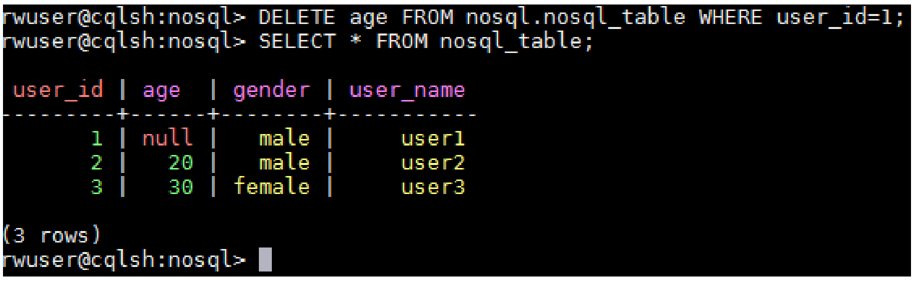
– Add a column to the table, for example:
  **ALTER TABLE nosql_table ADD gender text;**
– Update data in a table of a keyspace, for example:
  **UPDATE nosql.nosql_table SET gender = 'male' WHERE user_id = 1;**
  **UPDATE nosql.nosql_table SET gender = 'male' WHERE user_id = 2;**
  **UPDATE nosql.nosql_table SET gender = 'female' WHERE user_id = 3;**

**Figure 5-29** Viewing the update results



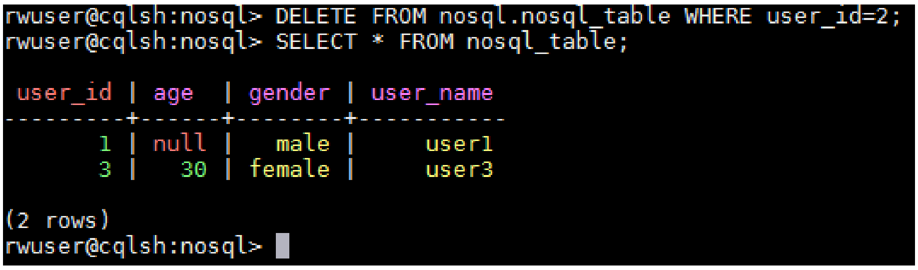

– Delete data from a table in a keyspace, for example:
  Delete age data of the user whose ID is **1**.
  **DELETE age FROM nosql.nosql_table WHERE user_id=1;**

**Figure 5-30** Deleting a data record




  Delete the entire record of the user whose ID is **2**.
  **DELETE FROM nosql.nosql_table WHERE user_id=2;**

**Figure 5-31** Deleting the entire record




– Delete an entire table, for example:
  **DROP TABLE nosql.nosql_table;**

**Figure 5-32** Deleting an entire table

```
rwuser@cqlsh:nosql> DROP TABLE nosql.nosql_table;
rwuser@cqlsh:nosql> DESC nosql_table;

'nosql_table' not found in keyspace 'nosql'
rwuser@cqlsh:nosql>
```

- **HELP** command
  - Run **HELP** to view all supported commands.

    **Figure 5-33** Viewing all supported commands

```
rwuser@cqlsh> HELP

Documented shell commands:
===========================
CAPTURE  CLS          COPY  DESCRIBE  EXPAND  LOGIN   SERIAL  SOURCE   UNICODE
CLEAR    CONSISTENCY  DESC  EXIT      HELP    PAGING  SHOW    TRACING

CQL help topics:
================
AGGREGATES               CREATE_KEYSPACE           DROP_TRIGGER      TEXT
ALTER_KEYSPACE           CREATE_MATERIALIZED_VIEW  DROP_TYPE         TIME
ALTER_MATERIALIZED_VIEW  CREATE_ROLE               DROP_USER         TIMESTAMP
ALTER_TABLE              CREATE_TABLE              FUNCTIONS         TRUNCATE
ALTER_TYPE               CREATE_TRIGGER            GRANT             TYPES
ALTER_USER               CREATE_TYPE               INSERT            UPDATE
APPLY                    CREATE_USER               INSERT_JSON       USE
ASCII                    DATE                      INT               UUID
BATCH                    DELETE                    JSON
BEGIN                    DROP_AGGREGATE            KEYWORDS
BLOB                     DROP_COLUMNFAMILY         LIST_PERMISSIONS
BOOLEAN                  DROP_FUNCTION             LIST_ROLES
COUNTER                  DROP_INDEX                LIST_USERS
CREATE_AGGREGATE         DROP_KEYSPACE             PERMISSIONS
CREATE_COLUMNFAMILY      DROP_MATERIALIZED_VIEW    REVOKE
CREATE_FUNCTION          DROP_ROLE                 SELECT
CREATE_INDEX             DROP_TABLE                SELECT_JSON
```

  - Run **HELP** *<COMMAND>* to query the usage of a command.

    Example: **HELP DESC**

# 6 Performance White Paper

## 6.1 Performance Test Methods

This section describes performance testing of GeminiDB Cassandra instances, including the test environment, procedure, and results.

### Test Environment

- Region: AP-Singapore
- AZ: AZ1, AZ2, and AZ3 (three-AZ deployment)
- Elastic Cloud Server (ECS): h3.4xlarge.2 with 16 vCPUs, 32 GB of memory, and CentOS 7.5 64-bit image
- Nodes per instance: 3
- Instance specifications: All specifications described in **Table 6-1**

**Table 6-1** Instance specifications

| No. | Specifications |
| --- | --- |
| Cluster 1 | 4 vCPUs \| 16 GB |
| Cluster 2 | 8 vCPUs \| 32 GB |
| Cluster 3 | 16 vCPUs \| 64 GB |
| Cluster 4 | 32 vCPUs \| 128 GB |

### Test Tool

YCSB is an open-source tool for testing performance of databases. In this test, YCSB 0.15.0 is used. RoundRobinPolicy is used for load balancing.

For details on how to use this tool, see **YCSB**.

## Test Metrics

Operations per Second (OPS): operations executed by a database per second

## Test Procedure

1. Configure the **workload** file.

   Set values for fields **readproportion**, **insertproportion**, **updateproportion**, **scanproportion**, and **readmodifywriteproportion** in the file by referring to **Table 6-2**.

   Set a value for field **recordcount** in the file by referring to **Table 6-3**.

2. Use workload-insert-mostly as an example. Run the following command to prepare test data:

   **sh bin/ycsb.sh load cassandra-cql -P workloads/workload-insert-mostly -p "hosts=${ContactPoints}" -p "port=${port}" -p "cassandra.username=${username}" -p "cassandra.password=${password}" -p operationcount=400000000 -p recordcount=400000000 -p exportfile=./data_load.exp -threads ${threadNum} -s > data_load.log 2>&1 &**

3. Use workload-insert-mostly as an example. Run the following command to test performance:

   **sh bin/ycsb.sh run cassandra-cql -P workloads/workload-insert-mostly -p "hosts=${ContactPoints}" -p "port=${port}" -p "cassandra.username=${username}" -p "cassandra.password=${password}" -p operationcount=9000000 -p recordcount=9000000 -p maxexecutiontime=3600 -p exportfile=./workload-insert-mostly.exp -threads ${threadNum} -s > workload-insert-mostly_run.log 2>&1 &**

## Test Models

- Workload model

**Table 6-2** Workload models

| Workload Model | Description |
| --- | --- |
| More read requests than write requests<br>workload-read-mostly | 95% read, 5% update |
| Balanced read and write requests<br>workload-read-write-combination | 50% update, 50% read |
| Balanced read and rewrite requests<br>workload-read-modify-write | 50% read, 50% read-modify-write |

| Workload Model | Description |
| --- | --- |
| Read, update, and write requests<br>workload-mixed-operational-analytical | 65% read, 25% update, 10% insert |
| More write requests than read requests<br>workload-insert-mostly | 90% insert, 10% read |

- Data model

  fieldlength = 100, fieldcount = 10

- Preset data volume

  Different preset data volumes were used to test performance of instances of each type of specifications.

  The following table describes the preset data volumes.

**Table 6-3** Preset data volumes

| No. | Specifications | Preset Data Volume |
| --- | --- | --- |
| Cluster 1 | 4 vCPUs \| 16 GB | 50 GB |
| Cluster 2 | 8 vCPUs \| 32 GB | 100 GB |
| Cluster 3 | 16 vCPUs \| 64 GB | 200 GB |
| Cluster 4 | 32 vCPUs \| 128 GB | 400 GB |

## 6.2 Performance Test Data

The OPS of instances of different specifications can be tested using different service models with the same preset data volume. For details, see the numbers in bold in **Table 6-4**.

**Table 6-4** Test data

| Node Specifications | 4 vCPUs \| 16 GB | 8 vCPUs \| 32 GB | 16 vCPUs \| 64 GB | 32 vCPUs \| 128 GB |
| --- | --- | --- | --- | --- |
| Concurrent Client Requests | 32 | 64 | 128 | 256 |
| Preset Data Volume | 50 GB | 100 GB | 200 GB | 400 GB |
| More read requests than write requests<br>workload-read-mostly | **15627** | **44612** | **90713** | **169037** |

| | | | | |
|---|---|---|---|---|
| Balanced read and write requests<br>workload-read-write-combination | **19565** | **46240** | **95065** | **158793** |
| Balanced read and rewrite requests<br>workload-read-modify-write | **11768** | **29488** | **59332** | **96964** |
| Read, update, and write requests<br>workload-mixed-operational-analytical | **17534** | **40214** | **80661** | **137616** |
| More write requests than read requests<br>workload-insert-mostly | **23830** | **44548** | **67290** | **111540** |

**NOTE**

- Operations per Second (OPS): operations executed by a database per second
- Test Model No.: indicates the test model sequence number. For details, see **Table 6-2**.

# 7 FAQs

## 7.1 Product Consulting

### 7.1.1 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.

### 7.1.2 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.

## 7.2 Billing

## 7.2.1 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.

## 7.2.2 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 a Yearly/Monthly Instance to Pay-per-Use**.
- If you want to change the billing mode from pay-per-use to yearly/monthly, see **Changing a Pay-per-Use Instance to Yearly/Monthly**.

# 7.3 Database Usage

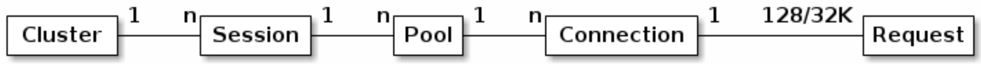
## 7.3.1 Why Does the Overall Instance Performance Deteriorate When QPS Increases After the Batch Size Is Decreased?

### 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.

# 7.3.2 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**

# 7.3.3 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 7-1**.

**Table 7-1** 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 7-2**.

**Table 7-2** 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 7-3**.

**Table 7-3** 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 7-4**.

**Table 7-4** 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';
  ```

## 7.3.4 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 provided by Huawei 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.

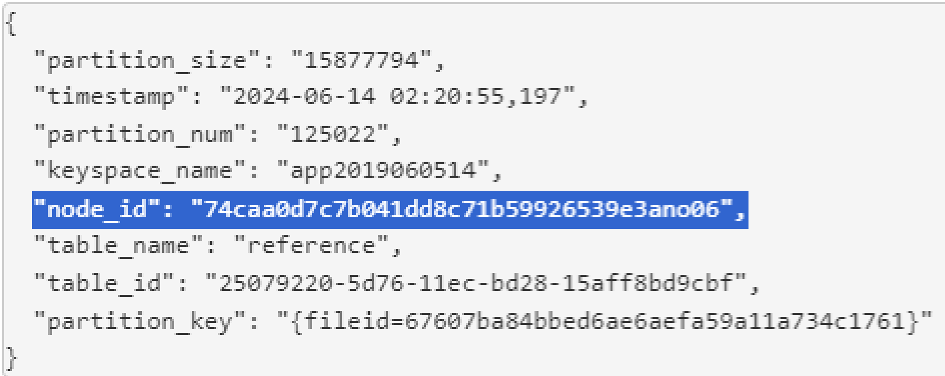
  a. The number of rows of a single partition key cannot exceed 100,000.
  b. The size of a single partition cannot exceed 100 MB.

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 7-1** Viewing events of big key alarms




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.

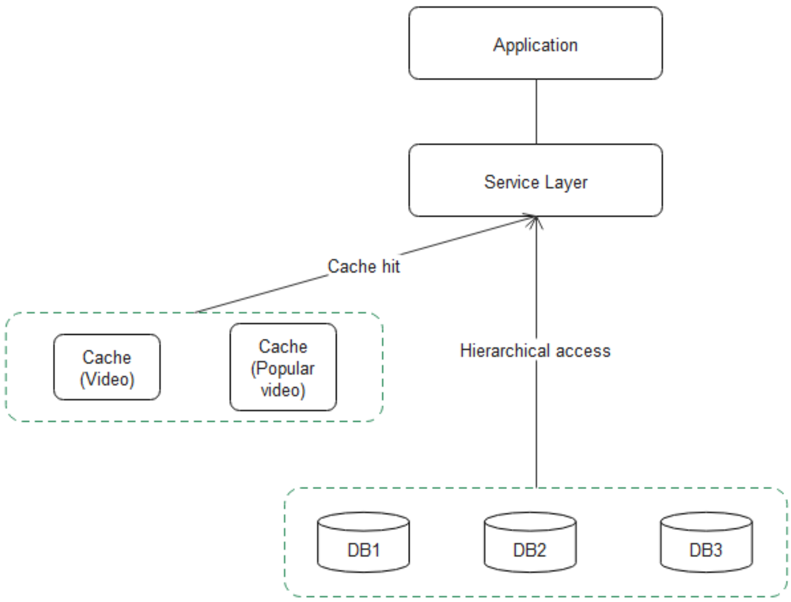
   Create a table to store the short video information. Only public short video information is retained, and user information is not included. This ensures that the table does not generate large partition keys. Write the short video information to the table.

```
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

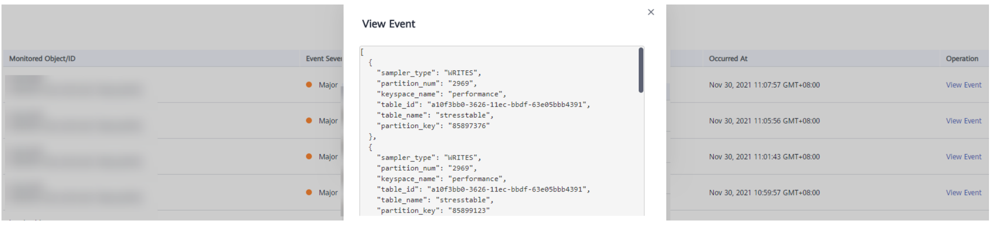
  A key whose access frequency is greater than 100,000 times per minute is defined as a HotKey.

  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 HotKey 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 7-2** 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.

# 7.3.5 How Do I Set Up a Materialized View?

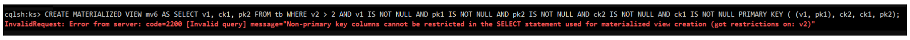
## Concept

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 7-3** 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:

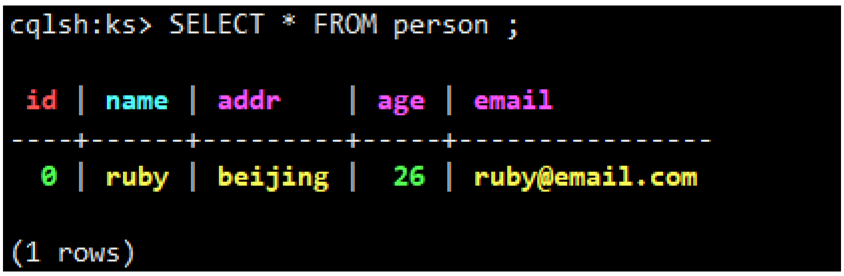
   ```
   CREATE TABLE person (
     id int,
     name text,
     addr text,
     age int,
     email text,
     PRIMARY KEY (id, name));
   ```

   Insert a record.

   **insert into person(id, name, age, addr, email) values (0, 'ruby', 26, 'beijing', 'ruby@email.com');**

   Query the result.

   **Figure 7-4** 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 7-5** 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');**

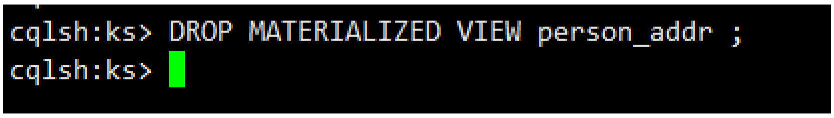
   Query the data in the base table and materialized view.

   **Figure 7-6** Querying the result


4. Delete the materialized view.

   **DROP MATERIALIZED VIEW person_adder;**

**Figure 7-7** Deleting a view




## 7.3.6 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:

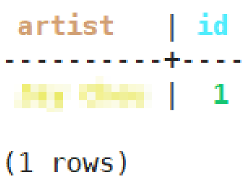
```
CREATE TABLE playlists (
id int,
song_id int,
song_order int,
album text,
artist text,
title text,
PRIMARY KEY (id, song_id));
```

The query result is as follows.

**Figure 7-8** Querying the result


If an index is created for the **artist** field, the hidden table structure is as follows.

**Figure 7-9** Querying the result




**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 7-10** 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 7-11** 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.

# 7.3.7 How Can I Use the Search Index of Lucene?

GeminiDB Cassandra API supports Lucene search indexes, which are used for multi-dimensional queries, text retrieval, document counting and analysis. Search indexes are used the similar way as native secondary indexes, but search indexes support more syntax types.

## Pain Points of Open-Source Cassandra Secondary Indexes

Cassandra secondary indexes are stored in an implicit table. This table's primary key is an index column with values from the original table's primary keys, which is easy to implement. The following constraints are inevitably introduced:

- Only "=" can be used to query the first primary key.
- "=", ">", "<", ">=", and "<=" can be used to query the second primary key.
- Only "=" can be used to query the index column.
- Indexes cannot be created for columns that are frequently deleted or updated.
- Creating an index on a high-cardinality column is not recommended.

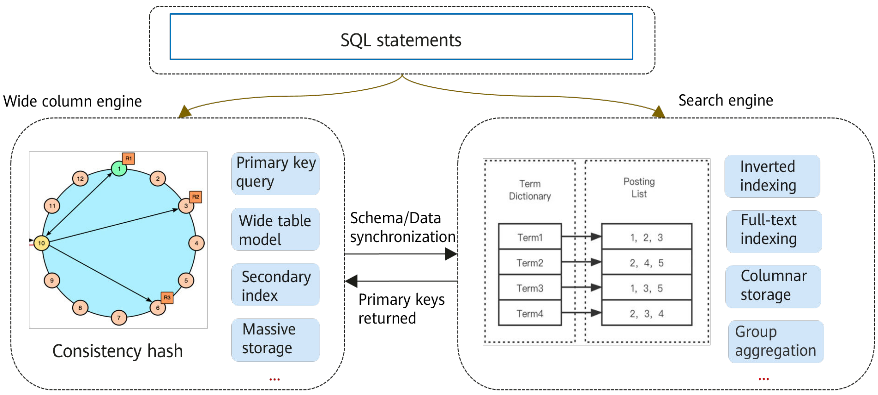
So you can see the query function provided by Cassandra secondary indexes is somewhat limited.

## Lucene Search Index Architecture

Key technologies:

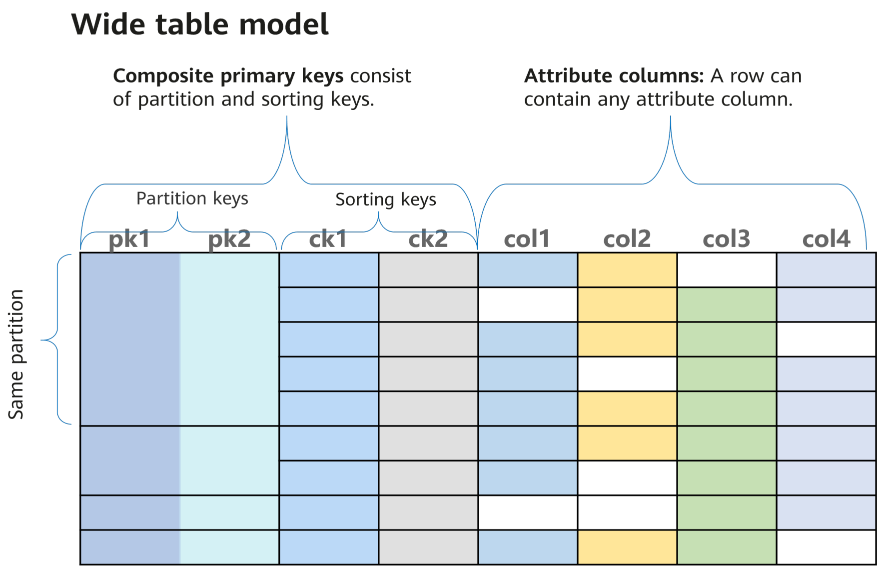
- Search indexes are provided based on integration of the search engine of Lucene and the storage engine of a wide table.
- Lucene is compatible with Apache Cassandra syntax and supports more SQL statements. Lucene provides capabilities such as multi-dimensional query, text retrieval, fuzzy query, and COUNT query, comprehensively improving user experience in querying massive volumes of data.

**Figure 7-12** Lucene search index architecture



## Usage Example of Lucene Search Indexes

**Figure 7-13** Lucene search index usage




Example of creating a table structure:

```
CREATE TABLE example (pk1 text, pk2 bigint, ck1 int,ck2 text,col1 int, col2 int, col3 text, col4 text, PRIMARY KEY ((pk1,pk2),ck1, ck2));
```

Example of creating Lucene search indexes:

```
# Creates Lucene indexes in col1, col2, col3, and col4.
CREATE CUSTOM INDEX index_lucene ON test.example(col1,col2,col3,col4) USING 'LuceneGlobalIndex'
WITH OPTIONS = {
'table_tokens': '3', # Sets the number of Lucene search index shards to be initialized to 3.
'analyzed_columns': 'col4', # Specifies col4 for full-text search.
'disable_doc_value': 'col4', # Disables DocValues for col4.
'ordered_columns': 'col3,col4', # Specifies col3 and col4 as sorting columns.
'ordered_sequences': 'desc,asc', # Sorts col3 in the descending order and col4 in the ascending order.
'analyzer_class': 'StandardAnalyzer', # Specifies StandardAnalyzer as the analyzer for full-text search.
'case_insensitive': 'col3' # Sets col3 to be case-insensitive for characters.
};
```

**Table 7-5** Optional parameters of OPTIONS

| Parameter | Description |
| --- | --- |
| table_tokens | Number of initialized Lucene search index shards. If this parameter is not specified, default value **3** is used. Shards occupy CPU and memory resources and increase with data volumes. |
| analyzed_columns | Column used for full-text search. By default, characters are converted to lowercase letters for storage. To enable case-insensitive searches, you need to specify **case_insensitive**. |

| Parameter | Description |
| --- | --- |
| analyzer_class | Analyzer used for full-text search<br>Chinese parser:<br>'analyzer_class': 'SmartChineseAnalyzer'<br>Standard parser:<br>'analyzer_class': 'StandardAnalyzer'<br>IK parser:<br>'analyzer_class': 'IKAnalyzer' |
| ordered_columns | Default sorting columns of Lucene search index columns. **ordered_sequences** is configured for columns specified for this parameter. If this parameter is not specified, the sorting mode is the same as that of sorting keys of GeminiDB Cassandra instances by default. Multiple index columns are separated by commas (,).<br>**Note**: The query efficiency is the highest only in the default sorting mode. |
| ordered_sequences | Sorting order of Lucene search index columns specified in **ordered_columns**. **asc** indicates the ascending order and **desc** descending order. |
| disable_doc_value | **docValues** is set to **false** for index columns that do not require operations such as sorting and aggregate query. |
| case_insensitive | Character case is ignored in specified index columns. The characters are automatically converted to lowercase letters during storage and query. |

Multi-dimensional query: Implement a nested query based on any combination of index columns. Both exact query and range query are supported.

```
SELECT * from example WHERE pk1>='a' and pk2>=1000 and ck2 in ('a','b','c') and col1 <= 4 and col2 >= 2;
```

COUNT query: The number of rows in a data table or the number of matched rows is obtained based on a specific query criteria of index columns.

```
SELECT count(*) FROM example WHERE col1 > 3 AND EXPR(index_lucene, 'count');
```

Index column sorting order: Multiple index column sorting rules can be specified. A result set is returned based on the multi-dimensional query. For details about supported extended JSON syntax, see **Extended JSON Syntax**.

Fuzzy query: Prefix query and wildcard query are supported.

```
SELECT * FROM example WHERE col3 LIKE 'test%';
SELECT * FROM example WHERE col3 LIKE 'start*end';
```

Aggregate analysis: Perform simple aggregate analysis using functions SUM, MAX, MIN, and AVG based on the combination of index columns.

```
SELECT sum(col1) from example WHERE pk1>='a' and pk2>=1000 and col1 <= 4 and col2 >= 2;
```

Full-text search: Specify a Chinese or an English analyzer to perform word segmentation and obtain data related to the word segmentation results.

```
SELECT * FROM example WHERE col4 LIKE '%+test -index%';
```

## Extended JSON Syntax

**Table 7-6** Extended JSON syntax

| Keyword | Function |
| --- | --- |
| filter | Keyword of queries in the extended JSON format |
| term | Whether a document contains a specific value in a query |
| match | Segments a queried value and performs full-text search. |
| range | Queries a specified field in a specified range (range query sub-keywords: **eq**, **gte**, **gt**, **lte**, and **lt**). |
| bool | Complex query that uses the combination of **must**, **should** and **must not** |
| must | A query clause in a bool query, including term, match, and range queries. The results must match all queries. |
| should | A query clause in a bool query, including term, match, and range queries. The results must match at least one of the queries. |
| must not | A query clause in a bool query, including term, match, and range queries. All matches are excluded from the results. |
| sort | Global index columns can be sorted in ascending or descending order. |

Example of a typical JSON query statement

```
{
  "filter": {
    "bool": {
      "should": [
        {"term": {"col1": 1, "col1": 2, "col1": 3, "col3": "testcase7"}}
      ],
      "must": [
        {"range": {"col2": {"lte": 7, "gt": 0}, "ck1": {"gte": 2}}},
        {"match": {"col4": "+lucene -index"}}
      ]
    }
  },
  "sort": [{"col1":"desc"}, {"col2":"asc"}]
}
```

Complete CQL statement

```
SELECT * from example where expr(index_lucene, '{"filter": {"bool": {"should": [{"term": {"col1": 1, "col1": 2,
"col1": 3, "col3": "testcase7"}}], "must": [{"range": {"col2": {"lte": 7, "gt": 0}, "ck1": {"gte": 2}}},{"match":
{"col4": "+lucene -index"}}]}}, "sort": [{"col1":"desc"}, {"col2":"asc"}]}');
```

Example of comparing a CQL statement with a JSON statement for typical queries

1. When partition keys (pk1 and pk2) are specified in a query, pk1 and pk2 must be separated from the JSON query condition. Otherwise, the performance will be affected.

```
SELECT * from example where pk1=*** and pk2=*** and expr(index_lucene, 'json');
```

2. Query condition: col1=1

```
SELECT * from example WHERE col1=1;
SELECT * from example WHERE expr(index_lucene, '{"filter": {"term": {"col1": 1}}}');
SELECT * from example WHERE expr(index_lucene, '{"filter": {"bool": {"must": [{"term": {"col1": 1}}]}}}');
```

These three statements are equivalent and recommended in sequence. You are advised to execute the first common CQL query. If the common CQL query is not supported, execute the extended JSON query.

3. Query condition: col1=1 and col2>=2

```
SELECT * from example WHERE col1=1 and col2>=2;
SELECT * from example WHERE expr(index_lucene, '{"filter": {"term": {"col1": 1},"range": {"col2": {"gte": 
2}}}}');
SELECT * from example WHERE expr(index_lucene, '{"filter": {"bool": {"must": [{"term": {"col1": 1}}, 
{"range": {"col2": {"gte": 2}}}]}}}');
```

The common CQL query is recommended.

4. Query condition: col1=1 and (col2<2 or col2>3)

```
SELECT * from example WHERE expr(index_lucene, '{"filter": {"bool": {"must": [{"term": {"col1": 1}}], 
"should": [{"range": {"col2": {"lt": 2}, "col2": {"gt": 3}}}]}}}');
SELECT * from example WHERE expr(index_lucene, '{"filter": {"bool": {"must": [{"term": {"col1": 1}}], 
"must_not": [{"range": {"col2": {"gte": 2, "lte": 3}}}]}}}');
```

These two statements are equivalent, but must_not is not recommended because of its lower query performance than should.

5. Query condition: col1 in (1,2,3,4) and (col2<2 or col2>3)

```
SELECT * from example WHERE expr(index_lucene, '{"filter": {"bool": {"should": [{"term": {"col1": 1, "col1": 
2, "col1": 3, "col1": 4}}], "should": [{"range": {"col2": {"lt": 2}, "col2": {"gt": 3}}}]}}}');
SELECT * from example WHERE expr(index_lucene, '{"filter": {"bool": {"should": [{"term": {"col1": 1, "col1": 
2, "col1": 3, "col1": 4}}], "must_not": [{"range": {"col2": {"gte": 2, "lte": 3}}}]}}}');
```

These two statements are equivalent, but must_not is not recommended because of its lower query performance than should.

6. Single query with partitions specified: pk1='a' and pk2=1000 and col1 in (1,2,3,4) and (col2<2 or col2>3)

```
SELECT * from example WHERE pk1='a' and pk2=1000 and expr(index_lucene, '{"filter": {"bool": {"should": 
[{"term": {"col1": 1, "col1": 2, "col1": 3, "col1": 4}}], "should": [{"range": {"col2": {"lt": 2}, "col2": {"gt": 
3}}}]}}}');
```

7. Query condition: (((ck1<2 or ck1>=4) and (col1<2 or col1 >3)) or (pk1 in ('a', 'b', 'c'))) or (5<=col2<15 and pk2 > 2000)

```
SELECT * from example WHERE expr(index_lucene, '{"filter": {"bool": {"should": [{"bool": {"should": 
[{"bool": {"must": [{"bool": {"should": [{"range": {"ck1": {"lt": 2}, "ck1": {"gte": 4}}}]}}, {"bool": {"should": 
[{"range": {"col1": {"lt": 2}, "col1": {"gt": 3}}}]}}]}}, {"bool": {"should": [{"term": {"pk1": "a", "pk1": "b", 
"pk1": "c"}}]}}]}}, {"bool": {"must": [{"range": {"col2": {"gte":5, "lte": 15}, "pk2": {"gt": 2000}}}]}}]}}}');
```

8. COUNT quey: The statement is as follows. You can also create the preceding query conditions in JSON format.

```
SELECT count(*) from example WHERE expr(index_lucene, '{"filter": {"bool": {"should": [{"bool": {"should": 
[{"bool": {"must": [{"bool": {"should": [{"range": {"ck1": {"lt": 2}, "ck1": {"gte": 4}}}]}}, {"bool": {"should":
```

```
[{"range": {"col1": {"lt": 2}, "col1": {"gt": 3}}}]}}]}}, {"bool": {"should": [{"term": {"pk1": "a", "pk1": "b",
"pk1": "c"}}]}}]}}, {"bool": {"must": [{"range": {"col2": {"gte":5, "lte": 15}, "pk2": {"gt": 2000}}}]}}]}}}}');
```

Precautions

- If common CQL statements can be executed, avoid JSON query statements as much as possible.
- For a single-partition query, the partition key condition must be used independently instead of being added to the JSON query condition. Otherwise, the single query performance will be affected.
- Avoid the must_not clause as much as possible.
- If query results always need to be sorted based on sorting orders of some index columns, you can specify their sorting orders as the default when creating indexes to improve performance.

# 7.3.8 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**.

# 7.3.9 How Do I Set Paging Query with Python?

## Using Paging Query

The fetch size controls how many rows will be fetched per page.

```
query = "SELECT * FROM space3.table3;"  # table3 contains 100 rows
statement = SimpleStatement(query, fetch_size=10)
```

After the setting is successful, for all sessions spawned with this configuration, 10 rows are fetched from the server at a time. When the cache (10 rows) is exhausted, the system triggers a request for fetching another 10 rows from the server and there can be a waiting period.

```
result = session.execute(statement)
# Print the number of current cache rows. The number is 10.
print(result.current_rows)
# The next page is automatically obtained.
for row in result:
    print(row)
```

## Saving and Reusing the Paging State

1. Save the current paging state.
   ```
   # Save the paging status.
   web_session['paging_stage'] = results.paging_state
   ```
2. Load and reuse the current paging state.
   ```
   statement = SimpleStatement(query, fetch_size=10)
   ps = web_session['paging_state']
   results = session.execute(statement, paging_state=ps)
   ```

   NOTE

   For more advanced usage and introduction, see **Paging Large Queries**.

# 7.3.10 How Can I Use the Counter Column?

The counter column of GeminiDB Cassandra API is used to keep track of metrics like page views, messages, and other measurable statistics.

## Supported Operations

In open-source Cassandra, the counter column stores values as 64-bit signed integers which can only be incremented or decremented. In addition, GeminiDB Cassandra API provides the following functions:

- You can set values in the counter column.

- You can delete counters by column, row, range, and partition.
- Write performance of a table with counters will deteriorate if they are deleted by range and partition, so these functions are disabled by default.
- To delete values by range and partition, set **counter_multiline_deletion_enabled** to **true**.

## Constraints

- A counter cannot be defined as a primary key.
- If a counter is used, all of the columns other than primary keys must be counters.
- TTL cannot be set for counters.
- INSERT cannot be executed for counters, which can only be inserted through UPDATE statements.

## Example

```
CREATE TABLE t0(pk int, ck int, v1 counter, v2 counter, v3 counter, PRIMARY KEY(pk, ck));
# Writes counter columns. Sets values for v1, increments values for v2, and decrements values for v3.
UPDATE t0 SET v1 = 1, v2 = v2 + 1, v3 = v3 - 1 WHERE pk = 1 AND ck = 1;
# Deletes counters by column.
DELETE v1 FROM t0 where pk = 1 AND ck = 1;
# Deletes counters by row.
DELETE FROM t0 where pk = 1 AND ck = 1;
```

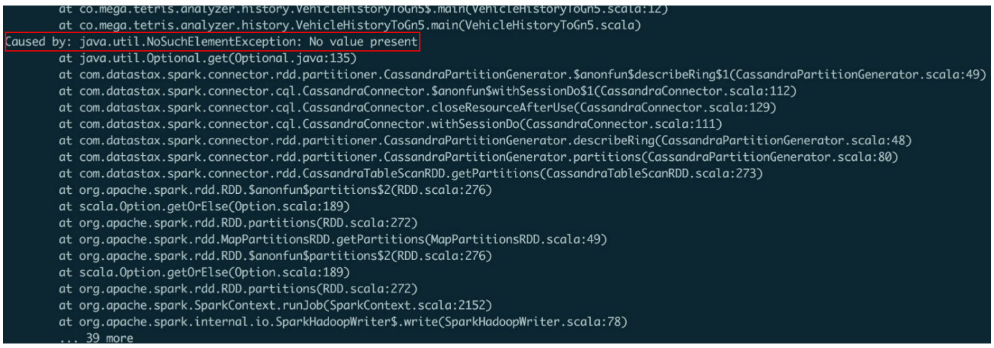
# 7.4 Database Connection

## 7.4.1 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 7-7** 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

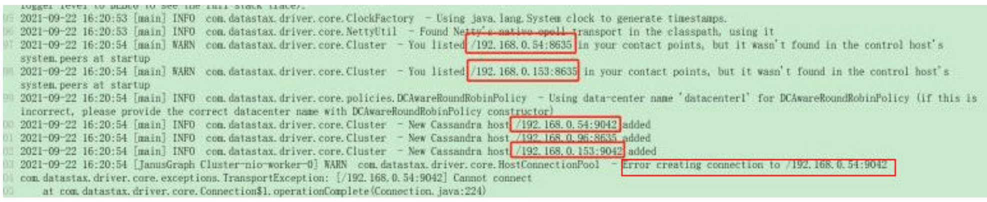
1. Grant the keyspace system query permission to **user1** as user **rwuser**.
2. Use spark-cassandra-connector 2.4.1.

# 7.4.2 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 7-14** Log information




## Possible Cause

Java Driver was not used correctly, as shown in **Figure 7-15**. Do not use addContactPointsWitPorts when using Java Driver and do not map each IP address.

**Figure 7-15** 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[illegible];
    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[illegible];
    System.out.println(dataList.toString());
    System.out.println[illegible];
    System.out.println[illegible];

    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();
```

# 7.4.3 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*.

# 7.4.4 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**.

# 7.5 Backup and Restoration

## 7.5.1 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.

# 7.6 Regions and AZs

## 7.6.1 What Is AZ and How Can I Select an AZ?

### AZ

An AZ is a part of a physical region with its own independent power supply and network. An AZ is generally an independent physical equipment room, ensuring independence of the AZ.

Each region contains multiple AZs. If one AZ becomes faulty, the other AZs in the same region can continue to provide services.

AZs in the same region can communicate with each other over an intranet.

### Selecting an AZ

You can deploy your instances in different AZs for high availability. If one of an AZ becomes faulty, databases in other AZs will not be affected. When selecting AZs:

- If only one AZ is available in a region, there is only one AZ in the region.
- The AZ of a purchased DB instance cannot be changed.
- The AZs in one region can communicate with each other over an intranet.

For more information, see **Regions and AZs**.

## 7.6.2 Can Different AZs Communicate with Each Other?

An AZ is a part of a physical region with its own independent power supply and network. An AZ is generally an independent physical equipment room, ensuring independence of the AZ.

Each region contains multiple AZs. If one AZ becomes faulty, the other AZs in the same region can continue to provide services normally.

By default, different AZs in the same VPC can communicate with each other through an internal network.

For more information, see **Regions and AZs**.

## 7.6.3 Can I Change the Region of a GeminiDB Cassandra Instance?

No. After an instance is created, its region cannot be changed.

# 7.7 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. For details, see **Resource Suspension and Release**.

### 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 **Renewal Management** 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 a Pay-per-Use Instance**.
- To delete a yearly/monthly instance, see **How Do I Unsubscribe from a Yearly/Monthly Instance?**.

# 8 GeminiDB HBase Instance

## 8.1 What Is GeminiDB HBase API?

GeminiDB Cassandra API is compatible with Apache HBase. You can access GeminiDB through open-source HBase Java SDKs or HBase Shell. Apache HBase Driver can be directly connected over a protocol, so you can smoothly migrate data to GeminiDB HBase instances without refactoring.

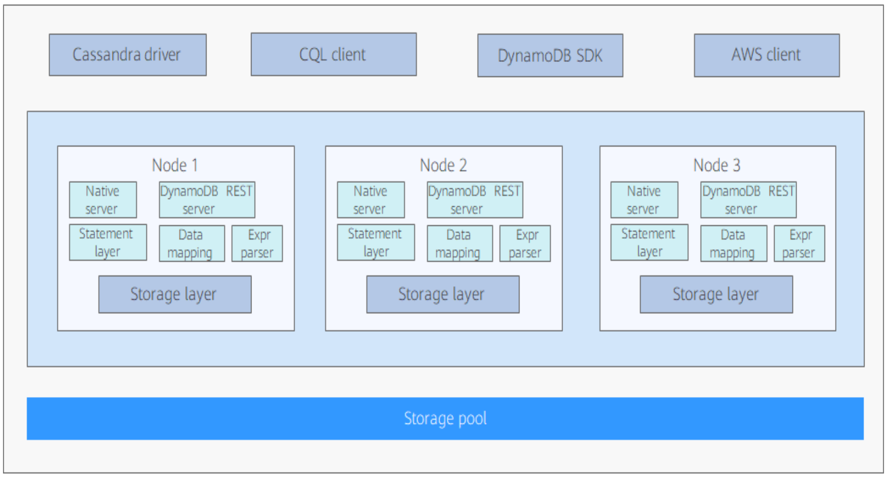
GeminiDB HBase API strictly complies with the HBase syntax and data model. Therefore, Apache HBase applications can be easily migrated to GeminiDB HBase instances. In addition, the GeminiDB HBase protocol provides multiple automatic management and O&M functions, such as minute-level cluster scaling, automated backup, fault detection, and multi-AZ fault tolerance. Furthermore, you are not required to be heavily involved in O&M and parameter optimization, which relieves you from the burden of managing and optimizing open-source clusters.

### Product Architecture

Embedded in kernel services of GeminiDB Cassandra API, GeminiDB HBase API provides standard features of HBase for external systems. In addition, the original Cassandra CQL protocol can be retained. You can select a proper database connection mode based on service requirements. The following figure shows the overall architecture.

**Figure 8-1** Architecture




## Highlights

- The read and write performance of instances with the same specifications is better than that of open-source HBase, providing a better performance solution.
- It is secure, reliable, and stable, and supports O&M methods such as multi-AZ DR and backup and restoration.
- Workloads can be smoothly migrated. GeminiDB HBase API is compatible with the model and connection methods of open-source HBase, greatly reducing migration costs and eliminating the need of modifying application logic.

# 8.2 Buying a GeminiDB HBase Instance

This section describes how to buy a GeminiDB HBase instance.

GeminiDB Cassandra, DynamoDB-Compatible, and HBase 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.

## Usage Notes

The default version of GeminiDB HBase is 3.11.

## Procedure

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

**Step 2** On the **Instances** page, click .

**Step 3** On the displayed page, select a billing mode, select **HBase** for **Compatible API**, configure information about your instance. Click **Next**.

**Figure 8-2** Billing mode and basic information (classic storage)

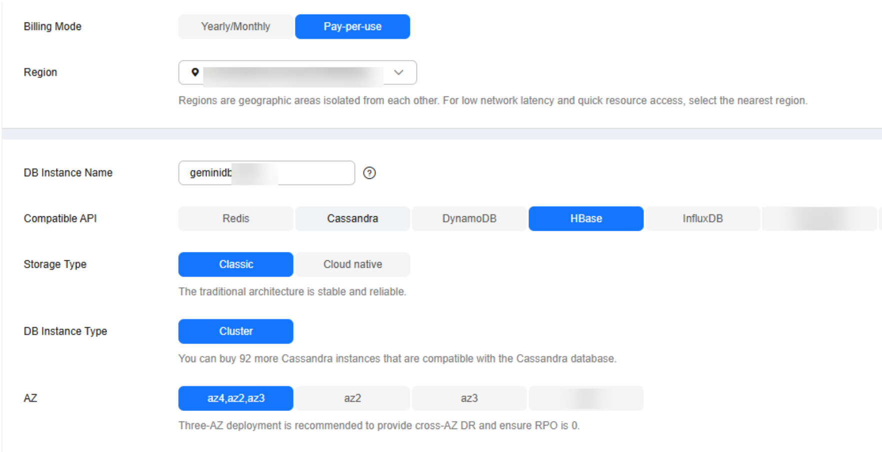



**Figure 8-3** Billing mode and basic information (cloud native storage)

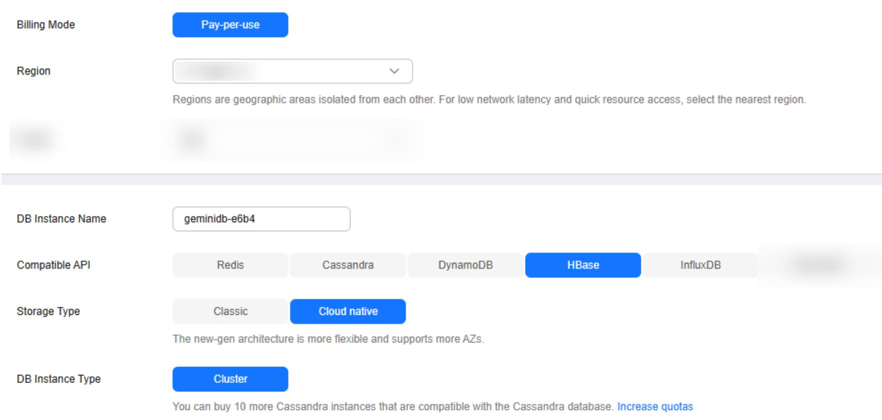


**Table 8-1** Billing mode

| Parameter | Description |
| --- | --- |
| Billing Mode | Select **Yearly/Monthly** or **Pay-per-use**.<br>● **Yearly/Monthly**<br>– Specify **Required Duration**. The system deducts fees from your account based on the service price.<br>– 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 a Yearly/Monthly Instance to Pay-per-Use**.<br>NOTE<br>■ Yearly/Monthly instances cannot be deleted directly. If such an instance is no longer required, unsubscribe from it. For details, see **How Do I Unsubscribe from a Yearly/Monthly Instance?**.<br>■ Yearly/Monthly instances with cloud native storage are now in OBT. To use such an instance, choose **Service Tickets > Create Service Ticket** in the upper right corner of the console and contact the customer service.<br>● **Pay-per-use**<br>– You are billed for usage based on how much time the service is in use.<br>– To use an instance for a long time, change its billing mode to yearly/monthly to reduce costs. For details, see **Changing a Pay-per-Use Instance to Yearly/Monthly**. |

**Table 8-2** Basic information

| Parameter | Description |
| --- | --- |
| Region | Region where a tenant is located<br>**NOTE**<br>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:<br>● Can be the same as an existing instance name.<br>● Can include 4 to 64 bytes and must start with a letter. It is case-sensitive and allows only letters, digits, hyphens (-), and underscores (_). |

| Parameter | Description |
| --- | --- |
| Compatible API | HBase<br>GeminiDB is compatible with mainstream NoSQL APIs, including Redis, DynamoDB, Cassandra, HBase, MongoDB, and InfluxDB. You can select GeminiDB APIs by following **How Do I Select an API?** |
| Storage Type | • **Classic**: classic architecture with decoupled storage and compute<br>• **Cloud native**: more flexible, new-gen version with support for more AZs<br>**NOTE**<br>• 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.<br>• Cloud native storage 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. |
| DB Instance Type | **Cluster**<br>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. |
| 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 8-4** Specifications and storage

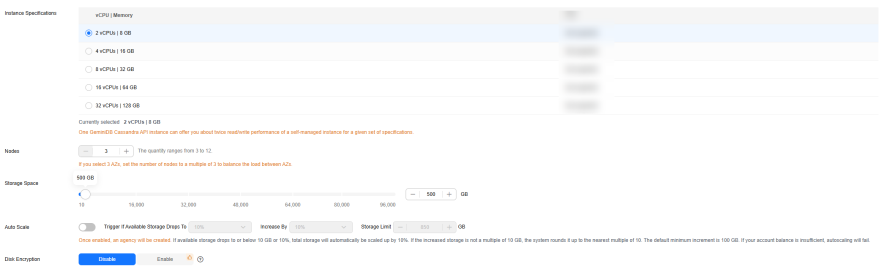


**Table 8-3** Specifications and storage

| Parameter | Description |
| --- | --- |
| Instance Specifications | Higher CPU specifications provide better performance. Select specifications as needed. |
| Nodes | The number of nodes ranges from 3 to 80. |

| Parameter | Description |
|---|---|
| Storage Space | Instance storage space. The range depends on the instance specifications.<br>To scale up storage, you need to add at least 1 GB each time. The value must be an integer. |
| Disk Encryption | You can determine whether to encrypt disks.<br>● **Disable**: Data is not encrypted.<br>● **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 I/O performance.<br>– **Key Name**: Select an existing key or create one.<br>– To use a shared key, ensure that you have created an agency. For details, see **Creating an Agency (by a Delegating Party)**. Select another account from the drop-down list to share the key of the current account. VPC owners can share the keys with one or multiple accounts through **Resource Access Manager (RAM)**. For details, see **Creating a Resource Share**.<br>– Enter a key ID. The key must be in the current region.<br>**NOTE**<br>– 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.<br>– An agency will be created after disk encryption is enabled.<br>– After an instance is created, the disk encryption status and key cannot be changed.<br>– The key cannot be disabled, deleted, or frozen when used, or the database becomes unavailable.<br>– For details about how to create a key, see "Creating a CMK" in *Data Encryption Workshop User Guide*. |

**Figure 8-5** Network settings

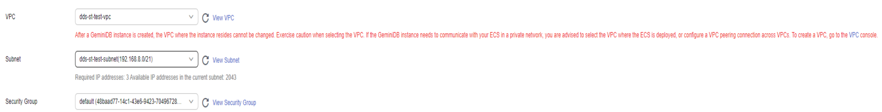


**Table 8-4** Network settings

| 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.<br>If there are no VPCs available, the system allocates resources to you by default.<br>For details, see "Creating a VPC" in the *Virtual Private Cloud User Guide*.<br>With VPC sharing, you can also use a VPC and subnet shared by another account.<br>VPC owners can share the subnets in a VPC with one or multiple accounts through Resource Access Manager (RAM). This allows for more efficient use of network resources and reduces O&M costs.<br>For more information about VPC subnet sharing, see **VPC Sharing** in the *Virtual Private Cloud User Guide*.<br>**NOTE**<br>• After an instance is created, the VPC where the instance is deployed cannot be changed.<br>• If you want to connect to an instance using an ECS over a private network, ensure that the instance and the ECS are in the same VPC. If they are not, create a **VPC peering connection** between them for access. |
| Subnet | A subnet where your instance is created. The subnet provides dedicated and isolated networks, improving network security.<br>**NOTE**<br>An IPv6 subnet cannot be associated with your instance. Select an IPv4 subnet. |
| Security Group | A security group controls access between instances and other services. When you select a security group, you must ensure that it allows your client to access your instances.<br>If there are no security groups available, the system allocates resources to you by default. |

**Figure 8-6** Database configuration

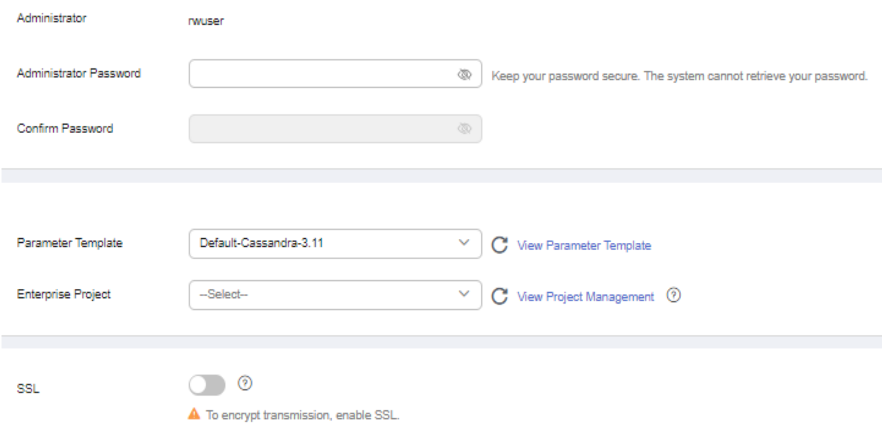



**Table 8-5** Database configuration

| Parameter | Description |
| --- | --- |
| Administrator | The default administrator account is **rwuser**. |
| Administrator Password | Password of the administrator account. The password:<br>● Must be 8 to 32 characters long.<br>● Must contain uppercase letters, lowercase letters, digits, and any of the following special characters: ~!#%^*-_=+?<br>● Cannot contain @ or /<br>● For security reasons, set a strong password. The system will verify the password strength.<br>Keep your password secure. The system cannot retrieve it if it is lost. |
| Confirm Password | Enter the administrator password again. |
| Enterprise Project | This parameter is provided for enterprise users.<br>An enterprise project groups cloud resources, so you can manage resources and members by project. The default project is **default**.<br>Select an enterprise project from the drop-down list. For more information about enterprise project, see ***Enterprise Management User Guide***. |
| Parameter Template | A parameter template contains engine configuration values that can be applied to one or more instances. You can modify the instance parameters as required after the DB instance is created.<br>After an instance is created, you can change the parameter template based on service requirements. |

| Parameter | Description |
| --- | --- |
| SSL | A security protocol. Secure Sockets Layer (SSL) certificates set up encrypted connections between clients and servers, preventing data from being tampered with or stolen during transmission.<br>You are advised to enable SSL connection to improve data security.<br>**NOTE**<br>If SSL is not enabled when you create an instance, you can enable it after the instance is created. For details, see **Encrypting Data over SSL for a GeminiDB Cassandra Instance**. |

**Table 8-6** Tags

| Parameter | Description |
| --- | --- |
| Tag | Tags a instance. This parameter is optional. Adding tags helps you better identify and manage your instances.<br>A maximum of 20 tags can be added for each instance.<br>If your organization has configured a tag policy for , you need to add a tag to the instance based on the tag policy. If the tag does not comply with the tag policy, the instance may fail to be created. Contact the organization administrator to learn details about the tag policy.<br>A tag consists of a tag key and a tag value.<br>● A tag key is mandatory if the instance is going to be tagged.<br>Each tag key is unique for each instance. It can contain a maximum of 128 characters, cannot start with **_sys_**, and cannot start or end with a space. Only letters, digits, spaces, and the following special characters are allowed: -_@.:/+=<br>● A tag value is optional if the instance is going to be tagged. The value can be empty.<br>The value can contain a maximum of 255 characters. Only letters, digits, spaces, and the following special characters are allowed: _.-:+=@/ |

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

- 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 your instances.

- The instance status is displayed as **Creating**.
- 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 a DB instance is created.

NOTE

After the instance is created, Cassandra 3.11.3 is displayed in the **Compatible API** column on the **Instances** page and is compatible with HBase. The usage is the same as that of HBase.

**----End**

# 8.3 Connecting to a GeminiDB HBase Instance

This section describes how to connect to a GeminiDB HBase instance using a private IP address and Java.

## Prerequisites

- A GeminiDB HBase instance has been created and is running normally. For details about how to create a GeminiDB HBase instance, see **Buying a GeminiDB HBase 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.
- Download the **HBase client**. Click a directory of the latest version 2.6.*X* and download **hbase-2.6.X-client-bin.tar.gz**. For example, if the latest version is 2.6.1, click that directory and download **hbase-2.6.1-client-bin.tar.gz**. HBase 1.*X* is not recommended due to compatibility issues.

## Viewing the IP Address of an Instance

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

**Step 2** On the **Instances** page, click the name of the target instance.

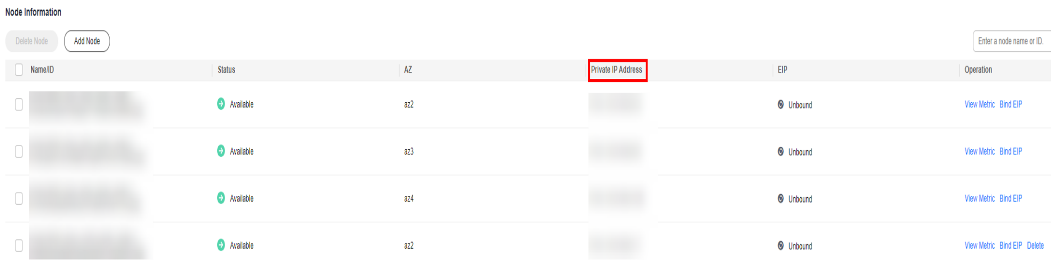
**Method 1**

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

NOTE

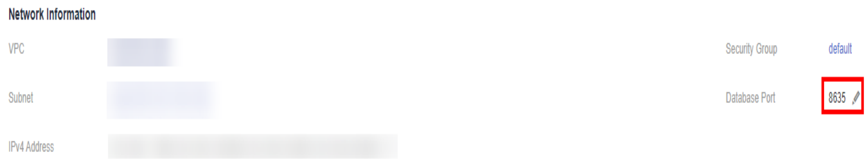
Public IP addresses cannot be bound to GeminiDB HBase instances.

**Figure 8-7** Viewing the IP address




In the **Network Information** area, you can view the port of the GeminiDB HBase instance. The default port displayed on the page is 8635, but the default port in use is 2181.

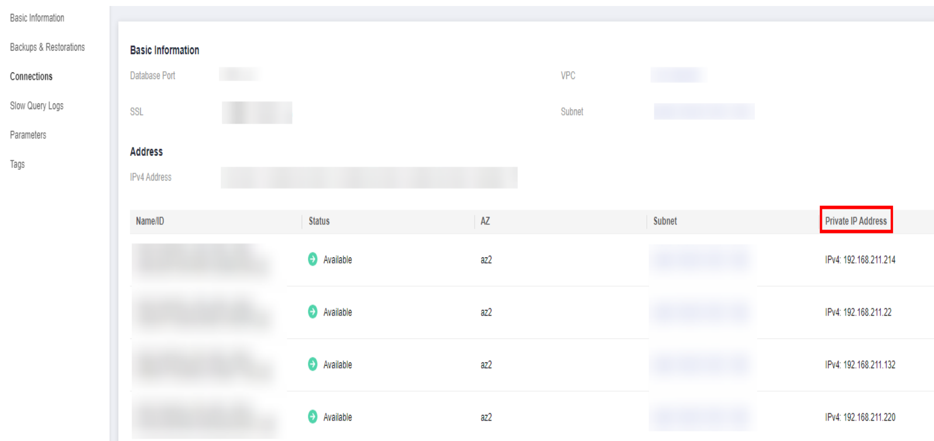
**Figure 8-8** Viewing the port number




### Method 2

In the navigation pane on the left, click **Connections**.

**Figure 8-9** Viewing the IP addresses and port number




**----End**

## Connecting to an Instance over a Private Network

**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 **HBase client** installation package to the ECS.

**Step 3** Run the following command to decompress the package:

```
tar -xvf hbase-2.6.1-client-bin.tar.gz
```

**Step 4** Add the following configurations to **conf/hbase-site.xml** in the client directory and set **value** to the IP address of your instance. Use commas (,) to separate multiple IP addresses. The private IP address can be obtained by following **Viewing the IP Address of an Instance**.

```
<configuration>
  <property>
    <name>hbase.zookeeper.quorum</name>
    <value>127.0.0.1,127.0.0.2,127.0.0.3</value>
  </property>
</configuration>
```

**Step 5** Go to the **bin** directory of the decompressed client and run the following command to connect to the instance: Replace **YOUR_USERNAME** and **YOUR_PASSWORD** with the user password set during instance creation. The username is fixed to **rwuser**.

```
export HADOOP_PROXY_USER="YOUR_USERNAME"
export HADOOP_USER_NAME="YOUR_PASSWORD"
./hbase shell
```

**Step 6** If information similar to the following is displayed, the connection was successful.

```
hbase:001:0>
```

**----End**

## Connecting to an Instance Using Java

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

For details about how to obtain the private IP address and port number, see **Viewing the IP Address of an Instance**.

**Step 2** Log in to the ECS. For details, see **Logging In to an ECS** in *Getting Started with Elastic Cloud Server*.

**Step 3** Add the following Maven dependencies to your project. HBase 1.*X* is not recommended due to compatibility issues.

```
<dependency>
    <groupId>org.apache.hbase</groupId>
    <artifactId>hbase-client</artifactId>
    <version>2.6.1</version>
</dependency>
```

**Step 4** Edit the code for connecting to the instance.

```
import org.apache.hadoop.conf.Configuration;
import org.apache.hadoop.hbase.HBaseConfiguration;
import org.apache.hadoop.hbase.TableName;
import org.apache.hadoop.hbase.client.*;
import org.apache.hadoop.hbase.util.Bytes;

import java.io.IOException;

public class HBaseExample {
    public static void main(String[] args) throws IOException {
        // Creates a configuration object and sets HBase connection parameters.
        Configuration config = HBaseConfiguration.create();
        config.set("hbase.zookeeper.quorum", "your_hbase_instance_quorum");
        config.set("hbase.zookeeper.property.clientPort", "your_hbase_instance_port");
```

```
        // Enters a username and password.
        UserGroupInformation ugi = UserGroupInformation.createProxyUser("your_user_name",
UserGroupInformation.createRemoteUser("your_password"));

        // Establishes a connection to the HBase instance.
        Connection connection = ConnectionFactory.createConnection(config, User.create(ugi));

        try {
            // Obtains table objects.
            TableName tableName = TableName.valueOf("your_table_name");
            Table table = connection.getTable(tableName);

            // Inserts data.
            Put put = new Put(Bytes.toBytes("row_key"));
            put.addColumn(Bytes.toBytes("cf"), Bytes.toBytes("col"), Bytes.toBytes("value"));
            table.put(put);

            // Obtains a single row of data.
            Get get = new Get(Bytes.toBytes("row_key"));
            Result result = table.get(get);
            byte[] value = result.getValue(Bytes.toBytes("cf"), Bytes.toBytes("col"));
            System.out.println("Value: " + Bytes.toString(value));

        } finally {
           // Closes the connection.
            connection.close();
        }
    }
}
```

**Step 5** Run the sample code to check whether the result is normal.

**----End**

# 8.4 How Do I Set Pre-partition Keys When Creating a Table on a GeminiDB HBase Instance?

This section describes how to set a pre-partition key when creating a table on a GeminiDB HBase instance.

## What Is Pre-partitioning

On a GeminiDB HBase instance, data is stored in different data partitions. Row key prefixes uniquely identify entities within each partition. Data is evenly distributed across partitions to balance loads and improve cluster performance.

For example, if two pre-partition keys are set to **[1111, 2222]** during table creation, data is divided into three ranges. The partitions to which the data belongs are divided based on the lexicographic order of row keys and partition keys. If **rowkey < '1111'** is specified, data is stored in the first partition. If **'1111' <= rowkey < '2222'** is specified, data is stored in the second partition. If **rowkey >='2222'** is specified, data is stored in the third partition. Ideally, the three partitions belong to different nodes. If partition keys are not properly set, partitions may belong to one cluster node.

## Designing Pre-partition Keys

Theoretically, customer's application data can be evenly distributed by prefix in each partition. On a GeminiDB HBase instance, the ideal data volume in a partition is about 100 GB. There is no upper limit on the data volume in a single partition. If there is more than 100 GB of data in a partition, the data will be automatically partitioned. You can choose **Service Tickets > Create Service Ticket** in the upper right corner of the console to disable automated partitioning.

- Example 1:

If the first digit of row key values are evenly distributed from **0** to **9**, 10 partition keys can be set: **[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]**. Values starting with these digits belong to their own partitions.

- Example 2:

If the first two digits of row key values are evenly distributed from **00** to **FF** and the estimated data volume in each partition is about 100 GB, 256 partition keys are recommended: **[00, 01, 02, ..., FD, FE, FF]**.

## Specifying Pre-partitions During Table Creation

On a GeminiDB HBase instance, HBase Shell or Java code can be used to specify pre-partitions during table creation.

- Specify pre-partitions using HBase Shell when creating a table.

```
create 'tb','cf1','cf2', 'cf3', SPLITS => ['1111', '2222', '3333']
```

You can replace **'1111'**, **'2222'**, and **'3333'** with other custom partition key values. Use commas (,) to separate multiple values.

- Specify pre-partitions using Java code when creating a table.

```
import java.util.ArrayList;
import java.util.List;
import org.apache.hadoop.conf.Configuration;
import org.apache.hadoop.hbase.HBaseConfiguration;
import org.apache.hadoop.hbase.TableName;
import org.apache.hadoop.hbase.client.Admin;
import org.apache.hadoop.hbase.client.ColumnFamilyDescriptor;
import org.apache.hadoop.hbase.client.ColumnFamilyDescriptorBuilder;
import org.apache.hadoop.hbase.client.Connection;
import org.apache.hadoop.hbase.client.ConnectionFactory;
import org.apache.hadoop.hbase.client.TableDescriptor;
import org.apache.hadoop.hbase.client.TableDescriptorBuilder;
import org.apache.hadoop.hbase.security.User;
import org.apache.hadoop.security.UserGroupInformation;
public class ExampleCreateTable
{
    public static void main(String[] args) throws Throwable
    {
        // Create HBase configuration
        Configuration hbaseConfig = HBaseConfiguration.create();
// Enters an IP address of the HBase cluster.
        hbaseConfig.set("hbase.zookeeper.quorum", "127.0.0.1");
// Sets the default port number to 2181.
        hbaseConfig.set("hbase.zookeeper.property.clientPort", "2181");
        TableName tableName = TableName.valueOf("default", "tb1");
        // Enters a username and password.
        UserGroupInformation ugi = UserGroupInformation.createProxyUser("your_user_name",
UserGroupInformation.createRemoteUser("your_password"));
        // Establishes a connection to the HBase instance.
        try (Connection connection = ConnectionFactory.createConnection(hbaseConfig, User.create(ugi)))
        {
            Admin admin = connection.getAdmin();
            // provide your split key here
            byte[][] splitkey = new byte[][]{ "row1".getBytes(), "row2".getBytes()};
            // 5 column families
```

```
            List<ColumnFamilyDescriptor> cfs = new ArrayList<>();
            cfs.add(ColumnFamilyDescriptorBuilder.newBuilder("cf1".getBytes()).build());
            cfs.add(ColumnFamilyDescriptorBuilder.newBuilder("cf2".getBytes()).build());
            cfs.add(ColumnFamilyDescriptorBuilder.newBuilder("cf3".getBytes()).build());
            cfs.add(ColumnFamilyDescriptorBuilder.newBuilder("cf4".getBytes()).build());
            cfs.add(ColumnFamilyDescriptorBuilder.newBuilder("cf5".getBytes()).build());
            TableDescriptor tableDescriptor =
TableDescriptorBuilder.newBuilder(tableName).setColumnFamilies(cfs).build();
            // create table
            admin.createTable(tableDescriptor, splitkey);
        }
    }
}
```

# 8.5 GeminiDB HBase Compatibility List

Currently, GeminiDB HBase does not support APIs related to cluster O&M and management. If you need to perform instance-level operations, such as restarting an instance, you can use functions on the console

**Table 8-7**

| Function | API | Description | Supported |
| --- | --- | --- | --- |
| Data | Get | Get: single-line query. | Yes |
| | | Filter: server-side filtering. | SingleColumnValueFilter and PageFilter are supported. Other filters are in OBT. |
| | | Consistency: strong data consistency. | Yes |
| | | Versions: multi-version query. | Yes |
| | | Batch: batch single-row read | Yes |
| | Put | Put: single-row insertion. | Yes |
| | | Condition: condition insertion. | Yes |
| | | TTL: automatic deletion of expired data. | Yes |
| | | Batch: batch single-row write | Yes |

| Function | API | Description | Supported |
|---|---|---|---|
| | | Versions: multi-version write. | Yes |
| | Delete | Delete: all data records deletion in a single line. | Yes |
| | | Delete: deletion of a specified column family or qualifier. | Yes |
| | | Versions: multi-version deletion. | Yes |
| | | Batch: batch single-row deletion | No |
| | | Prefix Delete: prefix deletion | Yes |
| | Scan | Scan: full scan. | Yes |
| | | Scan: specified start or stop line scanning. | Yes |
| | | TimeRange: time range filtering. | Yes |
| | | Filter: server filtering | SingleColumnValueFilter and PageFilter are supported. Other filters are in OBT. |
| | | Versions: multi-version query. | Yes |
| | | Reversed: reverse query | Yes |
| | Increment | Common increment operation. | No |
| | Append | Common append operation. | No |
| | Bulk Load | Batch data import from a file. | No |

| Function | API | Description | Supported |
| --- | --- | --- | --- |
| Metadata operations | Table | Create: common table creation operation. | Yes |
| | | SplitKey: keys specified when a table is created. | Yes |
| | | Region: automatic splitting. | Yes |
| | | Disable/Delete: table disabling or deletion. | Yes |
| | | Compress: compression algorithm (built-in data compression) specification. | No |
| | | Alter Table: column family addition. | Yes |
| | | Alter Table: column family deletion. | No |
| | Cluster Management | You do not need to pay attention to this parameter. | |
| | Namespace | Namespace creation or deletion. | Yes |
| | | Namespace attributes cannot be specified. | No |
| Additional support | Coprocessor | User-defined plugins are supported. | No |
| | Secondary Index | Customized queries for column fields are supported. | No |

# 8.6 Performance White Paper

## 8.6.1 Test Method

This section describes performance testing of GeminiDB HBase instances, including the test environment, procedure, and results.

### Test Environment

- Region: CN North-Beijing4
- AZ: AZ1
- Elastic Cloud Server (ECS): h3.4xlarge.2 with 16 vCPUs, 32 GB of memory, and CentOS 7.5 64-bit image
- Stress test data model: 20 columns in a single row and 20 bytes in each column
- Instance specifications: All specifications described in **Table 8-8**.

**Table 8-8** Specifications

| No. | Specifications |
| --- | --- |
| cluster1 | 16U64GB |

### Test Tool

In this test, the YCSB 0.17.0 test tool provided by the open-source community is used to connect to the GeminiDB HBase-compatible instance and fix issues in high concurrency scenarios.

For details on how to use this tool, see **YCSB**.

### Test Metrics

Operations per Second (OPS): operations executed by a database per second

### Test Procedure

1. Configure the **workload** file.

   Set values for fields **readproportion**, **insertproportion**, **updateproportion**, **scanproportion**, and **readmodifywriteproportion** in the file by referring to **Table 8-9**.

2. Use workload-insert-mostly as an example. Run the following command to prepare test data:
   ```
   nohup ./bin/ycsb load hbase -s -P workloads/workload-insert-mostly -P hbase.properties -p operationcount=400000000 -p recordcount=400000000 -threads ${thread} -p maxexecutiontime=${maxexecutiontime} -s 1> data_load.log 2>&1 &
   ```

3. Use workload-insert-mostly as an example. Run the following command to test performance:
   ```
   nohup ./bin/ycsb run hbase -s -P workloads/workload-insert-mostly -P hbase.properties -p operationcount=9000000 -p recordcount=9000000 -threads ${thread} -s 1> workload-insert-mostly_run.log 2>&1 &
   ```

## Test Model

- Workload model

**Table 8-9** Test Model

| Test Model No. | Test Model |
|---|---|
| workload-read (single-line read) | 100% read |
| workload-insert (single-line write) | 100% insert |
| workload-range-read (read) | 100% range read |

- Preset data volume

  In this performance test, 2 billion data records are preset for instances of each specification. Each data record contains 20 fields, and the size of each field is 20 bytes.

# 8.6.2 Test Data

The OPS of instances of different specifications can be tested using different service models with the same preset data volume. For details, see the numbers in bold in **Table 8-10**.

**Table 8-10** Test data

| DB Instance Type | **Data Size** | Service Performance-Throughput (QPS) | | | Service Performance-P99 Latency(ms) | | |
|---|---|---|---|---|---|---|---|
| | | Single-Line Write | Single-Line Read | Range Read | Single-Line Write | Single-Line Read | Range Read |
| GeminiDB HBase instance | 2 billion | **258729** | **87581** | **754631** | **3.5** | **6** | **20** |
| Open-Source HBase Instance | 2 billion | **139723** | **18931** | **63350** | **3.9** | **5** | **46** |

NOTE

- Operations per Second (OPS): operations executed by a database per second
- Test Model No.: test model sequence number. **Table 8-9** lists the test models.

# 8.7 Instructions for Use

## 8.7.1 How Can I Delete Rows Based On Prefixes?

You can delete rows with keys matching a given prefix on GeminiDB HBase instances. This function takes effect quickly. You do not need to scan data before deleting it anymore. Unlike Apache HBase that supports only single-row deletion, GeminiDB HBase allows you to delete rows based on prefixes, which is more convenient and faster. This capability provides significant advantages for tasks like range data clearance and historical data purging.

**CAUTION**

Incorrect use of this function may have a significant impact on data. Before using this function, ensure that you have fully understood the following usage notes.

### Usage Notes (Mandatory)

A message is displayed, indicating the data has been deleted as expected, but it is not deleted immediately and is only marked for deletion. The marked data needs to be gradually deleted in stored procedures, and generated range tombstones will also be deleted. Therefore, to ensure that this does not affect database performance, you need to comply with the following conventions:

- Do not repeatedly delete and write a single range of data within a short period of time.
- Do not delete massive volumes of data within a short period of time.
- Do not scan data that has been deleted.
- Verify the range in advance to prevent accidental deletion of a large amount of data.

### Typical Violations

- In a specific data range, rows with 1 billion keys matching a given prefix are deleted 50,000 times within a day. At the same time, a large amount of data in the same range is written.
- Rows are deleted based on a given prefix 100,000 or more times within a day.
- A large amount of data is deleted by mistake based on short prefixes (for example, **0** or **a**) without verification.

**CAUTION**

Severe violations will increase the read latency, result in failed requests, and affect read and write performance. You need to check the service status in a timely manner. Before final deletion, verify the results in a test environment.

If the preceding issues occur for a large amount of data, stop using this function immediately and consult experts. In the upper right corner of the console, choose **Service Tickets > Create Service Ticket** and contact the customer service.

## How to Use

The customer can add additional attributes to mark deletion requests as those based on prefixes. After the requests are marked, only **key** takes effect. Other parameters, such as the specified column and qualifier, will not take effect. Data that matches a prefix is deleted immediately.

Currently, this function can be used only through the Java HBase client. In the following Java code, all keys starting with **row1** will be deleted.

```
Delete delete = new Delete(Bytes.toBytes("row1"));
delete.setAttribute("PREFIXDELETE", "true".getBytes(StandardCharsets.UTF_8));
table.delete(delete);
```

## FAQs

1. Q: If a request times out or fails, has my data been deleted?

   A: GeminiDB HBase does not provide transactions and cannot ensure atomicity. In the case of a failed request, the target data may be completely or partially deleted. If the request is successful, all data is deleted. If the failure is caused by network disconnection or other reasons, try again.

2. Q: How can I perform a large number of deletions for historical data based on prefixes?

   A: Specify the range of historical data to be deleted. Verification in a test environment is recommended to prevent unexpected data deletion. No more than 2,000 times are recommended for this function. In a short period of time, a few deletions based on prefixes can be applied to massive volumes of data, which can meet your requirements. Continuously check the read latency while data is deleted. If there is any exception, stop deleting data immediately.

# 8.7.2 How Can I Connect to a GeminiDB HBase Instance over TLS (SSL)?

## Preparations

**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 instance name. The **Basic Information** page is displayed.

**Step 4** In the **DB Information** area, toggle on for **SSL** and click the download icon next to **SSL** to download the **ca.cert** file.


**Step 5** Upload the **ca.cert** file to the ECS.

**Step 6** Connect to a GeminiDB HBase instance by following **Connecting to a GeminiDB Cassandra Instance on the DAS Console** and run the following command to save the server certificate file to a truststore and set the password (default value: **PASSWORD**):

```
keytool -importcert -alias hw -file ca.cert -keystore truststore.jks -storepass PASSWORD
```

**----End**

## Establishing a TLS Connection Using HBase Shell

Add the following configuration items to the **hbase-site.xml** file on the client:

- The value of **hbase.rpc.tls.truststore.location** is the path of the **truststore.jks** file generated in **Step 6**.
- Set **hbase.rpc.tls.truststore.password** to the password set in **Step 6**. The default password is **PASSWORD**.

```
<property>
  <name>hbase.client.netty.tls.enabled</name>
  <value>true</value>
</property>
<property>
  <name>hbase.rpc.tls.truststore.location</name>
  <value>conf/truststore.jks</value>
</property>
<property>
  <name>hbase.rpc.tls.truststore.password</name>
  <value>PASSWORD</value>
</property>
```

Start HBase Shell to check whether the connection is successful.

## Establishing a TLS Connection Using a Java Application

Java code:

```
package com.huawei;

import java.util.ArrayList;
import java.util.List;

import org.apache.hadoop.conf.Configuration;
import org.apache.hadoop.hbase.HBaseConfiguration;
import org.apache.hadoop.hbase.TableName;
import org.apache.hadoop.hbase.client.Admin;
import org.apache.hadoop.hbase.client.ColumnFamilyDescriptor;
import org.apache.hadoop.hbase.client.ColumnFamilyDescriptorBuilder;
import org.apache.hadoop.hbase.client.Connection;
import org.apache.hadoop.hbase.client.ConnectionFactory;
import org.apache.hadoop.hbase.client.TableDescriptor;
import org.apache.hadoop.hbase.client.TableDescriptorBuilder;
```

```
import static org.apache.hadoop.hbase.io.crypto.tls.X509Util.HBASE_CLIENT_NETTY_TLS_ENABLED;
import static
org.apache.hadoop.hbase.io.crypto.tls.X509Util.HBASE_CLIENT_NETTY_TLS_VERIFY_SERVER_HOSTNAME;
import static org.apache.hadoop.hbase.io.crypto.tls.X509Util.TLS_CONFIG_TRUSTSTORE_LOCATION;
import static org.apache.hadoop.hbase.io.crypto.tls.X509Util.TLS_CONFIG_TRUSTSTORE_PASSWORD;

public class ExampleTlsConnection
{
    public static void main(String[] args) throws Throwable
    {
        Configuration conf = HBaseConfiguration.create();
        // todo: change connect address
        conf.set("hbase.zookeeper.quorum", "127.0.0.1");
        conf.set("hbase.zookeeper.property.clientPort", "2181");

        // todo: change those two strings.
        String path = "/absolute/path/to/your/truststore.jks";
        String password = "your_pass_word";

        conf.setBoolean(HBASE_CLIENT_NETTY_TLS_ENABLED, true);
        conf.setBoolean(HBASE_CLIENT_NETTY_TLS_VERIFY_SERVER_HOSTNAME, false);
        conf.set(TLS_CONFIG_TRUSTSTORE_LOCATION, path);
        conf.set(TLS_CONFIG_TRUSTSTORE_PASSWORD, password);

        try (Connection connection = ConnectionFactory.createConnection(conf))
        {
            Admin admin = connection.getAdmin();
            TableName tb = TableName.valueOf("test");

            List<ColumnFamilyDescriptor> cfs = new ArrayList<>();
            cfs.add(ColumnFamilyDescriptorBuilder.newBuilder("cf1".getBytes()).build());
            cfs.add(ColumnFamilyDescriptorBuilder.newBuilder("cf2".getBytes()).build());
            cfs.add(ColumnFamilyDescriptorBuilder.newBuilder("cf3".getBytes()).build());
            cfs.add(ColumnFamilyDescriptorBuilder.newBuilder("cf4".getBytes()).build());
            cfs.add(ColumnFamilyDescriptorBuilder.newBuilder("cf5".getBytes()).build());

            TableDescriptor tableDescriptor =
TableDescriptorBuilder.newBuilder(tb).setValue().setColumnFamilies(cfs).build();

            admin.createTable(tableDescriptor);
        }

    }
}
```

## 8.7.3 How Can I Count the Number of Rows in a Table?

This section describes how to count the total number of rows in a table of a GeminiDB HBase instance.

You can run COUNT on the HBase client to count the number of rows in a table of a GeminiDB HBase instance. For details, see **Connecting to an Instance over a Private Network**.

COUNT scans all data in a table in batches. You are advised to run COUNT on an ECS in the same VPC as the GeminiDB HBase instance. The scan speed varies depending on the table structure. Run the following statement to count the total number of rows in the table:

```
count 'table_name'
```