## **GeminiDB Redis**

## **User Guide**

 Issue
 01

 Date
 2025-06-30





HUAWEI TECHNOLOGIES CO., LTD.

#### Copyright © Huawei Technologies Co., Ltd. 2025. All rights reserved.

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

#### **Trademarks and Permissions**

NUAWEI and other Huawei trademarks are trademarks of Huawei Technologies Co., Ltd. All other trademarks and trade names mentioned in this document are the property of their respective holders.

#### Notice

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

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

## Huawei Technologies Co., Ltd.

- Address: Huawei Industrial Base Bantian, Longgang Shenzhen 518129 People's Republic of China Website: https://www.huawei.com
- Email: <u>support@huawei.com</u>

## **Security Declaration**

## Vulnerability

Huawei's regulations on product vulnerability management are subject to the *Vul. Response Process.* For details about this process, visit the following web page:

https://www.huawei.com/en/psirt/vul-response-process

For vulnerability information, enterprise customers can visit the following web page: <u>https://securitybulletin.huawei.com/enterprise/en/security-advisory</u>

## **Contents**

1 Service Overview	1
1.1 Enterprise-Level Features	1
1.2 Highlights	3
1.3 Product Architecture	7
1.4 Application Scenarios	10
1.5 Compatible API and Versions	11
1.6 Instance Specifications	
1.7 Instance Statuses	16
1.8 Constraints	17
2 Billing	22
2.1 Billing Overview	22
2.2 Billing Modes	23
2.2.1 Overview	23
2.2.2 Yearly/Monthly Billing	24
2.2.3 Pay-per-Use Billing	29
2.3 Billing Items	33
2.4 Billing Examples	35
2.5 Billing Mode Changes	
2.5.1 Overview	
2.5.2 Changing a Pay-per-Use Instance to Yearly/Monthly	
2.5.3 Changing a Yearly/Monthly Instance to Pay-per-Use	41
2.6 Renewing Subscriptions	
2.6.1 Overview	43
2.6.2 Manually Renewing an Instance	
2.6.3 Auto-renewing an Instance	48
2.7 Bills	50
2.8 Arrears	54
2.9 Billing Termination	56
2.10 Cost Management	58
2.10.1 Cost Composition	58
2.10.2 Cost Allocation	
2.10.3 Cost Analysis	59
2.10.4 Cost Optimization	60

2.11 Billing FAQs	60
2.11.1 What Are the Differences Between Yearly/Monthly and Pay-per-Use Billing?	60
2.11.2 Can I Switch Between Yearly/Monthly and Pay-per-Use Billing?	60
2.11.3 How Do I Renew a Single or Multiple Yearly/Monthly Instances?	61
2.11.4 How Do I Unsubscribe from a Yearly/Monthly Instance?	62
3 Getting Started with GeminiDB Redis API	65
3.1 Getting to Know GeminiDB Redis API	65
3.2 Buying and Connecting to a Cluster Instance	67
3.3 Buying and Connecting to a Primary/Standby Instance	78
3.4 Getting Started with Common Practices	88
4 Working with GeminiDB Redis API	
4.1 Using IAM to Grant Access to GeminiDB Redis API	90
4.1.1 Creating a User and Granting GeminiDB Redis API Permissions	90
4.1.2 Custom Policies of GeminiDB Redis API	
4.2 Buying a GeminiDB Redis Instance	93
4.2.1 Buying a GeminiDB Redis Cluster Instance	
4.2.2 Buying a Primary/Standby GeminiDB Redis Instance	104
4.3 Instance Connection and Management	114
4.3.1 Connecting to a GeminiDB Redis Instance	114
4.3.2 Connecting to a GeminiDB Redis Instance on the DAS Console	117
4.3.3 Connecting to a GeminiDB Redis Instance Over a Private Network	123
4.3.3.1 Connecting to an Instance Using a Load Balancer Address (Recommended)	123
4.3.3.2 Connecting to an Instance Using a Private Domain Name	125
4.3.3.3 Connecting to an Instance Using a Private IP Address	131
4.3.4 Connecting to a GeminiDB Redis Instance Over a Public Network	132
4.3.4.1 Connecting to an Instance Using an EIP Bound to a Load Balancer (Recommended)	132
4.3.4.2 Connecting to an Instance Using an EIP	136
4.3.4.3 Connecting to an Instance Using a Public Domain Name	138
4.3.5 Connection Information Management	144
4.3.5.1 Setting Security Group Rules for a GeminiDB Redis Instance	145
4.3.5.2 Viewing the IP Address and Port Number of a GeminiDB Redis Instance	147
4.3.5.3 Changing the Port of a GeminiDB Redis Instance	149
4.3.5.4 Binding an EIP to a GeminiDB Redis Instance Node	149
4.3.5.5 Encrypting Data over SSL for a GeminiDB Redis Instance	151
4.3.5.6 Connecting a GeminiDB Redis Instance over SSL	152
4.3.5.7 Changing the Security Group of a GeminiDB Redis Instance	155
4.3.5.8 Configuring Private Network Access to a GeminiDB Redis Instance	156
4.4 Data Migration	158
4.4.1 Migration Solution	158
4.4.2 (Recommended) Using DRS to Migrate Data from a GeminiDB Redis Instance to an Open- Redis Instance	Source 160
4.4.3 Migrating the Alibaba Cloud Database Redis/Tair To GeminiDB Redis	

4.4.4 (Recommended) Using DRS to Migrate Data from Open-source Redis or Redis Cluster to Gemini Redis API	DB 165
4.4.5 Migrating Data from Redis to GeminiDB Redis API Using Redis-Shake	. 166
4.4.6 Using Redis-Shake to Import an RDB or AOF File to a GeminiDB Redis Instance	.171
4.4.7 (Recommended) Importing Data to Restore RDB Files to a GeminiDB Redis Instance	.173
4.4.8 Migrating Data from Kvrocks to GeminiDB Redis API	. 176
4.4.9 Migrating Data from Pika to GeminiDB Redis API	.179
4.4.10 Migrating Data from SSDB to GeminiDB Redis API	. 180
4.4.11 Migrating Data from LevelDB to GeminiDB Redis API	.183
4.4.12 Migrating Data from RocksDB to GeminiDB Redis API	.185
4.4.13 Migrating Data from Amazon ElastiCache for Redis to GeminiDB Redis API	. 186
4.4.14 Verifying Redis Data Consistency After Migration	. 189
4.5 Instance Management	. 191
4.5.1 Managing Sessions of a GeminiDB Redis Instance	191
4.5.2 Renaming Commands of a GeminiDB Redis Instance	193
4.5.3 Clearing GeminiDB Redis Instance Data	. 195
4.5.4 Instance Lifecycle Management	197
4.5.4.1 Restarting a GeminiDB Redis Instance	.197
4.5.4.2 Exporting Instance Information	. 199
4.5.4.3 Deleting a Pay-per-Use Instance	. 199
4.5.4.4 Recycling a GeminiDB Redis Instance	200
4.6 Modifying Instance Settings	202
4.6.1 Upgrading a Minor Version	202
4.6.2 Modifying a GeminiDB Redis Instance Name	. 204
4.6.3 Changing the Administrator Password of a GeminiDB Redis Database	. 205
4.6.4 Changing vCPUs and Memory	.206
4.6.5 Setting a Maintenance Window	209
4.6.6 Adding and Deleting Instance Nodes	211
4.6.6.1 Overview	.211
4.6.6.2 Adding Instance Nodes	.212
4.6.6.3 Adding Instance Shards	. 218
4.6.6.4 Deleting Instance Nodes	. 219
4.6.7 Scaling Up and Down Storage Space	.220
4.6.7.1 Overview	. 220
4.6.7.2 Manually Scaling Up Storage Space	. 222
4.6.7.3 Automatically Scaling Up Storage Space	. 226
4.6.7.4 Manually Scaling Down Storage Space	229
4.6.8 Performing a Primary/Standby Switchover for GeminiDB Redis Instances	.232
4.7 Data Backup	. 233
4.7.1 Overview	234
4.7.2 Managing Automated Backups	.235
4.7.3 Managing Manual Backups	.241
4.8 Data Restoration	.244

4.8.1 Restoration Methods.       24         4.8.2 Restoring Data to a New Instance.       22         4.8.3 Restoring Data to a New Instance. Using PITR.       24         4.8.3 Restoring to the Original Instance Using PITR.       24         4.9.1 Big Key Diagnosis.       22         4.9.1 Big Key Diagnosis.       22         4.9.2 Hot Key Diagnosis.       22         4.10.1 Enabling Password-Free Access.       25         4.10.2 ACL Account Management.       25         4.10.3 Enabling Automated Database Redirection for ACL Accounts.       25         4.10.4 Static Porce Attack Defense.       25         4.10.4 Rute Force Attack Defense.       25         4.11.2 Creating a Parameter Template.       26         4.11.3 Viewing Parameter Change History.       27         4.11.4 Exporting a Parameter Template.       27         4.11.5 Comparing Parameter Template.       27         4.11.6 Replication Records of a Parameter Template.       27         4.11.7 Resetting a Parameter Template.       27         4.11.8 Viewing Application Records of a Parameter Template.       27         4.11.9 Viewing Application Records of a Parameter Template.       27         4.11.10 Modifying the Description of a Parameter Template.       27         4.11.10 Modifying the Description of a Paramete		
4.8.2 Restoring Data to a New Instance.       24         4.8.3 Restoring to the Original Instance Using PITR.       22         4.9 Diagnosis Analysis.       24         4.9.1 Big Key Diagnosis.       22         4.9.2 Hot Key Diagnosis.       22         4.9.2 Hot Key Diagnosis.       22         4.0.2 Account and security.       22         4.10.1 Enabling Password-Free Access.       25         4.10.2 Account Management.       25         4.10.3 Enabling Automated Database Redirection for ACL Accounts.       25         4.10.4 Brute Force Attack Defense.       25         4.11.1 Modifying Parameter Sof GeminiDB Redis Instances.       26         4.11.2 Creating a Parameter Template.       26         4.11.3 Viewing Parameter Change History.       27         4.11.4 Kaporting a Parameter Template.       27         4.11.5 Comparing Parameter Template.       27         4.11.6 Replicating a Parameter Template.       27         4.11.7 Resetting a Parameter Template.       27         4.11.8 Applying a Parameter Template.       27         4.11.9 Viewing And Audit.       27         4.11.1 Paleting a Parameter Template.       27         4.11.1 Paleting a Parameter Template.       27         4.11.1 Paleting a Parameter Template.       2	4.8.1 Restoration Methods	
4.8.3 Restoring to the Original Instance Using PITR.       24         4.9 Diagnosis Analysis       24         4.9.1 Big Key Diagnosis.       22         4.9.2 Hok Key Diagnosis.       25         4.0.4 Key Diagnosis.       25         4.10 Account and security.       25         4.10.1 Enabling Password-Free Access.       26         4.10.2 ACL Account Management.       25         4.10.3 Enabling Automated Database Redirection for ACL Accounts.       25         4.10.4 Brute Force Attack Defense.       25         4.11.1 Modifying Parameters of GeminiDB Redis Instances.       25         4.11.2 Creating a Parameter Template.       26         4.11.3 Viewing Parameter Template.       27         4.11.4 Exporting a Parameter Template.       27         4.11.5 Comparing Parameter Template.       27         4.11.6 Replicating a Parameter Template.       27         4.11.7 Resetting a Parameter Template.       27         4.11.9 Viewing Application Records of a Parameter Template.       27         4.11.1 Modifying the Description of a Parameter Template.       27         4.11.1 Modifying the Description of a Parameter Template.       27         4.11.10 Nodifying the Description of a Parameter Template.       27         4.11.10 Nodifying the Description of a Parameter Template.<	4.8.2 Restoring Data to a New Instance	
4.9 Diagnosis Analysis244.9.1 Big Key Diagnosis.224.9.2 Hot Key Diagnosis.224.10 Account and security.254.10.1 Enabling Password-Free Access.254.10.2 ACL Account Management.254.10.3 Enabling Automated Database Redirection for ACL Accounts.254.10.4 Brute Force Attack Defense.264.11.1 Modifying Parameters of GeminiDB Redis Instances.264.11.1 Modifying Parameter Template.264.11.2 Creating a Parameter Template.264.11.3 Viewing Parameter Template.274.11.4 Exporting a Parameter Template.274.11.5 Comparing Parameter Template.274.11.6 Replicating a Parameter Template.274.11.7 Resetting a Parameter Template.274.11.8 Applying a Parameter Template.274.11.9 Viewing Application Records of a Parameter Template.274.11.1 Deleting a Parameter Template.274.11.1 Deleting a Parameter Template.274.12.2 Viewing Audit Logs.264.12.4 Viewing Audit Logs.264.12.4 Viewing Audit Logs.264.12.4 Viewing Metrics and Configuring Alarms.264.13.4 Viewing Metrics.264.13.4 Viewing Metrics.264.13.4 Viewing Metrics.264.13.4 Viewing Metrics.264.13.4 Viewing Metrics.264.14.4 Monitoring.364.13.4 Viewing Event Monitoring Data.364.13.6 Configuring Alarm Rules.374.13.6 Coreitye Metrics. <t< td=""><td>4.8.3 Restoring to the Original Instance Using PITR</td><td></td></t<>	4.8.3 Restoring to the Original Instance Using PITR	
4.9.1 Big Key Diagnosis.244.9.2 Hot Key Diagnosis.254.10 Account and security.254.10.1 Enabling Password-Free Access.254.10.2 ACL Account Management.254.10.3 Enabling Automated Database Redirection for ACL Accounts.254.10.4 Brute Force Attack Defense.254.11.1 Modifying Parameters of Gemini0B Redis Instances.254.11.1 Audifying Parameters of Gemini0B Redis Instances.264.11.2 Creating a Parameter Template.274.11.4 Keporting a Parameter Template.274.11.5 Comparing Parameter Template.274.11.6 Replicating a Parameter Template.274.11.7 Resetting a Parameter Template.274.11.8 Applying a Parameter Template.274.11.9 Viewing Aprilication Records of a Parameter Template.274.11.1 Modifying the Description of a Parameter Template.274.11.1 Modifying to Description of a Parameter Template.274.11.1 Modifying and Exporting Slow Query Logs.264.12.2 Viewing and Exporting Slow Query Logs.264.12.4 Usey Operations Supported by CTS.264.13.4 Viewing Mutit Logs.264.13.4 Viewing Mutrics.274.13.5 Configuring Alarm Rules.374.13.4 Key Operations and Configuring Alarms.264.13.4 Viewing Metrics.274.13.5 Loperations Alarm Rules.374.13.6 Loperations Alarm Rules.374.13.6 Loperations Alarm Rules.364.13.6 Viewing Keynt Monitoring Data.36	4.9 Diagnosis Analysis	
4.9.2 Hot Key Diagnosis.254.10 Account and security.254.10.1 Enabling Password-Free Access.254.10.2 ACL Account Management.254.10.3 Enabling Automated Database Redirection for ACL Accounts.254.10.4 Brute Force Attack Defense.254.11.1 Modifying Parameters of GeminiDB Redis Instances.254.11.2 Creating a Parameter Template.264.11.3 Viewing Parameter Template.274.11.4 Exporting a Parameter Template.274.11.5 Comparing Parameter Template.274.11.6 Replicating a Parameter Template.274.11.7 Resetting a Parameter Template.274.11.8 Applying a Parameter Template.274.11.9 Viewing Application Records of a Parameter Template.274.11.1 Deleting a Parameter Template.274.12.1 Cosg and Audit.274.12.2 Viewing Audit Logs.264.12.4 Viewing Mudit Logs.264.12.4 Viewing Metrics.264.13.3 Viewing Metrics.264.13.4 Viewing Metrics.264.13.4 Viewing Metrics.264.13.4 Viewing Metrics.264.13.4 Viewing Metrics.264.13.5 Configuring Alarm Rules.374.13.4 Viewing Metrics.364.13.4 Viewing Metrics.36 <trr>4.13.4 Viewing Metrics.36<td>4.9.1 Big Key Diagnosis</td><td></td></trr>	4.9.1 Big Key Diagnosis	
4.10 Account and security.254.10.1 Enabling Password-Free Access.254.10.2 ACL Account Management.254.10.3 Enabling Automated Database Redirection for ACL Accounts.254.10.4 Brute Force Attack Defense.254.11 Parameter Management.254.11 Modifying Parameters of GeminiDB Redis Instances.254.11.1 Modifying Parameter Template.264.11.2 Creating a Parameter Template.264.11.3 Viewing Parameter Template.274.11.4 Exporting a Parameter Template.274.11.5 Comparing Parameter Template.274.11.6 Replicating a Parameter Template.274.11.7 Resetting a Parameter Template.274.11.8 Applying a Parameter Template.274.11.9 Viewing Application Records of a Parameter Template.274.11.10 Modifying the Description of a Parameter Template.274.12.1 Enabling or Disabling Log Reporting.274.12.2 Viewing and Exporting Slow Query Logs.264.12.3 Viewing Audit Logs.264.12.4 Viewing Audit Logs.264.12.4 Viewing Metrics.264.13.5 Configuring Alarm Rules.374.13.8 Recommended Alarm Policies.374.13.4 Viewing Metrics.264.13.4 Viewing Metrics.364.13.4 Viewing Metrics.364.13.5 Origuring Alarm Rules.374.13.6 Viewing Metrics.364.13.6 Viewing Event Monitoring Data.364.13.6 Viewing Event Monitoring Data.364.13.6 Viewing Event Mon	4.9.2 Hot Key Diagnosis	250
4.10.1 Enabling Password-Free Access.254.10.2 ACL Account Management.254.10.3 Enabling Automated Database Redirection for ACL Accounts.254.10.4 Brute Force Attack Defense.224.11 Parameter Management.254.11.1 Modifying Parameters of GeminiDB Redis Instances.254.11.2 Creating a Parameter Template.264.11.3 Viewing Parameter Template.274.11.4 Exporting a Parameter Template.274.11.5 Comparing Parameter Template.274.11.6 Replicating a Parameter Template.274.11.7 Resetting a Parameter Template.274.11.8 Applying a Parameter Template.274.11.9 Viewing Application Records of a Parameter Template.274.11.1 Deleting a Parameter Template.274.12 Logs and Audit.274.12 Logs and Audit.274.12.4 Viewing Audit Logs264.12.4 Viewing Audit Logs264.13.4 Key Operations Supported by CTS.264.13.4 Key Operations Supported by CTS.264.13.4 Viewing Metrics.264.13.4 Viewing Metrics.264.13.5 Configuring Alarm R	4.10 Account and security	
4.10.2 ACL Account Management.254.10.3 Enabling Automated Database Redirection for ACL Accounts.264.10.4 Brute Force Attack Defense.254.11 Parameter Cattack Defense.254.11 Parameter Management.254.11.1 Modifying Parameters of GeminiDB Redis Instances.254.11.2 Creating a Parameter Template.264.11.3 Viewing Parameter Change History.274.11.4 Exporting a Parameter Template.274.11.5 Comparing Parameter Template.274.11.6 Replicating a Parameter Template.274.11.8 Applying a Parameter Template.274.11.9 Viewing Application Records of a Parameter Template.274.11.1 Deleting a Parameter Template.274.11.1 Deleting a Parameter Template.274.11.1 Deleting a Parameter Template.274.12.1 Enabling or Disabling Log Reporting.274.12.2 Viewing and Exporting Slow Query Logs.264.12.4 Cr S Audit264.12.4 Cr S Audit Logs.264.13.4 Viewing Metrics and Configuring Alarms.264.13.4 Viewing Metrics.264.13.4 Viewing Metrics.264.13.4 Viewing Metrics.264.13.5 Configuring Alarm Rules.374.13.6 Configuring Alarm Rules.374.13.6 Configuring Dashboard.364.13.6 Viewing Event Monitoring Data.364.13.6 Viewing Event Monitoring Data.364.13.6 Origo Figure Events.364.13.6 Admitored Events.364.13.6 Admitored Events.	4.10.1 Enabling Password-Free Access	
4.10.3 Enabling Automated Database Redirection for ACL Accounts.       25         4.10.4 Brute Force Attack Defense.       25         4.11 Parameter Management.       25         4.11.1 Modifying Parameters of GeminiDB Redis Instances.       25         4.11.2 Creating a Parameter Template.       26         4.11.3 Viewing Parameter Change History.       27         4.11.4 Exporting a Parameter Template.       27         4.11.5 Comparing Parameter Template.       27         4.11.6 Replicating a Parameter Template.       27         4.11.7 Resetting a Parameter Template.       27         4.11.8 Applying a Parameter Template.       27         4.11.9 Viewing Application Records of a Parameter Template.       27         4.11.1 Deleting a Parameter Template.       27         4.11.1 Deleting a Parameter Template.       27         4.11.2 Viewing and Exporting Slow Query Logs.       26         4.12.4 Viewing Mult Logs.       26         4.12.4 Sueving Matrix Supported by CTS.       26         4.13.4 Supported Metrics.       26         4.13.4 Suported Metrics.       26         4.12.4 Tis Audit       27         4.11.10 Modifying the Description of a Parameter Template.       27         4.12.4 Superiting Auges.       26         4.12.4 Liss Au	4.10.2 ACL Account Management	
4.10.4 Brute Force Attack Defense254.11 Parameter Management224.11.1 Modifying Parameters of GeminiDB Redis Instances254.11.2 Creating a Parameter Template264.11.3 Viewing Parameter Change History274.11.4 Exporting a Parameter Template274.11.5 Comparing Parameter Template274.11.6 Replicating a Parameter Template274.11.7 Resetting a Parameter Template274.11.8 Applying a Parameter Template274.11.9 Viewing Application Records of a Parameter Template274.11.1 Deleting a Parameter Template274.11.2 Logs and Audit274.12.1 Enabling or Disabling Log Reporting274.12.2 Viewing and Exporting Slow Query Logs264.12.4 CTS Audit.264.12.4 CTS Audit.264.12.4 CTS Audit.264.12.4 CTS Audit.264.12.4 CTS Audit.264.13.4 Key Operations Supported by CTS264.13.4 Viewing Metrics and Configuring Alarms264.13.5 Upported Metrics.274.13.6 Event Monitoring364.13.6 Event Monitoring364.13.6 Event Monitoring364.13.6 Atomic Metrics.374.13.6 Atomic Metrics.364.13.6 Atomic Metrics.364.13.6 Atomic Metrics.364.14 Atomic Metrics.364.13.6 Loweriew364.13.6 Loweriew364.13.6 Loweriew364.13.6 Loweriew364.13.6 Loweri	4.10.3 Enabling Automated Database Redirection for ACL Accounts	
4.11 Parameter Management.254.11.1 Modifying Parameters of GeminiDB Redis Instances.254.11.2 Creating a Parameter Template.264.11.3 Viewing Parameter Template.274.11.4 Exporting a Parameter Template.274.11.5 Comparing Parameter Template.274.11.6 Replicating a Parameter Template.274.11.7 Resetting a Parameter Template.274.11.8 Applying a Parameter Template.274.11.9 Viewing Application Records of a Parameter Template.274.11.10 Modifying the Description of a Parameter Template.274.11.11 Deleting a Parameter Template.274.12.1 Cigs and Audit.274.12.2 Viewing and Exporting Slow Query Logs.264.12.4 CTS Audit.264.12.4 CTS Audit.264.13.1 Supported Metrics.264.13.1 Supported Metrics.264.13.2 Configuring Alarm Rules.274.13.4 Newing Metrics.264.13.4 Viewing Metrics.264.13.4 Viewing Metrics.264.13.4 Viewing Metrics.264.13.4 Viewing Metrics.264.13.5 Configuring Alarm Rules.374.13.6 Event Monitoring.364.13.6 Event Monitoring Data.364.13.6 A Monitoring Data.364.13.6 A Monitored Events.364.13.6 A Monitored Events.36 <td>4.10.4 Brute Force Attack Defense</td> <td>258</td>	4.10.4 Brute Force Attack Defense	258
4.11.1 Modifying Parameters of GeminiDB Redis Instances       25         4.11.2 Creating a Parameter Template       26         4.11.3 Viewing Parameter Template       27         4.11.4 Exporting a Parameter Templates       27         4.11.5 Comparing Parameter Templates       27         4.11.6 Replicating a Parameter Template       27         4.11.7 Resetting a Parameter Template       27         4.11.8 Applying a Parameter Template       27         4.11.9 Viewing Application Records of a Parameter Template       27         4.11.1 Deleting a Parameter Template       27         4.11.1 Deleting a Parameter Template       27         4.11.1 Deleting a Parameter Template       27         4.12.1 Viewing Audit       27         4.12.1 Viewing and Exporting Slow Query Logs       28         4.12.4 Viewing Matil Logs       26         4.12.4 Viewing Matil Logs       26         4.12.4 Viewing Matil Logs       26         4.13.4 Viewing Matrics and Configuring Alarms       26         4.13.4 Viewing Matrics       26         4.13.4 Viewing Matrics       26         4.13.4 Viewing Matrics       26         4.12.4 (TS Audit       26         4.12.4 Viewing Matrics       26         4.13.4 Viewing Matrics	4.11 Parameter Management	
4.11.2 Creating a Parameter Template.264.11.3 Viewing Parameter Change History.274.11.4 Exporting a Parameter Template.274.11.5 Comparing Parameter Templates.274.11.6 Replicating a Parameter Template.274.11.7 Resetting a Parameter Template.274.11.8 Applying a Parameter Template.274.11.9 Viewing Application Records of a Parameter Template.274.11.10 Modifying the Description of a Parameter Template.274.11.11 Deleting a Parameter Template.274.12.12 Logs and Audit.274.12.2 Viewing and Exporting Slow Query Logs.264.12.3 Viewing Audit Logs.264.12.4 CTS Audit.264.12.4 CTS Audit.264.13.5 Supported by CTS.264.13.4 Supported Metrics.274.13.3 Recommended Alarm Policies.374.13.4 Viewing Metrics.374.13.5 Configuring Alarm Rules.374.13.6 Event Monitoring.364.13.6 Event Monitoring.364.13.6 Viewing Metrics.364.13.6 Viewing Metrics.3	4.11.1 Modifying Parameters of GeminiDB Redis Instances	
4.11.3 Viewing Parameter Change History.274.11.4 Exporting a Parameter Template.274.11.5 Comparing Parameter Templates.274.11.6 Replicating a Parameter Template.274.11.7 Resetting a Parameter Template.274.11.8 Applying a Parameter Template.274.11.9 Viewing Application Records of a Parameter Template.274.11.10 Modifying the Description of a Parameter Template.274.11.11 Deleting a Parameter Template.274.12 Logs and Audit.274.12.1 Enabling or Disabling Log Reporting.274.12.2 Viewing and Exporting Slow Query Logs.284.12.4 CTS Audit.264.12.4 CTS Audit.264.13.4 Supported by CTS.264.13.4 Supported Metrics.274.13.3 Recommended Alarm Policies.374.13.4 Viewing Metrics.374.13.5 Configuring Alarm Rules.374.13.6 Event Monitoring.364.13.6 Zviewing and Larm Rule for Events.364.13.6 A Monitored Events.364.13.6 A Monitored Events.364.13.6 A Monitored Events.364.13.6 A Monitoring Data.364.13.6 A Monitored Events.364.13.6 A Monitored Events.36 <tr <td="">4.13.6 A Monitored E</tr>	4.11.2 Creating a Parameter Template	
4.11.4 Exporting a Parameter Template	4.11.3 Viewing Parameter Change History	
4.11.5 Comparing Parameter Templates.274.11.6 Replicating a Parameter Template.274.11.7 Resetting a Parameter Template.274.11.8 Applying a Parameter Template.274.11.9 Viewing Application Records of a Parameter Template.274.11.10 Modifying the Description of a Parameter Template.274.11.10 Logs and Audit.274.12 Logs and Audit.274.12.1 Enabling or Disabling Log Reporting.274.12.2 Viewing and Exporting Slow Query Logs.264.12.4 CTS Audit.264.12.4 CTS Audit.264.12.4 CTS Audit.264.13.4 Viewing Metrics.264.13.1 Supported Metrics.264.13.2 Configuring Alarm Rules.374.13.3 Recommended Alarm Policies.374.13.4 Viewing Metrics.364.13.6 Viewing a Dashboard.364.13.6 Viewing an Alarm Rule for Events.364.13.6 Viewing an Alarm Rule for Events.364.13.6 Monitoried Events.364.13.6 Monitoried Events.364.13.6 Monitoried Events.364.13.6 Monitoried Events.364.14 Tag Management.40	4.11.4 Exporting a Parameter Template	
4.11.6 Replicating a Parameter Template.274.11.7 Resetting a Parameter Template.274.11.8 Applying a Parameter Template.274.11.9 Viewing Application Records of a Parameter Template.274.11.10 Modifying the Description of a Parameter Template.274.11.10 Logs and Audit.274.12 Logs and Audit.274.12.1 Enabling or Disabling Log Reporting.274.12.2 Viewing and Exporting Slow Query Logs.264.12.4 CTS Audit.264.12.4 CTS Audit.264.12.4.1 Key Operations Supported by CTS.264.13.1 Supported Metrics.264.13.2 Configuring Alarm Rules.374.13.3 Recommended Alarm Policies.374.13.4 Viewing Metrics374.13.6 Event Monitoring.384.13.6.1 Overview384.13.6.2 Viewing Event Monitoring Data.384.13.6.4 Monitored Events.384.13.6 A Monitored Events.384.14 Tag Management.40	4.11.5 Comparing Parameter Templates	
4.11.7 Resetting a Parameter Template.274.11.8 Applying a Parameter Template.274.11.9 Viewing Application Records of a Parameter Template.274.11.10 Modifying the Description of a Parameter Template.274.11.11 Deleting a Parameter Template.274.12 Logs and Audit.274.12.1 Enabling or Disabling Log Reporting.274.12.2 Viewing and Exporting Slow Query Logs.264.12.4 CTS Audit.264.12.4 CTS Audit.264.12.4 CTS Audit.264.12.4 Querying Traces.264.13.1 Supported Metrics.264.13.2 Configuring Alarm Rules.374.13.3 Recommended Alarm Policies.374.13.6 Event Monitoring.364.13.6.1 Overview364.13.6.2 Viewing Event Monitoring Data.364.13.6.4 Monitored Events.364.14 Tag Management.36	4.11.6 Replicating a Parameter Template	
4.11.8 Applying a Parameter Template.274.11.9 Viewing Application Records of a Parameter Template.274.11.0 Modifying the Description of a Parameter Template.274.11.1 Deleting a Parameter Template.274.12 Logs and Audit.274.12.1 Enabling or Disabling Log Reporting.274.12.2 Viewing and Exporting Slow Query Logs.264.12.4 CTS Audit.264.12.4 CTS Audit.264.12.4 CTS Audit.264.12.4 Querying Traces.264.13.1 Supported Metrics.264.13.2 Configuring Alarm Rules.374.13.3 Recommended Alarm Policies.374.13.4 Viewing Metrics.364.13.6 Event Monitoring.384.13.6.1 Overview384.13.6.2 Viewing Event Monitoring Data.364.13.6.4 Monitored Events.364.14 Tag Management.40	4.11.7 Resetting a Parameter Template	
4.11.9 Viewing Application Records of a Parameter Template.274.11.10 Modifying the Description of a Parameter Template.274.11.11 Deleting a Parameter Template.274.12 Logs and Audit.274.12.1 Enabling or Disabling Log Reporting.274.12.2 Viewing and Exporting Slow Query Logs.264.12.3 Viewing Audit Logs.264.12.4 CTS Audit.264.12.4 CTS Audit.264.12.4.4 CTS Audit.264.12.4.2 Querying Traces.264.13.1 Supported by CTS.264.13.1 Supported Metrics.264.13.2 Configuring Alarm Rules.374.13.3 Recommended Alarm Policies.374.13.6 Event Monitoring.364.13.6 Event Monitoring.364.13.6.1 Overview364.13.6.2 Viewing Event Monitoring Data.364.13.6.4 Monitored Events.364.14 Tag Management.40	4.11.8 Applying a Parameter Template	
4.11.10 Modifying the Description of a Parameter Template.274.11.11 Deleting a Parameter Template.274.12 Logs and Audit.274.12 Logs and Audit.274.12.1 Enabling or Disabling Log Reporting.274.12.2 Viewing and Exporting Slow Query Logs.264.12.3 Viewing Audit Logs.264.12.4 CTS Audit.264.12.4 CTS Audit.264.12.4.4 CTS Audit.264.12.4.2 Querying Traces.264.13.4 Viewing Metrics and Configuring Alarms.264.13.1 Supported Metrics.264.13.2 Configuring Alarm Rules.374.13.3 Recommended Alarm Policies.374.13.4 Viewing Metrics.374.13.5 Configuring a Dashboard.364.13.6 Event Monitoring.364.13.6.1 Overview364.13.6.2 Viewing Event Monitoring Data.364.13.6.4 Monitored Events.364.13.6.4 Monitored Events.364.14 Tag Management.40	4.11.9 Viewing Application Records of a Parameter Template	
4.11.11 Deleting a Parameter Template.274.12 Logs and Audit.274.12.1 Enabling or Disabling Log Reporting.274.12.2 Viewing and Exporting Slow Query Logs.264.12.3 Viewing Audit Logs.264.12.4 CTS Audit.264.12.4 CTS Audit.264.12.4 CTS Audit.264.12.4 Querying Traces.264.12.4 Querying Traces.264.13 Viewing Metrics and Configuring Alarms.264.13.1 Supported Metrics.264.13.2 Configuring Alarm Rules.374.13.3 Recommended Alarm Policies.374.13.6 Event Monitoring.364.13.6.1 Overview364.13.6.2 Viewing Event Monitoring Data.364.13.6.3 Creating an Alarm Rule for Events.364.13.6.4 Monitored Events.364.14 Tag Management.40	4.11.10 Modifying the Description of a Parameter Template	
4.12 Logs and Audit274.12.1 Enabling or Disabling Log Reporting.274.12.2 Viewing and Exporting Slow Query Logs.284.12.3 Viewing Audit Logs.284.12.4 CTS Audit.284.12.4 CTS Audit.284.12.4.1 Key Operations Supported by CTS.264.12.4.2 Querying Traces.264.13 Viewing Metrics and Configuring Alarms.264.13.1 Supported Metrics.264.13.2 Configuring Alarm Rules.374.13.3 Recommended Alarm Policies.374.13.4 Viewing Metrics.374.13.5 Configuring a Dashboard.364.13.6 Event Monitoring.384.13.6.1 Overview364.13.6.2 Viewing Event Monitoring Data.364.13.6.4 Monitored Events.364.13.6.4 Monitored Events.364.14 Tag Management.40	4.11.11 Deleting a Parameter Template	
4.12.1 Enabling or Disabling Log Reporting.274.12.2 Viewing and Exporting Slow Query Logs.284.12.3 Viewing Audit Logs.284.12.4 CTS Audit.284.12.4 CTS Audit.284.12.4 Let X (Key Operations Supported by CTS.284.12.4.1 Key Operations Supported by CTS.284.12.4.2 Querying Traces.284.13 Viewing Metrics and Configuring Alarms.284.13.1 Supported Metrics.264.13.2 Configuring Alarm Rules.374.13.3 Recommended Alarm Policies.374.13.4 Viewing Metrics.374.13.5 Configuring a Dashboard.384.13.6 Event Monitoring.384.13.6.1 Overview384.13.6.2 Viewing Event Monitoring Data.364.13.6.4 Monitored Events.364.13.6.4 Monitored Events.364.14 Tag Management.40	4.12 Logs and Audit	
4.12.2 Viewing and Exporting Slow Query Logs.284.12.3 Viewing Audit Logs.284.12.4 CTS Audit.284.12.4 CTS Audit.284.12.4.1 Key Operations Supported by CTS.284.12.4.2 Querying Traces.284.13 Viewing Metrics and Configuring Alarms.284.13.1 Supported Metrics.264.13.2 Configuring Alarm Rules.264.13.3 Recommended Alarm Policies.374.13.4 Viewing Metrics.374.13.5 Configuring a Dashboard.384.13.6 Event Monitoring.384.13.6.1 Overview364.13.6.2 Viewing Event Monitoring Data.364.13.6.4 Monitored Events.384.13.6.4 Monitored Events.384.14 Tag Management.40	4.12.1 Enabling or Disabling Log Reporting	
4.12.3 Viewing Audit Logs	4.12.2 Viewing and Exporting Slow Query Logs	
4.12.4 CTS Audit.264.12.4.1 Key Operations Supported by CTS.264.12.4.2 Querying Traces.264.13 Viewing Metrics and Configuring Alarms.264.13.1 Supported Metrics.264.13.2 Configuring Alarm Rules.374.13.3 Recommended Alarm Policies.374.13.4 Viewing Metrics.374.13.5 Configuring a Dashboard.364.13.6 Event Monitoring.364.13.6.1 Overview364.13.6.2 Viewing Event Monitoring Data.364.13.6.4 Monitored Events.364.13.6.4 Monitored Events.364.14 Tag Management.40	4.12.3 Viewing Audit Logs	
4.12.4.1 Key Operations Supported by CTS.264.12.4.2 Querying Traces.264.13 Viewing Metrics and Configuring Alarms.264.13.1 Supported Metrics.264.13.2 Configuring Alarm Rules.274.13.3 Recommended Alarm Policies.374.13.4 Viewing Metrics.374.13.5 Configuring a Dashboard.364.13.6 Event Monitoring.364.13.6.1 Overview364.13.6.2 Viewing Event Monitoring Data.364.13.6.4 Monitored Events.364.13.6.4 Monitored Events.364.14 Tag Management.40	4.12.4 CTS Audit	
4.12.4.2 Querying Traces.264.13 Viewing Metrics and Configuring Alarms.264.13.1 Supported Metrics.264.13.2 Configuring Alarm Rules.374.13.3 Recommended Alarm Policies.374.13.4 Viewing Metrics.374.13.5 Configuring a Dashboard.384.13.6 Event Monitoring.364.13.6.1 Overview364.13.6.2 Viewing Event Monitoring Data.364.13.6.4 Monitored Events.364.13.6.4 Monitored Events.364.14 Tag Management.40	4.12.4.1 Key Operations Supported by CTS	
4.13 Viewing Metrics and Configuring Alarms.284.13.1 Supported Metrics.284.13.2 Configuring Alarm Rules.374.13.3 Recommended Alarm Policies.374.13.4 Viewing Metrics.374.13.5 Configuring a Dashboard.384.13.6 Event Monitoring.384.13.6.1 Overview384.13.6.2 Viewing Event Monitoring Data.384.13.6.3 Creating an Alarm Rule for Events.384.13.6.4 Monitored Events.384.14 Tag Management.40	4.12.4.2 Querying Traces	
4.13.1 Supported Metrics.284.13.2 Configuring Alarm Rules.374.13.3 Recommended Alarm Policies.374.13.4 Viewing Metrics.374.13.5 Configuring a Dashboard.384.13.6 Event Monitoring.384.13.6.1 Overview384.13.6.2 Viewing Event Monitoring Data.384.13.6.3 Creating an Alarm Rule for Events.384.13.6.4 Monitored Events.384.14 Tag Management.40	4.13 Viewing Metrics and Configuring Alarms	
4.13.2 Configuring Alarm Rules.374.13.3 Recommended Alarm Policies.374.13.4 Viewing Metrics.374.13.5 Configuring a Dashboard.384.13.6 Event Monitoring.384.13.6.1 Overview384.13.6.2 Viewing Event Monitoring Data.384.13.6.3 Creating an Alarm Rule for Events.384.13.6.4 Monitored Events.384.14 Tag Management.40	4.13.1 Supported Metrics	
4.13.3 Recommended Alarm Policies.374.13.4 Viewing Metrics.374.13.5 Configuring a Dashboard.384.13.6 Event Monitoring.384.13.6.1 Overview384.13.6.2 Viewing Event Monitoring Data.384.13.6.3 Creating an Alarm Rule for Events.384.13.6.4 Monitored Events.384.14 Tag Management.40	4.13.2 Configuring Alarm Rules	
4.13.4 Viewing Metrics.374.13.5 Configuring a Dashboard.384.13.6 Event Monitoring.384.13.6.1 Overview384.13.6.2 Viewing Event Monitoring Data.384.13.6.3 Creating an Alarm Rule for Events.384.13.6.4 Monitored Events.384.14 Tag Management.40	4.13.3 Recommended Alarm Policies	
4.13.5 Configuring a Dashboard.384.13.6 Event Monitoring.384.13.6.1 Overview384.13.6.2 Viewing Event Monitoring Data.384.13.6.3 Creating an Alarm Rule for Events.384.13.6.4 Monitored Events.384.14 Tag Management.40	4.13.4 Viewing Metrics	
4.13.6 Event Monitoring.384.13.6.1 Overview384.13.6.2 Viewing Event Monitoring Data.384.13.6.3 Creating an Alarm Rule for Events.384.13.6.4 Monitored Events.384.14 Tag Management.40	4.13.5 Configuring a Dashboard	
4.13.6.1 Overview384.13.6.2 Viewing Event Monitoring Data.384.13.6.3 Creating an Alarm Rule for Events.384.13.6.4 Monitored Events.384.14 Tag Management.40	4.13.6 Event Monitoring	
4.13.6.2 Viewing Event Monitoring Data	4.13.6.1 Overview	
4.13.6.3 Creating an Alarm Rule for Events.       38         4.13.6.4 Monitored Events.       38         4.14 Tag Management.       40	4.13.6.2 Viewing Event Monitoring Data	
4.13.6.4 Monitored Events	4.13.6.3 Creating an Alarm Rule for Events	
4.14 Tag Management	4.13.6.4 Monitored Events	
	4.14 Tag Management	

4.15 Managing User Resource Quotas of a GeminiDB Redis Instance	404
4.16 Memory Acceleration	405
4.16.1 RDS Memory Acceleration	405
4.16.1.1 Memory Acceleration Overview	405
4.16.1.2 Enabling and Using Memory Acceleration	406
4.16.1.3 Modifying and Deleting a Memory Acceleration Rule	414
4.16.1.4 Viewing and Removing Mappings	415
4.16.2 TaurusDB Memory Acceleration	417
4.16.2.1 Overview	417
4.16.2.2 Enabling and Using Memory Acceleration	417
4.16.2.3 Modifying and Deleting a Memory Acceleration Rule	425
4.16.2.4 Viewing and Removing Mappings	426
5 Development Reference	429
5.1 Development and O&M Rules	429
5.2 Compatible Commands	437
5.3 Examples of Connecting to an Instance Using Programming Languages	442
5.3.1 Connecting to an Instance Using Jedis	442
5.3.2 Connecting to an Instance Using Redisson	445
5.3.3 Connecting to an Instance Using Hiredis	448
5.3.4 Connecting to an Instance Using Node.js	450
5.3.5 Connecting to an Instance Using PHP	452
5.3.6 Connecting to an Instance Using Python	455
5.3.7 Connecting to an Instance Using Go	456
5.3.8 Connecting to an Instance Using C#	458
5.3.9 Connecting to an Instance Using Sentinel	460
5.4 Lua Script Compilation Specifications	462
5.5 Compact Encoding for Small Objects	465
5.6 Keyspace Notification	466
5.7 ExHash Commands	468
5.8 Large Bitmap Initialization	475
5.9 Querying Large Bitmaps by Page	478
5.10 Configuring Parameters for a Client Connection Pool	482
5.11 Using Parallel SCAN to Accelerate Full Database Scanning	484
5.12 Accessing a GeminiDB Redis Instance Using a Pipeline	486
5.13 Processing Transactions on a GeminiDB Redis Instance	488
5.14 Retry Mechanism for Clients Accessing GeminiDB Redis API	489
5.15 GeminiDB Redis API Pub/Sub	494
6 Best Practices	498
6.1 Automated Database Access Using an Account for Multitenancy Management of GeminiDB Rec	lis
Instances	498
6.2 FastLoad for RTA-based Ad Placement	500
6.3 PITR for Restoring Gaming Data	502

6.4 ExHash for Ad Frequency Control	.503
6.5 GeminiDB Redis API for Instant Messaging	509
6.6 Implementing Distributed Locks Using Lua Scripts for GeminiDB Redis API	. 516
6.7 Suggestions on Alarm Rules of GeminiDB Redis Instance Metrics	. 519
6.8 GeminiDB Redis API for Product Correlation Analysis	.522
6.9 GeminiDB Redis API for Online Classroom	.525
6.10 GeminiDB Redis API for Session Management in Web Applications	. 528
7 Performance White Paper	531
7.1 General Performance Data	.531
7.1.1 Performance Test Methods	. 531
7.1.2 Performance Test Results	535
7.2 Performance Data in RTA Scenarios	536
7.2.1 Performance Test Methods	. 537
7.2.2 Performance Test Results	540
8 FAOs	542
8.1 About GeminiDB Redis API	542
8.1.1 What Are the Differences Between GeminiDB Redis API, Open-Source Redis, and Other Open-So Redis Cloud Services?	urce 542
8.1.2 How Is the Performance of GeminiDB Redis API Compared with Open-Source Redis?	. 543
8.1.3 What Redis Versions and Commands Are Compatible with GeminiDB Redis API? Whether Application Code Needs to Be Refactored for Connecting to a Redis Client?	.543
8.1.4 Can Data Be Migrated from Open-Source Redis to GeminiDB Redis API? What Are the Precaution	ns? 543
8.1.5 What Is the Availability of a GeminiDB Redis Instance?	. 544
8.1.6 Are Total Memory and Total Capacity of a GeminiDB Redis Instance the Same? What Is the Relationship Between Memory and Capacity?	.544
8.1.7 How Do I Select Proper Node Specifications and Node Quantity When Purchasing a GeminiDB R Instance?	edis . 544
8.1.8 Is a Primary/Standby or Cluster Deployment Mode Preferred for GeminiDB Redis Instances with Several GB of Storage Space?	544
8.1.9 How Does GeminiDB Redis API Persist Data? Will Data Be Lost?	.545
8.1.10 What Is the Memory Eviction Policy of GeminiDB Redis API?	.545
8.1.11 Does GeminiDB Redis API Support Modules Such as a Bloom Filter?	. 545
8.2 Billing	. 545
8.2.1 What Are the Differences Between Yearly/Monthly and Pay-per-Use Billing Modes?	.546
8.2.2 Can I Switch Between Yearly/Monthly and Pay-per-Use Payments?	. 546
8.3 Database Usage	546
8.3.1 Why Is the Key Not Returned Using Scan Match?	. 546
8.3.2 How Do I Process Existing Data Shards After Migrating Workloads to GeminiDB Redis API?	.547
8.3.3 Does GeminiDB Redis API Support Fuzzy Queries Using KEYS?	.547
8.3.4 Does the GeminiDB Redis API Support Multiple Databases?	.547
8.3.5 Why the Values Returned by Scan Operations Are Different Between GeminiDB Redis API and Open-Source Redis 5.0?	. 547

8.3.6 Why Is the Cursor Length Returned by SCAN of GeminiDB Redis API Is Longer Than That of Open Source Redis?	- 548
8.3.7 Why Are Error Messages Returned by Some Invalid Commands Different Between GeminiDB Redi API and Open-Source Redis 5.0?	s 548
8.3.8 How Do I Resolve the Error "CROSSSLOT Keys in request don't hash to the same slot"?5	548
8.3.9 How Many Commands Can Be Contained in a GeminiDB Redis Transaction?5	549
8.3.10 Which Commands Require Hashtags on GeminiDB Redis Cluster Instances?	549
8.3.11 What Should I Do If "ERR unknown command sentinel" Is Displayed?5	551
8.3.12 Why Return Values of Blocking Commands Differ Between Primary/Standby GeminiDB Redis Instances and Open-Source Redis Instances?	551
8.3.13 How Long Does It Take to Scale Up GeminiDB Redis Instance Storage? Will Services Be Affected?	? 552
8.3.14 How Long Does It Take to Add GeminiDB Redis Nodes at the Same Time? What Are the Impacts on Services?	; 552
8.3.15 What Are the Differences Between Online and Offline Specification Changes of GeminiDB Redis Nodes? How Long Will the Changes Take? What Are the Impacts on Services?	552
8.3.16 What Are the Differences Between Online and Offline Patch Installation of GeminiDB Redis Nodes? How Long Will the Upgrades Take? What Are the Impacts on Services?	553
8.3.17 Can I Download Backups of a GeminiDB Redis Instance to a Local PC and Restore Data Offline?	553
8.3.18 What Is the Data Backup Mechanism of GeminiDB Redis API? What Are the Impacts on Services	? 553
8.3.19 Why Does the CPU Usage Remain High Despite Low Service Access Volume on a GeminiDB Redi Preferential Instance with 1 CPU and 2 Nodes?	s 554
8.3.20 Why Does the Number of Keys Decrease and Then Become Normal on the Monitoring Panel on the GUI of GeminiDB Redis API?	554
8.3.21 Why Is CPU Usage of GeminiDB Redis Instance Nodes Occasionally High?5	554
8.3.22 How Do I Upgrade GeminiDB Redis API from 5.0 to 6.2?5	554
8.3.23 When Does a GeminiDB Redis Instance Become Read-Only?5	554
8.3.24 How Is the Number of Keys Collected? Why Is the Number of Keys Monitored by GeminiDB Inaccurately Increasing After Migration?	555
8.4 Database Connection	555
8.4.1 How Do I Connect to a GeminiDB Redis Instance?	556
8.4.2 How Do I Use Multiple Node IP Addresses Provided by GeminiDB Redis API?	556
8.4.3 How Does Load Balancing Work in GeminiDB Redis API?5	556
8.4.4 How Can I Create and Connect to an ECS?5	556
8.4.5 Can I Change the VPC of a GeminiDB Redis Instance?5	557
8.4.6 Why Can't I Connect to the Instance After an EIP Is Bound to It?	557
8.4.7 How Do I Access a GeminiDB Redis Instance from a Private Network?	557
8.4.8 Do I Need to Enable Private Network Access Control for a Load Balancer After Setting a Security Group?	558
8.4.9 What Should I Do If the Client Connection Pool Reports Error " Could not get a resource from the pool"?5	558
8.4.10 Common Client Errors and Troubleshooting Methods5	559
8.5 Backup and Restoration	560
8.5.1 How Long Can a GeminiDB Redis Instance Backup Be Saved?5	560

8.6 Regions and AZs	560
8.6.1 Can Different AZs Communicate with Each Other?	560
8.6.2 Can I Change the Region of a GeminiDB Redis Instance?	560
8.7 Data Migration	561
8.7.1 What Do I Do if the Error "ERR the worker queue is full, and the request cannot be executed" Is Displayed?	561
8.7.2 What Do I Do If the Error "ERR the request queue of io thread is full, and the request cannot be executed" Is Displayed?	561
8.7.3 What Do I DO If the Error "read error, please check source redis log or network" Is Displayed?	.561
8.7.4 What Do I Do If the Error "slaveping_thread.cc-ThreadMain-90: error: Ping master error" Is Displayed?	561
8.7.5 What Do I Do If the Forward Migration Speed of the Synchronization Status Is Too Slow?	561
8.7.6 What Do i Do When the Forward Migration Speed of the Synchronization Status Is Too Fast, and the Error Message "ERR Server Reply Timeout, Some Responses May Lose, but Requests Have Been Executed" Is Displayed?	562
8.7.7 Can Data Be Migrated from Self-Built Redis 4.0, 5.0, and 6.2 to GeminiDB Redis API?	562
8.7.8 How Do I Migrate Data from Self-Built Primary/Standby and Cluster Redis Instances to GeminiDI Redis Instances?	B 562
8.7.9 Why Cannot DRS Migrate Data from Third-Party Redis Such as ApsaraDB for Redis and TencentD for Redis?	)В 562
8.7.10 Which of the Following Factors Need to Be Considered When Data Is Migrated from Self-Built Primary/Standby Redis Instances to a GeminiDB Redis cluster?	.563
8.7.11 Only 20% to 30% of 100 GB of Data Was Migrated to GeminiDB Redis. Is the Migration Incomplete?	563
8.8 Memory Acceleration	563
8.8.1 Will All Data Be Cached to GeminiDB Redis Instances After Memory Acceleration Is Enabled and MySQL Database Data Is Updated?	563
8.8.2 If Memory Acceleration Is Enabled, GeminiDB Redis Instance Data Increases Continuously. Do I Need to Scale Out the Capacity? How Do I Manage Cached Data?	564
8.8.3 Is Memory Acceleration Recommended When Customers' Service Data Can Be Synchronized Between MySQL and Redis? In Which Scenarios Can Memory Acceleration Be enabled?	.564
8.8.4 How Long Is the Latency of Synchronization from RDS for MySQL to GeminiDB Redis API? What Factors Affect the Latency?	564
8.8.5 Will the Source MySQL Database Be Affected After Memory Acceleration Is Enabled?	564
8.8.6 GeminiDB Redis Instances with Memory Acceleration Enabled Needs to Process a Large Number Binlogs in a Short Period of Time. Will a Large Number of Resources Be Occupied and Online Services Affected?	of Be 564
8.9 Freezing, Releasing, Deleting, and Unsubscribing from Instances	565

## **1** Service Overview

- 1.1 Enterprise-Level Features
- 1.2 Highlights
- **1.3 Product Architecture**
- **1.4 Application Scenarios**

As a key-value database compatible with Redis APIs, GeminiDB Redis API extends application scenarios of Redis so that it can better meet diversified service requirements such as persistent and hybrid storage.

- 1.5 Compatible API and Versions
- **1.6 Instance Specifications**
- 1.7 Instance Statuses
- 1.8 Constraints

## **1.1 Enterprise-Level Features**

GeminiDB Redis API uses a cloud-native distributed architecture designed to decouple storage resources from compute resources and is fully compatible with Redis 7.0, 6.2 (including 6.2.X), 5.0 and earlier editions, so it provides more enterprise-grade features.

- Exclusive resources, no traffic limiting for shards
  - Compute nodes are deployed in exclusive containers. Tenants are isolated from each other, ensuring high stability. In the case of high concurrent traffic, traffic on nodes is unlimited.
  - Built-in exclusive load balancers provide higher forwarding performance and stability.
  - Public IP addresses can be bound to compute nodes, facilitating cloud migration and remote debugging.
- Second-level auto scaling, easily coping with service peaks and valleys
  - Independent scaling is supported for storage and compute resources. A single instance supports tens of millions of QPS and dozens of TB of storage space.

- In scenarios where data volume increases, capacity expansions can be performed with a few clicks, without affecting service applications.
- In scenarios where workloads increase suddenly (for example, gaming and e-commerce activities have higher QPS requirements), you can expand the capacity by adding nodes or increasing specifications. In the future, you can easily reduce the capacity. Only second-level reconnection may occur to services.
- A unified database used to simplify the service architecture
  - With high-performance storage pools, instances automatically load frequently accessed hot data to the memory of compute nodes and exchange cold and hot data internally. Services preferentially read hot data from the memory, ensuring high data reliability and low latency.
  - GeminiDB Redis API is suitable for storing ever-increasing important service data (such as game player data, user profiles, behavior logs, and article information). Compared with the Redis+MySQL architecture, it has a simpler architecture, more reliable data storage, and higher comprehensive performance and cost-effectiveness.
- 3-AZ deployment
  - 3-AZ instances allow compute and storage resources to be evenly distributed across the AZs. The deployment rules strictly comply with anti-affinity groups, delivering ultra-high reliability.
  - If a node is faulty, services can be taken over in seconds. With the dedicated, decoupled storage and compute, GeminiDB Redis API provides fault tolerance (*N*-1 reliability) to allow you to restore service access in seconds, achieving ultra-high availability.
- Account management for database-level permission control
  - A maximum of 65,536 databases can be used, and up to 200 subaccounts can be created.
  - You cannot only set read-only or read/write permissions for sub-accounts, but also configure accessible databases for sub-accounts, preventing misoperations between tenants.
- Setting an expiration time for each field of a hash key
  - Redis only supports an overall expiration time for hash keys. GeminiDB Redis API adds a group of hash commands. This allows you to set an expiration time for a specified field in a hash key and implement the elimination logic at the service layer in the database, simplifying the service architecture.
  - For details about the best practices of ExHash, see 6.4 ExHash for Ad Frequency Control.
- Strong data consistency, preventing dirty reads
  - Open-source Redis adopts asynchronous replication, and data copies are weakly consistent. In common service scenarios where counters, rate limiters, and distributed locks are used, dirty reads may occur, which may cause service logic disorder.
  - GeminiDB Redis API connects data copies to a high-performance storage pool. Once data is successfully written into the storage pool, the three copies of data are stored consistently, preventing dirty reads from occurring in subsequent service access.

#### • Enhanced transactions

 The transaction MULTI/EXEC is supported. Compared with the opensource Redis, GeminiDB Redis API has transactions complying with the ACID feature. It supports rollback from the bottom of its architecture, to meet transaction atomicity.

#### • Better prefix scanning

When you run a SCAN command, for example, match prefix\*, on an instance, the scanning performance is much higher than that of open-source Redis. GeminiDB Redis API optimizes the complexity of delivering commands to O(log*N* + *M*), where *N* indicates the overall data volume and *M* indicates the matched data volume. The scanning complexity of open-source Redis is O(N), which is slower.

#### • Real-time persistence

- GeminiDB Redis API uses the persistence mechanism of write-ahead logging (WAL) to ensure data atomicity and durability. To ensure write performance, a response is returned immediately after data is written to the OS buffer. Data is written to disks in real time asynchronously to ensure real-time persistence and high-speed low-latency writes.

## 1.2 Highlights

Cloud-native GeminiDB is a key-value (KV) database service featuring high stability, cost-effectiveness, elasticity, and easy O&M. It is fully compatible with the Redis protocol, supports advanced functions such as PITR recoveries for game rollback and FastLoad for feature data import, and it allows you to set the field expiration time for hash keys and blacklist for high-risk keys.

GeminiDB is widely used in scenarios such as game friends list and player rankings, ad placement, personalized recommendations, e-commerce inventory, IoV data storage, and ERP systems. For details, see **1.4 Application Scenarios**.

GeminiDB has the following advantages over open-source KV databases (such as Redis and Pika databases):

Dimension	ltem	Open-Source KV Database	GeminiDB
Stability	Performance jitter caused by forks	Service stability is severely affected by fork issues. When RDB backups are generated, the Append Only File (AOF) is rewritten, or full data is synchronized, a fork is called. This increases latency and causes out of memory (OOM) issues.	Service stability is improved as fork issues are addressed. There is no performance jitter during backup and synchronization.
	Long latency in big key scenarios	The single-thread architecture slows down subsequent requests. In a single-thread architecture, big key requests slow down all subsequent requests and may trigger flow control or OOM on shards.	The multi-thread architecture reduces the impact on subsequent keys. GeminiDB uses a multi-thread architecture, which improves concurrency and reduces the impact of big keys on subsequent read and write operations of other keys.
	Bandwidth limiting during peak hours	Flow control is easily triggered, affecting services. Open-Source databases are deployed in hybrid mode, and bandwidth is strictly limited. Flow control is easily triggered for databases with low specifications.	Up to 10 Gbit/s is supported, allowing GeminiDB to handle service surges. By using an independent container deployment, GeminiDB can enable a load balancer to support a bandwidth of 10 Gbit/s.

 Table 1-1
 Comparison between GeminiDB and open-source KV databases

Dimension	ltem	Open-Source KV	GeminiDB
		Database	
	Impact of scale- out on services	Scale-out can take several minutes or sometimes even hours, greatly affecting services. Adding nodes involves data migration. Services may be affected for a few minutes or up to several hours.	Smooth scale-out is supported and has minimal impact on services. Scale-out can be completed in seconds and without interrupting services. Node addition does not involve data migration. Services are only affected for seconds.
	HA scenarios such as node breakdowns and primary/ secondary switchovers	Long switchover time: RTO > 30s	Second-level jitters, RTO < 10s
Performance	QPS	QPS per shard: 80,000 to 100,000	QPS per shard: 10,000 to 300,000
		In a single-thread architecture, the QPS of a single shard does not increase after CPUs are added.	In a multi-thread architecture, the QPS can increase linearly as CPUs are added.
	Latency	Low latency	Low latency
			In most service scenarios, the average latency is 1 ms, and the p99 latency is about 2 ms.
O&M capabilities	Audit logs of risky operations	Not supported	High-risk commands can be traced.
	Circuit breakers triggered by abnormal requests to keys	Not supported	Key blacklists and one-click circuit breakers for high- risk operations are supported, so the entire instance is not affected.
	Slow query logs	Supported	Supported. More details can be found in the logs.

Dimension	ltem	Open-Source KV Database	GeminiDB
	Big key diagnosis	Not supported	Online diagnosis of big keys by category is supported.
	Hot key diagnosis	Supported	Online diagnosis of hot keys is supported.
Cost	Utilization cost	The in-memory storage is expensive.	30% cost reduction with the same specifications
			Users can purchase additional compute resources and storage resources independently to eliminate the resource waste associated with coupled storage and compute.
	Data compression	Not supported	The compression ratio (4:1) enables databases with the same specifications to store more data.
	Scale-out	Coupled storage and compute increases costs exponentially.	Decoupled storage and compute supports independent scaling of compute and storage resources.
Availability	/	If any pair of primary and standby nodes is faulty, the entire cluster becomes unavailable.	GeminiDB provides superlative fault tolerance (N-1 reliability).
Data	/	Weak	High reliability
reliability		Thousands or tens of thousands of records will be lost if nodes are restarted and the network fluctuates. Weak data consistency may cause dirty reads.	GeminiDB provides three-copy storage, so it can serve as the primary database to replace the traditional DB+Cache solution, and it also ensures strong data consistency and avoids dirty reads.

Dimension	Item	Open-Source KV Database	GeminiDB
Advanced	Autoscaling	Not supported	Supported
features	Hash field expiration	Not supported	Supported. Service design is less complex and concurrency is increased.
	Fast data loading	Not supported	FastLoad allows feature data to be imported faster, reducing the impact on online services.
	Point-In-Time Recovery (PITR)	Not supported	Supported PITR rollbacks and quick data restoration to the original instance are supported, making GeminiDB a great fit for gaming applications.
	DR instances	Not supported	Intra-region and cross-region DR instances can be created.

## **1.3 Product Architecture**

GeminiDB Redis API supports the following instance types: proxy cluster, Redis Cluster, and primary/standby.

- Both proxy cluster and Redis Cluster instances support horizontal and vertical scaling and can handle millions of QPS and tens of terabytes of data. Redis Cluster is recommended because it features low latency, high concurrency, and high scalability.
- All proxy cluster and Redis Cluster instances can be read and written, improving resource utilization. Shared storage provides high availability. In the primary/standby architecture, only a primary instance can be read and written. Therefore, the cluster architecture is recommended.

The following table lists architecture types and application scenarios.

Туре	Description	Architectu re	Application Scenarios
Proxy cluster	In a sharded cluster, a Proxy Cluster GeminiDB Redis instance is connected through proxies to a standalone Redis instance, Redis Sentinel, and Redis Cluster.	For details, see Figure 1-1.	<ul> <li>Advantages: This type is easy to use. Sharding is not a concern. You can use a cluster like a single node. The proxies can distribute your requests to corresponding nodes for processing.</li> <li>Application scenario: The usage logic is simplified, and you do not need to pay much attention to shard management. For example, this type is recommended if you want to migrate data from a single node to a cluster and it is inconvenient to modify code on a client. Redis Cluster can be used to meet higher requirements on concurrency and latency.</li> </ul>
Redis Cluster (recom mende d)	With the native Redis Cluster architecture, a Redis Cluster GeminiDB Redis instance is directly connected to Redis Cluster.	For details, see Figure 1-2.	<ul> <li>Advantages: A Redis Cluster GeminiDB Redis instance is directly connected to clients, so there is less latency. A single shard of a Redis Cluster instance can handle more concurrency than that of a proxy cluster instance. Up to 128 nodes are supported.</li> <li>Application scenario: This type is applicable to services that are more sensitive to latency and have higher requirements on concurrency and scalability.</li> </ul>
Primar y/ Standb y	A primary/standby instance is compatible with a standalone Redis node and Redis Sentinel.	For details, see Figure 1-3.	A primary/standby Redis instance can be seamlessly switched to a primary/standby GeminiDB Redis instance. No code modification is required. Cluster instances provide better performance and scalability. They are recommended for new business.



#### Figure 1-1 Proxy cluster

#### Figure 1-2 Redis Cluster





#### Figure 1-3 Primary/Standby

## **1.4 Application Scenarios**

As a key-value database compatible with Redis APIs, GeminiDB Redis API extends application scenarios of Redis so that it can better meet diversified service requirements such as persistent and hybrid storage.

## **E-Commerce**

- For e-commerce applications, some commodity data is more frequently queried than others. GeminiDB Redis API stores frequently queried commodity information in memory as hot data, and cold data in the shared storage pool. This not only meets the quick access requirements of popular commodities, but also avoid excessive in-memory storage costs
- GeminiDB Redis API can permanently store massive amounts of historical order data of e-commerce applications. It allows you to access data through the Redis API and provides TB-level storage.
- There may be a large number of concurrent access requests within a short period of time during an e-commerce promotion. GeminiDB Redis API works as a front-end cache (large memory required) to help back-end databases handle service peaks. You can easily add compute nodes in seconds to handle the expected peak traffic.

## Gaming

• The schema of gaming services is simple. You can select GeminiDB Redis API as a persistent database and use simple Redis APIs to quickly develop and

launch services. For example, the sorted set structure of Redis can be used to display game rankings in real time.

• In delay-sensitive gaming scenarios, GeminiDB Redis API can be used as the front-end cache (large memory required) to accelerate access to applications.

#### Live Streaming

Live streams generate large amounts of hot data. Most of the data comes from popular live channels. To reduce costs for customers, GeminiDB Redis instances can store data from these popular live channels in memory and other data in shared disks.

#### **Online Education**

Online education applications store a large amount of data such as courses and Qs&As. However, only hot data (including most-viewed courses, latest question libraries, and lectures by famous teachers) is frequently accessed. GeminiDB Redis instances can store data separately in memory and shared disks, achieving a balance between performance and costs.

### **Persistent Storage for Other Applications**

With the rapid development of the Internet, various large-scale applications have increasing requirements for persistent storage. Specifically, a massive amount of data needs to be stored, including historical orders, feature engineering, log records, location coordinates, machine learning, and user profiles. A common feature of these scenarios is large data volume and long validity period. Therefore, a large-capacity and low-cost key-value storage service is required to collect and transfer data. Redis is the most widely used key-value service. Its various data structures and operation APIs have innate advantages in storing such data. However, the native Redis can only be used as a cache and cannot guarantee persistence.

In addition to compatibility with Redis APIs, GeminiDB Redis API provides largecapacity, low-cost, and high-reliability data storage capabilities, making it wellsuited to persistent storage scenarios.

## **1.5 Compatible API and Versions**

This section describes the compatible API and versions supported by GeminiDB Redis API.

Compatible API	Version
Redis	7.0, 6.2 (including 6.2. <i>X</i> ), 5.0, and earlier versions

## **1.6 Instance Specifications**

This section describes available GeminiDB Redis instance specifications. The instance specifications depend on the selected CPU model.

GeminiDB Redis instances facilitate hot and cold data exchanges while offering storage that significantly exceeds the memory limit. Hot data is stored in the memory, and full data is stored in the high-performance storage pool. The total instance space refers to the total storage capacity, which determines the upper limit of data storage. Table 1-5 lists the node memory capacity.

Ins ta nc e Ty pe	Stor age (GB )	Node Flavor	No de CP Us	No de Me mo ry (G B)	Nod es	QPS	Max. Conne ctions	Datab ases	ACL Accou nts
Cl ust er	4	geminidb.r edis.mediu m.2	1	2	2	20,000	20,000	256	200
	8	geminidb.r edis.mediu m.4	1	4	2	20,000	20,000	256	200
	16	geminidb.r edis.large. 4	2	8	2	40,000	20,000	256	200
	24	geminidb.r edis.large. 4	2	8	3	60,000	30,000	256	200
	32	geminidb.r edis.large. 4	2	8	4	80,000	40,000	256	200
	48	geminidb.r edis.xlarge .4	4	16	3	120,00 0	30,000	1,000	200
	64	geminidb.r edis.xlarge .4	4	16	4	160,00 0	40,000	1,000	200
	96	geminidb.r edis.2xlarg e.4	8	32	3	240,00 0	30,000	1,000	200

 Table 1-3 GeminiDB Redis cluster instance specifications (fast configuration)

lns ta nc e Ty pe	Stor age (GB )	Node Flavor	No de CP Us	No de Me mo ry (G B)	Nod es	QPS	Max. Conne ctions	Datab ases	ACL Accou nts
	128	geminidb.r edis.2xlarg e.4	8	32	4	320,00 0	40,000	1,000	200
	192	geminidb.r edis.2xlarg e.4	8	32	6	480,00 0	60,000	1,000	200
	256	geminidb.r edis.2xlarg e.4	8	32	8	640,00 0	80,000	1,000	200
	384	geminidb.r edis.2xlarg e.4	8	32	10	800,00 0	100,00 0	1,000	200
	512	geminidb.r edis.4xlarg e.4	16	64	6	960,00 0	60,000	1,000	200
	768	geminidb.r edis.4xlarg e.4	16	64	9	1,440,0 00	90,000	1,000	200
	102 4	geminidb.r edis.4xlarg e.4	16	64	12	1,920,0 00	120,00 0	1,000	200
	204 8	geminidb.r edis.4xlarg e.4	16	64	22	3,520,0 00	220,00 0	1,000	200
	409 6	geminidb.r edis.8xlarg e.4	32	12 8	24	7,680,0 00	240,00 0	1,000	200
	819 2	geminidb.r edis.8xlarg e.4	32	12 8	36	11,520, 000	360,00 0	1,000	200

<b>Table 1-4</b> Primary/Standby GeminiDB Redis instance specifications (fast	
configuration)	

In st an ce Ty pe	Stor age (GB )	Node Flavor	No de CP Us	No de M e m or y (G B)	No de s	Sh ar ds	QP S	Max. Conn ectio ns	Datab ases	ACL Accoun ts
Pri m ar	4	geminidb.r edis.mediu m.2	1	2	2	1	8,0 00	10,00 0	1,000	200
y/ st an db	8	geminidb.r edis.mediu m.2	1	2	2	1	8,0 00	10,00 0	1,000	200
у	16	geminidb.r edis.mediu m.4	1	4	2	1	10, 000	10,00 0	1,000	200
	24	geminidb.r edis.large. 4	2	8	2	1	20, 000	10,00 0	1,000	200
	32	geminidb.r edis.large. 4	2	8	2	1	20, 000	1,000 0	1,000	200
	48	geminidb.r edis.xlarge .4	4	16	2	1	40, 000	2,000 0	1,000	200
	64	geminidb.r edis.xlarge .4	4	16	2	1	40, 000	2,000 0	1,000	200
	96	geminidb.r edis.2xlarg e.4	8	32	2	1	80, 000	2,000 0	1,000	200
	128	geminidb.r edis.4xlarg e.4	16	64	2	1	160 ,00 0	2,000 0	1,000	200

Node Flavor	vCPU s	Mem ory (GB)	Max. Persistent Storage per Node (GB)	Maximum Connections per Node	Assu red Band widt h (Mbi t/s)	Datab ases
geminidb.re dis.medium. 2	1	2	4	10,000	800	256
geminidb.re dis.large.2	2	4	8	10,000	1200	256
geminidb.re dis.xlarge.2	4	8	16	10,000	2500	1,000
geminidb.re dis.2xlarge.2	8	16	32	10,000	5000	1,000
geminidb.re dis.4xlarge.2	16	32	64	10,000	9000	1,000
geminidb.re dis.8xlarge.2	32	64	128	10,000	1800 0	1,000
geminidb.re dis.medium. 4	1	4	8	10,000	800	256
geminidb.re dis.large.4	2	8	16	10,000	1200	256
geminidb.re dis.xlarge.4	4	16	32	10,000	2500	1,000
geminidb.re dis.2xlarge.4	8	32	64	10,000	5000	1,000
geminidb.re dis.4xlarge.4	16	64	128	10,000	9000	1,000
geminidb.re dis.8xlarge.4	32	128	256	10,000	1800 0	1,000
geminidb.re dis.medium. 8	1	8	16	10,000	800	256
geminidb.re dis.large.8	2	16	32	10,000	1200	256
geminidb.re dis.xlarge.8	4	32	64	10,000	2500	1,000

Table 1-5 GeminiDB Redis instance node specifications

Node Flavor	vCPU s	Mem ory (GB)	Max. Persistent Storage per Node (GB)	Maximum Connections per Node	Assu red Band widt h (Mbi t/s)	Datab ases
geminidb.re dis.2xlarge.8	8	64	128	10,000	5000	1,000
geminidb.re dis.4xlarge.8	16	128	256	10,000	9000	1,000
geminidb.re dis.8xlarge.8	32	256	512	10,000	1800 0	1,000

The bandwidth of a single node of some existing instances is 768 Mbit/s, which is the same as that of other Redis Cloud services. If existing instances experience high throughput, you can choose **Service Tickets > Create Service Ticket** in the upper right corner of the console to consult on the bandwidth usage.

## **1.7 Instance Statuses**

The status of a DB instance indicates the health of the instance. You can view the DB instance statuses on the management console.

Status Description				
Available	The instance is available.			
Abnormal The instance is abnormal.				
Creating The instance is being created.				
Creation failed	The instance failed to be created.			
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 up	The storage space of an instance is being scaled up.			
Changing instance class	The vCPUs and memory of an instance are being changed.			

 Table 1-6 DB instance statuses

Status	Description				
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.				
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.				
Configuring SSL	SSL is being enabled or disabled.				
Frozen	The instance is frozen because your balance drops to or below zero.				
Unfreezing	The instance is being unfrozen after the overdue payments are cleared.				
Checking changes	The yearly/monthly instance is pending check when its billing mode is changed.				

## **1.8 Constraints**

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

## Specifications

Resource Type	Specifications	Description	
CPU and memory	GeminiDB Redis API supports proxy cluster and primary/standby	• For details about specifications of different instance types, see <b>1.6 Instance Specifications</b> .	
Instances.	<ul> <li>You can change the specifications to meet your service requirements by following 4.6.4 Changing vCPUs and Memory.</li> </ul>		
Storage space	The storage space depends on the selected <b>instance specifications</b> .	Storage can be scaled up or down. For details, see <b>4.6.7.1 Overview</b> .	

Resource Type	Specifications	Description
Connections	The maximum value is the number of instance nodes multiplied by 10,000.	The maximum number of connections varies depending on the memory. For details, see <b>1.6</b> Instance Specifications.

## 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 <b>4.14</b> Tag Management.
Free backup space	GeminiDB Redis instances provide free backup storage.	For more information, see <b>Backup</b> <b>Storage</b> .
Retention period	The default value is 7 days. The value ranges from 1 to 3660 days.	For more information, see Configuring an Automated Backup Policy.

## Naming Rules

#### Table 1-9 Naming rules

ltem	Description
Instance name	<ul> <li>Contains 4 to 64 characters.</li> <li>Must start with a letter. Only letters (case-sensitive), digits, hyphens (-), and underscores (_) are allowed.</li> </ul>
Backup name	<ul> <li>Contains 4 to 64 characters.</li> <li>Must start with a letter. Only letters (case sensitive), digits, hyphens (-), and underscores (_) are allowed.</li> </ul>
Parameter template name	<ul> <li>Contains 1 to 64 characters.</li> <li>Only letters (case sensitive), digits, hyphens (-), underscores (_), and periods (.) are allowed.</li> </ul>

## Security

## Table 1-10 Security

ltem	Description	
Password of database administrator <b>rwuser</b>	<ul> <li>Contains 8 to 32 characters.</li> <li>Can contain at least two types of the following characters: uppercase letters, lowercase letters, digits, and special characters ~!@#%^*=+? For more information, see 4.6.3 Changing the Administrator Password of a GeminiDB Redis Database.</li> <li>Keep your password secure. The system cannot retrieve it if it is lost.</li> </ul>	
Database port	Database port number. You can specify a port number, which ranges from 1024 to 65535 except 2180, 2887, 3887, 6377, 6378, 6380, 8018, 8079, 8091, 8479, 8484, 8999, 9864, 9866, 9867, 12017, 12333, and 50069. If you do not specify a port number, port 6379 is used by default.	
VPC	After a GeminiDB Redis instance is created, the VPC where the instance is deployed cannot be changed.	
Security group	A security group controls access between GeminiDB Redis API and other services. Ensure that the security group you selected allows your client to access the instance. If no security group is available, the system creates one for you.	
Access control	A load balancer address does not support security groups. After an instance is created, configure IP address access control. If no whitelist is configured, all IP addresses that can communicate with the VPC can access the instance.	
ACL account	GeminiDB Redis API provides enterprise-grade multi- tenancy. You can add read-only or read/write accounts for your instance to control access to each database to avoid misoperations from other tenants. A maximum of 200 ACL accounts can be created for each instance. For more information, see <b>4.10.2 ACL Account</b> <b>Management</b> .	

## **Instance Operations**

Table 1-11	Instance	operations
------------	----------	------------

Function	Constraint	
Database access	<ul> <li>If remote access is not enabled, GeminiDB Redis instances and their associated ECSs must be in the same VPC subnet.</li> </ul>	
	<ul> <li>The security group must allow access from the associated ECS.</li> <li>By default, a GeminiDB Redis instance cannot be accessed through an ECS in a different security group. You need to add an inbound rule to the security group.</li> </ul>	
	• The default port of a GeminiDB Redis instance is 6379.	
	• The database port can be set when an instance is created and can be changed 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 Redis instance	• GeminiDB Redis instances cannot be rebooted through commands. They must be rebooted on the console.	
	• Restarting an instance will interrupt services, so off- peak hours are the best time. Ensure that your application can be reconnected.	
Viewing GeminiDB Redis instance backups	GeminiDB Redis instance backups are stored in OBS buckets and are invisible to you.	
Changing the CPU or memory of a GeminiDB Redis 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.	
	• 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.	

Function	Constraint
Primary/Standby switchover	Only primary/standby GeminiDB Redis instances are supported. During a primary/standby switchover, the instances are disconnected for less than 10 seconds, which can cause slow latency or command execution failures. Ensure commands can be retried or the client can be reconnected. You are advised to perform the switchover during off-peak hours.
Data restoration	To prevent data loss, you are advised to back up key data before data restoration.
Storage space	If the storage space of an instance is full, data cannot be written to databases. You are advised to periodically check the storage space. GeminiDB Redis instance storage can be automatically scaled up in case of a sudden surge in data volumes. Enable autoscaling by following <b>4.6.7.3</b> Automatically Scaling Up Storage Space.
Recycle bin	<ul> <li>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.</li> <li>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.</li> <li>Currently, you can put a maximum of 100 instances into the recycle bin.</li> <li>If you delete an instance running out of storage, it will not be moved to the recycle bin.</li> </ul>

For details about other development and O&M specifications that can effectively evaluate and improve service system stability, see **5.1 Development and O&M Rules**.

# **2** Billing

- 2.1 Billing Overview
- 2.2 Billing Modes
- 2.3 Billing Items
- 2.4 Billing Examples
- 2.5 Billing Mode Changes
- 2.6 Renewing Subscriptions
- 2.7 Bills
- 2.8 Arrears
- 2.9 Billing Termination
- 2.10 Cost Management
- 2.11 Billing FAQs

## 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 service 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 **2.2.1 Overview**.

You can also change the billing mode later if it no longer meets your needs. For details, see **2.5.1 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 **2.3 Billing Items**.

For more information about billing samples and the billing for each item, see **2.4 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 **2.6.1 Overview**.

#### • Viewing Bills

You can choose **Billing & Costs** > **Bills** to check the instance transactions and bills. For details, see 2.7 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 **2.8** Arrears.

• Stopping Billing

If you no longer need to use your GeminiDB Redis instance, you can unsubscribe from or delete it to stop the billing. For details, see **2.9 Billing Termination**.

#### • Managing Costs

GeminiDB Redis costs include resource costs and O&M costs. You can allocate, analyze, and optimize GeminiDB costs to save more money. For details, see **2.10 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.

Billing Mode	Yearly/Monthly	Pay-per-use
Payment	Prepaid Billed by the subscription term you purchase	Postpaid Billed for what you use
Billing Method	Billed by the subscription term you purchase	Calculated by the second but billed every hour
Billing 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 2.5.3 Changing a Yearly/Monthly Instance to Pay-per-Use.	Pay-per-use can be changed to yearly/monthly. For details, see 2.5.2 Changing a Pay-per-Use Instance to Yearly/Monthly.
Changing the Specificati ons	Supported	Supported
Applicatio n 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.

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

## 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 for yearly/ monthly GeminiDB Redis 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.

## **Billed Items**

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

Billed Item	Description
Instance specificatio ns	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 Redis provides backup storage up to 100% of the database storage you purchase at no additional charge.
	After the free backup space is used up, charges are applied based on the backup space pricing details. Pricing is listed on a per-hour basis, but bills are calculated based on the actual usage duration.
(Optional) Public network bandwidth	GeminiDB Redis instances are accessible from public networks, and you are billed for the generated public network traffic, but not for private network traffic.

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

If you want to purchase a 3-node (specifications of each node: 2 vCPUs) GeminiDB Redis instance with 12 GB of storage space. At the bottom of the instance buying page, price details (excluding the backup space fee) will be displayed.

#### Figure 2-1 Example price





The price includes:
- Selected specifications for your instance
- Storage space

#### 

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

#### Billed Usage Period

A yearly/monthly GeminiDB Redis 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 purchased a one-month GeminiDB Redis instance on March 08, 2023, 15:50:04, the billed usage period is from March 08, 2023, 15:50:04 to April 08, 2023, 23:59:59.

#### Billing Examples

Suppose you purchased a one-month GeminiDB Redis instance (instance specifications: 2 vCPUs, Dedicated Edition; nodes: 3; storage: 40 GB; backup space: 50 GB (40 GB for free)) on March 08, 2023, 15:50:04, and renewed the subscription for one more month before the initial subscription expired. That would include two usage periods:

- March 08, 2023, 15:50:04 to April 08, 2023, 23:59:59
- April 08, 2023, 23:59:59 to May 08, 2023, 23:59:59
  - From April 08, 2023, 23:59:59 to May 01, 2023, 23:59:59, 20 GB of free backup space was used.
  - From May 01, 2023, 23:59:59 to May 08, 2023, 23:59:59, another 10 GB of backup space was used, which was billed for 168 hours.

You will be billed for both usage periods. GeminiDB Redis resources are billed individually as follows:

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 <b>Cluster CPU/Memory</b> on <b>Product Pricing Details</b> .

Table 2-3 Formulas for billing yearly/monthly resources

Resource	Formula	Unit Price	
Storage space	Storage space unit price x Required duration x Storage space (GB)	For details about the unit price, see <b>Storage Space</b> on <b>Product</b> <b>Pricing Details</b> .	
Backup space	Backup space unit price x Required duration x (Backup space – Storage space) (GB)	For details about the unit price, see <b>Backup Storage Space</b> on <b>Product Pricing Details</b> .	
	NOTE The billed duration refers to the length of time the billed backup space was used for.		
Public network bandwidth	Billed by fixed bandwidth	For details, see <b>Product Pricing</b> <b>Details</b> .	

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

#### NOTICE

Prices in the figure are just examples. Actual prices are subject to **Product Pricing Details**.





#### **Price Change After Specification Change**

If the specifications of a yearly/monthly GeminiDB Redis 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 upgrade your GeminiDB Redis instance specifications, you need to pay the difference in price.

• If you downgrade your GeminiDB Redis instance specifications, Huawei Cloud will refund you the difference.

You are not advised to downgrade your GeminiDB Redis instance to a lower specification because the instance performance may be affected. Suppose you purchased a yearly/monthly instance (1 vCPU | 6 GB and 3 nodes) on April 08, 2023 and upgraded the instance specifications to 2 vCPUs | 12 GB and 3 nodes on April 18, 2023. The price for the original specifications was \$199.59 USD/month, and that for the new specifications was \$399.18 USD/month. The price difference will be calculated as follows:

#### Price difference for the specification upgrade = Price for the new specifications × Remaining period - Price for the original specifications × 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, Remaining period = 12 (Remaining days in April)/30 (Maximum number of days in April) + 8 (Remaining days in May)/31 (Maximum number of days in May) = 0.6581. Cost of upgrade = 399.18 USD × 0.6581 – 199.59 USD × 0.6581 = 131.35 USD.

For more details, see Pricing of a Changed Specification.

#### Impact of Expiration

**Figure 2-3** shows the statuses a yearly/monthly GeminiDB Redis instance can go through throughout its lifecycle. After a GeminiDB Redis 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 Redis instance



#### **Expiration Reminder**

The system will send you a reminder (by email, SMS, or in-app message) 7 days before a yearly/monthly GeminiDB Redis instance expires to remind you to renew the subscription.

#### Impact of Expiration

If your yearly/monthly GeminiDB Redis 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 Redis 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 Redis 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 Redis instance while it is in the retention period.

If the yearly/monthly GeminiDB Redis 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 **2.6.1 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 Redis 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.

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

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

If you want to purchase a pay-per-use 3-node (specifications of each node: 2 vCPUs) GeminiDB Redis instance with 12 GB of storage space. At the bottom of the page for buying an instance, price details (excluding the backup storage fee) will be displayed.



<b>.</b> .	¢0 02		0
Price	ΦU.05	USD/nour	9

The price includes:

- 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**

A pay-per-use GeminiDB Redis instance is billed by the second and settled on the hour (GMT+8). After the bill is generated, a new billing cycle starts. The billing starts when the GeminiDB Redis instance is created and ends when the instance 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 Redis instance at 8:45:30 and deleted 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 0.5 vCPUs (dedicated), 40 GB of storage, and 50 GB of backup storage (40 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 Redis instances are billed individually as follows.

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 <b>GeminiDB Price Calculator</b> .		
Storage	Storage unit price x Required duration	See the estimated price of a cluster instance with specified storage in <b>GeminiDB Price Calculator</b> .		
Backup storage	Backup storage unit price x Required duration x (Backup storage – Storage) (GB) <b>NOTE</b> The billed duration refers to the length of time the billed backup storage was used for.	See the estimated price of a cluster instance with specified backup storage in <b>GeminiDB Price</b> <b>Calculator</b> .		
Public network traffic	<ul> <li>Tiered billing by fixed bandwidth</li> <li>0 Mbit/s to 5 Mbit/s (included): billed at a fixed unit price per Mbit/s</li> <li>Greater than 5 Mbit/s: billed at a different price per Mbit/s</li> </ul>	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.

#### NOTICE

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

For pay-per-use billing, decimal numerals on the price calculator are rounded off and are accurate to two decimal places. If the rounded price falls below USD0.01, it will be shown as USD0.01.





#### 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 0.54 vCPUs and 3 GB of memory at 9:00:00 and upgraded it to 1 vCPU and 6 GB memory at 9:30:00, two billing records are generated between 9:00:00 and 10:00:00:

- 0.5 vCPUs and 3 GB of memory from 9:00:00 to 9:30:00
- 1 vCPU and 6 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 Redis instance throughout its lifecycle. After a GeminiDB Redis 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 Redis 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 Redis 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 public network traffic. For details, see **Table 2-6**.

#### **NOTE**

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

Billing Item	Description	Billing Mode	Formula		
* Specific ations	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 For details about the unit price, see Cluster CPU/Memory on Product Pricing Details.		
* Storage space	Billed based on unified standards.	Yearly/ Monthly and pay- per-use	Unit price x Storage space x Required duration 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 For details about the unit price, see Backup Storage Space on Product Pricing Details. NOTE The billed duration refers to the length of time the billed backup space was used for.		
Public network traffic	<ul> <li>An EIP is required if a GeminiDB Redis instance needs to access the Internet.</li> <li>Billed by bandwidth, traffic, and the EIP reservation price.</li> <li>EIP for a yearly/monthly GeminiDB Redis instance: billed by bandwidth.</li> <li>EIP for a pay-per-use GeminiDB Redis instance: billed by bandwidth, traffic, or shared bandwidth, traffic, or shared bandwidth. You are also charged for IP reservation if you do not bind the EIP to any instance.</li> </ul>	Yearly/ Monthly and pay- per-use. You can purchase a bandwidt h add-on package or a shared traffic package.	<ul> <li>Tiered pricing based on fixed bandwidth.</li> <li>0 Mbit/s to 5 Mbit/s (included): billed at a fixed unit price per Mbit/s.</li> <li>Greater than 5 Mbit/s: billed at a different price per Mbit/s.</li> <li>For details about the unit price, see</li> <li>Bandwidth Price on Product Pricing Details or Product Pricing Details.</li> </ul>		

#### **Billing Examples**

Suppose you purchased a one-month GeminiDB Redis instance (instance specifications: 2 vCPUs, Dedicated Edition; nodes: 3; storage: 40 GB; backup space: 50 GB (40 GB for free)) on March 08, 2023, 15:50:04, and renewed the subscription for one more month before the initial subscription expired. That would include two usage periods:

- March 08, 2023, 15:50:04 to April 08, 2023, 23:59:59
- April 08, 2023, 23:59:59 to May 08, 2023, 23:59:59
  - From April 08, 2023, 23:59:59 to May 01, 2023, 23:59:59, 20 GB of free backup space was used.
  - From May 01, 2023, 23:59:59 to May 08, 2023, 23:59:59, another 10 GB of backup space was used, which was billed for 168 hours.

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

#### NOTICE

Prices in the figure are only for reference. For details, see Product Pricing Details.





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

## 2.4 Billing Examples

#### **Billing Scenario**

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

- Specifications: 2 vCPUs | 12 GB
- Nodes: 3
- Public network bandwidth: 6 Mbit/s

After a period of time, the user found that the current GeminiDB Redis instance specifications no longer met service requirements and updated the specifications to 4 vCPUs | 24 GB at 09:00:00 on March 20, 2023. Since the user wanted to use the instance long term, the user then changed the instance to yearly/monthly billing with a one-month duration at 10:30:00 on the same day. So how much will the user be billed for this GeminiDB Redis instance in March and April?

#### **Billing Analysis**

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

- Pay-per-use usage: March 18, 2023, 15:30:00 to March 20, 2023, 10:30:00
  - March 18, 2023, 15:30:00 to March 20, 2023, 9:00:00
    - Instance specifications: 2 vCPUs | 20 GB
    - Nodes: 3
    - Used storage space: 20 GB
    - Used backup space: 20 GB
    - Public network bandwidth: 6 Mbit/s
  - March 20, 2023, 9:00:00 to March 20, 2023, 10:30:00
    - Instance specifications: 4 vCPUs | 40 GB
    - Nodes: 3
    - Used storage space: 40 GB
    - Used backup space: 50 GB (billed on a pay-per-use basis from March 20, 2023, 10:00:00 to March 20, 2023, 10:30:00)
    - Public network bandwidth: 6 Mbit/s
- Yearly/Monthly: March 20, 2023, 10:30:00 to April 20, 2023, 23:59:59
  - Instance specifications: 4 vCPUs | 80 GB
  - Nodes: 3
  - Used storage space: 80 GB
  - Used backup space: 100 GB (billed on a pay-per-use basis from April 10, 2023, 23:59:59 to April 20, 2023, 23:59:59)
  - Public network bandwidth: 6 Mbit/s
  - Billed duration: one month

#### NOTICE

Unit prices in this example are used for reference only. The prices shown here are only estimates. As unit prices change from time to time, the prices shown here will differ from actual prices. For details, see the data released on the Huawei Cloud official website.

#### Pay-per-use

From March 18, 2023, 15:30:00 to March 20, 2023, 09:00:00, a GeminiDB Redis instance with specifications 2 vCPUs | 20 GB was used for 41.5 hours, so the price would be calculated as follows.



From March 20, 2023, 09:00:00 to March 20, 2023, 10:30:00, a GeminiDB Redis instance with specifications 4 vCPUs | 40 GB was used for 1.5 hours, so the price would be calculated as follows.



#### Yearly/Monthly

From March 20, 2023, 10:30:00 to April 20, 2023, 23:59:59, a GeminiDB Redis instance purchased using yearly/monthly billing was used for one month, so the price would be calculated as follows.



From March to April, the total price of this GeminiDB Redis instance is \$901.44 USD (38.18 + 2.65 + 860.61).

## 2.5 Billing Mode Changes

## 2.5.1 Overview

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

Billing Item	Change Description	Reference	
Instance specification s (vCPUs and nodes)	Changing the billing mode of a GeminiDB Redis instance includes the changes to compute resources (vCPUs and nodes).	<ul> <li>2.5.2 Changing a Pay-per-Use Instance to Yearly/Monthly</li> <li>2.5.3 Changing a</li> </ul>	
	<ul> <li>Change from pay-per-use to yearly/monthly to enjoy lower prices.</li> </ul>	Yearly/Monthly Instance to Pay-per- Use	
	<ul> <li>Change from yearly/monthly to pay-per-use to use the GeminiDB Redis instance more flexibly.</li> </ul>		
	NOTE Such a change takes effect only after the yearly/monthly subscription ends.		

Table 2-7	Changeable	billing	items of	GeminiDB	Redis	instances
		· · · J				

Billing Item	Change Description	Reference
EIP	<ul> <li>A yearly/monthly EIP can be changed to a pay-per-use EIP billed by bandwidth after the yearly/monthly subscription ends.</li> <li>A pay-per-use EIP billed by</li> </ul>	<ul> <li>2.5.2 Changing a Pay-per-Use Instance to Yearly/Monthly</li> <li>2.5.3 Changing a Yearly/Monthly</li> </ul>
	yearly/monthly EIP.	Instance to Pay-per- Use
	• 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.	
	For details, see <b>Figure 2-8</b> .	

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



1: The change takes effect immediately.

2: The change takes effect only after the yearly/monthly subscription period expires.

×: The billing mode cannot be changed.

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

If you have a pay-per-use GeminiDB Redis 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 Redis 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**.

Resourc e	Billing Mode	Billed By	Band width Type	Changed to Yearly/ Monthly Billing with the GeminiDB Redis Instance	Handling Measure
EIP	Pay- per-use	Bandwid th	Dedica ted	Supported	Change the EIP to yearly/ monthly billing on the EIP console.
					For details, see <b>Changing</b> EIP Billing Mode.
EIP	Pay- per-use	Traffic	Dedica ted	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:
					<ol> <li>Change the EIP to be billed by bandwidth on a pay-per-use basis.</li> </ol>
					<ol> <li>Change the EIP to be billed on a yearly/ monthly basis.</li> </ol>
					For details, see <b>Changing</b> EIP Billing Mode.

 Table 2-8 EIP billing mode change rules

#### **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 Changing a pay-per-use instance to yearly/monthly

 
 NameID θ
 DB Instance...
 Compatible....
 Stor...
 Status θ
 Specifications
 Storage Space
 Load balan...
 Enterprise...
 Billing Mode
 Operation

 Cluster
 Reds 6.2
 Shared
 O Available
 2 VCPUs Standard Oracide
 0.01%
 0.01%
 0.0160B
 default
 Payoen-Upt Created on J...
 Log In Charge to Yearly Monthly More ~

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 Changing a pay-per-use instance to yearly/monthly

Billing Info	rmation	
Billing Mode		Created
Pay-per-use	Change to Yearly/Monthly	Jun 25, 2024 16:29:58 GMT+08:00

#### 

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 C 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 Redis 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 Redis instance at 15:29:16 on April 18, 2023 and changed it to pay-per-use billing at 16:30:00 on the same day. On the **Billing Center > Billing** page, bills 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.

#### D 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**.

Resour ce	Billing Mode	Billed By	Bandwi dth Type	Change to Pay-Per-Use Billing with GeminiDB Redis Instance	Handling Measure
EIP	Yearly/ Monthl y	Bandwi dth	Dedicat ed	Not supported	Change the EIP to yearly/monthly billing on the EIP console. For details, see Changing EIP Billing Mode.
EIP	Yearly/ Monthl y	Traffic	Dedicat ed	Not supported	<ul> <li>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:</li> <li>1. Change the EIP to be billed by bandwidth on a pay-per-use basis.</li> <li>2. Change the EIP to be billed by traffic on a pay-per-use basis.</li> <li>For details, see Changing EIP Billing Mode.</li> </ul>

Table 2-9 EIP billing mode change rules

#### 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 2-1	Figure 2-11 Change to Fay-per-ose									
	Cluster	Redis 6.2	Shared 🧿 Available	2 vCPUs Standard 0 2 nodes	0.01% 0/16GB	default	Yearly/Monthly 30 days until Log In Renew More ~			
							Change to Pay-per-Use Change Specifications			
							Create Backup			
							Scale Storage Space			
							Add Node Restart			
							Reset Password			
							Unsubscribe			
							Create Dual-Active Relationship			
							Create DR Relationship			
							Rename High-risk Command			
							Import Data			

#### 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 Payper-Use**. The billing mode will change to pay-per-use after the instance expires. Auto renewal will be disabled after the billing mode of your instances change to pay-per-use. Exercise caution when performing this operation.
- 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 Payper-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 Redis instance manually or automatically.

Method	Description
2.6.2 Manually Renewing an Instance	You can renew a yearly/monthly instance anytime on the console before it is automatically deleted.
2.6.3 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.

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

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 Auto Renewal



- 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 management console.
- Step 2 In the service list, choose Databases > GeminiDB.
- **Step 3** On the **Instances** page, locate the instance that you want to renew and click **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 Renewal button

#### **Billing Information**

Billing Mode		
Yearly/Monthly	Renew	Enable Auto-Renewal
Created		
Jun 26, 2024 17	7:27:52 G	MT+08:00
Upon Expiration	ı	
Entering grace	period (	?)

Order CS2406261716BFYGA

Expiration Date Jul 26, 2024 23:59:59 GMT+08:00

#### **NOTE**

To renew multiple yearly/monthly instances at a time, perform the following steps:

- 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 a Subscription in Billing Center

- **Step 1** Log in to the management console.
- **Step 2** Hover over **Billing & Costs** in the upper part of the console and choose **Renewal** from the drop-down list.

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

Billing	Enterprise	Tools
Unpaid C	Orders	
Renewal		
My Pack	ages	
Bills		
Invoices		
Cost Cer	nter	

You can move all resources that need to be manually renewed to the **Manual Renewals** tab page. 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

Manual Renewals (	9) Auto Renewals	(0) Pay-per-Use After Expiration (0)	) Renewals Cance	eled (0)			
Batch Renew	Enable Auto-Renew	Change to Pay-per-Use After Expiration	Cancel Renewal	Batch Export			Export Renewal Prices
Instan	ce Name/ID	Product Type/Specifications	Region	Provisioned/Expires	Status	Validity Period	Operation
~				Apr 26, 2023 10:38:08 GMT+08:00 May 26, 2023 23:59:59 GMT+08:00	Frozen	3 days until deletion Delete after retention period	Cancel Renewal   More 👻
× 🗆				Apr 26, 2023 09:55:03 GMT+08:00 May 26, 2023 23:59:59 GMT+08:00	Frozen	3 days until deletion Delete after retention period	Renew More -

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

#### Figure 2-17 Batch renewal

Manual Renewals (9) Auto Renewals (1	Manual Renewals (9) Auto Renewals (0) Pay-per-Use After Expiration (0) Renewals Canceled (0)								
Batch Renew Enable Auto-Renew	Change to Pay-per-Use After Expiration	Cancel Renewal E	Batch Export			Export Renewal Prices			
Instance Name/ID	Product Type/Specifications	Region	Provisioned/Expires	Status	Validity Period	Operation			
~ []			Apr 26, 2023 10:38:08 GMT+08:00 May 26, 2023 23:59:59 GMT+08:00	Frozen	3 days until deletion Delete after retention period	Cancel Renewal   More 👻			
~			Apr 26, 2023 09:55:03 GMT+08:00 May 26, 2023 23:59:59 GMT+08:00	Frozen	3 days until deletion Delete after retention period	Renew   More -			
~ 🛛			May 16, 2023 15:29:36 GMT+08:00 Jun 16, 2023 23:59:59 GMT+08:00	Frozen	4 days until deletion Delete after retention period	Renew   More +			

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

Instance Name/ID	Product Informatio	in		Auto-Renew	Renewal Duration	Renewal Date		Fee
~				(€ <sup>1</sup> None	1 month	Current: May 26, 20 Renewed: Jun 26, 2	23 23: 023 2	
enewal Duration 1 month 2 enewal Date Renew on	months 3 months the standard renewal date, th	4 months 5	months 6 mor at 23:59:59 GMT+08	onths 7 months 8:00 🖉	8 months 9 n	months 1 year 🖻	2 years 🖬	3 years 📾
If you chan	ge the expiration date to Re	newal Date, the expen	ditures will be added.	d. You can check the rene	ewal days in the Renewa	al Duration column.		

**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.

3		· · · · · <b>,</b> · · · · · ·	·· · · · · · ·
Procedure	1. Configure a renewal date.	2. Select resources for operations.	3. Renew to the renewal date.
Rules	For example, the renewal date is the Resource A Expiration: April 17 Resource B Renewal for 1 r Expiration: May 08	nonth Additional renewal for 14 days Additional renewal for 14 days Additional renewal fo days	Expiration: June 01

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 of an instance depends on the subscription term.
  - Monthly subscriptions renew each month.
  - Yearly subscriptions renew each year.
- 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.
- 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 management console.
- **Step 2** Hover over **Billing & Costs** in the upper part of the console and choose **Renewal** from the drop-down list.
- **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**, **Payper-Use After Expiration**, and **Renewals Canceled** pages.

Figure 2-21 Renewal management

Renewals	Feedback IP Quick Links	Renew Domain Set Renewal Date	Modify Message Recipient	Set Deduction Date for Auto-Renewal
1. If you want to continue using any resources about to expire, refer to How Do I Renew Resources     2. Renewals or changes to pay-per-use will be applied after the current subscription term ends.     3. If you want to renew your subscriptions more easily, refer to Muther a Cardo Period and     4. If you want to renew your subscriptions more easily, refer to Automatically Renewing a Resource	? and How Do I Change the Billing Mode from Yearly/Monthly to Pay- I a Retention Period? and Setting a Renewal Date	per-Use?		
() Instances expiring soon: 0 ; Instances to be frozen: 0 ; Instances to be released: 7 . Please renew i	n time. View instances.			
Expires Expire in 30 days Expire in 15 days Expire in 7 d	Custom Name/ID/Order Number	ame, resource ID, or orde – Q		
Manual Renewals (1) Auto Renewals (0) Pay-per-Use After Expiration (0)	Renewals Canceled (0)			
Batch Renew Enable Auto-Renew Change to Pay-per-Use After Expiration	Cancel Renewal Batch Export			Export Renewal Prices C
Instance Name/ID Product Type/Specifications	Region Provisioned/Expires	Status	Validity Period	Operation
▼ □	Jul 18, 2023 11:34:01 GMT+08:00 Aug 18, 2023 23:59:59 GMT+08:00	Provisioned	25 days until expiration Delete after retention period	Renew   More +

**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

Renewals		Feedback      Quick Links     Rener	v Domain Set Renewal Date	Modify Message Recipient	Set Deduction Date for Auto-Renewal
<ol> <li>I. If you want to continue using any resources about to expire, refe 2. Renewals or changes to pay-per-use will be applied after the cu 3. If you want to learn what happens after a resource expires, refe 4. If you want to renew your subscriptions more easily, refer to Aut</li> </ol>	er to How Do I Renew Resources? and How Do I Change rrent subscription term ends. r to What Are a Grace Period and a Retention Period? omatically Renewing a Resource and Setting a Renewal I	the Billing Mode from Yearly/Monthly to Pay-per-Use? Date			
Instances expiring soon: 0 ; Instances to be frozen: 0 ; Instances t	o be released: 7 . Please renew in time. View instances.				
Expires Expire in 30 days Expire in 15 days Expire in 7 days Service Type	lays Expired Frozen Custom Name/ID/0 All     Do not sho	Order Number  F Enter a resource name, res wresources that have orders pending payment	surce ID, or orde <b>Q</b>		
Manual Renewals (1) Auto Renewals (0) Pay-pe	r-Use After Expiration (0) Renewals Cancel	ed (0)			
Batch Renew Enable Auto-Renew Change to Pay	-per-Use After Expiration Cancel Renewal	Batch Export			Export Renewal Prices
Instance Name/ID Product T	rpe/Specifications Region	Provisioned/Expires	Status	Validity Period	Operation
~ 🗆		Jul 18, 2023 11:34:01 GMT+08:00 Aug 18, 2023 23:59:59 GMT+08:00	Provisioned	25 days until expiration Delete after retention period Enable Auto-Renew	Renew   More .
				Change to Pay-per-Us	se After Expiration
				Release ()	

• 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.

Manual Renewals	(9) Auto Renewals	(0) Pay-per-Use After Expiration (0	) Renewals Cance	• led (0)			
Batch Renew	Enable Auto-Renew	Change to Pay-per-Use After Expiration	Cancel Renewal	Batch Export			Export Renewal Prices
🚺 Insta	nce Name/ID	Product Type/Specifications	Region	Provisioned/Expires	Status	Validity Period	Operation
× 🗉				Apr 26, 2023 10:38:08 GMT+08:00 May 26, 2023 23:59:59 GMT+08:00	() Frozen	3 days until deletion Delete after retention period	Cancel Renewal   More +
× 🗆				Apr 26, 2023 09:55:03 GMT+08:00 May 26, 2023 23:59:59 GMT+08:00	() Frozen	3 days until deletion Delete after retention period	Renew   More 🕶
~ 🗆				May 16, 2023 15:29:36 GMT+08:00 Jun 16, 2023 23:59:59 GMT+08:00	Frozen	4 days until deletion Delete after retention period	Renew   More 👻
~ 🗆				May 18, 2023 18:19:32 GMT+08:00 Jun 18, 2023 23:59:59 GMT+08:00	Frozen	6 days until deletion Delete after retention period	Renew   More 💌
~ 🗆				May 18, 2023 17:06:19 GMT+08:00 Jun 18, 2023 23:59:59 GMT+08:00	Frozen	6 days until deletion Delete after retention period	Renew   More 💌
~				Jun 01, 2023 22:51:24 GMT+08:00 Jul 01, 2023 23:59:59 GMT+08:00	() Frozen	19 days until deletion Delete after retention period	Cancel Renewal   More 💌
~				Jun 02, 2023 11:34:42 GMT+08:00 Jul 02, 2023 23:59:59 GMT+08:00	() Frozen	6 hours 48 minutes until del Delete after retention period	··· Cancel Renewal   More 💌
~ 🗹				Jul 18, 2023 11:34:01 GMT+08:00 Aug 18, 2023 23:59:59 GMT+08:00	Provisioned	25 days until expiration Delete after retention period	Renew   More 🔻
~				Jul 24, 2023 15:54:35 GMT+08:00 Jul 24, 2024 23:59:59 GMT+08:00	Provisioned	366 days until expiration Delete after retention period	Renew   More +

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

1. Huawei Cli 2. You can m account 7 da 3. You can pa	oud starts deducting renew anually renew your resour ys before the expiration of ty for auto-renewal using y	val fees from your ces at any time ev the new subscript your account balar	account 7 days before the e en if auto-renew is enabled. ion term. ice, discounts, coupons, and	expiration of the curr After a manual rene d stored-value cards	ent subscription ewal is complete, . Payment Rules	term. Ensure that your a auto-renew is still in eff for Auto-Renewal	iccount balance is suffi fect, and Huawei Cloud	icient. 1 will start deductin <u>c</u>	) renewal fees from your
	Instance Name/ID	Service	Current Configuration	Region	Billing M	Validity Period	Current Auto-R	Remaining	End Time
× 🗹					Monthly	25 days until exp	None	Unlimited	-
New Auto-Ren	ew (III)								
renou	1 month	3 m	onths	6 months		9 months		1 year	
Auto-renewals	Preset Auto-rener	wals							
	ок								

----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**

Transaction records for yearly/monthly subscriptions are generated immediately after being paid for.

A pay-per-use resource is billed by the hour, day, or month, depending on the resource's usage type. The GeminiDB Redis instance usage is billed by the hour. For details, see **Bill Run for Pay-per-Use Resources** 

You are not charged immediately after a record is generated. For example, if a pay-per-use GeminiDB Redis instance (which is billed on an hourly basis) is deleted at 08:30, you will still have expenditures for the 08:00 to 09:00 hour. However, you will not likely be billed for the 08:00 to 09:00 hour until about 10:00.

On the **Bills** page of the Billing Center, select the **Bill Details** tab. **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 management console and choose Databases > GeminiDB Redis 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

Basic Information	
DB Instance Name	DB Instance ID
A ()	đ

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

The **Bills** page is displayed.

Step 5 Choose Transactions and Detailed Bills > Bill Details. On the displayed page, select Resource ID as the filter criteria, enter the obtained instance ID, and click the Q icon.

#### Figure 2-26 Searching for a bill

Transactio	n Bills 🕜 🛛	Bill Details ⑦											
Billing Cycle	Dec 2023	*											
Sort By	Usage 🙆	Data Period	y billing cycle	By day	Details	Search for res	ources?						
Resource	Name:	S Ad	d filter									× C	λ ∓ ⊚
Billing	Enterpr 🍞	Account Name (?)	Service 7	Resour 🍞	Billing 🍸	Bill Type 🍞	Resource N	Resource Tag	Specificatio	Region 🍞	AZ	Usage Type	Unit Price
Dec 2	default		GeminiDB (	GeminiDB S	Pay-per-Use	Expenditure	geminidb-6e a9307387cb	-			AZ1,AZ2	Duration	0.003
Dec 2	default		GeminiDB (	GeminiDB N	Pay-per-Use	Expenditure	geminidb-6e e4e2103b54	-			AZ2	Duration	1.3
Dec 2	default		GeminiDB (	GeminiDB N	Pay-per-Use	Expenditure	geminidb-6e 36/6da1900	-			AZ1	Duration	13
Dec 2	default		GeminiDB (	GeminiDB N	Pay-per-Use	Expenditure	geminidb-6e 1e99218fd9	-			AZ3	Duration	1.3
Dec 2	default		GeminiDB (	GeminiDB I	Pay-per-Use	Expenditure	geminidb-6e a9307387cb	-			AZ1,AZ2	architecture	0

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

----End

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

- Step 1 Log in to the management console and choose Databases > GeminiDB Redis 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** page, obtain the instance name.



Basic Information						
DB Instance Name	•					
	1	0				

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

The **Bills** page is displayed.

**Step 5** Choose **Transactions and Detailed Bills** > **Bill Details**. On the displayed page, select **Resource Name** as the filter criteria, enter the obtained instance ID, and click the *Q* icon.

Figure 2-28 Searching for a bill

Transactio	n Bills 🕜	Bill Details (?)											
Billing Cycle	Dec 2023	*											
Sort By	Usage 🛞	Data Period B	y billing cycle	By day	Details	Search for res	ources?						
Resource	Name:	IV Ada	d filter									×Q	. ± 💿
Billing	Enterpr 🏹	Account Name ③	Service 7	Resour 🏹	Billing 🍸	Bill Type 🏼 🏹	Resource N	Resource Tag	Specificatio	Region 7	AZ	Usage Type	Unit Price (
Dec 2	default		GeminiDB (	GeminiDB S	Pay-per-Use	Expenditure	geminidb-6e a9307387cb	-			AZ1,AZ2	Duration	0.003
Dec 2	default		GeminiDB (	GeminiDB N	Pay-per-Use	Expenditure	geminidb-6e e4e2103b54	-			AZ2	Duration	1.7
Dec 2	default		GeminiDB (	GeminiDB N	Pay-per-Use	Expenditure	geminidb-6e 36f6da1900	-			AZ1	Duration	1.7
Dec 2	default		GeminiDB (	GeminiDB N	Pay-per-Use	Expenditure	geminidb-6e 1e99218fd9	-			AZ3	Duration	1.7
Dec 2	default		GeminiDB (	GeminiDB I	Pay-per-Use	Expenditure	geminidb-6e a9307387cb	-			AZ1,AZ2	architecture	0.

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

----End

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

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

• Transaction Records

Pay-per-use GeminiDB Redis instance usage is calculated by the second and but billed on an hourly basis. You can check the transaction records against the actual usage. The billed resources are billed separately. For details, see **Table 2-11**.

Table	2-11	GeminiDB	Redis	transaction	records
iubic	~ ! !	GCHIIIDD	ricuis	uansaction	records

Service Type	GeminiDB Redis
Resour ce Type	GeminiDB Redis storage
Billing Mode	Pay-per-use
Expend iture Time	<ul> <li>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:</li> <li>2023/04/08 10:09:06 - 2023/04/08 11:00:00</li> <li>2023/04/08 11:00:00 - 2023/04/08 12:00:00</li> <li>2023/04/08 12:00:00 - 2023/04/08 12:09:06</li> </ul>
List Price	List price on the official website = Usage x Unit price x Capacity The GeminiDB Redis instance was used for 3,054 seconds in the first period, and the unit price can be obtained on the <b>Pricing</b> <b>Details</b> page. The list price for the first period = $(3054 \div 3600) \times$ 0.0007 × 40 = \$0.02375333 USD. Similarly, you can calculate the GeminiDB Redis instance list price for the other periods.
Discou nted Amoun t	Discounts offered for cloud services, for example, commercial discounts, partner authorized discounts, and promotional discounts. It is the discounted amount based on the list price.
Truncat ed Amoun t	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. Take the first period as an example. The truncated amount is \$0.00375333 USD.
Amoun t Due	Amount due = List price – Discount amount – Truncated amount 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 Redis instance

Bill details can display in multiple ways. By default, the bill details of a resource are displayed by usage and by billing cycle. Table 2-12 illustrates the GeminiDB Redis instance bill details, which can be used to check against the actual usage.

Table 2-12 GeminiDB Re	dis bill details
------------------------	------------------

Service	GeminiDB Redis
Туре	

Resour ce Type	GeminiDB Redis storage
Billing Mode	Pay-per-use
Resour ce Name/I D	Name and ID of a specific GeminiDB Redis instance Example: nosql-b388 and 21e8811a64bf4de88bc2e2556da17983in12
Specific ations	GeminiDB Redis storage
Usage Type	Duration for a pay-per-use GeminiDB Redis instance
Unit Price	<ul> <li>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.</li> <li>You can search for the unit price for pay-per-use GeminiDB Redis instances on Product Pricing Details.</li> </ul>
Unit	Displayed on the <b>Product Pricing Details</b> page. Example: USD/GB/hour.
Usage	Depends on the unit of the unit price, which is USD/GB/hour. Storage usage is billed by the hour. Example: 2 hours.
Usage Unit	Hour
List Price	List price on the official website = Usage x Unit price x Capacity GeminiDB Redis instance is used for 2 hours in total, and the unit price is obtained on the <b>Product Pricing Details</b> page. The list price = 2 * 0.0007 * 40 = \$0.056 USD.
Discou nted Amoun t	Discounts offered for cloud services, for example, commercial discounts, partner authorized discounts, and promotional discounts. It is the discounted amount based on the list price.
Amoun t 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 Redis resources even if your account is in arrears. However, you cannot perform operations such as purchasing GeminiDB Redis instances, upgrading instance specifications, and renewing subscriptions, because they will generate new expenditures.

• Pay-per-Use

If your account is in arrears due to automated deduction for pay-per-use GeminiDB Redis 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 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

#### D 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 Redis 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 Redis instance, you make a one-time up-front payment. By default, the billing automatically stops when the purchased subscription expires.

- If a yearly/monthly resource is no longer needed before the subscription expires, you can unsubscribe from the resource. The system will return a certain amount of money to your account based on whether the resource is subject to five-day unconditional unsubscription or whether cash coupons or discount coupons are used. 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 Redis 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.]

The **Bills** page is displayed.

**Step 2** Choose **Transactions and Detailed Bills** > **Bill Details**, and 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 management console and choose Databases > GeminiDB Redis API.

**Step 4** Select the region where the resource is located, select **Instance ID** and enter the resource ID copied in **Step 2**, and click the <sup>Q</sup> icon 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.]

The **Bills** page is displayed.

**Step 2** Choose **Transactions and Detailed Bills** > **Bill Details**, and click the icon shown in the following figure to copy the resource name.

Figure 2-32 Copying the resource name

Transactio	n Bills 🕜	Bill Details ⑦											
Billing Cycle	Dec 2023	•											
Sort By	Usage 📀	Data Period	By billing cycle	By day	Details	Search for reso	urces?						
Resource	ID: 3e2f3299ad1b	4d45a2f0473c296db72	idin13 🔘 🖓 Ad	ld filter								×Q	. <b>∓</b> ⊚
Billing	Enterpr 7	Account Name (?	) Service 🏹	Resour 🍞	Billing 7	Bill Type 🏼 🏹	ø			Region 🏹	AZ	Usage Type	Unit Price
Dec 2	default		GeminiDB (	GeminiDB I	Pay-per-Use	Expenditure	geminidb-38 3e2f3299ad	-	GeminiDB I		AZ1,AZ2	architecture	0

- Step 3 Log in to the management console and choose Databases > GeminiDB Redis API.
- **Step 4** Enter the instance name copied in **Step 2** in the search box and click  $\mathbf{Q}$ .

Figure 2-33 Searching for resources										
All projects	Q Instance name: × Add filter									
Name/ID \varTheta	DB Instance Compatible Stor Status 🖯	Specifications Storage Space	Load balan Enterprise	Billing Mode Ope						

**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.

2 vCPUs Standard 0.01% 0/16GB

**NOTE** 

Cluster Redis 6.2 Shared O Available

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 Redis costs consist of two parts:

- Resource costs: costs of compute and storage resources. For details, see 2.2 Billing Modes.
- O&M costs: labor costs incurred during the use of GeminiDB Redis.

Resource costs	O&M costs
EIP bandwidth	0
Compute resources	6
Storage Backup space	Labor costs

## 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

Enterprise Project	<b>.</b>	С	View Project Management	

• 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 details, see **Viewing Cost 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 Redis 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 Redis resources and delete idle instances in a timely manner.
- If your services require high performance stability, purchase a general-purpose instance to reduce your costs. For example, in the same specifications (4 vCPUs and 24 GB memory), the cost of a general-purpose instance is 30% lower than that of a dedicated instance.

#### 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-peruse or vice versa.

- For details about how to change the billing mode from yearly/monthly to a pay-per-use, see **2.5.3 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 **2.5.2 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 Redis instances.

#### **Usage Notes**

- For billing information of a GeminiDB Redis instance, see **Billing Overview**.
- Pay-per-use instances cannot be renewed.

#### 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 instance that you want to renew and click **Renew** in the **Operation** column.

Figure 2-36 Renewal

Name/ID 🔶	DB Instance	Compatible	Stor	Status 🕀	Specifications	Storage Space		Load balan	Enterprise	Billing Mode	Operation
	Cluster	Redis 6.2	Shared	Available	2 vCPUs Standard 2 nodes	0.01%	0/16GB		default	Yearly/Monthly 30 days until	Log In Renew More s

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

# Billing Information O Billing Mode O Yearly/Monthly Renew Enable Auto-Renewal C Created E: Jun 26, 2024 17:27:52 GMT+08:00 Ju Upon Expiration Entering grace period Entering grace period ?

Order CS2406261716BFYGA

Expiration Date Jul 26, 2024 23:59:59 GMT+08:00

**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

Auto Scale Renew	Change to Yearly/Month	ly Change	to Pay-pe	r-Use Unsubscribe	Upgrade Mino	r Version					
All projects	✓ ] Q	A	dd filter								
✓ Name/ID ⊖	DB Instance	Compatible	Stor	Status 😔	Specifications	Storage Space	•	Load balan	Enterprise	Billing Mode	Operation
	Cluster	Redis 6.2	Shared	Available	2 vCPUs Standard 2 nodes	0.01%	0/16GB		default	Yearly/Monthly 30 days until	Log In Renew More ~
	Cluster	Redis 6.2	Shared	Available	2 vCPUs Standard 2 nodes	0.01%	0/16GB	-	default	Yearly/Monthly 30 days until	Log In Renew More ~

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.

#### Usage Notes

- For billing information of a GeminiDB Redis instance, see **Billing Overview**.
- 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. Ensure that the manual backup is complete before submitting the unsubscription request.

#### Unsubscribing from a Single Yearly/Monthly Instance

Step 1	Log i	n 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 and choose More > Unsubscribe in the Operation column.

#### Figure 2-39 Unsubscribe





**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**.

#### NOTICE

- 1. After an unsubscription request is submitted, resources and data will be deleted and cannot be retrieved.
- 2. Ensure that the manual backup is complete 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

#### Unsubscribing from Yearly/Monthly Instances In Batches

- Step 1 Log in to the Huawei Cloud console.
- Step 2 In the service list, choose Databases > GeminiDB Redis API.
- **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

Auto Scale Renew	Change to Yearly/Month	thange	e to Pay-pe	r-Use Unsubscribe	Upgrade Mino	r Version						
All projects	✓ ] Q	1	Add filter									
✓ Name/ID ⊕	DB Instance	Compatible	Stor	Status 🖯	Specifications	Storage Spac	0	Load balan	Enterprise	Billing Mode	Operation	
•	Cluster	Redis 6.2	Shared	Available	2 vCPUs Standard 2 nodes	0.01%	0/16GB	-	default	Yearly/Monthly 30 days until	Log In Renew M	ore ~
	Cluster	Redis 6.2	Shared	Available	2 vCPUs Standard 2 nodes	0.01%	0/16GB	-	default	Yearly/Monthly 30 days until	Log In Renew M	ore ~

- 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**.

#### NOTICE

- 1. After an unsubscription request is submitted, resources and data will be deleted and cannot be retrieved.
- 2. Ensure that the manual backup is complete 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 Redis

- 3.1 Getting to Know GeminiDB Redis API
- 3.2 Buying and Connecting to a Cluster Instance
- 3.3 Buying and Connecting to a Primary/Standby Instance
- 3.4 Getting Started with Common Practices

# 3.1 Getting to Know GeminiDB Redis API

This section describes GeminiDB Redis product type and instance type, helping you quickly create and connect to a GeminiDB Redis instance.

Product Type	Scenario	Supported Instance Type
Standard	Stable and low-latency performance is provided, suitable for common scenarios such as advertising and recommendation, gaming, e-commerce, and Internet of Vehicles (IoV).	<ul> <li>Proxy cluster</li> <li>Redis Cluster</li> <li>Primary/ Standby</li> </ul>
Capacity- oriented	Large-capacity key-value storage is provided, suitable for average performance requirements and expectations of low costs.	Proxy cluster

#### Table 3-1 Product types

Table 3-2	Instance types
-----------	----------------

Instance Type	Scenario	Reference
Cluster	<ul> <li>In a sharded cluster, a proxy cluster GeminiDB Redis instance is connected through proxies to a standalone Redis instance, Redis Sentinel, and Redis Cluster. The proxy cluster instance has strong horizontal scaling capabilities and can handle millions of QPS and dozens of terabytes of data.</li> <li>With the native Redis Cluster architecture, a Redis Cluster GeminiDB Redis instance is directly connected to Redis Cluster. It can greatly reduce latency while improving performance.</li> </ul>	3.2 Buying and Connecting to a Cluster Instance
Primary/ Standby	A primary/standby instance is compatible with a standalone Redis node and Redis Sentinel. This instance type is used when hashtags are unavailable.	3.3 Buying and Connecting to a Primary/Standby Instance

#### **Connection Methods**

Data Admin Service (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 Redis instances.

	Table	3-3	Connection	on	DAS
--	-------	-----	------------	----	-----

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

More Connection Operations

• See 4.3.1 Connecting to a GeminiDB Redis Instance.

# 3.2 Buying and Connecting to a Cluster Instance

This section describes how to buy and connect to a proxy cluster or Redis Cluster GeminiDB Redis instance on the GeminiDB console.

- In a sharded cluster, a proxy cluster GeminiDB Redis instance is connected through proxies to a standalone Redis instance, Redis Sentinel, and Redis Cluster. The proxy cluster instance has strong horizontal scaling capabilities and can handle millions of QPS and dozens of terabytes of data.
- With the native Redis Cluster architecture, a Redis Cluster GeminiDB Redis instance is directly connected to Redis Cluster. It can greatly reduce latency while improving performance.

Each tenant can create a maximum of 50 GeminiDB Redis 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.

- Step 1: Buy an instance.
- Step 2: Connect to the instance through DAS.

For details about other connection methods, see **4.3 Instance Connection** and Management.

#### Step 1: Buying an 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 specifications, and click **Next**.

The following parameters are for reference only. Select proper specifications as needed. Table 4-1 lists details about the parameters.

Billing Mode	Yearly/Monthly Pay-per-ase
Region	• · ·
	Regions are geographic areas isolated from each other. For low network latency and quick resource access, select the nearest region.
DB Instance Name	geminide.  ©  If you buy multicle DB instances, the wystem automatically accords a date, time, and serial number to the end of the instance names (format, instance name-AMNDD-HHmmss-SH).
Compatible ADI	Parla Custadas DusunDB UBus IndurDB HassiDB
Companye APT	necka Casancas Ojuanocu Fuzas menucu monjocu
Storage Type	Classic Cloud nailye
	The traditional architecture is stable and reliable.
Product Type	Standard Capacity-oriented
	This type provides stable and low-latency performance. It is good for advertising and recommendations, gaming, e-commerce, and Internet of Vehicles (IoV).
DB Instance Type	Pruzy dualer PedraCluster Primary/Standby O Primary/standby instances with 4 GB of memory free for a limited time
	With a shaded cluster architecture, the type of instance supports connections through provide and is compatible with Redis clusters and Codis.
	too kee loof vir mone moose maannook.
Compatible Version	<b>62</b> 7.0
	Fully compatible with 6 = 4 and earlier versions, such as 5.0,4.0, and 2.8.
CPU Type	286
AZ	as(az2,az3 az2 az3 az5
	Three-AZ deployment is recommended to provide cross-AZ DR and ensure RPO is 0.

Figure 3-1 Billing mode and basic information (proxy cluster)



Parameter	Example Value	Description
Billing mode description	Pay-per-use	<ul> <li>Billing mode of an instance</li> <li>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 discount. This mode is a good option for long-term stable services.</li> <li>Pay-per-use: A postpaid billing mode. Pay as you go and just pay for what you use. The DB instance usage is calculated by the second but billed every hour. 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.</li> </ul>
Region	Select <b>CN-Hong</b> Kong.	Region where a tenant is located <b>NOTICE</b> To reduce network latency, select a region nearest from which you will access the instance. Instances deployed in different regions cannot communicate with each other over a private network. After you buy an instance, you cannot change its region.
DB Instance Name	User-defined	<ul> <li>The instance name:</li> <li>Can be the same as an existing instance name.</li> <li>Can contain 4 to 64 characters and must start with a letter. It is case-sensitive and allows only letters, digits, hyphens (-), and underscores (_).</li> </ul>

#### Figure 3-2 Billing mode and basic information (Redis Cluster)

Parameter	Example Value	Description
Compatible API	Redis	GeminiDB is compatible with mainstream NoSQL APIs, including Redis, DynamoDB, Cassandra, HBase, MongoDB, and InfluxDB. You can select GeminiDB APIs by following <b>How Do I Select an API?</b>
Storage Type	Classic	<ul> <li>Classic: classic architecture with decoupled storage and compute</li> <li>Cloud native: more flexible, new-gen version with support for more AZs</li> </ul>
Product Type	Standard	• <b>Standard</b> : Stable and low-latency performance is provided for common scenarios such as advertising and recommendation, gaming, e-commerce, and Internet of Vehicles (IoV).
		• <b>Capacity-oriented</b> : Large-capacity key- value storage is suitable for average performance requirements and expectations of low costs.
		<b>Standard</b> is unavailable for instances with cloud native storage.
DB Instance Type	Proxy cluster	<ul> <li>Proxy cluster:         <ul> <li>In a sharded cluster, a proxy cluster</li> <li>GeminiDB Redis instance is connected</li> <li>through proxies to a standalone Redis</li> <li>instance, Redis Sentinel, and Redis Cluster.</li> <li>The proxy cluster instance has strong</li> <li>horizontal scaling capabilities and can</li> <li>handle millions of QPS and dozens of</li> <li>terabytes of data.</li> </ul> </li> <li>RedisCluster</li> <li>With the native Redis Cluster architecture,</li> <li>a Padis Cluster CominiDR Padia instance is</li> </ul>
		a Redis Cluster GeminiDB Redis instance is directly connected to Redis Cluster. It can greatly reduce latency while improving performance.
Compatible Version	6.2	7.0, 6.2 (including 6.2. <i>X</i> ), 5.0, and earlier versions
CPU Type	x86	x86 CPUs use the Complex Instruction Set Computing (CISC) instruction set. Each instruction can be used to execute low- level hardware operations. Executing these instructions is complex and time- consuming.

Parameter	Example Value	Description
AZ	AZ 1, AZ 2, and AZ 3	Availability zone where the instance is created. An AZ is a part of a region with its own independent power supplies and networks. AZs are physically isolated but can communicate with each other over a private network.

#### Figure 3-3 Specifications and storage

Instance Creation Method	Fast configure	Standard configure						
instance Specifications	Storage		vCPUs	Nodes	ops @	Maximum Connections	Databases (defaultimaximum) 💿	Accounts
	0 4.69		Special offer 1 vCPU	2	16.000	20,000	1.000	200
	. 8 08		Standard 1 vCPU	2	20.000	20.000	1,000	200
	0 16.68		Standard 2 vCPUs	2	40.000	20.000	1.000	200
	O 24 G8		Standard 2 vCPUs	5	60,000	30,000	1,000	200
	O 32 GB		Standard 2 vCPUs	4	80,000	40,000	1,000	200
	_ 40 GB		Standard 4 vCPUs	0	120,000	30,000	1,000	200
	O 64 68		Standard 4 vCPUs	4	160,000	40,000	1,000	200
Specification Preview	Telal capacity 8 GB   Note 5	ecilications Blandard 1 v	CPU   Nodes 2Count   GPS 20,000   Maximum	Connections 20,000   Data copies 3				

Parameter	Example Value	Description
Instance Creation Method	Fast configure	<ul> <li><b>Description</b></li> <li>Two options are available:</li> <li><b>Fast configure</b>         Provides you with         recommended specifications.         You can select one of them         based on service         requirements, without the         need to specify the         specifications, node         quantity, and storage space.     </li> <li><b>Standard configure</b>         Provides a standard process         to configure instance         specifications, including         specifying the specifications,</li> </ul>
		node quantity, and storage space. Currently, a maximum of 36 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.

Parameter	Example Value	Description
Instance Specifications	2U8GB	Higher CPU specifications provide better performance. Select specifications as needed.
		For details, see <b>1.6 Instance</b> <b>Specifications</b> .

#### Figure 3-4 Network and database configurations

VPC	
Subret	O (Vectore)     Manador Statement Production The control state (1)
Security Group	lefaul 🕐 🕐 Vive Security Grap
Database Port	
Access Control	Carlone Sig
Password	Codigate Dig
Password	٩
Confirm Password	<u>ه</u>
Password-Pine Access	Certiper 9a
Enterprise Project	(-dasa- v ) ♂ two/hastlangerer ⊗
Purchased Quantity -	

Parameter	Example Value	Description	
VPC	default_vpc	Virtual private network where your instances are located. A VPC isolates networks for different services. You can seled an existing VPC or create a VPC <b>NOTE</b>	
		<ul> <li>After a GeminiDB Redis instance is created, its VPC cannot be changed.</li> </ul>	
		• If you want to connect to a GeminiDB Redis instance through an ECS over an internal network, the GeminiDB Redis instance and the ECS must be in the same VPC. If they are not in the same VPC, you can create a <b>VPC peering connection</b> to enable access.	
Subnet	default_subnet	A subnet provides dedicated network resources that are logically isolated from other networks for security purposes.	

Parameter	Example Value	Description
Database Port	6379	Port number for accessing a database. If you do not specify a port number, default port <b>6379</b> is used. You can specify a port number based on your requirements. The port number ranges from 1024 to 65535 except 2180, 2887, 3887, 6377, 6378, 6380, 8018, 8079, 8091, 8479, 8484, 8999, 12017, 12333, and 50069.
Access Control	Skip	<ul> <li>Skip: Access is restricted based on the VPC access policy by default.</li> <li>Configure:         Specify how access is controlled. Three options are available:         All IP addresses: All IP addresses can access the instance.         Whitelist: Only IP addresses in a group can access the instance.         Blacklist: IP addresses in a group cannot access the instance.         NOTE Redis Cluster GeminiDB Redis instances do not support access control.     </li> </ul>
Password	Skip	<ul> <li>Skip: You can set the database password after creating an instance.</li> <li>Configure: You can set the database password when creating an instance.</li> <li>NOTE You cannot set a password after creating a Redis Cluster GeminiDB Redis instance.</li> </ul>

Parameter	Example Value	Description		
Password	Configured based on the password policy	If <b>Password</b> is set to <b>Configure</b> , you need to set the database password.		
		• Must be 8 to 32 characters long.		
		<ul> <li>Can contain at least two types of the following characters: uppercase letters, lowercase letters, digits, and special characters ~!@#%^*- _=+?</li> </ul>		
		• For security reasons, set a strong password. The system will verify the password strength.		
		Keep your password secure. The system cannot retrieve it if it is lost.		
Password-Free Access	Skip	If you configure password-free access for a CIDR Block of an instance, you do not need to enter a password when connecting to the instance.		
		For details, see Table 4-5. NOTE Redis Cluster GeminiDB Redis instances do not support password- free access.		
Enterprise Project	default	This parameter is provided for enterprise users.		
		An enterprise project groups cloud resources, so you can manage resources and members by project. The default project is <b>default</b> .		
		Select an enterprise project from the drop-down list. For more information about enterprise projects, see <i>Enterprise</i> <i>Management User Guide</i> .		

Retain the default values for other parameters.

5. On the order confirmation page, check the instance information. If you need to modify the information, click **Previous**. If no modification is required, read and agree to the service agreement and click **Submit**.

6. Click **Back to Instance Management** to go to the instance list.

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

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

Figure 3-5 Available instance

pennish Otekan Prov dudar Redisto Cauce O Auduar 3100% Streter 810% 91000 Pv erbut Oberk Light Dueryth Traybothy New v Debut

#### 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 the permission of remote login. DAS is secure and convenient for connecting to DB instances.

- 1. Log in to the Huawei Cloud console.
- 2. In the service list, choose **Databases** > **GeminiDB**.
- 3. In the instance list, locate the target instance and click **Log In** in the **Operation** column.

**Figure 3-6** Connecting to a GeminiDB Redis instance

Alternatively, click the instance name to go to the **Basic Information** page. Click **Log In** in the upper right corner of the page.

Figure 3-7	Connecting to a	GeminiDB	Redis	instance
	5			

< Gemman	V Vision		g manual can be	Contran Cogin
Basic Information				
Backups & Restorations	Basic Information			
Node Management	DB Instance Name	D6 Instance ID	Storage Type	Product Type
Accounts	geminido_	12966	Classic	Standard
Slaw Guery Logs				
Audit Logs	Status	Region	AZ	CB Instance Type
Parameters	Available		822	Proxy cluster
Metrica	Enterprise Project	Maintenance Window ③		
Sessions	(select	62:00-06:00 Change		
Prinanacia Annhaia				

4. Enter a password for logging in to the instance.

Figure 3-8 Logging in to the GeminiDB Redis instance



If you need to log in again after the password is reset, click **Re-login** in the upper right corner and use the new password.

Figure 3-9 Re-login

		_		-
Current Database (	Re-login	Save to Executed Command	s 🛛 🔍	)
00 v Exclut FR See (4) Contants (Car #10)			Ī	
Executed Commands Messages Results				

5. Manage relevant databases.

#### Figure 3-10 Instance homepage

Current Database:DB0   DB Instance Name	Save to Executed Commands 🕥 🔵
DB0 V Execute (F8) Save (My Commands) (Clear (F19)	
1	
Executed Commands Messages Results	

- Save commands to the execution record.

This function is enabled by default to save the recently executed commands for your later query.

Then you can click the **Executed Commands** tab on the lower page to view historical commands.

#### Figure 3-11 Viewing executed commands

Executed Commands Messages Results			
Executed	Command	Time Required	Result
Jun 26, 2024 10:34:29 GMT+08:00	SCAN 0	2ms	Succeeded
Jun 26, 2024 10:33:52 GMT+08:00	SCAN 0	3ms	Succeeded

If this function is disabled, the commands executed subsequently are not displayed. You can click 
 next to **Save Executed SQL Statements** in the upper right corner to disable this function.

– Execute a command.

Enter a command in the command window and click **Execute** or **F8**.

**NOTE** 

- Do not use transactions, Lua scripts, Pub/Sub commands, or other commands that have blocking semantics.
- For an instance that supports multiple databases, you can change the current database on the console but cannot change it using a SELECT statement.

#### Figure 3-12 Executing a command

DB0 ~	Execute (F8)	Save My Commands Clear (F10)		
1 SCAN 0				
Executed Commands	Messages	Results		
Command			Cursor	Result
SCAN 0			0	(empty list or set)

After a command is executed, you can view the execution result on the **Results** page.

– Save a command.

You can save a command to all instances, the current instance, or the current database. Then you can view details in **My Commands**.

#### Figure 3-13 Saving a command

DB0		Save		×
1	SCAN 0	Title Application Scope	All instances   Current instance Current database	
			OK Cancel	)

- View my commands.

Common commands are displayed the **My Commands** page.

You can set a filter to narrow the scope of commands. If you select **All**, all commands saved in the current account are displayed.

#### Figure 3-14 Filtering commands

My Com	mands					
Create	Command			All ^	Enter a title or command.	Q
No.	Title	Application Scope	Com	All	Operation	
1	scan	Current instance	scan	All instances	Edit Delete	Copy to Command Window
2	select	All instances	sele	Current instance Current database	Edit Delete	Copy to Command Window

Alternatively, you can enter a command title or statement in the search box to search for the corresponding command.

#### Figure 3-15 Searching for a command

My Commands				×
Create Command		All	~ s	X   Q
No. Title	Application Scope	Command		Operation
1 scan	Current instance	scan O		Edit Delete Copy to Command Window
2 select	All instances	select 1		Edit Delete Copy to Command Window

On the **My Commands** page, you can also create, edit, and delete a command or copy it to the command window.

Figure 3-16 Managing a command

My Con	mands				
Create	Command		All	~	Enter a title or command. Q
No.	Title	Application Scope	Command		Operation
1	scan	Current instance	scan O		Edit Delete Copy to Command Window
2	select	All instances	select 1		Edit Delete Copy to Command Window

– Clear a command.

You can also press F10 to clear the command in the command window.

#### Figure 3-17 Clearing a command



#### 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 Buying and Connecting to a Primary/Standby Instance

This section describes how to buy and connect to a primary/standby GeminiDB Redis instance on the GeminiDB console.

A primary/standby instance is compatible with a standalone Redis node and Redis Sentinel. This instance type is used when hashtags are unavailable.

Each tenant can create a maximum of 50 GeminiDB Redis 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.

- Step 1: Buy an instance.
- Step 2: Connect to the instance through DAS.

For details about other connection methods, see **4.3 Instance Connection** and Management.

#### Step 1: Buying an 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 specifications, and click **Next**.

The following parameters are for reference only. Select proper specifications as needed. Table 4-9 lists details about the parameters.

Figure 3-18 Billing mode and basic information

Billing Mode	YearlyMonthly Pay on use
Region	<b>0</b> · · )
	Regions are geographic areas isolated from each other. For low network latency and quick resource access, select the nearest region.
DB Instance Name	geminidb
	If you buy multiple DB instances, the system automatically appends a date, time, and serial number to the end of the instance names (format.instance_name-MMDD-0+timms-SN).
Compatible API	Redis Cassandra DynamoD8 HBase InfluxD8 MongoD8
Storage Type	Classic Cloud table
	The fraditional architecture is statele and reliable.
Product Type	Standard Capacity-oriented
	This type provides stable and low-latency performance. It is good for advertising and recommendations, gaming, e-commerce, and Internet of Vehicles (IoV).
DB Instance Type	Pray duater RedisCluster Primary/Standay O Primary/standay instances with 4 GB of memory free for a limited time
	With a standard primary/standby architecture, this type of instance is compatible with single-node instances and Redis Sentinet.
	You can buy 47 more Redis instances.
Compatible Version	62 7.0
	Fully compabilite with 6.2 and earlier versions, such as 5.0,4.0, and 2.8.
CPU Type	x36
AZ	
	глен-ис издирутеля в технитичного рагонов стовы-ис им али влачат го-о в о.
Primary AZ	<b>az4</b> az2 <b>az3</b>
Standby AZ	224 <b>22 23</b>

Parameter	Example Value	Description
Billing mode description	Pay-per-use	<ul> <li>Billing mode of an instance</li> <li>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.</li> <li>Pay-per-use: A postpaid billing mode. Pay as you go and just pay for what you use. The DB instance usage is calculated by the second but billed every hour. 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.</li> </ul>
Region	Select <b>CN-Hong</b> <b>Kong</b> .	Region where a tenant is located <b>NOTICE</b> To reduce network latency, select a region nearest from which you will access the instance. Instances deployed in different regions cannot communicate with each other over a private network. After you buy an instance, you cannot change its region.
DB Instance Name	User-defined	<ul> <li>The instance name:</li> <li>Can be the same as an existing instance name.</li> <li>Can contain 4 to 64 characters and must start with a letter. It is case-sensitive and allows only letters, digits, hyphens (-), and underscores (_).</li> </ul>
Compatible API	Redis	GeminiDB is compatible with mainstream NoSQL APIs, including Redis, DynamoDB, Cassandra, HBase, MongoDB, and InfluxDB. You can select GeminiDB APIs by following <b>How Do I Select an API?</b>
Storage Type	Classic	Classic: classic architecture with decoupled storage and compute

Parameter	Example Value	Description
Product Type	Standard	• <b>Standard</b> : Stable and low-latency performance is provided for common scenarios such as advertising and recommendation, gaming, e-commerce, and Internet of Vehicles (IoV).
		• <b>Capacity-oriented</b> : Large-capacity key- value storage is suitable for average performance requirements and expectations of low costs.
DB Instance	Primary/Standby	Primary/Standby
Туре		A primary/standby instance is compatible with a standalone Redis node and Redis Sentinel. This instance type is used when hash tags are unavailable.
Compatible Version	6.2	7.0, 6.2 (including 6.2. <i>X</i> ), 5.0, and earlier versions
CPU Type	x86	x86 CPUs use the Complex Instruction Set Computing (CISC) instruction set. Each instruction can be used to execute low- level hardware operations. Executing these instructions is complex and time- consuming.
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. If there are multiple AZs, you need to select primary and standby AZs.
		Instances can be deployed in a single AZ or three AZs.
		<ul> <li>If low network latency is required, deploy your instance in one AZ.</li> </ul>
		• To meet disaster recovery requirements, select three AZs and specify primary and standby AZs.
		<ul> <li>Primary AZ: AZ where a primary node is located</li> </ul>
		<ul> <li>Standby AZ: AZ where a standby node is located</li> </ul>

IPv6 Net supported Net supported Net supported Net supported Net supported

Instance Creation Method	Past configure 81andard	contigure					
Instance Specifications	Storage	VCPUs	Nodes	ops op	Maximum Connections	Databases (default/maximu	Accounts
	16 GB	Standard 2 vCPUs	2	40,000	20,000	1,000	200
	24 GB	Standard 2 vCPUs	a	60,000	30,000	1,000	200
	32 GB	Standard 2 vCPUs	4	80,000	40,000	1,000	200
	48 GB	Standard 4 vCPUs	3	120,000	30,000	1,000	200
	O 64 GB	Standard 4 vCPUs	4	160,000	40,000	1,000	200
	96 GB	Standard 8 vCPUs	3	240,000	30,000	1,000	200
	128 GB	Standard 0 vCPUs	4	320,000	40,000	1,000	200
Specification Preview	Total capacity 16 GB   Node Specifications Standard 2 vCPUs   Nodes 2Count   QP5 40,010   Maximum Connections 20,000   Data copies 3						

#### Figure 3-19 Specifications and storage

descaling Storage Usage a 10% v Increase by 20% v Occur mobiled, in specy will be created. See the product documentationth team for the operand the measure being and a material with carriert percentations can be advantationally scaled up to 40.0%. To review the upper test, support specifications.

Parameter	Example Value	Description
Instance Creation	Fast configure	Two options are available:
Method		<ul> <li>Fast configure         Provides you with             recommended specifications.             You can select one of them             based on service             requirements, without the             need to specify the             specifications, node             quantity, and storage space.     </li> </ul>
		• Standard configure Provides a standard process to configure instance specifications, including specifying the specifications, node quantity, and storage space.
Instance Specifications	2U8GB	Higher CPU specifications provide better performance. Select specifications as needed. For details, see <b>1.6 Instance</b> Specifications.

#### Figure 3-20 Network and database configurations

VPC	(sch) v C have
Subert	
Security Group	and at C V V Security Draw The security of the
Database Port	LB/IF         View First Asset (0)           Use and RDS First were to use the indexes for disclosing (0).
Access Control	Cerlips Sa
Password	Codym 5a
Password	•
Confirm Password	•
Password-Free Access	Confuse Die
Enterprise Project	(-daa- ∨ ) ⊘ two hastlinequest Ø
Purchased Quantity -	(x +)

Parameter	Example Value	Description
VPC	default_vpc	<ul> <li>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.</li> <li>NOTE <ul> <li>After a GeminiDB Redis instance is created, its VPC cannot be changed.</li> <li>If you want to connect to a GeminiDB Redis instance through an ECS over an internal network, the GeminiDB Redis instance and the ECS must be in the same VPC. If they are not in the same VPC, you can create a VPC peering connection to enable access.</li> </ul> </li> </ul>
Subnet	default_subnet	A subnet provides dedicated network resources that are logically isolated from other networks for security purposes.
Database Port	6379	Port number for accessing a database. If you do not specify a port number, default port <b>6379</b> is used. You can specify a port number based on your requirements. The port number ranges from 1024 to 65535 except 2180, 2887, 3887, 6377, 6378, 6380, 8018, 8079, 8091, 8479, 8484, 8999, 12017, 12333, and 50069.
Access Control	Skip	<ul> <li>Skip: Access is restricted based on the VPC access policy by default.</li> <li>Configure: Specify how access is controlled. Three options are available:</li> <li>All IP addresses: All IP addresses can access the instance.</li> <li>Whitelist: Only IP addresses in a group can access the instance.</li> <li>Blacklist: IP addresses in a group cannot access the instance.</li> </ul>

Parameter	Example Value	Description
Password	Skip	• <b>Skip</b> : You can set the database password after creating an instance.
		• <b>Configure</b> : You can set the database password when creating an instance.
Password	Configured based on the password policy	If <b>Password</b> is set to <b>Configure</b> , you need to set the database password.
		• Must be 8 to 32 characters long.
		<ul> <li>Can contain at least two types of the following characters: uppercase letters, lowercase letters, digits, and special characters ~!@#%^*=+?</li> </ul>
		• For security reasons, set a strong password. The system will verify the password strength.
		Keep your password secure. The system cannot retrieve it if it is lost.
Password-Free Access	Skip	If you configure password-free access for a CIDR Block of an instance, you do not need to enter a password when connecting to the instance.
		For details, see <b>Table 4-5</b> .
project	default	This parameter is provided for enterprise users.
		An enterprise project groups cloud resources, so you can manage resources and members by project. The default project is <b>default</b> .
		Select an enterprise project from the drop-down list. For more information about enterprise projects, see <i>Enterprise</i> <i>Management User Guide</i> .

Retain the default values for other parameters.

5. On the order confirmation page, check the instance information. If you need to modify the information, click **Previous**. If no modification is required, read and agree to the service agreement and click **Submit**.

6. Click **Back to Instance Management** to go to the instance list.

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

• Creating an instance takes about 5 to 9 minutes. During the process, the instance status becomes **Creating**.

Primary/Standby Redis 6.2 Classic O Available IPv4: 192.168.0.128 Pay-per-use Log In Change Sp

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

 Name1D θ
 DB Instance Type
 Compatible API
 Storage Type
 Status θ
 Load balancer address
 Billing Mode
 Operation

Figure 3-21 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 the permission of remote login. DAS is secure and convenient for connecting to DB instances.

- 1. Log in to the Huawei Cloud console.
- 2. In the service list, choose **Databases** > **GeminiDB**.
- 3. In the instance list, locate the target instance and click **Log In** in the **Operation** column.

Figure 3-22 Logging in to a GeminiDB Redis instance

Name/ID 0	DB Instance	Compatible API	Stora	Status 🖯	Specifications	Load balance	Enterprise Pr	Billing Mode	Operation
geminidb d0d1a02543ae43cfa04cc05d9371	Primary/Standby	Redis 6.2	Classic	Available	2 vCPUs Standard 2 nodes	-	default	Pay-per-Use Created on Se	Log In Change to Yearly/Monthly More $\sim$

Alternatively, click the instance name to go to the **Basic Information** page. Click **Log In** in the upper right corner of the page.

Figure 3-23 Logging in to a GeminiDB Redis instance

<	🗸 🧿 Available			Feedback Quick Links Log In     View Metric
Basic Information				
Backups & Restorations	Basic Information			
Node Management	DB Instance Name	DB Instance ID	Storage Type	Product Type
Accounts			Classic	Standard
Slow Query Logs				
Audit Logs	Status	Region	AZ	DB Instance Type
Parameters	Available		az4,az2,az3	Primary/Standby
Metrics	Enterprise Project	Maintenance Window ①		
Sessions	default	10:00 - 14:00 Change		

4. Enter a password for logging in to the instance.

Figure 3-24 Logging in to a GeminiDB Redis instance

stance
Ø

If you need to log in again after the password is reset, click **Re-login** in the upper right corner and use the new password.

#### Figure 3-25 Re-login

Current Database	Re-login	Save to Executed Commands 🛞 👥
Bito V Decisit /R See (My Connection) Chier (Phil)		
1		
Executed Commands Messages Results		

5. Manage relevant databases.

#### Figure 3-26 Instance homepage

Current Database DB0   DB Instance Name: S	ave to Executed Commands	۲ 🕥
DB0 V Execute (73) Sere (My Commands) Clear (FH)		
1		
Executed Commands Messages Results		

- Save commands to the execution record.

This function is enabled by default to save the recently executed commands for your later query.

Then you can click the **Executed Commands** tab on the lower page to view historical commands.

Figure 3-27 Viewing executed commands

Executed Commands Messages Results			
Executed	Command	Time Required	Result
Jun 26, 2024 10:34:29 GMT+08:00	SCAN 0	2ms	Succeeded
Jun 26, 2024 10:33:52 GMT+08:00	SCAN 0	3ms	Succeeded

If this function is disabled, the commands executed subsequently are not displayed. You can click 
 next to Save Executed SQL Statements in the upper right corner to disable this function.

– Execute a command.

Enter a command in the command window and click **Execute** or **F8**.

**NOTE** 

- Do not use transactions, Lua scripts, Pub/Sub commands, or other commands that have blocking semantics.
- For an instance that supports multiple databases, you can change the current database on the console but cannot change it using a SELECT statement.

#### Figure 3-28 Executing a command

DB0 ~ E	Execute (F8)	Save My Commands Clear (F10)		
1 SCAN 0				
Executed Commands	Massages	Paquita		
Executed Commands	messages			
Command			Cursor	Result
SCAN 0			0	(empty list or set)

After a command is executed, you can view the execution result on the **Results** page.

– Save a command.

You can save a command to all instances, the current instance, or the current database. Then you can view details in **My Commands**.

#### Figure 3-29 Saving a command

ſ	DB0		Save				×
	1	SCAN 0	Title				
			Application Scope	<ul> <li>All instances</li> </ul>	<ul> <li>Current instance</li> </ul>	Current database	
						OK Cancel	)

– View my commands.

Common commands are displayed the **My Commands** page.

You can set a filter to narrow the scope of commands. If you select **All**, all commands saved in the current account are displayed.

#### Figure 3-30 Filtering commands

wy Com	manus					
Create	Command			All ^	Enter a title or command.	Q
No.	Title	Application Scope	Com	All	Operation	
1	scan	Current instance	scan	All instances	Edit Delete	Copy to Command Window
2	select	All instances	sele	Current Instance Current database	Edit Delete	Copy to Command Window

х

Alternatively, you can enter a command title or statement in the search box to search for the corresponding command.

Figure 3-31	Searching	for a	command
-------------	-----------	-------	---------

Create Command All ~ [4	X   Q
No. Title Application Scope Command Operation	
1 scan Current instance scan 0 Edit Delete Copy to Command	Window
2 select All instances select 1 Edit Delete Copy to Command	Window

On the **My Commands** page, you can also create, edit, and delete a command or copy it to the command window.

#### Figure 3-32 Managing a command

	My Comm	ands					
	Create Co	mmand		All V Enter a title or command.			
1	No.	Title	Application Scope	Command		Operation	
	1	scan	Current instance	scan O		Edit Delete	Copy to Command Window
	2	select	All instances	select 1	•	Edit Delete	Copy to Command Window

#### – Clear a command.

You can also press F10 to clear the command in the command window.

#### Figure 3-33 Clearing a command

DB0	
1	scan 0

#### 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.

# **3.4 Getting Started with Common Practices**

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

Table	3-4	Common	practices
-------	-----	--------	-----------

Refere	ence	Description
Data migr ation	4.4.5 Migrating Data from Redis to GeminiDB Redis API Using Redis- Shake	Describes how to migrate data from an on-premises Redis instance to a GeminiDB Redis instance using Redis-Shake.
	4.4.8 Migrating Data from Kvrocks to GeminiDB Redis API	Describes how to migrate data from a Kvrocks instance to a GeminiDB Redis instance using kvrocks2redis.
	4.4.9 Migrating Data from Pika to GeminiDB Redis API	Describes how to migrate data from a Pika instance to a GeminiDB Redis instance using pika-port.
	4.4.10 Migrating Data from SSDB to GeminiDB Redis API	Describes how to migrate data from an SSDB instance to a GeminiDB Redis instance using ssdb-port.
	4.4.11 Migrating Data from LevelDB to GeminiDB Redis API	Describes how to migrate data from a LevelDB instance to a GeminiDB Redis instance using leveldb-port.
	4.4.12 Migrating Data from RocksDB to GeminiDB Redis API	Describes how to migrate data from a RocksDB instance to a GeminiDB Redis instance using rocksdb-port.
Data back up	4.7.2 Managing Automated Backups	Describes how to enable automated backup so that GeminiDB Redis API can automatically create backups for a DB instance during a backup window and saves the backups based on the configured retention period.

Refere	ence	Description
	4.7.3 Managing Manual Backups	Describes how to manually create backups for a DB instance. These backups can be used to restore data for improved reliability.
Data resto ratio n	4.8.2 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.
Log man age ment	4.12.2 Viewing and Exporting Slow Query Logs	Describes how to view slow query logs of a GeminiDB Redis 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 Redis API

- 4.1 Using IAM to Grant Access to GeminiDB Redis API
- 4.2 Buying a GeminiDB Redis Instance
- 4.3 Instance Connection and Management
- 4.4 Data Migration
- 4.5 Instance Management
- 4.6 Modifying Instance Settings
- 4.7 Data Backup
- 4.8 Data Restoration
- 4.9 Diagnosis Analysis
- 4.10 Account and security
- 4.11 Parameter Management
- 4.12 Logs and Audit
- 4.13 Viewing Metrics and Configuring Alarms
- 4.14 Tag Management
- 4.15 Managing User Resource Quotas of a GeminiDB Redis Instance
- 4.16 Memory Acceleration

# 4.1 Using IAM to Grant Access to GeminiDB Redis API

# 4.1.1 Creating a User and Granting GeminiDB Redis API 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 supported by GeminiDB and choose policies or roles based on your requirements. For details about the permissions, see **Permissions Management**. For system policies of other services, see **Permissions 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 **GaussDB 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 the user's 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 Custom Policies of GeminiDB Redis API

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



# 4.2 Buying a GeminiDB Redis Instance

### 4.2.1 Buying a GeminiDB Redis Cluster Instance

This section describes how to buy a GeminiDB Redis cluster instance on the GeminiDB console.

- In a sharded cluster, a proxy cluster GeminiDB Redis instance is connected through proxies to a standalone Redis instance, Redis Sentinel, and Redis Cluster. The proxy cluster instance has strong horizontal scaling capabilities and can handle millions of QPS and dozens of terabytes of data.
- With the native Redis Cluster architecture, a Redis Cluster GeminiDB Redis instance is directly connected to Redis Cluster. It can greatly reduce latency while improving performance.

Each tenant can create a maximum of 50 GeminiDB Redis 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 (proxy cluster)

#### Figure 4-3 Billing mode and basic information (Redis Cluster)

Billing Mode	Раурания
Region	compared and a programming and a set of the set of the relation of the relation of the set of the relation of the set of the relation
DB Instance Name	evented=0 Types log multiple DB instances, the system automatically appends a date, time, and antial number to the end of the instance numee (lonnet instance, name MAGD Himmes SR).
Compatible API	Rede Cassandra DynamoDB HBase InflutDB MongoDB
Storage Type	Create Cloud raflwe
Product Type	Chanced     Capacity oriented     This type provides stable and toe-latency performance. It is good for advertising and ecommendations, paming, e-commence, and Internet of Vehicles (sV).
DB Instance Type	Prev duble Preve duble Preve duble Preve state and the state of the st
Compatible Version	4.2 7.0 Fully computed with 6.2 and earlier ventors, such as 5.8,4.0, and 2.8.
CPU Type	x68
AZ	actual2.00 82 83 85

Parameter	Description
Billing Mode	Select Yearly/Monthly or Pay-per-use.
	Yearly/Monthly
	<ul> <li>Specify Required Duration. The system deducts fees from your account based on the service price.</li> </ul>
	<ul> <li>If you do not need such an instance any longer after it expires, change the billing mode to pay-per-use. For details, see 2.5.3 Changing a Yearly/Monthly Instance to Pay-per-Use.</li> </ul>
	NOTE
	Yearly/Monthly instances cannot be deleted directly. If such an instance is no longer required, unsubscribe from it. For details, see 2.11.4 How Do I Unsubscribe from a Yearly/ Monthly Instance?.
	<ul> <li>Yearly/Monthly instances with cloud native storage are now in OBT. To use such an instance, choose Service Tickets &gt; Create Service Ticket in the upper right corner of the console and contact the customer service.</li> </ul>
	Pay-per-use
	<ul> <li>If you select this billing mode, you are billed based on how much time the instance is in use.</li> </ul>
	<ul> <li>To use an instance for a long time, change its billing mode to yearly/monthly to reduce costs. For details, see 2.5.2 Changing a Pay-per-Use Instance to Yearly/ Monthly.</li> </ul>

#### Table 4-1 Billing mode description

#### Table 4-2 Basic information

Parameter	Description
Region	Region where a tenant is located <b>NOTICE</b> 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	<ul> <li>The instance name:</li> <li>Can be the same as an existing instance name.</li> <li>Can contain 4 to 64 characters and must start with a letter. It is case-sensitive and allows only letters, digits, hyphens (-), and underscores (_).</li> <li>You can change the name of an instance after it is created. For details, see 4.6.2 Modifying a GeminiDB Redis Instance Name.</li> </ul>

Parameter	Description
Compatible API	<b>Redis</b> GeminiDB is compatible with mainstream NoSQL APIs, including Redis, DynamoDB, Cassandra, HBase, MongoDB, and InfluxDB. You can select GeminiDB APIs by following <b>How Do I Select an</b> <b>API?</b>
Storage Type	<ul> <li>Classic: classic architecture with decoupled storage and compute</li> <li>Cloud native: more flexible, new-gen version with support for more AZs</li> <li>NOTE         <ul> <li>Cloud native storage supports only proxy cluster instances.</li> <li>Classic and cloud native are different deployment modes. Cloud native supports more AZs. If both classic and cloud native are supported, you can select any of them.</li> </ul> </li> </ul>
Product Type	<ul> <li>Standard: Stable and low-latency performance is provided for common scenarios such as advertising and recommendation, gaming, e-commerce, and Internet of Vehicles (IoV).</li> <li>Capacity-oriented: Large-capacity key-value storage is suitable for average performance requirements and expectations of low costs.</li> <li>NOTE Standard is unavailable for instances with cloud native storage.</li> </ul>
DB Instance Type	<ul> <li>Proxy cluster: In a sharded cluster, a proxy cluster GeminiDB Redis instance is connected through proxies to a standalone Redis instance, Redis Sentinel, and Redis Cluster. The proxy cluster instance has strong horizontal scaling capabilities and can handle millions of QPS and dozens of terabytes of data.</li> <li>RedisCluster With the native Redis Cluster architecture, a Redis Cluster GeminiDB Redis instance is directly connected to Redis Cluster. It can greatly reduce latency while improving performance.</li> </ul>
Compatible Version	7.0, 6.2 (including 6.2.X), 5.0, and earlier versions
СРИ Туре	x86 x86 CPUs use the Complex Instruction Set Computing (CISC) instruction set. Each instruction can be used to execute low-level hardware operations. Executing these instructions is complex and time-consuming.

Parameter	Description
AZ	Availability zone where the instance is created. An AZ is a part of a region with its own independent power supplies and networks. AZs are physically isolated but can communicate with each other over a private network.
	Instances can be deployed in a single AZ or three AZs.
	• If low network latency is required, deploy your instance in one AZ.
	• If disaster recovery is required, select three AZs, and nodes of your instance will be evenly distributed across the three AZs.

#### Figure 4-4 Storage and specifications (standard)

Instance Creation Method	Fast contigure	Standard contigure							
Instance Specifications	Storage		CPU   Memory	Nodes	QPS 💮	Maximum Connections	Databases (default/maxim	um) Accounts	
	O 4 GB		Special offer 1 vCPU   2 GB	2	16,000	20,000	1,000	200	
	8 GB		Standard 1 vCPU   4 GB	2	20,000	20,000	1,000	200	
	🔿 16 GB		Standard 2 vCPUs   8 GB	2	40,000	20,000	1,000	200	
	🔿 24 GB		Standard 2 vCPUs   8 GB	3	60,000	30,000	1,000	200	
	🔿 32 GB		Standard 2 vCPUs   8 GB	4	80,000	40,000	1,000	200	
	🔿 48 GB		Standard 4 vCPUs   16 GB	3	120,000	30,000	1,000	200	
	🔿 64 GB		Standard 4 vCPUs   16 GB	4	160,000	40,000	1,000	200	
Specification Preview	Node Specifications Standard 1vCPU 4 GB   Modes 2Court   Memory II GB   Storage 8 GB   QPS 20,000   Maximum Connections 20,000   Data copies 3 Data is shared to data in real time, so I frave is more storage than memory, much more data can be handled. The maximum size of an instance depends on the storage, not the memory. You can specify the memory and storage.								

#### Figure 4-5 Storage and specifications (cloud native)

Instance Specifications	CPU   Memory	CPU   Memory			de	Databases (default/maximum)	Accounts
	1 vCPU   4 GB	1 vCPU   4 GB				1,000	200
	2 vCPUs   8 GB			10,000		1,000	200
	4 vCPUs   16 GB			10,000		1,000	200
	O 8 vCPUs   32 GB			10,000		1,000	200
	0 16 vCPUs   64 GB			10,000		1,000	200
	32 vCPUs   128 GB			10,000		1,000	200
	Currently selected 1 vCPU						
Nodes	- 2 + The	quantity ranges from 2 to 12.					
Total Storage Space	20 GB				- 20	+ 08	
	10	100	190	280	400		
Spacification Province	Instance Specifications 1 vC	PUs 4 GB   Nodes 2Count   N	lemory 8 GB   Storage 20	GB   QP\$ 20,000   Maximum Conn	ctions 20,000   Data copies 3		
Parameter	Description						
--------------------------------	--						
Instance Creation Method	<ul> <li>Two options are available:</li> <li>Fast configure Recommended specifications, node quantity, and storage space</li> <li>NOTE <ul> <li>Instance specifications with the memory of 8 GB and 16 GB are available only in single AZs. The console shows available specifications.</li> <li>The QPS is only for reference.</li> </ul> </li> <li>Standard configure <ul> <li>Instance flavor, node specifications, node quantity, and storage space that can be specified</li> </ul> </li> </ul>						
Instance Specifications	You need to specify instance specifications after selecting <b>Fast</b> <b>configure</b> for <b>Instance Creation Method</b> . Higher CPU specifications provide better performance. Select specifications as needed. For details, see <b>1.6 Instance Specifications</b> .						
Specification Type	<ul> <li>You need to select a specification type after selecting Standard configure for Instance Creation Method.</li> <li>Standard: The default and recommended CPU-to-memory ratio is 1:4, which balances low latency demands with high concurrency requirements.</li> <li>Enhanced: The CPU-to-memory ratio is 1:8, which boosts the access hit rate while reducing latency.</li> </ul>						
Node Specifications	You need to select node specifications after selecting <b>Standard</b> <b>configure</b> for <b>Instance Creation Method</b> and <b>Classic</b> for <b>Storage Type</b> . For details, see <b>1.6 Instance Specifications</b> .						
Nodes	You need to specify the node quantity after selecting <b>Standard</b> <b>configure</b> for <b>Instance Creation Method</b> and <b>Classic</b> for <b>Storage Type</b> . Number of required nodes. After an instance is created, you can add nodes. Currently, a maximum of 36 nodes are supported. To add more, choose <b>Service Tickets &gt; Create Service Ticket</b> in the upper right corner of the console and contact the customer service.						
Shard Specifications	You need to select shard specifications after selecting <b>Standard</b> <b>configure</b> for <b>Instance Creation Method</b> and <b>Cloud native</b> for <b>Storage Type</b> . For details, see nosql_05_0059.xml#nosql_05_0059/ table16744444102213.						

## Table 4-3 Specifications and storage

Parameter	Description
Shards	You need to specify the shard quantity after selecting <b>Standard</b> <b>configure</b> for <b>Instance Creation Method</b> and <b>Cloud native</b> for <b>Storage Type</b> .
	You can add shards after creating an instance.
	Currently, a maximum of 12 shards are supported. To add more, choose <b>Service Tickets &gt; Create Service Ticket</b> in the upper right corner of the console and contact the customer service.
Total Storage Space	You need to specify the storage space after selecting <b>Standard configure</b> for <b>Instance Creation Method</b> .
	Higher CPU specifications provide better performance. Select specifications as needed.
	For details, see <b>1.6 Instance Specifications</b> .
Specification Preview	After you select instance specifications, the system automatically shows details of the total capacity, node specifications, number of nodes, QPS benchmark, total number of connections, and number of data copies. This helps keep track of the selected instance specifications.
Auto Scale	You can determine whether to enable the function based on the site requirements.
	• <b>Trigger If Available Storage Drops To</b> : If the storage usage exceeds the specified value, autoscaling will be triggered. The value can be <b>60%</b> , <b>65%</b> , <b>70%</b> , <b>75%</b> , <b>80%</b> , <b>85%</b> , and <b>90%</b> .
	• Increase By: percentage that your instance storage will be scaled up at. The value can be 10%, 15%, or 20%.
Static Data	You can determine whether to encrypt static data.
Encryption	Disable: Data is not encrypted.
	• 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 plaintext and then sent to you. Disk encryption can improve data security and may have slight impacts on database writes and reads. Key Name: Select an existing key or create one.
	NOTE
	<ul> <li>This function is now in OBT. To use it, choose Service Tickets &gt; Create Service Ticket in the upper right corner of the console and contact the customer service.</li> </ul>
	- An agency will be created after static data encryption is enabled.
	<ul> <li>After an instance is created, the static data encryption status and the key cannot be changed.</li> </ul>
	<ul> <li>The key cannot be disabled, deleted, or frozen when it is in use.</li> <li>Otherwise, the database becomes unavailable.</li> </ul>
	<ul> <li>For details about how to create a key, see "Creating a Key" in Data Encryption Workshop User Guide.</li> </ul>

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.
	For details about how to create a VPC, see "Creating a VPC" in <i>Virtual Private Cloud User Guide</i> .
	With VPC sharing, you can also use a VPC and subnet shared by another account.
	VPC owners can share the subnets in a VPC with one or multiple accounts through Resource Access Manager (RAM), which ensures cost efficiency of network resources.
	For more information about VPC subnet sharing, see VPC Sharing in Virtual Private Cloud User Guide.
	If there are no VPCs available, the system allocates resources to you by default.
	<ul> <li>After a GeminiDB Redis instance is created, its VPC cannot be changed.</li> </ul>
	• If you want to connect to a GeminiDB Redis instance through an ECS over an internal network, the GeminiDB Redis instance and the ECS must be in the same VPC. If they are not in the same VPC, you can create a VPC peering connection to enable access.
Subnet	A subnet where your instance is created. The subnet provides dedicated and isolated networks, improving network security.
	An IPv6 subnet cannot be associated with your instance. Select an IPv4 subnet.
Security Group	A security group controls access between GeminiDB Redis instances and other services. Ensure that the security group you selected allows your client to access the instance.
	If no security group is available, the system creates one for you.
Database Port	Database port number.
	You can specify a port number based on your requirements. The port number ranges from 1024 to 65535 except 2180, 2887, 3887, 6377, 6378, 6380, 8018, 8079, 8091, 8479, 8484, 8999, 12017, 12333, and 50069.
	If you do not specify a port number, port 6379 is used by default.

Parameter	Description
Access Control	Skip
	• <b>Skip</b> : Access is restricted based on the VPC access policy by default.
	Configure:
	Specify how access is controlled. Three options are available:
	All IP addresses: All IP addresses can access the instance.
	<b>Whitelist</b> : Only IP addresses in a group can access the instance.
	<b>Blacklist</b> : IP addresses in a group cannot access the instance.
	<b>NOTE</b> Redis Cluster GeminiDB Redis instances do not support access control.

Table 4-5 Database	e configuration
--------------------	-----------------

Parameter	Description
Password	• <b>Skip</b> : You can set the database password after creating an instance.
	• <b>Configure</b> : You can set the database password when creating an instance.
	<b>NOTE</b> You cannot set a password after creating a Redis Cluster GeminiDB Redis instance.
Password	Password of database administrator <b>rwuser</b> :
	Must be 8 to 32 characters long.
	• Can contain at least two types of the following characters: uppercase letters, lowercase letters, digits, and special characters ~!@#%^*=+?
	• For security reasons, set a strong password. The system will verify the password strength.
	Keep your password secure. The system cannot retrieve it if it is lost.
Confirm Password	Enter the database password again.

Parameter	Description
Password-Free Access	Configure password-free access for a CIDR Block of the instance you want to access. After that, the password is not required the instance access.
	• Skip If you select Skip, you can set password-free access after the GeminiDB Redis instance is created. For details, see 4.10.1 Enabling Password-Free Access.
	• <b>Configure</b> Enter a CIDR block that you want to enable password-free access for. A maximum of 30 password-free CIDR blocks can be configured.
	NOTE Redis Cluster GeminiDB Redis instances do not support password- free access.

## Table 4-6 Enterprise project

Parameter	Description
Enterprise Project	This parameter is provided for enterprise users. An enterprise project groups cloud resources, so you can manage resources and members by project. The default project is <b>default</b> .
	Select an enterprise project from the drop-down list. For more information about enterprise projects, see <i>Enterprise Management User Guide</i> .

## Table 4-7 Advanced settings

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.
	You can enable SSL to improve data security. After an instance is created, you can connect to it using <b>SSL</b> .
	NOTE
	• If SSL is not enabled when you create an instance, you can enable it after the instance is created. For details, see <b>4.3.5.5 Encrypting Data over SSL for a GeminiDB Redis Instance</b> .
	Redis Cluster GeminiDB Redis instances do not support SSL.

Parameter	Description
Parameter Template	You can use parameters in a parameter template to manage database API configurations. A database parameter template acts as a container for API configuration values that can be applied to one or more instances.
	Each user can create up to 100 parameter templates. All types of instances in the same project can share the quota. For details, see <b>4.11.2 Creating a Parameter Template</b> .
Tags	This setting is optional. Adding tags helps you better identify and manage your instances. A maximum of 20 tags can be added for each instance.
	If your organization has configured a tag policy for your GeminiDB Redis instance, 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.
	A tag consists of a tag key and a tag value.
	<ul> <li>A tag key is mandatory if the instance is going to be tagged.</li> <li>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:@.:/+=</li> </ul>
	• A tag value is optional if the instance is going to be tagged. The value can be empty.
	The value can contain a maximum of 255 characters. Only letters, digits, spaces, and the following special characters are allowed::+=@/
	After an instance is created, you can view its tag details on the <b>Tags</b> tab. In addition, you can add, modify, and delete tags of an existing instance. For details, see <b>4.14 Tag Management</b> .

## Table 4-8 Required duration

Parameter	Description
Required Duration	The length of your subscription if you select <b>Yearly/Monthly</b> billing. Subscription lengths range from one month to three years.
Auto-renew	<ul> <li>By default, this option is not selected.</li> <li>If you select this parameter, the auto-renew cycle is determined by the selected required duration.</li> </ul>

**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.

The instance creation process takes about 5 to 15 minutes. After the creation is complete, the status changes to **Available**.

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

----End

## 4.2.2 Buying a Primary/Standby GeminiDB Redis Instance

This section describes how to buy a primary/standby Redis instance on the GeminiDB console.

Each tenant can create a maximum of 50 GeminiDB Redis 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, select a billing mode, select instance specifications and click **Next**.

Billing Mode	YearlyMonthly Pay per ease
Region	Comparison are geographic areas included from each other. For low notwork labercy and quick resource access, soled the nearest region.
DB Instance Name	pressure © Fyrea by matrices. It is system advantacially apports a date, time, and senial number to the end of the instance names (format instance, name MMCD-HHmmes SH).
Compatible API	Redis Cassandra DynamoDB HBase InfluiDB MongoDB
Storage Type	Chance Cloud rathe
Product Type	Standard Capacity oriented This type provides stable and low-latency performance. It is good for advertising and recommendations, gaming, a-commerce, and Internet of Vehicles (MV)
DB Instance Type	Procycluster Restrictioner Processing O Primary/standby instances with 4 GB of memory fees for a limited time VMM advances primary/standby advances. We have a second relative second relative second relative second relative Vm advances and Restrictioner.
Compatible Version	7.0 Fully computative with 6.2 and earlier versions, such as 5.0.40, and 2.0.
CPU Type	and the second se
AZ	<u>කරු කට, කට</u> කට කට
	Three-AZ deployment is recommended to provide cross-AZ DR and ensure RPO is 0.
Primary AZ	az4 az2 az3
Standby AZ	2014 <b>2021</b> 2021

## Figure 4-6 Billing mode and basic information

## Table 4-9 Billing mode description

Parameter	Description	
Billing Mode	Select Yearly/Monthly or Pay-per-use. <ul> <li>Yearly/Monthly</li> </ul>	
	<ul> <li>Specify Required Duration. The system deducts fees from your account based on the service price.</li> </ul>	
	<ul> <li>If you do not need such an instance any longer after it expires, change the billing mode to pay-per-use. For details, see 2.5.3 Changing a Yearly/Monthly Instance to Pay-per-Use.</li> </ul>	
	NOTE Yearly/Monthly instances cannot be deleted directly. If such an instance is no longer required, unsubscribe from it. For details, see 2.11.4 How Do I Unsubscribe from a Yearly/Monthly Instance?.	
	Pay-per-use	
	<ul> <li>If you select this billing mode, you are billed based on how much time the instance is in use.</li> </ul>	
	<ul> <li>To use an instance for a long time, change its billing mode to yearly/monthly to reduce costs. For details, see 2.5.2 Changing a Pay-per-Use Instance to Yearly/ Monthly.</li> </ul>	

Table 4-10	<b>Basic information</b>
------------	--------------------------

Parameter	Description
Region	Region where a tenant is located <b>NOTICE</b> 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	<ul> <li>The instance name:</li> <li>Can be the same as an existing instance name.</li> <li>Can contain 4 to 64 characters and must start with a letter. It is case-sensitive and allows only letters, digits, hyphens (-), and underscores (_).</li> <li>You can change the name of an instance after it is created.</li> <li>For details, see 4.6.2 Modifying a GeminiDB Redis Instance Name.</li> </ul>
Compatible API	<b>Redis</b> GeminiDB is compatible with mainstream NoSQL APIs, including Redis, DynamoDB, Cassandra, HBase, MongoDB, and InfluxDB. You can select GeminiDB APIs by following <b>How Do I Select an API?</b>
Storage Type	<b>Classic</b> : classic architecture with decoupled storage and compute
Product Type	• <b>Standard</b> : Stable and low-latency performance is provided for common scenarios such as advertising and recommendation, gaming, e-commerce, and Internet of Vehicles (IoV).
DB Instance Type	<b>Primary/Standby</b> A primary/standby instance is compatible with a standalone Redis node and Redis Sentinel. This instance type is used when hashtags are unavailable.
Compatible Version	7.0, 6.2 (including 6.2.X), 5.0, and earlier versions
СРU Туре	<b>x86</b> x86 CPUs use the Complex Instruction Set Computing (CISC) instruction set. Each instruction can be used to execute low- level hardware operations. Executing these instructions is complex and time-consuming.

Parameter	Description
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. If there are multiple AZs, you need to select primary and standby AZs. Instances can be deployed in a single AZ or three AZs.
	• If low network latency is required, deploy your instance in one AZ.
	• To meet disaster recovery requirements, select three AZs and specify primary and standby AZs.
	<ul> <li>Primary AZ: AZ where a primary node is located</li> <li>Standby AZ: AZ where a standby node is located</li> </ul>
	- Standby AZ: AZ where a standby hode is located

## Figure 4-7 Specifications and storage

Instance Creation Method	Fast configure Standard conf	gure				
Instance Specifications	Storage	Nodes	QPS (D)	Maximum Connections	Databases (default/maximum)	Accounts
	○ 4 GB	1	8,000	10,000	1,000	200
	○ 8 GB	1	8,000	10,000	1,000	200
	16 GB	1	10,000	10,000	1,000	200
	🔿 24 GB	1	20,000	10,000	1,000	200
	🔿 32 GB	1	20,000	10,000	1,000	200
	○ 48 GB	1	40,000	20,000	1,000	200
	🔿 64 GB	1	40,000	20,000	1,000	200
Specification Preview	Node Specifications Standard 1 vCPU 4	iB   Nodes 2Count   Memory 4 GB   Sto	rage 16 GB   QPS 10,000   Maximum Connection	s 10,000   Data copies 3	a motile the memory and elemen	

Table 4-11	Specifications	and storage
------------	----------------	-------------

Parameter	Description			
Instance Creation Method	<ul> <li>Two options are available:</li> <li>Fast configure         Provides you with recommended specifications. You can select one of them based on service requirements, without the need to specify the specifications, node quantity, and storage space.     </li> <li>NOTE         <ul> <li>The QPS is only for reference.</li> </ul> </li> <li>Standard configure         <ul> <li>Provides a standard process to configure instance specifications, including specifying the specifications, node quantity, and storage space.</li> </ul> </li> </ul>			
Instance Specifications	You need to specify instance specifications after selecting <b>Fast</b> <b>configure</b> for <b>Instance Creation Method</b> .			
	specifications as needed.			
	For details, see <b>1.6 Instance Specifications</b> .			

Parameter	Description		
Specification Type	<ul> <li>You need to select a specification type after selecting Standard configure for Instance Creation Method.</li> <li>Standard: The default and recommended CPU-to-memory ratio is 1:4, which balances low latency demands with high concurrency requirements.</li> </ul>		
Node Specifications	You need to select node specifications after selecting <b>Standard</b> <b>configure</b> for <b>Instance Creation Method</b> . For details, see <b>1.6 Instance Specifications</b> .		
Nodes	The default value is <b>2</b> . One is the primary node and the other is standby. If there is a fault, the primary and standby nodes can automatically switch over.		
Total Storage Space	You need to specify the storage space after selecting <b>Standard</b> <b>configure</b> for <b>Instance Creation Method</b> . Higher CPU specifications provide better performance. Select specifications as needed. For details, see <b>1.6 Instance Specifications</b> .		
Specification Preview	After you select instance specifications, the system automatically shows details of the total capacity, node specifications, number of nodes, QPS benchmark, total number of connections, and number of data copies. This helps keep track of the selected instance specifications.		
Autoscaling	<ul> <li>You can determine whether to enable the function based on the site requirements.</li> <li>Trigger If Available Storage Drops To: If the storage usage exceeds the specified value, autoscaling will be triggered. The value can be 60%, 65%, 70%, 75%, 80%, 85%, and 90%.</li> <li>Increase By: percentage that your instance storage will be</li> </ul>		
	scaled up at. The value can be <b>10%</b> , <b>15%</b> , or <b>20%</b> .		

Parameter	Description		
Static Data	You can determine whether to encrypt static data.		
Lincipption	• <b>Disable</b> : Data is not encrypted.		
	• 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. Key Name: Select an existing key or create one.		
	NOTE		
	<ul> <li>This function is now in OBT. To use it, choose Service Tickets &gt; Create Service Ticket in the upper right corner of the console and contact the customer service.</li> </ul>		
	– An agency will be created after static data encryption is enabled.		
	<ul> <li>After an instance is created, the static data encryption status and the key cannot be changed.</li> </ul>		
	<ul> <li>The key cannot be disabled, deleted, or frozen when it is in use.</li> <li>Otherwise, the database becomes unavailable.</li> </ul>		
	<ul> <li>For details about how to create a key, see "Creating a Key" in Data Encryption Workshop User Guide.</li> </ul>		

## Figure 4-8 Network and database configurations

VPC	ec.64 v C WerVIC
	After GeneralD instance is unable, the VPC-when the ECI is cogated, a condition is unable, the VPC, the GeneralD instance relets is communicatively built (C) is a product interval in the CO is cogated, a condition is unable of the VPC on the
Subset	name v C the factor of the control o
danak dana	
38.219 0.00	Leader and a contraction of and Leader and pages of the Alley are ward in the contraction of the Contraction
Database Port	ST Werentas 0
Access Cantral	ca jor mu n je vara tu an en monor te ala active uk. Confuse 19
Password	Cedapa Dia
Password	•
Confirm Password	
Password-Pree Access	Certyex Dia
Enterprise Project	L-Sted- V C Ver-Fuel-Management @
Purchased Quantity -	1 +

#### Table 4-12 Network

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.
	For details about how to create a VPC, see "Creating a VPC" in <i>Virtual Private Cloud User Guide</i> .
	With VPC sharing, you can also use a VPC and subnet shared by another account.
	VPC owners can share the subnets in a VPC with one or multiple accounts through Resource Access Manager (RAM), which ensures cost efficiency of network resources.
	For more information about VPC subnet sharing, see VPC Sharing in Virtual Private Cloud User Guide.
	If there are no VPCs available, the system allocates resources to you by default.
	• After an instance is created, the VPC where the instance is deployed cannot be changed.
	• 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.
Subnet	A subnet where your instance is created. The subnet provides dedicated and isolated networks, improving network security.
	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. Ensure that the security group you selected allows your client to access the instance.
	If no security group is available, the system creates one for you.
Database Port	Database port number.
	You can specify a port number based on your requirements. The port number ranges from 1024 to 65535 except 2180, 2887, 3887, 6377, 6378, 6380, 8018, 8079, 8091, 8479, 8484, 8999, 12017, 12333, and 50069.
	If you do not specify a port number, port 6379 is used by default.

Parameter	Description				
Access Control	Skip				
	• <b>Skip</b> : Access is restricted based on the VPC access policy by default.				
	Configure:				
	Specify how access is controlled. Three options are available:				
	All IP addresses: All IP addresses can access the instance.				
	<b>Whitelist</b> : Only IP addresses in a group can access the instance.				
	<b>Blacklist</b> : IP addresses in a group cannot access the instance.				

Table 4-13 Databa	ase configuration
-------------------	-------------------

Parameter	Description				
Password	• <b>Skip</b> : You can set the database password after creating an instance.				
	• <b>Configure</b> : You can set the database password when creating an instance.				
Password	Password of database administrator rwuser:				
	Must be 8 to 32 characters long.				
	<ul> <li>Can contain at least two types of the following characters: uppercase letters, lowercase letters, digits, and special characters ~!@#\$%^&amp;*()=+?\$()&amp;</li> </ul>				
	• For security reasons, set a strong password. The system will verify the password strength.				
	Keep your password secure. The system cannot retrieve it if it is lost.				
Confirm Password	Enter the administrator password again.				
Password-Free Access	Configure password-free access for a CIDR Block of the instance you want to access. After that, the password is not required the instance access.				
	<ul> <li>Skip If you select Skip, you can set password-free access after the GeminiDB Redis instance is created. For details, see 4.10.1 Enabling Password-Free Access.</li> </ul>				
	• <b>Configure</b> Enter a CIDR block that you want to enable password-free access for. A maximum of 30 CIDR blocks can be configured.				

Parameter	Description
Enterprise project	This parameter is provided for enterprise users. An enterprise project groups cloud resources, so you can manage resources and members by project. The default project is <b>default</b> .
	Select an enterprise project from the drop-down list. For more information about enterprise projects, see <i>Enterprise Management User Guide</i> .

## Table 4-14 Advanced settings

Parameter	Description
Static Data Encryption	<ul> <li>Data is encrypted when stored.</li> <li>Disable</li> <li>Enable Static data encryption improves security but slightly affects database I/O performance. An agency will be created after disk encryption is enabled.</li> </ul>
Кеу	<ul> <li>Set when Enable is selected for Static Data Encryption.</li> <li>Select: Select a key name from the drop-down list.</li> <li>Enter: Enter a key ID. The ID cannot be empty, and your KMS key must be in the current region.</li> </ul>
Key Name	You can select an existing key or create a key. The key cannot be disabled, deleted, or frozen when used, or the database becomes unavailable.
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. You can enable SSL to improve data security. After an instance is created, you can connect to it using <b>SSL</b> . <b>NOTE</b> If SSL is not enabled when you create an instance, you can enable it after the instance is created. For details, see 4.3.5.5 Encrypting Data
Rename High- risk Command	To prevent data loss, instance restart, and performance jitter caused by misoperations, you can rename high-risk commands of a GeminiDB Redis instance. For details, see <b>4.5.2 Renaming Commands of a GeminiDB Redis Instance</b> .

Parameter	Description			
Parameter Template	You can use parameters in a parameter template to manage database API configurations. A database parameter template acts as a container for API configuration values that can be applied to one or more instances.			
	Each user can create up to 100 parameter templates. All types of instances in the same project can share the quota. For details, see <b>4.11.2 Creating a Parameter Template</b> .			
Maintenance Window	<ul> <li>Time range of change tasks that affect instances, such as specifications change and patch upgrade.</li> <li>Skip</li> <li>Configure</li> </ul>			
	• Configure			
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.			
	If your organization has configured a tag policy for GeminiDB Redis, 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.			
	A tag consists of a tag key and a tag value.			
	• A tag key is mandatory if the instance is going to be tagged.			
	Each tag key is unique for each instance. It can contain 1 to 128 characters, cannot start with <b>_sys_</b> , and cannot start or end with a space. Only letters, digits, spaces, and the following special characters are allowed:@.:/+=			
	• A tag value is optional if the instance is going to be tagged. The value can be empty.			
	The value can contain a maximum of 255 characters. Only letters, digits, spaces, and the following special characters are allowed::+=@/			
	After an instance is created, you can view its tag details on the <b>Tags</b> tab. In addition, you can add, modify, and delete tags of an existing instance. For details, see <b>4.14 Tag Management</b> .			

## Table 4-15 Required duration

Parameter	Description
Required duration	The length of your subscription if you select <b>Yearly/Monthly</b> billing. Subscription lengths range from one month to three years.

Parameter	Description
Auto-renewing an Instance	<ul><li>By default, this option is not selected.</li><li>If you select this option, the auto-renew cycle is determined by the selected required duration.</li></ul>

**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.

The instance creation process takes about 5 to 15 minutes. After the creation is complete, the status changes to **Available**.

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

----End

## **4.3 Instance Connection and Management**

## 4.3.1 Connecting to a GeminiDB Redis Instance

GeminiDB Redis API is compatible with open-source Redis and allows traffic from applications using different types of SDKs. It can also be accessed through Data Admin Service (DAS), private networks, and public networks.

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

#### Figure 4-9 Connection Methods



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

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

#### Table 4-16 Connection methods

Metho d	Scenario	De fau lt Por t	Description
DAS	You can connect to a GeminiDB Redis instance using a web- based console.	-	-

Metho d	Scenario	De fau lt Por t	Description
Private networ k	You can connect to a GeminiDB Redis instance through a <b>private IP address</b> , <b>private domain name</b> , or <b>load balancer</b> <b>address</b> . This method is suitable when your application is deployed on an ECS that is in the same region and VPC as your instance.	63 79	<ul> <li>You are advised to use the load balancer address to connect to the instance. This ensures high reliability and eliminates the impact of SPOFs.</li> <li>High security and performance</li> <li>If the ECS and GeminiDB Redis 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.</li> <li>Configure inbound rules of a security group for the GeminiDB Redis instance by following 4.3.5.1 Setting Security Group Rules for a GeminiDB Redis 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>
Public networ k	You can connect to a GeminiDB Redis instance through a <b>public domain name</b> or an <b>EIP</b> . This method is suitable when an instance cannot be accessed over a private network. You can connect to the instance from an ECS using a public domain name or an EIP.	63 79	<ul> <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>Use a public domain name to ensure high reliability and eliminate SPOFs.</li> <li>For EIP pricing details, see EIP billing details.</li> <li>NOTE Redis Cluster GeminiDB Redis instances cannot be accessed over a public network.</li> </ul>

Metho d	Scenario	De fau lt Por t	Description
Progra m code	You can connect to a GeminiDB Redis instance using different code. For details, see <b>5.3 Examples of</b> <b>Connecting to an</b> <b>Instance Using</b> <b>Programming</b> <b>Languages</b> .	63 79	-

# 4.3.2 Connecting to a GeminiDB Redis Instance on the DAS Console

DAS makes DB instance management secure and efficient from a web-based console. By default, you have permissions required for remote login. It is recommended to use DAS for connecting to your instance.

## Usage Notes

• If SSL is enabled, you cannot connect to a GeminiDB Redis instance through DAS.

## **Configuring the Required Permissions**

If you have an IAM account, assign DAS FullAccess permissions to all users of the account. For details, see **Create User Groups and Assign Permissions**.

You can create a custom policy to specify the type of databases that you have permissions for.

1. Log in to the IAM console and choose **Permissions** > **Policies/Roles**.

#### Figure 4-10 Creating a custom policy

IAM	Policies/Roles ()			Tresback     Create Gadon Policy
Users				
User Groups	Delete Duators policies available for creation: 200			
Permissiona ^	All policies/toles v All services	<ul> <li>Fuzzy search</li> </ul>	<ul> <li>C, Enter a policy name, role name, or description.</li> </ul>	
Authorization	Policy/Role Name	Type	Description	Operation
Policies/Roles	AAD fulkcom	System-defined policy	Pull parmissions for Advanced Anh-DDeS.	Modify Delate
Projects	A4D ReadOnlyAccess	System-defined policy	Read-only permissions for Advanced Anti-2DuS.	Modily Delete
Identity Providers	ADN Publicoss	System-defined policy	Pull permissions for Application Delivery Network.	Modity Delete
Security Settings	ADN Instance Operations	System-defined policy	Instance operation permissions for Application Delivery Network.	Modify Delate
	ADN ReadOnlyAccess	System-defined policy	Read-only permissions for Application Delivery Network.	Modify Delate
	AOM FullAccess	System-defined policy	AOM Access All	Modify Delete

2. Specify a policy name, policy view, and content.

#### Figure 4-11 Configuring a custom policy

* Policy Name						
· Defen Central						
* Panj Galles	Alter     Deny	C Select service	Select action	(Optional) Select resource	C(Dptonal) Ass request condition	E' U
	Select Existing Policy/Role     Add Permissions					
Description	Enter a brief description.	0256 <i>a</i>				
Scope	- OK Cancel					

 Table 4-17 Custom policy description

Parameter	Description
Policy Name	Enter a policy name.
Policy View	Select <b>JSON</b> .
Policy Content	Configure the following policy content: {     "Version": "1.1",     "Statement": [         {             "Action": [             "das:*:*",             "nosql:instance:list"             ],             "Effect": "Allow"         }     ] } Alternatively, click Select Existing Policy/Role, select DAS FullAccess as a template, and retain only the DB type information. In this example, retain only     posal-instance:list
Description	Enter a policy description
Description	Enter a policy description.
Scope	Retain the default settings (project-level service).

3. Click **OK**. You can then view the created custom policy on the **Permissions** page.

#### Figure 4-12 Viewing the created policy

IAM	Policies/Roles ()			Freedback     Create Duston Palicy
Users				
User Groups	Delete Custom policies available for creation: 199			
Permissions ^	Al policies/roles	<ul> <li>Fatty search</li> </ul>	✓ ) ○ Criter a policy name, role name, or description.	
Authorization	Policy/Role Name	Type	Description	Operation
Policies/Roles		Custom policy	-	Modify Delete

4. Create a user group.

Figure 4-13 Creating a user group

IAM	User Groups 🕚				Cirelite User Oroup
Users					
User Groups	Delete User groups available for creation: 19				
Permissiona ^	C. Enter a group name.				
Authorization	Nome @	Users	Description @	Created @	Operation
Policies/Roles	admin .	1	Pull permissions	Nov 68, 2023 14 17:52 GMT+88:00	Authorize Modily Manage User Delete
Projects Agencies	Total Records: 1 19 🗸 < 1				

5. Authorize the user group created in **4** using the created custom policy.

Figure 4-14 Authorizing the user group using the created custom policy

IAM	User Groups 💿			•
Users				
User Groups	Delete User groups available for creation: 18			
Permissions ^	C. Enter a group name.			
Authorization	Name 0	Users Description ()	Created 🖨	Operation
Policies/Roles	C user-group	0 –	Aug 02, 2024 11:25:35 GMT+03:00	Authorize Modify Manage User Delete
Projects	admin	1 Pull permissions	Nov 60, 2023 14:17:52 GMT+60:00	Authorize Modily Manage User Delete
Agencies				

Figure 4-15 Selecting the created custom policy

<	Authorize User Group					
•	Steled PolicyRide (2) Select Scope (3) Fresh					
	Assign selected permissions to user-group.					Create Policy
	View Selected (1) Copy Permissions from Another Project		All policies/toles	✓ Al services	→ ) [ Fuzzy search → ) [ Enter a policy name, role name, or do	scription. Q.
	PolicyRole Name	Type				
	☑ ✓ mfldefition _	Outlone p	olicy			

6. Click the name of the user group and add the required users.

#### Figure 4-16 Adding users

IAM	User Groups / user-group	
Users User Groups	Name user-group 🖉	Group Manage User ×
Permissions V	Description	User Group user-group
Projects	uosinpuor - u	Available Users (1) Enter a usemame. Q Selected Users (0) Enter a usemame. Q
Agencies		Username User Groups Username Operation
Identity Providers	Permissions Users	siying_cen View
Security Settings	Add Remove	
	Username 🕀	
		No data available.
	<	
		Cancel

## **Prerequisites**

There is an available GeminiDB Redis instance.

## Procedure

- **Step 1** Log in to the Huawei Cloud console.
- **Step 2** In the service list, choose **Databases** > **GeminiDB**.
- **Step 3** In the instance list, locate the target instance and click **Log In** in the **Operation** column.

Figure 4-17 Logging in to a GeminiDB Redis instance

 Name/D (i)
 DB Instance Type
 Compatible API
 Strange Type
 Status (i)
 Load Induneer address
 Billing Mode
 Operation

 Compatible API
 Strange Type
 Status (i)
 Available
 PN-132:158.103
 Payaevase
 Compatible API

Alternatively, click the instance name to go to the **Basic Information** page. Click **Log In** in the upper right corner of the page.

Figure 4-18 Logging in to a GeminiDB Redis instance

<   geminidb-2df4-free	etest 🟮 Available	Feedback Calific Linis     Log In     View Metric     Reset Password     Restart     C		
Basic Information				
Backups & Restorations	Basic Information			
Note Management	DB Instance Name	DB Instance ID	Storage Type	Product Type
Accounts	<b>,,,,,,,,,,,,,,,,</b> ,,,,,,,,,,,,,,,,,,,,		Classic	Standard
Slow Query Logs	Status	Resion	42	DB Instance Type
Parameters	Available		ct-north-40, cn-north-4a, cn-north-4c	Primary/Standby
Netrics				
Sessions	Maintenance Window 🕥			
Diagnosis Analysis	10:00 - 14:00 Change			

**Step 4** Enter the password for logging in to the instance.

Figure 4-19 Login page

DB Instance Name
Enter your pageward
Enter your passworu.

If you need to log in again after the password is reset, click **Re-login** in the upper right corner and use the new password.

Figure 4-20 Re-login

Current Database	Re-login Save to Executed Commands ()
C60         v         Creater (FII)         Error         My Commands         Creat of FII)	
Executed Commands Messages Results	

**Step 5** Manage relevant databases.

Figure 4-21 Instance homepage

mmands 🕤 🚺

Save commands to the execution record.

This function is enabled by default to save the recently executed commands for your later query.

Then you can click the **Executed Commands** tab on the lower page to view historical commands.

#### Figure 4-22 Executed commands

Executed Co	mmands Messages Results			
Executed		Command	Time Required	Result
Jun 26, 2024	10:34:29 GMT+08:00	SCAN 0	2ms	Succeeded
Jun 26, 2024	10:33:52 GMT+08:00	SCAN 0	3ms	Succeeded

If this function is disabled, the commands executed subsequently are not displayed any longer. You can click 
 react to Save Executed SQL
 Statements in the upper right corner to disable this function.

• Execute a command.

You can enter a command in the command window and click **Execute** or **F8**.

**NOTE** 

- Do not use transactions, Lua scripts, Pub/Sub commands, or other commands that have blocking semantics.
- For an instance that supports multiple databases, you can change the current database on the console, but cannot change it using a SELECT statement.

#### Figure 4-23 Executing a command

DB0 ~	Execute (F8)	Save My Commands Clear (F10)		
1 SCAN 0				
Executed Commands	Messages	Results		
Command			Cursor	Result
SCAN 0			0	(empty list or set)

After a command is executed, you can view the execution result on the **Results** page.

• Save a command.

You can save a command to all instances, the current instance, or the current database. Then you can view details in **My Commands**.

#### Figure 4-24 Save

DB0	Execute (F8)     Save     My Commands	Save		×
1	SCAN 0	Title		
		Application Scope	All instances (e) Current instance Current database	
			OK Cancel	)

• View my commands.

Common commands are displayed the My Commands page.

Set a filter to narrow the scope of commands. If you select **All**, all commands saved in the current account are displayed.

#### Figure 4-25 Filtering commands

My Cor	nmands					×
Create	e Command			All ^	Enter a title or command.	Q
No.	Title	Application Scope	Com	All	Operation	
1	scan	Current instance	scan	All instances	Edit Delete	Copy to Command Window
2	select	All instances	selee	Current instance Current database	Edit Delete	Copy to Command Window

Alternatively, enter a command title or statement in the search box to search for the corresponding command.

#### Figure 4-26 Searching for a command

My Con	nmands				~
Create	Command		All	~ s	X   Q
No.	Title	Application Scope	Command	Ор	peration
1	scan	Current instance	scan O	Ed	it Delete Copy to Command Window
2	select	All instances	select 1	Ed	it Delete Copy to Command Window

On the **My Commands** page, you can also create, edit, and delete a command or copy it to the command window.

#### Figure 4-27 Managing a command

My Con	nmands					
Create	Command		All	~	Enter a title or command.	Q
No.	Title	Application Scope	Command		Operation	
1	scan	Current instance	scan O		Edit Delete	Copy to Command Window
2	select	All instances	select 1		Edit Delete	Copy to Command Window

#### • Clear a command.

You can also press F10 to clear the command in the command window.

Figure 4-28 Clearing a command

DB0	
1	scan 0

----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 Redis Instance Over a Private Network

## **4.3.3.1 Connecting to an Instance Using a Load Balancer Address** (Recommended)

This section describes how to connect to a GeminiDB Redis instance using a load balancer address on a Linux ECS. Load balancing can improve data reliability and eliminate SPOFs.

## **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.

Scenario 1: If the instance is associated with the default security group, you do not need to configure security group rules.

Scenario 2: If the instance is not associated with the default security group, check whether the security group rules allow the ECS to connect to the instance.

- If yes, the ECS can connect to the instance.
- If no, add an inbound rule to the security group.

For details about how to configure a security group, see **4.3.5.1 Setting** Security Group Rules for a GeminiDB Redis Instance.

## Prerequisites

• 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*.

• Download the **Redis client installation package**.

## Procedure

- **Step 1** Log in to the ECS. For details, see **Logging In to an ECS** in *Getting Started with Elastic Cloud Server*.
- Step 2 Obtain the Redis client.

#### Method 1

Run the following command to download the Redis client.

wget http://download.redis.io/releases/redis-6.2.0.tar.gz

#### Method 2

Download the Redis client from the address provided in **Prerequisites** and upload the Redis client installation package to the ECS.

**Step 3** Decompress the client package.

tar -xzf redis-6.2.0.tar.gz

**Step 4** Open the **src** directory and connect to the DB instance.

cd redis-6.2.0

make

cd src

./redis-cli -h <DB\_HOST> -p <DB\_PORT> -a <DB\_PWD>

Example:

```
./redis-cli -h 192.xx.xx.-p 6379 -a <DB_PWD>
```

Table 4-18 Parameter description

Parameter	Description
<db_host></db_host>	Load balancer IP address of the instance to be connected. After the load balancer IP address is created, click the instance name to go to the <b>Basic Information</b> page and obtain the load balancer IP address in the <b>Connection</b> <b>Information</b> area.
<db_port></db_port>	Access port corresponding to the load balancer IP address of the instance. The procedure is as follows:
	Click the instance name to go to the <b>Basic Information</b> page. In the <b>Connection Information</b> area, you can see the instance port.
<db_pwd></db_pwd>	Administrator password set when you buy a GeminiDB Redis instance

Step 5 Check the results. If the following information is displayed, the connection is successful. IP:port>

----End

## 4.3.3.2 Connecting to an Instance Using a Private Domain Name

This section describes how to connect to a GeminiDB Redis instance using a private domain name on a Linux ECS.

## 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.

Scenario 1: If the instance is associated with the default security group, you do not need to configure security group rules.

Scenario 2: If the instance is not associated with the default security group, check whether the security group rules allow the ECS to connect to the instance.

- If yes, the ECS can connect to the instance.
- If no, add an inbound rule to the security group.

For details about how to configure a security group, see **4.3.5.1 Setting Security Group Rules for a GeminiDB Redis Instance**.

## Prerequisites

- 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*.
- Download the **Redis client installation package**.

## Procedure

## Configuring a Private Domain Name of the GeminiDB Redis Instance

#### Creating a Private Domain Name

- Step 1 Log in to the Huawei Cloud console.
- Step 2 Click Service List. Under Network, click Domain Name Service.
- **Step 3** On the displayed page, click **Private Zones**.

#### Figure 4-29 Private zones

Domain Name Service ⑦							
My Resources							
Public Zones	1	Private Zones	1	PTR Records	0	Record Sets	6

#### Step 4 Click Create Private Zone.

#### Figure 4-30 Creating a private zone

Private Zones ⑦							(?) User Guide	Create Private Zone
We would much appreciate it if you could complete our qu	estionnaire on Domain Name Service. Your feed	back will help us provide a	better user experience.					×
You can create 40 more private zones.			All projects 💌	All statuses	• Name	¥	٩	Search by Tag 😸 🛛 🖸
Name	Status	Record Sets	Associated VPC		Enterprise Project	Description	Operation	
У Пурсер-	Normal	3	v		default		Associate VPI	Modify Delete

#### **Step 5** Set parameters as prompted.

## Figure 4-31 Private zone parameters

	Enter a domain name, for example, example.com.
Region	•
VPC	▼ C View VPC ⑦
Enterprise Project	-Select- C (?) Create Enterprise Project
Tag	It is recommended that you use TMS's predefined tag function to add the same tag to different cloud resources. View predefined tags ${f C}$
	To add a tag, enter a tag key and a tag value below.
	To add a tag, enter a tag key and a tag value below.          Enter a tag key       Enter a tag value       Add
	To add a tag, enter a tag key and a tag value below.          Enter a tag key       Enter a tag value       Add         10 tags available for addition.       Enter a tag value       Add

#### Table 4-19 Parameter description

Parameter	Description	Example Value
Domain Name	Domain name of a private zone	example.com
	You can enter a top-level domain that complies with the domain naming rules.	
	For details about the domain name format, see <b>Domain Name Format and</b> <b>DNS Hierarchy</b> .	
Region	Region where a tenant is located	CN East-Shanghai1

Parameter	Description	Example Value
VPC	The VPC associated with the private domain name must be the same as the VPC where the GeminiDB Redis instance is located. Otherwise, the private domain name cannot be resolved.	-
Enterprise Project	Enterprise project associated with the private domain name. You can manage private domain names by enterprise project. <b>NOTE</b> This parameter is available and mandatory only when <b>Account Type</b> is set to <b>Enterprise Account</b> . Configuration notes: If you do not manage domain names by	default
	<ul> <li>If you manage domain names by enterprise project, select an existing enterprise project.</li> </ul>	

Parameter	Description	Example Value
Tag	(Optional) Identifier of a resource. Each tag contains a key and a value. You can add a maximum of 20 tags to a domain name.	example_key1 example_value1
	Key and value naming rules.	
	<ul> <li>Cannot be left blank.</li> <li>Must be unique for each resource.</li> </ul>	
	• Can contain a maximum of 128 characters.	
	<ul> <li>Can contain letters, digits, spaces, and special characters:=+-</li> <li>@ but cannot start or end with a space or start with _sys</li> </ul>	
	Value:	
	• Can contain a maximum of 255 characters.	
	<ul> <li>Can contain letters, digits, spaces, and the following special characters: _::/=+-@</li> </ul>	
Description	(Optional) Description of the zone, which cannot exceed 255 characters	This is a zone example.

**Step 6** Click **OK**. On the **Private Zones** page, view the created private domain name in the zone list.

If the status of the private domain name is **Normal**, the domain name has been successfully created.

Figure 4-32 Private domain name status

Priv	vate Zones ⑦						User Guide     Create Private Zone
•	We would much appreciate it if you could complete our questionnaire on Dor	nain Name Service. Your fe	edback will help us provide a	i better user experience.			×
Ì	fou can create 39 more private zones. Delete			All projects	▼ Name	v	Q Search by Tag 😸 C
	Name	Status	Record Sets	Associated VPC	Enterprise Project	Description	Operation
	✓ 🔲 nosqLcom.	<ul> <li>Normal</li> </ul>	2	vpc-	default		Associate VPC   Modify   Delete

#### ----End

#### Adding a Record Set for a Domain Name

After creating a private domain name, configure a record set for it so that you can access instances using the domain name.

**Step 1** Click the private domain name you created. On the displayed page, click **Add Record Set** in the upper right corner.

Figure 4-33 Adding a record set

<   nosql.com.	×						Add Record Set
Record Sets							
Export and Import	Record sets in public and private zones can be searched global	lly. Try Now					
	Private zones take effect only after you change the DNS serve	rs used by subnets in the associated	I VPCs to	and			
	You can add 434 more record sets. Delete			All statuses	• Altypes • Name •		Q Search by Tag 😸 C
	□ Name ↓⊟	Status	Type J⊟	TTL (s)	Value	Description	Operation
	✓ 📄 nesql.com.	<ul> <li>Normal</li> </ul>	NS	172,800	ns2 private.hwclouds-dns.com. ns1.private.hwclouds-dns.com.		Modify   Delete
	✓ mosqlcom.	<ul> <li>Normal</li> </ul>	SOA	300	ns1.private.hwclouds-dns.com. hwcloudsi,cs.huawel.com. (1 7200 90	0	Modify   Delete

**Step 2** In the displayed **Add Record Set** dialog box, set parameters as prompted.

Value: Enter the load balancer IP address.

					redistest. (	?
Туре	A – Map domains t	o IPv4 addresses				
TTL (s)	300	5 min	1 h	12 h	1 day	?
Value	Example: 192 168 10 10					
						?
						h
Tag	It is recommended th different cloud resour To add a tag, enter a	at you use TMS's ces. View predefi	predefined tag fu ned tags C	nction to add the	e same tag to	
	To add a tag, enter a	tag key and a tag	Value Delow.			
	Enter a tag key		Enter a tag val	ue	Add	
	10 tags available for	addition.				
Description						
					0/2	//
					0120	

For details about how to configure parameters, see Adding an A Record Set.

- **Step 4** Switch back to the **Record Sets** page.
- **Step 5** View the created record set in the record set list. If the status of the record set is **Normal**, the record set is added successfully.

----End

## Logging In to an ECS and Connecting an Instance to the Redis Client

- **Step 1** Log in to the ECS. For details, see **Logging In to an ECS** in *Getting Started with Elastic Cloud Server*.
- Step 2 Obtain the Redis client.

#### Method 1

Run the following command to download the Redis client.

wget http://download.redis.io/releases/redis-6.2.0.tar.gz

#### Method 2

Download the Redis client from the address provided in **Prerequisites** and upload the Redis client installation package to the ECS.

**Step 3** Decompress the client package.

#### tar -xzf redis-6.2.0.tar.gz

**Step 4** Open the **src** directory and connect to the DB instance.

cd redis-6.2.0

make

cd src

./redis-cli -h <DB\_Domain\_Name> -p <DB\_PORT> -a <DB\_PWD>

Example:

./redis-cli -h redis.com -p 6379 -a <DB\_PWD>

#### Table 4-20 Parameter description

Parameter	Description
<i><db_domain_na me&gt;</db_domain_na </i>	Private domain name of the instance to be connected. The private domain name is the one created in <b>Configuring a Private Domain Name of the GeminiDB Redis Instance</b> .
<db_port></db_port>	Port for accessing the target instance. Configure this parameter based on service requirements. To obtain the port number, perform the following steps: Click the instance name to go to the <b>Basic Information</b> page. In the <b>Connection Information</b> area, you can see the instance port.

Step 3 Click OK.

Parameter	Description
<db_pwd></db_pwd>	Administrator password set when you buy a GeminiDB Redis instance

**Step 5** Check the results. If the following information is displayed, the connection is successful.

Domain\_Name:port>

----End

## 4.3.3.3 Connecting to an Instance Using a Private IP Address

You can use the private IP address to connect to the GeminiDB Redis instance.

This section uses the Linux OS as an example to describe how to connect to a GeminiDB Redis instance using the Redis-cli client. You can connect to an instance through SSL to secure your data. For details, see **4.3.5.6 Connecting a GeminiDB Redis Instance over SSL** This section describes how to connect to a GeminiDB Redis instance in non-SSL mode.

To ensure data reliability, you are advised to use a **load balancer address** or **domain name** to access the instance.

## 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 **4.3.5.1 Setting Security Group Rules for a GeminiDB Redis** Instance.
- To connect to a DB instance over a non-SSL connection, SSL must be disabled.
   For details about how to disable SSL, see 4.3.5.5 Encrypting Data over SSL for a GeminiDB Redis Instance.

## Prerequisites

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*.

## Procedure

- **Step 1** Log in to the ECS. For details, see **Logging In to an ECS** in *Getting Started with Elastic Cloud Server*.
- Step 2 Obtain the Redis client.

#### Method 1

Run the following command to download the Redis client.

wget --no-check-certificate https://download.redis.io/releases/redis-6.2.0.tar.gz

#### Method 2

Download the **Redis client** and upload it to the ECS.

- Step 3 Decompress the client package. tar -xzf redis-6.2.0.tar.gz
- **Step 4** Open the **src** directory and connect to the DB instance.

cd redis-6.2.0 make cd src ./redis-cli -h <*DB\_HOST*> -p <*DB\_PORT*> -a <*DB\_PWD*>

Example:

./redis-cli -h 192.xx.xx.xx -p 6379 -a <*DB\_PWD*>

Table 4-21 Parameter description

Parameter	Description
<db_host></db_host>	Private IP address of an instance to be connected.
	To obtain this IP address, go to the <b>Instances</b> page and click the target instance name. In the navigation pane, choose <b>Node Management</b> . You can see the private IP address in the <b>Node Information</b> area on the <b>Basic Information</b> page.
	If the instance you purchased has multiple nodes, select the private IP address of any node.
<db_port></db_port>	Port for accessing the target instance. Configure this parameter based on service requirements.
	To obtain the port number, perform the following steps:
	Click the instance name to go to the <b>Basic Information</b> page. In the <b>Connection Information</b> area, you can see the instance port.
<db_pwd></db_pwd>	Administrator password set when you buy a GeminiDB Redis instance

**Step 5** Check the results. If the following information is displayed, the connection is successful.

IP:port>

----End

# 4.3.4 Connecting to a GeminiDB Redis Instance Over a Public Network

## 4.3.4.1 Connecting to an Instance Using an EIP Bound to a Load Balancer (Recommended)

This section describes how to access a GeminiDB Redis instance over a public network by creating a load balancer and binding it to an EIP. To prevent single points of failure (SPOFs) in the production environment and implement load balancing, you are advised to connect to the GeminiDB Redis instance using an EIP bound to a load balancer.

To connect to a GeminiDB Redis instance over a public network, use a public domain name to ensure instance reliability. For details, see **4.3.4.3 Connecting to an Instance Using a Public Domain Name**.

## **Usage Notes**

Redis Cluster GeminiDB Redis instances do not support this function.

## Creating and Configuring a Dedicated Load Balancer

- **Step 1** Purchase a **dedicated load balancer**. For details, see **Creating a Dedicated Load Balancer**. Pay attention to the following:
  - When creating a flavor, you need to select the TCP/UDP network.
  - In the network configuration, **Cross-VPC Backend** must be enabled so that backend IP addresses can be added to the load balancer.
  - You need to use a new or existing EIP to support public network access.
- **Step 2** Locate the target load balancer and click **Add Listener** in the **Operation** column. For details, see **Adding a TCP Listener**. Pay attention to the following:

#### Figure 4-35 Adding a listener



- When configuring a listener, select the TCP protocol to and the **6379** port, which is commonly used by Redis.
- When adding a backend server, click the Cross-VPC Backend Servers tab and then click Add Cross-VPC Backend Server. Configure the load balancer address and port number of the GeminiDB Redis instance in the cross-VPC backend IP address.
- Enable the health check.
- **Step 3** If ELB and GeminiDB Redis instances are in the same VPC, go to **Step 7**. If they are not, go to **Step 4** to create a VPC peering connection.
- Step 4 In the navigation pane, choose Virtual Private Cloud > VPC Peering. On the VPC Peering page, click Create VPC Peering Connection and set Local VPC and Peer VPC.

**Local VPC** indicates the VPC where ELB is deployed.

- If the selected VPC and GeminiDB Redis instance are in the same VPC, set the peer VPC to a VPC where no ELB is deployed.
- If the selected VPC and GeminiDB Redis instance are not in the same VPC, set the peer VPC to the VPC where the GeminiDB Redis instance is deployed.

For details, see **Creating a VPC Peering Connection to Connect Two VPCs in the Same Account**.
**Step 5** Go to the **Summary** page of the VPC peering connection.

- In the dialog box displayed after the VPC peering connection is created, if you click **Add Now**, the **Summary** page is displayed.
- In the dialog box displayed after the VPC peering connection is created, if you click **Add Later**, the VPC peering connection list is displayed. If you click the peering connection name, the **Summary** page is displayed.

Figure 4-36 VPC peering connection

PC	Peering Connections ③	)					ا Feedback	Overview	Process Flow	Create VPC Peering Connection
	Q Select a property or enter a key	word.								00
	Name:1D 😔	Status 🖯	Local VPC 🖯	Local VPC CIDR Block 😔	Peer Project ID 😣	Peer VPC 😔	Peer VPC CIDR Block 😣	Descrip	⊖   Operation	
C	peering-93d9 c#31226-b006-45df-9709- 7b764a34b00a	Accepted	default_vpc		0b23a362270025c52/5 4c001c913beba	vpc-4920	192.168.0.0/16	-	Modify D	slote

**Step 6** Click **Add Route** and configure the local and peer routes of the VPC peering connection.

#### Figure 4-37 Add Route

^ peering-93d9						
Name	pearing-93d9 🖉		Star	fus O Accepted		
ID	cf31226-b01d-45df-9709-7b764a34b00a	ð	Pee	er Project ID 0623a3622700	25c52f54c001c913beba 🖒	
Local VPC Name	default_vpc		Pee	er VPC Name vpc-4920		
Local VPC ID		đ	Pee	er VPC ID 371afbd4-4ee7	-4492-6966-1728852e6561	
Local VPC CIDR Block			Per	er VPC CIDR Block		
Description	- 2					
Associated Routes	s	eering connection				
<ul> <li>Select a property</li> </ul>	v or enter a keyword.					
Destination $\Theta$	VPC	Next Hop Type	Next Hop	Route Table	Description	Operation
192.0.0.0/16	default_vpc	VPC peering connection	peering-93d9(ctf3t226-b0fd	H45df-9709-7b rtb-default_vpc	-	Defete

- Local route: In the displayed Add Route dialog box, set Destination to the value of Peer VPC CIDR Block of the VPC peering connection, set Next Hop Type to VPC peering connection, set Next Hop to the VPC peering connection created in Step 4, and click OK.
- Peer route: In the displayed Add Route dialog box, set Destination to the value of Local VPC CIDR Block of the VPC peering connection, set Next Hop Type to VPC peering connection, set Next Hop to the VPC peering connection created in Step 4, and click OK.
- Step 7 Perform a health check on the added load balancer address. In the navigation pane on, choose Elastic Load Balance > Load Balancers. Click the target load balancer name and then click the Listeners tab. If Healthy in the health check result, the address is available. You can access the GeminiDB Redis instance using the IP address displayed on the Summary tab page.

Elastic Load Balance	r / Load balancer ( ) 💿 Running		Add Listener View Backend Server
nary Listeners	Monitoring Access Logs Tags		
lame		VPC	dds-st-test-vpc
D	đ	IPv4 Subnet	dds-st-test-subnet
ype	Dedicated	IPv6 Subnet	
Z	AZ1	Backend Subnet	dds-st-test-subnet 🖉
pecification	Network load balancing(TCP/UDP)   Small   Application load balancing(HTTP/HTTPS)   Small	IP as a Backend	Enable (?)
lescription	- 🖉	IP Address	Private IPv4 address   DI Modify   Un
			IPv4 EIP   10
			IPv6 address Bind
andwidth Information	IPv4   5 Mbit/s	Deletion Protection	
reated	Nov 25, 2023 11:01:08 GMT+08:00		

Figure 4-38 IP address



## Procedure

- **Step 1** Log in to the ECS. For details, see **Logging In to an ECS** in *Getting Started with Elastic Cloud Server*.
- **Step 2** Obtain the Redis client.

#### Method 1

Run the following command to download the Redis client.

wget --no-check-certificate http://download.redis.io/releases/redis-6.2.0.tar.gz

#### Method 2

Download the **Redis client** and upload it to the ECS.

- Step 3 Decompress the client package. tar -xzf redis-6.2.0.tar.gz
- Step 4 Open the src directory and connect to the DB instance.

cd redis-6.2.0 make cd src ./redis-cli -h <*DB\_HOST*> -p <*DB\_PORT*> -a <*DB\_PWD*>

Example:

./redis-cli -h 192.168.0.208 -p 6379 -a <*DB\_PWD*>

Parameter	Description
<db_host></db_host>	EIP bound to the instance to be connected.
	To obtain the EIP, go to the <b>Instances</b> page and click the target instance name. In the navigation pane, choose <b>Node Management</b> . You can see the EIP in the <b>Node Information</b> area on the <b>Basic Information</b> page.
	If the instance you bought has multiple nodes, you can bind the EIP to any node to connect to the instance.
	If a message is displayed indicating that no EIP has been bound to the instance, bind an EIP to the instance by following <b>4.3.5.4 Binding an EIP to a GeminiDB Redis</b> Instance Node.
<db_port></db_port>	Port for accessing the target instance. Configure this parameter based on service requirements.
	To obtain the port number, perform the following steps:
	Click the instance name to go to the <b>Basic Information</b> page. In the <b>Connection Information</b> area, you can see the instance port.
<db_pwd></db_pwd>	Administrator password set when you buy a GeminiDB Redis instance

Table 4-22 Para	meter description
-----------------	-------------------

**Step 5** Check the results. If information similar to the following is displayed, the connection is successful.

IP:port>

----End

## 4.3.4.2 Connecting to an Instance Using an EIP

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

This section uses the Linux OS as an example to describe how to connect to a GeminiDB Redis instance using the Redis-cli client. You can connect to a GeminiDB Redis instance using an EIP bound to a load balancer to avoid SPOFs and achieve load balancing in the production environment.

You can connect to an instance over SSL or non-SSL connections. SSL encrypts data and is more secure. For details, see **4.3.5.6 Connecting a GeminiDB Redis Instance over SSL**. This section describes how to connect to a GeminiDB Redis instance over a non-SSL connection.

## Usage Notes

• To connect to a DB instance over a non-SSL connection, SSL must be disabled. For details about how to disable SSL, see **4.3.5.5 Encrypting Data over SSL** for a GeminiDB Redis Instance.

- You need to estimate the bandwidth required by services and purchase an EIP with sufficient bandwidth resources. Client access exceptions caused by poor public network performance will not be included in the SLA.
- Redis Cluster GeminiDB Redis instances do not support this function.

## Prerequisites

- 1. 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*.
- 2. You have bound an EIP to a node of the purchased instance and configure security group rules for the node. For details, see **4.3.5.4 Binding an EIP to a GeminiDB Redis Instance Node** and **4.3.5.1 Setting Security Group Rules** for a GeminiDB Redis Instance.

#### **NOTE**

A GeminiDB Redis instance can have multiple nodes. Select any node and bind an EIP to it.

## Procedure

- **Step 1** Log in to the ECS. For details, see **Logging In to an ECS** in *Getting Started with Elastic Cloud Server*.
- **Step 2** Obtain the Redis client.

#### Method 1

Run the following command to download the Redis client.

wget http://download.redis.io/releases/redis-6.2.0.tar.gz

#### Method 2

Download the **Redis client** and upload it to the ECS.

- Step 3 Decompress the client package. tar -xzf redis-6.2.0.tar.gz
- Step 4 Open the src directory and connect to the DB instance.

cd redis-6.2.0 make cd src ./redis-cli -h <*DB\_HOST*> -p <*DB\_PORT*> -a <*DB\_PWD*>

Example:

./redis-cli -h 192.168.0.208 -p 6379 -a <*DB\_PWD*>

Parameter	Description
<db_host></db_host>	EIP bound to the instance to be connected.
	To obtain the EIP, go to the <b>Instances</b> page and click the target instance name. In the navigation pane, choose <b>Node Management</b> . You can see the EIP in the <b>Node Information</b> area on the <b>Basic Information</b> page.
	If the instance you bought has multiple nodes, you can bind the EIP to any node to connect to the instance.
	If a message is displayed indicating that no EIP has been bound to the instance, bind an EIP to the instance by following <b>4.3.5.4 Binding an EIP to a GeminiDB Redis</b> Instance Node.
<db_port></db_port>	Port for accessing the target instance. Configure this parameter based on service requirements.
	To obtain the port number, perform the following steps:
	Click the instance name to go to the <b>Basic Information</b> page. In the <b>Connection Information</b> area, you can see the instance port.
<db_pwd></db_pwd>	Administrator password set when you buy a GeminiDB Redis instance

**Step 5** Check the results. If the following information is displayed, the connection is successful.

IP:port>

----End

## 4.3.4.3 Connecting to an Instance Using a Public Domain Name

A public domain name is a domain name used to access websites or web applications on the Internet.

You can use Domain Name Service (DNS) to translate common domain names (for example, www.example.com) into IP addresses (for example, 1.2.3.4) required for network connection. In this way, you can access GeminiDB Redis instances using the resolved IP addresses.

This section uses the Linux OS as an example to describe how to use the public network domain name configured by the DNS service to connect to a GeminiDB Redis instance.

## **Usage Notes**

Redis Cluster GeminiDB Redis instances do not support this function.

## Prerequisites

- 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*.
- You have registered a domain name and an EIP.
- You have bound an EIP to a node of the purchased instance and configure security group rules for the node. For details, see **4.3.5.4 Binding an EIP to a GeminiDB Redis Instance Node** and **4.3.5.1 Setting Security Group Rules for a GeminiDB Redis Instance**.

#### **NOTE**

A GeminiDB Redis instance can have multiple nodes. Select any node and bind an EIP to it.

• Download the **Redis client installation package**.

## Procedure

## Configuring a Public Domain Name of the GeminiDB Redis Instance

#### Domain Name Not Created on Huawei Cloud

If a third-party domain name is used, create a public zone and add record sets to it on the DNS console.

- Step 1 Log in to the Huawei Cloud console.
- **Step 2** Click **Service List** and choose **Network > Domain Name Service**.
- **Step 3** In the navigation pane, choose **Public Zones**.

#### Figure 4-39 Public zones

DNS	Public Zones ⑦ RP Process Flow	Public Zones 🕥 🏘 Process Flow						
Dashboard Public Zones	Note     1. To build a website on servers located in the Chines     2. Complete real-name authentication for your domain	e mainland, you need to license your domain n name to be accessible. Otherwise, the dom	name and server IP addresses. Otherwise, your ain name will be in the ServerHold state and can	website cannot be accessed through you not be accessed. View details	r domain name. Wew details			
Private Zones	You can cruste 49 more sublic zones.							
PTR Records	Delete Batch Operation •						© C	
Custom Lines	Exact match by domain name	Exact match by domain name						
Domain Registration	□ Domain Name ⑦ J≣	Status	Record Sets	Enterprise Project	Description	Operation		
Elastic IP dP	1234456.com.	Normal	3	default	141	Manage Record Set Disab	sie More +	

**Step 4** In the upper right corner of the page, click **Create Public Zone**.

**Step 5** Set the parameters as prompted.

#### Figure 4-40 Creating a public zone

	Enter a domain name, for example, ex	ample.com.	
nterprise Project	-Select-	▼ C ⑦ Create Enterprise Proj	ect
g	It is recommended that you use TMS's cloud resources. View predefined tags To add a tag, enter a tag key and a tag	predefined tag function to add the sam C g value below.	e tag to different
	Enter a tag key	Enter a tag value	Add
	Enter a tag key 10 tags available for addition.	Enter a tag value	Add
escription	Enter a tag key 10 tags available for addition.	Enter a tag value	Add

#### Table 4-24 Public zone parameters

Parameter	Description	Example Value
Domain Name	Domain name you have registered.	example.com
	It can include two levels in addition to the top-level domain, for example:	
	<ul> <li>abc.example.com, the subdomain name of example.com</li> </ul>	
	<ul> <li>abc.example.com.cn, the subdomain name of example.com.cn</li> </ul>	
	For details about the domain name format, see <b>Domain Name</b> Format and DNS Hierarchy.	

Parameter	Description	Example Value
Enterprise Project	<ul> <li>Enterprise project associated with the public domain name. You can manage public domain names by enterprise project.</li> <li>NOTE <ul> <li>This parameter is available and mandatory only when Account Type is set to Enterprise Account.</li> </ul> </li> <li>Configuration notes: <ul> <li>If you do not manage domain names by enterprise project, select default.</li> </ul> </li> </ul>	default
	<ul> <li>If you manage domain names by enterprise project, select an existing enterprise project.</li> </ul>	
Tag	<ul> <li>(Optional) Identifier of a resource. Each tag contains a key and a value. You can add a maximum of 20 tags to a domain name.</li> <li>Key and value naming rules:</li> <li>Key: <ul> <li>Cannot be left blank.</li> <li>Must be unique for each resource.</li> </ul> </li> <li>Can contain a maximum of 128 characters.</li> <li>Can contain letters, digits, spaces, and special characters:=+-@ but cannot start or end with a space or start withsys</li> <li>Value: <ul> <li>Can contain a maximum of 255 characters.</li> <li>Can contain letters, digits, spaces, and special characters:=+-@ but cannot start or end with a space or start withsys</li> </ul> </li> </ul>	example_key1 example_value1
Description	(Optional) Description of the zone, which cannot exceed 255 characters	This is a zone example.

## Step 6 Click OK.

After the domain name is created, you can view it in the domain name list on the **Public Zones** page.

----End

#### Adding a Record Set for a Domain Name

After creating a public domain name, configure a record set for it so that you can access instances using the domain name.

**Step 1** Click the name of the public domain name you created. On the displayed page, click **Add Record Set** in the upper right corner.

Figure 4-41 Adding a record set



Step 2 In the displayed Add Record Set dialog box, set parameters as prompted.

Name			123	4456.com. 🥐	
к Туре	A – Map domains to IPv4 address	ses		•	
k Alias	🔾 Yes 💿 No 🥐				
k Line	Default			•	?
tTL (s)	300 5 min	1 h	12 h	1 day	?
k Value	Example: 192.168.10.10				?
Weight	1				?
Tag	It is recommended that you use TM different cloud resources. View pred To add a tag, enter a tag key and a	IS's predefined tag defined tags C tag value below.	function to add the s	ame tag to	
	Enter a tag key	Enter a tag v	alue	Add	
	10 tags available for addition.				
Description					
				ہر 0/255	

Figure 4-42 Adding a record set

For details about how to configure parameters, see Adding an A Record Set.

- Step 3 Click OK.
- **Step 4** Switch back to the **Record Sets** page.
- **Step 5** View the created record set in the record set list. If the status of the record set is **Normal**, the record set is added successfully.

----End

## Logging In to an ECS and Connecting an Instance to the Redis Client

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

**Step 2** Obtain the Redis client.

#### Method 1

Run the following command to download the Redis client.

wget http://download.redis.io/releases/redis-6.2.0.tar.gz

#### Method 2

Download the Redis client from the address provided in **Prerequisites** and upload the Redis client installation package to the ECS.

**Step 3** Decompress the client package.

#### tar -xzf redis-6.2.0.tar.gz

**Step 4** Connect to the instance in the **src** directory.

cd redis-6.2.0

make

cd src

./redis-cli -h <DB\_Domain\_Name> -p <DB\_PORT> -a <DB\_PWD>

Example:

./redis-cli -h redis.com -p 6379 -a <DB\_PWD>

Table 4-25 Parameter description

Parameter	Description
<i><db_domain_na me&gt;</db_domain_na </i>	Public domain name of the instance to be connected. The public domain name is the one created in <b>Configuring a Public Domain Name of the GeminiDB Redis Instance</b> .
<db_port></db_port>	Port for accessing the target instance. Configure this parameter based on service requirements.
	To obtain the port number, perform the following steps:
	Click the instance name to go to the <b>Basic Information</b> page. In the <b>Connection Information</b> area, you can see the instance port.
<db_pwd></db_pwd>	Administrator password set when you buy a GeminiDB Redis instance

**Step 5** Check the results. If the following information is displayed, the connection is successful.

Domain\_Name:port>

----End

## 4.3.5 Connection Information Management

## 4.3.5.1 Setting Security Group Rules for a GeminiDB Redis Instance

A security group is a collection of access control rules for ECSs and GeminiDB Redis 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 Redis instances.

This section describes how to configure security group rules for a GeminiDB Redis instance that is connected through a private or a public network.

## **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 Redis instance.
- For details about how to configure security group rules, see Table 4-26.

Table 4-26 Parameter descript	tion
-------------------------------	------

Scenario	Description			
Connecting to an instance over a private network	<ul> <li>Configure security group rules as follows:</li> <li>If a GeminiDB Redis instance and the ECS used for accessing the instance are in the same security group, they can communicate with each other by default. No security group rules need to be configured.</li> <li>If the instance and the ECS are not in the same security group, configure security group rules, respectively.</li> <li>Configure inbound rules for the security group associated with the GeminiDB Redis instance. For details, see Procedure.</li> <li>There is no need to configure security group rule of the ECS allows all outbound data packets. If not all outbound traffic is allowed in the security group, configure an outbound rule for the ECS. For details, see Configuring a Security Group Rule.</li> </ul>			
Connecting to an instance over a public network	If you connect to a GeminiDB Redis instance through a public network, configure inbound rules for the security group associated with the GeminiDB Redis instance. For details, see <b>Procedure</b> .			

## 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.

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

#### Figure 4-43 Security group

#### Security Group

Security Group default 🖉

#### Step 5 Add Inbound Rule

1. Click the Inbound Rules tab.

#### Figure 4-44 Inbound rules

Add Rule Fast-A	Add Rule Delete	Allow Common Ports Inb	ound Rules: 9				
Q Select a property or en	nter a keyword.						0
Priority	Action	Туре	Protocol & Port	Source	Description	Last Modified	Operation
□ 1	Allow	IPv4	TCP: 8635	0.0.0.0/0 ③	-	May 29, 2024 16:09:06	Modily Replicate Delete
□ 1	Allow	IPv4	TCP : 20-21	0.0.0.00 ③		May 29, 2024 16:08:47	Modily Replicate Delete
0.1	Allow	IPv4	TCP : 80	0.0.0.00 ④		May 29, 2024 16:08:47	Modify Replicate Delete
□ 1	Allow	IPv4	ICMP : All	0.0.0.00 ③		May 29, 2024 16:08:47	Modify Replicate Delete
0.1	Allow	IPv4	TCP: 3389	0.0.0.00 ③		May 29, 2024 16:08:47	Modily Replicate Delete
□ 1	Allow	IPv4	TCP : 22	0.0.0.00 ③		May 29, 2024 16:08:47	Modify Replicate Delete
□ 1	Allow	IPv4	TCP : 443	0.0.0.0/0 ④		May 29, 2024 16:08:47	Modify Replicate Delete
100	Allow	IPv4	All	default (3)	-	Aug 10, 2022 15:13:25	Modify Replicate Delete
100	Allow	IPv6	All	default 🕤		Aug 10, 2022 15:13:25	Modily Replicate Delete

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

#### Figure 4-45 Adding a rule



3. Add a security group rule as prompted.

Table	4-27	Inbound	rule	settings
-------	------	---------	------	----------

Parame ter	Description	Example Value			
Protoco l & Port	Protoco – Network protocol. Currently, GeminiDB Redis & Port instances can be accessed only over TCP.				
	<ul> <li>Port: The port or port range that allows the access to the ECS. Range: 1 to 65535 Common ports are listed in Common Ports Used by ECS.</li> </ul>				
Туре	Pe IP address type. This parameter is available after IPv6 is enabled.				
	– IPv4				
	– IPv6				
Source	The IP address, IP address group, or security group that the rule applies to, which allows access from IP addresses or instances in another security group. Examples:	0.0.0.0/0			
	<ul> <li>IPv4 single IP address: 192.168.10.10/32</li> </ul>				
	– Subnet: 192.168.1.0/24				
	<ul> <li>All IP addresses: 0.0.0.0/0</li> </ul>				
	– sg-abc (security group)				
	For more information about IP address groups, see <b>IP Address Group</b> .				
Descrip tion	(Optional) Provides supplementary information about the security group rule.	-			
	The description can contain up to 255 characters and cannot contain angle brackets (<>).				

#### Step 6 Click OK.

----End

## 4.3.5.2 Viewing the IP Address and Port Number of a GeminiDB Redis Instance

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

## Viewing the Load Balancer IP Address and Port

- 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.

**Step 4** In the **Connection Information** area, view the load balancer IP address and corresponding port.

Figure 4-46 Viewing the load balancer IP address and port

Connection Information			
Load Balancer Address	Database Port	Maximum Connections 20000 Sessions	Password-Free Access Disabled Enable
Access Control Configure() whitelist or blackiet configure() The listener added to your load balancer does not support socurity group rakes, so your need to configure access control for listener. "Sho belonging access control of address any listener using the load balancer IP address.	SSL ف Disabled ف		

----End

## Viewing the Private IP Address or 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 instance whose node IP addresses you want to view and click its name.

In the navigation pane on the left, click **Node Management** to view the private IP addresses and EIPs of the instance.

Figure 4-47 Obtaining IP addresses

Node information								
Stop Start Add Node Change								
Select one or more filters from the pop-up I	lists. If you enter a keyword without a filter	applied, the system will search for all nan	tes matching this keyword.					
Name/ID	Status	۸7	Private ID Address	FID	Operation			
Ramono		74.	THINK I AGAICSS	6-81	operation			
	Available	az4			View Metric Unbind EIP More $\checkmark$			
	Available	822		Unbound	View Metric Bind EIP More ~			
	Add Node Add Node Add Node Solect one or more filters from the pop-up MamelID	Add Node     Change     Start     Add Node     Change      Solect one or more films from the popula list. For one rate a knyword without a films     NamelD     Status     Anakable     CAvailable	Add Nods     Change Solat     Add Nods     AZ     Az     Az     Az     Az     Az     Az     Az	Add Node     Change     Solat     Add Node     Change     Add Node     A	Add Node         Change           Solid core or more Bless from he pop-up list. If you order a logword without a Bler applied, the system will search for all names matching this keyword.           Name/D         Status         AZ         Private IP Address         EIP <ul></ul>			

----End

## Viewing the Port for Accessing Each Instance Node

- 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 node access ports you to want view and click its name.

In the **Connection Information** area on the **Basic Information** page, view the port of each instance node.

Figure 4-48 Obtaining the port number

Connection Information			
Load Balancer Address	Database Port 6379 /	Maximum Connections 20000 Sessions	Password-Free Access Disabled Enable
Access Control	SSL		
The listener added to your load balancer does not support security group rules, so you need to configure access control for the listener. WPC of your instance to access your instance using the load balancer IF address.	ے Disabled		

#### ----End

## 4.3.5.3 Changing the Port of a GeminiDB Redis Instance

## **Scenarios**

You can change the port of a GeminiDB Redis instance to ensure security.

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

- Frozen
- Restarting
- Adding nodes
- Changing specifications
- Scaling up
- 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, click the target instance name.
- **Step 4** In the navigation pane, choose **Connections**.
- **Step 5** In the **Connection Information** area on the **Basic Information** page, click  $\checkmark$  next to **Database Port**.

The port number ranges from 1024 to 65535 except 2180, 2887, 3887, 6377, 6378, 6380, 8018, 8079, 8091, 8479, 8484, 8999, 9864, 9866, 9867, 12017, 12333, and 50069. Click  $\checkmark$  to submit the change. This process takes about 1 to 5 minutes.

- To cancel the change, click imes.
- **Step 6** Check the result in the **Connection Information** area on the **Basic Information** page.

----End

## 4.3.5.4 Binding an EIP to a GeminiDB Redis Instance Node

## Scenarios

After you create a GeminiDB Redis 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.

- You need to estimate the bandwidth required by services and purchase an EIP with sufficient bandwidth resources. Client access exceptions caused by poor public network performance will not be included in the SLA.
- Redis Cluster GeminiDB Redis instances do not support this function.

## 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, click the target instance. The **Basic Information** page is displayed.
- **Step 4** In the navigation pane, choose **Nodes**. In the **Node Information** area, browse to the target node and click **Bind EIP** in the **Operation** column.

Figure 4-49 Binding an EIP

Node Information								
Stop Start Add Node Change								
Q Select one or more filters from the pop	p-up lists. If you enter a keyword	without a filter applied, the system will se	earch for all names matching this keyword.					
Name/ID	Status	AZ	Private IP Address	EIP	Operation			
	Available	az4			View Metric Unbind EIP More $\sim$			
	<ul> <li>Available</li> </ul>	az2		Outpound	View Metric Bind EIP More ~			

**Step 5** In the displayed dialog box, view all available EIPs, select the required EIP, and click **OK**. If no available EIPs are displayed, click **View EIP** and create an EIP.

#### Figure 4-50 Selecting an EIP

Bind EIP			
After you bind an E rules in its security the GeminiDB, inst	IP to your instance, connect to it through { group to secure your data. If you want to u ead of the EIP console.	SSL and configure strict inbound and outbound unbind the EIP from your instance, do this on	
Node Information Not	ie Name	Status	
Select EIP No EIPs are a	vailable. Apply for new EIPs.	Available	C
EIP	Status	Bandwidth	
	No EIPs available View EIP	3	
		OK Cancel	

**Step 6** In the **EIP** column, view the EIP that is successfully bound.

To unbind the EIP from the DB 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** If a node has been bound to an EIP, click the target instance on the **Instances** page. The **Basic Information** page is displayed.
- **Step 4** In the navigation pane, choose **Nodes**. In the **Node Information** area, browse to the target node and click **Unbind EIP** in the **Operation** column.

Figure 4-51 Unbinding an EIP

Stop Start Add Node Change								
Q. Select one or more filters from the pop-up lists. If you enter a keyword without a filter applied, the system will search for all names matching this keyword.								
Name/ID	Status	AZ	Private IP Address	EIP	Operation			
	<ul> <li>Available</li> </ul>	az4	IPv4: 192.168.0.136	10.83.35.253	View Metric Unbind EIP More $ \times $			
	<ul> <li>Available</li> </ul>	az2	IPv4: 192.168.0.195	Unbound	View Metric Bind EIP More ~			

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

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

----End

Node Information

## 4.3.5.5 Encrypting Data over SSL for a GeminiDB Redis 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 time and increase CPU usage. So, evaluate impacts on service performance before enabling SSL.
- The SSL function provided by GeminiDB Redis supports only TLS 1.3 or later.

• Redis Cluster GeminiDB Redis instances do not support this function.

## 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 Connection Information area, click 

Figure 4-52 Enabling SSL
Figure 4-52 Enabling SSL
Intervention of the service of th

After SSL is enabled, you can connect to the instance through SSL connections. For details, see **4.3.5.6 Connecting a GeminiDB Redis Instance over 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 **Connection Information** area, click **C** to disable SSL.

#### Figure 4-53 Disabling SSL

onnection Information			
oad Balancer Address 92.168.0.144.6379	Database Port 6379 🖋	Maximum Connections 20000 Sessions	Password-Free Access Disabled Enable
Configure (No whitelist or blacklist configured) Configure (No whitelist or blacklist configured) In Bisterer added to your lead balancer does not support security group its jour part of the configure access control for the bisterer. PC of your Instance to access your instance using the load balancer IP dires.	SSL Enabled 그		

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

----End

## 4.3.5.6 Connecting a GeminiDB Redis Instance over SSL

GeminiDB Redis allows you to connect to a GeminiDB Redis instance through Redis-cli in SSL mode for data encryption and higher security. This section describes how to connect to a GeminiDB Redis instance using SSL.

## 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 4.3.5.1 Setting Security Group Rules for a GeminiDB Redis Instance.
- After the SSL connection is enabled, download the SSL certificate for your applications to access to the GeminiDB Redis instance.
- If the SSL connection is used, ensure that the Redis client, for example, Rediscli 6.x, supports SSL.
- Redis Cluster GeminiDB Redis instances do not support this function.

## Prerequisites

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*.

## Procedure

- **Step 1** Log in to the ECS. For details, see **Logging In to an ECS** in *Getting Started with Elastic Cloud Server*.
- **Step 2** Obtain the Redis client.

#### Method 1

Run the following command to download the Redis client.

wget https://download.redis.io/releases/redis-6.2.6.tar.gz

#### Method 2

Download the **Redis client** and upload the Redis client installation package to the ECS.

#### **Step 3** Obtain the SSL certificate.

Click the target instance name. On the **Basic Information** page, in the **Connection Information** area, click the download button in the **SSL** field to obtain the SSL certificate.

#### Figure 4-54 Obtaining the SSL certificate

connection mornation			
Load Balancer Address	Database Port 6379 🖋	Maximum Connections 20000 Sessions	Password-Free Access Disabled Enable
Access Control The listener added to your load balancer does not support security group rules, so you need to configure access control for the listener, e-to-plasting access control rol advors any IP address that can access the VPC of your instance to access your instance ting the load balancer IP address	SSL Enabled		

**Step 4** Upload the SSL certificate to the ECS.

**Step 5** Check the OpenSSL version supported by the ECS OS. openssl version

- The SSL function provided by GeminiDB Redis supports only TLS 1.3 or later.
- The OpenSSL version in the ECS OS must be 1.1.1 or later so that redis-cli can support TLS 1.3 or later.
- If the OS version is earlier than 1.1.1, perform the following steps to install OpenSSL:

```
wget https://www.openssl.org/source/openssl-1.1.1m.tar.gz
tar -zxvf openssl-1.1.1m.tar.gz
cd openssl-1.1.1m/
./config --prefix=/usr/local/openssl-1.1.1m_install_dir
make
make install
```

After OpenSSL is installed, go to **Step 6**.

- If the OS is 1.1.1 or later, go to Step 6.
- **Step 6** Decompress the client package.

tar -xzf redis-6.2.6.tar.gz

- **Step 7** Open the **src** directory and connect to the DB instance.
  - If the required OpenSSL version has been installed by performing **Step 5** and the version is earlier than 1.1.1, you can connect to the DB instance using the following method:

```
cd redis-6.2.6
make BUILD_TLS=yes OPENSSL_PREFIX=/usr/local/openssl-1.1.1m_install_dir
cd src
```

LD\_PRELOAD=/usr/local/openssl-1.1.1m\_install\_dir/lib/libssl.so.1.1:/usr/local/ openssl-1.1.1m\_install\_dir/lib/libcrypto.so.1.1 ./redis-cli -h <DB\_HOST> -p <DB\_PORT> -a <DB\_PWD> --tls --cacert <CACERT\_PATH>

Example:

LD\_PRELOAD=/usr/local/openssl-1.1.1m\_install\_dir/lib/libssl.so.1.1:/usr/local/ openssl-1.1.1m\_install\_dir/lib/libcrypto.so.1.1 ./redis-cli -h 192.168.0.208 -p 6379 -a *<DB\_PWD>* -tls --cacert ./cacert.crt

If the OpenSSL version in the ECS OS is 1.1.1 or later, you can connect to the DB instance using the following method:
 cd redis-6.2.6
 make BUILD\_TLS=yes
 cd src

./redis-cli -h <DB\_HOST> -p <DB\_PORT> -a <DB\_PWD> --tls --cacert <CACERT\_PATH>

Example:

./redis-cli -h 192.168.0.208 -p 6379 -a *<DB\_PWD>* --tls --cacert ./cacert.crt

#### Table 4-28 Parameter Description

Parameter	Description	
<db_host></db_host>	Private IP address of an instance to be connected.	
	To obtain this IP address, go to the <b>Instances</b> page and click the target instance name. In the navigation pane, choose <b>Nodes</b> . You can see the private IP address in the <b>Node Information</b> area on the <b>Basic Information</b> page.	
	If the instance you purchased has multiple nodes, select the private IP address of any node.	

Parameter	Description
<db_port></db_port>	Port for accessing the target instance. Configure this parameter based on service requirements.
	To obtain the port number, perform the following steps:
	Click the instance name to go to the <b>Basic Information</b> page. In the <b>Connection Information</b> area, you can see the instance port.
<db_pwd></db_pwd>	Administrator password set when you buy a GeminiDB Redis instance
<cacert_path></cacert_path>	SSL certificate path

Step 8 Check the results. If information similar to the following is displayed, the connection is successful. IP:port>

----End

-

## 4.3.5.7 Changing the Security Group of a GeminiDB Redis Instance

## **Scenarios**

You can change the security group of a GeminiDB Redis instance.

## Precautions

• If you are adding nodes to an instance, the security group 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, choose **Node Management**.
- **Step 5** In the **Security Group** area, click  $\swarrow$  to select a security group.
  - To submit the change, click  $\checkmark$ . This process takes about 1 to 3 minutes.
  - To cancel the change, click  $\times$ .
- **Step 6** View the modification result.

----End

## 4.3.5.8 Configuring Private Network Access to a GeminiDB Redis Instance

## Scenarios

GeminiDB Redis allows you to enable or disable private network access for a load balancer.

## **Usage Notes**

- A load balancer address does not support security groups. After instance creation is complete, configure IP address access control. If no whitelist is configured, all IP addresses that can communicate with the VPC can access the instance.
- Redis Cluster GeminiDB Redis instances do not support this function.

## Enabling a Blacklist/Whitelist for a Load Balancer IP Address

- 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 **Connection Information** area, click **Interview** next to **Access Control**.

#### Figure 4-55 Enabling private network access for a load balancer

Connection Information			
Load Balancer Address	Database Port 6379 //	Maximum Connections 20000 Sessions	Password-Free Access Disabled Enable
Access Control	SSL		
Configure(No whiteist or biackist configured) The Insteiner added to your load balancer does not support security group rules, so you need to configure access control for the listener. - or-Disability access control allows any IP address that can access the VPC of your instance to access your instance using the load balancer IP address.	C Enabled 土		

Step 5 Select Blacklist or Whitelist and specify IP addresses in that list.

## Figure 4-56 Configuring access control Configure Access Control

<ol> <li>Select an New setti</li> </ol>	access policy. If you change the poli ngs are applied to both new and exis	icy, this setting becomes invalid. sting connections.
Access Policy	Whitelist Blacklist	e not allowed to access your instance.
IP Address	Example: 192.168.0.1   proxy	3
		Yes No

- Blacklist: The blacklist and whitelist cannot be configured at the same time. If you switch between lists, your previously entered settings will be lost. IP addresses in the blacklist cannot be accessed. Exercise caution when performing this operation.
- Whitelist: The blocklist and allowlist cannot be configured at the same time. If you switch between lists, your previously entered settings will be lost. Only IP addresses in the whitelist are allowed to access the system. Exercise caution when performing this operation.
- ----End

## Disabling Private Network Access for a Load Balancer

- 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 **Connection Information** area, click **V** next to **Access Control**. In the displayed dialog box, click **Yes**.

Figure 4-57 Disabling private network access for a load balancer

connection mormation			
Load Balancer Address	Database Port 6379 🖋	Maximum Connections 20000 Sessions	Password-Free Access Disabled Enable
Access Control	SSL		
The listener added to your load balancer does not support security group rules, so you need to configure access control for the listenerbroliabiling access control allows any IP address that can access the VPC of your instance to access your instance users the load behaver. IP address.	Disabled 🕹		

**Step 5** Check the load balancer address cannot take effect.

----End

## 4.4 Data Migration

## 4.4.1 Migration Solution

This section describes how to migrate services to a GeminiDB Redis 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.

## Usage Notes

Cloud services such as Alibaba Cloud Tair (Redis<sup>®</sup> OSS-Compatible) and TencentDB for Redis cannot use Huawei Cloud DRS for data migration due to the following factors:

- Some self-developed Redis-like databases are not compatible with the PSync protocol.
- Architecture restrictions: For many cloud vendors, the proxy component is added between users and Redis. PSync is not supported due to the proxy.
- Security restrictions: In native Redis, fork() is used over PSync, which causes memory expansion, user request delay increase, and even out of memory.
- Business strategy: A large number of users use RedisShake to migrate services from the cloud or change the cloud, so PSync is shielded.

## **Migration Tool**

- Data Replication Service (DRS) is used for full and incremental data migration while ensuring data security. For details, see **Migration Overview**.
- **Redis-Shake** tool: is an open-source migration tool that supports migration modes such as full scanning (rump), data restoration (restore), and incremental synchronization (sync). You can download the tool to an ECS and use CLI to facilitate migration.

## **Required Permissions**

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

## **Migration Scenarios**

Table 4-29 Migration scenario	S
-------------------------------	---

No.	Source	Destination	Migration Solution
1	GeminiDB Redis API	Redis/ GeminiDB Redis API	4.4.2 (Recommended) Using DRS to Migrate Data from a GeminiDB Redis Instance to an Open-Source Redis Instance
2	Alibaba Cloud Tair (Redis® OSS- Compatible)	GeminiDB Redis	4.4.3 Migrating the Alibaba Cloud Database Redis/Tair To GeminiDB Redis
3	Redis	GeminiDB Redis	4.4.4 (Recommended) Using DRS to Migrate Data from Open-source Redis or Redis Cluster to GeminiDB Redis API
4	Redis	GeminiDB Redis	4.4.5 Migrating Data from Redis to GeminiDB Redis API Using Redis- Shake
5	RDB file	GeminiDB Redis	4.4.6 Using Redis-Shake to Import an RDB or AOF File to a GeminiDB Redis Instance
6	RDB file	GeminiDB Redis	4.4.7 (Recommended) Importing Data to Restore RDB Files to a GeminiDB Redis Instance
7	Kvrocks	GeminiDB Redis	4.4.8 Migrating Data from Kvrocks to GeminiDB Redis API
8	Pika	GeminiDB Redis	4.4.9 Migrating Data from Pika to GeminiDB Redis API
9	SSDB	GeminiDB Redis	4.4.10 Migrating Data from SSDB to GeminiDB Redis API
10	LevelDB	GeminiDB Redis	4.4.11 Migrating Data from LevelDB to GeminiDB Redis API
11	RocksDB	GeminiDB Redis	4.4.12 Migrating Data from RocksDB to GeminiDB Redis API
12	AWS ElastiCache for Redis	GeminiDB Redis	4.4.13 Migrating Data from Amazon ElastiCache for Redis to GeminiDB Redis API

# 4.4.2 (Recommended) Using DRS to Migrate Data from a GeminiDB Redis Instance to an Open-Source Redis Instance

Data Replication Service (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 a GeminiDB Redis instance to an open-source Redis instance, see **From GeminiDB Redis to Redis**.

# 4.4.3 Migrating the Alibaba Cloud Database Redis/Tair To GeminiDB Redis

This section describes how to migrate Alibaba Cloud databases Redis or Tair to GeminiDB Redis.

## **Migration Principles**

• The data migration function of the Alibaba Cloud data migration tool DTS is used to migrate data from Alibaba Cloud Redis to other Redis services. This tool avoids the restrictions of shielding the sync and psync commands of Alibaba Cloud Redis and migrates data from Alibaba Cloud Redis to Huawei Cloud GeminiDB Redis.

## Precautions

- The source end on Alibaba Cloud needs to communicate with the destination end on Huawei Cloud. Ensure that a private line is enabled or that binding a public IP address is performed.
- The Alibaba Cloud DTS data migration function is charged in real time. Before using this function, ensure that your Alibaba Cloud account balance is sufficient.
- The Huawei Cloud GeminiDB Redis capacity must be greater than or equal to the memory capacity of the Alibaba Cloud Redis database.
- Ensure that the security group configuration on the source and target ends is enabled.
- Some Redis databases on Alibaba Cloud are special. For example, Tair hybrid storage does not support online full and incremental migration. You can complete the migration by scanning all the data.

## Preparations

- Migrating data using a public IP address
  - Purchase a Huawei Cloud EIP in advance. The bandwidth must be greater than the source database traffic.
  - Bind the EIP to a Huawei Cloud GeminiDB Redis node.
  - When configuring DTS, ensure that the destination database is connected through a public IP address.
- Migrating data using a private line
  - Purchase an Alibaba ECS in advance to ensure that it can connect to Huawei Cloud GeminiDB Redis.

 Configure data forwarding to forward the traffic received by the local port to the destination end, implementing migration from Alibaba Cloud Redis to GeminiDB Redis.

ssh -g -L (Forwarding port): (LB IP address of Huawei Redis): (Huawei Redis port) -N -f root@ (Local ECS IP Address)

- When configuring DTS, ensure that the destination database is connected through a self-built ECS database.

## Procedure

## Purchasing the Data Synchronization Function of DTS

**Step 1** Select the Redis service on Alibaba Cloud as the source end. If an EIP is used for migration, select a public IP address as the destination end and enter the EIP as the host name. If Direct Connect is used for migration, select the self-built Redis database on ECS as the destination end, set the host name to the IP address of the ECS, set the port number to the forwarding port number, enter the database password, and click the test link. If no exception occurs during the test, the next page is displayed. Otherwise, check whether the entire link is normal and whether the whitelist configuration is correct.

Figure 4-58 Source and destination configuration information

015 / Data Migrator	n / Gere bas Myssie fait	Al Asistant 🦄 105 Dirgald Group Kunteer (1) Productive oblication (Hd)
Create     Configure     Seasce and     Destination     Databases	LOTA Mingration Lask     Configuration Lask     Configuration Lask and Destination Database     Configuration	9 fait bleast ban
2 Configure To Objects	Proceeds in synar inverse of data for a large participation, we accorned that give a field at a professional set data professiona set data professional set data profesional set data professiona	Devinetes Delaces Sala Dolog Constants V n
Object Configuration res	Endows Type O     Notice	- Outstaan Type © Ch Aryanetiik ko feels tahaneet Kilaan (tah)
Advanced Settings	Robble Horgediz, MacDello Robbitz Aparoli Constitue to h(dz, BD for Sent (dott), BD for 2011 Mac, MacDell, MacCenpute, Sentata Tota	* Anex Method Alaza Couch Interver Eposit Convert, VMV Gravey, or Snart Alaza Colevery PAdde: IP Address Self-managed Database on ES Couch Employee Network (CDI)
Configuration re	Annu Melval     Annu Melva     Annu Melva     Append Convext VM Obrang or Freet Access Galeway     Padic IP Address     Soft-managed Studies on ESS     Courd Streptice Network ESN     Courd Streptice Network ESN	Extension General Prynominghene das transportable Proteins, retracted henny traces new non-To ensure induced studied years and Access Method to Expense Connect, VHI Galeway, or Senal Access Extension Connect Access Method for Connect Access Extension Connect Access Exten
Peteck	* Interes Regim Chies (Rangia) *	- totala nipol Cove Brevel v
Purchase Instance	National Star Even and Ball Duck House 0           No         No           * Assoc 0         V	Sociare or # Advect
	Details Record The primed loss of heir Characteria (Amandonum), (B) Per Lyper Method	Lith Cathor Forced
		🖲 the expand 🔿 33, expand

Step 2 Select Full Data Migration or Full Data Migration + Incremental Data Migration. Select Pre-check and Report Errors and select the database to be migrated.

## 

If you use the multi-DB function, select all databases to be migrated. If the multi-DB function is not used, select only **DB0**.



Figure 4-59 Database to be migrated



075	/ Oats Migration / Col	as Dat Myanin Tak	Al Assister i 🐂 OS Dirgtak Group Number - 🛞 Production	oduction FAQ	
<del>(</del>	- Create Data Migration Task				
0		Advanced Cardiguetters           Very Twen for each conserve ()           Twen Twen for each conserve ()	The first to be	Ventout Venton	
0	Configure Task Objects	To the Thread of the Hyperteen O			
1	Object Configuratio re	hammend by they no investigation of (2) such that you is not also be some around probable to be some a			
	Advanced Settings Verification Configuratio re Precheck	$\because the lower to be a lower to get a set of the lower to get a $			
(	Purchase Instance	V = V = N → N → N → N → N → N → N → N → N → N			
				2	
		Preview Step Object Centerpretors Read Step Methodes Configurations Cand			

**Step 3** After the pre-check is complete, click **Next: Purchase Instance**.

Figure 4-61 Pre-check

0% / Ora Higo	adaa / Cours Data Migazian Task			Al Asimare i 👒 015 Dirgalk Group Number 🛞 Product Innoduction - FAQ
← Crea	te Data Migration Task			
-	" Precheck			10 Back to Previous Vendory
Contigan Season as	e Second Max			
Destination	01 Total number of tasks 5.0 tasks are running, 1 tasks 6	have triggered alerts. I tasks have failed. I tasks are not rooted. O tasks are stipped. 4 tasks are	completed.	125
	Check Item		Content	Check Result
Contern	Redo Driction Policy		Check whether the Reds exiction policy is noeviction	🔒 Varving 🔹 Van Delah
Objects	Source Database Connectivity		Check whether the ETS services can connect to the source delabeles	Summed a
	Check the connectivity of the destination database	*	Oracle schedure the 27% servers are convent to the destination statutane.	Successful
Configure	atio Source Database Version		Check the version of the source database.	Soccessful
ns	Check the existence of objects in the destination	database.	Check whether the table in the destination database contains data.	Decemble
Settings     Weifcati     Configur     re     Pretheak     Pachase     Instance	90 909			
				(e) (0) (0)

**Step 4** Select the bandwidth for the migration and click **Buy and Start**.

Figure 4-62 Bandwidth configuration



**Step 5** If **Full Data Migration + Incremental Data Migration** is selected, the migration task will not automatically end. If there is no delay (0 ms), the full synchronization is complete.

Figure 4-63 Task status

Alibaba Goud ApsaraD6	DTS w / Data Nigration								A Asistant	♦ DTS Dirgtelk Group Number ③ Product Introduction 1942
ors	Data Migration									Show M
Cate Synchronization	Data Migration Tasks 💿									
Charge Tracking	Croste Tack Participationse Took Form	e Y Seach by keyword	Q Status Select	V Sortily Crestor Time (	learning v Environment Top Select	1 × Riterly1	by Y			V 0 0
Orte Migration										
Orte Verification	DNare	SourceDestination	Status	Rennie	g Information	Instance Class	Billing Method	Tug (3	Created At	Adors
Dedicated Cluster										
ETL Paste Provins	magilini koviši (1)g	I Albaba Coud Instance - Reds - Orina (	Renie	Incenental Data Migration	2008PS(000046/s)	box.	Permanan	0	Joy 8, 2024, 16/08/11	Del Debik   Ouid Davardis   E
Buent Center	estitivity (D Reputer	() Internet - Redis - China (Shangha)		(i) kur una remanor	LateropOhiliseconds 😜					
Database Connections	1 Start Task Assos Task Step Task	Deter Test Set Tag							hams per Page 10	✓ Total Items 1 Items — C. Perrices 1 Heat >

----End

## Stopping the Data Synchronization Function of DTS

**Step 1** After the Redis service migration, stop the data synchronization task.

Figure 4-64 Stopping the data migration task

Alibaba Cloud ApsaraDB	015 V / Data Migration								H heistert	♥ DTS Dingtalk Group Number ③ Product Introduction 540
DTS	Data Migration									Show V
Orta Synchronization	Data Migration Tasks 💿									
Charge Tracking Data Migration	Orate Task Anchase Instance Task	Name V Seach by keyword	Q. Status Select	V Sot By Destion Time (De	sanding) V EnvironmentTag Select	<ul> <li>✓ Filter by 1</li> </ul>	laj V			₽ ♦ 0
Data Verification	D/am	SourceDestination	Status	Running I	Morreation	Instance Cless	Biling Nethod	Tag O	Created At	Ations
ETL (Table Parsion) Event Carder	dogidar4alad415g dot26h3y (0 Regular	<ul> <li>Albaba Coud Instance - Redis - Drine (</li> <li>Instance - Redis - Drine (Shangha)</li> </ul>	Running	<ul> <li>Incremental Data Migration</li> <li>Full Data Verification</li> </ul>	20095/00004653 Lateny@Alilacords@	lage	Ny-Keyos-po	0	Apr 8, 2004, 165851	Terk Details   Quick Diagnostics
Database Connections	1 Section Associat Sec	Task Delete Task Edit Tag							items per Page 10	✓ Totalitiens 1 items → C hericus 1 liest >

----End

## Verifying Redis Data Consistency After Migration

After the migration is complete, you can check the consistency of Redis data.

#### NOTICE

- Data has been migrated from Redis, or incremental migration has started.
- Redis-full-check must be deployed on the ECS, and the ECS is connected to the source and destination databases.
- During incremental migration, data may be inconsistent due to network latency between the source and destination databases. You are advised to stop writing data to the source and then verify data consistency.
- When Redis is used, an expiration time is usually set for keys. During migration, setting a key expiration time affects data consistency. Data may be inconsistent due to inconsistent expiration time.
- During migration, DTS writes temporary probing keys to Redis on the destination database. Non-service data may be detected during data verification, which is normal.

#### Procedure

- **Step 1** Log in to the ECS and ensure it is connected to the source and destination Redis databases.
- Step 2 Deploy redis-full-check.
- Step 3 Verify data.

/redis-full-check -s {Source IP address}:{Source port} -p {Source password} -t
{Destination IP address}:{Destination port} -a {Destination password} -m 1

Parameter	Description	Example Value		
-S	Source Redis database address and port number	-s 10.0.0.1:6379		
-р	Password of the source Redis database	-		
-t	Destination GeminiDB Redis database address and port number	-t 10.0.0.2:6379		
-a	Password of the destination GeminiDB Redis database	-		

Parameter	Description	Example Value		
-m	Verification mode:	-m 1		
	1. All key-value pairs			
	2. Value length only			
	3. Key integrity only			
	<ol> <li>All key values are verified, but only the length of big keys is verified.</li> </ol>			
	By default, the second verification mode is used.			
-q	Maximum QPS. The default value is <b>15000</b> .	-q 5000		
-d	Name of the file for saving the verification result. The default value is <b>result.db</b> .	-d result.db		

**Step 4** View the verification result file.

By default, three rounds of verification are performed and three verification result files are generated. Generally, you only need to view the last verification result file.

- Run the **sqlite3 result.db.3** command.
- Run the **select** \* **from key** command.
- Check whether there are abnormal keys.

Enter ".help" for usage hints.					
sqlite> select * from key;					
1 b string lack target 0 1 0					
2 c string lack target 0 1 0					
3 a string lack target 0 1 0					
sqlite>					

----End

## 4.4.4 (Recommended) Using DRS to Migrate Data from Opensource Redis or Redis Cluster to GeminiDB Redis API

Data Replication Service (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 Redis to GeminiDB Redis API, see **From Redis to GeminiDB Redis API**.

For details about how to use DRS to migrate data from Redis Cluster to GeminiDB Redis API, see **From Redis Cluster to GeminiDB Redis API**.

# 4.4.5 Migrating Data from Redis to GeminiDB Redis API Using Redis-Shake

You can use DRS or Redis-Shake to migrate data from Redis to GeminiDB Redis API. Redis-Shake is taken as an example in this section.

## **Migration Principles**

Use Redis-Shake to migrate data from an on-premises Redis instance (source) to a GeminiDB Redis instance (destination). Full and incremental migrations are both supported. The source can be a single-node, primary/standby, or cluster instance, or an RDB file.

- Full migration: Redis-Shake works as a slave node for the source, obtains data of an RDB file generated by the source, and then parses the data and sends it to the destination by running commands. You can also use an RDB file as the source to import snapshot data generated at a specific time point.
- Incremental migration: After full migration is complete, Redis-Shake continues sending incremental data to the destination by running commands until you stop Redis-Shake.

## **Usage Notes**

- If data synchronization between master and slave Redis nodes is disconnected, stop Redis-Shake, clear all data in the destination, and retry a migration. To ensure a smooth synchronization, migrate data during off-peak hours and set a large value for parameter **client-output-buffer-limit** to increase the ring buffer size for incremental synchronization.
- Redis-Shake does not write data into the source, but may have a temporary impact on the source performance.
- If the migration involves multiple databases, ensure that source databases are correctly mapped to destination databases to prevent unexpected data overwriting.
- Streaming data cannot be migrated.
- Ensure that network communication among Redis-Shake, the source instance, and the destination instance is normal.
- To migrate data from open-source Redis to GeminiDB Redis API, choose Service Tickets > Create Service Ticket in the upper right corner of the console and contact the customer service.

# Migrating Data from a Single-Node or Primary/Standby Redis Instance to a GeminiDB Redis Instance

You can import a file similar to the above or perform the following operations to migrate data from an open-source single-node or primary/standby Redis instance to a GeminiDB Redis instance.

- **Step 1** Deploy the required migration tool.
  - 1. Obtain the **Redis-Shake package**.

#### **NOTE**

Download the Redis-Shake release package and decompress it.

2. Modify the **Redis-Shake.conf** configuration file and configuring the following items:

**log.level = info** #Default log level. A printed INFO log contains migration progress information, based on which you can judge whether the migration is complete.

**source.address = <host>:<port> #** IP address and port of a host where an open-source Redis instance is deployed

**source.password\_raw = \*\*\*\*\*** # Password for logging in to a source instance

source.type = standalone # Source instance type

target.address = <host>:6379 # Destination instance IP address

**target.password\_raw** = **\*\*\*\*\* #** Password for logging in to a destination instance

target.version = 5.0 # Version of the destination Redis instance

**target.type = standalone** # Destination instance type

**target.db** = -1 **#** Specific database on the destination that all data will be migrated to. If this parameter is set to -1, a mapping relationship is established between migrated databases and databases in the source instance.

3. Specify whether data of the destination is overwritten.

#### key\_exists = none

#### D NOTE

If there are duplicate keys on the source and destination, specify whether data of the destination is overwritten. The options are as follows:

- **rewrite** indicates that the source overwrites the destination.
- none indicates that the migration process exists once duplicate keys are detected.
- **ignore** indicates that keys in the source are retained and keys in the destination are ignored. This value does not take effect in rump mode.

**none** is recommended. There will be no duplicate data because the source is an RDB file. If the migration exits unexpectedly, you can choose **Service Tickets > Create Service Ticket** in the upper right corner of the console and contact the customer service.

#### Step 2 Migrate data.

Run the following command to start migration:

#### ./redis-shake.linux -conf=redis-shake.conf -type=sync

- If the following information is displayed, the full synchronization is completed and incremental synchronization begins.
- If the following information is displayed, no new data is incremented. You can stop the migration process to disconnect incremental synchronization:
   sync: +forwardCommands=0 +filterCommands=0 +writeBytes=0

#### **Step 3** Verify data.

Download and decompress **RedisFullCheck** and use it to verify data by referring to **Migrating Data from a Single-Node or Primary/Standby Redis Instance to a GeminiDB Redis Instance**.

## ./redis-full-check -s SOURCE\_IP:SOURCE\_PORT -p SOURCE\_PWD -t TARGET\_IP:6379 -a TARGET\_PWD

If the following information is displayed, the migration is successful, and data is consistent between the source and destination:

all finish successfully, totally 0 key(s) and 0 field(s) conflict

----End

## Migrating Data from a Redis Cluster Instance to a GeminiDB Redis Instance

Configure the following items in the configuration file:

source.address = <host1>:<port1>,<host2>:<port2>,<host2>:<port2> # IP
addresses and ports of source hosts

source.type = cluster # Cluster type of the source.

For other steps, see **Migrating Data from a Single-Node or Primary/Standby Redis Instance to a GeminiDB Redis Instance**.

# Migrating Data from an Open-Source Codis Cluster Instance to a GeminiDB Redis Instance

Obtain host IP addresses and ports of all shards of the Codis cluster instance and configure the configuration file as follows:

source.address = <host1>:<port1>,<host2>:<port2>,<host2>:<port2> # IP
addresses and ports of hosts at the source.

source.type = cluster # Cluster type of the source.

For other steps, see Migrating Data from a Single-Node or Primary/Standby Redis Instance to a GeminiDB Redis Instance.

## Fully Scanning Data on and Migrating It from an Open-Source Redis Instance to a GeminiDB Redis Instance

If data cannot be migrated with any of the above methods, try rump of Redis-Shake to scan databases one by one and migrate them.

- **Step 1** Deploy the required migration tool.
  - 1. Obtain the **Redis-Shake package**.

**NOTE** 

Download the Redis-Shake release package and decompress it.

2. Modify the **Redis-Shake.conf** configuration file and configuring the following items:

**log.level = info** #Default log level. A printed INFO log contains migration progress information, based on which you can judge whether the migration is complete.

source.address = <host>:<port> # IP address and port of a host where an
open-source Redis instance is deployed

source.password\_raw = \*\*\*\*\* # Password for logging in to a source instance
source.type = standalone # Source instance type

target.address = <host>:6379 # Destination instance IP address

**target.password\_raw = \*\*\*\*\* #** Password for logging in to a destination instance

**target.version = 5.0 #** Version of the destination Redis instance

**target.type = standalone** # Destination instance type

**target.db** = **-1 #** Specific database on the destination that all data will be migrated to. If this parameter is set to **-1**, a mapping relationship is established between migrated databases and databases in the source instance.

3. Specify whether data of the destination is overwritten.

#### key\_exists = none

#### **NOTE**

If there are duplicate keys on the source and destination, specify whether data of the destination is overwritten. The options are as follows:

- **rewrite** indicates that the source overwrites the destination.
- **none** indicates that the migration process exists once duplicate keys are detected.
- **ignore** indicates that keys in the source are retained and keys in the destination are ignored. This value does not take effect in rump mode.

**none** is recommended. There will be no duplicate data because the source is an RDB file. If the migration exits unexpectedly, you can choose **Service Tickets > Create Service Ticket** in the upper right corner of the console and contact the customer service.

#### Step 2 Migrate data.

Run the following command to start migration:

#### ./redis-shake.linux -conf=redis-shake.conf -type=rump

 If information similar to the following is displayed, synchronizing full data is complete. dbRumper[0] executor[0] finish

#### **Step 3** Verify data.

Download and decompress **RedisFullCheck** and use it to verify data.

## ./redis-full-check -s SOURCE\_IP:SOURCE\_PORT -p SOURCE\_PWD -t TARGET\_IP:6379 -a TARGET\_PWD

If the following information is displayed, the migration is successful, and data is consistent between the source and destination:

all finish successfully, totally 0 key(s) and 0 field(s) conflict

----End

## Verifying Data Consistency After Migration

After the migration is complete, you can check data consistency.
#### NOTICE

- Data has been migrated from Redis, or incremental migration has started.
- Redis-full-check must be deployed on the ECS, and the ECS is connected to the source and destination databases.
- During incremental migration, data may be inconsistent due to network latency between the source and destination databases. You are advised to stop writing data to the source and then verify data consistency.
- When Redis is used, an expiration time is usually set for keys. During migration, setting a key expiration time affects data consistency. Data may be inconsistent due to inconsistent expiration time.
- During migration, DTS writes temporary probing keys to Redis on the destination database. Non-service data may be detected during data verification, which is normal.

#### Procedure

- **Step 1** Log in to the ECS and ensure it is connected to the source and destination Redis databases.
- Step 2 Deploy redis-full-check.
- Step 3 Verify data.

/redis-full-check -s {Source IP address}:{Source port} -p {Source password} -t
{Destination IP address}:{Destination port} -a {Destination password} -m 1

Parameter	Description	Example Value
-S	Source Redis database address and port number	-s 10.0.0.1:6379
-р	Password of the source Redis database	-
-t	Destination GeminiDB Redis database address and port number	-t 10.0.0.2:6379
-a	Password of the destination GeminiDB Redis database	-

<b>Tuble - 31</b> Fullineter description	Table	4-31	Parameter	description
--	-------	------	-----------	-------------

Parameter	Description	Example Value
-m	Verification mode:	-m 1
	1. All key-value pairs	
	2. Value length only	
	3. Key integrity only	
	<ol> <li>All key values are verified, but only the length of big keys is verified.</li> </ol>	
	By default, the second verification mode is used.	
-q	Maximum QPS. The default value is <b>15000</b> .	-q 5000
-d	Name of the file for saving the verification result. The default value is <b>result.db</b> .	-d result.db

**Step 4** View the verification result file.

By default, three rounds of verification are performed and three verification result files are generated. Generally, you only need to view the last verification result file.

- Run the **sqlite3 result.db.3** command.
- Run the **select** \* **from key** command.
- Check whether there are abnormal keys.

Enter ".help" for usage hints.
<pre>sqlite&gt; select * from key;</pre>
1 b string lack_target 0 1 0
2 c string lack_target 0 1 0
3 a string lack target 0 1 0
sqlite>

----End

# 4.4.6 Using Redis-Shake to Import an RDB or AOF File to a GeminiDB Redis Instance

# Importing an RDB File to a GeminiDB Redis Instance

**Step 1** Deploy the required migration tool.

1. Obtain Redis-Shake.

D NOTE

Download the Redis-Shake release package and decompress it.

2. Modify the **Redis-Shake.conf** configuration file and configuring the following items:

**log.level** = **info** #Default log level. A printed INFO log contains migration progress information, based on which you can judge whether the migration is complete.

source.rdb.input = /xx/xx.rdb # Absolute path of the source RDB file

target.address = <host>:6379 # Destination instance IP address

**target.password\_raw** = \*\*\*\*\* # Password for logging in to a destination instance

target.version = 5.0 # Version of the destination Redis instance

target.type = standalone # Destination instance type

**target.db** = **-1 #** Specific database on the destination that all data will be migrated to. If this parameter is set to **-1**, a mapping relationship is established between migrated databases and databases in the source instance.

**target.dbmap** = #Configure the database migration mapping. The value of **target.db** must be -1, for example, 0-5. 1-3 indicates that data in source database **db0** will be written to destination database **db5** and data in source database **db1** will be written to destination database **db3**.

**big\_key\_threshold =** 52428800 # Big key threshold. If the number of value bytes corresponding to a key exceeds the threshold, data is written in batches.

**resume\_from\_break\_point = false** #Disable resumable download. This function is unavailable.

3. Specify whether data of the destination is overwritten.

#### key\_exists = none

#### **NOTE**

If there are duplicate keys on the source and destination, specify whether data of the destination is overwritten. The options are as follows:

- **rewrite** indicates that the source overwrites the destination.
- none indicates that the migration process exists once duplicate keys are detected.
- **ignore** indicates that keys in the source are retained and keys in the destination are ignored. This value does not take effect in rump mode.

**none** is recommended. There will be no duplicate data because the source is an RDB file. If the migration exits unexpectedly, you can choose **Service Tickets > Create Service Ticket** in the upper right corner of the console and contact the customer service.

#### Step 2 Migrate data.

Run the following command to start migration:

#### ./redis-shake.linux -conf=redis-shake.conf -type=restore

#### 

Use the restore mode because the source is an RDB file.

Stop the migration process after the migration is complete.

**Step 3** Verify data.

Data is obtained from the RDB file. Therefore, you need to check the GeminiDB Redis data at the destination end from the service perspective.

----End

# Importing an AOF File to a GeminiDB Redis Instance

- **Step 1** Upload the generated AOF file to an ECS.
- **Step 2** Start the open-source Redis 5.0 single-node process on the ECS to load the AOF file and wait till the process is started. Ensure that the startup directory of the open-source Redis is the same as the directory containing the AOF file.
- **Step 3** Run the SAVE command to generate an RDB file. Place the RDB file in the startup directory of the open-source Redis.
- **Step 4** Stop the open-source Redis 5.0 process.
- **Step 5** Perform the migration by following **Importing an RDB File to a GeminiDB Redis Instance**.

----End

# 4.4.7 (Recommended) Importing Data to Restore RDB Files to a GeminiDB Redis Instance

# Scenarios

Redis data of other vendors or self-hosted Redis can be migrated to GeminiDB Redis API.

You need to download the source Redis data, then upload the data to an OBS bucket in the same region as the GeminiDB Redis instance, and create a data import task on the GeminiDB console to import the data to the GeminiDB Redis instance.

# Precautions

- Importing data will overwrite data of the current database.
- Importing backups generated by a later-version Redis instance to an earlier one may fail.
- Before importing backups, ensure that resource-intensive commands (such as FLUSHALL, KEYS, and HGETALL) have been disabled on the target Redis instance.
- If a backup contains multi-DB data, its database count cannot exceed what is supported by the target Redis instance.
- Only .rdb files can be imported.

# **Creating an OBS Bucket and Uploading Backups**

If the backup to be uploaded is larger than 5 GB, follow the **instructions** provided by OBS.

Perform the following steps if the backup is smaller than 5 GB:

#### Step 1 Create an OBS bucket.

When creating an OBS bucket, configure the following parameters. For details, see **Creating a Bucket** in *Object Storage Service User Guide*.

1. Region:

The OBS bucket must be in the same region as the destination Redis instance.

2. **Storage Class**: Available options are **Standard**, **Infrequent Access**, and **Archive**.

Do not select **Archive**. Otherwise, the backup may fail to be imported.

- 3. Click Create Now.
- **Step 2** In the bucket list, click the bucket created in **Step 1**.
- Step 3 In the navigation pane, choose Objects.
- **Step 4** On the **Objects** tab page, click **Upload Object**.
- Step 5 Specify Storage Class.

Do not select **Archive**. Otherwise, the backup may fail to be imported.

**Step 6** Upload the objects.

Drag files or folders to the **Upload Object** area or click **add file**.

A maximum of 100 files can be uploaded at a time. The total size cannot exceed 5 GB.

#### Figure 4-65 Uploading objects in batches

Upload Obj	ect How to Upload a	File Greater than 5 GB?			
Storage Class	Standard	Infrequent Access	Archive		
	Optimized for frequer	ntly accessed (multiple times	per month) data suo	h as small and essential files that require low latency.	
	The default storage c more	lass is the same as that of th	e bucket. You can cl	nange the storage class according to your actual needs. Learn	
Upload Object	Note: If the bucket is existing file/folder.	not versioning-enabled, uploa	ding a file/folder wi	h the name that already exists in the bucket will replace the	
		Dias	OBS File of folders berg		
		Drag (A maximum of 100 files o	tiles of folders here can be uploaded at a	to upload. Of add file I time. The total size cannot exceed 5 GB.)	
Encryption	Encrypts the file for s KMS encryption	ecure storage. The encryption	n status of the encry	pted file cannot be changed.	
		Up	load Canc	el	

- Step 7 (Optional) Select KMS encryption to encrypt the uploaded files.
- Step 8 Click Upload.
  - ----End

# **Importing Backups**

- 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** > **Import Data** in the **Operation** column.

#### Figure 4-66 Importing data

Name/ID \ominus		DB Insta	Compati	St	Status 😔	Specificatio	Storage 5	Space	Load bal	Enterpri	Billing M	Opera	tion
	2	Cluster	Redis 6.2		Available	2 vCPUs St 3 nodes	0.01%	0/16GB		default	Pay-per-Usi Created	Log In	Change to Yearly/Monthly More ~
													Change Specifications
													Create Backup
													Scale Storage Space
													Add Node
													Restart
													Reset Password
													Delete
													Create Dual-Active Relationship
													Create DR Relationship
													Rename High-risk Command
													Import Data

- **Step 4** On the **Data Import** page, specify **OBS Bucket** to which a backup have been uploaded.
- **Step 5** Click **Add Backup** and select the backups to be imported.

Figure 4-67 Adding backups

mport Data						
1 Importing data will overwrite database data.     2. Importing backups perversited by a littler-version Redis Instance to an earlier one may fail.     3. Before importing backups, mixing that resource-intensive commands (such as FLUGHALL, KEYS, and HOETALL) have been disabled on the target Redis instance.     4. If the backup contains multi-DB data its database count cannot exceed what is supported by the target Redis instance.     5. Only rule files are supported.						
DB Instance Name OBS Bucket	DB Instance Name OBS Bucket asobleeprints-ch-storth-7  C View OBS Bucket					
Backup	Backup To insport more backups, oreale multiple import tasks and nin them at the same time.           Add Backup         Cear         You can add 126 more backups.					
	File Name	File Path	Size	Last Modified	Operation	
	dump.rdb	1	130 B	Jan 15, 2024 19:36:34 GMT+08:00	Delete	
	dump1.rdb	I	411 B	Jan 15, 2024 19:36:34 GMT+08:00	Delete	

- A maximum of 128 backups can be added at a time.
- To delete a backup, locate the target backup and click **Delete** in the **Operation** column.
- To delete all backups, select **Clear** for **Backup**.
- Step 6 Click Create Now.
- **Step 7** Confirm the data import and click **OK**.

Importing data will overwrite data of the current database.

X

#### Figure 4-68 Confirming to import data





# 4.4.8 Migrating Data from Kvrocks to GeminiDB Redis API

Kvrocks is an open-source NoSQL key-value database that is compatible with the Redis ecosystem. It uses namespace to partition data based on the underlying RocksDB. However, it is relatively weak in cluster management. Kvrocks needs to cooperate with other components to create clusters and does not support some Redis commands, such as stream and hyperloglog that are frequently used in message flow and statistics scenarios.

GeminiDB Redis API is a cloud-native NoSQL database with decoupled compute and storage and full compatibility with Redis. To ensure data security and reliability, it provides multi-copy, strict consistency based on a shared storage pool. It provides high compatibility, cost-effectiveness, high reliability, elastic scalability, high availability, and hitless scale-out. GeminiDB Redis API functions as good as Redis Cluster does and is completely compatible with Redis. You can migrate data from Redis instances to GeminiDB Redis instances without refactoring. In addition to adapting to Kvrocks, GeminiDB Redis API also improves management capability and compatibility with Redis.

This section describes how to migrate data from Kvrocks to GeminiDB Redis API.

# **Migration Principles**

The open-source tool kvrocks2redis is used to migrate data from Kvrocks to GeminiDB Redis API. At the code layer, Kvrocks namespace is adapted to the source GeminiDB Redis database.

The migration process consists of two phases: full migration and incremental migration. During full migration that is first performed, snapshots are created for Kvrocks and the corresponding data version (seq) is recorded. Then, the complete data files are parsed into Redis commands and written to GeminiDB Redis API. After the full migration is complete, the incremental migration starts. The migration tool cyclically sends PSYNC commands to Kvrocks and continuously forwards the obtained incremental data to GeminiDB Redis API.

# Usage Notes

 Kvrocks2redis needs to extract data from Kvrocks to local files, parse commands from the files, and send the commands to the target GeminiDB Redis instance. During this process, the performance of the source DB may be affected, but no data is compromised theoretically.

- If a fault occurs when the migration tool is running, the migration tool automatically stops to facilitate fault locating.
- For security purposes, GeminiDB Redis API does not provide database clearing commands. Ensure that no data exists in the database before the migration.

# Prerequisites

- Deploy the kvrocks2redis on an independent host.
- Ensure that the source DB, target DB, and migration tool can communicate with each other.
- Back up data of the source Kvrocks instance in advance.
- Clear all data on the destination GeminiDB Redis instance.

# Procedure

To migrate data from Kvrocks to GeminiDB Redis API, choose **Service Tickets** > **Create Service Ticket** in the upper right corner of the console and contact the customer service.

# Verifying Data Consistency After Migration

After the migration is complete, you can check data consistency.

# NOTICE

- Data has been migrated from Redis, or incremental migration has started.
- Redis-full-check must be deployed on the ECS, and the ECS is connected to the source and destination databases.
- During incremental migration, data may be inconsistent due to network latency between the source and destination databases. You are advised to stop writing data to the source and then verify data consistency.
- When Redis is used, an expiration time is usually set for keys. During migration, setting a key expiration time affects data consistency. Data may be inconsistent due to inconsistent expiration time.
- During migration, DTS writes temporary probing keys to Redis on the destination database. Non-service data may be detected during data verification, which is normal.

#### Procedure

- **Step 1** Log in to the ECS and ensure it is connected to the source and destination Redis databases.
- Step 2 Deploy redis-full-check.
- Step 3 Verify data.

/redis-full-check -s {Source IP address}:{Source port} -p {Source password} -t
{Destination IP address}:{Destination port} -a {Destination password} -m 1

Parameter	Description	Example Value
-S	Source Redis database address and port number	-s 10.0.0.1:6379
-р	Password of the source Redis database	-
-t	Destination GeminiDB Redis database address and port number	-t 10.0.0.2:6379
-a	Password of the destination GeminiDB Redis database	-
-m	<ol> <li>Verification mode:</li> <li>All key-value pairs</li> <li>Value length only</li> <li>Key integrity only</li> <li>All key values are verified, but only the length of big keys is verified.</li> <li>By default, the second verification mode is used.</li> </ol>	-m 1
-q	Maximum QPS. The default value is <b>15000</b> .	-q 5000
-d	Name of the file for saving the verification result. The default value is <b>result.db</b> .	-d result.db

 Table 4-32
 Parameter description

**Step 4** View the verification result file.

By default, three rounds of verification are performed and three verification result files are generated. Generally, you only need to view the last verification result file.

- Run the **sqlite3 result.db.3** command.
- Run the **select** \* **from key** command.
- Check whether there are abnormal keys.

Enter ".help" for usage hints.
sqlite> select * from key;
1 b string lack_target 0 1 0
2 c string lack target 0 1 0
3 a string lack target 0 1 0
sqlite>

----End

# 4.4.9 Migrating Data from Pika to GeminiDB Redis API

Pika is a persistent large-capacity Redis storage service. It breaks through the memory bottleneck of Redis due to the large amount of data. However, it is relatively weak in cluster management, and requires twemproxy or codis to shard static data. Compared with the Redis community edition, the database performance is significantly lowered because Pika stores all data in disks.

GeminiDB Redis API is a cloud-native NoSQL database with decoupled compute and storage and full compatibility with Redis. To ensure data security and reliability, it provides multi-copy, strict consistency based on a shared storage pool. It supports cold and hot data separation. Hot data can be read from the cache directly, improving read efficiency. RocksDB has been customized to allow the storage capacity to be scaled in seconds. A proxy is used to ensure that upperlayer applications are not affected by underlying sharding or scaling.

This section describes how to migrate data from Pika to GeminiDB Redis API.

# **Migration Principles**

The pika-port tool is used and acts as a slave node of Pika and data is migrated in master/slave replication mode. The master Pika node compares pika-port with its own binlog offset to determine whether to perform full migration or incremental migration. If full migration is required, the master Pika node sends the full data snapshot to pika-port, and pika-port sends the parsed snapshot data to GeminiDB Redis API. After the full migration is complete, incremental migration starts. pika-port parses the incremental data and sends the data to GeminiDB Redis API in the form of Redis commands.





# Usage Notes

- pika-migrate and pika-port act as the slave node of the source Pika and reads only full and incremental data without damaging your data.
- The master/slave synchronization process between the source DB and pikamigrate and pika-port is added, which may affect the performance of the source DB.
- Full and incremental migration can be performed without service interruption. Services are interrupted for a short period of time when services are switched over to GeminiDB Redis API.

# Prerequisites

Deploy the migration tool pika-port to ensure that the network connection between the source DB and target Pika instance is normal.

# Procedure

To migrate data from Pika to GeminiDB Redis API, choose **Service Tickets** > **Create Service Ticket** in the upper right corner of the console and contact the customer service.

# **Migration Performance Reference**

- Environment: Pika (single node) and pika-port are deployed on an ECS with 8 vCPUs and 32 GB memory on Huawei Cloud. The target DB is a three-node GeminiDB Redis instance with 8 vCPUs and 16 GB memory.
- Preset data: Use the memtier\_benchmark tool to preset 200 GB of data.
- Migration performance: about 50,000 QPS.

# 4.4.10 Migrating Data from SSDB to GeminiDB Redis API

SSDB is a high-performance NoSQL database written in C/C++. It is compatible with Redis APIs and supports multiple data structures, including key-value pairs, hashmap, sorted set, and list. SSDB is a persistent KV storage system and uses leveldb as the underlying storage engine. Its services directly interact with LevelDB. Operations such as compaction have direct impact on service read and write. GeminiDB Redis API is a cloud-native NoSQL database with decoupled compute and storage and full compatibility with Redis. To ensure data security and reliability, it provides multi-copy, strict consistency based on a shared storage pool. RocksDB is used as the storage engine. Compared with leveldb, RocksDB greatly improves performance, solves the problem that leveldb proactively restricts write, and implements cold and hot separation, reducing the impact of operations at the storage layer on performance.

This section describes how to migrate data from SSDB to GeminiDB Redis API.

# **Migration Principles**

ssdb-port acts as a slave node (replica) of the master node of the source SSDB database and migrates data through master/slave replication. Then, it parses and converts the obtained data into the format supported by Redis, and sends the data to the Redis instance specified in the configuration file. The following figure shows

the migration process. After the full synchronization is complete, the new data in SSDB is also synchronized to the Redis instance.





# **Usage Notes**

- As the slave node of the SSDB master node, ssdb-port reads only full and incremental data without damaging your data.
- The performance of the source SSDB is affected for running ssdb-port.
- Full migration and incremental migration can be performed without service interruption. After all data is migrated, services need to be stopped for a short period of time.

# Prerequisites

Create an ECS in the VPC where the GeminiDB Redis instance is located and deploy the migration tool ssdb-port to ensure that the source SSDB instance can communicate with the target GeminiDB Redis instance.

# Procedure

To migrate data from SSDB to GeminiDB Redis API, choose **Service Tickets** > **Create Service Ticket** in the upper right corner of the console and contact the customer service.

# Verifying Data Consistency After Migration

After the migration is complete, you can check data consistency.

#### NOTICE

- Data has been migrated from Redis, or incremental migration has started.
- Redis-full-check must be deployed on the ECS, and the ECS is connected to the source and destination databases.
- During incremental migration, data may be inconsistent due to network latency between the source and destination databases. You are advised to stop writing data to the source and then verify data consistency.
- When Redis is used, an expiration time is usually set for keys. During migration, setting a key expiration time affects data consistency. Data may be inconsistent due to inconsistent expiration time.
- During migration, DTS writes temporary probing keys to Redis on the destination database. Non-service data may be detected during data verification, which is normal.

#### Procedure

- **Step 1** Log in to the ECS and ensure it is connected to the source and destination Redis databases.
- Step 2 Deploy redis-full-check.
- Step 3 Verify data.

/redis-full-check -s {Source IP address}:{Source port} -p {Source password} -t
{Destination IP address}:{Destination port} -a {Destination password} -m 1

Parameter	Description	Example Value
-S	Source Redis database address and port number	-s 10.0.0.1:6379
-р	Password of the source Redis database	-
-t	Destination GeminiDB Redis database address and port number	-t 10.0.0.2:6379
-a	Password of the destination GeminiDB Redis database	-

Parameter	Description	Example Value
-m	Verification mode:	-m 1
	1. All key-value pairs	
	2. Value length only	
	3. Key integrity only	
	<ol> <li>All key values are verified, but only the length of big keys is verified.</li> </ol>	
	By default, the second verification mode is used.	
-q	Maximum QPS. The default value is <b>15000</b> .	-q 5000
-d	Name of the file for saving the verification result. The default value is <b>result.db</b> .	-d result.db

**Step 4** View the verification result file.

By default, three rounds of verification are performed and three verification result files are generated. Generally, you only need to view the last verification result file.

- Run the **sqlite3 result.db.3** command.
- Run the **select** \* **from key** command.
- Check whether there are abnormal keys.



----End

# Migration Performance Reference

- Environment: The source SSDB and ssdb-port are deployed on an ECS with 4 vCPUs and 16 GB memory. The destination is a three-node instance with 8 vCPUs and 16 GB memory.
- Preset data: Use the memtier\_benchmark tool to preset 100 GB of data.
- Migration performance: about 3000 QPS.

# 4.4.11 Migrating Data from LevelDB to GeminiDB Redis API

LevelDB is an open-source, persistent, and single-node KV database engine. It provides high random write performance and sequential read/write performance,

and applies to write intensive scenarios. LevelDB does not provide the C/S network structure and must be deployed on the same server as your services. Compared with RocksDB developed based on LevelDB, LevelDB has many disadvantages. For example, it cannot make the most out of multi-core servers, and does not support TB-level data storage, and cannot read data from HDFS.

GeminiDB Redis API uses RocksDB as the storage engine. It is compatible with the Redis protocol and provides various data types to meet LevelDB requirements. In addition, RocksDB has been customized to allow storage to be scaled in seconds, making it easy to migrate LevelDB workloads to the Redis ecosystem. You do not need to migrate data during scaling.

This section describes how to migrate data from LevelDB to GeminiDB Redis API.

# **Migration Principles**

- Use the self-developed migration tool leveldb-port to deploy LevelDB on the same server as your services, prepare the configuration file, and start the migration task to automatically complete full and incremental migration.
- The full migration process is efficient. It takes a snapshot of the LevelDB data, scans the entire database, packs the data into a format that can be identified by GeminiDB Redis API, and then sends the data to GeminiDB Redis API.
- During incremental migration, the WAL file of LevelDB and the LevelDB operations are parsed, and the keys in the WAL file are sharded and sent by multiple threads.

# Usage Notes

- The migration tool needs to be deployed on the source DB, which consumes certain performance. You can modify the configuration file to control the performance.
- During the migration, the source data file of LevelDB is read-only. There is no risk of data damage.
- Services do not need to be stopped during the migration.
- If a fault occurs during the migration, clear the GeminiDB Redis instance and restart the migration.

# Procedure

To migrate data from LevelDB to GeminiDB Redis API, choose Service Tickets > Create Service Ticket in the upper right corner of the console and contact the customer service.

# Migration Performance Reference

- Environment: The source LevelDB and leveldb-port are deployed on a Huawei Cloud ECS with 4 vCPUs and 16 GB memory. The target DB is a three-node GeminiDB Redis instance with 2 vCPUs and 8 GB memory.
- Full migration: 10 GB data is preconfigured, and the migration speed is about 8 MB/s.
- Incremental migration: Set the value to 1 KB and the migration speed to 7,000 QPS.

# 4.4.12 Migrating Data from RocksDB to GeminiDB Redis API

RocksDB is a persistent key-value store, single-node DB engine developed by Facebook based on LevelDB. It has powerful sequential read/write and random write performance. Compared with LevelDB, RocksDB has many optimizations. Its performance is greatly improved and the problem that LevelDB proactively restricts write operations is solved. As a DB engine, RocksDB does not provide the C/S network structure. It must be deployed on the same server as your services.

GeminiDB Redis API uses RocksDB as the storage engine and is compatible with the Redis protocol, meeting the usage requirements of RocksDB. In addition, RocksDB has been customized to allow storage to be scaled in seconds, making it easy to migrate RocksDB workloads to the Redis ecosystem. You do not need to migrate data during scaling.

This section describes how to migrate data from RocksDB to GeminiDB Redis API.

# **Migration Principles**

- Use the self-developed migration tool rocksdb-port to deploy RocksDB on the same server as your services, prepare the configuration file, and start the migration task to automatically complete full and incremental migration.
- The full migration process is efficient. It takes a snapshot of the RocksDB data, scans the entire database, packs the data into a format that can be identified by GeminiDB Redis API, and then sends the data to GeminiDB Redis API.
- During incremental migration, the WAL file of RocksDB and the RocksDB operations are parsed, and the keys in the WAL file are sharded and sent by multiple threads.

# Usage Notes

- The migration tool needs to be deployed on the source DB, which consumes certain performance. You can modify the configuration file to control the performance.
- During the migration, the source data file of RocksDB is read-only. There is no risk of data damage.
- Services do not need to be stopped during the migration.
- If a fault occurs during the migration, clear the GeminiDB Redis instance and restart the migration.

# Procedure

To migrate data from RocksDB to GeminiDB Redis API, choose **Service Tickets** > **Create Service Ticket** in the upper right corner of the console and contact the customer service.

# 4.4.13 Migrating Data from Amazon ElastiCache for Redis to GeminiDB Redis API

# **Migration Principles**

After backing up and exporting an RDB file in an Amazon ElastiCache for Redis database, you can use Redis-Shake to restore data to a GeminiDB Redis instance.

# Usage Notes

- AWS does not support the **psync/sync** command and data cannot be incrementally migrated.
- Before the migration, ensure that the network between the ECS where Redisshake is deployed and the destination GeminiDB Redis is normal.
- Ensure that the security group configuration on the source and target ends is enabled.

# Procedure

**Step 1** Deploy the required migration tool.

1. Obtain Redis-Shake.

D NOTE

Download the Redis-Shake release package and decompress it.

2. Modify the **Redis-Shake.conf** configuration file and configuring the following items:

**log.level** = **info** #Default log level. A printed INFO log contains migration progress information, based on which you can judge whether the migration is complete.

source.rdb.input = /xx/xx.rdb # Absolute path of the source RDB file

target.address = <host>:6379 # Destination instance IP address

**target.password\_raw = \*\*\*\*\*** # Password for logging in to a destination instance

**target.version = 5.0** # Version of the destination Redis instance

target.type = standalone # Destination instance type

**target.db** = **0** #Data is migrated to the specified database of the destination GeminiDB Redis. The default value is **db0**.

big\_key\_threshold = 1 #Setting the big key threshold

3. Specify whether data of the destination is overwritten.

key\_exists = none

### D NOTE

If there are duplicate keys on the source and destination, specify whether data of the destination is overwritten. The options are as follows:

- **rewrite** indicates that the source overwrites the destination.
- **none** indicates that the migration process exists once duplicate keys are detected.
- **ignore** indicates that keys in the source are retained and keys in the destination are ignored. This value does not take effect in rump mode.

**none** is recommended. There will be no duplicate data because the source is an RDB file. If the migration exits unexpectedly, you can choose **Service Tickets > Create Service Ticket** in the upper right corner of the console and contact the customer service.

#### Step 2 Migrate data.

Run the following command to start migration:

#### ./redis-shake.linux -conf=redis-shake.conf -type=restore

**NOTE** 

Use the restore mode because the source is an RDB file.

Stop the migration process after the migration is complete.

Step 3 Verify data.

Data is obtained from the RDB file. Therefore, you need to check the GeminiDB Redis data at the destination end from the service perspective.

----End

# Verifying Data Consistency After Migration

After the migration is complete, you can check data consistency.

#### NOTICE

- Data has been migrated from Redis, or incremental migration has started.
- Redis-full-check must be deployed on the ECS, and the ECS is connected to the source and destination databases.
- During incremental migration, data may be inconsistent due to network latency between the source and destination databases. You are advised to stop writing data to the source and then verify data consistency.
- When Redis is used, an expiration time is usually set for keys. During migration, setting a key expiration time affects data consistency. Data may be inconsistent due to inconsistent expiration time.
- During migration, DTS writes temporary probing keys to Redis on the destination database. Non-service data may be detected during data verification, which is normal.

#### Procedure

**Step 1** Log in to the ECS and ensure it is connected to the source and destination Redis databases.

#### Step 2 Deploy redis-full-check.

#### **Step 3** Verify data.

/redis-full-check -s {Source IP address}:{Source port} -p {Source password} -t
{Destination IP address}:{Destination port} -a {Destination password} -m 1

Parameter	Description	Example Value
-S	Source Redis database address and port number	-s 10.0.0.1:6379
-р	Password of the source Redis database	-
-t	Destination GeminiDB Redis database address and port number	-t 10.0.0.2:6379
-a	Password of the destination GeminiDB Redis database	-
-m	<ol> <li>Verification mode:</li> <li>All key-value pairs</li> <li>Value length only</li> <li>Key integrity only</li> <li>All key values are verified, but only the length of big keys is verified.</li> <li>By default, the second verification mode is used.</li> </ol>	-m 1
-q	Maximum QPS. The default value is <b>15000</b> .	-q 5000
-d	Name of the file for saving the verification result. The default value is <b>result.db</b> .	-d result.db

Table 4-34 Parameter desci	ription
----------------------------	---------

**Step 4** View the verification result file.

By default, three rounds of verification are performed and three verification result files are generated. Generally, you only need to view the last verification result file.

- Run the **sqlite3 result.db.3** command.
- Run the **select \* from key** command.
- Check whether there are abnormal keys.

Enter ".help" for usage hints.
sqlite> select * from key;
1 b string lack_target 0 1 0
2 c string lack_target 0 1 0
3 a string lack_target 0 1 0
sqlite>

----End

# 4.4.14 Verifying Redis Data Consistency After Migration

After the migration is complete, you can check the consistency of Redis data.

# Usage Notes

- The Redis migration has been completed, or the incremental migration has started.
- Redis-full-check must be deployed on the ECS, and the ECS is connected to the source and destination databases.
- If the migration task is in the incremental state, data consistency cannot be ensured due to network latency between the source and target ends. If conditions permit, you are advised to stop writing data to the source end and then perform the verification.
- When Redis is used, an expiration time is usually set for keys. During migration, setting a key expiration time affects data consistency. If verification results show that data is inconsistent, the possible cause is that the key expiration time is inconsistent.
- During the migration, DTS writes temporary probing keys to Redis on the destination end. Non-service data may be detected during data verification, which is normal.

# Procedure

- **Step 1** Log in to the ECS and ensure that the ECS can connect to the source and destination Redis databases.
- Step 2 Deploy Redis-Full-Check.
- **Step 3** Verify data.

/redis-full-check -s {Source IP address}:{Source port} -p {Source password} -t
{Destination IP address}:{Destination port} -a {Destination password} -m 1

Parameter	Description	Example Value
-S	Source Redis connection address and port number	-s 10.0.0.1:6379
-р	Password of the source Redis database.	-

Table 4-35	Parameter	description
------------	-----------	-------------

Parameter	Description	Example Value
-t	Destination Redis connection address and port number	-t 10.0.0.2:6379
-a	Password of the destination Redis database.	-
-m	<ul><li>Verification mode:</li><li>1. Verify all key-value pairs.</li><li>2. Only value length is verified</li></ul>	-m 1
	<ol> <li>Only key integrity is verified.</li> </ol>	
	<ol> <li>All key values are verified, but only the length of big keys is verified.</li> </ol>	
	By default, the second verification mode is used.	
-q	Maximum QPS. The default value is <b>15000</b> .	-q 5000
-d	Name of the file for saving the verification result. The default value <b>is result.db</b> .	-d result.db

**Step 4** View the verification result file.

By default, three rounds of verification are performed and three verification result files are generated. Generally, you only need to view the last verification result file.

- Run the **sqlite3 result.db.3** command.
- Run the **select** \* **from key** command.
- Check whether abnormal keys exist.

```
Enter ".help" for usage hints.
sqlite> select * from key;
1|b|string|lack_target|0|1|0
2|c|string|lack_target|0|1|0
3|a|string|lack_target|0|1|0
sqlite>
```

----End

# 4.5 Instance Management

# 4.5.1 Managing Sessions of a GeminiDB Redis Instance

# **Scenarios**

You can manage sessions of a GeminiDB Redis instance on the console.

# Usage Notes

- To use this function, you need to update instances of earlier kernel minor versions by following **4.6.1 Upgrading a Minor Version**.
- Redis Cluster GeminiDB Redis instances do not support sessions.
- This function is available only when the target instance node is normal.
- Session details are displayed after a client connects to the instance.
- Redis Cluster GeminiDB Redis instances do not support this function.

### 

Killing a session will disconnect applications.

# **Viewing Instance Sessions**

- 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 navigation pane on the left, choose **Sessions**.
- **Step 5** On the **Sessions** page, select a node to view its session information. **Table 4-36** lists the session parameters.

#### Figure 4-71 Sessions

Node geminide-		~										
Node Name gemini	a	Node ID 0443c										
Sessions												
KII Selected	KILAI Node Sessions	Kill All Instance Sessions	Export All Node Sessions	Export All Instance Sessions	Auto Refresh	105	× [	C. Enter an address or select another filter filter.	on the drop-down list to see	arch.		C
addr 8	Иθ	name 🖯	and @	491 0	idle ()		¢ ⊕	14	sub	psub	multi	
					<u> </u>							
					Man share and balance							

#### Table 4-36 Instance session parameters

Parameter	Description
addr	IP address and port number of a client
id	Session ID

Parameter	Description
name	Connection name
cmd	Last executed command
age	Connection duration, in seconds
idle	Idle duration, in seconds
db	ID of a database that is being used by the client, for example, DB0, DB1, and DB2
fd	File descriptor for sockets
sub	Number of subscribed channels
psub	Number of subscribed modes
multi	Number of commands executed in a transaction

#### Step 6 You can select By source or By database. Table 4-37 lists the parameters.

- **By source**: Sessions on clients connected to a node. IP addresses of top ten clients are displayed and ordered based on sessions on each client. There may be clients with the same quality of sessions.
- **By database**: IP addresses of clients connected to each database and sessions of each client. Top ten clients are displayed and ordered based on their sessions. There may be clients with the same quality of sessions.

#### Figure 4-72 Session statistics



Parameter	Description
ltem	Total and active clients
	Total clients: Total client connections
	Active clients: Active client connections
Result	Statistical results
Source	Client IP address
DB	ID of a GeminiDB Redis database, for example, DB0, DB1, and DB2
Total	Total client connections

----End

# Auto Refresh

**Auto Refresh** is disabled by default. You can toggle it on. After **Auto Refresh** is enabled, the page is refreshed every 10s by default. You can set the interval to 10s, 30s, or 60s.

#### Figure 4-73 Auto Refresh

Node geninido-		~									
Node Name geminic	b-	Node ID 0443c									
Sessions											
Kill Selected	Kill All Node Sessions	Kill All Instance Sessions	Expert All Node Sessions	Export All Instance Sessions	Auto Refresh	1ts v 0. Ente	r an address or select another B	ter from the drop-down list to sear	ch.		C
addr 0	14 0	name ()	end ()	aga (j)	ide ()	ab ()	nd	sub	paub	matt	
					1:1						

# **Killing Sessions**

You can click Kill Selected, Kill All Node Sessions, or Kill All Instance Sessions.

Killing a session will disconnect applications.

#### Figure 4-74 Killing sessions



# **Exporting Sessions**

You can click **Export All Node Sessions** or **Export All Instance Sessions**.

Figu	re 4-7!	5 Expor	ting se	essions							
Node geminick-		×.									
Node Name geminide		Node ID 0443cf									
Sessions Kill Selected	KILAI Node Sessions	Kill All Instance Sessions	Export Al Node Sessions	Expert All Instance Sessions	Auto Refresh 🕥	10s	r an address or select another fit	or from the drop-down list to search			C
	нө	neme O	ond $\Theta$	age 0	idle 🖯	db ⊕	fid	sub	peub	multi	
					No data available						

# 4.5.2 Renaming Commands of a GeminiDB Redis Instance

# Scenarios

Commands of a general-purpose GeminiDB Redis instance can be renamed. To prevent data loss, instance restart, and performance jitter caused by misoperations, you can rename high-risk commands.

# Usage Notes

- To rename commands, you need to update instances of earlier kernel minor versions by following **4.6.1 Upgrading a Minor Version**.
- High-risk commands can be renamed only when your instance is in the **Available** state.
- Commands of a Redis Cluster instance cannot be renamed.
- Redis Cluster GeminiDB Redis instances do not support this function.

# 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** > **Rename High-risk Command** in the **Operation** column.

#### Figure 4-76 Command renaming



**Step 4** Select the required commands and disable them. For involved parameters, see **Table 4-38**.

Parameter	Description
flushall	Clears all buffers.
flushdb	Deletes all keys in the current database.
hgetall	Returns all fields and values in the hash table.
hkeys	Returns all keys in the hash table.
hvals	Returns all values in the hash table.
keys	Finds all keys that match a given pattern.
smembers	Returns all members in a set. Any set without keys is considered empty.
New Name	Name of the command that takes effect currently. The name can include 0 to 30 characters. Command names are case-insensitive and can contain only letters, digits, hyphens (-), and underscores (_). If the name is left blank, the command is disabled. The new name must be unique.

Table 4-38 Parameter description

Х

Parameter	Description
Disabled	The <b>Disabled</b> option is toggled off by default. You can click to toggle on it. If a command is disabled, its new name is empty.

**Step 5** Modify parameter information and click **OK**.

Figure 4-77 Renaming commands

Original Name	New Name	Disabled
flushall	flushall	
flushdb	flushdb	
hgetall	hgetall	
hkeys	hkeys	
hvals	hvals	
keys	keys	
smembers	smembers	

**Step 6** Check the renaming result.

- You can view new command names on the **Rename High-risk Command** page.
- After the renaming is complete, original commands become invalid and you need to use new commands to perform operations.

----End

# 4.5.3 Clearing GeminiDB Redis Instance Data

# Scenarios

GeminiDB Redis API allows you to clear all data in an instance or data in a specified database to release instance space.

# Usage Notes

- Cleared data cannot be restored. Back up the instance before you clear it. For details, see **Creating a Manual Backup**.
- After you select **Specified DB**, only the data in the selected database is cleared.
- Redis Cluster GeminiDB Redis instances do not support this function.

# 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.
- **Step 4** In the upper right corner of the **Basic Information** area, click **Clear Data**.

#### Figure 4-78 Clearing data

<   geminidb,	Austation	G Feedbeck Outch Links Clear Data	Login VestMetic	Rest Passent Restart		
Rasic Information						
Dackape & Rentmations	Basic Information					
Node Management	D8 Instance Name	DB Indance D	Starage Type	Product 1	ype	
Accounts	pemiridb_		Characia	Standard		
Slow Query Loga						
Anditopp	Status	Regim	A2	D6 Instan	ce Type	
Parameters	O Available		822	Ploy du	law'	
Metrica	Enterprise Project	Mandananan Manhar (T)				
Sectors	extent					

• If you want to clear all data in the instance, select **All** and click **OK**.

### Figure 4-79 Clearing all data

#### Clear Data

<b>A</b>	FLUSHDB will be execute be restored.	ed to clear all data of the	instance. Cleared data cannot
Туре	All data	Specified DB	
			OK Cancel

• If you need to clear data of a specified database, select **Specified DB**, enter the number of the target database, and click **OK**.

#### **NOTE**

The number of the database must be an integer ranging from 0 to 65535.

#### Figure 4-80 Clearing data of a specified database

Clear	Data	$\sim$
A Fi	LUSHDB will be executed to clear all data in the specified database. Cleared ata cannot be restored.	
Туре	All data Specified DB	
DB No.		
	OK Cancel	)

----End

# 4.5.4 Instance Lifecycle Management

# 4.5.4.1 Restarting a GeminiDB Redis Instance

# **Scenarios**

You may need to restart an instance for routine maintenance.

# **Usage Notes**

- Only instances in states **Available**, **Abnormal**, or **Checking restoration** can be restarted.
- After you restart an instance, all nodes in the instance are also restarted.
- Restarting an instance will interrupt services. Wait until off-peak hours and ensure that your application can re-connect.
- 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*.

# 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 Restart or choose More > Restart in the Operation column.

Alternatively, locate the instance you want to restart and click its name. 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** or **Immediate**.

• Instance with classic storage

For a GeminiDB Redis instance with classic storage, you can restart nodes one by one or all at once.

#### Figure 4-81 Restarting an instance

Restart DB Instance								
Restart this instance?								
Restart all nodes at once     Restart	nodes in sequence							
DB Instance Name	Status							
	Available							
Scheduled Time Immediate	During maintenance window							
A This DB instance will not be available when it is being restarted.								
	Yes No							

• Instance with cloud native storage

For GeminiDB Redis instances with cloud native storage, click **Yes** or **Immediate**.

#### Figure 4-82 Restarting an instance

Restart DB Instance								
Restart this instance?								
DB Instance Name	Status							
	Available							
Scheduled Time Immediate	During maintenance window	0						
▲ This DB instance will not be available when it is being restarted.								
	Ye	es No						

----End

# 4.5.4.2 Exporting Instance Information

### **Scenarios**

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

# Exporting All Instance Information

- 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  $\Box$  in the upper right corner of the page. By default, information about all DB 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 is generated locally.

----End

# **Exporting Information About Selected Instances**

Step 1 On the Instances page, select the instances that you want to export or search for

required instances by project, compatible API, name, ID, or tag and click  $\Box$  in the upper right corner of the page. 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 is generated locally.

----End

# 4.5.4.3 Deleting a Pay-per-Use Instance

# Scenarios

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

# Precautions

- Instances that an operation is being performed on 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 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.
- A deleted instance will be retained in the recycle bin for a period of time after being released, so you can rebuild the instance and restore data 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 that you want to delete and in the **Operation** column choose **Delete** or **More** > **Delete**.
- 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

# 4.5.4.4 Recycling a GeminiDB Redis Instance

Unsubscribed yearly/monthly instances and deleted pay-per-use instances are moved to the recycle bin and can be restored.

# **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.
- 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.

X

# Modifying the Recycling Policy

#### NOTICE

You can modify the retention period, and the new retention period only takes effect for the instances that are deleted after the modification.

- 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 4-83 Modifying the recycling policy

#### Modify Recycling Policy

Retention Period	− 1  + days					
	You can change the retention period to between 1 and 7 days. The changes only apply to the DB instances deleted after the changes.					
	You can put up to 100 instances into the recycle bin. If the maximum number of instances is reached, you cannot put instances into the recycle bin anymore.					
	OK Cancel					

----End

#### **Rebuilding an Instance**

You can rebuild 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 4-84 Rebuilding an instance

 Off Indext Name()
 Off Indext Name()
 Off Indext Name()
 Off Indext Name()
 One off Ind

**Step 4** On the displayed page, set required parameters (you are advised to set the specifications to be the same as those of the original instance) and submit the rebuilding task.

----End

# 4.6 Modifying Instance Settings

# 4.6.1 Upgrading a Minor Version

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

After a new patch version involving performance improvement, new functions, or problem rectification is released, you can upgrade your instance to the latest version at a proper time based on service requirements.

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 4-85 Patch installation

Name/ID \ominus	DB Instance Type	Compatible API	Storage Type	Status 🖨	Specifications	Storage Spa	ace	Load balan	Enterpris	Billing M	Operation	
	Primary/Standby	Redis 6.2 Upgrade Minor Version		Available	2 vCPUs Sta 2 nodes	0%	0/24GB	-	default	Pay-per-Use Created o	Log In Change to Yearly/Monthly	More ~

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 is a new patch released.
- If the database version is a risky version, the system prompts you to upgrade the database patch.
- Upgrading the minor version of an instance will restart each node of the instance in sequence. When a node is being restarted, its services will be taken over by another node. Each takeover will interrupt services for 3 to 5 seconds. So, perform an upgrade during off-peak hours and enable automatic reconnection so that each node can be reconnected immediately after being restarted.
- Upgrading basic components takes about 15 minutes. The duration of upgrading data components depends on the number of nodes. The upgrade of each node takes about 1 to 2 minutes.
- The system automatically checks the minor version of the instance. If the **Upgrade Minor Version** button does not exist on the console, the current minor version is the latest version.
- After GeminiDB Redis API is upgraded, the compatible Redis version may be upgraded too. For example, after upgrade, Redis 5.0 is compatible with 6.2. A later version is compatible with earlier versions. After upgrade, the service usage remains unchanged, and Redis 6.2 commands are supported.
- Services are unavailable during the parallel upgrade. You are advised to perform the parallel upgrade during off-peak hours. The parallel upgrade takes about 17 to 20 minutes, regardless of the number of 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 you want to upgrade and click Upgrade Minor Version in the Compatible API column.

Figure 4-86 Patch installation

Name/ID 😔	DB Instance Type	Compatible API	Storage Type	Status \ominus	Specifications	Storage Sp	ace	Load balan	Enterpris	Billing M	Operation
	Primary/Standby	Redis 6.2 Upgrade Minor Version		Available	2 vCPUs Sta 2 nodes	0%	0/24GB	-	default	Pay-per-Use Created o	Log In Change to Yearly/Monthly More $\sim$

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

#### Figure 4-87 Patch installation

Specification Information				
Specifications	Nodes	Compatible API		
2 vCPUs Standard Change	2	Redis 6.2 Upgrade Minor Versio		
Total Storage Space				
Not encrypted	Scale Autoscaling			
Used 0.00/24 GB	0%			

Step 4 In the displayed dialog box, click OK.

- You can upgrade nodes one by one or all at once.
- You can select **Immediate** or **During maintenance window** for **Scheduled Time**. If **During maintenance window** is selected, the scheduled upgrade task will be executed in the next time window.

Figure 4-88 Confirming information

Upgrade Minor Version

<ol> <li>Notes:</li> <li>Upgrading the minor version of an instance will restart each node of the instance in sequence. When a node is being restarted, its services will be taken over by another node. Each takeover will interrupt services for 3 to 5 seconds. So, perform the upgrade during off-peak hours and enable automatic reconnection so that each node can be reconnected immediately after being restarted.</li> <li>Upgrading basic components takes about 15 minutes. Upgrading data components takes about 1 to 2 minutes, which depends on how many nodes there are.</li> </ol>				
Upgrade in sequence Upgrade all nodes at once				
DB Instance Name	Status			
geminidb-5313-leiyufei-redis580	<ul> <li>Available</li> </ul>			
Scheduled Time Immediate	During maintenance window	0		
		OK Cancel		

**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 to **Available**.

----End

# 4.6.2 Modifying a GeminiDB Redis Instance Name

# **Scenarios**

This section describes how to modify the name of a GeminiDB Redis instance.

# 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  $\swarrow$  next to the target instance name and change it.
  - To submit the change, click **OK**.
  - To cancel the change, click **Cancel**.

#### D 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  $\stackrel{\checkmark}{=}$  next to **DB Instance Name** and change the instance name.
  - To submit the change, click  $\stackrel{\checkmark}{\sim}$
  - To cancel the change, click  $\times$  .

# 

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** Check the results on the **Instances** page.

----End

# 4.6.3 Changing the Administrator Password of a GeminiDB Redis Database

# Scenarios

For security reasons, regularly change your administrator password.

# Precautions

- You can reset the administrator password only when the **instance status** is **Available**, **Backing up**, or **Scaling up**.
- 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*.

# 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 administrator 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 any two of uppercase letters, lowercase letters, digits, and the following special characters:  $\sim!$  @#%^\*-\_=+?\$()&

**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, locate the instance whose administrator password you want to reset and click its name. The **Basic Information** page is displayed.
- 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 any two of uppercase letters, lowercase letters, digits, and the following special characters:  $\sim!$  @#%^\*-\_=+?\$()&

**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

## **Scenarios**

You can change the vCPUs and memory of all nodes. You can change the vCPUs and memory of your instance as needed. If an instance is overloaded and compute resources need to be added urgently, you are advised to add compute nodes first.

#### Usage Notes

- During online specification change, second-level intermittent disconnection occurs once when the change is performed on a single node. Therefore, the entire instance is intermittently disconnected for several times. The client must have an automatic reconnection mechanism. You are advised to perform the specification change during off-peak hours.
- 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.
- After specifications of a standard instance with cloud native storage are changed, the system automatically adjusts the storage to the number of shards multiplied by shard specifications (GB).

## 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 Specifications in the Operation column.
  - Instance with classic storage

Figure 4-89 Changing specifications



• Figure 4-90 Changing specifications

 NameD 0
 DB Indiano Type
 Competition
 Name 0
 Specification
 NameD 0
 Last Indiano rations
 NameD 0
 Competition
 Last Indiano rations
 NameD 0
 Competition
 Last Indiano rations
 NameD 0
 Competition
 Competition</t

In the **DB Information** area on the **Basic Information** page, click **Change** under **Node Specifications**.

• Instance with classic storage

#### Figure 4-91 Changing specifications

Specification Information	l	
Node Specifications		N
2 vCPUs Standard Change		
Total Storage Space		
Not encrypted	Scale Autoscaling	)
Used 0/16 GB	0.019	5
PITR Storage Used 0 GB		

• Capacity-oriented instance with cloud native storage

#### Figure 4-92 Changing specifications

Specif	fication Information	
Specific	cations	Node
1 vCPU	4 GB Change	2
Total St	orage Space	
		Scale
Used	0/200 GB	0%

- **Step 4** On the displayed page, select a specification change mode and required specifications, and click **Next**.
  - Online change: During the change, instance nodes are upgraded in rolling mode, which has the minimum impact on services. The change duration is positively related to the number of nodes. Each node takes about 5 to 10 minutes. If there are a large number of nodes, wait patiently.
  - Offline change: During offline change, all nodes are changed concurrently, which interrupts services for about 10 to 20 minutes. Exercise caution when performing this operation. For your online production services, you are advised to perform the change online.
  - Instance with classic storage

#### Figure 4-93 Changing specifications

Current Configuratio	on			
DB Instance Name			Node Specifications	geminidb.reds.large.4 ( 2 vCPUs Standard
DB Instance ID			Current Nodes	3
Billing Mode	Pay-per-use		Current Storage	18 GB
Change Mode		Online Office		
		to Services may be interrupted multiple times, for several seconds each time. Noise sure that your stient supports automatic reconnection, and change instan	ce specifications during off-	peak hours. The time required depends on the number of instance nodes whose specifications are to be changed. Each node takes about 5 to 10 minutes.
		vOPUs		Flavor Name
		0.5 vCPUs (recommended for tasting)		geninido redis small 4
		0 1vCPU Special offer		geninido redis medium 2
		O 2VCR/6 Enhanced		geminido resta large 8
		O 4vCPUs Standard		geminido realiz-starge, 4
		4vCRUs Enhanced		geninido redis xlerge 8
New Node Cessific	ntinen	8 vCPUs Standard		geminido redis Zolarge 4
New Node Specific	cauons	B vCPUs Ethanoad		geminido redis 2clarge 8
		O 15 KCPUs Enhanced		geminico redis. Adarge 8
		O 18 vCPUs Standard		geminico radiz. Adarge 4
		22 vCPUs Standard		geminido redis Bolarge 4
		O 32 vCPUs Enhanced		geminido redis Bolarge 8
		New Specifications geminiduareds.smail.4 (0.5 vCPUs		
		Fautocooling is enabled, changing specifications of the current instance will change the storage limit of the instance to the maximum storage supported by the ne	v specifications.	
Scheduled Time		Immediate During maintenance window		

• Capacity-oriented instance with cloud native storage

#### Figure 4-94 Changing specifications

change opechication	is 🕜			
Current Configuratio	n			
DB Instance Name			Node Specifications	geminidb.redis-geminifs.medium.4   1 vCPU   4 GB
DB Instance ID			Current Nodes	2
Billing Mode	Pay-per-use		Storage	200 GB
		Press Nove		
		Flavor Name		VCPU   Wemory
		<ul> <li>geminidt redis-geminifs large 4</li> </ul>		2 vCPUs   8 GB
		geminids.redis-geminifs.xlarge.4		4 vCPUs   16 GB
		geminidb.redis-geminifs.2xlarge.4		8 vCPUs   32 GB
New Specifications		geminidb redis-geminids.4xlarge.4		16 vCPUs   64 GB
		geminidb.redis-geminifs.8xlarge.4		32 vCPUs   128 GB
		Currently selected geminids.redis-geminids.large.4   2 vCPUs   8 GB		
		Services may be interrupted multiple times, for several seconds each time. Make sure that your client supports automo	fic reconnection, and change in	stance specifications during of-peak hours. The time required depends on the number of instance nodes whose specifications are to be
		changed. Each node takes about 5 to 10 minutes.		
		After the instance class is changed, some associated parameters for the new instance class are automatically change	d to the default values.	
Scheduled Time		Immediate During maintenance window ③		

• Standard instance with cloud native storage

#### Figure 4-95 Changing specifications

Change Specifications (9)

Current Configuration						
DB Instance Name		Node Specifications	perminido.readis-germinifs.alarge.4   4 vCPUs   16 GB			
DB Instance ID		Current Nodes	3			
		Storage	5,000 GB			
	Flavor Name		vCPU   Memory			
	geninida seda-periinda Jarge 4		2 vCPUs   8 08			
New Specifications	Currently selected geminido redis-geminido lerge 4   2 xCPUs   8 GB					
	Services may be interrupted multiple times, for serveral second each time. Make user that your cleart supports automatic econnection, and charge instance specifications during off-seak hours. The time required depends on the number of instance index whose specifications are to be charged. Each mode tables about 5 to 10 minutes.					
	After the instance class is changed, some associated parameters for the new instance class are automatically changed to the default values.					
Scheduled Time	ternelate					

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

- 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 the Maintenance Period

- **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 On the Basic Information page, locate Maintenance Window and click Change.

#### Figure 4-96 The change button

Basic Information			
D6 Instance Name gemindo-	DB Instance ID	Storage Type Classic	Product Type Standard
Status O Available	Region	AZ 824,822,823	DB Instance Type Proxy cluster
Maintenance Window			

**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 4-97 Changing a maintenance window

Change Maintenance Window			×
Time Zone	GMT+08:00		
Maintenance Window	10:00 - 14:00	~	
	▲ Changing the maintenance v execution of scheduled tasks in	vindow will not affect the the original maintenance window	-
		OK Cancel	

**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 4-98 Querying a task

Instant Tasks Scheduled Tasks				
Last7 days v	. Select one or more filters from the pop-up lists. If you enter a keyword without a filt	r applied, the system will search for all task names	matching this keyword.	
Task Name/Task ID	Status	DB Instance Name/ID	Created	Completed
CreateInstance	C Running (11% complete)	geminidb 35976b49	Feb 25, 2025 10.01:55 GMT+08:00	Feb 25, 2025 10:01:55 GMT+08:00
CreateInstance	Running (33% complete)	gaminidb 8b513e3i	Feb 25, 2025 09 56 35 GMT+08:00	Feb 25, 2025 09:58:36 GMT+08:00
Modily Part	O Completed	gaminidb 89ed425x	Feb 25, 2025 09 47:52 GMT+08.00	Feb 25, 2025 09:48:59 GMT+08.00

----End

Task Center

## **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 4-99 Canceling a task

Task Co	enter								
Insta	nt Tasks	Scheduled Tasks							
Q	Select one or n	nore filters from the pop-up list	s. If you enter a keyword w	ithout a filter applied, the sys	item will search for all	instance id matching this keyword.			Q
Tas	k Name/Task II	0	Status	DB Instance Name/ID	Compatible API	Created	Execution Time Period (GMT+08:00)	Operation	
Cha	inging a DB inst	ance class	C To be executed		Redis	Jun 26, 2024 14:53:56 GMT+08:00	Jun 27, 2024 10:00:00 - Jun 27, 2024 14:00:00	Cancel	

#### **Step 4** View the result.

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

Figure 4-100 Viewing cancelled tasks

Tas	ik Center						
	Instant Tasks Scheduled Tasks	If you enter a keyword with	out a filter applied, the syst	om will search for all instar	ice id matching this keyword.		a
	Task Name/Task ID	Status	DB Instance Name/ID	Compatible API	Created	Execution Time Period (GMT+08:00)	Operation
	Changing a DB instance class 051db44a-9e11-4922-995e-a4077171c575	Canceled	geminidb-redis-lss eff016e6a1144ca89	Redia	Jun 26, 2024 14:53:56 GMT+00:00	Jun 27, 2024 10:00:00 - Jun 27, 2024 14:00:00	-

----End

# 4.6.6 Adding and Deleting Instance Nodes

## 4.6.6.1 Overview

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

Table	4-39	Scaling	methods
-------	------	---------	---------

Method	Supported Instance Type	
4.6.6.2 Adding Instance Nodes	• Capacity-oriented cluster instance with cloud native storage	
	Proxy cluster	
	Redis Cluster	
4.6.6.3 Adding Instance Shards	Standard cluster instance with cloud native storage	
4.6.6.4 Deleting Instance Nodes	Standard cluster instance with classic storage <ul> <li>Proxy cluster</li> <li>Redis Cluster</li> </ul>	

## 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 **4.6.6.2** Adding Instance Nodes.

Figure 4-101 Adding instance nodes

Before	a scale-out		After sca	ale-out					
	Image: Node 1         Node 2         Node 3			Node 1	Node 2	Node 3	Node 4	Node 5	
	ţ	$ \longrightarrow $				¢			
1			(						
	Distributed storage pool				Dist	ributed storage	e pool		
<u></u>	)		\						J

## Adding Instance Shards

For example, if three shards have been deployed and two more shards need to be added, there will be five shards in total. For details, see **4.6.6.3 Adding Instance Shards**.





## **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 **4.6.6.4 Deleting Instance Nodes**.

Figure 4-103 Deleting instance shards



## 4.6.6.2 Adding Instance Nodes

## Scenarios

This section describes how to add nodes to an instance to suit your service requirements. You can also delete a node as required. For details, see **4.6.6.4 Deleting Instance Nodes**.

## Usage Notes

- Adding nodes will trigger fast load balancing, which may cause a request timeout for a few seconds. Enable automatic retry for services.
- You can add nodes only when the instance status is **Available** or **Checking restoration**.
- An instance cannot be deleted when one or more nodes are being added.
- If the storage is insufficient, adding nodes is not supported. Expand the storage first. For details about the storage supported by instances of different specifications, see **1.6 Instance Specifications**.

- Nodes cannot be added if any node is stopped.
- Currently, nodes can be added only for proxy cluster and Redis Cluster instances.
- Currently, a maximum of 36 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 which you want to add nodes for, click its name, and choose **More** > **Add Node** in the **Operation** column.

#### Figure 4-104 Adding nodes

Name/ID \ominus	DB Insta	Compati	Sto	Status \ominus	Specifications	Storage Sp	oace	Load bal	Enterpris	Billing M	Operat	tion
geminidb-redis-Iss Z eff016e6a1144ca896f	Cluster	Redis 6.2	Sh	Available	2 vCPUs Sta 2 nodes	0.01%	0/16GB	-	default	Pay-per-Use Created o	Log In	Change to Yearly/Monthly More ~
												Change Specifications Create Backup Scale Storage Space Add Node Restart Reset Password Dates Create Dati-Active Relationship Create DR Relationship Rename High-risk Command
												Import Data

- **Step 4** On the **Add Node** page, specify the number of nodes to be added and view the storage of the instance.
  - If the storage capacity is sufficient, click **Next** and go to **Step 8**.
  - If the storage capacity is insufficient, click **Next** and go to **Step 5**.

Add Node ⑦	
DB Instance Name	geminidb-reds-ks
DB Instance ID	e#016e6a11144ca896f3b022ld199f6bin12
Node Specifications	2 vCPUs Standard
Current Storage	16 GB
Current Nodes	2
New Nodes	1         +         You can add 33 more nodes. The total quota is 34.           Required IP addresses: 1 Available IP addresses in the current subnet: 233
	Note Adding nodes will trigger fast load balancing, which may cause request timeouts for a few seconds. Enable automatic retry for services. If autoscaling is enabled, adding new nodes will automatically increase the storage limit of the current instance to the maximum storage supported by all instance nodes.
Total Nodes	3

New nodes are of the same specifications as existing nodes. Once a new node is added, its specifications cannot be changed.

**Step 5** On the **Scale Storage Space** page, select your target storage capacity and click **Next**.

Figure 4-105 Storage change

Scale Storage Space	0					
Current Configuration	n					
DB Instance Name					Node Specifications	geminidb.redis.large.4   2 vCPUs Standard
DB Instance ID					Current Nodes	2
Billing Mode	Pay-per-use				Current Storage	16 GB
Current Storage (GB)		17 GB				
	2	17	32	47		64
	0.100.0.1.101					

**Step 6** 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** to scale up the storage space.
- Step 7 After the storage is scaled up, go to 5 to add nodes again.
- **Step 8** On the displayed page, confirm the node configuration details.
  - Yearly/Monthly
    - If you need to modify your settings, click **Previous**.
    - If you do not need to modify your settings, click **Submit** 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**.

----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 you want to add nodes for and click its name.
- **Step 4** In the navigation pane, choose **Node Management**.

Figure 4-106	Node	management
--------------	------	------------

PURSAC INTERFERING												
Backups & Restorations	Node information											
Node Management	Stop Sturt Add Node Charge											
Accounts	Q. Select one or more filters from th	Select one or more litters from the pop-up left. If you enter a keyword without a filter applied, the system will search for all names matching this keyword.										
Slow Query Logs												
	Name(1D	Status	AZ	Private IP Address	EIP	Operation						
Parameters												
Metrics		Avašable	824		Unbound	View Metric Bind EIP More v						
Sessions												
Diagnosis Analysis		Available	az2		Unbound	View Metric Bind EIP More ~						
Rename High-risk Command		Available	823		Unbound	View Metric Bind EIP. More $\vee$						
Tags		Available	az4		Unbound	View Metric Bind EIP. More $\sim$						

- **Step 5** Click **Add Node**, on the displayed page, specify the number of nodes to be added and view the storage of the instance.
  - If the storage is sufficient, click **Next** and go to **12**.
  - If the storage is insufficient, click **Next** and go to **8**.

Ad	d Node 💿	
	DB Instance Name	geminidb-reds-lss
	DB Instance ID	eff016e6s1144ca096f3b022l6f199fbbin12
	Node Specifications	2 vCPUs Standard
	Current Storage	16 GB
	Current Nodes	2
	New Nodes	-     1     +)     You can add 33 more nodes. The total quota is 34.       Required IP addresses: 1 Available IP addresses in the current subnet: 233
		Note Adding nodes will trigger fast load balancing, which may cause request timeouts for a few seconds. Enable automatic retry for services. If autoscaling is enabled, adding new nodes will automatically increase the storage limit of the current instance to the maximum storage supported by all instance nodes.
	Total Nodes	3

New nodes are of the same specifications as existing nodes. Once a new node is added, its specifications cannot be changed.

Step 6 On the Scale Storage Space page, select your target storage capacity and click Next.

#### Figure 4-107 Storage change

Scale Storage Space	0					
Current Configuratio	n					
DB Instance Name					Node Specifications	geminidb.redis.large.4   2 vCPUs Standard
DB Instance ID					Current Nodes	2
Billing Mode	Pay-per-use				Current Storage	16 GB
Current Storage (GB)	)					
		17 GB				
						- 17 +
	2	17	32	47		64
	CominIDB Podia API sunnas	rte fact coalles, which can complete in a	seconds without offecting convices			
	Certifico Regis Ar Lauppor	no laococaling, which can complete in a	recorrus without anecting services.			

- **Step 7** 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** to scale up the storage space.
- **Step 8** After the storage capacity is expanded, go to **Step 3** to add nodes again.
- **Step 9** On the displayed page, confirm the node configuration details.
  - Yearly/Monthly
    - If you need to modify your settings, click **Previous**.
    - If you do not need to modify your settings, click **Submit** 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**.

----End

## Method 3

- 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 **Specification Information** area on the **Basic Information** page, click **Add Node**.

#### Figure 4-108 Adding nodes

Basic Information		
Backups & Restorations	Basic Information	
Node Management	DB Instance Name	DB Instance ID
Accounts	1 3	D
Slow Query Logs		
Audit Logs	Region	AZ
Parameters	CN North-Beijing4	cn-north-4a,cn-north-4b,cn-north-4c
Metrics	Maintenance Window 💿	Compatible API
Sessions	10:00 - 14:00 Change	Redis 6.2 (Fully compatible with 6.2 and earlier versions, such as $5.0, 4.0$ , and $2.8.$ )
Diagnosis Analysis		
Key Blocklist	Connection Information	
Rename High-risk Command	Load Balancer Address	Database Port
Tags	192.168.0.204:6379	6379 🖉
	Access Control	SSL
	The listener added to your load balancer does not support security group rules, so you need to configure access control for the listener. Disabiling access control adons any IP address that can access the VPC of your instance to access your instance using the load balancer IP address.	Disabled 占
	Specification Information Node Specifications geminidb.redis.medium.4   1 vCPU   4 GB Standard Change	Nodes 2 Add Node

- **Step 5** On the **Add Node** page, specify the number of nodes to be added and view the instance storage.
  - If the storage is sufficient, click **Next** and go to **Step 9**.
  - If the storage is insufficient, click **Next** and go to **Step 6**.

Add Node ③	
DB Instance Name	geminidb-redis-lss
DB Instance ID	eff016e6a1144ca896f3b022fd199f6bin12
Node Specifications	2 vCPUs Standard
Current Storage	16 GB
Current Nodes	2
New Nodes	1         +         You can add 33 more nodes. The total quota is 34.           Required IP addresses: 1 Available IP addresses in the current subnet: 233
	Note Adding nodes will trigger fast load balancing, which may cause request timeouts for a few seconds. Enable automatic retry for services. If autoscaling is enabled, adding new nodes will automatically increase the storage limit of the current instance to the maximum storage supported by all instance nodes.
Total Nodes	3

By default, specifications of the new node are the same as the instance specifications and cannot be modified.

Step 6 On the Scale Storage Space page, select your target storage capacity and click Next.

#### Figure 4-109 Storage change

Scale Storage Space	0					
Current Configuratio	on					
DB Instance Name					Node Specifications	geminidb.redis.large.4   2 vCPUs Standard
DB Instance ID					Current Nodes	2
Billing Mode	Pay-per-use				Current Storage	16 GB
Current Storage (GE	5)	17.CP				
		ii du				
	2	17	32	47		64 - 17 +
	GeminiDB Redis API si	upports fast scaling, which car	n complete in seconds without affecting se	rvices.		

**Step 7** 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 the specifications, click **Submit** to scale up the storage space.
- **Step 8** After the storage is scaled up, go to **Step 3** to add nodes again.
- **Step 9** On the displayed page, confirm the node configuration details.
  - Yearly/Monthly
    - If you need to modify your settings, click **Previous**.
    - If you do not need to modify your settings, click **Submit** 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**.

----End

## 4.6.6.3 Adding Instance Shards

#### **Scenarios**

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

#### Precautions

- Shards can be added only for standard instances with cloud native storage.
- Adding shards will trigger fast load balancing, which may cause a request timeout for a few seconds. Enable automatic retry for services.
- You can only add shards when the instance status is **Available** or **Checking restoration**.
- A DB instance cannot be deleted when one or more shards are being added.
- After shards are successfully added, the system automatically expands the storage capacity (*Number of new shards* x *Shard specification (GB)*).
- Currently, shards can be added only for proxy cluster 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, click the target instance.
- **Step 4** In the navigation pane, choose **Shard Management**.

#### Figure 4-110 Shard management

<   🔕 Available						@ Feedback Restart C
Basic Information						
Backups & Restorations	Shard information					
Shard Management	Add Shard					
Tags	NemolD	Status	AZ	Private IP Address	EIP	Operation
		<ul> <li>Available</li> </ul>	422		Unbound	
		Available	82)		Unbound	

**Step 5** Click **Add Shard**. On the displayed page, select the number of shards to be added.

Add Shard 🕑	
DB Instance Name	
DB Instance ID	
Current Shard Size	4 68
Current Shards	2
Shards To Be Added	- 1 + Maximum shards: 10; Shards you can still add: 9
	Required IP addresses: 1 Available IP addresses in the current subnet: 31983
Total Shards	3

New shards are of the same specifications as existing shards. Once a shard is added, its specification cannot be changed.

- **Step 6** On the displayed page, confirm the shard configuration.
  - Pay-per-use
    - To modify the configuration, click **Previous** to go back to the page where you specify details.
    - If you do not need to modify the configuration, click **Submit**.

----End

## 4.6.6.4 Deleting Instance Nodes

#### Scenarios

You can delete nodes of pay-per-use or yearly/monthly instances to release resources.

## **Usage Notes**

- Deleted nodes cannot be recovered. Exercise caution when performing this operation.
- 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*.

## 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 navigation pane, choose **Node Management**. On the displayed page, check the node you want to add or delete.
  - Yearly/Monthly
    - In the Node Information area, locate the target node and choose More
       > Delete in the Operation column.

Figure 4-111 Node information

Node Information						
Add Node Change						
Q. Select one or more filters from the pop-up lists. If you enter a keyword without a filter applied, the system will search for all names matching this keyword.						
Name1D	Status	AZ	Private IP Address	EP	Operation	
worker_node_1 1043beb8fdaf4a9f91ef13de9705be19no12	Available	cn north-4a	IPv4: 192.168.0.129	Unbound	View Metric Bind EIP Delete	

- Pay-per-use
  - In the Node Information area, locate the target node and click More > Delete in the Operation column.

#### Figure 4-112 Node information

Node Information						
Add Node Change						
$\ensuremath{\mathbb{Q}}$ . Select one or more filters from the pop-up lists	Q. Select one or more filters from the pop-up lists. If you enter a keyword without a filter applied, the system will search for all names matching this keyword.					
Name1D	Status	AZ	Private IP Address	EP	Operation	
_worker_node_1 1043beb8daf4a9f91ef13de9705be19no12	Available	cn-north-4a	IPv4: 192.168.0.129	Unbound	View Metric Bind EIP Delete	

- 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 Up and Down Storage Space

#### 4.6.7.1 Overview

As more data is added, you may run out of storage. This section describes how to scale up storage of your instance. As data volumes decrease, you can scale down storage to avoid low database node utilization and resource waste. **Table 4-40** lists the scaling methods supported by GeminiDB Redis instances.

Table 4-4	<b>)</b> Scaling	methods
-----------	------------------	---------

Method	Supported Instance Type	Description
4.6.7.2 Manually Scaling Up Storage Space	<ul> <li>Instance with classic storage</li> <li>Capacity- oriented instance with cloud native storage</li> <li>Proxy cluster</li> <li>Redis Cluster</li> <li>Primary/ Standby</li> </ul>	You can specify how much disk space needs to be added. The added value must be a multiple of 1 (GB). The total storage space cannot exceed the upper limit defined by your instance specifications.

Method	Supported Instance Type	Description
4.6.7.3 Automatically Scaling Up Storage Space	Standard cluster instance with classic storage	If storage usage exceeds the configured threshold, autoscaling will be triggered. The storage is scaled up by a percentage you specify. The added storage space is the current storage space multiplied by the scaling increment.
4.6.7.4 Manually Scaling Down Storage Space	Instance with classic storage • Proxy cluster • Redis Cluster • Primary/ Standby	You can specify how much disk space needs to be reduced. The storage space to be reduced must be a 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 classic storage space is 24 GB and is increased by 8 GB, the storage space will become 32 GB.

#### Figure 4-113 Manually scaling up storage space



## Automatically Scaling Up Storage Space

For example, if the classic storage space 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%. For example, if the storage usage is greater than or equal to 80%, the storage space is automatically scaled up by 2.4 GB ( $24 \times 10\%$ ), which is rounded up to 3 GB. In this case, the total storage space becomes 27 GB (24 + 3).

Figure 4-114 Automatically scaling up storage space



## Manually Scaling Down Storage Space

For example, if the classic storage space is 32 GB and is decreased by 8 GB, the storage space will become 24 GB.





## 4.6.7.2 Manually Scaling Up Storage Space

## **Scenarios**

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

## Usage Notes

- To keep services accessible, scale up storage space when the storage usage exceeds 80%.
- Storage scaling does not interrupt your services. After storage scaling is complete, you do not need to restart your instance.
- Cloud native storage of standard instances cannot be changed. You can add shards or upgrade instance specifications instead.

# Setting an Instance Status to Read-only

To ensure that the GeminiDB Redis instance can still run properly when the storage space is about to be used up, the database is set to read-only, and data cannot be modified. If this happens, you can scale up the storage to restore the database status to read/write.

Storage Space	Description
Less than 600 GB	<ul> <li>When the storage usage reaches 97%, the instance is set to read-only.</li> </ul>
	• When the storage usage decreases to 85%, the read- only status is automatically disabled for the instance.
Greater than or equal to 600 GB	• When the remaining storage space is less than 18 GB, the instance is read-only.
	• When the remaining storage space is greater than or equal to 90 GB, the read-only status is automatically disabled for the instance.

**Table 4-41** Setting an instance status to read-only

## 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 the target instance.
- **Step 4** In the **Specification Information** area on the **Basic Information** page, click **Scale** for an instance.

#### Figure 4-116 Scaling classic storage

Specification Information					
Node Specifications		Nodes	Compati		
2 vCPUs Standard	Change	2 Add Node	Redis 6.		
Total Storage Space					
Not encrypted	Scale	Autoscaling			
Used 0/16 GB		0.01%			



Specification Information		
Specifications	Nodes	Compatible API
2 vCPUs   8 GB Change	3 Add Node	Redis 6.2 (Fully compatible with 6.2 and earlier versions, such as 5.0,4.0, and 2.8 Minor Version
Total Storage Space		
	Scale	
Used Q/200 GB	0%	

**Step 5** On the displayed page, specify the new storage space and click **Next**.

#### Figure 4-118 Scaling up classic storage

Scale Storage Space	0					
Current Configuratio	n					
DB Instance Name					Node Specifications	geminidb.redis.large.4   2 vCPUs Standard
DB Instance ID					Current Nodes	2
Billing Mode	Pay-per-use				Current Storage	16 GB
Current Storage (GB	)					
		17 GB				
						- 17 +
	2	1/	32	47		64
	GeminiDB Redis API sup	ports fast scaling, which can complete	in seconds without affecting services.			

#### Figure 4-119 Scaling up cloud native storage

scale storage space	0						
Current Configurati	on						
DB Instance Name	nasql-					Node Specifications	geminidb.redis-geminifs.large.4   2 vCPUs   8 G
DB Instance ID						Current Nodes	3
Billing Mode	Pay-per-use					Storage	200 GB
Storage (GB)							
			210 GB				
					- 210	1+	
	10	120	230	340	480	+	

- 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 6** 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 Submit 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** Check the results.

- When the scaling task is ongoing, the instance status is **Scaling storage space**.
- After the scaling task is complete, the instance status becomes **Available**.
- Click the instance name. In the **Specification Information** area on the **Basic Information** page, you can view the new storage space.
- ----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 storage space you want to scale and choose **More** > **Scale Storage Space** in the **Operation** column.

□ Name/ID ⊕		DB Insta	Compati	Sto	Status \ominus	Specifications	Storage Space	Load bal	Enterpris	Billing M	Operation
	æ	Primary/S	Redis 6.2 Upgrade		<ul> <li>Available</li> </ul>	2 vCPUs Sta 2 nodes	0% 0/24GB	-	default	Pay-per-Use Created o	Log In Change to Yearly/Monthly More ~
											Change specifications Create Backup Sole Steage Space Restar Restar Delete Remand High-Hik Command Import Data
Name/ID 🕀		DB Insta	Compati	Sto	Status \ominus	Specifications	Storage Space	Load bal	Enterpris	Billing M	Operation
Name/ID 😔	C2	DB Insta Primary/S	Compati Redis 6.2 Upgrade	Sto	Status 🖗	Specifications 2 vCPUs Sta 2 nodes	Storage Space 0% 0/24GB	Load bal	Enterpris default	Billing M Pay-per-Use Created o	Operation Log In Change to Yearly/Monthly More ~ Change Specifications

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

Figure 4-121 Scaling up classic storage

Scale Storage Space	0					
Current Configuratio	on					
DB Instance Name					Node Specifications	geminidb.redis.large.4   2 vCPUs Standard
DB Instance ID					Current Nodes	2
Billing Mode	Pay-per-use				Current Storage	16 GB
Current Storage (GE	n					
Current Storage (OL	,,	17 GB				
	2	17	32	47		64
	GeminiDB Redis API supports	fast scaling, which can complete in sec	onds without affecting services.			



icale	Storage	Space	0	

Current Configurat	ion						
DB Instance Name	nosql-					Node Specifications	geminidb.redis-geminifs.large.4   2 vCPUs   8 GB
DB Instance ID						Current Nodes	3
Billing Mode	Pay-per-use					Storage	200 GB
Storage (GB)							
eterage (eas)			210 GB				
					- 2	10 +	
	10	120	230	340	480		

- 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 **Submit** 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 task is complete, the instance status becomes **Available**.
- Click the instance name. In the **Specification Information** area on the **Basic Information** page, you can view the new storage space.

----End

## 4.6.7.3 Automatically Scaling Up Storage Space

You can enable storage autoscaling for GeminiDB Redis instances. When the storage space usage reaches the upper limit, autoscaling is triggered.

You can enable storage autoscaling when or after creating an instance. For details, see **4.2 Buying a GeminiDB Redis Instance**.

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

## **Configuring Permissions**

If you are an IAM user, perform the following operations to configure GeminiDB permissions and IAM permissions before you enable storage autoscaling:

1. Configure fine-grained permissions for IAM and minimum permissions for GeminiDB.

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

{						
-	"∖	/er	sion":"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	]					
J						

2. Create a user group and assign permissions to it.

You can create a user group on the IAM console and grant it custom permissions created in **1** and the security administrator role.

#### 3. Add a user to a user group.

Log in to the IAM console using a Huawei Cloud account or an IAM account, locate the IAM user that the target instance belongs to, and add it to the user group created in **2**. The IAM user will inherit permissions of the user group.

## **Usage Notes**

- Autoscaling is available only when your account balance is sufficient.
- The instance is in the **Available** status.
- Once autoscaling is enabled, an agency will be created and fees will be automatically deducted.
- Only general-purpose GeminiDB Redis instances are supported.
- When the storage usage is greater than 98%:
  - If the total storage is less than 600 GB, the storage usage after autoscaling (used storage space/total storage space) will be less than 85%. For example, if the total storage is 500 GB and the used storage space is 495 GB, the storage usage (495/total storage space) after autoscaling will be less than 85%.
  - If the total storage is greater than or equal to 600 GB, the system automatically scales up the storage by over 90 GB. For example, if the total storage is 700 GB, the storage after autoscaling will be greater than 790 GB (700 + 90).
- Changing instance specifications or the number of nodes will affect the upper limit of the instance storage.

## Autoscaling 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 **Total Storage Space** area, click **Auto Scale**.

Figure 4-123 Auto Scale

Specification Information		
Node Specifications	Nodes	Compatible API
1 vCPU   2 GB Special offer Change	2 Add Node	Redis 6.2
Total Storage Space		
Not encrypted	e Auto Scale	
Used 0/4 GB	0.03%	



Auto Scale				×
<ol> <li>Additional stor</li> </ol>	age will be billed.			
Auto Scale				
Trigger If	10	~ %		
Available Storage				
Drops To				
Increase By	10	~ %		
Storage Limit	- 96,000	+ GB		
If available storage dri scaled up by 10%. If ti up to the nearest mult balance is insufficient,	ops to or below 10 G he increased storage ple of 10. The defau autoscaling will fail.	B or 10%, total s is not a multiple It minimum increa	torage will automat of 10 GB, the syste ment is 100 GB. If y	ically be em rounds it your account Cancel

Figure 4-124 Configuring autoscaling parameters

#### Table 4-42 Parameter 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 of the current storage to be automatically scaled up at. The value can be <b>10%</b> , <b>15%</b> , or <b>20%</b> . If the value is not a multiple of 10, it is rounded up. At least 1 GB is added each time.

#### Step 6 Click OK.

----End

## Autoscaling Storage Space of 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 DB instance list page, select the instances you want to enable autoscaling for and click **Auto Scale**.

#### Figure 4-125 Auto Scale

Auto Scale Renew	Change to Year	ty/Monthly Chang	e to Pay-per-Use	Unsubscribe	Upgrade Minor	/ersion				
All projects	Y Q	Compatible API: Redis $\times$	Instance name: leiyufe	i-redis × Add fil	ter					× 0
✓ Name/ID ⊖	DB Insta	Compati Sto	Status 😔	Specifications	Storage Space	Load bal	Enterpris	Billing M	Operation	
	Primary/S	Redis 6.2 Upgrade	Available	2 vCPUs Sta 2 nodes	0% 0/240	в	default	Pay-per-Use Created o	Log In Change to Yearly/Monthly	More ~
	Primary/S	Redis 6.2 Upgrade	Available	2 vCPUs Sta 2 nodes	0% 0/240	8	default	Pay-per-Use Created o	Log In Change to Yearly/Monthly	More ~

**Step 4** Toggle on **Autoscaling** and specify the trigger condition and increment.

Figure 4-126 Configuring autoscaling parameters

Batch Auto Scale ×					
<ol> <li>Additional sto</li> </ol>	rage will be billed.				
Auto Scale					
Trigger If	10	~ %			
Available Storage					
Drops To					
Increase By	10	~ %			
Storage Limit	Maximum storage	e supported by the current instance specifications			
The upper limit for au specification. If availat If the increased stora, minimum increment is	toscaling can only be s ble storage drops to or ge is not a multiple of 1 s 100 GB. If your accou	set to the maximum storage supported by the current instance r below 10 GB or 10%, total storage will automatically be scaled up by 10%. 10 GB, the system rounds it up to the nearest multiple of 10. The default unt balance is insufficient, autoscaling will fail.			

Table 4-43 Parameter description

Parameter	Description
Auto Scale	If you toggle on this switch, autoscaling is enabled.
Storage Usage	When the available storage usage drops to a specified threshold or the available storage drops to 10 GB, autoscaling is triggered.
Increase By	Percentage of the current storage to be automatically scaled up at. The value can be <b>10%</b> , <b>15%</b> , or <b>20%</b> . If the value is not a multiple of 10, it is rounded up. At least 1 GB is added each time.

#### Step 5 Click OK.

----End

## 4.6.7.4 Manually Scaling Down Storage Space

## Scenarios

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.
- Storage scaling does not interrupt your services. After storage space is scaled, you do not need to restart your instance.
- Cloud native storage of standard instances cannot be scaled down.

## Setting an Instance Status to Read-only

To ensure that the GeminiDB Redis instance can still run properly when the storage space is about to be used up, the database is set to read-only, and data cannot be modified. If this happens, you can scale up the storage to restore the database status to read/write.

Storage Space	Description
< 600 GB	• When the storage usage reaches 97%, the instance is read-only.
	• When the storage usage decreases to 85%, the read- only status is automatically disabled for the instance.
≥ 600 GB	• When the remaining storage space is less than 18 GB, the instance is read-only.
	• When the remaining storage space is greater than or equal to 90 GB, the read-only status is automatically disabled for the instance.

|--|

## 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 target instance and choose More > Scale Storage Space in the Operation column.

#### Figure 4-127 Scale Storage Space

□ Name/ID 🕀	DB Insta	Compati S	Sto Status 🕀	Specifications	Storage Space	Load bal	Enterpris	Billing M	Operat	tion
	& Primary/S	Redis 6.2 Upgrade	Available	2 vCPUs Sta 2 nodes	0% 0/240	GB	default	Pay-per-Use Created o	Log In	Change to Yearly/Monthly More ~
										Change Specifications
										Create Backup
										Scale Storage Space
										Restart
										Reset Password
										Delete
										Rename High-risk Command
										Import Data

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

#### Figure 4-128 Scale Storage Space

Scale Storage Space	0					
Current Configuratio	on					
DB Instance Name					Node Specifications	geminidb.redis.large.4   2 vCPUs Standard
DB Instance ID					Current Nodes	2
Billing Mode	Pay-per-use				Current Storage	16 GB
Current Storage (GB	0					
ourient otorage (ob	,	17 GB				
						- 17 +
	2	17	32	47		64
	GeminiDB Redis API supp	ports fast scaling, which can corr	plete in seconds without affecting servic	es.		

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
  - If you need to modify your settings, click **Previous**.
  - If you do not need to modify your settings, click **Submit** 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 task is complete, the instance status becomes **Available**.
- Click the instance name. In the **Specification Information** area on the **Basic Information** page, you can view the new storage space.

----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.
- Step 4 In the Specification Information area on the Basic Information page, click Scale.

#### Figure 4-129 Scaling storage

Specification Information		
Node Specifications	Nodes	Compatil
2 vCPUs Standard Change	2 Add Node	Redis 6.2
Total Storage Space		
Not encrypted Scale	Autoscaling	
Used 0/16 GB	0.01%	

**Step 5** On the displayed page, specify the new storage space and click **Next**.

#### Figure 4-130 Scaling storage

Scale Storage Space 💿							
Current Configuration	on						
DB Instance Name					Node Specifications	geminidb.redis.large.4   2 vCPUs Standard	
DB Instance ID					Current Nodes	2	
Billing Mode	Pay-per-use				Current Storage	16 GB	
0							
Current Storage (GE	,	17 GB					
	2	17	32	47		64 - 17 +	
	GeminiDB Redis API supports fast scaling, which can complete in seconds without affecting services.						

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

**Step 6** On the displayed page, confirm the storage space.

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

**Step 7** Check the results.

- When the scaling task is ongoing, the instance status is **Scaling storage space**.
- After the scaling task is complete, the instance status becomes **Available**.
- Click the instance name. In the **Specification Information** area on the **Basic Information** page, you can view the new storage space.

----End

# 4.6.8 Performing a Primary/Standby Switchover for GeminiDB Redis Instances

## Scenarios

GeminiDB Redis instances provide an automatic HA mechanism. Generally, you do not need to manually perform a primary/standby switchover. You can use this feature to perform DR drills and test client processing capabilities in HA scenarios. You can also perform a primary/standby switchover as needed to meet service requirements.

## Prerequisites

Currently, this operation can be performed only on primary/standby GeminiDB Redis instances that are in the **Available** state.

## Usage Notes

- When you perform a primary/standby switchover, the instance IP addresses remain unchanged, so there is no need to update the service connection address.
- During a primary/standby switchover, the connection is disconnected for less than 10 seconds, which can cause slow latency or command execution failures. To address this issue, you need to have a command retry or connection retry mechanism on the client. Perform primary/standby switchovers during off-peak hours.

## 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 navigation pane, choose **Node Management**. In the **Node Information** area, click **Primary/Standby Switchover**.

Figure 4-131 Performing a primary/standby switchover.

Node Information								
Stop Start Change Primary/Standby Switchover								
Q. Select one or more filters from the	pop-up lists. If you enter a keyword	without a filter applied, the system	will search for all names matching	this keyword.				
Name/ID	Role	Status	AZ	Private IP Address	EIP	Operation		
	Primary	Available	az2		O Unbound	Bind EIP Stop		
	Standby	Available	az2		Unbound	Bind EIP Stop		

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



# 4.7 Data Backup

# 4.7.1 Overview

You can create backups for GeminiDB Redis instances to ensure data reliability. After an instance is deleted, its automated backups are also deleted while manual backups are retained. The backups cannot be exported. GeminiDB Redis instances support only full backups.

## **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**

Both automatic backup and manual backup are supported.

• Automated backup

You can **modify a 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 configure in the backup policy and will store the data for a length of time you specify.

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. A 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 a 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.
Manual backup	You can manually create full backups for your instance based on service requirements.

Table 4-45 Bad	ckup methods
----------------	--------------

## How Backup Works

GeminiDB Redis API uses an architecture with decoupled storage and compute. Figure 4-133 shows how GeminiDB Redis API backs up data. GeminiDB Redis API takes snapshots of data in the storage pool in seconds and transmits them to the compute layer, which will then pass them to OBS. Snapshots are stored as compressed files in OBS buckets and do not occupy any storage space of your instance. Snapshot creation and deletion have no impacts on the computing layer. When snapshots are uploaded, some compute resources are consumed, so the instance CPU and memory usage may go high.

GeminiDB Redis API backs up data faster than Redis and does not produce jitter.





## Backup Storage

Backups are stored in OBS buckets to provide disaster recovery and save storage space.

After you purchase an instance, GeminiDB Redis will provide additional backup storage of the same size as what you purchased. For example, if you purchase an instance with 100 GB of storage, you will obtain additional 100 GB of storage free of charge. If the backup data does not exceed 100 GB, it is stored on OBS free of charge. If there is more than 100 GB of data, you will be billed at standard OBS rates.

# 4.7.2 Managing Automated Backups

GeminiDB Redis allows you to create automated backups to protect your data. If a database or table is deleted, maliciously or accidentally, backups can help recover your data.

## **Configuring an Automated Backup Policy**

Automated backups are generated based on a backup policy and saved as packages in OBS buckets to secure and protect your data. Regularly backing up your database is recommended. If your database becomes faulty or data is corrupted, you can restore it from backup. Backing up data affects the database read and write performance, so you are advised to set the automated backup time window to off-peak hours.

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

#### Figure 4-134 Modifying a backup policy

Modify Backu	ıp Policy	^
Automated Backup		
Retention Period	────────────────────────────────────	
	Enter an integer from 1 to 3660.	
Time Zone	GMT+08:00	
Time Window	03:00-04:00 V	
Backup Cycle	All	
	🗸 Monday 🗸 Tuesday 🗸 Wednesday 🗸 Thursday	
	🖌 Friday 🗸 Saturday 🗸 Sunday	
	A minimum of one day must be selected.	
	OK Cancel	)

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

If **Retention Period** is set to **2**, full and incremental backups that have been stored for more than two days will be automatically deleted. For instance, a backup generated on Monday will be deleted on Wednesday; or a 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**:

 A 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 the next Wednesday. 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 the next Monday and will expire on the next Wednesday. So the full backup generated on Tuesday will not be automatically deleted until the next Wednesday.

- Backup Cycle: All options are selected by default.
  - **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.

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 the DB instance is created, you can modify the automated backup policy as needed. You can change the time window after the DB instance is created. The system backs up data based on the automated backup policy you have set.
- 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.
- 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. Click OK.

For details about how to set a backup policy, see **Configuring an Automated Backup Policy**.

#### Figure 4-135 Modifying a backup policy

Modify Back	up Policy	×
Automated Backup		
Retention Period	-     7     +     days       Enter an integer from 1 to 3660.	
Time Zone	GMT+08:00	
Time Window	03:00-04:00 V	
Backup Cycle	III	
	🖌 Monday 🛛 🗹 Tuesday 🗹 Wednesday 🗹 Thursday	
	Friday Saturday Sunday A minimum of one day must be selected.	
	OK Cancel	$\supset$

Step 5 Check or manage the generated backups on the Backups or Backups & Restorations page.

----End

## **Disabling Automated 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 instance whose backup policy you want to modify.
- **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 to disable Automated Backup and click OK.

#### Figure 4-136 Disabling backup policies

Modify Back	up Policy							
Automated Backup	If the automated backup policy is disabled, automated backups will not be created. Existing automated backups will be retained.							
	Delete automated backups							
Retention Period	- 7 + days							
	Enter an integer from 1 to 3660.							
Time Zone	GMT+08:00							
Time Window	03:00-04:00 ~							
Backup Cycle	I Ali							
	S Monday S Tuesday Wednesday Thursday							
	Saturday							
	ОК	Cancel						

When disabling the automated backup policy, you can decide whether to delete the automated backups by selecting **Delete automated backups**.

- If you select it, all backup files within the retention period will be deleted. No automated backups are displayed in the backup list until you enable the automated backup policy again.
- If you do not select it, all 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 as they expire. You cannot delete them manually.

#### NOTICE

To delete an automated backup, disable the automated backup policy first. For details, see **Disabling Automated Backup**.

Deleted backups cannot be recovered. Exercise caution when performing this operation.

#### 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 backup you want to delete.
- d. Choose **Backups & Restorations** in the navigation pane, locate the target backup and click **Delete** in the **Operation** column.
- e. In the **Delete Backup** 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**.
  - d. In the **Delete Backup** dialog box, confirm the backup details and click **Yes**.

## Setting the Policy for Restoring Data to a Specified Time Point

- 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** Choose **Backups & Restorations** in the navigation pane one the left, and click **Point in Time Restoration**. After the setting is complete, click **OK**.

Figure 4-137 Setting the policy for restoring data to a specified time point

#### Point-in-Time Restoration

Enable				
Backup Interval	_	20	+	
Retention Period	-	1	+	
				OK Cancel

- You can toggle on or off **Enable** to configure point-in-time restoration.
- **Backup Interval** refers to the time interval, in minutes, for automated backups. The value ranges from 5 to 120. For example, if the backup interval is set to 5 minutes and the first backup is performed at 04:00, the next backup will be performed at 04:05.

х

• **Retention Period** determines how long automated backups are kept in days. The value ranges from 1 to 7. Full backups are retained till the retention period expires.

----End

# 4.7.3 Managing Manual Backups

GeminiDB Redis API allows you to manually back up instances whose status is **Available** to protect your data. If a database or table is deleted, maliciously or accidentally, backups can help recover your data. Manual backups are full backups.

## **Usage Notes**

• 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 you want to back up and choose **More** > **Create Backup** in the **Operation** column.

Figure 4-138 Creating a manual backup

## Create Backup

DB Instance Name geminidb-redis-Iss

\* Backup Name backup-19e9 ⑦
Description ⑦
O(256 \*)
O(256 \*)
Cancel

#### Method 2

1. On the **Instances** page, click the instance you want to back up.
х

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

Figure 4-139 Creating a manual backup

Create Backup	,	>
DB Instance Name	geminidb-redis-lss	
★ Backup Name	backup-19e9	0
Description		0
	0/256 //	
	ОК Сап	icel

### Method 3

In the navigation pane on the left, choose **Backups**. On the displayed page, click **Create Backup**.

#### Figure 4-140 Creating a manual backup

Create Backup			
★ Compatible API	Cassandra	MongoDB	
	InfluxDB	Redis	
★ DB Instance Type	Cluster	Performance-oriented	
	Primary/Standby		
★ DB Instance Name		~	0
★ Backup Name	backup-d4e6		0
Description			0
		0/256 //	J
		ОК	Cancel

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

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 long and start with a letter. It is case- insensitive and contains only letters, digits, hyphens (-), and underscores (_).
Description	Can include a maximum of 256 characters and cannot contain line breaks and the following special characters: >!<"&'=

Table 4-4	<b>16</b> Parameter	description
-----------	---------------------	-------------

**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 changes to **Completed**.

----End

### **Deleting a Manual Backup**

If you do not need a manual backup any longer, delete it on the **Backups** or **Backups & Restorations** page.

Deleted backups are not displayed in the backup list.

### NOTICE

Deleted backups cannot be recovered. Exercise caution when performing this operation.

### 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, locate the target backup 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 that you want to delete and click **Delete**.
- 4. In the displayed dialog box, confirm the backup details and click **Yes**.

# 4.8 Data Restoration

# 4.8.1 Restoration Methods

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

Table 4-47	Restoration	methods
------------	-------------	---------

Reference	Scenario
Rebuilding an Instance	If an instance is deleted by mistake, you can rebuild it within a retention period in the recycle bin.
4.8.2 Restoring Data to a New Instance	You can restore an existing backup file to a new instance.
4.8.3 Restoring to the Original Instance Using PITR	GeminiDB Redis API offers Point-In-Time Recovery (PITR), which allows for quick data recovery by restoring the database to its previous state before any errors occurred.

# 4.8.2 Restoring Data to a New Instance

### **Scenarios**

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

### Procedure

- Step 1 Log in to the Huawei Cloud console.
- **Step 2** In the service list, choose **Databases** > **GeminiDB**.
- **Step 3** Restore a DB instance from the 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 4-141 Restoration

Batch	Delete Create Backup Modify Backup Pol	icy Point-in-Time Res	storation				Enter a backup name.
	Backup Name/ID \ominus	Backup Type $\ \ominus$	Backup Time $\ominus$	Status 🖯	Size	Description	Operation
	redis-geminidb-redis-lss-20240625083059630 5c510ed61cc34049aff5f6f7aa93d2ebr12	Automated	Jun 26, 2024 03:23:01 - Jun 26, 2024 03:25:05 GMT	Completed	1.12 MB		Restore

×

#### Method 2

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

#### Figure 4-142 Restoration

🗌 Backup NamelID 👙	DB Instance Name/ID  🕀	Compatible API	Backup Type  🖯	Backup Time  🖯	Status 🖯	Size Description	Operation
redis_snapshot_backup-17193870857 4a31b9635ce74258990cddi9ic8d		Redis 6.2	Manual	Jun 26, 2024 15:31:27 — Ju	Completed	1.13 MB -	Restore Delete

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

#### Figure 4-143 Restoring data to a new DB instance

Restore DB Ins	stance		
DB Instance	Backup Name	DB Instance Name	
	redis_snapshot_backup-1719387085788300493		
Restoration Method	Create New Instance		
			OK Cancel

- The default API type and DB engine version are the same as those of the original instance and cannot be changed.
- The new instance must have no less than nodes than the original instance.
- 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.
- You need to set a new administrator password.
- To modify other parameters, see the description of buying instances of other DB APIs in *Getting Started*.

#### **Step 5** View the restoration 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 DB instance is independent from the original one.

----End

# 4.8.3 Restoring to the Original Instance Using PITR

In real-world service scenarios, databases may experience faults such as data damage, loss, or accidental deletion. GeminiDB Redis API offers Point-In-Time Recovery (PITR), which allows for quick data recovery by restoring the database to its previous state before any errors occurred.

# Functions

Point-in-Time Recovery (PITR) is a database feature that enables restoration to a specific time, useful for recovering data lost or damaged due to misoperations or accidental deletion.

In gaming scenarios, some players may exploit vulnerabilities to duplicate equipment and currency, leading to unfairness. Traditional databases are backed up once a day, with a long restoration time, making it difficult to restore data to a specific point in time. With PITR of GeminiDB Redis API, you can choose a specific time point for data recovery, with a minimum granularity of 5 minutes, ensuring speedy data restoration.

# Constraints

- Only GeminiDB Redis cluster instances are supported. DR instances are not supported.
- Data can only be restored to the original instance, and the database is unavailable during the restoration.
- 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.

# Setting Point-in-Time Restoration

- 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** Choose **Backups & Restorations** in the navigation pane. On the **Data Flashback** tab page, click **Point-in-Time Restoration**. Configure parameters in the displayed dialog box. After the setting is complete, click **OK**.

-		÷
Scheduled Backu	p and Restoration ⑦ Data Flashback ⑦	
Point-in-Time Re	storation	
Point-in-Tim	e Restoration ×	
Enable		
Backup Interval	-       20       +         The shorter the backup interval, the faster the data growth. You are advised to set an appropriate interval after conducting a test on a test instance.	
Retention Period	─ 1 +	
	OK Cancel	

Figure 4-144 Point-in-Time Restoration

- You can toggle on or off **Enable** to enable or disable the backup function.
- **Backup Interval** refers to the time interval, in minutes, for automated backups. The range of values is from 5 to 120 minutes. For example, if the backup interval is set to 5 minutes and the first backup is performed at 04:00, the next backup will be performed at 04:05.
- **Retention Period** determines how long automated backups are kept in days. The range of values is from 1 to 7 days. Backups are retained till the retention period expires.

### 

To avoid rapid data bloat, it is important to set an appropriate backup interval in the test environment before your actual use.

----End

### **Restoring Data to the Original 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.

# **Step 4** Choose **Backups & Restorations** in the navigation pane. On the **Data Flashback** tab page, click **Restore to Point in Time**.



Basic Information	
Backups & Restorations	Scheduled Backup and Restoration (?) Data Flashback (?)
Node Management	
Accounts	Point-in-Time Restoration

**Step 5** Select the date and time point to which the data is restored.

#### Figure 4-146 Restoring data to a point in time

Restore to Poi	nt in Time	X
Date	2024/06/25	×
Time Point		
Restoration Method	DB Instance	
		OK Cancel

Step 6 Click OK.

----End

# 4.9 Diagnosis Analysis

# 4.9.1 Big Key Diagnosis

### Functions

A key that contains a large volume of data is considered a big key. GeminiDB Redis API can diagnose big keys, allowing you to collect statistics on big keys in the current instance.

GeminiDB Redis API uses shared storage, so big keys do not cause data skew or out of memory (OOM) issues on shards. Access to big keys is common when you are using Redis databases. Big key diagnosis obtains analysis results from the background, minimizing the impact on services.

# Viewing the Diagnostic Information of Big Keys

- Step 1 Log in to the Huawei Cloud console.
- Step 2 In the service list, choose Databases > GeminiDB.
- **Step 3** On the **Instances** page, click the name of the target instance to go to the **Basic Information** page.
- Step 4 In the navigation pane on the left, choose Diagnosis Analysis.
- **Step 5** Choose **Big Key Diagnosis** and select the target types.

### Figure 4-147 Big key diagnosis

Basic Information				
Backups & Restorations	Big Key Diagnosis Hot Key Diag	Inosis		
Node Management	Type 💙 All 💙 string 🗹 hash 💙 z	tset 💟 set 💟 exhash 💟 list 💟 stream		
Slow Query Logs	Туре	Big Key Name	Length	DB
Audit Logs				
Parameters			6 D	
Metrics				
Sessions			No data susilabla	
Diagnosis Analysis			no usia avaliante.	

**Step 6** View big key parameters by referring to **Table 4-48**.

Table 4-48 Big	g key	parameters
----------------	-------	------------

Parameter	Description
Туре	Type of a big key.
	• string
	• hash
	• zset
	• set
	• exhash
	• list
	• stream
Big Key Name	Name of a big key.
Length	Length of the value.
DB	Database where a big key is located.

----End

# Setting Parameters for Big Key Diagnosis

The value size of a key determines whether the key is a string key. The number of members in the key determines whether the key is a hash, list, zset, set, or stream key.

There are two parameters involved:

- **bigkeys-string-threshold**: If the value size of a string key is greater than the value of this parameter, the key is determined as a big key. The unit is byte. The default value is **102400** (1 MB).
- **bigkeys-composite-threshold**: If the number of elements in a hash, list, zset, set, or stream key is greater than the value of this parameter, the key is identified as a big key. The default value is **10240**.
- **Figure 4-148** Parameters for big key diagnosis

Parameters Change History				
Save Cancel Preview	Export Compare			Enter a parameter name. Q
Parameter Name	Effective upon Restart 🖯	Value	Allowed Values	Description
AuthFailLockTime	No	6	0-10,000	The length of time, in second, that a suspicious IP ad
BigkeysQuantityLimitation	No	100	1-10,000	string/hash/list/zset/set/exhash/stream type of large k
CompatibleMode	No	3 v	0, 1, 2, 3	Whether StackExchange Redis is available. Set this p
EnableAdDbDirect	No	no v	yes, no	is the DB direct function enabled. The default is false.
MaxAuthFallTimes	No	6	0-10,000	Maximum failed access attempts. When this limit is re
ProxyTimeout	No	0	0-100,000	The length of time, in seconds, that a proxy-client con
bigkeys-composite-threshold	No	10240	1-2,147,483,647	A key of the hash/list/zset/set/stream type whose num
bigkeys-string-threshold	No	102400	1-2,147,483,647	If the value is greater than the value of a string key, th
databases	No	1000	1-1,000	Allow a limit on the number of supported DBs.
maxmemory-policy	Yes	noeviction	noeviction	Whether new keys can be saved when the storage sp
notify-keyspace-events	No		-	The type of event that needs to be monitored. The def
slawlog-threshold	No	300000	80,000-100,000,000	Maximum time in microseconds for executing a query

Do not set these two parameters to small values. Otherwise, too many invalid results are generated and occupy the network bandwidth, slowing down data access.

For details about how to modify parameters for big key diagnosis, see **Modifying Parameters of an Instance**.

# 4.9.2 Hot Key Diagnosis

A key that is frequently accessed is considered a hot key. This section describes how to use hot key diagnosis for GeminiDB Redis instances.

### Constraints

- For GeminiDB Redis instances in a proxy cluster or Redis Cluster, the top 30 hot keys that are accessed most frequently can be diagnosed. For primary/ standby instances, the top 20 hot keys that are accessed most frequently can be diagnosed.
- After audit log is enabled, hot key diagnosis history will be audited.
- A key that receives 1,000 or more queries per second (QPS) is considered a hot key. When the QPS value is greater than 6,000, the accurate value is not collected.

# 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 its name.
- **Step 4** In the navigation pane on the left, choose **Diagnosis Analysis**.

### Step 5 Choose Hot Key Diagnosis.

#### Figure 4-149 Hot key diagnosis

Big Key Diagnosis	Hot Key Diagnosis			
Name	Туре	Latest Command	QPS ③	DB
		the second se		
		No data available		

Table 4-49 Parameters for hot key diagnosis

Parameter	Description
Name	Name of a hot key.
Туре	Type of a hot key, which can be <b>string</b> , <b>hash</b> , <b>list, set,</b> or <b>sorted set</b> .
Latest Command	Latest command executed for the hot key.
QPS	Number of accesses to a hot key per second. <b>NOTE</b> The maximum QPS that can be displayed is 6,000. If a QPS exceeds 6,000, the accurate value will not be collected.
DB	Database where a hot key is located.

----End

# 4.10 Account and security

# 4.10.1 Enabling Password-Free Access

### Constraints

- Password-free access can be enabled for a maximum of 30 CIDR blocks.
- Redis Cluster GeminiDB Redis instances do not support this function.

### Procedure

Step 1	Log in to the Huawei Cloud console.
Step 2	In the service list, choose <b>Databases</b> > <b>GeminiDB</b> .
Step 3	Click the instance name to go to the <b>Basic Information</b> page.
Step 4	In the <b>Connection Information</b> area, click <b>Enable</b> .

#### Figure 4-150 Connection information

Connection Information			
Load Balancer Address	Database Port 6379 🖉	Maximum Connections 20000 Sessions	Password-Free Access Disabled Enable
Access Control Configure(NoviNetia to backtist configured) Configure(NoviNetia to backtist configured) Tables to provi need to configure access control for the latence. VPC of typer Instance to access your instance using the load balancer iP address.	SSL لے Enabled		

**Step 5** In the displayed dialog box, enter the CIDR block that you want to enable password-free access for.

Figure 4-151 Configuring password-fre	e access
Password-Free Access	×
Client CIDR Block	/⊙ ⊙
	OK Cancel
<ul> <li>To add a CIDR block, click ⊕.</li> <li>To delete a CIDR block, click ⊖.</li> </ul>	

#### Step 6 Click OK.

----End

### FAQs

Can I access an instance using a password if the instance supports password-free IP address?

Yes. A GeminiDB Redis instance can be accessed no matter you use a correct password or not.

# 4.10.2 ACL Account Management

### **Scenarios**

GeminiDB Redis API provides the enterprise-grade multi-tenant capability. You can add read-only or read/write accounts in your instance to control access to each database to avoid misoperations on data of other tenants. This section describes how to manage accounts.

### Precautions

• A maximum of 200 ACL accounts can be created for each GeminiDB Redis instance.

- Account change takes effect 10 seconds later after it is performed.
- If you use a backup to restore data to a new instance, the account information of the original instance will not be inherited.
- The account to be created must meet the requirements in Table 4-50.

Table 4-50 Parameter	requirements
----------------------	--------------

Parameter	Requirement	Example Value
Account	Cannot be left blank.	Organization
Name	<ul> <li>Can contain a maximum of 36 characters.</li> </ul>	
	<ul> <li>Must start with a letter and can contain only digits, letters, underscores (_), and hyphens (-).</li> </ul>	
Permission	• Read/Write	Read/Write
	Read-only	
Database	Authorize all databases	Authorize all
	Unauthorized	databases
	Authorized	
	NOTE	
	<ul> <li>You can add a database on the right of <b>Database</b> as required.</li> </ul>	
	<ul> <li>You can select the databases to be authorized as required.</li> </ul>	
	• <b>Database</b> refers to the DB of the open-source Redis.	
Password	Cannot be left blank.	test123456
	Can include 8 to 32 characters.	
	<ul> <li>Must contain at least two of the following types: uppercase letters, lowercase letters, digits, and special characters. The following special characters are allowed: ~!@#%^*= +?\$()&amp;</li> </ul>	

### Creating an ACL Account

- Step 1 Log in to the Huawei Cloud console.
- Step 2 In the service list, choose Databases > GeminiDB.
- **Step 3** In the instance list, select an instance and click its name to go to the **Basic Information** page.
- **Step 4** In the navigation pane on the left, choose **Accounts**.
- **Step 5** On the displayed page, click **Create Account**.

#### Figure 4-152 Creating an account



**Step 6** In the displayed dialog box, enter a username, select permissions for databases, authorize required databases (DBs), enter a password, confirm the password, and click **OK**.



Create Accou	int				
Account Name					
Permission	Read/Write     Read-only				
Database	Authorize all databases			Enter an unauthorized	Add
	Unauthorized	0/0		Authorized	0 / 1
	No data available		>	Authorize all databases	
Password			Ø		
Confirm Password			Ö		
					OK Cancel

**Step 7** View and manage the created account in the account list.

----End

### **Resetting a Password**

- Step 1 Log in to the Huawei Cloud console.
- **Step 2** In the service list, choose **Databases** > **GeminiDB**.
- **Step 3** In the instance list, locate the target instance and click its name to go to the **Basic Information** page.
- **Step 4** In the navigation pane on the left, choose **Accounts**.
- **Step 5** On the displayed page, select the account whose password needs to be reset and click **Reset Password**.

#### Figure 4-154 Resetting a password



**Step 6** In the displayed dialog box, enter a new password, confirm the password, and click **OK**.

Reset Passwo	ord	2	>
DB Instance Name			
Account Name			
Password		2	
Confirm Password		- De la construcción de la const	
		OK Cancel	

Figure 4-155 Resetting a password

**Step 7** Run **auth** *USER PWD* or **auth** *USER:PWD* to log in to the instance again.

----End

### **Changing Permissions of an Account**

- Step 1 Log in to the Huawei Cloud console.
- **Step 2** In the service list, choose **Databases** > **GeminiDB**.
- **Step 3** In the instance list, locate the target instance and click its name to go to the **Basic Information** page.
- **Step 4** In the navigation pane on the left, choose **Accounts**.
- **Step 5** On the displayed page, select the account whose permissions need to be modified and click **Change Permission**.

Figure 4-156 Changing permissions

Create Account				
Account Name	Account Type	Permission	Authorized	Operation
Administrator (only password required f	Administrator	Read/Write	all .	-
	Regular	Read/Write	art .	Reset Password Change Permission Delete

**Step 6** In the displayed dialog box, select the required permissions and database, and click **OK**.

Figure 4-157 Changing permissions

Change Pe	rmission				,
Account Name					
Permission	Read/Write   Read-only				
Database	Authorize all databases			Enter an unauthorized	Add
	Unauthorized	0 / 0		Authorized	0 / 1
	No data available		>	Authorize all databases	
				ОК	Cancel

**Step 7** Run **auth** *USER PWD* or **auth** *USER:PWD* to log in to the instance again.

----End

### **Deleting an ACL Account**

- Step 1 Log in to the Huawei Cloud console.
- **Step 2** In the service list, choose **Databases** > **GeminiDB**.
- **Step 3** In the instance list, locate the target instance and click its name to go to the **Basic Information** page.
- **Step 4** In the navigation pane on the left, choose **Accounts**.
- **Step 5** On the displayed page, locate the account that you want to delete and click **Delete**. In the displayed dialog box, click **OK**.

#### Figure 4-158 Deleting an account

**Step 6** Verify that the deleted account is no longer displayed.

----End

### How to Use a New Account to Access a Database

- 1. Access the database by running **auth** USER PWD or **auth** USER:PWD.
- 2. When you access the database using an SDK on your application, use the value of *USER* and *PWD* as the username and password, or use *USER:PWD* as the password parameter.

When you access the database by running **auth** *argc*, ensure that *argc* does not contain colons. If an incorrect password contains colons, the returned value is the same as that of **auth** *argc1 argc2*.

# 4.10.3 Enabling Automated Database Redirection for ACL Accounts

# Scenarios

For example:

- 1. An account (username **user1** and password **p1**) has been set to access only DB 10 in service A.
- 2. An account (username **user2** and password **p2**) has been set to access only DB 18 in service B.

Applications usually want fewer program changes. For example, to retain the default value of the database parameter of a client instead of setting it to **18** for application B, you can enable automated database redirection so that the account can be automatically redirected to database 18 using only a password. Even if SELECT is mistakenly executed on DB 10 for application B, application A is not affected.

Second Seco

# Usage Notes

- Only one database can be specified for each ACL account for automated redirection. Otherwise, the authentication fails.
- The password of a new account cannot be the same as an existing password. Otherwise, the authentication fails.
- Redis Cluster GeminiDB Redis instances do not support this function.

### **Enabling Automated Database Redirection for ACL Accounts**

Set **EnableAclDbDirect** to **yes** to enable this feature. For details, see **Modifying Parameters of an Instance**.

#### Figure 4-159 Parameters

Parameters Change History				
Save Cancel Preview	r Export Compare			Enter a parameter name. Q
Parameter Name	Effective upon Restart 🖯	Value	Allowed Values	Description
AuthFailLockTime	No	8	0-10,000	The length of time, in second, that a suspicious IP ad
BigkeysQuantityLimitation	No	100	1-10,000	string hash list/zset/set/exhash/stream type of large k
CompatibleMode	No	3 ~	0, 1, 2, 3	Whether StackExchange Redis is available. Set this p
EnableAclDbDirect	No	yes ^	yes, no	is the DB direct function enabled. The default is false.
MaxAuthFailTimes	No	Q, Search yes	0-10,000	Maximum failed access attempts. When this limit is re
ProxyTimeout	No	no	0-100,000	The length of time, in seconds, that a proxy-client con

### How to Use a New Account to Access a Database

- 1. Run **auth** *PWD*.
- 2. When you access a database using an SDK, use *PWD* as the password parameter.

When you access a database by running **auth** *argc*, ensure that *argc* does not contain colons. If an incorrect password contains colons, the returned value is the same as that of **auth** *argc1 argc2*.

# 4.10.4 Brute Force Attack Defense

- Brute-force attack defense mechanism
   GeminiDB Redis enables brute force attack defense by default, to automatically lock out an IP address after 5 failed authentication attempts.
- Automatic unlocking

After an IP address is locked for 5s, the IP address is automatically unlocked and can be authenticated again.

• Manual unlocking

To manually unlock the IP address or disable anti-brute-force attack, choose **Service Tickets > Create Service Ticket** in the upper right corner of the console and contact the customer service.

### 

To improve security, you can submit a service ticket and ask technical engineers to help adjust authentication times and locking duration.

Make sure to fully evaluate risks and exercise caution when you disable or modify the security policy. After you adjust the security policy, risks and accidents incurred will not be accounted in the SLA and shall be borne by yourself.

# 4.11 Parameter Management

# 4.11.1 Modifying Parameters of GeminiDB Redis Instances

To ensure optimal GeminiDB Redis performance, you can modify instance parameters based on service requirements.

### **Usage Notes**

- You can directly modify parameters on the parameter modification page of an instance.
- 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.

### 

Exercise caution when modifying parameter values to prevent exceptions.

### **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, choose **Parameters**. On the displayed page, modify parameters listed in **Table 4-51**.

### Figure 4-160 Parameters

Parameters Change History				
Save Cancel Preview	Export Compare			Enter a parameter name. Q
Parameter Name	Effective upon Restart 🕀	Value	Allowed Values	Description
AuthFailLockTime	No	8	0-10,000	The length of time, in second, that a suspicious IP ad
BigkeysQuantityLimitation	No	100	1-10,000	string/hash/list/zset/set/exhash/stream type of large $k\ldots$
CompatibleMode	No	3 ~	0, 1, 2, 3	Whether StackExchange.Redis is available. Set this p
EnableAcIDbDirect	No	no v	yes, no	is the DB direct function enabled. The default is false.
MaxAuthFallTimes	No	8	0-10,000	Maximum failed access attempts. When this limit is re
ProxyTimeout	No	0	0-100,000	The length of time, in seconds, that a proxy-client con
bigkeys-composite-threshold	No	10240	1-2,147,483,647	A key of the hash-listizset/set/stream type whose num
bigkeys-string-threshold	No	102400	1-2,147,483,647	If the value is greater than the value of a string key, th
databases	No	1000	1-1,000	Allow a limit on the number of supported DBs.
maxmemory-policy	Yes	noeviction	noeviction	Whether new keys can be saved when the storage sp
notify-keyspace-events	No		-	The type of event that needs to be monitored. The def
slowlog-threshold	No	300000	80,000-100,000,000	Maximum time in microseconds for executing a query

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

#### Table 4-51 GeminiDB Redis instance parameters

Parameter	Effe ctiv e upo n Rest art	Defa ult Valu e	Value Range	Description	Exc ept ion
AuthFailLockT ime	No	5	0- 10,000	Authentication failure lock time, in seconds. This parameter specifies the duration during which a suspicious IP address is locked. After the duration expires, the IP address is automatically unlocked.	Re dis Clu ste r Ge mi niD B Re dis inst anc es do not sup por t this par am ete r.

Parameter	Effe ctiv e upo n Rest art	Defa ult Valu e	Value Range	Description	Exc ept ion
BigkeysQuanti tyLimitation	No	100	1– 10,000	Maximum number of big keys of the STRING, HASH, LIST, ZSET, SET, EXHASH, and STREAM data types that can be queried.	Re dis Clu ste r Ge mi niD B Re dis inst anc es do not sup por t this par am ete r.

Parameter	Effe ctiv e upo n Rest art	Defa ult Valu e	Value Range	Description	Exc ept ion
CompatibleM ode	No	3	0, 1, 2, 3	Adaptation switch for the StackExchange.Redis client. If StackExchange.Redis reports error "Multiple databases are not supported on this server", change the parameter value to <b>0</b> .	Re dis Clu ste r Ge mi niD B Re dis inst anc es do not sup por t this par am ete r.

Parameter	Effe ctiv e upo n Rest art	Defa ult Valu e	Value Range	Description	Exc ept ion
EnableAclDbD irect	No	no	yes, no	Whether direct database access is enabled. The default value is <b>no</b> .	Re dis Clu ste r Ge mi niD B Re dis inst anc es do not sup por t this par am ete r.

Parameter	Effe ctiv e upo n Rest art	Defa ult Valu e	Value Range	Description	Exc ept ion
MaxAuthFailTi mes	No	5	0- 10,000	Maximum number of authentication attempts permitted per connection. When the number of incorrect password attempts reaches the threshold, the instance will forbid access from a suspicious IP address for a short period of time. <b>0</b> indicates this function is disabled.	Re dis Clu ste r Ge mi niD B Re dis inst anc es do not sup por t this par am ete r.

Parameter	Effe ctiv e upo n Rest art	Defa ult Valu e	Value Range	Description	Exc ept ion
ProxyTimeout	No	0	0- 100,00 0	Timeout (in seconds) when a proxy receives no response from a client. When the timeout reaches the threshold, the proxy proactively closes the connection. If the value is <b>0</b> , the proxy will not proactively disconnect the connection.	Re dis Clu ste r Ge mi niD B Re dis inst anc es do not sup por t this par am ete r.

Parameter	Effe ctiv e upo n Rest art	Defa ult Valu e	Value Range	Description	Exc ept ion
enable-acl- direct	No	no	yes, no	Whether to enable direct database access The default value is <b>no</b> .	Pri ma ry/ Sta nd by an d pro xy clu ste r Ge mi niD B Re dis inst anc es do not sup por t this par am ete r.
bigkeys- composite- threshold	No	1024	1– 2,147, 483,64 7	If the number of elements in a HASH, LIST, ZSET, SET, or STREAM key is greater than the value of this parameter, the key is identified as a big key. The default value is <b>1024</b> .	-

Parameter	Effe ctiv e upo n Rest art	Defa ult Valu e	Value Range	Description	Exc ept ion
bigkeys- string- threshold	No	1024 00	1– 2,147, 483,64 7	If the size of a STRING key is greater than the value of this parameter, the key is determined as a big key. The unit is byte. The default value is <b>102400</b> .	-
databases	No	1000	1– 1,000	Maximum number of supported databases	-
key-scan- batch	No	5000	1– 2,147, 483,64 7	Number of keys scanned each time	-
maxmemory- policy	Yes	noev ictio n	noevic tion	Key discarding policy after storage space is used up. GeminiDB Redis instance storage can be scaled up in seconds. After storage space is used up, the instance becomes read-only and its service data is retained. Autoscaling will be available later.	-
notify- keyspace- events	No	_	Combi nation of A, K, E, g, \$, l, s, h, z, x, e, and t	Type of an event to be listened on. The default value is empty, indicating that the parameter does not take effect. Combination of A, K, E, g, \$, l, s, h, z, x, e, and t	-
slowlog- threshold	No	3000 00	80,000 - 100,00 0,000	Time threshold (in us) used to define when slow queries are logged on the console. A small value may affect instance performance. You are advised to retain the default value.	-

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

For details about how to view parameter modification details, see **4.11.3 Viewing Parameter Change History**.

### NOTICE

After you modify instance parameters, the modifications immediately take effect for the instance.

Check the value in the Effective upon Restart column.

- If the value is Yes and the instance status on the Instances page is
   Parameter change. Pending reboot, you must reboot the instance for the modifications to take effect.
- If the value is No, the modifications take effect immediately.

----End

### Modifying a Custom Parameter Template and Applying It 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 4-161 Editing a parameter template

Patameters					
Change History	Save Cancel Preview				Enter a parameter name. Q.
	Parameter Name	Effective upon Restart (8)	Value	Allowed Values	Description
	max-concurrent-query-limit	Yes	4	4-32	Concurrent queries. The default value depends on th
	max-concurrent-write-limit	Yes	16	16-128	Concurrent write requests. The default value depend
	max-connection-limit	Yes	500	500-4,000	Maximum connections. The default value depends on
	query-timeout	Yes	0	0-60	Query command timeout (minutes).

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

#### Figure 4-162 Preview Change

#### **Preview Change**

Parameter NameCurrentNewmax-concurrent-query-li...45

Х

Close

**Step 6** After parameters are modified, click **Change History** to view parameter modification details.

For details about how to view parameter modification details, see **4.11.3 Viewing Parameter Change History**.

### NOTICE

- The modifications take effect only after you apply the parameter template to your instance. For details, see **4.11.8 Applying a Parameter Template**.
- The change history page displays only the modifications of the last seven days.

#### ----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 4-163 Creating a parameter template

Create Parameter	Template
------------------	----------

+ Compatible API	Cassandra	MangaDP	
× Compatible AFT	Cassanura	INIONGOLDE	
	InfluxDB	Redis	
* DB Engine Version	5.0	~	
. Deservation Translate Name	0		
* Parameter Template Name	paramsGroup-10f4		0
			_
Description	Enter a parameter terr	plate description.	?
		0/256	4
You can create 96 more paramete instances in a project	er templates. The parame	eter template quota is shared	by all DB
instances in a project.			
		ОК	Cancel

- **Compatible API**: Select the API type that is compatible with your DB engine parameter template.
- **DB Engine Version**: Select a DB engine version, for example, 5.0.
- **Parameter Template Name**: The template name is 1 to 64 characters long. It contains 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 4-164 Viewing change history of a customer parameter template

Change History	The parameter change history of the last se	ven days is displayed.			Enter a parameter na
	Parameter Name 🔶	Original Value	New Value	Modification Status	Modification Time \ominus
	AuthFallLockTime	5	6	🥑 Successful	Jun 25, 2024 20:08:07 GMT+08:00

After you change a parameter template, you can apply it to an instance based on service requirements by referring to **4.11.8** 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 4-165 Viewing parameter change history of an instance

Parameters	Change History						
The parameter cha	nge history of the last seven days is disp	layed.					Enter a parameter name. Q
Parameter Name		Original Value	New Value	Modification Status	Modification Time 🖯	Application Status	Application Time
AuthFallLockTime		5	6	Successful	Jun 25, 2024 20:10:27 GMT+08:00	Applied	Jun 25, 2024 20:10:27 GMT+08:00
MaxAuthFailTime	5	5	6	Successful	Jun 25, 2024 20:10:51 GMT+08:00	<ul> <li>Applied</li> </ul>	Jun 25, 2024 20:10:51 GMT+08:00

----End

# 4.11.4 Exporting a Parameter Template

- You can export parameters of your instance to a new parameter template for future use. To learn how to apply the parameter template to another instance, refer to 4.11.8 Applying a Parameter Template.
- You can also export parameter template information (including parameter names, values, and descriptions) of your instance to a CSV file for review and analysis.

### Procedure

Step 1 Log in to the Huawei Cloud console.

Export Parameters

- **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** page, click **Export**.

#### Figure 4-166 Exporting a parameter template

Export To	Parameter Template	File	
* New Parameter Template	paramsGroup-ec39		0
Description	Enter a parameter template de	escription.	?
		0/256 //	
		0/256 //	Cancel

• **Parameter Template**: You can export parameters of your instance to a template for future use.

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

### D NOTE

- 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 can include a maximum of 256 characters and cannot include line breaks or the following special characters: >!<"&'=</li>

After the export is complete, a new parameter template is generated and displayed on the **Parameter Templates** page.

• File: You can export the parameter template information (including parameter names, values, and descriptions) of your instance to a CSV file for review and analysis.

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

### D NOTE

The file name must start with a letter and can include 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 4-167 Comparing two parameter templates

Compare Parameter Templates		
Parameter Template	Default-Redis-5.0	~
		OK Cancel

- 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 Specific Instance**

- Step 1 Log in to the Huawei Cloud console.
- Step 2 In the service list, choose Databases > GeminiDB Redis API.
- **Step 3** In the navigation pane on the left, choose **Instances**.
- **Step 4** On the **Instances** page, 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 4-168** Comparing the parameter template of the current instance with another parameter template

Compare Parameter Templates		
Parameter Template	Default-Redis-5.0	~
		OK Cancel

- 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 that you have created. When you have already created a parameter template and want to use most of the custom parameters and values from that template to 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 custom 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 parameter template that you want to replicate and click **Replicate** in the **Operation** column.

Alternatively, click the instance name 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 a parameter template name and description and click **OK**.

#### Figure 4-169 Replicating a parameter template

### Replicate Parameter Template

* Source Parameter Template	paramsGroup-61c9
* New Parameter Template	paramsGroup-ccd5
Description	Enter a parameter template description.
You can replicate 96 more parame instances in a project.	0/256 2/256 0/256 2/26 2/2
	OK Cancel

- New Parameter Template: 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 can contain a maximum of 256 characters and cannot include line breaks or the following special characters: >!<"&'=

After replication is complete, a new template is generated and displayed 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.

----End

# 4.11.8 Applying a Parameter Template

GeminiDB Redis allows you to apply a parameter template. After you modify a parameter template, modifications you make do not take effect until you apply the template to an 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 as follows:
  - To apply a default template, click **Default Templates**, locate the target parameter template, and click **Apply** in the **Operation** column.
  - 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 to which the parameter template will be applied 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 Redis 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 displayed page, locate the parameter template whose application records you want to view and choose More > View Application Records in the Operation column.

You can view the name or ID of the instance that the parameter template is applied to, as well as the application status, application time, and failure cause.

----End

# 4.11.10 Modifying the Description of a Parameter Template

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 Click the Custom Templates tab. Locate the parameter template whose

description you want to modify and click  $\checkmark$  in the **Description** column.

- **Step 5** Enter a new description and click  $\checkmark$  to submit or  $\times$  to cancel the modification.
  - After you submit the modification, you can view the new description in the **Description** column in the parameter template list.
  - The description can include a maximum of 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 wish 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 Enabling or Disabling Log Reporting

### Scenarios

If you enable log reporting for your GeminiDB Redis instance, new logs generated for the instance will be uploaded to Log Tank Service (LTS).
### Usage Notes

- To use this function for a GeminiDB Redis instance with classic storage, choose **Service Tickets > Create Service Ticket** in the upper right corner of the console and contact the customer service.
- You will be billed for enabling this function.
- Ensure that there are available LTS log groups and log streams in the same region as your instance.

For more information about log groups and log streams, see **Log Management**.

### **Enabling Log Reporting to LTS in Batches**

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 **Log Reporting**.
- **Step 4** Select one or more instances and click **Enable Log Reporting**.
- Step 5 Select an LTS log group and log stream and click OK.

**NOTE** 

- This function does not take effect immediately. There is a delay of about 10 minutes.
- You will be billed for enabling this function. For details, see LTS pricing details.

х

#### Figure 4-170 Enabling log reporting

Enal	ble Log Reporting
0	Logs record all requests sent to your DB instance and are stored in Log Tank Service (LTS). This request does not take effect immediately. There is a delay of about 10 minutes. You will be billed for log reporting. After this function is enabled, all audit policies are reported by default. If Audit Policy is enabled, LTS reuses the audit policy set for your DB instance and you will also be billed for reporting audit logs to LTS. (Only after you disable Audit Policy, the fee will be terminated.) If you enable audit log reporting to LTS for an instance with the Audit Policy toggle switch turned on, you can turn off this switch only when the instance status becomes available.
Log Ty	pe 💿 Slow logs 🗌 Audit log
Report	t Slow Logs to LTS
★ Log	Group View Log Groups
★ Log	Stream V Q
	OK Cancel

- **Step 6** To disable log reporting, select one or more instances and click **Disable Log Reporting**.
- **Step 7** In the displayed dialog box, click **OK**.

Figure 4-171 Disabling log reporting

Disable Log Reporting							
If log reporting is disabled, logs generated for the DB instance will not be reported to Log Tank Service (LTS). This request does not take effect immediately. There is a delay of about 10 minutes.							
Log Type   Slow logs  Audit log  ( Cancel							
End							

# 4.12.2 Viewing and Exporting Slow Query Logs

GeminiDB Redis 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.

### **Reporting Slow Query Logs to LTS**

To use this function, choose **Service Tickets** > **Create Service Ticket** in the upper right corner of the console and contact the customer service.

- 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 **Log Reporting**.
- **Step 4** Select an instance and click **I** next to **Report Slow Log to LTS**.
- Step 5 Select an LTS log group and log stream and click OK.

#### **NOTE**

You will be billed for enabling this function. For details, see LTS pricing details.

#### Figure 4-172 Reporting slow query logs to LTS

#### Report Slow Logs to LTS

<ul> <li>Logs record all requests sent to your DB instance and are stored in Log Tank Service (LTS).</li> <li>This request does not take effect immediately. There is a delay of about 10 minutes.</li> <li>You will be billed for log reporting.</li> </ul>							
* Log Group	Its-group-aa_1714420040 View Log Groups						
★ Log Stream	Its-topic-aa_17144200412 V Q						
	OK Cancel						

----End

#### 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, click the target instance.
- 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.



Node Name	Statement Type	Execute Statement	Execution Time (ms)	Database Where a B	Request Size	Number of Request	Returned Packet Size	Number of Returned	Execution End Time
	shardsvr	"open" "default#830e71	907.73	0	0.23 KB	3	0.24 KB	1, resp: "\$238/rindefa	2024/06/25 16:29:34 GMT+08:00
	shardsvr	"open" "default#e3a9cb	871.78	0	0.24 KB	3	0.25 KB	1, resp: "\$246/rindefa	2024/06/25 16:29:34 GMT+08:00

#### **NOTE**

Some instances of earlier kernel minor version need to be upgraded by following **4.6.1 Upgrading a Minor Version** to enable metrics Database Where a Big Key Is Located, Request Size, Number of Request Parameters, Returned PACKET Size, and Number of Returned Values.

 Select All nodes and view slow query logs of all nodes. Alternatively, select a specific node to view its slow query logs.

Figure	4-174	Querying	nodes
--------	-------	----------	-------

All nodes	^
Q Search	
All nodes	

- Choose to view slow query logs of all types of SQL statements or a specific SQL statement.
  - SET
  - GET
  - DEL
  - INCR
  - INCRBY
  - INCRBYFLOAT
  - DECR
  - DECRBY
  - GETSET
  - APPEND
  - MGET

•••••

You can view slow query logs of all types of Redis SQL statements.

- View slow query logs of a specific node in different time ranges.
- **Step 6** On the **Log Details** page, click  $\square$  in the upper right corner of the log list to export log details.
  - You can view the CSV file exported to your local PC.

• Up to 2,000 logs can be exported at a time.

Figure 4-175 Exporting slow query logs

 grul\_uoter\_nod\_1
 // # different
 // # different
 // # different

 (Association State)
 C
 0

 Node Name
 Statement Type
 Execute Statement
 Execute Statement Type
 Execute Statement
 Number of Request Face.
 Number of Request Face.
 Number of Request Face.
 Executes Statement Type
 Execute Statement Type
 Execute Statement
 Number of Request Face.
 Executes Statement Type
 Executes Statement
 Executes Statement Type
 Executes Type
 Executes Type
 Exec

----End

# 4.12.3 Viewing Audit Logs

You can view audit logs of databases on GeminiDB Redis instances. You can analyze, search for, monitor, download, and view real-time logs on the LTS console.

#### **Usage Notes**

- No audit logs are generated for operations on internal connections.
- Commands that are always audited include BIGKEYS, KEYS, FLUSHALL, FLUSHDB, SCRIPT, CLIENT, and CONFIG.
- Audit logs are only generated for the following commands that you need to configure many parameters for:
   BITOP, MSETNX, PFCOUNT, PFMERGE, HDEL, HMGET, HMSET, HSET, LPUSH, LPUSHX, SADD, SREM, ZADD, GEOADD, GEOHASH, BFINSERT, BFMADD, and BFMEXISTS.
- An audit log is generated only when EXEC executes more than 100 commands in a transaction.
- Logs of Redis Cluster GeminiDB Redis instances cannot be audited.

### **Reporting Audit Logs to LTS**

To use this function, choose **Service Tickets** > **Create Service Ticket** in the upper right corner of the console and contact the customer service.

- 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 Log Reporting.
- **Step 4** Locate the row containing your target instance and click in the **Report Audit Log to LTS** column.
- **Step 5** Select an LTS log group and log stream and click **OK**.

**NOTE** 

You will be billed for enabling this function. For details, see LTS pricing details.

Figure 4-176 Enab	ling audit lo	og reporting to	LTS
-------------------	---------------	-----------------	-----

#### Report Audit Logs to LTS

Logs record all Service (LTS). This request do minutes. You will be bille After this functii If Audit Policy is and you will als Audit Policy, the If you enable au toggle switch tu status becomes	requests sent to your DB instance and are bes not take effect immediately. There is a d ed for log reporting. on is enabled, all audit policies are reported s enabled, LTS reuses the audit policy set f to be billed for reporting audit logs to LTS. ( e fee will be terminated.) udit log reporting to LTS for an instance wit urned on, you can turn off this switch only w s available.	stored in Log Tank Jelay of about 10 d by default. for your DB instance Only after you disable h the Audit Policy /hen the instance
* Log Group	<u> </u>	View Log Groups
★ Log Stream	~ Q	
		OK Cancel

----End

### **Viewing Log Details**

- **Step 1** Log in to the Huawei Cloud console.
- **Step 2** Click in the upper left corner of the page. Under **Management & Governance**, click **Log Tank Service**.
- **Step 3** On the **Log Management** page, click **Log Groups**, select a log group and log stream, and click the name of the selected log stream to go to the details page.

Figure 4-177 Selecting a log stream

Log Groups				
Create Log Group Q Click he	ere to choose a filter condition			S 4
Log Group Name		Remark	Enterprise Pr Log Streams Tags	
^			default 2	Modify Delete Details
Create Log Stream	Click here to choose a filter condit	ion		All Enterprise Pr >
Log Stream Name	Remark	Enterprise Pr Tags	Metric Filters Billing	
		default		2 🕸 🗑 🗇 💬
		default		2 🌣 🗑 🗓 🖘

- **Step 4** Set **From now** to select a relative time (15 minutes by default) in the upper right corner as needed.
- **Step 5** On the **Raw Logs** tab, view the audit logs generated in the relative time period.

•							
< Its-group-sel0 *							
Log Stream Favorited Quick Search	K Is-topic-max( ) Its-topic-max ()						
Q Enter a log stream name.	🗎 Its-topic-muwj 🏠				😳 Feedback 🤆	요 C I 🗰 30 da	iys(From now) • 🛛 🗘 •
E Its-topic-muwj	Q Enter a keyword in the log. Ex	act or fuzzy search by keyword. Exa	ample: "error", "er?or", "rro"", "er"r"				@ 🖿 🛛 Sea
E Its-topic-mxix	Raw Logs Visualization Beta Real-	Time Logs					
E Its-topic-silk	Quick Analysis 🕥 🔄 Collapse			Total 26,482			
		256					
		May 20 21:5 May 23 8:0	0 May 26 8:0 May 29 8:0	Jun 1 8:0 Jun 4 8:0	Jun 7 8:0	Jun 10 8:0 Jun 13 8:0	Jun 16 8:0 Jun 19 8
		Content					≡ ± ≔ (
	No fields added.	Collected 😑	Content				
	Bet Ques Analysis	✓ Jan 18, 2023 18 49 05 547 GMT+68 00	○         Q:         INTER INTERNATION          merrophiledia:i           merrophiledia:i           merrophiledia:i           merrophiledia:i           merrophiledia:i           merrophiledia:           merrophiledia:         _     merrophiledi:         <	098647863110368666612 #59476631338620012			

#### Figure 4-178 Audit logs

----End

# 4.12.4 CTS Audit

## 4.12.4.1 Key Operations Supported by CTS

With CTS, you can record GeminiDB Redis key operations for later query, audit, and backtracking.

Table 4-52	GeminiDB	Redis k	ev operations
	Germinibb	incuis in	cy 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	NoSQLExtendInstanceVo- lume
Resetting the password of an instance	instance	NoSQLResetPassword
Modifying the name of an instance	instance	NoSQLRenameInstance
Binding an EIP	instance	NoSQLResizeInstance

Operation	Resource Type	Trace Name		
Unbinding an EIP	instance	NoSQLBindEIP		
Changing specifications	instance	NoSQLUnBindEIP		
Freezing an instance	instance	NoSQLFreezeInstance		
Unfreezing an instance	instance	NoSQLUnfreezeInstance		
Creating a backup	backup	NoSQLCreateBackup		
Deleting a backup	backup	NoSQLDeleteBackup		
Setting a backup policy	backup	NoSQLSetBackupPolicy		
Adding an instance tag	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	NoSQLUpdateConfigura- tions		
Modifying instance parameters	parameterGroup	NoSQLUpdateInstanceConfi- gurations		
Replicating a parameter template	parameterGroup	NoSQLCopyConfigurations		
Resetting a parameter template	parameterGroup	NoSQLResetConfigurations		
Applying a parameter template	parameterGroup	NoSQLApplyConfigurations		
Deleting a parameter template	parameterGroup	NoSQLDeleteConfigurations		
Deleting the node that fails to be added	instance	NoSQLDeleteEnlargeFail- Node		
Enabling SSL	instance	NoSQLSwitchSSL		
Changing the security group of an instance	instance	NoSQLModifySecurityGroup		
Modifying the recycling policy	instance	NoSQLModifyRecyclePolicy		

## 4.12.4.2 Querying Traces

### Scenarios

After CTS is enabled, CTS starts recording operations on cloud resources. The CTS console stores the last 7 days of operation records for later query, audit, and backtracking.

This section describes how to query the last 7 days of operation records on the CTS console.

### Procedure

- Step 1 Log in to the Huawei Cloud console.
- **Step 2** Click <sup>(Q)</sup> in the upper left corner and select a region and a project.
- Step 3 Click Service List. Under Management & Governance, click Cloud Trace Service.
- **Step 4** Choose **Trace List** in the navigation pane on the left.
- **Step 5** Click **Filter** and specify filter criteria as needed. The following filters are available:
  - Trace Type: Select Management or Data.
  - Trace Source, Resource Type, and Search By Select a filter from the drop-down list.

When you select **Trace name** for **Search By**, you also 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 rather than tenant).
- **Trace Status**: Available options include **All trace statuses**, **normal**, **warning**, and **incident**. You can only select one of them.
- Start time and end time: You can specify a time range for querying traces.
- **Step 6** Click  $\checkmark$  on the left of the record to be queried to extend its details.
- **Step 7** Locate a trace and click **View Trace** in the **Operation** column.

----End

# 4.13 Viewing Metrics and Configuring Alarms

# 4.13.1 Supported Metrics

### Description

This section describes GeminiDB Redis 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.

You can view the instance-level and node-level GeminiDB Redis metrics described in GeminiDB Redis on each instance node by referring to **4.13.4 Viewing Metrics**. The instance-level metrics displayed on each instance node are the same.

Metric Level	Metric Type
Instance	Instance metrics
Node	Basic metrics
	String command metrics
	Hash command metrics
	List command metrics
	Set command metrics.
	Zset command metrics
	Bitmap command metrics
	Stream command metrics
	Geo command metrics
	Hyperloglog command metrics
	Pub/Sub command metrics
	Scripting command metrics
	Transactions command metrics
	ExHash command metrics
	Common command metrics

 Table 4-53 Metric classification

### Namespace

SYS.NoSQL

# **Instance Metrics**

# Supported Instance Metrics

Metric ID	Metric Name	Description	Valu e Ran ge	U ni t	Nu mb er Sys te m	Monitored Object	Mo nito ring Peri od (Ra W Dat a)
redis823_ cluster_p ersist_ke y_num	Keys Without Expiratio n Time in an Instance	Number of keys without expiration time of instances (all databases)	≥ 0	Co un ts	N/ A	GeminiDB Redis instances	1 min ute
redis822_ cluster_c md_err_n um	Number of Comman d Executio n Errors of an Instance	Number of times that an instance command fails to be executed per second. For example, the command does not exist or the parameter is incorrect.	≥ 0	Co un ts/ s	N/ A	GeminiDB Redis instances	1 min ute
redis821_ cluster_o ut_band width_us age	Network Outboun d Bandwidt h Usage	Percentage of traffic sent by an instance to the maximum ELB bandwidth	0- 100	%	N/ A	GeminiDB Redis instances	1 min ute

Metric ID	Metric Name	Description	Valu e Ran ge	U ni t	Nu mb er Sys te m	Monitored Object	Mo nito ring Peri od (Ra W Dat a)
redis820_ cluster_in _bandwi dth_usag e	Network Inbound Bandwidt h Usage	Percentage of traffic received by an instance to the maximum ELB bandwidth. If the inbound bandwidth usage reaches 100%, the maximum bandwidth of the ELB has been exceeded. Submit a service ticket to upgrade the ELB bandwidth. If the ELB bandwidth has reached the maximum 10 Gbit/s, check whether workloads are proper or split them.	0-100	%	N/ A	GeminiDB Redis instances	1 min ute
redis816_ cluster_di sk_usage	Storage Usage	Storage usage of an instance	0– 100	%	N/ A	GeminiDB Redis instances	1 min ute
redis815_ cluster_c pu_usage	CPU Usage	CPU usage of an 0– % N/ GeminiD instance 100 A Redis instance		GeminiDB Redis instances	1 min ute		
redis814_ cluster_ mem_usa ge	Memory Usage	Memory usage of 0– an instance 100		%	N/ A	GeminiDB Redis instances	1 min ute
redis813_ cluster_sl ow_quer y_count	Slow Queries per Instance	Number of slow queries on an instance	≥ 0	Co un ts	N/ A	GeminiDB Redis instances	1 min ute

Metric ID	Metric Name	Description	Valu e Ran ge	U ni t	Nu mb er Sys te m	Monitored Object	Mo nito ring Peri od (Ra W Dat a)
redis812_ cluster_p rocessed_ comman d_count	Total Comman ds per Instance	Total commands processed by a node	≥ 0	Co un ts	N/ A	GeminiDB Redis instances	1 min ute
redis811_ cluster_ max_con nect_cou nt	Max. Connecti ons per Instance	Maximum connections to an instance (nodes × 10,000)	≥ 0	Co un ts	N/ A	GeminiDB Redis instances	1 min ute
redis808_ cluster_n ew_client _connecti on	Connecti ons Created Per Second	Instance connections created per second	≥ 0	Co un ts	N/ A	GeminiDB Redis instances	1 min ute
redis807_ cluster_al l_connect ion_coun t	Total Connecti ons	Connections of an instance (used connections of all nodes)	≥ 0	Co un ts	N/ A	GeminiDB Redis instances	1 min ute
redis806_ cluster_ max_con nection_ usage	Connecti on Usage	Max. node connection usage of the instance	0– 100	%	N/ A	GeminiDB Redis instances	1 min ute
redis805_ cluster_a vg_hit_ra te	Average Hit Rate of the Instance	Average hit rate of multiple nodes in the instance	0– 100	%	N/ A	GeminiDB Redis instances	1 min ute
redis804_ cluster_al l_p99_us ec	p99 Latency	p99 latency of the instance	≥ 0	μs	N/ A	GeminiDB Redis instances	1 min ute
redis803_ cluster_al l_avg_use c	Average Latency	Average latency of the instance	≥ 0	μs	N/ A	GeminiDB Redis instances	1 min ute

Metric ID	Metric Name	Description	Valu e Ran ge	U ni t	Nu mb er Sys te m	Monitored Object	Mo nito ring Peri od (Ra w Dat a)
redis802_ cluster_ max_resp onse_arg c	Max. Elements Obtained in a Request	Max. elements obtained by the client in a request	≥ 0	Co un ts	N/ A	GeminiDB Redis instances	1 min ute
redis801_ cluster_ max_resp onse_byt es	Max. Bytes Obtained in a Request	Max. bytes obtained by the client in a request	≥ 0	By te s	102 4 (IE C)	GeminiDB Redis instances	1 min ute
redis800_ cluster_ max_req uest_argc	Max. Paramete rs Sent in a Request	Max. parameters sent in a request	≥ 0	Co un ts	N/ A	GeminiDB Redis instances	1 min ute
redis799_ cluster_ max_req uest_byte s	Max. Bytes Sent in a Request	Max. bytes sent in a request	≥ 0	By te s	102 4 (IE C)	GeminiDB Redis instances	1 min ute
redis798_ cluster_e xpire_key _counts	Keys with an Expiratio n Time Configur ed	Instance keys with an expiration time configured	≥ 0	Co un ts	N/ A	GeminiDB Redis instances	1 min ute
redis689_ qps_recei ve_total	Total Traffic Received by the Instance	Total traffic received by the instance, a reflection of the traffic volume on the application side.	≥ 0	By te s/s	102 4 (IE C)	GeminiDB Redis instances	1 min ute

Metric ID	Metric Name	Description	Valu e Ran ge	U ni t	Nu mb er Sys te m	Monitored Object	Mo nito ring Peri od (Ra W Dat a)
redis688_ qps_send _total	Total Traffic Sent by the Instance	Total traffic sent by the instance, a reflection of the traffic volume on the application side.	≥ 0	By te s/s	102 4 (IE C)	GeminiDB Redis instances	1 min ute
redis668_ cluster_k ey_count s	Instance Keys	Total keys of a cluster	≥ 0	Co un ts	N/ A	GeminiDB Redis instances	1 min ute
redis667_ cluster_q ps	QPS of the Instance	The value of this metric is the QPS of the instance.	≥ 0	Co un ts/ s	N/ A	GeminiDB Redis instances	1 min ute
redis834_ cluster_d bcache_r ecv_bps	DB Cache Synchron ization Rate	Rate of traffic received by GeminiDB Redis instances when DB Cache is used	≥ 0	By te s/s	102 4 (IE C)	GeminiDB Redis instances	1 min ute
redis833_ cluster_d bcache_r eplicatio n_delay	DB Cache Synchron ization Latency	Synchronization delay between a primary RDS database and a standby GeminiDB Redis database when DB Cache is used. A smaller value indicates more timely synchronization.	≥ 0	m s	N/ A	GeminiDB Redis instances	1 min ute

Metric ID	Metric Name	Description	Valu e Ran ge	U ni t	Nu mb er Sys te m	Monitored Object	Mo nito ring Peri od (Ra w Dat a)
redis832_ cluster_d bcache_b inlog_del ay	Variation in Rows of the Binlogs Between the Primary RDS and Standby GeminiD B Redis With DB Cache	Difference of synchronization start points between a primary RDS database and a standby GeminiDB Redis database when DB Cache is used. A smaller value indicates more timely synchronization.	≥ 0	Co un ts	N/ A	GeminiDB Redis instances	1 min ute
redis831_ cluster_d bcache_t otal_ops	DB Cache Executio ns per Second	Number of DB Cache executions per second	≥ 0	Co un ts/ s	N/ A	GeminiDB Redis instances	1 min ute

## **Node Metrics**

## **Basic Metrics**

	Table	4-54	Basic	metrics
--	-------	------	-------	---------

Metric ID	Metric Name	Descripti on	Value Rang e	Unit	Nu mb er Sys te m	Monitor ed Object	Monitori ng Period (Raw Data)
redis817_ slow_que ry_count	Slow Queries per Node	Slow queries on a node	≥ 0	Count s	N/A	GeminiD B Redis instance nodes	1 minute

Metric ID	Metric Name	Descripti on	Value Rang e	Unit	Nu mb er Sys te m	Monitor ed Object	Monitori ng Period (Raw Data)
redis810_ max_con nect_cou nt	Max. Connecti ons per Node	Maximu m connecti ons to a node	≥ 0	Count s	N/A	GeminiD B Redis instance nodes	1 minute
redis809_ processe d_comm and_cou nt	Total Comman ds per Node	Total comman ds processe d by a node	≥ 0	Count s	N/A	GeminiD B Redis instance nodes	1 minute
redis687_ client_ne twork_pa cket_retu rn_p99	p99 Latency for Packets Returnin g to the Client (Send)	p99 latency for packets sent from the proxy to the client, a reflection of the network quality on the applicati on side.	≥ 0	μs	N/A	GeminiD B Redis instance nodes	1 minute
redis686_ client_ne twork_pa cket_retu rn_max	Maximu m Speed for Packets Returnin g to the Client (Send)	Maximu m speed for packets sent by the proxy to the client, a reflection of the network quality on the applicati on side.	≥ 0	μs	N/A	GeminiD B Redis instance nodes	1 minute

Metric ID	Metric Name	Descripti on	Value Rang e	Unit	Nu mb er Sys te m	Monitor ed Object	Monitori ng Period (Raw Data)
redis685_ client_ne twork_pa cket_retu rn_avg	Average Speed for Packets Returnin g to the Client (Send)	Average speed for packets sent by the proxy to the client, a reflection of the network quality on the applicati on side.	≥ 0	μs	N/A	GeminiD B Redis instance nodes	1 minute
redis684_ client_pa cket_retu rn_queue _p99	p99 Latency for Packets Returnin g to the Client (Queuing )	Queuing p99 latency for packets returned from the proxy to the client, a reflection of the network quality on the applicati on side.	≥ 0	μs	N/A	GeminiD B Redis instance nodes	1 minute

Metric ID	Metric Name	Descripti on	Value Rang e	Unit	Nu mb er Sys te m	Monitor ed Object	Monitori ng Period (Raw Data)
redis683_ client_pa cket_retu rn_queue _max	Maximu m Speed for Packets Returnin g to the Client (Queuing )	Maximu m queuing speed for packets returned from the proxy to the client, a reflection of the network quality on the applicati on side.	≥ 0	μs	N/A	GeminiD B Redis instance nodes	1 minute
redis682_ client_pa cket_retu rn_queue _avg	Average Speed for Packets Returnin g to the Client (Queuing )	Average queuing speed for packets returned from the proxy to the client, a reflection of the network quality on the applicati on side.	≥ 0	μs	N/A	GeminiD B Redis instance nodes	1 minute
redis678_ max_resp onse_arg c	Maximu m Elements Obtained in a Request	Maximu m elements obtained by the client in a request	≥ 0	Count s	N/A	GeminiD B Redis instance nodes	1 minute

Metric ID	Metric Name	Descripti on	Value Rang e	Unit	Nu mb er Sys te m	Monitor ed Object	Monitori ng Period (Raw Data)
redis677_ max_resp onse_byt es	Maximu m Bytes Obtained in a Request	Maximu m bytes obtained by the client in a request	≥ 0	Bytes	102 4 (IE C)	GeminiD B Redis instance nodes	1 minute
redis676_ max_req uest_argc	Maximu m Paramete rs Sent in a Request	Maximu m paramete rs sent by the client in a request	≥ 0	Count s	N/A	GeminiD B Redis instance nodes	1 minute
redis675_ max_req uest_byte s	Maximu m Bytes Sent in a Request	Maximu m bytes sent by the client in a request	≥ 0	Bytes	102 4 (IE C)	GeminiD B Redis instance nodes	1 minute
redis674_ max_pipe lined	Maximu m Comman ds Sent in a Pipeline	Maximu m comman ds batch sent by the client in a pipeline	≥ 0	Count s	N/A	GeminiD B Redis instance nodes	1 minute
redis673_ wrong_a uth	Failed Authenti cation Attempts	Failed authentic ation attempts per second on a node	≥ 0	Count s	N/A	GeminiD B Redis instance nodes	1 minute

Metric ID	Metric Name	Descripti on	Value Rang e	Unit	Nu mb er Sys te m	Monitor ed Object	Monitori ng Period (Raw Data)
redis672_ no_auth	Request Attempts Due to Authenti cation Failure	Failed request attempts due to authentic ation failure on a node per second	≥ 0	Count s	N/A	GeminiD B Redis instance nodes	1 minute
redis671_ new_clie nt_conne ction	New Connecti ons	Connecti ons created on a node per second	≥ 0	Count s	N/A	GeminiD B Redis instance nodes	1 minute
redis670_ hit_rate	Hit Rate of a Key in Underlyi ng Storage	Hit percenta ge of a key in underlyin g storage in a collection period. Formula: Hit keys/ (Hit keys + Missed keys).	0-100	%	N/A	GeminiD B Redis instance nodes	1 minute
redis669_ connecti on_usage	Connecti on Usage	Connecti on usage in a collection period. Formula: Used connecti ons/Total connecti ons.	0–100	%	N/A	GeminiD B Redis instance nodes	1 minute

Metric ID	Metric Name	Descripti on	Value Rang e	Unit	Nu mb er Sys te m	Monitor ed Object	Monitori ng Period (Raw Data)
redis319_ all_qps	Proxy QPS	Proxy QPS on a node	≥ 0	Count s/s	N/A	GeminiD B Redis instance nodes	1 minute
redis318_ all_p99	p99 Access Latency	p99 latency when a node executes all comman ds	≥ 0	μs	N/A	GeminiD B Redis instance nodes	1 minute
redis317_ all_max_ usec	Maximu m Access Latency	Maximu m latency when a node executes all comman ds	≥ 0	μs	N/A	GeminiD B Redis instance nodes	1 minute
redis316_ all_avg_u sec	Average Access Latency	Average latency when a node executes all comman ds	≥ 0	μs	N/A	GeminiD B Redis instance nodes	1 minute
redis032_ shard_qp s	Shard QPS	Shard QPS on a node	≥ 0	Count s/s	N/A	GeminiD B Redis instance nodes	1 minute
redis021_ proxy_se nd_client _bps	Traffic Send Speed	Outgoing traffic speed of a node /s	≥ 0	Bytes/ s	102 4 (IE C)	GeminiD B Redis instance nodes	1 minute

Metric ID	Metric Name	Descripti on	Value Rang e	Unit	Nu mb er Sys te m	Monitor ed Object	Monitori ng Period (Raw Data)
redis020_ proxy_rec v_client_ bps	Traffic Receive Speed	Incoming traffic speed of a node /s	≥ 0	Bytes/ s	102 4 (IE C)	GeminiD B Redis instance nodes	1 minute
gemini00 4_bytes_i n	Network Input Through put	Average traffic received by all network adapters of a monitore d object per second	≥ 0	Bytes/ s	102 4 (IE C)	GeminiD B Redis instance nodes	1 minute
gemini00 3_bytes_ out	Network Output Through put	Average traffic sent from all network adapters of a monitore d object per second	≥ 0	Bytes/ s	102 4 (IE C)	GeminiD B Redis instance nodes	1 minute
redis019_ proxy_res ponse_ps	Proxy Response Rate	Speed at which proxy responds to clients	≥ 0	Count s/s	N/A	GeminiD B Redis instance nodes	1 minute
redis018_ proxy_re quest_ps	Request Receive Speed	Speed at which proxy receives requests from clients	≥ 0	Count s/s	N/A	GeminiD B Redis instance nodes	1 minute

Metric ID	Metric Name	Descripti on	Value Rang e	Unit	Nu mb er Sys te m	Monitor ed Object	Monitori ng Period (Raw Data)
redis017_ proxy_ac cept	Client Connecti ons	Client connecti ons to a node	≥ 0	Count s	N/A	GeminiD B Redis instance nodes	1 minute
nosql007 _disk_use d_size	Storage Space Usage	Used storage space of an instance	≥ 0	GB	102 4 (IE C)	GeminiD B Redis instance nodes	1 minute
nosql006 _disk_tot al_size	Total Storage Space	Total storage space of an instance	≥ 0	GB	102 4 (IE C)	GeminiD B Redis instance nodes	1 minute
nosql005 _disk_usa ge	Storage Usage	Storage usage of the current instance.	0–100	%	N/A	GeminiD B Redis instance nodes	1 minute
nosql002 _mem_us age	Memory Usage	Memory usage of the monitore d system	0–100	%	N/A	GeminiD B Redis instance nodes	1 minute
nosql001 _cpu_usa ge	CPU Usage	CPU usage of the monitore d system	0–100	%	N/A	GeminiD B Redis instance nodes	1 minute

# String Command Metrics

Table 4-55 String command metrics
-----------------------------------

Metric ID	Metric Name	Descripti on	Value Range	Un it	Nu mb er Sys te m	Monitor ed Object	Monitori ng Period (Raw Data)
redis107_ decr_qps	DECR QPS	QPS when a node executes the DECR comman d	≥ 0	Co unt s/s	N/A	GeminiD B Redis instance nodes	1 minute
redis106_ decr_p99	DECR p99 Latency	p99 latency when a node executes the DECR comman d	≥ 0	μs	N/A	GeminiD B Redis instance nodes	1 minute
redis104_ decr_avg _usec	DECR Average Latency	Average latency when a node executes the DECR comman d	≥ 0	μs	N/A	GeminiD B Redis instance nodes	1 minute
redis103_ incr_qps	INCR QPS	QPS when a node executes the INCR comman d	≥ 0	Co unt s/s	N/A	GeminiD B Redis instance nodes	1 minute
redis102_ incr_p99	INCR p99 Latency	p99 latency when a node executes the INCR comman d	≥ 0	μs	N/A	GeminiD B Redis instance nodes	1 minute

Metric ID	Metric Name	Descripti on	Value Range	Un it	Nu mb er Sys te m	Monitor ed Object	Monitori ng Period (Raw Data)
redis100_ incr_avg_ usec	INCR Average Latency	Average latency when a node executes the INCR comman d	≥ 0	μs	N/A	GeminiD B Redis instance nodes	1 minute
redis099_ strlen_qp s	STRLEN QPS	QPS when a node executes the STRLEN comman d	≥ 0	Co unt s/s	N/A	GeminiD B Redis instance nodes	1 minute
redis098_ strlen_p9 9	STRLEN p99 Latency	p99 latency when a node executes the STRLEN comman d	≥ 0	μs	N/A	GeminiD B Redis instance nodes	1 minute
redis096_ strlen_av g_usec	STRLEN Average Latency	Average latency when a node executes the STRLEN comman d	≥ 0	μs	N/A	GeminiD B Redis instance nodes	1 minute
redis095_ substr_qp s	SUBSTR QPS	QPS when a node executes the SUBSTR comman d	≥ 0	Co unt s/s	N/A	GeminiD B Redis instance nodes	1 minute

Metric ID	Metric Name	Descripti on	Value Range	Un it	Nu mb er Sys te m	Monitor ed Object	Monitori ng Period (Raw Data)
redis094_ substr_p9 9	SUBSTR p99 Latency	p99 latency when a node executes the SUBSTR comman d	≥ 0	μs	N/A	GeminiD B Redis instance nodes	1 minute
redis092_ substr_av g_usec	SUBSTR Average Latency	Average latency when a node executes the SUBSTR comman d	≥ 0	μs	N/A	GeminiD B Redis instance nodes	1 minute
redis091_ setrange_ qps	SETRAN GE QPS	QPS when a node executes the SETRAN GE comman d	≥ 0	Co unt s/s	N/A	GeminiD B Redis instance nodes	1 minute
redis090_ setrange_ p99	SETRAN GE p99 Latency	p99 latency when a node executes the SETRAN GE comman d	≥ 0	μs	N/A	GeminiD B Redis instance nodes	1 minute

Metric ID	Metric Name	Descripti on	Value Range	Un it	Nu mb er Sys te m	Monitor ed Object	Monitori ng Period (Raw Data)
redis088_ setrange_ avg_usec	SETRAN GE Average Latency	Average latency when a node executes the SETRAN GE comman d	≥ 0	μs	N/A	GeminiD B Redis instance nodes	1 minute
redis087_ getrange _qps	GETRAN GE QPS	QPS when a node executes the GETRAN GE comman d	≥ 0	Co unt s/s	N/A	GeminiD B Redis instance nodes	1 minute
redis086_ getrange _p99	GETRAN GE p99 Latency	p99 latency when a node executes the GETRAN GE comman d	≥ 0	μs	N/A	GeminiD B Redis instance nodes	1 minute
redis084_ getrange _avg_use c	GETRAN GE Average Latency	Average latency when a node executes the GETRAN GE comman d	≥ 0	μs	N/A	GeminiD B Redis instance nodes	1 minute

Metric ID	Metric Name	Descripti on	Value Range	Un it	Nu mb er Sys te m	Monitor ed Object	Monitori ng Period (Raw Data)
redis083_ mset_qps	MSET QPS	QPS when a node executes the MSET comman d	≥ 0	Co unt s/s	N/A	GeminiD B Redis instance nodes	1 minute
redis082_ mset_p9 9	MSET p99 Latency	p99 latency when a node executes the MSET comman d	≥ 0	μs	N/A	GeminiD B Redis instance nodes	1 minute
redis080_ mset_avg _usec	MSET Average Latency	Average latency when a node executes the MSET comman d	≥ 0	μs	N/A	GeminiD B Redis instance nodes	1 minute
redis079_ mget_qp s	MGET QPS	QPS when a node executes the MGET comman d	≥ 0	Co unt s/s	N/A	GeminiD B Redis instance nodes	1 minute
redis078_ mget_p9 9	MGET p99 Latency	p99 latency when a node executes the MGET comman d	≥ 0	μs	N/A	GeminiD B Redis instance nodes	1 minute

Metric ID	Metric Name	Descripti on	Value Range	Un it	Nu mb er Sys te m	Monitor ed Object	Monitori ng Period (Raw Data)
redis076_ mget_av g_usec	MGET Average Latency	Average latency when a node executes the MGET comman d	≥ 0	μs	N/A	GeminiD B Redis instance nodes	1 minute
redis075_ append_ qps	APPEND QPS	QPS when a node executes the APPEND comman d	≥ 0	Co unt s/s	N/A	GeminiD B Redis instance nodes	1 minute
redis074_ append_ p99	APPEND p99 Latency	p99 latency when a node executes the APPEND comman d	≥ 0	μs	N/A	GeminiD B Redis instance nodes	1 minute
redis072_ append_ avg_usec	APPEND Average Latency	Average latency when a node executes the APPEND comman d	≥ 0	μs	N/A	GeminiD B Redis instance nodes	1 minute
redis071_ getset_q ps	GETSET QPS	QPS when a node executes the GETSET comman d	≥ 0	Co unt s/s	N/A	GeminiD B Redis instance nodes	1 minute

Metric ID	Metric Name	Descripti on	Value Range	Un it	Nu mb er Sys te m	Monitor ed Object	Monitori ng Period (Raw Data)
redis070_ getset_p 99	GETSET p99 Latency	p99 latency when a node executes the GETSET comman d	≥ 0	μs	N/A	GeminiD B Redis instance nodes	1 minute
redis068_ getset_av g_usec	GETSET Average Latency	Average latency when a node executes the GETSET comman d	≥ 0	μs	N/A	GeminiD B Redis instance nodes	1 minute
redis067_ get_qps	GET QPS	QPS when a node executes the GET comman d	≥ 0	Co unt s/s	N/A	GeminiD B Redis instance nodes	1 minute
redis066_ get_p99	GET p99 Latency	p99 latency when a node executes the GET comman d	≥ 0	μs	N/A	GeminiD B Redis instance nodes	1 minute
redis064_ get_avg_ usec	GET Average Latency	Average latency when a node executes the GET comman d	≥ 0	μs	N/A	GeminiD B Redis instance nodes	1 minute

Metric ID	Metric Name	Descripti on	Value Range	Un it	Nu mb er Sys te m	Monitor ed Object	Monitori ng Period (Raw Data)
redis063_ set_qps	SET QPS	QPS when a node executes the SET comman d	≥ 0	Co unt s/s	N/A	GeminiD B Redis instance nodes	1 minute
redis062_ set_p99	SET p99 Latency	p99 latency when a node executes the SET comman d	≥ 0	μs	N/A	GeminiD B Redis instance nodes	1 minute
redis060_ set_avg_ usec	SET Average Latency	Average latency when a node executes the SET comman d	≥ 0	μs	N/A	GeminiD B Redis instance nodes	1 minute

## Hash Command Metrics

 Table 4-56 Hash command metrics

Metric ID	Metric Name	Description	Valu e Ran ge	Un it	Nu mb er Sys te m	Monitored Object	Moni torin g Perio d (Raw Data )
redis159_ hscan_qp s	HSCAN QPS	QPS when a node executes the HSCAN command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minu te

Metric ID	Metric Name	Description	Valu e Ran ge	Un it	Nu mb er Sys te m	Monitored Object	Moni torin g Perio d (Raw Data )
redis158_ hscan_p9 9	HSCAN p99 Latency	p99 latency when a node executes the HSCAN command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis156_ hscan_av g_usec	HSCAN Average Latency	Average latency when a node executes the HSCAN command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis155_ hvals_qps	HVALS QPS	QPS when a node executes the HVALS command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minu te
redis154_ hvals_p9 9	HVALS p99 Latency	p99 latency when a node executes the HVALS command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis152_ hvals_av g_usec	HVALS Average Latency	Average latency when a node executes the HVALS command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis151_ hstrlen_q ps	HSTRLEN QPS	QPS when a node executes the HSTRLEN command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minu te
redis150_ hstrlen_p 99	HSTRLEN p99 Latency	p99 latency when a node executes the HSTRLEN command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te

Metric ID	Metric Name	Description	Valu e Ran ge	Un it	Nu mb er Sys te m	Monitored Object	Moni torin g Perio d (Raw Data )
redis148_ hstrlen_a vg_usec	HSTRLEN Average Latency	Average latency when a node executes the HSTRLEN command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis147_ hlen_qps	HLEN QPS	QPS when a node executes the HLEN command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minu te
redis146_ hlen_p99	HLEN p99 Latency	p99 latency when a node executes the HLEN command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis144_ hlen_avg _usec	HLEN Average Latency	Average latency when a node executes the HLEN command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis143_ hkeys_qp s	HKEYS QPS	QPS when a node executes the HKEYS command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minu te
redis142_ hkeys_p9 9	HKEYS p99 Latency	p99 latency when a node executes the HKEYS command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis140_ hkeys_av g_usec	HKEYS Average Latency	Average latency when a node executes the HKEYS command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te

Metric ID	Metric Name	Description	Valu e Ran ge	Un it	Nu mb er Sys te m	Monitored Object	Moni torin g Perio d (Raw Data )
redis139_ hincrby_q ps	HINCRBY QPS	QPS when a node executes the HINCRBY command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minu te
redis138_ hincrby_p 99	HINCRBY p99 Latency	p99 latency when a node executes the HINCRBY command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis136_ hincrby_a vg_usec	HINCRBY Average Latency	Average latency when a node executes the HINCRBY command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis135_ hexists_q ps	HEXISTS QPS	QPS when a node executes the HEXISTS command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minu te
redis134_ hexists_p 99	HEXISTS p99 Latency	p99 latency when a node executes the HEXISTS command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis132_ hexists_a vg_usec	HEXISTS Average Latency	Average latency when a node executes the HEXISTS command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis131_ hgetall_q ps	HGETALL QPS	QPS when a node executes the HGETALL command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minu te

Metric ID	Metric Name	Description	Valu e Ran ge	Un it	Nu mb er Sys te m	Monitored Object	Moni torin g Perio d (Raw Data )
redis130_ hgetall_p 99	HGETALL p99 Latency	p99 latency when a node executes the HGETALL command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis128_ hgetall_a vg_usec	HGETALL Average Latency	Average latency when a node executes the HGETALL command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis127_ hdel_qps	HDEL QPS	QPS when a node executes the HDEL command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minu te
redis126_ hdel_p99	HDEL p99 Latency	p99 latency when a node executes the HDEL command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis124_ hdel_avg _usec	HDEL Average Latency	Average latency when a node executes the HDEL command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis123_ hmget_q ps	HMGET QPS	QPS when a node executes the HMGET command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minu te
redis122_ hmget_p 99	HMGET p99 Latency	p99 latency when a node executes the HMGET command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
Metric ID	Metric Name	Description	Valu e Ran ge	Un it	Nu mb er Sys te m	Monitored Object	Moni torin g Perio d (Raw Data )
---------------------------------	-----------------------------	---	------------------------	------------------	----------------------------------	--	---
redis120_ hmget_a vg_usec	HMGET Average Latency	Average latency when a node executes the HMGET command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis119_ hmset_q ps	HMSET QPS	QPS when a node executes the HMSET command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minu te
redis118_ hmset_p 99	HMSET p99 Latency	p99 latency when a node executes the HMSET command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis116_ hmset_av g_usec	HMSET Average Latency	Average latency when a node executes the HMSET command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis115_ hget_qps	HGET QPS	QPS when a node executes the HGET command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minu te
redis114_ hget_p99	HGET p99 Latency	p99 latency when a node executes the HGET command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis112_ hget_avg _usec	HGET Average Latency	Average latency when a node executes the HGET command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te

Metric ID	Metric Name	Description	Valu e Ran ge	Un it	Nu mb er Sys te m	Monitored Object	Moni torin g Perio d (Raw Data )
redis111_ hset_qps	HSET QPS	QPS when a node executes the HSET command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minu te
redis110_ hset_p99	HSET p99 Latency	p99 latency when a node executes the HSET command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis108_ hset_avg _usec	HSET Average Latency	Average latency when a node executes the HSET command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te

# List Command Metrics

Table 4-57 List co	ommand metrics
--------------------	----------------

Metric ID	Metric Name	Description	Value Rang e	Un it	Nu mb er Sys te m	Monitored Object	Moni torin g Perio d (Raw Data)
redis207_ ltrim_qps	LTRIM QPS	QPS when a node executes the LTRIM command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minut e

Metric ID	Metric Name	Description	Value Rang e	Un it	Nu mb er Sys te m	Monitored Object	Moni torin g Perio d (Raw Data)
redis206_ ltrim_p99	LTRIM p99 Latency	p99 latency when a node executes the LTRIM command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis204_ ltrim_avg _usec	LTRIM Average Latency	Average latency when a node executes the LTRIM command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis203_ lset_qps	LSET QPS	QPS when a node executes the LSET command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minut e
redis202_ lset_p99	LSET p99 Latency	p99 latency when a node executes the LSET command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis200_ lset_avg_ usec	LSET Average Latency	Average latency when a node executes the LSET command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis199_ lrem_qps	LREM QPS	QPS when a node executes the LREM command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minut e

Metric ID	Metric Name	Description	Value Rang e	Un it	Nu mb er Sys te m	Monitored Object	Moni torin g Perio d (Raw Data)
redis198_ lrem_p99	LREM p99 Latency	p99 latency when a node executes the LREM command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis196_ lrem_avg _usec	LREM Average Latency	Average latency when a node executes the LREM command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis195_ lrange_q ps	LRANGE QPS	QPS when a node executes the LRANGE command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minut e
redis194_ lrange_p 99	LRANGE p99 Latency	p99 latency when a node executes the LRANGE command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis192_ lrange_a vg_usec	LRANGE Average Latency	Average latency when a node executes the LRANGE command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis191_ linsert_q ps	LINSERT QPS	QPS when a node executes the LINSERT command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minut e

Metric ID	Metric Name	Description	Value Rang e	Un it	Nu mb er Sys te m	Monitored Object	Moni torin g Perio d (Raw Data)
redis190_ linsert_p 99	LINSERT p99 Latency	p99 latency when a node executes the LINSERT command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis188_ linsert_av g_usec	LINSERT Average Latency	Average latency when a node executes the LINSERT command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis187_ lindex_qp s	LINDEX QPS	QPS when a node executes the LINDEX command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minut e
redis186_ lindex_p9 9	LINDEX p99 Latency	p99 latency when a node executes the LINDEX command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis184_ lindex_av g_usec	LINDEX Average Latency	Average latency when a node executes the LINDEX command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis183_ llen_qps	LLEN QPS	QPS when a node executes the LLEN command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minut e

Metric ID	Metric Name	Description	Value Rang e	Un it	Nu mb er Sys te m	Monitored Object	Moni torin g Perio d (Raw Data)
redis182_ llen_p99	LLEN p99 Latency	p99 latency when a node executes the LLEN command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis180_ llen_avg_ usec	LLEN Average Latency	Average latency when a node executes the LLEN command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis179_ rpoplpus h_qps	RPOPLPU SH QPS	QPS when a node executes the RPOPLPUSH command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minut e
redis178_ rpoplpus h_p99	RPOPLPU SH p99 Latency	p99 latency when a node executes the RPOPLPUSH command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis176_ rpoplpus h_avg_us ec	RPOPLPU SH Average Latency	Average latency when a node executes the RPOPLPUSH command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis175_ rpop_qps	RPOP QPS	QPS when a node executes the RPOP command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minut e

Metric ID	Metric Name	Description	Value Rang e	Un it	Nu mb er Sys te m	Monitored Object	Moni torin g Perio d (Raw Data)
redis174_ rpop_p99	RPOP p99 Latency	p99 latency when a node executes the RPOP command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis172_ rpop_avg _usec	RPOP Average Latency	Average latency when a node executes the RPOP command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis171_ rpush_qp s	RPUSH QPS	QPS when a node executes the RPUSH command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minut e
redis170_ rpush_p9 9	RPUSH p99 Latency	p99 latency when a node executes the RPUSH command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis168_ rpush_av g_usec	RPUSH Average Latency	Average latency when a node executes the RPUSH command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis167_ lpop_qps	LPOP QPS	QPS when a node executes the LPOP command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minut e

Metric ID	Metric Name	Description	Value Rang e	Un it	Nu mb er Sys te m	Monitored Object	Moni torin g Perio d (Raw Data)
redis166_ lpop_p99	LPOP p99 Latency	p99 latency when a node executes the LPOP command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis164_ lpop_avg _usec	LPOP Average Latency	Average latency when a node executes the LPOP command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis163_ lpush_qp s	LPUSH QPS	QPS when a node executes the LPUSH command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minut e
redis162_ lpush_p9 9	LPUSH p99 Latency	p99 latency when a node executes the LPUSH command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis160_ lpush_av g_usec	LPUSH Average Latency	Average latency when a node executes the LPUSH command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis470_ brpoplpu sh_qps	BRPOPLP USH QPS	QPS when a node executes the BRPOPLPUS H command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minut e

Metric ID	Metric Name	Description	Value Rang e	Un it	Nu mb er Sys te m	Monitored Object	Moni torin g Perio d (Raw Data)
redis468_ brpoplpu sh_p99	BRPOPLP USH p99 Latency	p99 latency when a node executes the BRPOPLPUS H command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis466_ brpoplpu sh_avg_u sec	BRPOPLP USH Average Latency	Average latency when a node executes the BRPOPLPUS H command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis465_ brpop_qp s	BRPOP QPS	QPS when a node executes the BRPOP command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minut e
redis463_ brpop_p9 9	BRPOP p99 Latency	p99 latency when a node executes the BRPOP command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis461_ brpop_av g_usec	BRPOP Average Latency	Average latency when a node executes the BRPOP command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis460_ blpop_qp s	BLPOP QPS	QPS when a node executes the BLPOP command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minut e

Metric ID	Metric Name	Description	Value Rang e	Un it	Nu mb er Sys te m	Monitored Object	Moni torin g Perio d (Raw Data)
redis458_ blpop_p9 9	BLPOP p99 Latency	p99 latency when a node executes the BLPOP command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis456_ blpop_av g_usec	BLPOP Average Latency	Average latency when a node executes the BLPOP command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e

## Set Command Metrics

Metric ID	Metric Name	Description	Valu e Rang e	Un it	Nu mb er Sys te m	Monitored Object	Moni torin g Perio d (Raw Data )
redis247_ srandme mber_qp s	SRANDM EMBER QPS	QPS when a node executes the SRANDMEM BER command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minut e
redis246_ srandme mber_p9 9	SRANDM EMBER p99 Latency	p99 latency when a node executes the SRANDMEM BER command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e

Metric ID	Metric Name	Description	Valu e Rang e	Un it	Nu mb er Sys te m	Monitored Object	Moni torin g Perio d (Raw Data )
redis244_ srandme mber_av g_usec	SRANDM EMBER Average Latency	Average latency when a node executes the SRANDMEM BER command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis243_ sdiff_qps	SDIFF QPS	QPS when a node executes the SDIFF command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minut e
redis242_ sdiff_p99	SDIFF p99 Latency	p99 latency when a node executes the SDIFF command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis240_ sdiff_avg _usec	SDIFF Average Latency	Average latency when a node executes the SDIFF command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis239_ sismemb er_qps	SISMEMB ER QPS	QPS when a node executes the SISMEMBER command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minut e
redis238_ sismemb er_p99	SISMEMB ER p99 Latency	p99 latency when a node executes the SISMEMBER command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis236_ sismemb er_avg_u sec	SISMEMB ER Average Latency	Average latency when a node executes the SISMEMBER command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e

Metric ID	Metric Name	Description	Valu e Rang e	Un it	Nu mb er Sys te m	Monitored Object	Moni torin g Perio d (Raw Data )
redis235_ sinter_qp s	SINTER QPS	QPS when a node executes the SINTER command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minut e
redis234_ sinter_p9 9	SINTER p99 Latency	p99 latency when a node executes the SINTER command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis232_ sinter_av g_usec	SINTER Average Latency	Average latency when a node executes the SINTER command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis231_ sunion_q ps	SUNION QPS	QPS when a node executes the SUNION command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minut e
redis230_ sunion_p 99	SUNION p99 Latency	p99 latency when a node executes the SUNION command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis228_ sunion_a vg_usec	SUNION Average Latency	Average latency when a node executes the SUNION command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis227_ srem_qps	SREM QPS	QPS when a node executes the SREM command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minut e

Metric ID	Metric Name	Description	Valu e Rang e	Un it	Nu mb er Sys te m	Monitored Object	Moni torin g Perio d (Raw Data )
redis226_ srem_p99	SREM p99 Latency	p99 latency when a node executes the SREM command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis224_ srem_avg _usec	SREM Average Latency	Average latency when a node executes the SREM command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis223_ smember s_qps	SMEMBE RS QPS	QPS when a node executes the SMEMBERS command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minut e
redis222_ smember s_p99	SMEMBE RS p99 Latency	p99 latency when a node executes the SMEMBERS command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis220_ smember s_avg_us ec	SMEMBE RS Average Latency	Average latency when a node executes the SMEMBERS command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis219_ scard_qp s	SCARD QPS	QPS when a node executes the SCARD command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minut e
redis218_ scard_p9 9	SCARD p99 Latency	p99 latency when a node executes the SCARD command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e

Metric ID	Metric Name	Description	Valu e Rang e	Un it	Nu mb er Sys te m	Monitored Object	Moni torin g Perio d (Raw Data )
redis216_ scard_av g_usec	SCARD Average Latency	Average latency when a node executes the SCARD command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis215_ spop_qps	SPOP QPS	QPS when a node executes the SPOP command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minut e
redis214_ spop_p99	SPOP p99 Latency	p99 latency when a node executes the SPOP command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis212_ spop_avg _usec	SPOP Average Latency	Average latency when a node executes the SPOP command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis211_ sadd_qps	SADD QPS	QPS when a node executes the SADD command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minut e
redis210_ sadd_p99	SADD p99 Latency	p99 latency when a node executes the SADD command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis208_ sadd_avg _usec	SADD Average Latency	Average latency when a node executes the SADD command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e

Metric ID	Metric Name	Description	Valu e Rang e	Un it	Nu mb er Sys te m	Monitored Object	Moni torin g Perio d (Raw Data )
redis485_ sdiffstore _qps	SDIFFST ORE QPS	QPS when a node executes the SDIFFSTORE command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minut e
redis483_ sdiffstore _p99	SDIFFST ORE p99 Latency	p99 latency when a node executes the SDIFFSTORE command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis481_ sdiffstore _avg_use c	SDIFFST ORE Average Latency	Average latency when a node executes the SDIFFSTORE command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis480_ sinterstor e_qps	SINTERS TORE QPS	QPS when a node executes the SINTERSTOR E command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minut e
redis478_ sinterstor e_p99	SINTERS TORE p99 Latency	p99 latency when a node executes the SINTERSTOR E command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis476_ sinterstor e_avg_us ec	SINTERS TORE Average Latency	Average latency when a node executes the SINTERSTOR E command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis475_ sunionst ore_qps	SUNION STORE QPS	QPS when a node executes the SUNIONSTO RE command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minut e

Metric ID	Metric Name	Description	Valu e Rang e	Un it	Nu mb er Sys te m	Monitored Object	Moni torin g Perio d (Raw Data )
redis473_ sunionst ore_p99	SUNION STORE p99 Latency	p99 latency when a node executes the SUNIONSTO RE command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e
redis471_ sunionst ore_avg_ usec	SUNION STORE Average Latency	Average latency when a node executes the SUNIONSTO RE command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minut e

#### Zset Command Metrics

	Table 4-59	Zset	command	metrics
--	------------	------	---------	---------

Metric ID	Metric Name	Description	Valu e Rang e	Un it	Nu mb er Sys te m	Monitored Object	Moni torin g Perio d (Raw Data )
redis315_ zremrang ebylex_q ps	ZREMRA NGEBYLE X QPS	QPS when a node executes the ZREMRANGE BYLEX command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minu te
redis314_ zremrang ebylex_p 99	ZREMRA NGEBYLE X p99 Latency	p99 latency when a node executes the ZREMRANGE BYLEX command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te

Metric ID	Metric Name	Description	Valu e Rang e	Un it	Nu mb er Sys te m	Monitored Object	Moni torin g Perio d (Raw Data )
redis312_ zremrang ebylex_a vg_usec	ZREMRA NGEBYLE X Average Latency	Average latency when a node executes the ZREMRANGE BYLEX command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis311_ zremrang ebyscore _qps	ZREMRA NGEBYSC ORE QPS	QPS when a node executes the ZREMRANGE BYSCORE command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minu te
redis310_ zremrang ebyscore _p99	ZREMRA NGEBYSC ORE p99 Latency	p99 latency when a node executes the ZREMRANGE BYSCORE command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis308_ zremrang ebyscore _avg_use c	ZREMRA NGEBYSC ORE Average Latency	Average latency when a node executes the ZREMRANGE BYSCORE command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis307_ zremrang ebyrank_ qps	ZREMRA NGEBYR ANK QPS	QPS when a node executes the ZREMRANGE BYRANK command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minu te
redis306_ zremrang ebyrank_ p99	ZREMRA NGEBYR ANK p99 Latency	p99 latency when a node executes the ZREMRANGE BYRANK command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te

Metric ID	Metric Name	Description	Valu e Rang e	Un it	Nu mb er Sys te m	Monitored Object	Moni torin g Perio d (Raw Data )
redis304_ zremrang ebyrank_ avg_usec	ZREMRA NGEBYR ANK Average Latency	Average latency when a node executes the ZREMRANGE BYRANK command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis303_ zpopmin _qps	ZPOPMI N QPS	QPS when a node executes the ZPOPMIN command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minu te
redis302_ zpopmin _p99	ZPOPMI N p99 Latency	p99 latency when a node executes the ZPOPMIN command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis300_ zpopmin _avg_use c	ZPOPMI N Average Latency	Average latency when a node executes the ZPOPMIN command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis299_ zpopmax _qps	ZPOPMA X QPS	QPS when a node executes the ZPOPMAX command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minu te
redis298_ zpopmax _p99	ZPOPMA X p99 Latency	p99 latency when a node executes the ZPOPMAX command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis296_ zpopmax _avg_use c	ZPOPMA X Average Latency	Average latency when a node executes the ZPOPMAX command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te

Metric ID	Metric Name	Description	Valu e Rang e	Un it	Nu mb er Sys te m	Monitored Object	Moni torin g Perio d (Raw Data )
redis295_ zlexcount _qps	ZLEXCOU NT QPS	QPS when a node executes the ZLEXCOUNT command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minu te
redis294_ zlexcount _p99	ZLEXCOU NT p99 Latency	p99 latency when a node executes the ZLEXCOUNT command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis292_ zlexcount _avg_use c	ZLEXCOU NT Average Latency	Average latency when a node executes the ZLEXCOUNT command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis291_ zrevrank_ qps	ZREVRA NK QPS	QPS when a node executes the ZREVRANK command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minu te
redis290_ zrevrank_ p99	ZREVRA NK p99 Latency	p99 latency when a node executes the ZREVRANK command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis288_ zrevrank_ avg_usec	ZREVRA NK Average Latency	Average latency when a node executes the ZREVRANK command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis287_ zrank_qp s	ZRANK QPS	QPS when a node executes the ZRANK command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minu te

Metric ID	Metric Name	Description	Valu e Rang e	Un it	Nu mb er Sys te m	Monitored Object	Moni torin g Perio d (Raw Data )
redis286_ zrank_p9 9	ZRANK p99 Latency	p99 latency when a node executes the ZRANK command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis284_ zrank_av g_usec	ZRANK Average Latency	Average latency when a node executes the ZRANK command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis283_ zscore_q ps	ZSCORE QPS	QPS when a node executes the ZSCORE command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minu te
redis282_ zscore_p 99	ZSCORE p99 Latency	p99 latency when a node executes the ZSCORE command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis280_ zscore_av g_usec	ZSCORE Average Latency	Average latency when a node executes the ZSCORE command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis279_ zrem_qps	ZREM QPS	QPS when a node executes the ZREM command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minu te
redis278_ zrem_p9 9	ZREM p99 Latency	p99 latency when a node executes the ZREM command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te

Metric ID	Metric Name	Description	Valu e Rang e	Un it	Nu mb er Sys te m	Monitored Object	Moni torin g Perio d (Raw Data )
redis276_ zrem_avg _usec	ZREM Average Latency	Average latency when a node executes the ZREM command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis275_ zcount_q ps	ZCOUNT QPS	QPS when a node executes the ZCOUNT command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minu te
redis274_ zcount_p 99	ZCOUNT p99 Latency	p99 latency when a node executes the ZCOUNT command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis272_ zcount_a vg_usec	ZCOUNT Average Latency	Average latency when a node executes the ZCOUNT command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis271_ zrange_q ps	ZRANGE QPS	QPS when a node executes the ZRANGE command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minu te
redis270_ zrange_p 99	ZRANGE p99 Latency	p99 latency when a node executes the ZRANGE command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis268_ zrange_a vg_usec	ZRANGE Average Latency	Average latency when a node executes the ZRANGE command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te

Metric ID	Metric Name	Description	Valu e Rang e	Un it	Nu mb er Sys te m	Monitored Object	Moni torin g Perio d (Raw Data )
redis267_ zrevrang e_qps	ZREVRA NGE QPS	QPS when a node executes the ZREVRANGE command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minu te
redis266_ zrevrang e_p99	ZREVRA NGE p99 Latency	p99 latency when a node executes the ZREVRANGE command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis264_ zrevrang e_avg_us ec	ZREVRA NGE Average Latency	Average latency when a node executes the ZREVRANGE command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis263_ zincrby_q ps	ZINCRBY QPS	QPS when a node executes the ZINCRBY command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minu te
redis262_ zincrby_p 99	ZINCRBY p99 Latency	p99 latency when a node executes the ZINCRBY command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis260_ zincrby_a vg_usec	ZINCRBY Average Latency	Average latency when a node executes the ZINCRBY command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis259_ zscan_qp s	ZSCAN QPS	QPS when a node executes the ZSCAN command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minu te

Metric ID	Metric Name	Description	Valu e Rang e	Un it	Nu mb er Sys te m	Monitored Object	Moni torin g Perio d (Raw Data )
redis258_ zscan_p9 9	ZSCAN p99 Latency	p99 latency when a node executes the ZSCAN command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis256_ zscan_av g_usec	ZSCAN Average Latency	Average latency when a node executes the ZSCAN command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis255_ zcard_qp s	ZCARD QPS	QPS when a node executes the ZCARD command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minu te
redis254_ zcard_p9 9	ZCARD p99 Latency	p99 latency when a node executes the ZCARD command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis252_ zcard_av g_usec	ZCARD Average Latency	Average latency when a node executes the ZCARD command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis251_ zadd_qps	ZADD QPS	QPS when a node executes the ZADD command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minu te
redis250_ zadd_p99	ZADD p99 Latency	p99 latency when a node executes the ZADD command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te

Metric ID	Metric Name	Description	Valu e Rang e	Un it	Nu mb er Sys te m	Monitored Object	Moni torin g Perio d (Raw Data )
redis248_ zadd_avg _usec	ZADD Average Latency	Average latency when a node executes the ZADD command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis495_ zinterstor e_qps	ZINTERS TORE QPS	QPS when a node executes the ZINTERSTOR E command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minu te
redis493_ zinterstor e_p99	ZINTERS TORE p99 Latency	p99 latency when a node executes the ZINTERSTOR E command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis491_ zinterstor e_avg_us ec	ZINTERS TORE Average Latency	Average latency when a node executes the ZINTERSTOR E command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis490_ zunionst ore_qps	ZUNION STORE QPS	QPS when a node executes the ZUNIONSTO RE command	≥ 0	Co unt s/s	N/A	GeminiDB Redis instance nodes	1 minu te
redis488_ zunionst ore_p99	ZUNION STORE p99 Latency	p99 latency when a node executes the ZUNIONSTO RE command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te
redis486_ zunionst ore_avg_ usec	ZUNION STORE Average Latency	Average latency when a node executes the ZUNIONSTO RE command	≥ 0	μs	N/A	GeminiDB Redis instance nodes	1 minu te

# **Bitmap Command Metrics**

Table 4-60	Bitmap con	nmand metrics
------------	------------	---------------

Metric ID	Metric Name	Description	Value Range	U ni t	N u be r Sy st e m	Monitor ed Object	Moni torin g Perio d (Raw Data )
redis440_ bitfield_q ps	BITFIELD QPS	QPS when a node executes the BITFIELD command	≥ 0	Co un ts/ s	N/ A	GeminiD B Redis instance nodes	1 minu te
redis438_ bitfield_p 99	BITFIELD p99 Latency	p99 latency when a node executes the BITFIELD command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 minu te
redis436_ bitfield_a vg_usec	BITFIELD Average Latency	Average latency when a node executes the BITFIELD command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 minu te
redis435_ bitop_qp s	BITOP QPS	QPS when a node executes the BITOP command	≥ 0	Co un ts/ s	N/ A	GeminiD B Redis instance nodes	1 minu te
redis433_ bitop_p9 9	BITOP p99 Latency	p99 latency when a node executes the BITOP command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 minu te
redis431_ bitop_av g_usec	BITOP Average Latency	Average latency when a node executes the BITOP command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 minu te
redis430_ bitpos_q ps	BITPOS QPS	QPS when a node executes the BITPOS command	≥ 0	Co un ts/ s	N/ A	GeminiD B Redis instance nodes	1 minu te

Metric ID	Metric Name	Description	Value Range	U ni t	N u be r Sy st e m	Monitor ed Object	Moni torin g Perio d (Raw Data )
redis428_ bitpos_p9 9	BITPOS p99 Latency	p99 latency when a node executes the BITPOS command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 minu te
redis426_ bitpos_av g_usec	BITPOS Average Latency	Average latency when a node executes the BITPOS command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 minu te
redis425_ bitcount_ qps	BITCOUNT QPS	QPS when a node executes the BITCOUNT command	≥ 0	Co un ts/ s	N/ A	GeminiD B Redis instance nodes	1 minu te
redis423_ bitcount_ p99	BITCOUNT p99 Latency	p99 latency when a node executes the BITCOUNT command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 minu te
redis421_ bitcount_ avg_usec	BITCOUNT Average Latency	Average latency when a node executes the BITCOUNT command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 minu te
redis420_ getbit_qp s	GETBIT QPS	QPS when a node executes the GETBIT command	≥ 0	Co un ts/ s	N/ A	GeminiD B Redis instance nodes	1 minu te
redis418_ getbit_p9 9	GETBIT p99 Latency	p99 latency when a node executes the GETBIT command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 minu te
redis416_ getbit_av g_usec	GETBIT Average Latency	Average latency when a node executes the GETBIT command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 minu te

Metric ID	Metric Name	Description	Value Range	U ni t	N u be r Sy st e m	Monitor ed Object	Moni torin g Perio d (Raw Data )
redis415_ setbit_qp s	SETBIT QPS	QPS when a node executes the SETBIT command	≥ 0	Co un ts/ s	N/ A	GeminiD B Redis instance nodes	1 minu te
redis413_ setbit_p9 9	SETBIT p99 Latency	p99 latency when a node executes the SETBIT command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 minu te
redis411_ setbit_av g_usec	SETBIT Average Latency	Average latency when a node executes the SETBIT command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 minu te

## Stream Command Metrics

Table 4-61 Stream command m	metrics
-----------------------------	---------

Metric ID	Metric Name	Description	Value Rang e	U nit	N u be r Sy st e m	Monitor ed Object	Mon itori ng Peri od (Ra W Data )
redis590_x readgroup _qps	XREADG ROUP QPS	QPS when a node executes the XREADGROUP command	≥ 0	Co un ts/ s	N/ A	GeminiD B Redis instance nodes	1 min ute
redis588_x readgroup _p99	XREADG ROUP p99 Latency	p99 latency when a node executes the XREADGROUP command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute

Metric ID	Metric Name	Description	Value Rang e	U nit	N u be r Sy st e m	Monitor ed Object	Mon itori ng Peri od (Ra W Data )
redis586_x readgroup _avg_usec	XREADG ROUP Average Latency	Average latency when a node executes the XREADGROUP command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute
redis585_x read_qps	XREAD QPS	QPS when a node executes the XREAD command	≥ 0	Co un ts/ s	N/ A	GeminiD B Redis instance nodes	1 min ute
redis583_x read_p99	XREAD p99 Latency	p99 latency when a node executes the XREAD command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute
redis581_x read_avg_ usec	XREAD Average Latency	Average latency when a node executes the XREAD command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute
redis580_x info_qps	XINFO QPS	QPS when a node executes the XINFO command	≥ 0	Co un ts/ s	N/ A	GeminiD B Redis instance nodes	1 min ute
redis578_x info_p99	XINFO p99 Latency	p99 latency when a node executes the XINFO command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute
redis576_x info_avg_u sec	XINFO Average Latency	Average latency when a node executes the XINFO command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute
redis575_x pending_q ps	XPENDI NG QPS	QPS when a node executes the XPENDING command	≥ 0	Co un ts/ s	N/ A	GeminiD B Redis instance nodes	1 min ute
redis573_x pending_p 99	XPENDI NG p99 Latency	p99 latency when a node executes the XPENDING command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute

Metric ID	Metric Name	Description	Value Rang e	U nit	N u be r Sy st e m	Monitor ed Object	Mon itori ng Peri od (Ra W Data )
redis571_x pending_a vg_usec	XPENDI NG Average Latency	Average latency when a node executes the XPENDING command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute
redis570_x claim_qps	XCLAIM QPS	QPS when a node executes the XCLAIM command	≥ 0	Co un ts/ s	N/ A	GeminiD B Redis instance nodes	1 min ute
redis568_x claim_p99	XCLAIM p99 Latency	p99 latency when a node executes the XCLAIM command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute
redis566_x claim_avg _usec	XCLAIM Average Latency	Average latency when a node executes the XCLAIM command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute
redis565_x revrange_ qps	XREVRA NGE QPS	QPS when a node executes the XREVRANGE command	≥ 0	Co un ts/ s	N/ A	GeminiD B Redis instance nodes	1 min ute
redis563_x revrange_ p99	XREVRA NGE p99 Latency	p99 latency when a node executes the XREVRANGE command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute
redis561_x revrange_ avg_usec	XREVRA NGE Average Latency	Average latency when a node executes the XREVRANGE command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute
redis560_x range_qps	XRANG E QPS	QPS when a node executes the XRANGE command	≥ 0	Co un ts/ s	N/ A	GeminiD B Redis instance nodes	1 min ute

Metric ID	Metric Name	Description	Value Rang e	U nit	N u be r Sy st e m	Monitor ed Object	Mon itori ng Peri od (Ra W Data )
redis558_x range_p99	XRANG E p99 Latency	p99 latency when a node executes the XRANGE command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute
redis556_x range_avg _usec	XRANG E Average Latency	Average latency when a node executes the XRANGE command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute
redis555_x len_qps	XLEN QPS	QPS when a node executes the XLEN command	≥ 0	Co un ts/ s	N/ A	GeminiD B Redis instance nodes	1 min ute
redis553_x len_p99	XLEN p99 Latency	p99 latency when a node executes the XLEN command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute
redis551_x len_avg_u sec	XLEN Average Latency	Average latency when a node executes the XLEN command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute
redis550_x trim_qps	XTRIM QPS	QPS when a node executes the XTRIM command	≥ 0	Co un ts/ s	N/ A	GeminiD B Redis instance nodes	1 min ute
redis548_x trim_p99	XTRIM p99 Latency	p99 latency when a node executes the XTRIM command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute
redis546_x trim_avg_ usec	XTRIM Average Latency	Average latency when a node executes the XTRIM command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute
redis545_x del_qps	XDEL QPS	QPS when a node executes the XDEL command	≥ 0	Co un ts/ s	N/ A	GeminiD B Redis instance nodes	1 min ute

Metric ID	Metric Name	Description	Value Rang e	U nit	N u be r Sy st e m	Monitor ed Object	Mon itori ng Peri od (Ra W Data )
redis543_x del_p99	XDEL p99 Latency	p99 latency when a node executes the XDEL command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute
redis541_x del_avg_u sec	XDEL Average Latency	Average latency when a node executes the XDEL command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute
redis540_x group_qps	XGROU P QPS	QPS when a node executes the XGROUP command	≥ 0	Co un ts/ s	N/ A	GeminiD B Redis instance nodes	1 min ute
redis538_x group_p99	XGROU P p99 Latency	p99 latency when a node executes the XGROUP command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute
redis536_x group_avg _usec	XGROU P Average Latency	Average latency when a node executes the XGROUP command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute
redis535_x ack_qps	XACK QPS	QPS when a node executes the XACK command	≥ 0	Co un ts/ s	N/ A	GeminiD B Redis instance nodes	1 min ute
redis533_x ack_p99	XACK p99 Latency	p99 latency when a node executes the XACK command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute
redis531_x ack_avg_u sec	XACK Average Latency	Average latency when a node executes the XACK command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute
redis530_x add_qps	XADD QPS	QPS when a node executes the XADD command	≥ 0	Co un ts/ s	N/ A	GeminiD B Redis instance nodes	1 min ute

Metric ID	Metric Name	Description	Value Rang e	U nit	N u be r Sy st e m	Monitor ed Object	Mon itori ng Peri od (Ra W Data )
redis528_x add_p99	XADD p99 Latency	p99 latency when a node executes the XADD command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute
redis526_x add_avg_u sec	XADD Average Latency	Average latency when a node executes the XADD command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute

### Geo Command Metrics

 Table 4-62 Geo command metrics

Metric ID	Metric Name	Description	Val ue Ran ge	U ni t	N u be r Sy st e m	Monitore d Object	Mon itori ng Peri od (Ra W Data )
redis525_ geopos_q ps	GEOPOS QPS	QPS when a node executes the GEOPOS command	≥ 0	C o u nt s/ s	N/ A	GeminiD B Redis instance nodes	1 minu te
redis523_ geopos_p 99	GEOPOS p99 Latency	p99 latency when a node executes the GEOPOS command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 minu te

Metric ID	Metric Name	Description	Val ue Ran ge	U ni t	N u be r Sy st e m	Monitore d Object	Mon itori ng Peri od (Ra W Data )
redis521_ geopos_a vg_usec	GEOPOS Average Latency	Average latency when a node executes the GEOPOS command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 minu te
redis520_ geodist_q ps	GEODIST QPS	QPS when a node executes the GEODIST command	≥ 0	C o u nt s/ s	N/ A	GeminiD B Redis instance nodes	1 minu te
redis518_ geodist_p 99	GEODIST p99 Latency	p99 latency when a node executes the GEODIST command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 minu te
redis516_ geodist_a vg_usec	GEODIST Average Latency	Average latency when a node executes the GEODIST command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 minu te
redis515_ geohash_ qps	GEOHASH QPS	QPS when a node executes the GEOHASH command	≥ 0	C o u nt s/ s	N/ A	GeminiD B Redis instance nodes	1 minu te
redis513_ geohash_ p99	GEOHASH p99 Latency	p99 latency when a node executes the GEOHASH command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 minu te
redis511_ geohash_ avg_usec	GEOHASH Average Latency	Average latency when a node executes the GEOHASH command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 minu te

Metric ID	Metric Name	Description	Val ue Ran ge	U ni t	N u be r Sy st e m	Monitore d Object	Mon itori ng Peri od (Ra w Data )
redis510_ georadiu s_qps	GEORADIUS QPS	QPS when a node executes the GEORADIUS command	≥ 0	C o u nt s/ s	N/ A	GeminiD B Redis instance nodes	1 minu te
redis508_ georadiu s_p99	GEORADIUS p99 Latency	p99 latency when a node executes the GEORADIUS command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 minu te
redis506_ georadiu s_avg_us ec	GEORADIUS Average Latency	Average latency when a node executes the GEORADIUS command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 minu te
redis505_ georadiu sbymem ber_qps	GEORADIUS BYMEMBER QPS	QPS when a node executes the GEORADIUSBYM EMBER command	≥ 0	C o u nt s/ s	N/ A	GeminiD B Redis instance nodes	1 minu te
redis503_ georadiu sbymem ber_p99	GEORADIUS BYMEMBER p99 Latency	p99 latency when a node executes the GEORADIUSBYM EMBER command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 minu te
redis501_ georadiu sbymem ber_avg_ usec	GEORADIUS BYMEMBER Average Latency	Average latency when a node executes the GEORADIUSBYM EMBER command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 minu te

Metric ID	Metric Name	Description	Val ue Ran ge	U ni t	N u be r Sy st e m	Monitore d Object	Mon itori ng Peri od (Ra W Data )
redis500_ geoadd_ qps	GEOADD QPS	QPS when a node executes the GEOADD command	≥ 0	C o u nt s/ s	N/ A	GeminiD B Redis instance nodes	1 minu te
redis498_ geoadd_ p99	GEOADD p99 Latency	p99 latency when a node executes the GEOADD command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 minu te
redis496_ geoadd_ avg_usec	GEOADD Average Latency	Average latency when a node executes the GEOADD command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 minu te

# Hyperloglog Command Metrics

 Table 4-63 Hyperloglog command metrics

Metric ID	Metric Name	Description	Val ue Ran ge	U ni t	Nu m be r Sys te m	Monitor ed Object	Moni torin g Perio d (Raw Data )
redis455_ pfmerge_ qps	PFMERGE QPS	QPS when a node executes the PFMERGE command	≥ 0	C ou nt s/ s	N/ A	GeminiD B Redis instance nodes	1 minu te

Metric ID	Metric Name	Description	Val ue Ran ge	U ni t	Nu m be r Sys te m	Monitor ed Object	Moni torin g Perio d (Raw Data )
redis453_ pfmerge_ p99	PFMERGE p99 Latency	p99 latency when a node executes the PFMERGE command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 minu te
redis451_ pfmerge_ avg_usec	PFMERGE Average Latency	Average latency when a node executes the PFMERGE command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 minu te
redis450_ pfcount_ qps	PFCOUNT QPS	QPS when a node executes the PFCOUNT command	≥ 0	C ou nt s/ s	N/ A	GeminiD B Redis instance nodes	1 minu te
redis448_ pfcount_ p99	PFCOUNT p99 Latency	p99 latency when a node executes the PFCOUNT command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 minu te
redis446_ pfcount_ avg_usec	PFCOUNT Average Latency	Average latency when a node executes the PFCOUNT command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 minu te
redis445_ pfadd_qp s	PFADD QPS	QPS when a node executes the PFADD command	≥ 0	C ou nt s/ s	N/ A	GeminiD B Redis instance nodes	1 minu te
redis443_ pfadd_p9 9	PFADD p99 Latency	p99 latency when a node executes the PFADD command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 minu te
redis441_ pfadd_av g_usec	PFADD Average Latency	Average latency when a node executes the PFADD command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 minu te
# Pub/Sub Command Metrics

<b>Table 4-64</b>	Pub/Sub	command	metrics
-------------------	---------	---------	---------

Metric ID	Metric Name	Description	Valu e Ran ge	Un it	N u b er Sy st e m	Monitor ed Object	Mon itori ng Peri od (Ra W Data )
redis620_p ubsub_qps	PUBSUB QPS	QPS when a node executes the PUBSUB command	≥ 0	Co un ts/ s	N /A	GeminiD B Redis instance nodes	1 min ute
redis618_p ubsub_p99	PUBSUB p99 Latency	p99 latency when a node executes the PUBSUB command	≥ 0	μs	N /A	GeminiD B Redis instance nodes	1 min ute
redis616_p ubsub_avg _usec	PUBSUB Average Latency	Average latency when a node executes the PUBSUB command	≥ 0	μs	N /A	GeminiD B Redis instance nodes	1 min ute
redis615_p unsubscrib e_qps	PUNSUB SCRIBE QPS	QPS when a node executes the PUNSUBSCRIBE command	≥ 0	Co un ts/ s	N /A	GeminiD B Redis instance nodes	1 min ute
redis613_p unsubscrib e_p99	PUNSUB SCRIBE p99 Latency	p99 latency when a node executes the PUNSUBSCRIBE command	≥ 0	μs	N /A	GeminiD B Redis instance nodes	1 min ute
redis611_p unsubscrib e_avg_usec	PUNSUB SCRIBE Average Latency	Average latency when a node executes the PUNSUBSCRIBE command	≥ 0	μs	N /A	GeminiD B Redis instance nodes	1 min ute
redis610_p subscribe_ qps	PSUBSCR IBE QPS	QPS when a node executes the PSUBSCRIBE command	≥ 0	Co un ts/ s	N /A	GeminiD B Redis instance nodes	1 min ute
redis608_p subscribe_ p99	PSUBSCR IBE p99 Latency	p99 latency when a node executes the PSUBSCRIBE command	≥ 0	μs	N /A	GeminiD B Redis instance nodes	1 min ute

Metric ID	Metric Name	Description	Valu e Ran ge	Un it	N u b er Sy st e m	Monitor ed Object	Mon itori ng Peri od (Ra W Data )
redis606_p subscribe_ avg_usec	PSUBSCR IBE Average Latency	Average latency when a node executes the PSUBSCRIBE command	≥ 0	μs	N /A	GeminiD B Redis instance nodes	1 min ute
redis605_u nsubscribe _qps	UNSUBS CRIBE QPS	QPS when a node executes the UNSUBSCRIBE command	≥ 0	Co un ts/ s	N /A	GeminiD B Redis instance nodes	1 min ute
redis603_u nsubscribe _p99	UNSUBS CRIBE p99 Latency	p99 latency when a node executes the UNSUBSCRIBE command	≥ 0	μs	N /A	GeminiD B Redis instance nodes	1 min ute
redis601_u nsubscribe _avg_usec	UNSUBS CRIBE Average Latency	Average latency when a node executes the UNSUBSCRIBE command	≥ 0	μs	N /A	GeminiD B Redis instance nodes	1 min ute
redis600_s ubscribe_q ps	SUBSCRI BE QPS	QPS when a node executes the SUBSCRIBE command	≥ 0	Co un ts/ s	N /A	GeminiD B Redis instance nodes	1 min ute
redis598_s ubscribe_p 99	SUBSCRI BE p99 Latency	p99 latency when a node executes the SUBSCRIBE command	≥ 0	μs	N /A	GeminiD B Redis instance nodes	1 min ute
redis596_s ubscribe_a vg_usec	SUBSCRI BE Average Latency	Average latency when a node executes the SUBSCRIBE command	≥ 0	μs	N /A	GeminiD B Redis instance nodes	1 min ute
redis595_p ublish_qps	PUBLISH QPS	QPS when a node executes the PUBLISH command	≥ 0	Co un ts/ s	N /A	GeminiD B Redis instance nodes	1 min ute

Metric ID	Metric Name	Description	Valu e Ran ge	Un it	N u b er Sy st e m	Monitor ed Object	Mon itori ng Peri od (Ra W Data )
redis593_p ublish_p99	PUBLISH p99 Latency	p99 latency when a node executes the PUBLISH command	≥ 0	μs	N /A	GeminiD B Redis instance nodes	1 min ute
redis591_p ublish_avg _usec	PUBLISH Average Latency	Average latency when a node executes the PUBLISH command	≥ 0	μs	N /A	GeminiD B Redis instance nodes	1 min ute

# Scripting Command Metrics

Table 4-65	Scripting	command	metrics

Metric ID	Metric Name	Description	Valu e Ran ge	U nit	N u be r Sy st e m	Monitor ed Object	Mon itori ng Peri od (Ra W Data )
redis635_ script_qp s	SCRIPT QPS	QPS when a node executes the SCRIPT command	≥ 0	Co un ts/ s	N/ A	Gemini DB Redis instance nodes	1 minu te
redis633_ script_p9 9	SCRIPT p99 Latency	p99 latency when a node executes the SCRIPT command	≥ 0	μs	N/ A	Gemini DB Redis instance nodes	1 minu te

Metric ID	Metric Name	Description	Valu e Ran ge	U nit	N u be r Sy st e m	Monitor ed Object	Mon itori ng Peri od (Ra W Data )
redis631_ script_av g_usec	SCRIPT Average Latency	Average latency when a node executes the SCRIPT command	≥ 0	μs	N/ A	Gemini DB Redis instance nodes	1 minu te
redis630_ evalsha_ qps	EVALSHA QPS	QPS when a node executes the EVALSHA command	≥ 0	Co un ts/ s	N/ A	Gemini DB Redis instance nodes	1 minu te
redis628_ evalsha_ p99	EVALSHA p99 Latency	p99 latency when a node executes the EVALSHA command	≥ 0	μs	N/ A	Gemini DB Redis instance nodes	1 minu te
redis626_ evalsha_ avg_usec	EVALSHA Average Latency	Average latency when a node executes the EVALSHA command	≥ 0	μs	N/ A	Gemini DB Redis instance nodes	1 minu te
redis625_ eval_qps	EVAL QPS	QPS when a node executes the EVAL command	≥ 0	Co un ts/ s	N/ A	Gemini DB Redis instance nodes	1 minu te
redis623_ eval_p99	EVAL p99 Latency	p99 latency when a node executes the EVAL command	≥ 0	μs	N/ A	Gemini DB Redis instance nodes	1 minu te
redis621_ eval_avg _usec	EVAL Average Latency	Average latency when a node executes the EVAL command	≥ 0	μs	N/ A	Gemini DB Redis instance nodes	1 minu te

# **Transactions Command Metrics**

Table 4-66	Transactions	command	metrics
------------	--------------	---------	---------

Metric ID	Metric Name	Description	Val ue Ran ge	Un it	N u b er Sy st e m	Monitor ed Object	Mon itori ng Peri od (Ra w Dat a)
redis660_ discard_q ps	DISCARD QPS	QPS when a node executes the DISCARD command	≥ 0	Co unt s/s	N /A	GeminiD B Redis instance nodes	1 min ute
redis658_ discard_p 99	DISCARD p99 Latency	p99 latency when a node executes the DISCARD command	≥ 0	μs	N /A	GeminiD B Redis instance nodes	1 min ute
redis656_ discard_a vg_usec	DISCARD Average Latency	Average latency when a node executes the DISCARD command	≥ 0	μs	N /A	GeminiD B Redis instance nodes	1 min ute
redis655_ exec_qps	exec QPS	QPS when a node executes the EXEC command	≥ 0	Co unt s/s	N /A	GeminiD B Redis instance nodes	1 min ute
redis653_ exec_p99	EXEC p99 Latency	p99 latency when a node executes the EXEC command	≥ 0	μs	N /A	GeminiD B Redis instance nodes	1 min ute
redis651_ exec_avg _usec	EXEC Average Latency	Average latency when a node executes the EXEC command	≥ 0	μs	N /A	GeminiD B Redis instance nodes	1 min ute
redis650_ multi_qp s	MULTI QPS	QPS when a node executes the MULTI command	≥ 0	Co unt s/s	N /A	GeminiD B Redis instance nodes	1 min ute
redis648_ multi_p9 9	MULTI p99 Latency	p99 latency when a node executes the MULTI command	≥ 0	μs	N /A	GeminiD B Redis instance nodes	1 min ute

Metric ID	Metric Name	Description	Val ue Ran ge	Un it	N u b er Sy st e m	Monitor ed Object	Mon itori ng Peri od (Ra W Dat a)
redis646_ multi_av g_usec	MULTI Average Latency	Average latency when a node executes the MULTI command	≥ 0	μs	N /A	GeminiD B Redis instance nodes	1 min ute
redis645_ unwatch _qps	UNWATC H QPS	QPS when a node executes the UNWATCH command	≥ 0	Co unt s/s	N /A	GeminiD B Redis instance nodes	1 min ute
redis643_ unwatch _p99	UNWATC H p99 Latency	p99 latency when a node executes the UNWATCH command	≥ 0	μs	N /A	GeminiD B Redis instance nodes	1 min ute
redis641_ unwatch _avg_use c	UNWATC H Average Latency	Average latency when a node executes the UNWATCH command	≥ 0	μs	N /A	GeminiD B Redis instance nodes	1 min ute
redis640_ watch_qp s	WATCH QPS	QPS when a node executes the WATCH command	≥ 0	Co unt s/s	N /A	GeminiD B Redis instance nodes	1 min ute
redis638_ watch_p9 9	WATCH p99 Latency	p99 latency when a node executes the WATCH command	≥ 0	μs	N /A	GeminiD B Redis instance nodes	1 min ute
redis636_ watch_av g_usec	WATCH Average Latency	Average latency when a node executes the WATCH command	≥ 0	μs	N /A	GeminiD B Redis instance nodes	1 min ute

# **ExHash Command Metrics**

Table 4-67 ExH	ash command metrics
----------------	---------------------

Metric ID	Metric Name	Description	Valu e Ran ge	Uni t	N u b e r S y st e m	Monitor ed Object	Mo nito rin g Peri od (Ra w Dat a)
redis794 _exhdel_ avg_usec	EXHDEL Average Latency	Average latency when a node executes the EXHDEL command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute
redis796 _exhdel_ p99	EXHDEL p99 Latency	p99 latency when a node executes the EXHDEL command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute
redis797 _exhdel_ qps	EXHDEL QPS	QPS when a node executes the EXHDEL command	≥ 0	Co unt s/s	N / A	GeminiD B Redis instance nodes	1 min ute
redis766 _exhexist s_avg_us ec	EXHEXISTS Average Latency	Average latency when a node executes the EXHEXISTS command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute
redis768 _exhexist s_p99	EXHEXISTS p99 Latency	p99 latency when a node executes the EXHEXISTS command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute
redis769 _exhexist s_qps	EXHEXISTS QPS	QPS when a node executes the EXHEXISTS command	≥ 0	Co unt s/s	N / A	GeminiD B Redis instance nodes	1 min ute
redis722 _exhexpi re_avg_u sec	EXHEXPIRE Average Latency	Average latency when a node executes the EXHEXPIRE command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute

Metric ID	Metric Name	Description	Valu e Ran ge	Uni t	N u b e r S y st e m	Monitor ed Object	Mo nito rin g Peri od (Ra w Dat a)
redis724 _exhexpi re_p99	EXHEXPIRE p99 Latency	p99 latency when a node executes the EXHEXPIRE command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute
redis725 _exhexpi re_qps	EXHEXPIRE QPS	QPS when a node executes the EXHEXPIRE command	≥ 0	Co unt s/s	N / A	GeminiD B Redis instance nodes	1 min ute
redis718 _exhexpi reat_avg _usec	EXHEXPIREA T Average Latency	Average latency when a node executes the EXHEXPIREAT command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute
redis720 _exhexpi reat_p99	EXHEXPIREA T p99 Latency	p99 latency when a node executes the EXHEXPIREAT command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute
redis721 _exhexpi reat_qps	EXHEXPIREA T QPS	QPS when a node executes the EXHEXPIREAT command	≥ 0	Co unt s/s	N / A	GeminiD B Redis instance nodes	1 min ute
redis702 _exhget_ avg_usec	EXHGET Average Latency	Average latency when a node executes the EXHGET command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute
redis704 _exhget_ p99	EXHGET p99 Latency	p99 latency when a node executes the EXHGET command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute
redis705 _exhget_ qps	EXHGET QPS	QPS when a node executes the EXHGET command	≥ 0	Co unt s/s	N / A	GeminiD B Redis instance nodes	1 min ute

Metric ID	Metric Name	Description	Valu e Ran ge	Uni t	N u b e r S y st e m	Monitor ed Object	Mo nito rin g Peri od (Ra w Dat a)
redis782 _exhgeta ll_avg_us ec	EXHGETALL Average Latency	Average latency when a node executes the EXHGETALL command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute
redis784 _exhgeta ll_p99	EXHGETALL p99 Latency	p99 latency when a node executes the EXHGETALL command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute
redis785 _exhgeta ll_qps	EXHGETALL QPS	QPS when a node executes the EXHGETALL command	≥ 0	Co unt s/s	N / A	GeminiD B Redis instance nodes	1 min ute
redis750 _exhget withver_ avg_usec	EXHGETWIT HVER Average Latency	Average latency when a node executes the EXHGETWITHVER command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute
redis752 _exhget withver_ p99	EXHGETWIT HVER p99 Latency	p99 latency when a node executes the EXHGETWITHVER command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute
redis753 _exhget withver_ qps	EXHGETWIT HVER QPS	QPS when a node executes the EXHGETWITHVER command	≥ 0	Co unt s/s	N / A	GeminiD B Redis instance nodes	1 min ute
redis742 _exhincr by_avg_ usec	EXHINCRBY Average Latency	Average latency when a node executes the EXHINCRBY command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute
redis744 _exhincr by_p99	EXHINCRBY p99 Latency	p99 latency when a node executes the EXHINCRBY command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute

Metric ID	Metric Name	Description	Valu e Ran ge	Uni t	N u b e r S y st e m	Monitor ed Object	Mo nito rin g Peri od (Ra w Dat a)
redis745 _exhincr by_qps	exhincrby QPS	QPS when a node executes the EXHINCRBY command	≥ 0	Co unt s/s	N / A	GeminiD B Redis instance nodes	1 min ute
redis746 _exhincr byfloat_ avg_usec	EXHINCRBYF LOAT Average Latency	Average latency when a node executes the EXHINCRBYFLOA T command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute
redis748 _exhincr byfloat_ p99	EXHINCRBYF LOAT p99 Latency	p99 latency when a node executes the EXHINCRBYFLOA T command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute
redis749 _exhincr byfloat_ qps	EXHINCRBYF LOAT QPS	QPS when a node executes the EXHINCRBYFLOA T command	≥ 0	Co unt s/s	N / A	GeminiD B Redis instance nodes	1 min ute
redis774 _exhkeys _avg_use c	EXHKEYS Average Latency	Average latency when a node executes the EXHKEYS command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute
redis776 _exhkeys _p99	EXHKEYS p99 Latency	p99 latency when a node executes the EXHKEYS command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute
redis777 _exhkeys _qps	EXHKEYS QPS	QPS when a node executes the EXHKEYS command	≥ 0	Co unt s/s	N / A	GeminiD B Redis instance nodes	1 min ute
redis762 _exhlen_ avg_usec	EXHLEN Average Latency	Average latency when a node executes the EXHLEN command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute

Metric ID	Metric Name	Description	Valu e Ran ge	Uni t	N u b e r S y st e m	Monitor ed Object	Mo nito rin g Peri od (Ra w Dat a)
redis764 _exhlen_ p99	EXHLEN p99 Latency	p99 latency when a node executes the EXHLEN command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute
redis765 _exhlen_ qps	EXHLEN QPS	QPS when a node executes the EXHLEN command	≥ 0	Co unt s/s	N / A	GeminiD B Redis instance nodes	1 min ute
redis754 _exhmge t_avg_us ec	EXHMGET Average Latency	Average latency when a node executes the EXHMGET command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute
redis756 _exhmge t_p99	EXHMGET p99 Latency	p99 latency when a node executes the EXHMGET command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute
redis757 _exhmge t_qps	EXHMGET QPS	QPS when a node executes the EXHMGET command	≥ 0	Co unt s/s	N / A	GeminiD B Redis instance nodes	1 min ute
redis758 _exhmge twithver _avg_use c	EXHMGETWI THVER Average Latency	Average latency when a node executes the EXHMGETWITHV ER command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute
redis760 _exhmge twithver _p99	EXHMGETWI THVER p99 Latency	p99 latency when a node executes the EXHMGETWITHV ER command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute
redis761 _exhmge twithver _qps	EXHMGETWI THVER QPS	QPS when a node executes the EXHMGETWITHV ER command	≥ 0	Co unt s/s	N / A	GeminiD B Redis instance nodes	1 min ute

Metric ID	Metric Name	Description	Valu e Ran ge	Uni t	N u b e r S y st e m	Monitor ed Object	Mo nito rin g Peri od (Ra w Dat a)
redis706 _exhmse t_avg_us ec	EXHMSET Average Latency	Average latency when a node executes the EXHMSET command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute
redis708 _exhmse t_p99	EXHMSET p99 Latency	p99 latency when a node executes the EXHMSET command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute
redis709 _exhmse t_qps	EXHMSET QPS	QPS when a node executes the EXHMSET command	≥ 0	Co unt s/s	N / A	GeminiD B Redis instance nodes	1 min ute
redis714 _exhpex pire_avg _usec	EXHPEXPIRE Average Latency	Average latency when a node executes the EXHPEXPIRE command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute
redis716 _exhpex pire_p99	EXHPEXPIRE p99 Latency	p99 latency when a node executes the EXHPEXPIRE command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute
redis717 _exhpex pire_qps	EXHPEXPIRE QPS	QPS when a node executes the EXHPEXPIRE command	≥ 0	Co unt s/s	N / A	GeminiD B Redis instance nodes	1 min ute
redis710 _exhpex pireat_av g_usec	EXHPEXPIRE AT Average Latency	Average latency when a node executes the EXHPEXPIREAT command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute
redis712 _exhpex pireat_p 99	EXHPEXPIRE AT p99 Latency	p99 latency when a node executes the EXHPEXPIREAT command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute

Metric ID	Metric Name	Description	Valu e Ran ge	Uni t	N u b e r S y st e m	Monitor ed Object	Mo nito rin g Peri od (Ra w Dat a)
redis713 _exhpex pireat_q ps	EXHPEXPIRE AT QPS	QPS when a node executes the EXHPEXPIREAT command	≥ 0	Co unt s/s	N / A	GeminiD B Redis instance nodes	1 min ute
redis726 _exhpttl_ avg_usec	EXHPTTL Average Latency	Average latency when a node executes the EXHPTTL command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute
redis728 _exhpttl_ p99	EXHPTTL p99 Latency	p99 latency when a node executes the EXHPTTL command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute
redis729 _exhpttl_ qps	EXHPTTL QPS	QPS when a node executes the EXHPTTL command	≥ 0	Co unt s/s	N / A	GeminiD B Redis instance nodes	1 min ute
redis786 _exhscan _avg_use c	EXHSCAN Average Latency	Average latency when a node executes the EXHSCAN command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute
redis788 _exhscan _p99	EXHSCAN p99 Latency	p99 latency when a node executes the EXHSCAN command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute
redis789 _exhscan _qps	EXHSCAN QPS	QPS when a node executes the EXHSCAN command	≥ 0	Co unt s/s	N / A	GeminiD B Redis instance nodes	1 min ute
redis790 _exhscan unorder_ avg_usec	EXHSCANUN ORDER Average Latency	Average latency when a node executes the EXHSCANUNORD ER command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute

Metric ID	Metric Name	Description	Valu e Ran ge	Uni t	N u b e r S y st e m	Monitor ed Object	Mo nito rin g Peri od (Ra w Dat a)
redis792 _exhscan unorder_ p99	EXHSCANUN ORDER p99 Latency	p99 latency when a node executes the EXHSCANUNORD ER command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute
redis793 _exhscan unorder_ qps	EXHSCANUN ORDER QPS	QPS when a node executes the EXHSCANUNORD ER command	≥ 0	Co unt s/s	N / A	GeminiD B Redis instance nodes	1 min ute
redis698 _exhset_ avg_usec	EXHSET Average Latency	Average latency when a node executes the EXHSET command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute
redis700 _exhset_ p99	EXHSET p99 Latency	p99 latency when a node executes the EXHSET command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute
redis701 _exhset_ qps	EXHSET QPS	QPS when a node executes the EXHSET command	≥ 0	Co unt s/s	N / A	GeminiD B Redis instance nodes	1 min ute
redis738 _exhsetv er_avg_u sec	EXHSETVER Average Latency	Average latency when a node executes the EXHSETVER command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute
redis740 _exhsetv er_p99	EXHSETVER p99 Latency	p99 latency when a node executes the EXHSETVER command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute
redis741 _exhsetv er_qps	EXHSETVER QPS	QPS when a node executes the EXHSETVER command	≥ 0	Co unt s/s	N / A	GeminiD B Redis instance nodes	1 min ute

Metric ID	Metric Name	Description	Valu e Ran ge	Uni t	N u b e r S y st e m	Monitor ed Object	Mo nito rin g Peri od (Ra w Dat a)
redis770 _exhstrle n_avg_us ec	EXHSTRLEN Average Latency	Average latency when a node executes the EXHSTRLEN command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute
redis772 _exhstrle n_p99	EXHSTRLEN p99 Latency	p99 latency when a node executes the EXHSTRLEN command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute
redis773 _exhstrle n_qps	EXHSTRLEN QPS	QPS when a node executes the EXHSTRLEN command	≥ 0	Co unt s/s	N / A	GeminiD B Redis instance nodes	1 min ute
redis730 _exhttl_a vg_usec	EXHTTL Average Latency	Average latency when a node executes the EXHTTL command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute
redis732 _exhttl_p 99	EXHTTL p99 Latency	p99 latency when a node executes the EXHTTL command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute
redis733 _exhttl_q ps	EXHTTL Command QPS	QPS when a node executes the EXHTTL command	≥ 0	Co unt s/s	N / A	GeminiD B Redis instance nodes	1 min ute
redis778 _exhvals _avg_use c	EXHVALS Average Latency	Average latency when a node executes the EXHVALS command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute
redis780 _exhvals _p99	EXHVALS p99 Latency	p99 latency when a node executes the EXHVALS command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute

Metric ID	Metric Name	Description	Valu e Ran ge	Uni t	N u b e r S y st e m	Monitor ed Object	Mo nito rin g Peri od (Ra w Dat a)
redis781 _exhvals _qps	EXHVALS QPS	QPS when a node executes the EXHVALS command	≥ 0	Co unt s/s	N / A	GeminiD B Redis instance nodes	1 min ute
redis734 _exhver_ avg_usec	EXHVER Average Latency	Average latency when a node executes the EXHVER command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute
redis736 _exhver_ p99	EXHVER p99 Latency	p99 latency when a node executes the EXHVER command	≥ 0	μs	N / A	GeminiD B Redis instance nodes	1 min ute
redis737 _exhver_ qps	EXHVER QPS	QPS when a node executes the EXHVER command	≥ 0	Co unt s/s	N / A	GeminiD B Redis instance nodes	1 min ute

## **Common Command Metrics**

Metric ID	Metric Name	Description	Value Range	Uni t	N u be r Sy st e m	Monitor ed Object	Mo nito ring Peri od (Ra w Dat a)
redis059_ scan_qps	SCAN QPS	QPS when a node executes the SCAN command	≥ 0	Cou nts/ s	N/ A	GeminiD B Redis instance nodes	1 min ute
redis058_ scan_p99	SCAN p99 Latency	p99 latency when a node executes the SCAN command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute
redis056_ scan_avg _usec	SCAN Average Latency	Average latency when a node executes the SCAN command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute
redis055_ persist_q ps	PERSIST QPS	QPS when a node executes the PERSIST command	≥ 0	Cou nts/ s	N/ A	GeminiD B Redis instance nodes	1 min ute
redis054_ persist_p 99	PERSIST p99 Latency	p99 latency when a node executes the PERSIST command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute
redis052_ persist_a vg_usec	PERSIST Average Latency	Average latency when a node executes the PERSIST command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute
redis051_ ttl_qps	TTL QPS	QPS when a node executes the TTL command	≥ 0	Cou nts/ s	N/ A	GeminiD B Redis instance nodes	1 min ute
redis050_ ttl_p99	TTL p99 Latency	p99 latency when a node executes the TTL command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute

### Table 4-68 Common command metrics

Metric ID	Metric Name	Description	Value Range	Uni t	N u be r Sy st e m	Monitor ed Object	Mo nito ring Peri od (Ra W Dat a)
redis048_ ttl_avg_u sec	TTL Average Latency	Average latency when a node executes the TTL command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute
redis047_ del_qps	DEL QPS	QPS when a node executes the DEL command	≥ 0	Cou nts/ s	N/ A	GeminiD B Redis instance nodes	1 min ute
redis046_ del_p99	DEL p99 Latency	p99 latency when a node executes the DEL command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute
redis044_ del_avg_ usec	DEL Average Latency	Average latency when a node executes the DEL command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute
redis043_ expire_qp s	EXPIRE QPS	QPS when a node executes the EXPIRE command	≥ 0	Cou nts/ s	N/ A	GeminiD B Redis instance nodes	1 min ute
redis042_ expire_p9 9	EXPIRE p99 Latency	p99 latency when a node executes the EXPIRE command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute
redis040_ expire_av g_usec	EXPIRE Average Latency	Average latency when a node executes the EXPIRE command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute
redis039_ exists_qp s	EXISTS QPS	QPS when a node executes the EXISTS command	≥ 0	Cou nts/ s	N/ A	GeminiD B Redis instance nodes	1 min ute

Metric ID	Metric Name	Description	Value Range	Uni t	N u be r Sy st e m	Monitor ed Object	Mo nito ring Peri od (Ra W Dat a)
redis038_ exists_p9 9	EXISTS p99 Latency	p99 latency when a node executes the EXISTS command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute
redis036_ exists_av g_usec	EXISTS Average Latency	Average latency when a node executes the EXISTS command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute
redis410_ randomk ey_qps	RANDOMK EY QPS	QPS when a node executes the RANDOMKEY command	≥ 0	Cou nts/ s	N/ A	GeminiD B Redis instance nodes	1 min ute
redis408_ randomk ey_p99	RANDOMK EY p99 Latency	p99 latency when a node executes the RANDOMKEY command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute
redis406_ randomk ey_avg_u sec	RANDOMK EY Average Latency	Average latency when a node executes the RANDOMKEY command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute
redis405_ type_qps	TYPE QPS	QPS when a node executes the TYPE command	≥ 0	Cou nts/ s	N/ A	GeminiD B Redis instance nodes	1 min ute
redis403_ type_p99	TYPE p99 Latency	p99 latency when a node executes the TYPE command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute
redis401_ type_avg _usec	TYPE Average Latency	Average latency when a node executes the TYPE command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute

Metric ID	Metric Name	Description	Value Range	Uni t	N u be r Sy st e m	Monitor ed Object	Mo nito ring Peri od (Ra W Dat a)
redis400_ info_qps	INFO QPS	QPS when a node executes the INFO command	≥ 0	Cou nts/ s	N/ A	GeminiD B Redis instance nodes	1 min ute
redis398_ info_p99	INFO p99 Latency	p99 latency when a node executes the INFO command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute
redis396_ info_avg_ usec	INFO Average Latency	Average latency when a node executes the INFO command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute
redis395_ ping_qps	PING QPS	QPS when a node executes the PING command	≥ 0	Cou nts/ s	N/ A	GeminiD B Redis instance nodes	1 min ute
redis393_ ping_p99	PING p99 Latency	p99 latency when a node executes the PING command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute
redis391_ ping_avg _usec	PING Average Latency	Average latency when a node executes the PING command	≥ 0	μs	N/ A	GeminiD B Redis instance nodes	1 min ute

## Dimensions

Кеу	Value
redis_cluster_id	Cluster ID of the GeminiDB Redis instance
redis_node_id	Node ID of the GeminiDB Redis instance

# 4.13.2 Configuring Alarm Rules

## Scenarios

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 management 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 4-179 Creating an alarm rule

Cloud Eye	Alarm Rules 💮		© Feedba	ck 🖗 Usage Guide 🕂 Create Alarm Rule
Dashboard -				
Resource Groups				
Alarm Management				
Alarm Rules				
Alarm History	***			100
Alarm Templates				
Cloud Service				
Custom Monitoring				
Event Monitoring	1 Resource	2 Cloud Eye sends	3 Users locate the	- 4 Service
Data Dump	exceptions occur.	notifications.	them.	ensured.
		Country Allower Dudy		

### **Step 5** Set alarm parameters.

1. Configure basic alarm information.

Figure 4-180 Configuring basic information for an alarm rule

* Name	alarm-cag2	
Description		
		đ
		0/256

### **Table 4-69** 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

Parameter	Description	Example Value
Description	(Optional) Alarm rule description.	-

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

<b>Table 4-70</b>	Parameter	description
-------------------	-----------	-------------

Parameter	Description	Example Value
Alarm Type	Alarm type that the alarm rule is created for. The value can be <b>Metric</b> or <b>Event</b> .	Metric
Resource Type	Type of the resource the alarm rule is created for. Select <b>GeminiDB</b> .	-
Dimension	Metric dimension of the alarm rule. Select <b>Redis-Redis Nodes</b> .	-
Monitoring Scope	<ul> <li>Monitoring scope the alarm rule applies to.</li> <li>NOTE <ul> <li>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.</li> <li>If you select Resource groups and any resource in the group meets the alarm policy, an alarm notification will be sent.</li> <li>To specify Specific resources, click Select Specified Resources, select one or more resources, and click OK.</li> </ul> </li> </ul>	Specified resources
Group	This parameter is mandatory when Monitoring Scope is set to Resource groups.	-

3. Configure an alarm policy.

Figure 4-181 Configuring the alarm policy

* Method		Associate template	Use existing template			
* Alarm Poli	5y					
	Metric Name	1	Marm Policy		Alarm Se	verity Operation
	If Storage Spa	ace Us 👻	Raw data 💌 🖂 >=	▼ 80 % 3 times	s (consecutively) • Then Every 10 minutes • Major	• Delete
Ø	If CPU Usage	•	Raw data 💌 🖂 >=	• 80 % 3 times	s (consecutively) v Then Every 10 minutes v	▼ Delete
	If Memory Us	age 🔻	Raw data 💌 >=	• 80 % 3 times	s (consecutively) v Then Every 10 minutes v Major	▼ Delete
⊕ A	dd Alarm Policy You	can add 47 more.				

Parameter	Description	Example Value
Method	Select Associate template, Use existing template, or Configure manually. NOTE If you set Monitoring Scope to Specific resources, you can set Method to Use existing template.	Configure manually
Template	Select the template to be used.	-
	This parameter is available only when you select <b>Use existing template</b> for <b>Method</b> .	
Alarm Policy	Policy for triggering an alarm. You can configure the threshold, consecutive periods, alarm interval, and alarm severity based on service requirements.	Take the CPU usage as an example. The alarm policy
	<ul> <li>Metric Name: specifies the name of the metric configured in the alarm rule.</li> <li>The following metrics are recommended:</li> </ul>	configured in Figure 4-181 indicates that
	Storage Space Usage,	a major alarm
	which is used to monitor the storage usage of GeminiDB Redis instances. If the storage usage is greater than 80%, scale up the storage in a timely manner by referring to 4.6.7.2 Manually Scaling Up Storage Space.	notification will be sent to users every 10 minutes if the original CPU usage
	CPU Usage and Memory Usage,	reaches 80%
	which are used to monitor the compute resource usage of each GeminiDB Redis instance node. If the CPU usage or memory usage is greater than 80%, you can add nodes or upgrade node specifications in a timely manner.	three consecutive periods.
	For more metrics, see <b>4.13.1 Supported</b> Metrics.	
	<ul> <li>Alarm Severity: specifies the severity of the alarm. Valid values are Critical, Major, Minor, and Informational.</li> <li>NOTE</li> </ul>	
	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.	

### Table 4-71 Parameter description

4. Configure alarm notification information.

## Figure 4-182 Configuring alarm notification information

Alarm Notification		
* Notification Recipient	Notification group Topic subscription	on
* Notification Group	Select	C sh to make it available for selection. After you create the notification group, click Add Notification Object in the Operation column of the notification group list to add notification ob
* Notification Window	Daily 00:00 - 23:59 GMT+0	08.00 ()
* Trigger Condition	Generated alarm	

## Table 4-72 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.	Enabled Alarm Notification.
	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.	
Notification Recipient	Select Notification group or Topic subscription.	-
Notification Group	Notification group the alarm notification is to be sent to.	-
Notification Object	<ul> <li>Specifies the object that receives alarm notifications. You can select the account contact or a topic.</li> <li>Account contact is the mobile phone number and email address provided for registration.</li> <li><b>Topic</b> 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.</li> </ul>	-

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

## 5. Configure advanced settings.

### Figure 4-183 Advanced settings

Advanced Settings	Enterprise Project   Tag				
* Enterprise Project	default   C Create Enterprise Project				
	The enterprise project the alarm rule belongs to.				
Tag	It is recommended that you use TMS's predefined tag function to add the same tag to different cloud resources. View predefined tags C				
	To add a tag, enter a tag key and a tag value below.				
	Enter a tag value Add				
	20 tags available for addition.				

## Table 4-73 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 <b>Creating an</b> <b>Enterprise Project</b> .	default

Parameter	Description	Example Value
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 <b>Creating Predefined Tags</b> .	-
	<ul> <li>A key can contain a maximum of 128 characters, and a value can contain a maximum of 255 characters.</li> </ul>	
	- 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 Recommended Alarm Policies

This section describes recommended alarm policies of GeminiDB Redis instances and nodes.

Metric	Dimension	Value Range	Alarm Policy
Total Traffic Sent by the Instance	Instance	≥ 0	Alarm severity: major
			Number of consecutive periods: 3
			Alarm threshold: ≥ 875,000,000 bytes/s
			Alarm period: every 5 minutes

Table 4-74 Alarm	policies	for GeminiDB	<b>Redis instances</b>
------------------	----------	--------------	------------------------

Metric	Dimension	Value Range	Alarm Policy
Total Traffic Received by the Instance	Instance	≥ 0	Alarm severity: major Number of consecutive periods: 3 Alarm threshold: ≥ 875,000,000 bytes/s
			Alarm period: every 5 minutes
Average Latency	Instance	≥ 0	Alarm severity: major Number of consecutive periods: 5 Alarm threshold: ≥ 15,000 us Alarm period:
P99 Latency	Instance	≥ 0	every 15 minutes Alarm severity: major Number of consecutive periods: 5 Alarm threshold: ≥ 30,000 us Alarm period: every 15 minutes
Storage Usage	Instance	0-100%	Alarm severity: major Number of consecutive periods: 3 Alarm threshold: ≥ 70% Alarm period: once a day

Metric	Dimension	Value Range	Alarm Policy
Max. Parameters Sent in a Request	Instance	≥ 0	Alarm severity: major Number of consecutive periods: 1 Alarm threshold: ≥ 10,000 Alarm period: every 15 minutes
Max. Elements Obtained in a Request	Instance	≥ 0	Alarm severity: major Number of consecutive periods: 1 Alarm threshold: ≥ 10,000 Alarm period: every 15 minutes
Max. Bytes Sent in a Request	Instance	≥ 0	Alarm severity: major Number of consecutive periods: 1 Alarm threshold: ≥ 1MiB Alarm period: every 15 minutes
Max. Bytes Obtained in a Request	Instance	≥ 0	Alarm severity: major Number of consecutive periods: 1 Alarm threshold: ≥ 1MiB Alarm period: every 15 minutes

Metric	Dimension	Value Range	Alarm Policy
CPU Usage	Node	0-100%	Alarm severity: major Number of consecutive periods: 3
			Alarm threshold: ≥ 70% Alarm period:
			every 15 minutes
Memory Usage	Node	0-100%	Alarm severity: warning
			Number of consecutive periods: 3
			Alarm threshold: ≥ 70%
			Alarm period: every 15 minutes
Connection Usage	Node	0-100%	Alarm severity: major
			Number of consecutive periods: 3
			Alarm threshold: ≥ 50%
			Alarm period: every 5 minutes
Traffic Receive Speed	Node	≥ 0	Alarm severity: major
			Number of consecutive periods: 3
			Alarm threshold: ≥ 87,500,000 bytes/s
			Alarm period: every 5 minutes

 Table 4-75 Alarm policies for GeminiDB Redis nodes

Metric	Dimension	Value Range	Alarm Policy
Traffic Send Speed	Node	≥ 0	Alarm severity: major
			Number of consecutive periods: 3
			Alarm threshold: ≥ 87,500,000 bytes/s
			Alarm period:

# 4.13.4 Viewing Metrics

## **Scenarios**

Cloud Eye monitors the status of GeminiDB Redis instances. You can view 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.

## Prerequisites

• 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.

## 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 the instance whose metrics you want to view and click its name.
  - Instance metrics: In the upper right corner, click View Metric.

#### Figure 4-184 Viewing instance metrics

<	Available			@ Footback Ware Matric Reset Passward Restart C
Basic Information				
Backups & Restorations	Basic Information			
Node Management	DB Instance Name	D5 Instance ID	Storage Type	Product Type
Slow Query Legs	/ 0	đ	Cloud native	Capacity-oriented
Metrics				
Diagnosis Analysis	Status	Region	AZ	D6 Instance Type
Taga	Available		co-oxfo-la	Proxy cluster
	Entreprise Project	Maintenance Window (3)	Compatible API	
	detault	18.00 - 14.00 Change	Rods 5.2 (Fully compatible with 5.2 and earlier versions, such as 5.0,4.0, and 2.1	0

• Node metrics: In the navigation pane, choose **Node Management**. In the **Node Information** area, browse to the target node and click **View Metric** in the **Operation** column.

### Figure 4-185 Viewing node metrics

<i>i</i> ode Information							
Stop Sturt Add Node Change							
Q. Select one or more filters from the pop-up lists.	If you enter a keyword without a filter applied, the	system will search for all names matching this k	eyword.				
Name/ID	Status	AZ	Private IP Address	EIP	Operation		
	<ul> <li>Available</li> </ul>	824			View Metric Unbind EIP More ~		
	Available	az2		Unbound	View Metric Bind EIP More ~		

**Step 4** In the monitoring area, you can select a duration to view the monitoring data.

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

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

----End

### Method 2

- Step 1 Log in to the Huawei Cloud console.
- Step 2 In the service list, choose Databases > GeminiDB.
- **Step 3** Log in to the GeminiDB console.
- **Step 4** On the **Instance** page, click the instance whose metrics you want to view and click its name.
- **Step 5** In the navigation pane, choose **Metrics**.
- **Step 6** On the **Metrics** page, view real-time monitoring data.
  - Click the **DB Instance** tab to view real-time monitoring data, such as the instance QPS, average hit ratio, and connection usage.
  - Click the **Node-level Metrics** tab to view real-time monitoring data, such as CPU, memory, and connection usage.
  - The following monitoring time windows are supported: last 1 hour, last 3 hours, last 12 hours, last 24 hours, last 7 days, and a custom time period.
  - You can also enable auto refresh (every 60s).
  - The monitoring period can be 1 minute or 5 minutes.

----End

## 4.13.5 Configuring a Dashboard

Dashboards, serving as custom monitoring platforms, allow you to view metrics.

This section describes how to configure a dashboard for a GeminiDB Redis instance.

## Procedure

- Step 1 Log in to the Huawei Cloud console.
- **Step 2** In the service list, click **Cloud Eye** to go to the Cloud Eye console.
- **Step 3** Create a dashboard.
  - 1. In the navigation pane, choose **My Dashboards** > **Custom Dashboards**. On the displayed page, click **Create Dashboard**.

Figure 4-186 Creating a dashboard

Cloud Eye	My Dashboards ①					Create Dashboard
Overview My Dashboards	Delete					Display favoritas only 🔳 🕸
Dashboards NEW Custom Dashboards	O: Select a property or enter a keyword.					0
Resource Groups	Name1D 0	Enterprine Project (8)	Created by (e)	Created B	Favorite 🖯	Operation
Alarn Management 🤍 🗸		default		May 17, 2021 15:29-16 GMT+08:00	Ŷ	Copy Delete
Server Menitoring V Clead Service Menitoring		dofault		Apr 09, 2022 17:00:19 GMT+08:00	\$	Copy Delete

2. In the displayed **Create Dashboard** dialog box, set parameters.

Figure 4-187 Configuring parameters

Create Dashboard	×
* Name	Enter a dashboard name.
★ Enterprise Project	default     ✓       Q     Create Enterprise Project []
	Cancel OK

 Table 4-76 Parameter description

Parameter	Description
Name	Dashboard name. The name can include a maximum of 128 characters. Only letters, digits, hyphens (-), and underscores (_) are allowed.
Enterprise Project	If you associate a monitoring dashboard with an enterprise project, only users who have the permissions of the enterprise project can manage the monitoring dashboard.
	<b>NOTE</b> The enterprise project feature is available only in some regions.

- 3. Click OK.
- **Step 4** Create a graph for the dashboard.

After a dashboard is created, you can add graphs to monitor your GeminiDB Redis instances.

- 1. On the **My Dashboards** page, click the target dashboard name. On the displayed dashboard details page, click **Create** to create a graph or graph group.
  - Graph: The trend or instantaneous values of a metric are displayed in different charts.
  - Graph group: Graphs in a dashboard can be grouped into different groups, which are similar to file directories.

Figure 4-188 Creating a graph

( test. v	Lef Ib v	Nec./Textuals ~	Asta-adaptation	v Ter	netr. v	٥
Grad Fed Stores Bare						
Count Stat				×		
					Ungrouped	

- 2. Click **Create Graph**. On the displayed page, configure parameters by following **Adding a Graph**.
  - a. In the **Graph Settings** area, select **One graph for multiple metrics** or **One graph for a single metric**. Select an existing group from the **Graph Group** drop-down list or click **Create Graph Group** to create a graph group.
  - b. You can select Line chart , Stacked area line chart, Bar chart, Horizonal bar chart, Donut chart, or Table chart for Graph Type.

### Figure 4-189 Graph settings

Graph Settings	:	
One graph for	multiple metrics One graph for a single	e metric
Graph Group	Select	~
Graph Type	Line chart	岱 Stacked area line chart
	0l₀ Bar chart	l≘ Horizontal bar chart
	O Donut chart	Table chart
Graph	Graph Name	
^ Basic Informat	ion	
Remarks (Optional		
^ Legend		
Location	Hide Show on the bottom	Show on the right
Legend Value	Select	~
△ Thresholds		
Threshold Line1		

c. Earlier edition: In the **Monitoring Item Configuration** area in the lower left corner, set **Monitoring Scope**, **Compare With**, and **Quantity**.

New console: In the **Select Metric** area, set the metric, monitoring scope (**All resources** or **Specified resources**), and whether to enable **Aggregation** and aggregation rules. Select **same period last week** or **same period yesterday** for **Compare With** and set the number of records displayed in a graph for the metric.

d. In the upper right corner of **Select Metric** area, select **Left Y axis** or **Right Y axis**. View the configured chart in the **Preview** area.

### Figure 4-190 Monitoring item configuration (earlier edition)

<   test-redis / Add Graph	Select Resource and Metric			×
Preview				
	Enter a service name or at O, Al	resources 🛛 🗸 You have selected 0 resources.		Select Metric
Graph Title 1 minute/Max	(*) Elastic Cloud Server (0   0)	Search by name by default.		
100	Elastic Volume Service (01 0)	Neree ID Tag	Enterprise Project	Select M 0
75	Object Storage Service (3   8)	) peminidh-feadhyu a2ad9eb38c97460aa2	default	_ GPS
50	Cemb(06(0)0)	tr_gb_651 cx730ed2732242e85	default	Instance Keys
8	<ul> <li>Cassandra (010)</li> </ul>			Tatal Traffic Sent by the In
0	<ul> <li>Influendo (0   0)</li> </ul>			Tatal Traffic Received by th
1600 1610 1620	Redia (010)			Keys with an Expiration TL.
				Max. Bytes Sent in a Requ-
	C mass (s ( s)			Max. Parameters Sent In a -
				Max Elements Obtained L.
New Console				Average Latency
Monitoring Item Configuration				p69 Latency
Monitoring Scope Select Resource and Metric				Average Hit Rate of the Ins
Compare With Same period last week Same period yesterday				Connection Usage
				Tatal Connections
CAUMANY - 50 +				
	Resources Selected 0 Metrics Selected 0			Cancel OK
	Monitoring Berns Added: U ( A maximum of 50 monitoring Berns c	in be added to a graph.)		

### Figure 4-191 Selecting a metric (new console)

Earlier Edition		
CPU Usage (Gemin	iDB - Redis )	Left 🗸 📋 🗊
Metric	GeminDB - Redis V Redis / CPU Usage V	
Monitoring Scope	All resources Specific resources	
Aggregation		
Aggregation Settings	-Select- v	
Compare With	Same period last week Same period yesterday	
Display	Descending          -         50         +           The maximum number of records displayed in a graph for the metric: 50         -          -          -	

### **NOTE**

When you add a graph, select **One graph for a single metric**. Then a graph is generated for each metric, making it easy for you to view and analyze monitored data. If you need multiple metrics, add monitoring graphs.

3. On the selected dashboard, you can view the metric trend in the new graph.



### Figure 4-192 New graph

----End

# 4.13.6 Event Monitoring

## 4.13.6.1 Overview

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 Redis 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.

# 4.13.6.2 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 You can view event monitoring data in either of the following ways:
  - On the **Instances** page, click the target instance. In the navigation pane, choose **Metrics**. You can check monitoring data on the **DB Instance** or **Node-level Metrics** tab page.
  - On the **Instances** page, click the target instance. In the navigation pane, choose **Nodes**. On the displayed page, locate the target node and click **View Metric** in the **Operation** column. On the displayed Cloud Eye console, view the event monitoring data.
- **Step 4** Click <sup><</sup> 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

# 4.13.6.3 Creating an Alarm Rule for Events

# Scenarios

This topic describes how to create an alarm rule for events.

# **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 4-77 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 <b>Create</b> <b>Enterprise Project</b> 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. Selecting GeminiDB.
Monitoring Scope	Specifies the monitoring scope for event monitoring.
Method	Specifies the event creation method.
Alarm Policy	<b>Event Name</b> indicates the instantaneous operations users performed on system resources, such as login and logout.
	For details about events supported by Event Monitoring, see <b>4.13.6.4 Monitored Events</b> .
	You can select a trigger mode and alarm severity as needed.

Click **C** 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**.

Parameter	Description
Alarm Notification	Specifies whether to notify users when alarms are triggered. Notifications can be sent by email, text message, or HTTP/ HTTPS message.
Notification Object	Specifies the object an alarm notification is to be sent to. You can select the account contact or a topic.
	• Account contact is the mobile phone number and email address provided for registration.
	• Topic is used to publish messages and subscribe to notifications. If the required topic is unavailable, create one first and add subscriptions to it. For details, see <b>Creating a Topic</b> and <b>Adding Subscriptions</b> .
Validity Period	Cloud Eye sends notifications only within the validity period specified in the alarm rule.
	If you set <b>Validity Period</b> to <b>08:00-20:00</b> , Cloud Eye sends notifications only within 08:00-20:00.
Trigger Condition	Specifies the condition for triggering the alarm notification.

 Table 4-78
 Alarm notification parameters

**Step 6** After the configuration is complete, click **Create**.

----End

# 4.13.6.4 Monitored Events

Event Sourc e	Event Name	Event ID	Eve nt Sev erit y	Description	Solution	lmpa ct
NoSQ L	Instance creation failure	NoSQL Createl nstance Failed	Maj or	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.	Instan ces fail to be create d.
	Specificati ons change failure	NoSQL Resizel nstance Failed	Maj or	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.	Servic es are interr upted.

Table 4-79 Events Supported by Event Monitoring for GeminiDB

Event Sourc e	Event Name	Event ID	Eve nt Sev erit y	Description	Solution	lmpa ct
	Node adding failure	NoSQL AddNo desFail ed	Maj or	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	NoSQL Delete NodesF ailed	Maj or	Releasing underlying resources failed.	Delete the node again.	None
	Storage space scale-up failure	NoSQL ScaleU pStorag eFailed	Maj or	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.	Servic es may be interr upted.

Event Sourc e	Event Name	Event ID	Eve nt Sev erit y	Description	Solution	lmpa ct
	Password resetting failure	NoSQL ResetPa ssword Failed	Maj or	Resetting the password times out.	Reset the password again.	None
	Parameter template change failure	NoSQL Updatel nstance Param GroupF ailed	Maj or	Changing a parameter template times out.	Change the parameter template again.	None
	Backup policy configurat ion failure	NoSQL SetBack upPolic yFailed	Maj or	The database connection is abnormal.	Configure the backup policy again.	None
	Manual backup creation failure	NoSQL Create Manual Backup Failed	Maj or	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 canno t be backe d up.
	Automate d backup creation failure	NoSQL CreateA utomat edBack upFaile d	Maj or	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 canno t be backe d up.

Event Sourc e	Event Name	Event ID	Eve nt Sev erit y	Description	Solution	lmpa ct
	Instance status abnormal	NoSQL FaultyD BInstan ce	Maj or	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 datab ase servic e may be unava ilable.
	Instance status recovery	NoSQL DBInsta nceRec overed	Maj or	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	NoSQL FaultyD BNode	Maj or	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 datab ase servic e may be unava ilable.

Event Sourc e	Event Name	Event ID	Eve nt Sev erit y	Description	Solution	lmpa ct
	Node status recovery	NoSQL DBNod eRecov ered	Maj or	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 switchove r or failover	NoSQL Primary Standb ySwitch ed	Maj or	This event is reported when a primary/ secondary switchover or a failover is triggered.	No further action is required.	None
	Occurrenc e of hotspot partitionin g keys	HotKey Occurs	Maj or	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.	<ol> <li>Choose a proper partition key.</li> <li>Add service cache so that service applications read hotspot data from the cache first.</li> </ol>	The servic e reque st succes s rate is affect ed, and the cluste r perfor manc e and stabili ty deteri orates

Event Sourc e	Event Name	Event ID	Eve nt Sev erit y	Description	Solution	lmpa ct
	BigKey occurrenc e	BigKey Occurs	Maj or	The primary key design is improper. There are too many records or too much data in a single partition, causing load imbalance on nodes.	<ol> <li>Choose a proper partition key.</li> <li>Add a new partition key for hashing data.</li> </ol>	As more and more data is stored in the partiti on, cluste r stabili ty deteri orates
	Insufficien t storage space	NoSQL RiskyDa taDiskU sage	Maj or	The storage space is insufficient.	Scale up storage space. For details, see section "Scaling Up Storage Space" in the user guide of GeminiDB.	The instan ce is set to read- only and data canno t be writte n to the instan ce.
	Data disk expanded and being writable	NoSQL DataDi skUsag eRecov ered	Maj or	The data disk has been expanded and becomes writable.	No further action is required.	None

Event Sourc e	Event Name	Event ID	Eve nt Sev erit y	Description	Solution	lmpa ct
	Index creation failure	NoSQL Createl ndexFai led	Maj or	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.	<ol> <li>Select matched instance specifications based on the service loads.</li> <li>Create indexes during off- peak hours.</li> <li>Create indexes in the background.</li> <li>Select indexes as required.</li> </ol>	The index fails to be create d or is incom plete. Delet e the index and create a new one.
	Write speed decrease	NoSQL Stalling Occurs	Maj or	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.	<ol> <li>Adjust the cluster scale or node specifications based on the maximum write rate of services.</li> <li>Measure the maximum write request rate of services and distribute the peak write rate of services.</li> </ol>	The succes s rate of servic e reque sts is affect ed.

Event Sourc e	Event Name	Event ID	Eve nt Sev erit y	Description	Solution	lmpa ct
	Data write stopped	NoSQL Stoppin gOccur s	Maj or	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.	<ol> <li>Adjust the cluster scale or node specifications based on the maximum write rate of services.</li> <li>Measure the maximum write request rate of services and distribute the peak write rate of services.</li> </ol>	The succes s rate of servic e reque sts is affect ed.
	Database restart failure	NoSQL Restart DBFaile d	Maj or	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 instan ce status may be abnor mal.

Event Sourc e	Event Name	Event ID	Eve nt Sev erit y	Description	Solution	lmpa ct
	Restoratio n to new instance failure	NoSQL Restore ToNewl nstance Failed	Maj or	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 canno t be restor ed to a new instan ce.
	Restoratio n to existing instance failure	NoSQL Restore ToExistI nstance Failed	Maj or	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 curren t instan ce may be unava ilable.
	Backup file deletion failure	NoSQL DeleteB ackupF ailed	Maj or	The backup files fail to be deleted from OBS.	Delete the backup files again.	None

Event Sourc e	Event Name	Event ID	Eve nt Sev erit y	Description	Solution	lmpa ct
	Failure to display slow query logs in plaintext	NoSQL SwitchS lowlog PlainTe xtFailed	Maj or	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	NoSQL BindEip Failed	Maj or	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 instan ce canno t be access ed from a public netwo rk.
	EIP unbinding failure	NoSQL Unbind EipFaile d	Maj or	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 Sourc e	Event Name	Event ID	Eve nt Sev erit y	Description	Solution	lmpa ct
	Parameter modificati on failure	NoSQL Modify Parame terFaile d	Maj or	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 applicatio n failure	NoSQL ApplyP aramet erGrou pFailed	Maj or	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
	Enabling or disabling SSL failure	NoSQL SwitchS SLFaile d	Maj or	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.	The SSL conne ction mode canno t be chang ed.

Event Sourc e	Event Name	Event ID	Eve nt Sev erit y	Description	Solution	lmpa ct
	Too much data in a single row	LargeR owOcc urs	Maj or	If there is too much data in a single row, queries may time out, causing faults like OOM error.	<ol> <li>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.</li> <li>Check whether there are abnormal writes or coding, causing large rows.</li> </ol>	If there are too many record s in a single row, cluste r stabili ty will deteri orate as the data volum e increa ses.
	Schedule for deleting a KMS key	planDel eteKms Key	Maj or	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 auto matic ally delete d after it expire s. Deleti ng the key will affect the instan ce servic es.

Event Sourc e	Event Name	Event ID	Eve nt Sev erit y	Description	Solution	lmpa ct
	Too many tombston es	TooMa nyQuer yTombs tones	Maj or	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	TooLar geColle ctionCo lumn	Maj or	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. Check whether there is an error while data is written and encoded	The query on the collec tion colum n will fail.

# 4.14 Tag Management

# **Scenarios**

Tag Management Service (TMS) enables you to use tags on the management console to manage resources. TMS works with other cloud services to manage global tags, and other cloud services manage their own tags.

Adding tags to GeminiDB Redis instances helps you better identify and manage them. A DB instance can be tagged during or after it is created.

After a DB 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 4-80**.
- A maximum of 20 tags can be added for each instance.
- The tag name must comply with the naming rules described in Table 4-80.

Parameter	Requirement	Example Value
Tag key	Cannot be left blank.	Organization
	• Must be unique for each instance.	
	<ul> <li>Can contain a maximum of 128 characters.</li> </ul>	
	<ul> <li>Cannot start with _sys_ and cannot start or end with a space. Only letters, digits, spaces, and the following special characters are allowed:@.:/+=</li> </ul>	
Tag value	Can be left blank.	nosql_01
	<ul> <li>Can contain a maximum of 255 characters.</li> </ul>	
	<ul> <li>Only letters, digits, spaces, and the following special characters are allowed:@.:/+=</li> </ul>	

#### Table 4-80 Naming rules

# 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 instance that you want to add tags to and click its name.
- **Step 4** In the navigation pane on the left, choose **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 the tag 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, locate the instance whose tags you want to edit and click its name.
- **Step 4** In the navigation pane on the left, choose **Tags**.
- **Step 5** On the **Tags** page, locate the tag to be edited 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 the tag 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, locate the instance whose tags you want to delete and click its name.
- **Step 4** In the navigation pane on the left, choose **Tags**.
- **Step 5** On the **Tags** page, locate the tag to be deleted and click **Delete** in the **Operation** column. In the displayed dialog box, click **Yes**.
- **Step 6** View that the tag is no longer displayed on the **Tags** page.

----End

# Searching an Instance 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 4-193 Selecting tags



**Step 4** Select the tag to be queried and click **OK** to query information about instances associated with the tag.

#### Figure 4-194 Searching by tag

Q Tag	gs:Add filter	
	(Select all)	
	✓ tag_key_2024060709543	
	aa = aa	ng
	aa = bb	
	🗌 a = a	abl
	cc = cc	
	Cancel OK	abl

----End

# 4.15 Managing User Resource Quotas of a GeminiDB Redis Instance

# **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.

# **Viewing Quotas**

- Step 1 Log in to the Huawei Cloud console.
- **Step 2** In the service list, choose **Databases** > **GeminiDB**.
- **Step 3** Click **(2)** in the upper left corner and select a region and project.
- **Step 4** In the upper right corner, choose **Resources** > **My Quotas**.

The **Quota** page is displayed.

#### Figure 4-195 My quotas



**Step 5** On the **Quotas** page, view the used and total quotas of each type of GeminiDB 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 *Q* 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 4-196 Increasing quotas

My Quotas		
	Queter 0	
Quotas	Quotas 0	Increase Coudo

**Step 6** On the **Create Service Ticket** page, configure parameters as required.

In the **Problem Description** area, enter the required quota and reason for the quota adjustment.

**Step 7** After all necessary parameters are configured, select the agreement and click **Submit**.

----End

# 4.16 Memory Acceleration

# 4.16.1 RDS Memory Acceleration

# 4.16.1.1 Memory Acceleration Overview

GeminiDB Redis API offers memory acceleration to enhance the conventional cache solution. With this feature, users can set up rules on the GUI to cache MySQL data automatically, thereby speeding up MySQL access.

The conventional cache solution is inefficient and unreliable as it necessitates users to create code for writing MySQL data to the cache. The active cache solution with cloud data memory acceleration (DB Cache) supports visualized configuration on the GUI, making it easier to set up. Once the configuration is done, data can be synchronized automatically. DB Cache also supports data filtering and expiration time setting, which enhances development efficiency and data reliability.





# 4.16.1.2 Enabling and Using Memory Acceleration

Enable memory acceleration.

Step 1: Create a GeminiDB instance.

Step 2: Create a mapping rule.

#### Step 3: Use the memory acceleration module.

# Usage Notes

- After memory acceleration is enabled, commands such as RESET MASTER and FLUSH LOGS used to delete binlogs on MySQL instances are not allowed.
- Currently, only hash data from MySQL can be converted to GeminiDB Redis API.
- A Redis key prefix and a delimiter in a new rule can neither include those nor be included in those specified for an existing rule. For example, if the key prefix in a new rule is **pre1**: and is separated by a comma (,) and the key prefix in an existing rule is **pre1** and is separated by a colon (:), the new rule cannot be created.
- Currently, the ENUM, SET, and JSON data cannot be synchronized.
- Currently, only single-table queries are supported during lightweight incremental synchronization. Joint queries are not supported.
- Only GeminiDB Redis instances are charged. There are no other fees for this function.

- If you delete an RDS instance, the GeminiDB Redis instance with DB Cache enabled will not be deleted. If you do not need the GeminiDB Redis instance, delete it in a timely manner to avoid extra fees.
- When you purchase an RDS instance, if you select **Buy Now** for memory acceleration, a GeminiDB instance is automatically provisioned with DB Cache enabled. You can skip instance creation and start from Creating a Mapping Rule. This function is now in OBT. To use it, choose Service Tickets > Create Service Ticket in the upper right corner of the console and contact the customer service.

# Procedure

# **Creating a GeminiDB Instance**

- Step 1 Log in to the management console.
- **Step 2** Click <sup>1</sup> in the upper left corner and select a region and project.
- Step 3 Click in the upper left corner of the page and choose Databases > Relational Database Service.
- **Step 4** On the **Instances** page, click the target instance name to go to the **Overview** page.
- **Step 5** In the navigation pane, choose **Memory Acceleration**.
  - Click **Create GeminiDB Instance** and perform **Step 6**.
  - Click **Use Existing GeminiDB Instance** and select an existing GeminiDB Redis instance.

# NOTICE

When you select **Use Existing GeminiDB Instance**, only primary/standby instances are supported. The region, VPC, subnet, and security group of the GeminiDB and RDS instances must be the same.

**Step 6** Set parameters listed in **Table 4-81** and click **Submit**.

Table 4-81	Basic	information
------------	-------	-------------

Parameter	Description
Instance	CPU and memory of the instance. For details, see <b>Table</b>
Specifications	<b>4-82</b> .

Parameter	Description				
Database Port	Port number for accessing the instance.				
	You can specify a port number based on your requirements. The port number ranges from 1024 to 65535 except 2180, 2887, 3887, 6377, 6378, 6380, 8018, 8079, 8091, 8479, 8484, 8999, 12017, 12333, and 50069.				
	If you do not specify a port number, port 6379 is used by default.				
	<b>NOTE</b> You cannot change the database port after an instance is created.				
DB Instance	The instance name:				
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 (_).				
Database	Database password set by the user.				
Password	Must be 8 to 32 characters long.				
	<ul> <li>Can include two of the following: uppercase letters, lowercase letters, digits, and special characters: ~!@#%^*- _=+?</li> </ul>				
	• For security reasons, set a strong password. The system will verify the password strength.				
	Keep your password secure. The system cannot retrieve it if it is lost.				
Confirm Password	Enter the administrator password again.				

# 

By default, the region, AZ, VPC, and subnet of the GeminiDB and RDS instances are the same.

Table 4-82	GeminiDB	Redis	instance	specifications
------------	----------	-------	----------	----------------

Storage (GB)	Nodes	vCPUs	QPS	Maximum Connections per Single-node Instance	Databas es
16	2	1	10,000	10,000	1,000
24	2	2	20,000	10,000	1,000
32	2	2	20,000	10,000	1,000
48	2	4	40,000	20,000	1,000

Storage (GB)	Nodes	vCPUs	QPS	Maximum Connections per Single-node Instance	Databas es
64	2	4	40,000	20,000	1,000
96	2	8	80,000	20,000	1,000
128	2	16	160,000	20,000	1,000

----End

# Creating a Mapping Rule

- Step 1 Log in to the Huawei Cloud console.
- **Step 2** Click <sup>(Q)</sup> in the upper left corner and select a region and project.
- Step 3 Click in the upper left corner of the page and choose Databases > Relational Database Service.
- **Step 4** On the **Instances** page, click the target instance name to go to the **Overview** page.
- **Step 5** In the navigation pane, choose **Memory Acceleration**. In the **Mapping Rule** area, click **Create Mapping Rule**.

#### Figure 4-198 Mapping rule

Mapping Rule			
Create Mapping Rule			
Q. Enter a keyword.			00
Rule Name/ID	Status	Source Instance Database Source Instance Table Name Target Data Type Target Database No. Operation	
		No synchronization sules have been created	

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

1. Enter a rule name.

**Rule Name**: Enter a mapping rule name. The rule name must be unique within a GeminiDB instance and cannot exceed 256 characters or include number signs (#).

Figure 4-19	<b>9</b> Rule name
-------------	--------------------

Enter a rule name.	

- 2. Configure source instance information.
  - Database Name: Select a database of the acceleration instance.
  - **Table Name**: Select a table of the acceleration instance.

Figure 4-200 Configuring source instance information

Source instance Configuration			
Source Instance Name	Database Name	Database Name	
	Select a database.	Select a table.	~

- 3. Configure acceleration instance information.
  - Redis Key Prefix: This parameter is optional. The default value is in the format of *Database name*: *Table name*. *Field name 1*: *Field name 2...* and can contain a maximum of 1,024 characters. If you have created a custom prefix, it will be used instead of the default one.
  - Value Storage Type: Data type of the cache. Currently, only hash data is supported.
  - **Database No. (0-999)**: ID of a database that stores cached data in the acceleration instance. The default value is **0**.
  - TTL (s) Default value: 30 days: Validity period of cached data in the acceleration instance. The default value is 30 days (2,592,000 seconds). If you enter -1, the cached data will never expire.
  - Key Delimiter: Separator among the Redis key prefix, key, and key fields.
     It is a single character in length.

Figure 4-201 Configuring acceleration instance information

Acceleration Instance Configuration		
Acceleration Instance Name	Redis Key Prefix	Value Storage Type
		Hash 🗸
Database No. (0 to 999)	TTL (\$) Default value: 30 days	Key Delimiter
0	2,592,000	

4. Click **Set Key**, select a key field of the acceleration instance, and click **OK**.

#### **NOTE**

If an acceleration instance key consists of multiple source instance fields, the key must be unique in a MySQL instance. You can click **Up** or **Down** to adjust the sequence of each field.

×

Figure 4-202	Key	settings
--------------	-----	----------

Source instance fields 0 / 4	Acceleration instance key
Q Enter a keyword.	Q Enter a keyword.
sname	🗌 sid
sclass	
sgender	
sbirthday	
	Up Down
	OK Cancel

After the parameters are set, the key is displayed.

#### Figure 4-203 Key

Set Key Hash db0:student:sid:<sid>

Configure the acceleration instance fields.
 Move the required fields in the source instance to the acceleration instance.

Figure 4-204 Configuring acceleration instance fields

s	et Va	lue												
	Availa	ible					2/5		Select	ed				0/0
	Q	Enter a keyword.								Enter a keyword.				
	۰	Field	Туре	Null	Key	Extra				Field	Туре	Null	Key	Extra
		sid	int unsigned	NO	PRI			>						
	~	sname	varchar(32)	NO										
	~	sclass	varchar(32)	NO								No data available		
		sgender	varchar(32)	NO										

6. After setting the parameters, click **Submit**.

----End

# Using the Memory Acceleration Module

 Create database db1 in the source MySQL instance and create table students in db1.

```
mysql> CREATE DATABASE db1;
Query OK, 1 row affected (0.00 sec)
```

```
mysql> CREATE TABLE db1.students(
   sid INT UNSIGNED PRIMARY KEY AUTO_INCREMENT NOT NULL,
   sname VARCHAR(20),
   sclass INT,
   sgender VARCHAR(10),
   sbirthday DATE
   );
Query OK, 0 rows affected (0.00 sec)
mysgl> DESC db1.students;
      | Field | Type | Null | Key | Default | Extra |
sid | int unsigned | NO | PRI | NULL | auto_increment |
| sname | varchar(20) | YES | | NULL | |
sclass | int | YES | | NULL |
sclass | int | YES | | NULL |
sgender | varchar(10) | YES | | NULL |
                                           |sbirthday|date |YES||NULL|
                                         +-----
                 --+----+----+----
```

- 5 rows in set (0.00 sec)
- After the table is created, on the memory acceleration page, create a mapping rule to convert each row in the students table into a Redis hash. The key of a hash is in the format of *Database name*:*Data table name*:sid:<sid value>. The selected fields are sname, sclass, sgender, and sbirthday.

#### Figure 4-205 Configuring a mapping rule

Rule Name students-to-hash												
Source Instance Configuration Source Instance Name Database Name disc									Database	a Name		
Acceleration Instance C Acceleration Instance Name Database No. (0 to 999)	configuration			Redis Key Prefix Enter a Redis key prefix. TTL (s) Default value: 30 da 2,592,000	ays				Value Str Hash Key Delin	orage Type		
Set Key Hash Set Value				0/4		Sala	cted					0/4
C Enter a keyword.	Туре	Null	Key	Extra			Enter a keyword.	Type	Null	Key	Extra	
sid	int unsigned	NO	PRI				sname	varchar(32)	NO			
							sgender	varchar(32)	NO			
							soirtnday	varcnar(32)	NŬ			

3. After a mapping rule is created, check the mapping rule and information.

#### Figure 4-206 Mapping information

Mapping							C
Mapping Name C <sup>1</sup> Mapping ID	đ	Seurce Instance ID	CP Accelerator Instanc	10 d			
	•	Austing	configured	O Running			
Number Data							
Napping Hule							
Create Mapping Rule							
C), Enter a keyword.							
Rule Name/D	Status	Source Instance Database	Source Instance Table Name	Target Data Type	Target Database No.	Operation	
	O Available	60	student	HASH	1	Edit Delete	
Total Records: 1 11 V							

4. Insert a new data record to the **students** table in the MySQL instance. mysql> INSERT INTO db1.students (sname, sclass, sgender, sbirthday) VALUES ('zhangsan', 1, 'male', '2015-05-20'); Query OK, 1 row affected (0.01 sec)

5. After the mapping rule is created, the data is automatically synchronized to the GeminiDB instance. Run commands in the GeminiDB instance to query the data.

127.0.0.1:6379> KEYS \* 1) "db1:students:sid:1"

127.0.0.1:6379> HGETALL db1:students:sid:1 1) "sbirthday" 2) "2015-05-20" 3) "sclass" 4) "1" 5) "sgender" 6) "male" 7) "sname"

- 8) "zhangsan"
- 6. Insert a new data record to the **students** table in the MySQL instance. mysql> INSERT INTO db1.students (sname, sclass, sgender, sbirthday) VALUES ('lisi', 10, 'male', '2015-05-22');

Query OK, 1 row affected (0.00 sec)

mysql> SELECT \* FROM db1.students;

TT-TTTTTT
sid   sname   sclass   sgender   sbirthday
1   zhangsan   1   male   2015-05-20   2   lisi   10   male   2015-05-22   ++

2 rows in set (0.00 sec)

- 7. Check whether the new data is synchronized to the GeminiDB instance. 127.0.0.1:6379> KEYS \*
  - 1) "db1:students:sid:1"
  - 2) "db1:students:sid:2"

127.0.0.1:6379> HGETALL db1:students:sid:2

- 1) "sbirthday"
- 2) "2015-05-22"
- 3) "sclass"
- 4) "10"
- 5) "sgender"
- 6) "male"
- 7) "sname"
- Update data in the students table in the MySQL instance. mysql> UPDATE db1.students SET sclass=12, sname='wangwu' WHERE sid = 1; Query OK, 1 row affected (0.00 sec) Rows matched: 1 Changed: 1 Warnings: 0

mysql> SELECT \* FROM db1.students;

++			+
sid   sname	sclass	sgender	sbirthday
1   wangwu   2   lisi	12  10 mal	male   e  201	2015-05-20   5-05-22
2 rows in set (	0.00 sec)	)	,+

 Check whether the data is updated in the GeminiDB instance. 127.0.0.1:6379> KEYS \* 1) "db1:students:sid:1" 2) "db1:students:sid:2"

- 127.0.0.1:6379> HGETALL db1:students:sid:1 1) "sbirthday" 2) "2015-05-20" 3) "sclass" 4) "12" 5) "sgender" 6) "male" 7) "sname" 8) "wangwu"
- 10. Delete data from the **students** table in the MySQL instance. mysql> DELETE FROM db1.students WHERE sid = 1; Query OK, 1 row affected (0.00 sec)

```
mysql> SELECT * FROM db1.students;
+----+
| sid | sname | sclass | sgender | sbirthday |
+----+
| 2 | lisi | 10 | male | 2015-05-22 |
+----+
1 row in set (0.00 sec)
```

11. Check whether the data is deleted from the GeminiDB instance.
127.0.0.1:6379> KEYS \*
1) "db1:students:sid:2"

# 4.16.1.3 Modifying and Deleting a Memory Acceleration Rule

A memory acceleration rule can enable automated data synchronization from MySQL to GeminiDB. You can also modify and delete this rule.

# Precautions

- Currently, only hashes from MySQL can be converted to GeminiDB Redis API.
- If a table name of the MySQL instance in the memory acceleration rule is changed, you need to reconfigure the rule.
- Currently, the ENUM, SET, and JSON data cannot be synchronized.
- If you rename or delete one or more key fields of a memory acceleration rule, the rule becomes invalid.

# Modifying a Mapping Rule

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

- **Step 2** Click <sup>1</sup> in the upper left corner and select a region and project.
- Step 3 Click in the upper left corner of the page and choose Databases > Relational Database Service.
- **Step 4** On the **Instances** page, click the target instance name to go to the **Overview** page.
- **Step 5** In the navigation pane on the left, choose **Memory Acceleration**. In the **Mapping Rule** area, locate the target rule and click **Edit** in the **Operation** column.

#### Figure 4-207 The Edit button

Mapping Rule							
Create Mapping Rule							
C. Enter a largeword.							00
Rule Namo1D	Status	Source Instance Delabase	Source Instance Table Name	Target Data Type	Target Database No.	Operation	
	O Available	do0	student	наян	0	Edit Delete	
Total Recards: 1 $-$ 10 $$ $$ $$ $$ $$ $$ $$ $$ $$ $$							

#### **Step 6** After editing the fields, click **Submit**.

#### Figure 4-208 Editing a mapping rule

wailable					0/2	Sele	cted				
Q Enter a keyword.							Enter a keyword.				
Field	Туре	Null	Key	Extra		۰	Field	Туре	Null	Key	Extra
sid	int unsigned	NO	PRI		>		sname	varchar(32)	NO		
sclass	varchar(32)	NO			<		sgender	varchar(32)	NO		
							sbirthday	varchar(32)	NO		

----End

# Deleting a Mapping Rule

Step 1 Log in to the management console.

- **Step 2** Click <sup>1</sup> in the upper left corner and select a region and project.
- Step 3 Click in the upper left corner of the page and choose Databases > Relational Database Service.
- **Step 4** On the **Instances** page, click the target instance name to go to the **Overview** page.
- **Step 5** In the navigation pane on the left, choose **Memory Acceleration**. In the **Mapping Rule** area, locate the target rule and click **Delete** in the **Operation** column.

----End

# 4.16.1.4 Viewing and Removing Mappings

You can view the mapping list on the **Memory Acceleration Management** page and remove mappings.

# Usage Notes

- After a mapping is removed, service applications cannot obtain the latest data of the source database from the acceleration instance.
- The corresponding mapping rule will be cleared after a mapping is removed.
- If the source instance or acceleration instance is not normal, the mapping cannot be removed.

# **Querying the Mapping List**

- 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 **Memory Acceleration Management**. On the displayed page, search for your target mapping by keyword (such as the mapping name or mapping ID).

#### Figure 4-209 Mapping list

Q B	earch by instance name by default.						
Asp -	API Fillering	Status	Source Instance Name/D	Acceleration Instance NameID	Created	Operation	
	Mapping Name	0.0000			N 71 201 01 18 18 007-08 00	Process of	
	Mapping ID	C ALLON			0.11, 0.0 0.00 0.00 0.00	141.000	
	Source Instance Name						
	Source Instance ID						
	Acceleration Instance Name						
	Acceleration Instance (f)						

----End

# **Removing a Mapping**

- 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 Memory Acceleration Management. On the displayed page, locate the target mapping and click Remove in the Operation.

Figure 4-210 Memory acceleration management



# 4.16.2 TaurusDB Memory Acceleration

# 4.16.2.1 Overview

Different from conventional passive caching solutions, memory acceleration of GeminiDB Redis API speeds up access to TaurusDB databases. Users can create rules on the GUI to automatically cache TaurusDB database data.

The conventional caching solution is inefficient and unreliable because users have to develop code for caching TaurusDB instance data. GeminiDB provides an automated proactive caching solution to prevent transaction failures caused by cache breakdown. After parameters are configured on the GUI, data can be automatically synchronized. You can also filter data and set a validity period to enhance development efficiency and data reliability.

#### Figure 4-212 Memory acceleration



# 4.16.2.2 Enabling and Using Memory Acceleration

This section describes how to enable memory acceleration. The process is as follows:

**Step 1: Create a GeminiDB instance.** 

Step 2: Create a mapping rule.

Step 3: Use the memory acceleration module.

# 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.
- After memory acceleration is enabled, commands such as RESET MASTER and FLUSH LOGS used to delete binlogs on TaurusDB instances are not allowed.
- A Redis key prefix and a delimiter in a new rule can neither include those nor be included in those specified for an existing rule. For example, if the key prefix in a new rule is **pre1:** and is separated by a comma (,) and the key

prefix in an existing rule is **pre1** and is separated by a colon (:), the new rule cannot be created.

- Currently, the ENUM, SET, and JSON data cannot be synchronized.
- Currently, only single-table queries are supported during lightweight incremental synchronization. Joint queries are not supported.
- Only GeminiDB Redis instances are charged. There are no other fees for this function.
- When you purchase a TaurusDB instance, if you select **Buy Now** for memory acceleration, a GeminiDB instance is automatically provisioned with DB Cache enabled. You can skip instance creation and start from Creating a Mapping Rule.

# Procedure

# **Creating a GeminiDB Instance**

Step 1 Log in to the Huawei Cloud console.

- **Step 2** Click <sup>1</sup> in the upper left corner and select a region and project.
- **Step 3** Click = in the upper left corner of the page and choose **Databases** > **TaurusDB**.
- **Step 4** On the **Instances** page, click the instance name to go to the **Basic Information** page.
- **Step 5** In the navigation pane, choose **Memory Acceleration**. Click **Create GeminiDB Instance** or **Use Existing GeminiDB Instance**.
  - Click Create GeminiDB Instance and perform Step 6.
  - Click **Use Existing GeminiDB Instance** and select an existing GeminiDB Redis instance.

**Use Existing GeminiDB Instance** is only available for primary/standby instances.

**Step 6** Set parameters listed in **Table 4-83** and click **Submit**.

**NOTE** 

By default, the region, AZ, VPC, and subnet of the GeminiDB and TaurusDB instances are the same.

Table 4-83 Basic information

Parameter	Description
Instance	CPU and memory of the instance. For details, see <b>Table</b>
Specifications	<b>4-84</b> .

Parameter	Description
Database Port	Port number for accessing the instance.
	You can specify a port number based on your requirements. The port number ranges from 1024 to 65535 except 2180, 2887, 3887, 6377, 6378, 6380, 8018, 8079, 8091, 8479, 8484, 8999, 12017, 12333, and 50069.
	If you do not specify a port number, port 6379 is used by default.
	<b>NOTE</b> You cannot change the database port after an instance is created.
DB Instance	The instance name:
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 (_).
Database	Database password set by the user.
Password	Must be 8 to 32 characters long.
	<ul> <li>Can include two of the following: uppercase letters, lowercase letters, digits, and special characters: ~!@#%^*- _=+?</li> </ul>
	• For security reasons, set a strong password. The system will verify the password strength.
	Keep your password secure. The system cannot retrieve it if it is lost.
Confirm Password	Enter the administrator password again.

#### Table 4-84 GeminiDB instance specifications

Storage (GB)	Nodes	vCPUs	QPS	Maximum Connections per Single-node Instance	Databas es
16	2	2	10,000	10,000	1,000
24	2	2	40,000	10,000	1,000
32	2	2	40,000	10,000	1,000
48	2	4	80,000	20,000	1,000
64	2	4	80,000	20,000	1,000
96	2	8	160,000	20,000	1,000
128	2	16	320,000	20,000	1,000

----End

# **Creating a Mapping Rule**

Step 1 Log in to the Huawei Cloud console.

- **Step 2** Click <sup>(Q)</sup> in the upper left corner and select a region and project.
- **Step 3** Click = in the upper left corner of the page and choose **Databases** > **TaurusDB**.
- **Step 4** On the **Instances** page, click the instance name to go to the **Basic Information** page.
- **Step 5** In the navigation pane, choose **Memory Acceleration**. In the **Mapping Rule** area, click **Create Mapping Rule**.

#### Figure 4-213 Mapping rule

Mapping Rule							
Create Mapping Rule							
Q Enter a keyword.							0
Rule Name/ID	Status	Source Instance Database	Source Instance Table Name	Target Data Type	Target Database No.	Operation	
			0				
		4	- <b>A</b> -				
		=					

No synchronization rules have been created.

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

1. Enter a rule name.

**Rule Name**: Enter a mapping rule name. The rule name must be unique within a GeminiDB instance and cannot exceed 256 characters or include number signs (#).

Figure 4-214 Rule name

Dula Nama	
Rule Name	
Enter a rule name.	

- 2. Configure source instance information.
  - **Database Name**: Select a database of the acceleration instance.
  - **Table Name**: Select a table of the acceleration instance.

Figure 4-215 Configuring source instance information

•		
Source Instance Name	Database Name	Database Name
	Select a database.	Select a table.

3. Configure acceleration instance information.

Source Instance Configuration

- Redis Key Prefix: This parameter is optional. The default value is in the format of *Database name: Table name: Field name 1: Field name 2...* and can contain a maximum of 1,024 characters. If you have created a custom prefix, it will be used instead of the default one.
- **Value Storage Type**: Data type of the cache. Currently, only hash data is supported.
- **Database No. (0-999)**: ID of a database that stores cached data in the acceleration instance. The default value is **0**.
- TTL (s) Default value: 30 days: Validity period of cached data in the acceleration instance. The default value is 30 days (2,592,000 seconds). If you enter -1, the cached data will never expire.
- Key Delimiter: Separator among the Redis key prefix, key, and key fields.
   It is a single character in length.

Figure 4-216 Configuring acceleration instance information

Acceleration Instance Configuration		
Acceleration Instance Name	Redis Key Prefix	Value Storage Type
		Hash 🗸
Database No. (0 to 999)	TTL (s) Default value: 30 days	Key Delimiter
0	2,592,000	

4. Click **Set Key**, select a key field of the acceleration instance, and click **OK**.

#### **NOTE**

If an acceleration instance key consists of multiple source instance fields, the key must be unique in a TaurusDB instance. You can click **Up** or **Down** to adjust the sequence of each field.
$\times$ 

Figure 4-217	Key settings
--------------	--------------

#### Set Key

Source instance fields	0/4	Acceleration instance key
Q Enter a keyword.		Q Enter a keyword.
sname		id sid
sclass		
sgender		
sbirthday		
		Up Down
		OK Cancel

After the parameters are set, the key is displayed.

#### Figure 4-218 Key



Configure the acceleration instance fields.
 Move the required fields in the source instance to the acceleration instance.

Figure 4-219 Configuring acceleration instance fields

Set	/alue												
Ave	ilable					2/5		Selec	ted				0/0
	Enter a keyword.								Enter a keywo	ord.			
٠	Field	Туре	Null	Key	Extra				Field	Туре	Null	Key	Extra
	sid	int unsigned	NO	PRI			>						
	sname	varchar(32)	NO										
	sclass	varchar(32)	NO								No data availat	sle	
	sgender	varchar(32)	NO										
	sgender	varchar(32)	NO										

6. After setting the parameters, click **Submit**.

----End

## Using the Memory Acceleration Module

1. Create database **db1** in the source TaurusDB instance and create table **students** in **db1**.

```
mysql> CREATE DATABASE db1;
Query OK, 1 row affected (0.00 sec)
mysgl> CREATE TABLE db1.students(
    sid INT UNSIGNED PRIMARY KEY AUTO_INCREMENT NOT NULL,
    sname VARCHAR(20),
    sclass INT,
    sgender VARCHAR(10),
    sbirthday DATE
    );
Query OK, 0 rows affected (0.00 sec)
mysql> DESC db1.students;
     sid | int unsigned | NO | PRI | NULL | auto_increment |
sname | varchar(20) | YES | | NULL | |
| sclass | int | YES | | NULL | |

    sgender
    varchar(10)
    YES
    NULL
    |

    sbirthday
    date
    YES
    NULL
    |

    +-----+
    +----+
    +----+
    +----+

5 rows in set (0.00 sec)
```

2. After the table is created, on the memory acceleration page, create a mapping rule to convert each row in the **students** table into a Redis hash. The key of a hash is in the format of *Database name:Data table name:sid:*<*sid value>*. The selected fields are **sname**, **sclass**, **sgender**, and **sbirthday**.

Figure 4-220 Configuring a mapping rule

Rule Name students-to-hash										
Source Instance Config Source Instance Name	juration			Database Name				Database Name student	•	
Acceleration Instance O Acceleration Instance Name Database No. (0 to 999)	Configuration			Redis Key Pretx Enter a Rodis key prefix. TTL (s) Default value: 30 di 2,592,000	ays			Value Storage T Hash Key Delimiter	ype V	
Set Key Hash Set Value				0/1		Selected				0/4
C Enter a keyword.	Туре	Null	Key	Extra		C Enter a keyword.	Туре	Null	Key	Extra
sid	int unsigned	NO	PRI			sname sclass	varchar(32) varchar(32)	NO NO		
						sgender	varchar(32) varchar(32)	NO NO		

3. After a mapping rule is created, check the mapping rule and information.

#### Figure 4-221 Mapping information

Napping							C
Mapping Name 🖉 Mapping ID	đ	Source Instance ID	C Acceleration Instance II	0 0			
		o Tol	n configured	D Rareng			
Mapping Rule							
Create Mapping Rule							
Q. Enter a keyword.							00
Rule Name/D	Satus	Source Instance Database	Source Instance Table Name	Target Data Type	Target Database No.	Operation	
and the second s	O Available	60	student	HASH	1	Edit Dolete	
Total Recents: 1 11 v ( 1 )							

4. Insert a new data record to the **students** table in the TaurusDB instance. mysql> INSERT INTO db1.students (sname, sclass, sgender, sbirthday) VALUES ('zhangsan', 1, 'male', '2015-05-20');

Query OK, 1 row affected (0.01 sec)

mysql> SELECT \* FROM db1.students;



5. After the mapping rule is created, the data is automatically synchronized to the GeminiDB instance. Run commands in the GeminiDB instance to query the data.

127.0.0.1:6379> KEYS \* 1) "db1:students:sid:1"

127.0.0.1:6379> HGETALL db1:students:sid:1 1) "sbirthday" 2) "2015-05-20"

- 3) "sclass"
- 4) "1"
- 5) "sgender"
- 6) "male"
- 7) "sname"
- 8) "zhangsan"
- 6. Insert a new data record to the **students** table in the TaurusDB instance. mysql> INSERT INTO db1.students (sname, sclass, sgender, sbirthday) VALUES ('lisi', 10, 'male', '2015-05-22');

Query OK, 1 row affected (0.00 sec)

mysql> SELECT \* FROM db1.students;

+----+ | sid | sname | sclass | sgender | sbirthday | +----+ | 1 | zhangsan | 1 | male | 2015-05-20 | | 2 | lisi | 10 | male | 2015-05-22 | +----+

2 rows in set (0.00 sec)

- 7. Check whether the new data is synchronized to the GeminiDB instance.
  - 127.0.0.1:6379> KEYS \*
  - 1) "db1:students:sid:1"
  - 2) "db1:students:sid:2"

127.0.0.1:6379> HGETALL db1:students:sid:2

- 1) "sbirthday"
- 2) "2015-05-22"
- 3) "sclass"
- 4) "10"
- 5) "sgender"

- 6) "male" 7) "sname"
- Update data in the students table in the TaurusDB instance. mysql> UPDATE db1.students SET sclass=12, sname='wangwu' WHERE sid = 1; Query OK, 1 row affected (0.00 sec) Rows matched: 1 Changed: 1 Warnings: 0

9. Check whether the data is updated in the GeminiDB instance.

127.0.0.1:6379> KEYS \* 1) "db1:students:sid:1" 2) "db1:students:sid:2"

127.0.0.1:6379> HGETALL db1:students:sid:1

- 1) "sbirthday"
- 2) "2015-05-20"
- 3) "sclass"
- 4) "12" 5) "sgender"
- 6) "male"
- 7) "sname"
- 8) "wangwu"
- 10. Delete data from the **students** table in the TaurusDB instance. mysql> DELETE FROM db1.students WHERE sid = 1; Query OK, 1 row affected (0.00 sec)

mysql> SELECT \* FROM db1.students; +----+ | sid | sname | sclass | sgender | sbirthday | +----+ | 2 | lisi | 10 | male | 2015-05-22 | +----+ 1 row in set (0.00 sec)

11. Check whether the data is deleted from the GeminiDB instance.
127.0.0.1:6379> KEYS \*
1) "db1:students:sid:2"

## 4.16.2.3 Modifying and Deleting a Memory Acceleration Rule

A memory acceleration rule can automate data synchronization from TaurusDB to GeminiDB. You can also modify and delete this rule.

## Usage Notes

- Currently, only hash data from TaurusDB can be converted to GeminiDB Redis API.
- If a table name of the TaurusDB instance in the memory acceleration rule is changed, you need to reconfigure the rule.
- Currently, the ENUM, SET, and JSON data cannot be synchronized.
- If you rename or delete one or more key fields of a memory acceleration rule, the rule becomes invalid.

## Modifying a Mapping Rule

Step 1 Log in to the Huawei Cloud console.

- **Step 2** Click <sup>1</sup> in the upper left corner and select a region and project.
- **Step 3** Click = in the upper left corner of the page and choose **Databases** > **TaurusDB**.
- **Step 4** On the **Instances** page, click the instance name to go to the **Basic Information** page.
- **Step 5** In the navigation pane on the left, choose **Memory Acceleration**. In the **Mapping Rule** area, locate the target rule and click **Edit** in the **Operation** column.

Figure 4-222 The Edit button

and the second se							
Create Mapping Rule							
Q. Enter a topycard.							0
Rafe Nerro1D	\$120.25	Source Instance Database	Source Instance Table Name	Target Data Type	Target Database No.	Operation	
	O Available	80	student	HASH	0	Edit Delete	
Total Records: 1 19 🗸 < 4 >							

**Step 6** After editing the fields, click **Submit**.

Figure 4-223 Editing a mapping rule

Available					0/2	80	lected				1
C. Enter a keyword	d.						C Enter a keyword.				
Field	Type	Null	Key	Extra			Field	Туре	Null	Key	Extra
sid	int unsigned	NO	PRI				sname	varchar(32)	NO		
sclass	varchar(32)	NO				Solution	sgender	varchar(32)	NO		
							sbirthday	varchar(32)	NO		

----End

#### **Deleting a Mapping Rule**

Step 1 Log in to the Huawei Cloud console.

- **Step 2** Click <sup>(Q)</sup> in the upper left corner and select a region and project.
- **Step 3** Click = in the upper left corner of the page and choose **Databases** > **TaurusDB**.
- **Step 4** On the **Instances** page, click the target instance.
- **Step 5** In the navigation pane on the left, choose **Memory Acceleration**. In the **Mapping Rule** area, locate the target rule and click **Delete** in the **Operation** column.

----End

#### 4.16.2.4 Viewing and Removing Mappings

You can view the mapping list on the **Memory Acceleration Management** page and remove mappings.

## Usage Notes

- After a mapping is removed, service applications cannot obtain the latest data of the source database from the acceleration instance.
- The corresponding mapping rule will be cleared after a mapping is removed.
- If the source instance or acceleration instance is not normal, the mapping cannot be removed.

#### Querying the Mapping List

- 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 **Memory Acceleration Management**. On the displayed page, search for your target mapping by keyword (such as the mapping name or mapping ID).

#### Figure 4-224 Mapping list

r órfaut.				0
Status	Source Instance Name ID	Acceleration Instance Name/D	Created	Operation
C Available	and the second second	and the second	Jul 22, 2024 02:38:16 GNT+08:00	Remove
>				
270				
	o yohat Sata O Aatoo	n nonit Bana Soura-Mana New D O Autore Name	remont	rement rement Ratas Saras bases biene 0 Accientals interace bases 0 Censid O Autore ACCIENT of The ACCIENT o

----End

#### **Removing a Mapping**

- 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 **Memory Acceleration Management**. On the displayed page, locate the target mapping and click **Remove** in the **Operation** column.

Figure 4-225 Memory acceleration management

Q, Search by instance name by default.					
Mapping Name ID	Status	Source Instance Name/D	Acceleration Instance NameID	Created	Operation
	O Available		and the second second	JUI 22, 2024 02:30:16 GNT+00:00	Remove



#### Figure 4-226 Removing a mapping



----End

# **5** Development Reference

- 5.1 Development and O&M Rules
- 5.2 Compatible Commands
- 5.3 Examples of Connecting to an Instance Using Programming Languages
- 5.4 Lua Script Compilation Specifications
- 5.5 Compact Encoding for Small Objects
- 5.6 Keyspace Notification
- 5.7 ExHash Commands
- 5.8 Large Bitmap Initialization
- 5.9 Querying Large Bitmaps by Page
- 5.10 Configuring Parameters for a Client Connection Pool
- 5.11 Using Parallel SCAN to Accelerate Full Database Scanning
- 5.12 Accessing a GeminiDB Redis Instance Using a Pipeline
- 5.13 Processing Transactions on a GeminiDB Redis Instance
- 5.14 Retry Mechanism for Clients Accessing GeminiDB Redis API
- 5.15 GeminiDB Redis API Pub/Sub

## 5.1 Development and O&M Rules

This section describes usage rules of the GeminiDB Redis database based on Huawei's years of experience in cloud database development and O&M, so as to help you effectively evaluate and improve service system stability.

#### **Development Rules**

When developing a service program, you need to pay attention to the following development rules to prevent service instability caused by improper usage.

No.	Development Rules	Description
1	The service program must have a proper automatic reconnection mechanism.	In scenarios for the specification change, patch upgrade, HA switchover, network link jitter, or packet loss, the connection between the service program and the DB instance may be interrupted for a short period of time. The service program must support automatic reconnection. <b>NOTE</b> Use Jedis instead of Lettuce because Lettuce is not automatically reconnected after multiple requests time out.
2	Provide a connection pool and configure sufficient connections for a service program.	To prevent the program from failing to obtain connections when the number of concurrent requests increases sharply, you are advised to use a connection pool for the service program and configure proper connection pool parameters. For details about the recommended configuration of the connection pool of the client, see <b>5.10 Configuring Parameters for</b> <b>a Client Connection Pool</b> .
3	The service program must have a proper command retry mechanism for important operations.	When the connection is interrupted or the request times out, requests from the service program may fail to be executed within a short period of time. Therefore, a service fault tolerance mechanism is needed. Proper command retry intervals and times can ensure important data is successfully written or modified.
4	A correct HA connection address needs to be used to prevent services from being affected by a single point of failure.	When connecting a service program to a database through a private network, use a load balancer IP address to achieve HA. Do not directly connect the service program to an independent compute node. <b>NOTE</b> If a database is accessed from a public network, do not directly connect the service program to an independent compute node.

#### Table 5-1 Development rules of GeminiDB Redis

No.	Development Rules	Description
5	A connection pool needs to be used to avoid using a single connection or a large number of short connections.	A single connection has HA risks. The performance of short connections is poor, and short connections consume a large number of CPUs and network resources, which may cause bottlenecks. Therefore, you are advised to use mainstream SDK connection pools to connect a service program to GeminiDB Redis instances.
6	You are not allowed to run the <b>KEYS</b> command when there are more than 1,000 keys in an instance.	The <b>KEYS</b> command is a typical high-risk command. It obtains all data in the entire instance at a time and returns the data to the client. If there are a large number of keys in an instance, running the <b>KEYS</b> command will block requests or make the instance unavailable.
7	Do not set element values to excessively large or small values and ensure the number of elements contained in a single key is not too large.	The recommended maximum key size of the string data type is 10 KB. Fewer than 5,000 elements are recommended for keys of HASH, LIST, ZSET, or SET data types. The recommended value size of a single element is 1 KB or smaller. <b>NOTE</b> Similar to Redis Community Edition, GeminiDB Redis API does not strictly restrict storage of big keys. Therefore, you need to develop service programs according to related specifications

No.	Development Rules	Description
8	A single command does not contain many elements at a time. Too large network packets cannot be generated by a single command.	<ul> <li>It is recommended that no more than 1000 keys be concurrently operated using MSET or MGET.</li> <li>Make sure that no more than 1000 key elements to be concurrently operated using HMSET, HGETALL, LRANGE, ZADD, or ZRANGE.</li> <li>The limit parameter is not included in the Redis ZREMRANGEBYSCORE command, so the number of elements to be deleted at a time cannot be limited. You are advised to use ZRANGEBYSCORE (with limit) + ZREM instead.</li> <li>NOTE Similar to Redis Community Edition, GeminiDB Redis API does not strictly restrict access of big keys. Therefore, you need to develop service programs according to related specifications</li> </ul>
9	Properly distribute keys to avoid performance bottlenecks in hotspot keys or hash tags.	<ul> <li>Frequent access to a single key or a group of keys with the same hash tags may cause hot key problems, which may cause bottlenecks such as compute resource skew, request queuing, and slow response. Hot keys are usually created due to insufficient key splitting in service design. Therefore, service splitting needs to be optimized.</li> <li>NOTE Negative examples:</li> <li>Ultra-large global rankings are accessed frequently and intensively.</li> <li>Several hash keys are stored in an instance. Each key is used to store information of the entire table.</li> <li>For product inventory, there are only a small number of bucket hash tags for all keys. As a result, requests for querying hot hash tags may queue.</li> </ul>

No.	Development Rules	Description	
10	Run less than 100 packaging commands at a time using the pipeline.	Do not include a large number of commands in a pipeline. Using pipelines improperly may cause bottlenecks in CPU and bandwidth resources and block requests. <b>NOTE</b> Similar to Redis Community Edition, GeminiDB Redis API does not strictly restrict pipeline usage. Therefore, you need to develop service programs according to related specifications.	
11	Do not use time-consuming code in Lua scripts.	When using LUA scripts, excessive command execution times and time-consuming statements such as long-time sleep and large loop statements should be avoided.	
12	Do not pack too many commands in a transaction.	When using a transaction, do not pack too many commands or complex commands in a single transaction. If a transaction contains too many commands, requests are blocked, or the instance may become abnormal.	
13	Different data types cannot use the same key.	In Redis Community Edition, the same key name cannot be used for different data types. Although GeminiDB Redis API does not have mandatory restrictions on this, avoid using the same key for different data types to keep programs clear and easy to maintain.	
14	Exercise caution when running batch deletion commands.	Do not delete too much data (tens or hundreds of thousands of elements) of the LIST and ZSET types via a single batch deletion command (for example, <b>LREM</b> , <b>LTRIM</b> , and <b>ZREMRANGEBYSCORE</b> ). Deleting a large amount of data may take a long time, affecting other commands. Processes causing OOM may restart and the instance may be abnormal. In extreme scenarios, service processes may fail to be started repeatedly.	

No.	Development Rules	Description	
15	Cursors returned by a SCAN command must be parsed as 64-bit unsigned integers.	Similar to the open-source Redis, GeminiDB encodes cursors into 64-bit unsigned integers. Therefore, the returned cursors need to be parsed as 64-bit unsigned integers. Otherwise, a parsing error may occur. If GeminiDB receives these incorrect results, error message "invalid cursor" will be reported.	
16	The length of Redis inline commands must be less than 64 KB.	This rule is also applied for open- source Redis to avoid risks such as OOM. If the length of a command exceeds 64 KB, error message "ERR Protocol error: too big inline request" will be returned to a server.	
17	Avoid frequently running SUBSCRIBE commands and blocking commands like BRPOP to prevent the server from exhausting connection resources and failing to accept new connections.	A shared connection pool is maintained for communication between proxies and shards. Common commands reuse these connections. To execute blocking commands (such as BRPOP) and SUBSCRIBE commands, dedicated connections need to be established for proxies and shards. As a result, the number of connections per shard might exceed the maximum capacity of an individual process, and new connections cannot be established. Therefore, ensure that fewer than 1,000 client connections are established for a single proxy when running these commands.	

## **O&M** Rules

During routine O&M, you need to pay attention to the following O&M rules to prevent potential risks and master key emergency solutions.

No.	Rule	Description
1	Ensure that the phone number and email address bound to your Huawei Cloud account are valid so that you can receive notifications in a timely manner.	Huawei Cloud will send notifications to you through websites, emails, SMS messages, or internal messages in scenarios for changes, upgrades, and fault notifications. Ensure that the contact information bound to your account is valid.
2	Subscribe to major alarms.	<ul> <li>You can subscribe to alarms such as big key access, high storage utilization, high connection usage, and high CPU usage to detect and handle instance risks in a timely manner.</li> <li>For details, see 4.13.2 Configuring Alarm Rules.</li> </ul>
3	When the LB address is used for access, you need to configure access control instead of security groups.	A load balancer address does not support security groups. After instance creation is complete, configure IP address access control. If no whitelist is configured, all IP addresses that can communicate with the VPC can access the instance.
4	Reserve more than 50% of storage space to prevent instances from becoming read-only.	GeminiDB uses RocksDB as the storage engine. Data is periodically compacted to improve read performance. While data is compacted, temporary files are generated, occupying more storage space temporarily. The storage space usage varies depending on the service model, and it may increase by two times. It is relatively inexpensive to store data on disks. You are advised to reserve 50% of storage space.
5	Configure autoscaling for instances.	GeminiDB Redis instance storage can be automatically scaled up in case of a sudden surge in data volumes. Enable autoscaling by following <b>4.6.7.3 Automatically</b> Scaling Up Storage Space.

Table 5-2 O&M rules of GeminiDB Redis API

No.	Rule	Description
6	Keep the instance load healthy.	<ul> <li>If the service data volume is greater than 80% for a long time, you are advised to expand the capacity in a timely manner.</li> </ul>
		• If the service traffic exceeds the QPS limit of an instance or the CPU usage maintains 80% or more for a long time, upgrade the specifications or add nodes to prevent overloading and affecting service access.
		• When the instance computing power is overloaded due to the sharp increase of service traffic and the number of connections, you can add nodes to quickly improve the cluster computing power. Scaling up CPU cores is performed in rolling mode and takes a long time, which is not recommended in emergency scenarios.
7	Rename high-risk commands.	Disable or rename high-risk commands such as FLUSHALL, or KEYS to enhance instance security. For details, see <b>Renaming</b> <b>Commands</b> .
8	Exercise caution when performing the FLUSHALL operation during account management.	<ul> <li>When an IAM user with the read and write permissions executes FLUSHALL, all data on the instance is cleared.</li> <li>Do not perform FLUSHALL. You can perform FLUSHDB after confirming the database. Exercise caution when performing this operation.</li> </ul>
9	Perform periodical online diagnosis of big keys.	You can periodically view the big key diagnosis report of the instance on the console to view keys that are the most frequently accessed in the Redis database. For details, see <b>4.9.1 Big Key</b> <b>Diagnosis</b> .

No.	Rule	Description
10	Run the <b>dbsize</b> command after migration.	Running <b>DBSIZE</b> can ensure data consistency. For example, if there are no expired keys, you can run <b>DBSIZE</b> several minutes after data is imported to obtain the accurate number of keys and ensure data consistency.

## 5.2 Compatible Commands

GeminiDB Redis API provides cluster and primary/standby instances. It is compatible with Redis Community Edition 5.0 and earlier versions, Redis 6.2 (including  $6.2.\lambda$ ), and Redis 7.0 commands. You do not need to modify code when migrating applications to the cloud.

This section describes the compatibility of commands supported by GeminiDB Redis API 5.0, 6.2 (including 6.2.X), and 7.0, providing references to DBAs and developers.

## Basic GeminiDB Commands

The following table lists commands provided by community Redis and compatibility of GeminiDB Redis API 5.0, 6.2 (including 6.2.*X*), and 7.0.

Redi s Com man d Clas sific atio n	Descri ption	Compa tibility with Gemini DB Redis 5.0	Compat ibility with Gemini DB Redis API 6.2 (Includi ng 6.2. <i>x</i> )	Compatibility with GeminiDB Redis API 7.0
Strin g	String comm ands	100%	100%	100%
Hash	Hash comm ands	100%	100%	100%
List	List comm ands	100%	100%	100%

Redi s Com man d Clas sific atio n	Descri ption	Compa tibility with Gemini DB Redis 5.0	Compat ibility with Gemini DB Redis API 6.2 (Includi ng 6.2. <i>x</i> )	Compatibility with GeminiDB Redis API 7.0
Sort ed set	Sorte d set comm ands	100%	100%	100%
Set	Set comm ands	100%	100%	100%
Bitm ap	Bitma p comm ands	100%	100%	100%
Strea m	Strea m comm ands	100%	The XGROU P subcom mand createc onsume r is not support ed.	Features of version 7.0 are not supported.
GEO	GEO comm ands	100%	100%	100%
Hyp erLo gLog	Hyper LogLo g comm ands	100%	100%	100%
Pub/ Sub	Pub/S ub comm ands	100%	100%	100%
Lua scrip ts	Lua comm ands	100%	100%	Features of version 7.0 are not supported.

Redi s Com man d Clas sific atio n	Descri ption	Compa tibility with Gemini DB Redis 5.0	Compat ibility with Gemini DB Redis API 6.2 (Includi ng 6.2. <i>x</i> )	Compatibility with GeminiDB Redis API 7.0
Tran sacti ons	Trans action comm ands	100%	100%	100%
Gen eric	Gener ic key comm ands	SWAPD B and MOVE comma nds are not suppor ted.	The followin g comma nds are not support ed: SWAPD B and MOVE	100%

GeminiDB Redis API is compatible with some CLUSTER commands, including CLUSTER INFO, CLUSTER KEYSLOT, CLUSTER MYID, CLUSTER NODES, CLUSTER SLOTS, CLUSTER SHARDS and CLUSTER HELP.

## **Advanced GeminiDB Commands**

- ExHash
  - Application scenarios: GeminiDB allows you to set an expiration time for each field of a hash key and is suitable for services such as frequency control and shopping cart.
  - Command list: See **5.7 ExHash Commands**.
  - Usage: See 6.4 ExHash for Ad Frequency Control.
- Bloom filter
  - Functions: A Bloom filter enables you to check if an element is present in a large-size collection. It is applicable to scenarios such as web interceptors and anti-cache penetration.
  - Command list: See **Bloom filter**.
  - Usage: See **Bloom filter description**.

## GeminiDB O&M Commands

GeminiDB provides comprehensive and easy-to-use O&M functions based on the community Redis commands such as INFO, CLIENT, SLOWLOG, MONITOR, and CONFIG.

Function	Description	Redis Community Edition Command/ Capability
Metrics	QPS, average latency, and p99 latency of commands, various metrics of nodes, and aggregation metrics of instances are provided.	INFO
Instance sessions	Client IP addresses and the top sessions of an instance are displayed. Users can kill sessions in batches with a few clicks.	CLIENT
Parameter configuration	Kernel parameters can be queried and modified with only a few clicks. Enhancement of CONFIG command provided by Redis Community Edition.	CONFIG
Viewing slow query logs	Slow query logs and diagnosis information can be displayed.	SLOWLOG
User permission management	The account management function is supported.	ACL (Access Control List)
Viewing audit logs	High-risk commands and operations can be audited.	MONITOR
Real-time big key diagnosis	Big keys are monitored in real time and without affecting services.	Not supported
Real-time hot key diagnosis	Hot keys are monitored in real time and without affecting services.	Not supported
Real-time key prefix analysis	Distribution of key prefixes is monitored in real time, which does not affect services.	Not supported
Critical command renaming	Users can modify command aliases and view renaming records.	You can rename commands by modifying configuration files.

۱۹
1

Function	Description	Redis Community Edition Command/ Capability
Abnormal key circuit break	Specified keys can be shielded in one click to avoid access to services, which can be used for rapid data recovery.	Not supported

## Other O&M Commands

To help you manage instances, GeminiDB Redis API provides O&M commands in specific scenarios.

## 

Before running the following O&M commands, to avoid adverse impacts on services, you are advised to choose **Service Tickets > Create Service Ticket** in the upper right corner of the console and contact the customer service.

ROCKSCONF:

Syntax: ROCKSCONF SETOPTION/ SETDBOPTION *[Item]* [Value] Function:

- ROCKSCONF commands can adjust the RocksDB parameters of the storage engine to meet service requirements.
- *Item* indicates a parameter to be adjusted and *Value* its value. For details, see the RocksDB documentation.
- COMPACT:

Syntax: COMPACT [Type] [Partition ID]

Function:

- COMPACT commands can effectively reduce RocksDB tombstones, improving read performance while reducing storage space.
- *type* indicates the data type to be compacted. The value can be **string**, **hash**, **zset**, **set**, or **list**.
- *partition\_id* indicates a data shard for which COMPACT is to be executed. You can run **info partitions** to obtain shard information of the instance. For details, see **4.3.3.1 Connecting to an Instance Using a Load** Balancer Address (Recommended).

As shown in Figure 5-1, 179b5efc712a506c is a valid partition ID.

Figure	5-1	info	partitions
--------	-----	------	------------

127.0.0.1:80	635> info partitions				
# Partitions					
0	179b5efc712a506c				
1	2e0f4b2bfdd185d5				
2	34dbac8a70931ae1				
3	3f9a19937025e813				
4	4de6802ed5750903				
5	799d54fd3f336e87				
6	9670d213ced6281d				
7	ccf9c568bcd3d948				
8	d8d240b825327f85				
9	dd73b20598cc3886				
10	dee0fe0aaa1d2f58				
11	f3602d53c8af4527				

## Usage Notes

- GeminiDB Redis API does not support RESP3. Redis Serialization Protocol (RESP) is used for communication between a Redis server and a client. Mainstream clients such as Jedis use RESP2 by default, which can be used to access a GeminiDB Redis instance.
- GeminiDB Redis API 7.0 does not support functions.

## 5.3 Examples of Connecting to an Instance Using Programming Languages

## 5.3.1 Connecting to an Instance Using Jedis

This section describes how to access a GeminiDB Redis instance using the Java client, Jedis.

The proxy cluster architecture of GeminiDB Redis API provides a unified load balancing address and high availability. So, JedisPool is recommended for easy access.

JedisSentinelPool and JedisCluster can also be used to connect to GeminiDB Redis instances.

## Prerequisites

- A GeminiDB Redis instance has been created and is running properly. For details about how to create a GeminiDB Redis instance, see **4.2.1 Buying a GeminiDB Redis Cluster Instance**.
- An ECS is available. For details, see **Purchasing an ECS**.
- GNU Compiler Collection (GCC) has been installed on the ECS.
- The created ECS is in the same region, AZ, VPC, and security group as the GeminiDB Redis instance.

## Dependencies on the POM File

```
<dependency>
<groupId>redis.clients</groupId>
<artifactId>jedis</artifactId>
<version>4.3.2</version>
</dependency>
<dependency>
<groupId>org.springframework.boot</groupId>
<artifactId>spring-boot-starter-data-redis</artifactId>
<version>2.3.6.RELEASE</version>
</dependency>
```

## Using JedisPool for Access (Recommended)

#### Example code:

import org.apache.commons.pool2.impl.GenericObjectPoolConfig; import redis.clients.jedis.Jedis; import redis.clients.jedis.JedisPool;

public class JedisPoolTests {

private static void testPool() {

// 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.

```
String pwd = System.getenv("EXAMPLE_PASSWORD_ENV");
  JedisPool pool = new JedisPool(new GenericObjectPoolConfig(), "172.xx.xx.xx", 6379,
     2000, pwd);
  Jedis jedis = pool.getResource();
  try {
    System.out.println(jedis.hgetAll("676296"));
    System.out.println(jedis.set("key1", "value1"));
  } finally {
   jedis.close();
  }
  pool.destroy();
 }
 public static void main(String[] args) {
  testPool();
 }
}
```

• In the preceding code, 172.xx.xx indicates the load balancer IP address of the GeminiDB Redis instance that you want to connect to.

You can click the instance name to go to the **Basic Information** page and obtain the load balancer IP address in the **Network Information** area.

#### Figure 5-2 Viewing the load balancer IP address

Network Information					
VPC	default_vpc		Security Group	Sys-default 🖉	
Subnet	default_subnet(	)			
Load Balancer IP Address	8 🗗				

 6379 in the preceding code is the port of the instance to be connected. Replace it with the actual port number. For details about how to obtain the port number, see 4.3.5.2 Viewing the IP Address and Port Number of a GeminiDB Redis Instance.

- For details about the supported and restricted commands, see 5.1 Development and O&M Rules.
- Redis Cluster and GeminiDB Redis API use different hash algorithms. Adding hashtags to keys in some commands of GeminiDB Redis API can avoid unexpected exceptions. For details about how to use hashtags, see 5.1
   Development and O&M Rules.

## Using JedisCluster for Access

Example code:

```
import org.apache.commons.pool2.impl.GenericObjectPoolConfig;
import redis.clients.jedis.HostAndPort;
import redis.clients.jedis.JedisCluster;
public class ClusterTests {
    private static void testCluster() {
        String pwd = "a";
            JedisCluster cluster = new JedisCluster(new HostAndPort("172.xx.xx.xx", 6379),
            200, 2000, 5, pwd, new GenericObjectPoolConfig());
        System.out.println(cluster.hgetAll("676296"));
        System.out.println(cluster.set("key1", "value1"));
    }
    public static void main(String[] args) {
        testCluster();
    }
}
```

• In the preceding code, 172.xx.xx indicates the load balancer IP address of the GeminiDB Redis instance that you want to connect to.

You can click the instance name to go to the **Basic Information** page and obtain the load balancer IP address in the **Network Information** area.

Figure 5-3 Viewing the load balancer IP address

Network Information			
VPC	default_vpc	Security Group	Sys-default 🖉
Subnet	default_subnet()		
Load Balancer IP Address			

- 6379 in the preceding code is the port of the instance to be connected. Replace it with the actual port number. For details about how to obtain the port number, see 4.3.5.2 Viewing the IP Address and Port Number of a GeminiDB Redis Instance.
- For details about the supported and restricted commands, see 5.1
   Development and O&M Rules.
- Redis Cluster and GeminiDB Redis API use different hash algorithms. Adding hashtags to keys in some commands of GeminiDB Redis API can avoid unexpected exceptions. For details about how to use hashtags, see 5.1
   Development and O&M Rules.

#### Using JedisSentinelPool for Access

Example code:

```
import org.apache.commons.pool2.impl.GenericObjectPoolConfig;
import redis.clients.jedis.Jedis;
import redis.clients.jedis.JedisSentinelPool;
import java.util.HashSet;
import java.util.Set;
public void SentinelTest {
  public static void main(String[] args) {
     GenericObjectPoolConfig<Jedis> config = new GenericObjectPoolConfig<Jedis>();
     Set<String> mySentinels = new HashSet<String>();
     mvSentinels.add("172.xx.xx.6379");
     JedisSentinelPool pool = new JedisSentinelPool(master-name, mySentinels, config, 1000, password, 0);
     Jedis jedis = pool.getResource();
     jedis.auth(password);
     jedis.set("foo", "bar");
     String s = jedis.get("foo");
     System.out.println(s);
     jedis.close();
     pool.close();
  }
}
```

• In the preceding code, 172.xx.xx indicates the load balancer IP address of the GeminiDB Redis instance that you want to connect to.

You can click the instance name to go to the **Basic Information** page and obtain the load balancer IP address in the **Network Information** area.

Figure 5-4 Viewing the load balancer IP address

Network Information	n				
VPC	default_vpc		Security Group	Sys-default	l
Subnet	default_subnet(	)			
Load Balancer IP Address	19: ::8635 🗇				

- 6379 in the preceding code is the port of the instance to be connected. Replace it with the actual port number. For details about how to obtain the port number, see 4.3.5.2 Viewing the IP Address and Port Number of a GeminiDB Redis Instance.
- In the preceding code, master-name can only be set to mymaster.
- For details about the supported and restricted commands, see 5.1 Development and O&M Rules.
- Redis Cluster and GeminiDB Redis API use different hash algorithms. Adding hashtags to keys in some commands of GeminiDB Redis API can avoid unexpected exceptions. For details about how to use hashtags, see 5.1
   Development and O&M Rules.

## 5.3.2 Connecting to an Instance Using Redisson

This section describes how to connect to a GeminiDB Redis instance using Redisson in single-node or sentinel mode.

## Prerequisites

• A GeminiDB Redis instance has been created and is running properly. For details about how to create a GeminiDB Redis instance, see **4.2.1 Buying a GeminiDB Redis Cluster Instance**.

- An ECS is available. For details, see **Purchasing an ECS**.
- GNU Compiler Collection (GCC) has been installed on the ECS.
- The created ECS is in the same region, AZ, VPC, and security group as the GeminiDB Redis instance.

## SingleServer Mode

#### Example code:

import org.redisson.Redisson; import org.redisson.api.RedissonClient; import org.redisson.config.Config;

public static void main(String[] args) {
 testSingleServer();
}

}

• In the preceding code, 172.xx.xx indicates the load balancer IP address of the GeminiDB Redis instance that you want to connect to.

You can click the instance name to go to the **Basic Information** page and obtain the load balancer IP address in the **Network Information** area.

Figure 5-5 Viewing the load balancer IP address

Network Information					
VPC	default_vpc		Security Group	Sys-default 🖉	
Subnet	default_subnet(	)			
Load Balancer IP Address	8 0				

- 6379 in the preceding code is the port of the instance to be connected. Replace it with the actual port number. For details about how to obtain the port number, see 4.3.5.2 Viewing the IP Address and Port Number of a GeminiDB Redis Instance.
- For details about the supported and restricted commands, see 5.1 Development and O&M Rules.
- Redis Cluster and GeminiDB Redis API use different hash algorithms. Adding hashtags to keys in some commands of GeminiDB Redis API can avoid unexpected exceptions. For details about how to use hashtags, see 5.1
   Development and O&M Rules.

## Sentinel Mode

Example code:

```
import org.redisson.Redisson;
import org.redisson.api.RedissonClient;
import org.redisson.config.Config;
import static org.redisson.config.ReadMode.MASTER;
public class SingleServerTests {
 public static void testSentinel() {
  Config config = new Config();
  // 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.
  String password = System.getenv("EXAMPLE_PASSWORD_ENV");
  config.useSentinelServers()
       .setMasterName(master_name)
        .setCheckSentinelsList(false)
        .setReadMode(MASTER)
        .setPassword(password)
        .addSentinelAddress("redis://172.xx.xx.6379");
  RedissonClient redisson = Redisson.create(config);
  execute(redisson);
                      // send requests to database
  redisson.shutdown();
 public static void main(String[] args) {
  testSentinel();
 }
}
```

• In the preceding code, 172.xx.xx indicates the load balancer IP address of the GeminiDB Redis instance that you want to connect to.

You can click the instance name to go to the **Basic Information** page and obtain the load balancer IP address in the **Network Information** area.

Figure 5-6 Viewing the load balancer IP address

- 6379 in the preceding code is the port of the instance to be connected. Replace it with the actual port number. For details about how to obtain the port number, see 4.3.5.2 Viewing the IP Address and Port Number of a GeminiDB Redis Instance.
- The Sentinel mode is only used for connection and its native availability is not used. In the code example, master\_name is fixed to mymaster.
   CheckSentinelsList must be set to false and ReadMode must be set to MASTER.
- For details about the supported and restricted commands, see 5.1 Development and O&M Rules.
- Redis Cluster and GeminiDB Redis API use different hash algorithms. Adding hashtags to keys in some commands of GeminiDB Redis API can avoid

unexpected exceptions. For details about how to use hashtags, see **5.1 Development and O&M Rules**.

#### **ClusterServer Mode**

#### Example code:

```
import org.redisson.Redisson;
import org.redisson.api.RedissonClient;
import org.redisson.config.Config;
public class ClusterServerTests {
 private static void testClusterServer() {
  Config config = new Config();
  config.useClusterServers()
          .addNodeAddress("redis://172.xx.xx.6379")
           .setPassword(password);
  RedissonClient redisson = Redisson.create(config);
  execute(redisson); // send requests to database
  redisson.shutdown();
 }
 public static void main(String[] args) {
  testClusterServer();
 }
}
```

• In the preceding code, 172.*xx.xx* indicates the load balancer IP address of the GeminiDB Redis instance that you want to connect to.

You can click the instance name to go to the **Basic Information** page and obtain the load balancer IP address in the **Network Information** area.

#### Figure 5-7 Viewing the load balancer IP address

Network Information	1		
VPC	default_vpc	Security Group	Sys-default 🖉
Subnet	default_subnet(	)	
Load Balancer IP Address	19: ::8635 🗇		

- 6379 in the preceding code is the port of the instance to be connected. Replace it with the actual port number. For details about how to obtain the port number, see 4.3.5.2 Viewing the IP Address and Port Number of a GeminiDB Redis Instance.
- For details about the supported and restricted commands, see 5.1 Development and O&M Rules.
- Redis Cluster and GeminiDB Redis API use different hash algorithms. Adding hashtags to keys in some commands of GeminiDB Redis API can avoid unexpected exceptions. For details about how to use hashtags, see 5.1
   Development and O&M Rules.

## 5.3.3 Connecting to an Instance Using Hiredis

This section describes how to use hiredis to access a GeminiDB Redis instance.

## Prerequisites

- A GeminiDB Redis instance has been created and is running properly. For details about how to create a GeminiDB Redis instance, see **4.2.1 Buying a GeminiDB Redis Cluster Instance**.
- An ECS is available. For details, see **Purchasing an ECS**.
- GNU Compiler Collection (GCC) has been installed on the ECS.
- The created ECS is in the same region, AZ, VPC, and security group as the GeminiDB Redis instance.

#### Procedure

- **Step 1** Obtain the load balancer IP address and port of the GeminiDB Redis instance that you want to access.
  - For how to obtain the load balancer IP address, see Viewing the Load Balancer IP Address and Port.
  - For how to obtain the port, see Viewing the Port for Accessing Each Instance Node.
- **Step 2** Log in to the ECS. For details, see **Logging In to an ECS** in *Getting Started with Elastic Cloud Server*.
- **Step 3** Run the following command to download and decompress the hiredis package.

#### wget https://github.com/redis/hiredis/archive/master.zip

**Step 4** Go to the directory where the decompressed hiredis package is saved, and compile and install hiredis.

make

#### make install

**Step 5** Write the test code connRedisTst.cc.

**NOTE** 

For details about how to use hiredis, see the usage description on the **Redis official website**.

#### The code is as follows:

```
#include <stdio.h>
#include <stdlib.h>
#include <strina.h>
#include <hiredis.h>
int main(int argc, char **argv) {
   unsigned int j;
   redisContext *conn;
   redisReply *reply;
   if (argc < 3) {
        printf("Usage: example {instance_ip_address} 6379 {password}\n");
         exit(0);
   }
   const char *hostname = argv[1];
   const int port = atoi(argv[2]);
   const char *password = argv[3];
   struct timeval timeout = { 1, 500000 }; // 1.5 seconds
   conn = redisConnectWithTimeout(hostname, port, timeout);
   if (conn == NULL || conn->err) {
         if (conn) {
```

```
printf("Connection error: %s\n", conn->errstr);
     redisFree(conn);
      } else {
     printf("Connection error: can't allocate redis context\n");
exit(1);
/* AUTH */
reply = redisCommand(conn, "AUTH %s", password);
printf("AUTH: %s\n", reply->str);
freeReplyObject(reply);
/* Set */
reply = redisCommand(conn,"SET %s %s", "key", "hiredis test ok!");
printf("SET: %s\n", reply->str);
freeReplyObject(reply);
/* Get */
reply = redisCommand(conn,"GET key");
printf("GET key: %s\n", reply->str);
freeReplyObject(reply);
/* Disconnects and frees the context */
redisFree(conn);
return 0;
```

**Step 6** Run the following command to perform compilation:

#### gcc connRedis.c -o connRedis -I /usr/local/include/hiredis -lhiredis

If an error is reported, locate the directory where the **hiredis.h** file is stored and modify the compile command.

After the compilation, an executable **connRedis** file is obtained.

Step 7 Run the following command to connect to the instance.

#### export LD\_LIBRARY\_PATH=/usr/local/lib/:\$LD\_LIBRARY\_PATH

./connRedis <redis\_ip\_address> 6379 <password>

Replace the following information based on the site requirements:

- <redis\_ip\_address> indicates the load balancer IP address obtained in Step 1.
- **6379** is the port number of the GeminiDB Redis instance.
- *<password>* indicates the password set when the instance is created.
- **Step 8** If the following information is displayed, the instance is successfully connected.

```
AUTH: OK
SET: OK
GET key: Hello, hiredis test ok!
```

----End

## 5.3.4 Connecting to an Instance Using Node.js

This section describes how to use Node.js to access a GeminiDB Redis instance.

#### Prerequisites

- A GeminiDB Redis instance has been created and is in the **Available** status.
- An ECS is available. For details, see Purchasing an ECS.

- If the Linux operating system is used, ensure that compilation tools such as GCC have been installed on the ECS.
- The created ECS is in the same region, AZ, VPC, and security group as the GeminiDB Redis instance.

## Procedure

- **Step 1** Obtain the load balancer IP address and port of the GeminiDB Redis instance that you want to access.
  - For how to obtain the load balancer IP address, see Viewing the Load Balancer IP Address and Port.
  - For how to obtain the port, see Viewing the Port for Accessing Each Instance Node.
- **Step 2** Log in to the ECS. For details, see **Logging In to an ECS** in *Getting Started with Elastic Cloud Server*.
- **Step 3** Run the following command to install Node.js:
  - Method 1: Run the following command to install Node.js: yum install nodejs

#### **NOTE**

CentOS (Red Hat series) is used as an example. If Ubuntu (Debian series) is used, run the corresponding installation command.

Method 2: If the method 1 fails, use the following method to install it.
 wget https://nodejs.org/dist/v0.12.4/node-v0.12.4.tar.gz --no-check-certificate;

tar -xvf node-v0.12.4.tar.gz;

cd node-v0.12.4;

./configure;

make;

#### make install;

#### **NOTE**

CentOS (Red Hat series) is used as an example. If Ubuntu (Debian series) is used, run the corresponding installation command.

**Step 4** After the Node.js is installed, run the following command to check the version number and ensure that the Node.js is successfully installed.

node -v

**Step 5** Install the JS package management tool npm.

#### yum install npm

Step 6 Install the Node.js Redis client ioredis.

#### npm install ioredis

- **Step 7** Edit the sample script for connecting to the instance.
  - Connect to a GeminiDB Redis cluster using the SDK for the single-node API on Node.js.

```
var Redis = require('ioredis');
// 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.
var pwd = process.env.EXAMPLE_PASSWORD_ENV;
var redis = new Redis({
 port: 6379,
                  //Port number of the GeminiDB Redis instance obtained in step 1
 host: '192.xx.xx.xx', //Enter the load balancer IP address obtained in step 1.
 family: 4,
                 //4 indicates IPv4, and the 6 indicates IPv6.
 password: pwd,
 db: 0
});
redis.set('key', 'Nodejs tst ok!');
redis.get('key', function (err, result) {
console.log(result);
});
```

 Connect to a GeminiDB Redis cluster using the SDK of the cluster API on Node.is.

```
const Redis = require("ioredis");
// 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.
var pwd = process.env.EXAMPLE_PASSWORD_ENV;
const cluster = new Redis.Cluster([
  {
                     //Port number of the GeminiDB Redis instance obtained in step 1
    port: 6379,
 host: '192.xx.xx.xx', //Enter the load balancer IP address obtained in step 1.
                  // 4 indicates IPv4, and the 6 indicates IPv6.
   family: 4.
   password: pwd,
   db: 0
 },
]);
cluster.set("foo", "nodejs is ok!");
cluster.get("foo", (err, res) => {
 console.log(res);
});
```

**Step 8** Run the sample script and verify that the result is normal.

#### node ioredisdemo.js

----End

## 5.3.5 Connecting to an Instance Using PHP

This section describes how to use PHP to access a GeminiDB Redis instance.

## Prerequisites

- A GeminiDB Redis instance has been created and is in the **Available** status.
- An ECS is available. For details, see **Purchasing an ECS**.
- GNU Compiler Collection (GCC) has been installed on the ECS.
- The created ECS is in the same region, AZ, VPC, and security group as the GeminiDB Redis instance.

## Procedure

- **Step 1** Obtain the load balancer IP address and port of the GeminiDB Redis instance that you want to access.
  - For how to obtain the load balancer IP address, see Viewing the Load Balancer IP Address and Port.
  - For how to obtain the port, see Viewing the Port for Accessing Each Instance Node.
- **Step 2** Log in to the ECS. For details, see **Logging In to an ECS** in *Getting Started with Elastic Cloud Server*.
- **Step 3** Install the PHP development kit and command line tool.

Run the following yum command to install the PHP development package:

#### yum install php-devel php-common php-cli

#### D NOTE

CentOS (Red Hat series) is used as an example. If Ubuntu (Debian series) is used, run the corresponding installation command.

**Step 4** After the installation is complete, check the version number to ensure that the installation is successful.

#### php --version

- **Step 5** Install the PHP client of Redis.
  - Run the following command to download the source phpredis package: wget http://pecl.php.net/get/redis-4.1.0RC3.tgz

#### D NOTE

The preceding clients are of the latest version. You can download the phpredis client of other versions from the **PHP official website**.

2. Run the following commands to decompress the source phpredis package:

#### tar -zxvf redis-4.1.0RC3.tgz

#### cd redis-4.1.0RC3

- Run the following extension command before compilation: phpize
- 4. Run the following command to configure the php-config file:

./configure --with-php-config=/usr/bin/php-config

#### **NOTE**

The PHP installation method and location depend on the operating system. Before the configuration, run the **find / -name php.ini** command to check the directory of the file.

5. Run the following command to compile and install the phpredis client:

#### make && make install

6. After the installation, add the extension configuration in the **php.ini** file to reference the Redis module.

Run the following command to find the **php.ini** file:

#### vim /usr/local/php/etc/php.ini

Add the following configuration item to the php.ini file:

extension = "/usr/lib64/php/modules/redis.so"

#### **NOTE**

The directories of the **php.ini** and **redis.so** files may be different. You can run the following command to query the directories.

find / -name php.ini

- find / -name redis.so
- 7. Save the configuration and exit. Then, run the following command to check whether the extension takes effect:

#### php -m |grep redis

If redis is returned, the PHP Redis client environment has been set up.

#### **Step 6** Use the phpredis client to connect to the instance.

- 1. Compile the test code redis.php.
  - Connect to a GeminiDB Redis cluster using the SDK of the single-node API on PHP.

<?php // 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.
```

\$pwd =getenv('EXAMPLE\_PASSWORD\_ENV');

```
$redis_host = "192.xx.xx.xx"; //Enter the load balancer IP address obtained in step 1.
```

```
$redis_port = 6379;
```

```
$user_pwd = pwd;
```

```
$redis = new Redis();
```

if (\$redis->connect(\$redis\_host, \$redis\_port) == false) {

```
die($redis->getLastError());
}
```

```
if ($redis->auth($user_pwd) == false) {
    die($redis->getLastError());
```

l ule(\$reuis->ge

if (\$redis->set("key", "php test ok!") == false) {

```
die($redis->getLastError());
```

```
$
$value = $redis->get("key");
echo $value;
$redis->close();
```

```
ייע
כי
```

## Connect to the GeminiDB Redis cluster using the SDK of the cluster API on PHP.

```
<?php
```

```
$redis_host = "192.xx.xxx"; //Enter the load balancer IP address obtained in step 1.
$redis_port = 6379;
$user_pwd = "pwd";
// Connect with read/write timeout as well as specify that phpredis should use
// persistent connections to each node.
$redis = new RedisCluster(NULL, Array("$redis_host:$redis_port"), 1.5, 1.5, true,
$user_pwd);
if ($redis->set("key", "php test ok!") == false) {
    die($redis->getLastError());
    }
$value = $redis->get("key");
    echo $value;
    $redis->close();
}
```

2. Run the redis.php command to check whether the result is normal.

----End

## 5.3.6 Connecting to an Instance Using Python

This section describes how to use Python to access a GeminiDB Redis instance.

#### Prerequisites

- A GeminiDB Redis instance has been created and is in the **Available** status.
- An ECS is available. For details, see **Purchasing an ECS**.
- GNU Compiler Collection (GCC) has been installed on the ECS.
- The created ECS is in the same region, AZ, VPC, and security group as the GeminiDB Redis instance.

#### Procedure

- **Step 1** Obtain the load balancer IP address and port of the GeminiDB Redis instance that you want to access.
  - For how to obtain the load balancer IP address, see Viewing the Load Balancer IP Address and Port.
  - For how to obtain the port, see Viewing the Port for Accessing Each Instance Node.
- **Step 2** Log in to the ECS. For details, see **Logging In to an ECS** in *Getting Started with Elastic Cloud Server*.
- **Step 3** Install the Python client Redis-py of Python and Redis.
  - 1. If the system does not provide Python, you can use yum to install it. **yum install python**
  - 2. Run the following command to download and decompress the redis-py package:

#### wget https://github.com/andymccurdy/redis-py/archive/master.zip

3. Go to the decompression directory and install the Python client Redis-py of Redis.

#### python setup.py install

 After the installation, run the python command. If the following information is displayed, Redis-py is successfully installed: Python 2.6.6 (r266:84292, Aug 18 2016, 15:13:37) [GCC 4.4.7 20120313 (Red Hat 4.4.7-17)] on linux2 Type "help", "copyright", "credits" or "license" for more information. >>> import redis >>>

**Step 4** Use the Redis-py client to connect to the instance.

In the following steps, commands are executed in CLI mode. (Alternatively, write the commands into a Python script and then execute the script.)

- Connect to the GeminiDB Redis cluster using the single-node API.
  - a. Run the **python** command to enter the CLI mode.

You have entered CLI mode if the following command output is displayed: Python 2.6.6 (r266:84292, Aug 18 2016, 15:13:37) [GCC 4.4.7 20120313 (Red Hat 4.4.7-17)] on linux2

Type "help", "copyright", "credits" or "license" for more information. >>> import redis

b. Run the following command in the CLI to check whether the result is normal.

>>> r = redis.StrictRedis(host='192.xx.xx.x', port=6379, password='pwd'); >>> r.set('key', 'Python tst ok!') True >>> r.get('key') 'Python tst ok!'

D NOTE

Modify the following information based on service requirements before running the preceding command.

- In the preceding command, host and port indicate the load balancer IP address and corresponding port of the GeminiDB Redis instance obtained in Step 1.
- **password** indicates the password of the instance.
- Connect to the GeminiDB Redis cluster using the cluster API.

```
Configure config set CompatibleMode ClusterClient first.
```

Python 3.7.4 (default, Jan 30 2021, 09:00:44) [GCC 7.3.0] on linux Type "help", "copyright", "credits" or "license" for more information. >>> from redis.cluster import RedisCluster as Redis >>> rc = Redis(host='127.0.0.1', port=6379, password='a') >>> rc.set('key', 'Python test ok!') True >>> rc.get('key') b'Python test ok!'

----End

## 5.3.7 Connecting to an Instance Using Go

This section describes how to use Go to access a GeminiDB Redis instance.

## Prerequisites

- A GeminiDB Redis instance has been created and is in the **Available** status.
- An ECS is available. For details, see **Purchasing an ECS**.
- GNU Compiler Collection (GCC) has been installed on the ECS.
- The created ECS is in the same region, AZ, VPC, and security group as the GeminiDB Redis instance.

#### Procedure

- **Step 1** Obtain the load balancer IP address and port of the GeminiDB Redis instance that you want to access.
  - For how to obtain the load balancer IP address, see Viewing the Load Balancer IP Address and Port.

- For how to obtain the port, see Viewing the Port for Accessing Each Instance Node.
- **Step 2** Log in to the ECS. For details, see **Logging In to an ECS** in *Getting Started with Elastic Cloud Server*.
- **Step 3** Use the Golang client to connect to the instance. The following uses the go-redis SDK as an example.
  - Go-redis download address: https://github.com/go-redis/redis

```
Connect to the GeminiDB Redis cluster using the single-node API.
package main
import (
  "fmt"
  "github.com/go-redis/redis"
  "os"
func main() {
  // 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.
  password = os.Getenv("EXAMPLE_PASSWORD_ENV")
  client := redis.NewClient(&redis.Options{
     Addr: "xx.xx.xx.6379", //Enter the load balancer IP address obtained in step 1.
     Password: password,
            0, // Use the default database 0.
     DB:
  })
  pong, err := client.Ping().Result()
  fmt.Println(pong, err)
  err = client.Set("key1", "value1", 0).Err()
  if err != nil {
     panic(err)
  val, err := client.Get("key1").Result()
  if err != nil {
     panic(err)
  fmt.Println("key1", val)
}
```

The expected output is as follows:

PONG key1 value1

#### **NOTE**

- If you use go-redis to connect a GeminiDB Redis instance, use the common mode instead of the cluster mode, as shown in the preceding sample code.
- In the preceding example, set the GeminiDB Redis address and password based on the site requirements.

#### Connect to the GeminiDB Redis cluster using the cluster API.

```
package main
import (
    "fmt"
    "github.com/go-redis/redis"
)
func main() {
    client := redis.NewClusterClient(&redis.ClusterOptions{
        Addrs: []string{ // Enter the load balancer IP address obtained in step 1.
        " xx.xx.xx.i6379",
        },
        Password: "xx", // Password of the cluster.
    })
```
```
pong, err := client.Ping().Result()
fmt.Println(pong, err)
err = client.Set("key1", "value1", 0).Err()
if err != nil {
    panic(err)
}
val, err := client.Get("key1").Result()
if err != nil {
    panic(err)
}
fmt.Println("key1", val)
```

----End

# 5.3.8 Connecting to an Instance Using C#

This section describes how to use C# to access a GeminiDB Redis instance.

#### Prerequisites

- A GeminiDB Redis instance has been created and is in the **Available** status.
- An ECS is available. For details, see Purchasing an ECS.
- GNU Compiler Collection (GCC) has been installed on the ECS.
- The created ECS is in the same region, AZ, VPC, and security group as the GeminiDB Redis instance.

#### Procedure

- **Step 1** Obtain the load balancer IP address and port of the GeminiDB Redis instance that you want to access.
  - For how to obtain the load balancer IP address, see Viewing the Load Balancer IP Address and Port.
  - For how to obtain the port, see Viewing the Port for Accessing Each Instance Node.
  - For details about how to view the IP address of each instance, see Viewing the Private IP Address or EIP.
- **Step 2** Log in to the ECS. For details, see **Logging In to an ECS** in *Getting Started with Elastic Cloud Server*.
- Step 3 Install .Net. For a Windows host, click here to download .NET. For a Linux host, you need to install .NET Core key and repository, and then install the .NET runtime and SDK.

sudo rpm -Uvh https://packages.microsoft.com/config/centos/8/packages-microsoft-prod.rpm sudo yum install dotnet-sdk-7.0 sudo yum install dotnet-runtime-7.0

Run the following code.

dotnet --version

You'll see your .Net version information.

- **Step 4** Use the StackExchange.Redis client to connect to the GeminiDB Redis instance.
  - Creating a project

Run the following command to create a C# console application or create a new C# console application in Visual Studio.

dotnet new console -o redisdemo

 Installing the StackExchange.Redis package of the C# client of the Redis. In Visual Studio, you can install StackExchange.Redis from the NuGet package manager. Run the following command in the command line window where the dotnet project is located: dotnet add package StackExchange.Redis

```
    Connecting to GeminiDB Redis in single-node mode
using System;
using StackExchange.Redis;
namespace redisdemo
```

```
class Program
  {
     static void Main(string[] args)
     {
      // Creates a ConnectionMultiplexer object that connects to the Redis server.
        string redisConnectionString = "192.xx.xx.xs:6379"; // Load balancer address obtained in step 1
        ConfigurationOptions options = ConfigurationOptions.Parse(redisConnectionString);
      // There will be security risks if the username and password used for authentication are directly
written into code. You are advised to store them in the configuration file or environment variables.
The password must be stored in ciphertext and decrypted when used.
         // In this example, the username and password are stored in the environment variables.
Before running the example commands, set environment variables EXAMPLE_USERNAME_ENV and
EXAMPLE_PASSWORD_ENV as needed.
        string password = Environment.GetEnvironmentVariable("EXAMPLE_PASSWORD_ENV");
        options.Password = password;
        ConnectionMultiplexer redis = ConnectionMultiplexer.Connect(options);
        // Obtains Redis database objects.
        IDatabase redisDb = redis.GetDatabase();
        //Sets a key-value pair.
        string key = "mykey";
        string value = "myvalue";
        redisDb.StringSet(key, value);
        string valueGet = redisDb.StringGet(key);
       Console.WriteLine ($"The value of {key} is {valueGet}.");
     }
  }
}
```

#### Expected output:

The value of **mykey** is **myvalue**.

Connecting to the GeminiDB Redis cluster in cluster mode

```
using System;
using StackExchange.Redis;
namespace redisdemo
{
  class Program
  {
     static void Main(string[] args)
     {
          ConfigurationOptions options = new ConfigurationOptions();
          options.EndPoints.Add("192.xx.xx:6379"); // IP address and port of node 1 in the
instance cluster obtained in step 1
         options.EndPoints.Add("192.xx.xx:6379"); // IP address and port of node 2 in the instance
cluster obtained in step 1
          options.Password = "your_password"; // Sets the password.
          ConnectionMultiplexer redis = ConnectionMultiplexer.Connect(options);
        // Obtains Redis database objects.
          IDatabase redisDb = redis.GetDatabase();
        //Sets a key-value pair.
```

```
string key = "mykey";
string value = "myvalue";
redisDb.StringSet(key, value);
string valueGet = redisDb.StringGet(key);
Console.WriteLine ($"The value of {key} is {valueGet}.");
}
}
```

Expected output:

The value of **mykey** is **myvalue**.

----End

# 5.3.9 Connecting to an Instance Using Sentinel

GeminiDB Redis API uses an in-house HA component and does not depend on Sentinel. To reduce code modifications and improve instance compatibility, GeminiDB Redis API is compatible with Redis Sentinel. After the Sentinel mode is enabled, you can connect to a GeminiDB Redis instance to Redis Sentinel.

#### Prerequisites

- A cluster or primary/standby GeminiDB Redis instance has been created and is in the **Available** state.
- An ECS is available. For details, see **Purchasing an ECS**.
- GNU Compiler Collection (GCC) has been installed on the ECS.
- The created ECS is in the same region, AZ, VPC, and security group as the GeminiDB Redis instance.
- To connect to a DB instance in Sentinel mode, you must enable the Sentinel compatibility mode first.

## Enabling the Sentinel Compatibility Mode

- 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 specifications you want to change. The **Basic Information** page is displayed.
- **Step 4** In the navigation pane on the left, choose **Parameters**.
- **Step 5** Change the value of **CompatibleMode** and click **Save**.
  - For a cluster DB instance, change the value of **CompatibleMode** to **3**.
  - For a primary/standby DB instance, change the value of CompatibleMode to 2.

| Basic Information                      |                             |                          |           |                    |   |
|--|-----------------------------|--------------------------|-----------|--------------------|---|
| Backups & Restorations                 | Parameters Change History   |                          |           |                    |   |
| Accounts                               | Save Cancel Preview         | opart Compare            |           |                    | Enter a parameter name. Q   |
| Slow Query Logs                        | Parameter Name              | Effective upon Restart 👙 | Value     | Allowed Values     | Description   |
| Parameters<br>Metrics                  | AuthFailLockTime            | No                       | 5         | 0-10,000           | The length of time, in second, that a suspicious IP ad $\! \ldots \!$ |
| Sessions                               | BigbeysGuantityLimitation   | No                       | 100       | 1-10,000           | string hash/list/zeel/sel/exhash/stream type of large k               |
| Diagnosis Analysis<br>Rename High-risk | CompatibleMode              | No                       | 3 *       | 0, 1, 2, 3         | Whether StackExchange Redis is available. Set this                    |
| Command                                | EnableAcIDbDirect           | No                       | no v      | yes, no            | is the DB direct function enabled. The default is false.              |
|  | MaxAuthFailTimes            | No                       | 5         | 0-10,000           | Maximum failed access attempts. When this limit is re                 |
|  | ProxyTimeout                | No                       | 0         | 0-100,000          | The length of time, in seconds, that a proxy-client con               |
|  | bigkeys-composite-threshold | No                       | 1024      | 1-2,147,483,647    | A key of the hashilistizsetiselistream type whose nu                  |
|  | bigkeys-string-threshold    | No                       | 102400    | 1-2,147,483,647    | If the value is greater than the value of a string key, th            |
|  | databases                   | No                       | 1000      | 1-1,000            | Allow a limit on the number of supported DBs.                         |
|  | maxmemory-policy            | Yes                      | nomiction | noeviction         | Whether new keys can be saved when the storage sp                     |
|  | nolify-keyspace-events      | No                       |           | -                  | The type of event that needs to be monitored. The de                  |
|  | slowlog-threshold           | No                       | 300000    | 80,000-100,000,000 | Maximum time in microseconds for executing a query                    |

#### Figure 5-8 Changing parameters

----End

#### Connecting to a DB Instance in Sentinel Mode

This section uses Java as an example to describe how to access a GeminiDB Redis instance through the open-source libraries Redisson and Jedis.

#### **Redisson Code Example**

```
import org.redisson.Redisson;
import org.redisson.api.RedissonClient;
import org.redisson.config.Config;
import static org.redisson.config.ReadMode.MASTER;
public class SingleServerTests {
 public static void testSentinel() {
  Config config = new Config();
  // 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.
  String password = System.getenv("EXAMPLE_PASSWORD_ENV");
  config.useSentinelServers()
        .setMasterName("mymaster")
        .setCheckSentinelsList(false)
        .setReadMode(ReadMode.MASTER)
        .setPassword(password)
        .addSentinelAddress("redis://172.xx.xx.6379");
  RedissonClient redisson = Redisson.create(config);
  execute(redisson);
                        // send requests to database
  redisson.shutdown();
 public static void main(String[] args) {
  testSentinel();
 }
```

#### Jedis Code Example

import java.util.HashSet; import java.util.Set; import org.apache.commons.pool2.impl.GenericObjectPoolConfig; import redis.clients.jedis.Jedis; import redis.clients.jedis.JedisSentinelPool;

```
public class TestJedisSentinel {
public static void main(String[] args) {
 Set<String> sentinels = new HashSet<>();
 sentinels.add("192.xx.xx.6379");
 GenericObjectPoolConfig<Jedis> poolConfig = new GenericObjectPoolConfig<>();
 poolConfig.setMaxIdle(100);
 poolConfig.setMaxWaitMillis(10000);
 poolConfig.setTestOnBorrow(true);
 int connectionTimeout = 5000;
 int soTimeout = 5000;
 int database = 0:
         // 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.
```

```
String password = System.getenv("EXAMPLE_PASSWORD_ENV");
try (JedisSentinelPool jspool = new JedisSentinelPool("mymaster", sentinels, poolConfig,
connectionTimeout, soTimeout, password, database)) {
Jedis jedis = jspool.getResource();
jedis.mset("testkey", "AAA", "b", "BBB");
} catch (Exception e) {
e.printStackTrace();
}
}
```

- MasterName: fixed character string mymaster.
- CheckSentinelsList: The value must be false.
- ReadMode: **ReadMode.MASTER** is used.
- Password: password of the instance
- SentinelAddress: load balancer address of the GeminiDB Redis instance. Replace it with the actual IP address and port number.

You can click the instance name to go to the **Basic Information** page and obtain the load balancer IP address in the **Connection Information** area.

Figure 5-9 Viewing the load balancer IP address

| Connection Information |                         |                                       |   |
|------------------------|-------------------------|---------------------------------------|---|
| Load Balancer Address  | Database Port<br>6379 🖋 | Maximum Connections<br>20000 Sessions | Password-Free Access<br>Disabled Enable |
| Access Control         | SSL ف<br>Disabled ف     |                                       |   |

The Sentinel mode is only used for connection and its native availability is not used. In the code example, **master\_name** is fixed to **mymaster**. **CheckSentinelsList** must be set to **false** and **ReadMode** must be set to **MASTER**.

# 5.4 Lua Script Compilation Specifications

Lua is a scripting language designed to be embedded in applications to provide flexible extension and customization functions for applications. GeminiDB Redis API uses Lua 5.1.5, which is the same as the Lua version used by the open-source Redis 5.0.

#### D NOTE

When you use Lua scripts, make sure to perform a careful verification. Otherwise, infinite loops, request timeouts, or other exceptions may occur, or even services may become unavailable.

### **Differences from Open-Source Redis Lua**

#### 1. EVAL/EVALSHA

Example command:

EVAL script numkeys key [key ...] arg [arg ...]

#### EVALSHA sha1 numkeys key [key ...] arg [arg ...]

You can use the preceding commands the same as you do in open-source Redis. Ensure that the Redis key used in the script is explicitly transferred through the key instead of being directly encoded in the script.

If multiple keys are specified for a cluster instance, all keys must have the same hashtag.

If the preceding constraints are not complied with, error messages may be returned and data consistency may be damaged when Redis operations involving these keys are performed in Lua.

#### 2. SCRIPT

**SCRIPT** contains a group of subcommands for managing Lua scripts. You can run **SCRIPT HELP** to query specific operations.

Most SCRIPT commands are compatible with open-source Redis. The following commands that need to be noted:

- SCRIPT KILL

GeminiDB Redis API is a multi-thread execution environment, so it allows multiple Lua scripts to be executed at the same time. If **SCRIPT KILL** is executed, all running Lua scripts will be terminated.

For ease of use, it extends the **SCRIPT KILL** command. You can use **SCRIPT KILL SHA1** to terminate the script of a specified hash value. If multiple nodes are executing scripts with the same hash value at the same time, these scripts will be terminated.

In addition, the Lua timeout period (config set lua-time-limit) cannot be configured. You can run **SCRIPT KILL** at any time to terminate the script, instead of waiting for the script to time out.

#### SCRIPT DEBUG

Currently, the DEBUG command is not supported.

SCRIPT GET

This command is added for querying scripts saved to a database with the SCRIPT LOAD command.

The syntax is SCRIPT GET SHA1.

#### 3. Run Redis commands in Lua scripts.

Similar to the open-source Redis, the Lua environment of GeminiDB Redis also provides a global Redis table to provide various functions for interacting with Redis Server.

**Table 5-5** shows the operations supported and not supported by GeminiDB Redis.

| Supported Operation                      | Unsupported Operation                          |
|--|--|
| • redis.call()                           | <ul> <li>redis.log()</li> </ul>                |
| • redis.pcall()                          | <ul> <li>redis.LOG_DEBUG</li> </ul>            |
| <ul> <li>redis.sha1hex()</li> </ul>      | <ul> <li>redis.LOG_VERBOSE</li> </ul>          |
| <ul> <li>redis.error_reply()</li> </ul>  | <ul> <li>redis.LOG_NOTICE</li> </ul>           |
| <ul> <li>redis.status_reply()</li> </ul> | <ul> <li>redis.LOG_WARNING</li> </ul>          |
|  | <ul> <li>redis.replicate_commands()</li> </ul> |
|  | <ul> <li>redis.set_repl()</li> </ul>           |
|  | <ul> <li>redis.REPL_NONE</li> </ul>            |
|  | <ul> <li>redis.REPL_AOF</li> </ul>             |
|  | <ul> <li>redis.REPL_SLAVE</li> </ul>           |
|  | <ul> <li>redis.REPL_REPLICA</li> </ul>         |
|  | <ul> <li>redis.REPL_ALL</li> </ul>             |
|  | <ul> <li>redis.breakpoint()</li> </ul>         |
|  | <ul> <li>redis.debug()</li> </ul>              |

#### 4. Lua execution environment restrictions

The open-source Redis has restrictions on the execution of Lua scripts, for example, restrictions on global variables, random function results, and system libraries and third-party libraries that can be used.

GeminiDB Redis inherits most restrictions of the open-source Redis, but there are some differences in the following scenarios.

- Write Dirty

According to the open-source Redis specifications, if a write operation has been executed by a script, the script cannot be terminated by SCRIPT KILL. You must run SHUTDOWN NOSAVE to directly stop Redis Server.

GeminiDB Redis does not support the SHUTDOWN command, so you can still run **SCRIPT KILL** to stop the script execution.

#### - Random Dirty

Due to the master/slave replication, the open-source Redis stipulates that if a script executes a command (Time or randomkey) to get a random key, the script cannot execute the command for writing semantics.

The following Lua script is used as an example.

local t = redis.call("time")
return redis.call("set", "time", t[1]);

When the execution of the script is transferred to the slave node, the time obtained by the **Time** command must be later than that obtained by the master node. Therefore, the value of the **Set** command executed on the slave node conflicts with that on the master node. The open-source Redis introduces replicate\_commands to allow users to determine the behavior mode in this scenario.

For GeminiDB Redis instances, there is no primary/standby relationship, and there is only one copy of data logically, so it is not limited by this restriction.

## Forbidden Commands in the Lua Script

Hash commands: HSCAN.

List commands: BLPOP, BRPOP, and BRPOPLPUSH.

Set commands: SSCAN.

Sorted set commands: BZPOPMAX, BZPOPMIN, and ZSCAN.

Stream commands: XREAD and XREADGROUP.

Generic commands: RENAME, RENAMENX, RESTORE, SCAN, CLIENT, COMMAND, CONFIG, DBSIZE, FLUSHALL, FLUSHDB, INFO, and KEYS.

Lua commands: EVAL, EVALSHA, and SCRIPT.

Pub/sub commands: PSUBSCRIBE, PUBLISH, PUNSUBSCRIBE, SUBSCRIBE, and UNSUBSCRIBE.

Transactions commands: DISCARD, EXEC, MULTI, UNWATCH, and WATCH.

# 5.5 Compact Encoding for Small Objects

GeminiDB uses RocksDB as the storage engine. RocksDB only operates on a keyvalue (KV) storage model. As a result, any data structure must first be transformed into the KV format before it can be stored, necessitating the use of data encoding techniques. In a conventional encoding scheme, storing a hash object with *N* field-value pairs necessitates *N*+1 KV pairs. This means executing the HGETALL command demands *N*+1 read operations. In certain service contexts, like RTA, where hash data consists of a limited number of elements, each with minimal byte size, the GeminiDB hash compact encoding scheme can significantly enhance read performance. The compact encoding scheme encodes *N* field-value pairs and one metadata pair into a single KV pair, reducing the number of read times and greatly improving the performance of HGETALL and HMGET. Currently, the compact encoding scheme supports only small object storage optimization for the hash data type.

#### Scenarios

With compact encoding schemes, all field-value pairs are stored into a single KV pair. Therefore, compact encoding is not suitable for a single large hash object but is ideal for hash data that meets the following requirements:

- The size of a single hash object is less than or equal to 1 KB.
- The number of field-value pairs in a single hash object ranges from 10 to 50.

#### Usage Notes

- Compact encoding supports only the hash data type.
- Compact encoding is not suitable for large objects. If it is used for large objects, the latency of the HSET and HMSET commands may increase.

• Encoding degradation is not supported. Compact encoding takes effect only for new objects.

# **Configuration and Usage**

- Enabling or disabling compact encoding
  - Run CONFIG SET to dynamically adjust the **small-hashobj-encode-size** parameter by following **4.3.3.1 Connecting to an Instance Using a Load Balancer Address (Recommended)** to determine whether to enable compact encoding.

| Parameter             | small-hashobj-encode-size   |
|-----------------------|---|
| Unit                  | Byte  |
| Default<br>Value      | <b>0</b> : Compact encoding is disabled. Traditional encoding is used.  |
| Recommen<br>ded Value | 1024  |
| Description           | • <b>0</b> : Compact encoding is disabled. All hash objects are stored in conventional encoding mode.   |
|                       | • <b>1024</b> : Compact encoding is enabled. This configuration takes effect only when the size of a hash object is less than or equal to 1024 bytes. If the size of a hash object exceeds 1024 bytes, conventional encoding is used. |

• Viewing the encoding format of an object

Run the following command to view the storage format of a specified key: OBJECT STORAGEFORMAT *<key>* 



• Manually converting the encoding format of an object

Run the following command to convert the compact encoding key back to the conventional encoding key:

OBJECT REFORMAT <key> NORMAL

Conventional encoding cannot be converted for small objects.

#### 

All operation commands (such as HGET and HSET) are compatible regardless of whether objects use conventional or compact encoding.

# 5.6 Keyspace Notification

The keyspace notification function is available on all clients that support subscription and release, without any modification.

## Precautions

- The keyspace notification function is disabled by default because it consumes CPU resources if enabled.
- Do not enable the keyspace notification in high-pressure scenarios. Enabling this function will affect instance performance, and some event notifications may be missed.

### **Differences from Open-source Redis**

1. Configuration methods

Run CONFIG SET to enable or disable the keyspace notification.

#### Config set notify-keyspace-events Ex

- The keyspace notification is disabled if **notify-keyspace-events** is empty or does not contain **K** and **E**.

D NOTE

- Double quotation marks indicate that the parameter is an empty string.
- The GeminiDB Redis console client does not allow you to disable the keyspace notification by setting **notify-keyspace-events** to an empty string.
- The keyspace notification is enabled if **notify-keyspace-events** is not empty and is correctly configured. For details, see **Table 5-6**.
- 2. Notification Types

Table 5-6 Supported notification types

| Char<br>acter | Notification  | Supported by<br>GeminiDB Redis |
|---------------|---|--------------------------------|
| К             | Keyspace notification. All notifications are prefixed by <b>keyspace@</b> .           | Yes                            |
| E             | Key event notification. All notifications are prefixed by <b>keyevent@<db></db></b> . | Yes                            |
| g             | Notifications for generic commands such as DEL, EXPIRE, and RENAME                    | Yes                            |
| \$            | Notifications for string commands   | Yes                            |
| l             | Notifications for list commands   | Yes                            |
| S             | Notifications for set commands  | Yes                            |
| h             | Notifications for hash commands   | Yes                            |
| z             | Notifications for sorted set commands   | Yes                            |
| x             | EXPIRED event notifications   | Yes                            |
| e             | EVICT event notifications   | N/A                            |
| А             | Alias for parameter <b>g\$lshzxe</b>  | Yes                            |

As shown in **Table 5-6**, EVICT event notifications are not applicable for the current version of GeminiDB Redis.

The parameter value must contain either  ${\bf K}$  or  ${\bf E}.$  Otherwise, no notifications are issued.

For example, if you want to subscribe only list-related keyspace notifications, set **notify-keyspace-events** to **Kl**.

The value **AKE** means all types of notifications are issued.

# **5.7 ExHash Commands**

ExHash is an enhanced hash data structure that allows you to specify expiration times and version numbers for fields. ExHash is flexible and can help simplify business development in most scenarios.

## Highlights

- The expiration time and version number can be specified for each field.
- Efficient, flexible active and passive expiration strategies are supported for fields.
- The syntax is similar to that of native Redis HASH.

#### Commands

| Command | Syntax   | Description  |
|---------|--|--|
| EXHSET  | EXHSET key field value [EX<br>time] [EXAT time] [PX time]<br>[PXAT time] [NX   XX] [VER  <br>ABS   GT version] [KEEPTTL] | Adds a field to an ExHash key.<br>If the key does not exist, it is<br>automatically created. If the<br>field already exists, this<br>command overwrites the<br>current value of the field. |
| EXHGET  | EXHGET key field   | Retrieves the value of a field<br>from an ExHash key. If the key<br>or field does not exist, <b>nil</b> is<br>returned.  |
| EXHPTTL | EXHPTTL key field  | Queries the remaining time to<br>live (in milliseconds) of a field<br>in an ExHash key   |
| EXHTTL  | EXHTTL key field   | Queries the remaining time to<br>live (in seconds) of a field in<br>an ExHash key  |
| EXHVER  | EXHVER key field   | Queries the current version of a field in an ExHash key.   |

Table 5-7 ExHash commands

| Command            | Syntax   | Description  |
|--------------------|--|--|
| EXHINCRBY          | EXHINCRBY key field num [EX<br>time] [EXAT time] [PX time]<br>[PXAT time] [VER   ABS   GT<br>version] [MIN minval] [MAX<br>maxval] [KEEPTTL]         | Increases the value of a field<br>in an ExHash key by the <b>num</b><br>value (an integer). If the key<br>does not exist, it is<br>automatically created. If the<br>field does not exist, this<br>command adds the field and<br>sets the value of the field to <b>0</b><br>before increasing the value of<br>the field.<br><b>NOTE</b><br>To add a field that does not<br>expire, you can run this<br>command to add the field<br>without specifying an expiration<br>time.              |
| EXHINCRBYFL<br>OAT | EXHINCRBYFLOAT key field<br>num [EX time] [EXAT time]<br>[PX time] [PXAT time] [VER  <br>ABS   GT version] [MIN<br>minval] [MAX maxval]<br>[KEEPTTL] | Increases the value of a field<br>in an ExHash key by the <b>num</b><br>value (a floating-point<br>number). If the key does not<br>exist, it is automatically<br>created. If the field does not<br>exist, this command adds the<br>field and sets the value of the<br>field to <b>0</b> before increasing the<br>value of the field.<br><b>NOTE</b><br>To add a field that does not<br>expire, you can run this<br>command to add the field<br>without specifying an expiration<br>time. |
| EXHMGET            | EXHMGET key field [field]  | Retrieves multiple field values<br>from an ExHash key in each<br>query. If the key or fields do<br>not exist, <b>nil</b> is returned.  |
| EXHLEN             | EXHLEN key [NOEXP]   | Retrieves the number of fields<br>in an ExHash key. The output<br>may include the number of<br>expired fields that are not<br>deleted, because this<br>command does not trigger a<br>passive eviction or filter out<br>expired fields. To obtain only<br>the number of fields that have<br>not expired, you can set<br>parameter <b>NOEXP</b> in the<br>command.   |

| Command   | Syntax                   | Description  |
|-----------|--------------------------|--|
| EXHGETALL | EXHGETALL key            | Retrieves all fields and their values from an ExHash key.  |
| EXHDEL    | EXHDEL key field [field] | Deletes a field from an<br>ExHash key. If the key or field<br>does not exist, <b>0</b> is returned. If<br>the field is deleted, <b>1</b> is<br>returned. |
| DEL       | DEL <key> [key]</key>    | Deletes one or more ExHash<br>keys.  |
| EXISTS    | EXISTS <key> [key]</key> | Checks whether there is one or more ExHash data records.   |

# **Complex Commands and Options**

• EXHSET

#### Table 5-8 EXHSET commands

| ltem   | Description   |
|--|---|
| SyntaxEXHSET key field value [EX time] [EXAT time] [PX tim<br>[PXAT time] [NX   XX] [VER   GT   ABS version] [KEEF |   |
| Description  | Adds a field to an ExHash key. If the key does not exist, it is<br>automatically created. If the field already exists, this<br>command overwrites the current value of the field.           |
|  | To add a field that does not expire, you can run this command to add the field without specifying an expiration time.   |
| Parameters   | <b>key</b> : a piece of ExHash data that you want to manage by running this command   |
|  | <b>field</b> : an element of the key. An ExHash key can have multiple fields.   |
|  | value: value of the field. A field can have only one value.   |
|  | <b>EX</b> : relative expiration time of the field, in seconds. <b>0</b> indicates that the field will expire immediately. If this parameter is not specified, the field does not expire.    |
|  | <b>EXAT</b> : absolute expiration time of the field, in seconds. <b>0</b> indicates that the field will expire immediately. If this parameter is not specified, the field does not expire.  |
|  | <b>PX</b> : relative expiration time of a field, in milliseconds. <b>0</b> indicates that the field will expire immediately. If this parameter is not specified, the field does not expire. |

| Item Description |   |
|------------------|---|
|                  | <ul> <li>PXAT: absolute expiration time of the field, in milliseconds.</li> <li>0 indicates that the field will expire immediately. If this parameter is not specified, the field does not expire.</li> </ul>   |
|                  | <b>NX</b> : This parameter is added only when the field does not exist.   |
|                  | <b>XX</b> : This parameter is inserted only if the field exists.  |
|                  | <b>VER</b> : version number of the field. If the field exists, the version number specified by this parameter is compared with the current version number. If they are the same, the system continues to run this command and increases the version number by 1. If they are different, an error message is returned. If the field does not exist or the current version of the field is 0, the system ignores this parameter and runs the command. After the operation completes, the version number changes to 1. |
|                  | <b>GT</b> : version later than the current one. If it is earlier than the current one, an error message is returned.  |
|                  | <b>ABS:</b> absolute version number of the field. The system forcibly writes the field to the key regardless of whether the field already exists.   |
|                  | <b>KEEPTTL</b> : retains the current time to live of the field if none of the <b>EX</b> , <b>EXAT</b> , <b>PX</b> , and <b>PXAT</b> parameters are specified.   |
| Returned         | If a field is created and a value is set for it, <b>1</b> is returned.  |
| values           | If a field already exists and the specified value overwrites the current value, <b>0</b> is returned.   |
|                  | If <b>XX</b> is specified and the field does not exist, <b>-1</b> is returned.  |
|                  | If <b>NX</b> is specified and the field exists, <b>-1</b> is returned.  |
|                  | If <b>VER</b> is specified and the value does not match the current version, the error message "ERR update version is stale" is returned.   |
|                  | Error messages are returned in other cases.   |

#### – Example

Setting the expiration time for a field 127.0.0.1:6579> EXHSET k1 f1 v1 ex 10 (integer) 1 127.0.0.1:6579> EXHGET k1 f1 "v1" 127.0.0.1:6579> EXHSET k1 f2 v2 ex 10 (integer) 1 127.0.0.1:6579> EXHGET k1 f1 (nil) 127.0.0.1:6579> EXHGETALL k1 127.0.0.1:6579> EXHGETALL k1 (empty array) Setting a version number for a field 127.0.0.1:6579> EXHSET k1 f1 v1 (integer) 1 127.0.0.1:6579> EXHVER k1 f1 (integer) 1 127.0.0.1:6579> EXHSET k1 f1 v1 ver 2 (error) ERR update version is stale 127.0.0.1:6579> EXHSET k1 f1 v1 ver 1 (integer) 0 127.0.0.1:6579> EXHVER k1 f1 (integer) 2 127.0.0.1:6579> EXHSET k1 f1 v1 (integer) 0 127.0.0.1:6579> EXHVER k1 f1 (integer) 3 127.0.0.1:6579> EXHSET k1 f1 v1 GT 3 (error) ERR update version is stale 127.0.0.1:6579> EXHSET k1 f1 v1 GT 2 (error) ERR update version is stale 127.0.0.1:6579> EXHSET k1 f1 v1 GT 4 (integer) 0 127.0.0.1:6579> EXHVER k1 f1 (integer) 4 127.0.0.1:6579> EXHSET k1 f1 v1 abs 2 (integer) 0 127.0.0.1:6579> EXHVER k1 f1 (integer) 2

• EXHINCRBY

Table 5-9 EXHINCRBY commands

| ltem        | Description  |
|-------------|--|
| Syntax      | EXHINCRBY key field num [EX time] [EXAT time] [PX time]<br>[PXAT time] [VER   GT   ABS version] [MIN minval] [MAX<br>maxval] [KEEPTTL]   |
| Description | Increases the value of a field in an ExHash key by the <b>num</b> value (an integer). If the key does not exist, it is automatically created. If the field does not exist, this command adds the field and sets the value of the field to <b>0</b> before increasing the value of the field. |
|             | command to add the field without specifying an expiration time.  |
| Parameters  | <b>key</b> : a piece of ExHash data that you want to manage by running this command  |
|             | <b>field</b> : an element of the key. An ExHash key can have multiple fields.  |
|             | <b>num</b> : an integer by which you want to increase the value of the field   |

| Item               | Description  |
|--------------------|--|
|                    | <b>EX</b> : relative expiration time of the field, in seconds. <b>0</b> indicates that the field will expire immediately. If this parameter is not specified, the field does not expire.   |
|                    | <b>EXAT</b> : absolute expiration time of the field, in seconds. <b>0</b> indicates that the field will expire immediately. If this parameter is not specified, the field does not expire.   |
|                    | <b>PX</b> : relative expiration time of a field, in milliseconds. <b>0</b> indicates that the field will expire immediately. If this parameter is not specified, the field does not expire.  |
|                    | <ul> <li>PXAT: absolute expiration time of the field, in milliseconds.</li> <li>0 indicates that the field will expire immediately. If this parameter is not specified, the field does not expire.</li> </ul>  |
|                    | <b>VER</b> : version number of the field. If the field exists, the version number specified by this parameter is compared with the current version number. If they are the same, the system continues to run this command and increases the version number by 1. If they are different, an error message is returned. If the field does not exist or the current version of the field is 0, ignore this parameter and run the command. After the operation completes, the version number changes to 1. |
|                    | <b>GT</b> : version later than the current one. If it is earlier than the current one, an error message is returned.   |
|                    | <b>ABS:</b> absolute version number of the field. The system forcibly writes the field to the key regardless of whether the field already exists.  |
|                    | <b>KEEPTTL</b> : retains the current time to live of the field if none of the <b>EX</b> , <b>EXAT</b> , <b>PX</b> , and <b>PXAT</b> parameters are specified.  |
|                    | <b>MIN</b> : minimum value of the field. If the field value is less than this lower limit, an error message is returned.   |
|                    | <b>MAX</b> : maximum value of the field. If the field value is greater than this upper limit, an error message is returned.  |
| Returned<br>values | If the operation is successful, the value increased by the <b>num</b> value is returned.   |
|                    | Otherwise, an error message is returned.   |

#### – Example

An example of using the MIN and MAX parameters

```
127.0.0.1:6579> EXHINCRBY k1 f1 5 min 6
(error) ERR increment or decrement would overflow
127.0.0.1:6579> EXHINCRBY k1 f1 5 min 4
(integer) 5
```

127.0.0.1:6579> EXHINCRBY k1 f1 5 max 9 (error) ERR increment or decrement would overflow 127.0.0.1:6579> EXHINCRBY k1 f1 3 max 9 (integer) 8

### **Example of ExHash Commands**

#### JAVA(Jedis)

package nosql.cloud.huawei.jedis;

import redis.clients.jedis.\*;
import redis.clients.jedis.util.SafeEncoder;

import java.util.ArrayList;

```
public class Main{
  public static void main(String[] args) throws InterruptedException {
     // Initialize the Jedis resource pool configuration.
     JedisPoolConfig jedisPoolConfig = new JedisPoolConfig();
     // Set the maximum number of connections in the resource pool.
     jedisPoolConfig.setMaxTotal(10);
     // Set the maximum number of idle connections allowed by the pool.
     jedisPoolConfig.setMaxIdle(10);
     // Set the minimum number of idle connections retained in the pool.
     jedisPoolConfig.setMinIdle(2);
    // Initialize the Jedis resource pool based on the configuration.
     // Note: If the version does not support Access Control List (ACL), the value of user must be null.
     JedisPool jedisPool = new JedisPool(jedisPoolConfig, "127.0.0.1", 6379, null, "*****");
     // Obtain connections from the pool.
     try (Jedis jedis = jedisPool.getResource()) {
        // example for: EXHSET key field value [EX time] [EXAT time] [PX time] [PXAT time] [NX | XX] [VER
ABS | GT version] [KEEPTTL]
        jedis.sendCommand(() -> SafeEncoder.encode("exhset"), "key", "field1", "value1");
        jedis.sendCommand(() -> SafeEncoder.encode("exhset"), "key", "field2", "value2", "EX", "5");
        // example for: EXHGET key field
        byte[] byteArray = (byte[]) jedis.sendCommand(() -> SafeEncoder.encode("exhget"), "key", "field1");
        System.out.println(new String(byteArray));
        byteArray = (byte[]) jedis.sendCommand(() -> SafeEncoder.encode("exhget"), "key", "field2");
        System.out.println(new String(byteArray));
        // example for: EXHGETALL key
        ArrayList<byte[]> byteArrayList = (ArrayList<byte[]>) jedis.sendCommand(() ->
SafeEncoder.encode("exhgetall"), "key");
        for (byte[] ba : byteArrayList) {
          System.out.print(new String(ba));
           System.out.print(" ");
        System.out.println();
        // sleep for 5 seconds
        Thread.sleep(5000);
        // exhgetall after sleeping
        byteArrayList = (ArrayList<byte[]>) jedis.sendCommand(() -> SafeEncoder.encode("exhgetall"),
"key");
        for (byte[] ba : byteArrayList) {
          System.out.print(new String(ba));
           System.out.print(" ");
        }
     }
     // Disable the pool.
     jedisPool.close();
  }
}
```

For details about the best practices of ExHash commands, see **6.4 ExHash for Ad Frequency Control**.

# 5.8 Large Bitmap Initialization

The open-source Redis uses string bitmaps, which may create super large strings and affect the performance of big keys in some scenarios. GeminiDB Redis API uses bitmaps in a special encoding format. The internal sharding algorithm prevents super large strings from being created and allows you to insert and delete a random number of bits efficiently.

However, in practice, a super large bitmap of the string type may be obtained from other sources. For example, inserting a super-large bitmap (64 MB) into GeminiDB Redis instances via the **SET** command takes a long time, which interferes with access and causes jitter or latency. To address these issues, we provide a smooth insertion solution. A super large bitmap is split into smaller strings (for example, 1 MB). The **SET** command is used for the first insertion, and then a GETBIT read-only command is used to convert the strings to bitmaps. The subsequent character strings are inserted by running the **APPEND** command.

#### Usage Notes

- Currently, this function is available in kernel 5.0.5.00 and later. You can choose Service Tickets > Create Service Ticket and contact the customer service to check whether the instance version supports this function. To use this function, upgrade the kernel by following 4.6.1 Upgrading a Minor Version.
- The **APPEND** command has requirements on the sequence. Therefore, **APPEND** disorder must be avoided in the entire process (in concurrent APPEND scenarios).
- PIPELINE acceleration and PIPELINE can ensure the execution sequence.
- The smaller (recommended: 256 KB to 1 MB) the substrings are, the less the latency will vary, but initialization will take longer.

#### **Code Reference**

#### C++ Example

```
#include <string>
#include <vector>
#include "hiredis/hiredis.h"
constexpr std::size_t kBitmapSubSize = 1024 * 1024; // 1 MB
void SmoothInitBitmap(std::string bitmap) {
// Split bitmap
std::vector<std::string> sub_bitmaps;
std::size_t pos = 0;
while (pos < bitmap.size()) {</pre>
 sub_bitmaps.emplace_back(bitmap.substr(pos, kBitmapSubSize));
  pos += kBitmapSubSize;
std::string key = "BITMAP_KEY";
// Connect to redis
redisContext* redis = redisConnect("127.0.0.1", 6666);
redisReply* reply = nullptr;
// First part use 'SET' command
reply = (redisReply*)redisCommand(redis, "SET %b %b", key.data(), key.size(), sub_bitmaps[0].data(),
sub_bitmaps[0].size());
```

```
freeReplyObject(reply);
// Use 'GETBIT' to transform to bitmap format
reply = (redisReply*)redisCommand(redis, "GETBIT %b 0", key.data(), key.size());
freeReplyObject(reply);
// Use 'APPEND' for remaining bitmap data
for (auto i = 1u; i < sub_bitmaps.size(); ++i) {
    reply = (redisReply*)redisCommand(redis, "APPEND %b %b", key.data(), key.size(), sub_bitmaps[i].data(),
    sub_bitmaps[i].size());
    freeReplyObject(reply);
}
}
}
int main() {
    std::string bitmap
="123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef123457890abcdef1234578
```

#### JAVA (Jedis) Example

```
package nosql.cloud.huawei.jedis;
import redis.clients.jedis.Jedis;
import java.nio.ByteBuffer;
import java.util.BitSet;
public class BitMapOperation {
  private Jedis jedis;
  public BitMapOperation(Jedis jedis) {
    this.jedis = jedis;
 }
  * SetBit operation especially for big bitmap
  * @param key
                       key
  * @param value
                       value
  * @param groupLength groupLength (Unit: byte)
  public void setBitGrouped(byte[] key, BitSet value, int groupLength) {
    if (value.isEmpty()) {
       jedis.set(key, new byte[0]);
       return:
   }
    byte[] byteArray = disposeBitMap(value);
    // round count
    int round = byteArray.length % groupLength == 0 ? byteArray.length / groupLength : byteArray.length /
groupLength + 1;
    // last round length
    int lastPacketLength = byteArray.length % groupLength == 0 ? groupLength : byteArray.length %
groupLength;
    if (round == 1) {
       // if only one round
       byte[] lastPacketByte = new byte[lastPacketLength];
       System.arraycopy(byteArray, 0, lastPacketByte, 0, lastPacketLength);
       // set and getBit
       setAndGetBit(key, lastPacketByte);
       return;
   }
    byte[] packetByte = new byte[groupLength];
    byte[] lastPacketByte = new byte[lastPacketLength];
    for (int i = 0; i < round; i++) {
```

if (i == 0) {
 // first set

```
System.arraycopy(byteArray, i * groupLength, packetByte, 0, groupLength);
          // set and getBit
          setAndGetBit(key, packetByte);
      } else if (i != round - 1) {
          // regular append
          System.arraycopy(byteArray, i * groupLength, packetByte, 0, groupLength);
          jedis.append(key, packetByte);
      } else {
          // last append
          System.arraycopy(byteArray, i * groupLength, lastPacketByte, 0, lastPacketLength);
          jedis.append(key, lastPacketByte);
      }
   }
 }
  private byte[] disposeBitMap(BitSet bitSet) {
    // get words and count the number of word(Long)
    long[] words = bitSet.toLongArray();
    int n = words.length;
    if (n == 0)
       return new byte[0];
    for (int i = 0; i < n; i++) {
       // reverse
       words[i] = reverseLong(words[i]);
    }
    return longToBytes(words);
 }
  public static byte[] longToBytes(long[] longArray) {
    ByteBuffer buffer = ByteBuffer.allocate(longArray.length * 8);
    for (long value : longArray) {
       buffer.putLong(value);
    }
    return buffer.array();
 }
  public void setAndGetBit(byte[] key, byte[] value) {
    jedis.set(key, value);
    jedis.getbit(key, 0);
 }
  public static long reverseLong(long n) {
    n = (n >>> 32) | (n << 32);
    n = ((n & 0xFFFF0000FFFF0000L) >>> 16) | ((n & 0x0000FFFF0000FFFFL) << 16);
    n = ((n & 0xFF00FF00FF00FF00L) >>> 8) | ((n & 0x00FF00FF00FF00FFL) << 8);
    n = ((n & 0xF0F0F0F0F0F0F0F0F0L) >>> 4) | ((n & 0x0F0F0F0F0F0F0F0F0L) << 4);
    n = ((n & 0xCCCCCCCCCCCCL) >>> 2) | ((n & 0x333333333333333333) << 2);
    n = ((n & 0xAAAAAAAAAAAAAAAAA) >>> 1) | ((n & 0x55555555555555555) << 1);
    return n:
 }
}
```

## **Python Example**

import redis import random import string from bitmap import BitMap # pip install bitmap # Parameters max\_bytes = 1024 \* 1024 \* 64 # Construct a 64 MB bitmap. max\_bits = max\_bytes \* 8 # A byte consists of eight bits (over 500 million characters). # Python built-in bitmaps are not required. # index\_list All subscripts that are to be set to 1 are stored. index\_list = [] for i in range(1000000): index\_list.append(random.randint(0, max\_bits - 1)) # Create a bitmap in a byte array. byte\_array = bytearray(max\_bytes) for i in index\_list:

```
index = i // 8
  offset = i % 8
  byte_array[index] |= (1 << (7 - offset))</pre>
# Convert the bitmap to bytes for subsequent operations.
bitmap_str = bytes(byte_array)
# Connect to Redis.
r = redis.Redis(host='127.0.0.1', port=6379)
r.execute_command("auth a")
key = "BITMAP_KEY
#Separate parameters.
bitmap_pos = 0
bitmap_sub_size = 256 * 1024 # Adjust the splitting granularity.
step = bitmap sub size - 1
# Process the first part.
first_part = bitmap_str[bitmap_pos : bitmap_pos + step]
r.execute_command("SET", key, first_part)
r.execute_command("GETBIT", key, 0) # Run GETBIT to optimize bitmap code.
# Process the remaining part.
bitmap_pos += step
while bitmap pos < len(bitmap str) :
  rest_part = bitmap_str[bitmap_pos : bitmap_pos + step]
  r.execute_command("APPEND", key, rest_part)
  bitmap pos += step
# The following is the test and verification code. Executing the code takes a long time as the GETBIT
command will be executed for 1 million times.
# The BITCOUNT command with O(N) time complexity will cause a latency spike of 100 milliseconds. Do
not use this command in the production environment.
# (Optional) Construct a Python built-in bitmap data verification.
bm = BitMap(max_bits)
for i in index list:
  bm.set(i)
print('BitMap.count(): ' + str(bm.count()))
# Call the Redis command to check whether the settings are correct.
success = True
for i in index list:
  if r.execute_command("GETBIT", key, i) != 1:
     print('GETBIT check error, pos is' + str(i))
     success = False
if success:
  print('GETBIT check success')
```

print("Bitcount: " + str(r.execute\_command("BITCOUNT", key)))

# 5.9 Querying Large Bitmaps by Page

To query large bitmaps or sub-bitmaps in the specified range [**start**, **end**], traditional solutions have the following bottlenecks:

- **Low-efficiency data appending**: If the client obtains data bit by bit via GETBIT in a loop and appends the data, high-frequency requests and computing overheads are generated, resulting in significant performance bottlenecks.
- **Big key access risks**: If GET is used to obtain a complete bitmap, databases need to transmit ultra-large binary data chunks. Network congestion will occur, leading to server delay jitter that could impact overall system stability.

To solve this problem, RANGEBITARRAY can be executed on databases. This enhanced bitmap command has the following advantages:

• Efficient query by block: You can obtain binary data chunks in a given range by specifying range parameters. The client can obtain and assemble a large data set by running RANGEBITARRAY multiple times.

- Avoiding big keys: Data is transferred by chunk to prevent a single operation from reaching the performance threshold of big keys and to effectively reduce server delay jitter.
- **End-to-end performance optimization**: RTT and server resource consumption are reduced to improve the query throughput and response stability.

You are advised to run RANGEBITARRAY to obtain massive bitmaps.

| Table 5-1 | 10 RANGE | BITARRAY |
|-----------|----------|----------|
|-----------|----------|----------|

| Туре               | Description   |
|--------------------|---|
| Syntax             | RANGEBITARRAY key start end   |
| Time<br>Complexity | O(C): C indicates the length of [ <b>start</b> , <b>end</b> ].  |
| Description        | Obtains the character string consisting of all bit values (0 and 1) in a specified range of a bitmap.   |
| Value              | <ul> <li>Key: key name (bitmap data structure)</li> <li>start: start offset (including the value)</li> <li>end: end offset (including the value)</li> </ul>   |
| Returned<br>Value  | <ul> <li>If the command is successfully executed, a string consisting of all bit values (0 and 1) in the specified range is returned.</li> <li>If the key does not exist, an empty string is returned.</li> <li>In other cases, an error message is returned, for example, "Unsupported old encoding" or "The key must be of the bitmap type."</li> </ul> |
| Example            | Run the following command to preset data:<br>SETBIT foo 2 1<br>SETBIT foo 3 1<br>SETBIT foo 5 1<br>• Example command:<br>RANGEBITARRAY foo 0 5<br>• Example response:<br>"001101"   |

#### **Usage Notes**

- To check whether the instance version supports this function, you can choose Service Tickets > Create Service Ticket in the upper right corner of the console and contact the customer service. To use this function, upgrade the kernel by following 4.6.1 Upgrading a Minor Version.
- The key must be of the bitmap type. If the key is of the string type, you need to run GETBIT to convert the key.
- The semantics of **start** and **end** are the same as that of GETRANGE.
- If the client needs to run RANGEBITARRAY for multiple times and append the results, you can run PIPELINE to accelerate the execution.

• You can select the range query granularity. A finer granularity indicates a smaller delay glitch. A coarser granularity indicates a larger delay glitch. Generally, a range query granularity of 32 KB is recommended for appending the results.

# **Code Reference**

## Java (jedis) Example

```
import redis.clients.jedis.Jedis;
import redis.clients.jedis.JedisPool;
import redis.clients.jedis.util.SafeEncoder;
import redis.clients.jedis.Protocol;
import java.nio.charset.StandardCharsets;
public class BitmapRangePager {
  private final JedisPool jedisPool;
  private static final int PAGE_SIZE_BYTES = 32 * 1024; // 32 KB per page (in bytes)
  public BitmapRangePager(JedisPool jedisPool) {
     this.jedisPool = jedisPool;
  }
  public String getFullBitmap(String key) {
     try (Jedis jedis = jedisPool.getResource()) {
       // Obtains the total number of bytes in the bitmap.
        long totalBytes = jedis.strlen(key);
        if (totalBytes == 0) return "";
        StringBuilder result = new StringBuilder();
       // Querying data page by page (in bytes) (more intuitive)
        for (long byteOffset = 0; byteOffset < totalBytes; byteOffset += PAGE_SIZE_BYTES) {
        // Calculates the number of bytes on the current page (the size of the last page may be less than
32 KB).
           int currentPageBytes = (int) Math.min(PAGE_SIZE_BYTES, totalBytes - byteOffset);
        // Converts to a bit interval (closed interval).
           long bitStart = byteOffset * 8;
           long bitEnd = (byteOffset + currentPageBytes) * 8 - 1; // Includes the end bit.
        // Runs the command and appends the results.
           String pageBits = executeRangeBitArray(jedis, key, bitStart, bitEnd);
           result.append(pageBits);
        }
        return result.toString();
     }
  }
  private String executeRangeBitArray(Jedis jedis, String key, long start, long end) {
   // Runs RANGEBITARRAY key start end.
     Object response = jedis.sendCommand(
           new Protocol.Command("RANGEBITARRAY"),
           SafeEncoder.encode(key),
           Protocol.toByteArray(start),
           Protocol.toByteArray(end)
     );
   // Processes the response (ensures that the binary data is converted to a character string).
     if (response instanceof byte[]) {
        return new String((byte[]) response, StandardCharsets.US_ASCII);
     } else {
        return response.toString();
     }
  }
}
```

#### **Python Example**

```
import redis
import math
class BitmapRangePager:
  def __init__(self, redis_client):
     self.redis = redis client
    self.PAGE_SIZE_BYTES = 32 * 1024 # 32 KB per page (in bytes)
  def get_full_bitmap(self, key):
    Obtains a complete bitmap character string.
     :param key: Redis key name
     :return: a complete character string consisting of 0 and 1
     # Obtains the total number of bytes in a bitmap.
     total_bytes = self.redis.strlen(key)
     if total_bytes == 0:
        return ""
     result = []
    # Querying data page by page (in bytes)
     for byte_offset in range(0, total_bytes, self.PAGE_SIZE_BYTES):
        Calculates the number of bytes on the current page (the size of the last page may be less than 32
KB).
        current_page_bytes = min(self.PAGE_SIZE_BYTES, total_bytes - byte_offset)
      # Converts to a bit interval [start, end].
        bit_start = byte_offset * 8
        bit_end = (byte_offset + current_page_bytes) * 8 - 1
       # Runs the command and collects the results.
        page_bits = self.execute_range_bitarray(key, bit_start, bit_end)
        result.append(page_bits)
     return ".join(result)
  def execute_range_bitarray(self, key, start, end):
      Runs RANGEBITARRAY key start end.
     :return: a character string consisting of {\bf 0} and {\bf 1}
    # Runs a custom command supported by the server.
     response = self.redis.execute_command(
        "RANGEBITARRAY", # The actual command name needs to be adjusted.
        key,
        start,
        end
    # Processes the response (bytes returned by redis-py).
    return response.decode('ascii') # Assumes that an ASCII character string is returned.
# Example
if __name__ == "__main__":
   # Creates a Redis client.
  r = redis.Redis(host='localhost', port=6379, decode_responses=False) # Keeps the original returned bytes.
  pager = BitmapRangePager(r)
  full_bitmap = pager.get_full_bitmap("my_large_bitmap")
  # Output example (only the first 100 characters are generated)
  print(f"Bitmap (first 100 chars): {full_bitmap[:100]}")
  print(f"Total length: {len(full_bitmap)} bits")
```

# 5.10 Configuring Parameters for a Client Connection Pool

Setting proper parameters for a connection pool can effectively improve Redis performance of the client. Improper configuration (for example, the number of maximum connections is set to a small value) may cause applications not to be connected, affecting production services. This section uses JedisPool of the Redis client Jedis as an example to describe how to use JedisPool and its parameters, providing optimal configuration reference for service developers.

#### **Usage Instructions**

Take Jedis 4.3.1 as an example. The Maven dependency configuration is as follows:

#### <dependency>

<groupId>redis.clients</groupId> <artifactId>jedis</artifactId> <version>4.3.1</version> <scope>compile</scope>

</dependency>

Jedis uses Apache Commons-pool2 to manage its connection pool. When defining JedisPool, pay attention to the key parameter **GenericObjectPoolConfig (Jedis)**. The following is an example for using this parameter:

```
GenericObjectPoolConfig<Jedis> config = new GenericObjectPoolConfig<>();
config.setMaxTotal(100);
config.setMaxIdle(50);
config.setMinIdle(5);
config.setTestWhileIdle(true);
```

The initialization method of Jedis is as follows:

```
JedisPool pool = new JedisPool(config, host, port, timeout, password);// Create a connection pool.
try (Jedis jedis = pool.getResource()) {//Obtain a connection and automatically release the connection after
the execution.
    //Run the following command on the Jedis client:
} catch (Exception e) {
    e.printStackTrace();
}
pool.close();//Close the connection pool.
```

#### **Parameter Description**

Jedis connections are resources managed by JedisPoo. JedisPool controls resources and protects threads. Proper **GenericObjectPoolConfig** configuration can improve Redis service performance and reduce resource overhead. The following table lists some important parameters and describes how to set the parameters.

|                    | I   |                          |   |
|--------------------|---|--------------------------|---|
| Jedis<br>Parameter | Description   | Defa<br>ult<br>Valu<br>e | Recommended Setting   |
| maxTotal           | Maximum number of<br>concurrent connections in<br>the current resource pool.<br>The number of Redis<br>connections must be set<br>based on the service<br>volume. If the value is set<br>to too large, resources are<br>wasted. If the value is set<br>to too small, connections<br>cannot be obtained,<br>affecting services.  | 8                        | The number of client nodes<br>multiplied by the value of<br><b>maxTotal</b> cannot exceed the<br>maximum number of Redis<br>connections.<br>Assume that the QPS of a<br>connection is about 1000 per<br>second or millisecond and<br>the expected QPS of a single<br>Redis instance is 50,000.<br>Theoretically, the required<br>resource pool size (MaxTotal)<br>is 50 (50000/1000). |
| maxIdle            | Maximum number of idle<br>connections in a resource<br>pool.<br>After the number of idle<br>connections reaches the<br>value of <b>maxIdle</b> , the<br>resource pool starts to<br>revoke idle connections<br>until the number of idle<br>connections reaches the<br>value of <b>minIdle</b> . This<br>prevents empty<br>connections from being<br>occupied and resources<br>from being wasted. | 8                        | <b>maxIdle</b> indicates the<br>maximum number of<br>connections required by the<br>service, and <b>maxTotal</b><br>indicates the margin. You are<br>not advised to set <b>maxIdle</b><br>to a small value. Otherwise,<br>new Jedis (new connection)<br>overhead occurs.  |
| minIdle            | Minimum number of idle<br>connections in a resource<br>pool.<br>These connections are not<br>revoked to prevent delayed<br>connection creation when<br>traffic increases.   | 0                        | 10 to 20  |
| maxWaitMi<br>llis  | Maximum wait time (in<br>milliseconds) of the invoker<br>after the resource pool<br>connections are used up   | -1                       | You are advised to set a<br>proper timeout interval to<br>prevent application blocking<br>after connection pools are<br>used up.  |

 Table 5-11
 Common Jedis parameters

| Jedis<br>Parameter | Description   | Defa<br>ult<br>Valu<br>e | Recommended Setting   |
|--------------------|---|--------------------------|---|
| testWhileId<br>le  | Indicates whether to use<br>the <b>ping</b> command to<br>monitor the connection<br>validity during idle resource<br>monitoring. Invalid<br>connections will be<br>destroyed.                   | false                    | true  |
| testOnBorr<br>ow   | Indicates whether to check<br>the connection validity (by<br>sending a ping request)<br>each time a connection is<br>obtained from a resource<br>pool. Invalid connections<br>will be released. | false                    | The preset value is<br>recommended. If this<br>parameter is set to <b>true</b> , a<br><b>ping</b> command is sent before<br>each command is executed,<br>which affects the<br>performance of applications<br>with a large number of<br>concurrent requests. To<br>ensure high availability and<br>valid connections, set this<br>parameter to <b>true</b> . |
| testOnRetu<br>rn   | Indicates whether to check<br>the connection validity (by<br>sending a ping request)<br>each time a connection is<br>returned to a connection<br>pool. Invalid connections<br>will be released. | false                    | The preset value is<br>recommended. If this<br>parameter is set to <b>true</b> , a<br><b>ping</b> command is sent after<br>each command is executed,<br>which affects the<br>performance of applications<br>with a large number of<br>concurrent requests.  |
| timeout            | <b>Socket timeout</b> value of Jedis, in milliseconds   | 2000                     | 200 to 1000   |

# 5.11 Using Parallel SCAN to Accelerate Full Database Scanning

When there are a large number of keys in an instance, Redis SCAN commands takes a long time to work. GeminiDB Redis API utilizes a distributed architecture that enables concurrent scanning of multiple data partitions, resulting in parallel acceleration.

# Precautions

• This solution applies only to GeminiDB Redis cluster instances.

• When using the SCAN command with the **PARTITION** parameter, the returned cursor must match the same partition when continuing the scanning process. Value of the **PARTITION** should not be changed temporarily; otherwise, the scanned data may not meet the expected results, or an error may occur.

## Procedure

**Step 1** Obtain information about all partitions of a GeminiDB Redis instance for subsequent parallel scanning.

Data partitioning: There are many data partitions at the bottom layer of a GeminiDB Redis cluster instance, which are distributed across nodes. Each partition name is a 16-character ID. The name and total number of data partitions at the bottom layer of an instance remain fixed and do not change with any modifications made to the instance.

Obtain the data partition list: Run the **INFO ROUTE** command to obtain all data partitions of the GeminiDB Redis instance. In the following example, the instance has four data partitions: efb06d5c7a4ecb31, c7a36e9eee0103c1, 6fd3dfdbcca37686, 7f7666870a88501b.

```
127.0.0.1:6379>info route
# Route
server: 127.0.0.1:16379 // Display the data partition on the first node.
efb06d5c7a4ecb31 // Data partition.
c7a36e9eee0103c1 // Data partition.
server: 127.0.0.1:26379 // Display the data partition on the second node.
6fd3dfdbcca37686 //Data partition.
7f7666870a88501b // Data partition.
```

**Step 2** Start multiple SCAN tasks to scan different data partitions.

GeminiDB Redis SCAN commands include a new parameter **PARTITION**, which allows users to scan specific data partitions using the open-source syntax. This feature allows for the creation of parallel scanning scripts, enabling SCAN operations on multiple data partitions simultaneously. As a result, scanning performance is greatly improved.

- For details about the standard SCAN command syntax, see **SCAN**.
- Syntax reference for the optional **PARTITION** parameter added to GeminiDB Redis API.

SCAN cursor [MATCH pattern] [COUNT count] [TYPE type] [PARTITION partition\_index]

The syntax of the **MATCH**, **COUNT**, and **TYPE** parameters is the same as that of the open-source Redis.

- PARTITION: specifies a data partition to be scanned. If the cursor returned by the SCAN command is 0, the data partition has been scanned.
- partition\_index: indicates the dictionary sequence number of all data shard IDs, starting from 0. For example, if there are four data partitions in an instance, partition\_index of the partitions is [0,3]. If there are 240 data partitions in an instance, partition\_index of the partitions is [0,239]. For example:

127.0.0.1:6379> scan 0 count 2 partition 1 1) "1125900712148994" 2) 1) "memtier-1" 2) "memtier-12

----End

# 5.12 Accessing a GeminiDB Redis Instance Using a Pipeline

This section describes principles and precautions for using a pipeline to access a GeminiDB Redis instance.

## Pipeline

GeminiDB Redis API is a request-reply model service.



Figure 5-10 Command execution process of GeminiDB Redis API

- 1. A client sends a command to a GeminiDB Redis server.
- 2. The GeminiDB Redis server receives the command and waits in a queue for processing.
- 3. The GeminiDB Redis server processes the command.
- 4. The GeminiDB Redis server sends the result to the client.

Steps 1 and 4 are I/O operations, which are slow and affected by network conditions. Therefore, bottlenecks may occur.

To reduce network costs and fully utilize performance of GeminiDB Redis API, you can execute multiple commands in a pipeline at once without waiting for individual command reply.



Figure 5-11 Accessing a GeminiDB Redis instance using a pipeline

The I/O operation is performed once to issue three commands.

Using a pipeline can reduce read() and write() system calls of a client and server and improve program execution efficiency.

#### **Pipeline Size Selection and Precautions**

Although pipelines can reduce I/O costs, a bigger pipeline size is not always better. Pipelines have a limit on improving program performance. If there are large amounts of data in a pipeline, the client has to wait longer time. If the socket buffer is full due to a large pipeline, network congestion may occur, causing performance deterioration.

Based on experience, 30 to 100 commands in a pipeline can fully utilize database performance. An optimal size is subject to the actual test result.

Other precautions:

- Pipelines do not guarantee atomicity. When processing batch commands, a server parses and executes commands in sequence. These commands are independent of each other. The server may also execute commands of other clients during this process. If a command fails, other commands are not affected. To achieve atomicity, you can execute transactions or Lua scripts.
- A GeminiDB Redis instance must cache the results before finishing processing all commands, so a large pipeline may cause out of memory (OOM) or network congestion. If a single command is too long, the pipeline size needs to be reduced.
- To fulfill stringent requirements on low latency, large pipelines are not recommended.

# 5.13 Processing Transactions on a GeminiDB Redis Instance

### **About Transactions**

A transaction is a logical unit which groups a set of read or write operations, so that they either succeed or fail collectively. In a connection, after a client executes the **multi** command, a GeminiDB Redis instance starts to cache subsequent commands in a queue. When the client sends the **exec** command, the GeminiDB Redis instance executes all commands in the queue in sequence. If a command fails, transactions will be rolled back. All commands are either successful or failed.

| Command | Description   |
|---------|---|
| WATCH   | Monitors one or more keys. If a key is modified before a transaction is executed, the whole transaction aborts. |
| UNWATCH | Flushes all watched keys.   |
| MULTI   | Identifies start of a transaction block.  |
| EXEC    | Executes all commands in a transaction block.   |
| DISCARD | Flushes queued commands in a transaction bock and exits the transaction block.                                  |

Table 5-12 Related commands

#### 

- When a proxy cluster is used, all keys in a transaction must have the same hashtag to ensure transaction atomicity. If hashtag is not used, a transaction will be split into common commands. In this case, the atomicity cannot be ensured.
- Atomic transactions mean that all of them will either succeed or fail, so you need to take care with command validity when compiling a transaction.
- The commands in a transaction are executed in sequence, so you need to take care with the command sequence when compiling a transaction.
- Do not pack too many or complex commands in a single transaction, or requests may be blocked or the instance status may be abnormal.

## **Example Code**

When the client modifies **key1** and **key2** at the same time in a transaction, they are either successfully modified or fail to be modified at the same time.

```
package nosql.cloud.huawei.jedis;
import java.util.List;
```

```
import redis.clients.jedis.Jedis;
import redis.clients.jedis.Transaction;
public class TranscationTest {
   private static final String host = "127.0.0.1";
  private static final int port = 6379;
   private static final String pwd = "password";
  private static Jedis jedis;
   static {
     jedis = new Jedis(host, port);
      String authString = jedis.auth(password);
     if (!authString.equals("OK")) {
        jedis.close();
        jedis = null;
     }
  }
  public static void main(String[] args) {
     if (jedis == null) {
        return;
     }
     String str_key1 = "{str}key1";
     String str_key2 = "{str}key2";
     jedis.set(str_key1, "0");
jedis.set(str_key2, "0");
     jedis.watch(str_key1);
      // Starts processing transactions.
      Transaction tx = jedis.multi();
     tx.set(str_key1, "500");
     tx.get(str_key1);
  tx.set(str_key2, "1000");
      tx.get(str_key2);
     List<Object> result = tx.exec();
     if (result.isEmpty()) {
        System.out.println ("Error: The transaction is interrupted.");
     } else {
       System.out.println ("Succ: The transaction is executed successfully.");
      System.out.println("str_key1: {}, str_key2: {}", jedis.get(str_key1), jedis.get(str_key2));
     jedis.close();
  }
}
```

# 5.14 Retry Mechanism for Clients Accessing GeminiDB Redis API

The retry mechanism for clients accessing GeminiDB Redis API can ensure high availability and stability of applications if the network is unstable or a server is temporarily faulty.

There may be the following temporary faults.

| Cause                      | Description  |
|----------------------------|--|
| HA is triggered.           | GeminiDB Redis API automatically monitors node health. If a<br>node breaks down, a primary/standby switchover or shard<br>takeover is automatically triggered. Generally, HA may be<br>triggered when:   |
|                            | <ul> <li>A GeminiDB process on a node restarts due to OOM or<br/>hardware faults.</li> </ul>   |
|                            | <ul> <li>Nodes are automatically removed or added when they are<br/>scaled or specifications are changed.<br/>In these scenarios, clients may be intermittently<br/>disconnected in seconds or commands time out.</li> </ul>                           |
| The network<br>fluctuates. | Complex network environments between clients and servers<br>running GeminiDB Redis instances may cause problems such<br>as occasional network jitter and data retransmission. In this<br>case, requests initiated by the clients may temporarily fail. |
| Servers are overloaded.    | Requests initiated by clients may not be responded<br>immediately due to heavy loads and slow queries on<br>GeminiDB Redis servers. As a result, the requests time out.  |

When setting the retry mechanism for clients, follow the rules below.

| Rule   | Description  |
|--|--|
| Configure a<br>proper interval<br>and retry times. | Configure a proper interval and retry times based on business<br>requirements. If an excessive number of retries are<br>attempted, it takes a longer time to recover from a fault. If<br>the interval between retries is shorter than expected, servers<br>may become overwhelmed. In heavy-load scenarios, you are<br>advised to increase the retry interval exponentially to prevent<br>server breakdown due to a large number of concurrent<br>retries.   |
| Retry only<br>idempotent<br>operations.            | Commands have been executed on a server, but a timeout<br>occurs when the result is returned to a client. In this case, the<br>commands may be executed repeatedly. Therefore, you are<br>advised to retry only idempotent operations (for example, the<br><b>SET</b> command), and the result remains unchanged after<br>multiple operations. For non-idempotent operations (for<br>example, the <b>INCR</b> command), you need to confirm whether<br>duplicate data can be tolerated, and multiple operations may<br>increase a counter value. |
| Generate client<br>logs.                           | You are advised to configure the system to generate client<br>logs during the retry process, such as the connected IP<br>address and port number, error commands, and keys, to<br>facilitate troubleshooting.  |

The following SDK code examples are used for reference only.

# Jedis (Java Client)

When JedisPool is used, Jedis 4.0.0 or later supports retries. The following uses Jedis 4.0.0 as an example to describe how to set **MAX\_ATTEMPTS** to a proper number of retry attempts. package nosql.cloud.huawei.jedis;

```
import redis.clients.jedis.DefaultJedisClientConfig;
import redis.clients.jedis.HostAndPort;
import redis.clients.jedis.JedisClientConfig;
import redis.clients.jedis.UnifiedJedis;
import redis.clients.jedis.providers.PooledConnectionProvider;
import java.time.Duration;
// UnifiedJedis API supported in Jedis >= 4.0.0
public class UnifiedJedisDemo {
  private static final int MAX_ATTEMPTS = 5;
  private static final Duration MAX_TOTAL_RETRIES_DURATION = Duration.ofSeconds(15);
  public static void main(String[] args) {
     // Basic connection config
     JedisClientConfig jedisClientConfig = DefaultJedisClientConfig.builder().password("xxx").build();
     // Implement retry
     PooledConnectionProvider provider = new
        PooledConnectionProvider(HostAndPort.from("{ip}:{port}"), jedisClientConfig);
    //Sets MAX_ATTEMPTS to a proper number of retry attempts.
     UnifiedJedis jedis = new UnifiedJedis(provider, MAX_ATTEMPTS, MAX_TOTAL_RETRIES_DURATION);
     trv {
        System.out.println("set key: " + jedis.set("key", "value"));
     } catch (Exception e) {
        // Signifies reaching either the maximum number of failures,
        MAX_ATTEMPTS, or the maximum query time, MAX_TOTAL_RETRIES_DURATION
        e.printStackTrace();
     }
  }
```

# Redisson (Java Client)

The following is an example of how to set **RETRY\_ATTEMPTS** and **RETRY\_INTERVAL** to the number of retry attempts and an interval.

```
package nosql.cloud.huawei.jedis;
import org.redisson.Redisson;
import org.redisson.api.RBucket;
import org.redisson.api.RedissonClient;
import org.redisson.config.Config;
public class RedissonDemo {
  private static final int TIME_OUT = 3000;
  private static final int RETRY_ATTEMPTS = 5;
  private static final int RETRY_INTERVAL = 1500;
  public static void main(String[] args) {
     Config config = new Config();
     config.useSingleServer()
         .setPassword("xxx")
         .setTimeout(TIME OUT)
         .setRetryAttempts(RETRY_ATTEMPTS)
        .setRetryInterval(RETRY_INTERVAL) // Sets a proper number of retry attempts and retry
interval.
         .setAddress("redis://{ip}:{port}");
     RedissonClient redissonClient = Redisson.create(config);
     RBucket<String> bucket = redissonClient.getBucket("key");
     bucket.set("value");
```

} }

# Go-redis (Go Client)

The following is an example of how to set MaxRetries, MinRetryBackoff, and MaxRetryBackoff to the number of retry attempts and intervals.

package main

```
import (
     "context"
     "fmt"
     "time"
     "github.com/redis/go-redis/v9"
var ctx = context.Background()
func main() {
   client := redis.NewClient(&redis.Options{
      Addr: "localhost:6379",
      Password: "", // no password set
             0, // use default DB
      DB:
      MaxRetries: 3, // set max retry times
      MinRetryBackoff: time.Duration(1) * time.Second, // set retry interval
MaxRetryBackoff: time.Duration(2) * time.Second, // set retry interval
   })
   // Execute command
   err := client.Set(ctx, "key", "value", 0).Err()
   if err != nil {
      panic(err)
   }
   // Test
   pong, err := client.Ping(ctx).Result()
   if err != nil {
      fmt.Println("Failed:", err)
      return
   fmt.Println("Success:", pong)
```

# Redis-py (Python Client)

}

The following is an example of how to set a proper number of retry attempts by setting parameters of the Retry function.

```
import redis
from redis.retry import Retry
from redis.exceptions import ConnectionError
from redis.backoff import ExponentialBackoff
from redis.client import Redis
from redis.exceptions import (
  BusyLoadingError,
  ConnectionError,
  TimeoutError
# Run 3 retries with exponential backoff strategy
retry_strategy = Retry(ExponentialBackoff(), 3)
# Redis client with retries
client = redis.Redis(
host = 'localhost',
```

```
port = 6379,
  retry = retry_strategy,
  # Retry on custom errors
  retry_on_error = [BusyLoadingError, ConnectionError, TimeoutError],
  # Retry on timeout
  retry on timeout = True
try:
  client.ping()
  print("Connected to Redis!")
except ConnectionError:
  print("Failed to connect to Redis after retries.")
try:
  client.set('key', 'value')
  print("Set key and value success!")
except ConnectionError:
  print("Failed to set key after retries.")
```

## Hiredis (C Client)

Hiredis is a minimalistic C client library and does not provide a preset automated retry mechanism. You need to manually compile the logic.

The following is a simple example of how to implement an automated connection retry in a loop and with a delay, similar to command retry settings.

```
#include <hiredis/hiredis.h>
#include <stdio.h>
#include <unistd.h>
redisContext* connect_with_retry(const char *hostname, int port, int max_retries, int retry_interval) {
  redisContext *c = NULL;
  int attempt = 0;
  //Uses a loop and delay to implement automatic retries during connection establishment.
  while (attempt < max_retries) {
     c = redisConnect(hostname, port);
     if (c != NULL && c->err == 0) {
        printf("Connection success!\n");
        return c;
     }
     if (c != NULL) {
        printf("Connection error: %s\n", c->errstr);
        redisFree(c);
     } else {
        printf("Connection failed\n");
     }
     printf("Retrying in %d seconds...\n", retry_interval);
     sleep(retry_interval);
     attempt++;
  }
  return NULL;
}
int main() {
  const char* hostname = "127.0.0.1";
  int port = 6379;
  int max_retries = 5;
  int retry_interval = 2;
  redisContext *c = connect_with_retry(hostname, port, max_retries, retry_interval);
  if (c == NULL) \{
     printf("Failed to connect to Redis after %d attempts\n", max_retries);
     return 1;
```
} redisFree(c); return 0;

# 5.15 GeminiDB Redis API Pub/Sub

Huawei Cloud GeminiDB Redis is fully compatible with Pub/Sub of open-source Redis. This section describes how to configure this model.

# Pub/Sub

**SUBSCRIBE**, **UNSUBSCRIBE**, and **PUBLISH** implement the **publish-subscribe pattern**. In this pattern, publishers do not directly send messages to a specific subscriber but publish them to a channel. All subscribers who are interested in this channel can receive the messages. Pub/Sub enables the decoupling of the publisher and subscribers, eliminating the need for publishers to know their subscribers.

# **Application Scenarios**

Pub/Sub plays an important role in many scenarios, for example:

• Real-time chat

In IM applications, messages need to be quickly transferred. With Pub/Sub, users can subscribe to their own chat channels. After message are published, the subscribes on this channel receive the messages immediately. In this manner, real-time performance and high efficiency can be achieved.

• Real-time notification system

On e-commerce websites or social media platforms, users need to receive notifications such as order status updates, comments, and likes in real time. With Pub/Sub of GeminiDB Redis API, the system can immediately publish a notification when the status changes, and all related users will receive the notification in a timely manner.

• Monitoring and log system

In the microservice architecture, the Pub/Sub model can be used for status monitoring and log collection between services. Services can publish status information or log messages to specific channels. The monitoring service or log collection service can subscribe to these channels to implement real-time monitoring and data collection.

• Real-time gaming messages

In an online game, data between players each time an action occurs needs to be synchronized in time. Pub/Sub can be used for message transfer and game event notification to ensure that all players receive status updates at the same time.

• Data stream processing

Real-time processing and analysis are key to data stream applications. With Pub/ Sub, data producers can publish data streams, and consumers can subscribe to these streams for real-time processing and analysis.

Basic operations

For example, to subscribe to "channel11" and "ch:00," clients can run the following command:

SUBSCRIBE channel11 ch:00

These clients will receive messages on these channels from other clients in the sequence in which the messages were sent.

Advanced function:

Pub/Sub supports pattern matching. Clients may subscribe to glob-style patterns to receive all the messages sent to channel names matching a given pattern. For example:

PSUBSCRIBE news.\*

Subscribers will receive all messages sent to channels such as news.art.figurative and news.music.jazz.

## 

- **Message loss**: Pub/Sub does not ensure message durability. Therefore, messages may be lost when the network is faulty or a subscriber is not connected.
- **Performance**: In a high-concurrency environment, the Pub/Sub performance may be limited. Performance need to be tested and improved based on specific scenarios.
- If both SUBSCRIBE and PSUBSCRIBE are executed, duplicate messages may be received. Check whether the business logic is correct.

# Java Sample Code (Jedis)

# Message Publisher

```
import redis.clients.jedis.Jedis;
public class GeminiDBPubClient {
  private Jedis jedis;
  public GeminiDBPubClient(String ip, int port, String password){
     jedis = new Jedis(ip, port);
     // The instance password for GeminiDB.
     String authString = jedis.auth(password);
     if (!authString.equals("OK"))
     {
        System.err.println("AUTH Failed: " + authString);
        return;
     }
  }
  public void pub(String channel, String message){
     System.out.println(" >>> Publish > Channel: " + channel + " > Sent Message: " + message);
     jedis.publish(channel, message);
```

```
public void close(String channel){
   System.out.println(" >>> Publish End > Channel:" + channel + " > Message:quit");
   // The message publisher has finished sending, sending a "quit" message.
   jedis.publish(channel, "quit");
}
```

# **Message Subscriber**

}

```
import redis.clients.jedis.Jedis;
import redis.clients.jedis.JedisPubSub;
public class GeminiDBSubClient extends Thread {
  private Jedis jedis;
  private String channel;
  private JedisPubSub listener;
  public GeminiDBSubClient(String ip, int port, String password){
     jedis = new Jedis(host,port);
     // The instance password for GeminiDB.
     String authString = jedis.auth(password); //password
     if (!authString.equals("OK"))
     {
        System.err.println("AUTH Failed: " + authString);
        return;
     }
  }
  public void setChannelAndListener(JedisPubSub listener, String channel){
     this.listener=listener;
     this.channel=channel;
  }
  private void subscribe(){
     if(listener==null || channel==null){
        System.err.println("Error:SubClient> listener or channel is null");
     System.out.println(" >>> Subscribe > Channel:" + channel);
     // The receiver will block the process while listening for subscribed messages until it receives a "quit"
message (passive mode) or actively cancels the subscription.
     jedis.subscribe(listener, channel);
  }
  public void unsubscribe(String channel){
     System.out.println(" >>> Unsubscribe > Channel:" + channel);
     listener.unsubscribe(channel);
  }
  @Override
  public void run(){
     try {
        System.out.println("-----Subscribe Start-----");
        subscribe();
        System.out.println("-----Subscribe End-----");
     } catch(Exception e){
        e.printStackTrace();
     }
  }
}
```

# **Message Listener**

import redis.clients.jedis.JedisPubSub; public class GeminiDBListener extends JedisPubSub { @Override public void onMessage(String channel, String message) { System.out.println(" <<< Subscribe < Channel:" + channel + " > Receive Message:" + message );

```
// When the received message is "quit," unsubscribe (passive mode).
     if(message.equalsIgnoreCase("quit")){
       this.unsubscribe(channel);
     }
  }
  @Override
  public void onPMessage(String pattern, String channel, String message) {
     // TODO Auto-generated method stub
  @Override
  public void onSubscribe(String channel, int subscribedChannels) {
     // TODO Auto-generated method stub
  }
  @Override
  public void onUnsubscribe(String channel, int subscribedChannels) {
     // TODO Auto-generated method stub
  }
  @Override
  public void onPUnsubscribe(String pattern, int subscribedChannels) {
     // TODO Auto-generated method stub
  }
  @Override
  public void onPSubscribe(String pattern, int subscribedChannels) {
     // TODO Auto-generated method stub
  }
}
```

# **6** Best Practices

6.1 Automated Database Access Using an Account for Multitenancy Management of GeminiDB Redis Instances

- 6.2 FastLoad for RTA-based Ad Placement
- 6.3 PITR for Restoring Gaming Data
- 6.4 ExHash for Ad Frequency Control
- 6.5 GeminiDB Redis API for Instant Messaging
- 6.6 Implementing Distributed Locks Using Lua Scripts for GeminiDB Redis API
- 6.7 Suggestions on Alarm Rules of GeminiDB Redis Instance Metrics
- 6.8 GeminiDB Redis API for Product Correlation Analysis
- 6.9 GeminiDB Redis API for Online Classroom
- 6.10 GeminiDB Redis API for Session Management in Web Applications

# 6.1 Automated Database Access Using an Account for Multitenancy Management of GeminiDB Redis Instances

GeminiDB Redis API continuously provides enhanced features for enterprises, one of which is multitenancy. With multitenancy, read-only accounts and read/write accounts can be added, and databases accessible to each account can be specified. This prevents misoperations on data of other tenants. This feature allows multiple tenants to use the same Redis instances while keeping their data isolated, facilitating development and management for enterprises.

# **Application Scenarios**

Multitenancy is a common function of database users. For example, an enterprise has service departments A and B, both of which need to use Redis to store their own data. If multitenancy is not used, data of departments A and B will be mixed. As a result, data breaches and misoperations may occur. After multitenancy is enabled, data of departments A and B can be stored in different Redis instances or databases, and permissions on these instances or databases can be controlled to ensure data security and reliability.

Multi-tenant databases usually have some standard attributes, such as read/write permission control as well as cross-database authentication and isolation. GeminiDB Redis instances use such comprehensive multitenancy technologies, allowing for read/write permission control and database isolation.

# Advantages

In contrast to multi-tenant databases, open-source Redis supporting an access control list (ACL) in its new version only grants accounts read-only and read/write permissions. Each account can still be used to view all databases. For example, a development engineer wants to use database 1 but accidentally clears another engineer's database 0, causing a production accident. Permission isolation of GeminiDB Redis API can avoid this problem. For example, if engineer A has only the permission of database 1, database 0 is not affected even if misoperations are performed.

In addition, multitenancy of open-source Redis can be used only on a single node. Once the service volume increases and a cluster is required, multiple databases are unavailable. As a result, only database 0 is left. More than 1,000 GeminiDB Redis databases can be deployed in a cluster, and more than 200 ACL sub-accounts can be created.

**Table 6-1** Comparison of permission management capabilities between opensource Redis and GeminiDB Redis API

| Product               | Account Read<br>and Write<br>Permission<br>Control | Account<br>Permission<br>Isolation | Multi-DB<br>Cluster | Default<br>Quantity<br>of<br>Supported<br>Databases |
|-----------------------|--|------------------------------------|---------------------|---|
| Open-source<br>Redis  | Supported  | Supported                          | Not supported       | 16  |
| GeminiDB<br>Redis API | Supported  | Supported                          | Supported           | 1,000   |

# Solution

To use the tenant management function of GeminiDB Redis instances, you need to create accounts on the account management page and set read-only and read/ write permissions for each account. For details, see **Managing Accounts**.

After an account is created, you can run **auth USER PWD** or **auth USER:PWD** for authentication and execute the SELECT DB statement to access the database on which the account has permissions. For details, see **Enabling Database Access With a Password**.

# 6.2 FastLoad for RTA-based Ad Placement

# **Scenarios**

Advertisement (ad) placement is indispensable for enterprise promotion and marketing, especially for new media facing fierce competition. Now there are diversified advertising channels and detailed tailored ad placement.

Customer's increased awareness of ad ROI demands precise audience selection. For example, on a short video platform, advertisers have to configure rules on placing ads such as age, gender, and education background of audiences. The inflexibility hinders advertisers from making decisions on ad placement, and hundreds of millions or even billions of advertising fees need to be paid every year. However, it is still difficult to accurately reach target audiences. To let advertisers have autonomy to deliver or reject each ad request, Real-Time API (RTA) rose to the challenge.

RTA is used to meet the real-time personalized delivery needs of advertisers.

# Challenges

Advertisers' RTA system read data from core profile databases, helping advertisers make placement decisions. The newer the data, the better the placement effect. Therefore, the latest data generated by a big data platform needs to be written into profile databases in a timely manner. To address high concurrency, ultra-low latency, and ultra-large volumes of RTA requests, core profile databases must have the following features:

# • Quick import of vast amount of data; accurate decision-making

Hundreds of GB or even several TB of all profile data needs to be periodically imported to profile databases. The faster the data is imported, the more accurate the model is, and the better ad placement effect is.

# • High-concurrency access

The RTA system needs to handle a large number of real-time bidding requests. For example, hundreds of thousands to millions of QPS are sent by RTA systems of e-commerce and financial customers.

# • Stable low latency

The media asks advertisers to provide decisions within 40 ms to 100 ms. Databases need to execute requests within single-digit milliseconds.

• Low cost

To achieve ultimate performance, open-source self-hosted Redis needs to be installed for handling RTA requests. However, expensive TB-level data storage devices give advertisers a dilemma in selecting suitable models.

In an RTA system, common profile databases have the following problems:

- MySQL: It is difficult to handle hundreds of thousands to millions of concurrent QPS at low latency.
- MongoDB/HBase: It is inexpensive to store TB-level data, but MongoDB and HBase instances cannot maintain a stable low latency. High timeout rate may cause project suspension, injurious to commercial interests.

• In-memory database: For example, open-source self-hosted Redis is widely used in the industry. In-memory databases provide ultimate performance with high concurrency and low latency. However, there are risks such as poor stability and data loss. It costs too much and takes long to import TB-level user profile data.

GeminiDB Redis API features stability, low latency, and cost-effectiveness, and data can be imported extremely fast using FastLoad.

# **Solution Overview**

In an RTA system, storage of GeminiDB Redis instances costs less than that of open-source self-hosted Redis instances. FastLoad enables quick offline data import. GeminiDB Redis API ensures stability and low latency and has rich practice cases of online advertising and recommendation services.

# Advantages

# • 5 to 10 times faster data import by FastLoad

Traditional databases can only write data one by one using standard protocols. Data is crunched by the compute layer through a complex process and then written to the storage layer. Therefore, it usually takes hours or days for a big data platform to periodically import hundreds of GB or even several TB of profile data, which has a great impact on online services.

FastLoad, an enterprise-level feature of GeminiDB Redis API, improves the data import speed by 5 to 10 times and reduces the impact on online services. To address RTA requests, the big data platform handles highly-concurrent workloads, FastLoad directly transfers massive volumes of data into a storage engine of the GeminiDB Redis instance via a dedicated high-speed persistent channel, and the storage engine on the GeminiDB Redis database orchestrates the data.

# • Millions of concurrent requests and latency in sub-milliseconds

GeminiDB Redis API uses a separated storage and compute architecture. Three copies of data are stored in a distributed shared storage pool. All nodes support efficient reads and writes, and compute power can be scaled up and out, making it easy to cope with workload spikes.

With a multi-thread architecture, high-performance storage pool, and indepth optimization of the memory data structure and access algorithm, GeminiDB Redis API can respond to requests in sub-milliseconds. Such an ultra-low latency is critical to real-time data processing and analysis, especially in scenarios such as online games, financial technologies, advertising systems, and real-time recommendation systems. Therefore, GeminiDB Redis API is ideal for large-scale real-time interaction and highfrequency transactions.

With vast experience on the live network, GeminiDB Redis API can handle more than one million QPS while ensuring average latency of 1 ms and p99 latency of 2 ms.

• Efficient data compression and storage at low costs

By compressing both logical and block data, GeminiDB Redis API greatly reduces storage resource consumption while maintaining ultimate performance. Compute and storage resources are decoupled from each other, so they can be flexibly scaled out. Achieving optimized resource utilization means a range of benefits for enterprises, most notably saving costs in storing data.

With vast experience on the live network, GeminiDB Redis API has a compression ratio of 4:1. That is, only about 3 TB out of 12 TB of data is occupied on a GeminiDB Redis instance.

# 6.3 PITR for Restoring Gaming Data

# **Application Scenarios**

Databases may experience faults such as data corruption, loss, or accidental deletion. To ensure services are running properly, you need to restore a database to a normal state before the faults occurred. Traditional databases adopt a periodic backup policy. That is, data is restored when the system is faulty. Data restoration takes a long time. As a result, customer services are severely affected.

# **Solution Overview**

Point-in-Time Recovery (PITR) allows you to restore the database to a particular point in time if data is lost or corrupted due to misoperations or accidental deletion.

Some game players may exploit vulnerabilities to duplicate equipment and currency, leading to unfairness. Traditional databases are backed up once a day, making it difficult to restore data to a specific point in time. With PITR of GeminiDB Redis API, you can specify a specific time point for data restoration. Data can be restored within 5 minutes at least.

# Advantages

PITR of GeminiDB Redis API maintains your data from past timestamps and does not affect data snapshots. If there is a fault, data can be restored to a specified point within 5 minutes. Therefore, GeminiDB Redis AP is widely used in industries such as gaming and finance.

# • Backup tasks are not affected, and services are running stably.

PITR does not affect data backup and access.

GeminiDB Redis API enables you to create snapshots by recording the file system status instead of by copying files. File metadata (such as data block information and addressing information) at the current moment is stored to generate snapshots. Therefore, services are not affected during snapshot creation.

• Data is restored in minutes regardless of its volume.

PITR snapshots can be stored on your local PC and do not need to be uploaded to cold storage media. Therefore, data replication and migration are not involved, and data can be restored anytime.

Even hundreds of GB of data can be restored within 5 minutes. After being restored to a specified point in time, data can also be restored multiple times to a state before or after that point in time.

# • GeminiDB Redis API has better backup performance than open-source Redis

Open-source Redis uses copy-on-write (CoW) to effectively enable snapshot persistence in its multi-thread architecture. When Redis calls fork() to create a child process, its parent process will be blocked for hundreds of milliseconds. As a result, jitter will occur. CoW may cause memory overuse. If a large number of writes are performed while the parent process is forked, memory is severely wasted or even OOM occurs (memory usage < 50%). PITR frees you from copying or migrating data, so services are not affected. Snapshots can be quickly created, and data can be restored stably and securely.

# Solution

For details about how to enable PITR of the GeminiDB Redis API and restore data to a specified time point, see **4.8.3 Restoring to the Original Instance Using PITR**.

# 6.4 ExHash for Ad Frequency Control

ExHash is an enhanced hash data structure that allows users to specify expiration times and version numbers for fields. ExHash is flexible and can help simplify business development in most scenarios.

This section describes how to use ExHash commands of GeminiDB Redis API to simplify business development of frequency control and shopping cart.

# **ExHash Commands**

For details, see **ExHash commands**.

# **Application Scenarios**

• Frequency control

Frequency control allows users to restrict the number of operations performed within a certain period (for example, one day, one week, or one month), and limit the number of times an ad or information displayed on a platform within a specified period. This helps prevent overexposure and ad fatigue, optimizes ad performance, improves the conversion rate, and avoids malicious activities, such as manipulating online traffic, comments, and likes.

There are three elements of frequency control: user ID (key), ad ID (field), and the number of times an ad is pushed (value) within a specified period. There are three ways to configure ad frequency control policies.

# Figure 6-1 Hash

| Key    | key TTL  | Field | Value | Key    | key TTL    | Field | Value     | Key    | key TTL     | Field    | Value | field TTL |
|--------|----------|-------|-------|--------|------------|-------|-----------|--------|-------------|----------|-------|-----------|
|        |          | AD_1  | 1     |        | one week   | AD_1  | 1#one day |        | _1 one week | AD_1     | 1     | one day   |
| User_1 | one day  | AD_2  | 2     | User_1 |            | AD_2  | 2#8 hours | User_1 |             | AD_2     | 2     | 8 hours   |
|        | AD_3 1   |       |       | AD_3   | 1#one week |       |           | AD_3   | 1           | one week |       |           |
| User_2 |          |       |       | User_2 |            |       |           | User_2 |             |          |       |           |
|        |          |       |       |        |            |       |           |        |             |          |       |           |
|        | Hash com | mands | ; 1   |        | Hash c     | ommai | nds 2     |        | ExHash      | comma    | inds  |           |

- In hash commands 1, the **expire** command sets the expiration time of User\_1 to one day. The hincrby command increases and records how many times an ad is pushed. Before an ad is pushed, the hget command obtains how many times it was pushed the previous day, so that users can determine whether to continue pushing it. One day later, user data automatically expires and does not need to be manually cleared. In this way, frequency control can be implemented. However, only one expiration time point can be set for each user (key). It is not possible to set flexible frequency control policies for a specified period, for example, three pushes within an eight-hour window.
- By running hash commands 2, users can specify timestamps for values.
   However, the workload of business development may increase.
- ExHash is better than hash because ExHash allows users to specify the expiration time for each field. In the frequency control scenario, GeminiDB Redis API allows you to configure a unique push frequency for each ad and in different time segments. Assume that the frequency control policy configured for AD\_2 is twice within 8 hours. Before pushing AD\_2 to User\_1, you can obtain the value 2 by running the EXHGET command, AD\_2 will not be pushed to User\_1. After eight hours, AD\_2 for User\_1 expires and the field information cannot be obtained by EXHGET. In this case, AD\_2 will be pushed to User\_1 again.
- Shopping cart

The following describes and compares several types of Redis commands for a shopping cart.

a. String

The shopping cart works easily with string commands. The platform combines the user ID and item ID as a key, for example, **User\_1#Earphones\_1**. The key value is the number of items to be purchased. There is an expiration time for items on flash sales.

## Figure 6-2 String

| Кеу                | Value | ΠL     |
|--------------------|-------|--------|
| User_1#Earphones_1 | 1     | null   |
| User_1#Keyboard_1  | 2     | time_1 |
| User_1#Charger_2   | 1     | time_2 |
| User_2#            |       |        |
|                    |       |        |

## String

# Related commands incrby User\_N#Product\_N [Number] #Increases the product quantity. set User\_N#Product\_N [Number] #Sets the item quantity. expire User\_N#Product\_N Time\_N # Sets the expiration time of a specified item in the shopping cart of a specified user. get User\_N#Product\_N #Obtains the product quantity. scan 0 match User\_N\* # Queries all items of User\_N. del User\_N#Product\_N #Deletes a specified item from the shopping cart of a specified user.

- Possible issues are as follows:
  - Extra splicing increases the encoding and decoding development workload.
  - To obtain the shopping item list, adding a prefix to the SCAN command can scan all keys. GET is used to obtain the key values.
  - To obtain the list length, the number of prefix keys need to be scanned.
  - There are a large number of duplicate username prefixes occupying the storage space.
- b. Hash

A user ID is used as the key and an item ID as the field. The value is the number of items in the shopping cart. For items in flash sales, the expiration time is combined to the value of the field.

### Figure 6-3 Hash

| Key    | Field       | Value    |
|--------|-------------|----------|
|        | Earphones_1 | 1        |
| User_1 | Keyboard_1  | 2        |
|        | Charger_2   | 1#time_1 |
| User_2 |             |          |
|        |             |          |



### Related commands hset User\_N Product\_N [Number#Time\_N] # Sets the quantity and expiration time of a specified item in the shopping cart of a specified user. hincrby User\_N Product\_N [Number] # Adds the number of a specified item to the shopping cart of a specified user. hget User\_N Product\_N # Obtains information about a specified item in the shopping cart of a specified user. hgetall User\_N #Obtains all item information of a specified user. hlen User\_N # Obtains the number of items in the shopping cart of a specified user. hdel User\_N Product\_N #Deletes a specified item from the shopping cart of a specified user.

- Hash is better than string in the following ways:
  - Only one HGETALL command is required to obtain the shopping cart list of a user.
  - The HLEN command can be used to obtain the item list length of a user.
  - There are few duplicate username prefixes.

However, the solution is complex for processing items in flash sales. For example, If the quantity for **Keyboard\_1** of **User\_1** needs to be added instead of using the HINCRBY command directly, you should obtain the value of **Keyboard\_1** by executing the HGET command first and decode the value. Then, specify the quantity to be added and encode the value using HSET.

c. ExHash

A user ID is used as the key and an item ID as the field. The value is the number of items in the shopping cart. You can run HSET to set expiration times for fields of items in flash sales.

### Figure 6-4 ExHash

| Key    | Field       | Value | TTL    |
|--------|-------------|-------|--------|
|        | Earphones_1 | 1     | null   |
| User_1 | Keyboard_1  | 2     | null   |
|        | Charger_2   | 1     | time_1 |
| User_2 |             |       |        |
|        |             |       |        |

### ExHash

Related commands exhset User N Product N ex Time N # Sets the quantity and expiration time of a specified item in the shopping cart of a specified user. exhincrby User\_N Product\_N [Number] keepttl # Adds the quantity of specified items in the shopping cart of a specified user and retains the original expiration time. exhget User\_N Product\_N # Obtains information about a specified item in the shopping cart of a specified user. exhgetall User\_N #Obtains all item information of a specified user. exhlen User N # Obtains the number of items in the shopping cart of a specified user. exhdel User\_N Product\_N #Deletes a specified item from the shopping cart of a specified user. del User\_N #Empties the shopping cart of a specified user.

ExHash is better than hash because ExHash allows users to specify the expiration time for each field. ExHash and hash commands have similar syntax. ExHash is easy to use and can avoid heavy workloads in code modifications.

# Code Example for Ad Frequency Control

import redis import datetime import os def get\_cur\_time(): return "[" + datetime.datetime.utcnow().strftime('%Y-%m-%d %H:%M:%S.%f')[:-3] + "]" def get\_redis():

This method is used to connect to a Redis instance.

- \* host: Instance connection address.
- \* port: Port of the instance. The default value is 6379.
- \* password: Password for connecting to the instance.

# 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.

password = os.getenv('EXAMPLE\_PASSWORD\_ENV')

return redis.Redis(host='\*\*\*', port=6379, password=password)

"Global frequency control policy. Display ad 1 for up to 2 times within three seconds and ad 2 for five times within five seconds.

```
frequency_stratege = {"ad_1" : [2, 3], "ad_2" : [5, 5]}
```

def push\_ad\_to\_user(userId: str, adId: str): This method is used to push a specified ad to a specified user. \* userId: User ID. \* adId: Ad ID. # If no frequency control policies are set for an ad, directly push the ad to the user. if adId not in frequency\_stratege: print("no need control frequency, push ", adld, "to", userId) return True # Obtain how many times an ad is pushed for a user by user ID and ad ID. # Syntax: EXHGET Key Field cnt = get\_redis().execute\_command("EXHGET " + userId + " " + adId) # If an ad has not been pushed to a user, directly push the ad to the user. if cnt == None: # Syntax: EXHINCRBY Key Field num [EX time] # Usage description: EXHINCRBY User ID Ad ID Push times (1) Expiration time of the ad cmd = "EXHINCRBY " + userId + " " + adId + " 1 EX " + str(frequency\_stratege[adId][1]) cur\_cnt = get\_redis().execute\_command(cmd) print(get\_cur\_time(),"push", adld, "to", userld, "first time during", str(frequency\_stratege[adld][1]), "seconds") return True # The result returned from Redis Python client is in bytes. Convert the result to a string and then to an integer. cnt = int(cnt.decode("utf-8")) if cnt < frequency\_stratege[adId][0]: # Syntax: EXHINCRBY Key Field num KEEPTTL (Retains the original expiration time of the field.) cmd = "EXHINCRBY " + userId + " " + adId + " 1 KEEPTTL" cur\_cnt = get\_redis().execute\_command(cmd) print(get\_cur\_time(), "push", adId, "to", userId, "current cnt:", cur\_cnt) return True print(get\_cur\_time(), "Control frequency, can't push", adId, "to", userId, ", max cnt:", frequency\_stratege[adId][0]) return False if \_\_name\_\_ == "\_\_main\_\_": for i in range(3): push\_ad\_to\_user("usr\_1", "ad\_1") for i in range(6): push\_ad\_to\_user("usr\_1", "ad\_2") for i in range(3): push\_ad\_to\_user("usr\_1", "ad\_1") for i in range(12): push\_ad\_to\_user("usr\_1", "ad\_2")

The script output is as follows:

The Python script executes slowly, and the expiration time of ad 2 is set to 5 seconds. Ad 2 can thus be successfully pushed to the user at December 15, 2023 07:09:56.530, 5 seconds after the first push time of December 15, 2023 07:09:51.349.

```
[2023-12-15 07:09:50.086] push ad_1 to usr_1 first time during 3 seconds
[2023-12-15 07:09:50.503] push ad_1 to usr_1 current cnt: 2
[2023-12-15 07:09:50.794] Control frequency, can't push ad_1 to usr_1 , max cnt: 2
[2023-12-15 07:09:51.349] push ad_2 to usr_1 first time during 5 seconds
[2023-12-15 07:09:51.745] push ad_2 to usr_1 current cnt: 2
[2023-12-15 07:09:52.128] push ad_2 to usr_1 current cnt: 3
[2023-12-15 07:09:52.889] push ad_2 to usr_1 current cnt: 4
[2023-12-15 07:09:53.417] push ad_2 to usr_1 current cnt: 5
[2023-12-15 07:09:53.632] Control frequency, can't push ad_2 to usr_1, max cnt: 5
[2023-12-15 07:09:54.120] push ad_1 to usr_1 first time during 3 seconds
[2023-12-15 07:09:54.769] push ad_1 to usr_1 current cnt: 2
[2023-12-15 07:09:54.915] Control frequency, can't push ad_1 to usr_1 , max cnt: 2
[2023-12-15 07:09:55.211] Control frequency, can't push ad_2 to usr_1 , max cnt: 5
[2023-12-15 07:09:55.402] Control frequency, can't push ad_2 to usr_1 , max cnt: 5
[2023-12-15 07:09:55.601] Control frequency, can't push ad_2 to usr_1 , max cnt: 5
[2023-12-15 07:09:55.888] Control frequency, can't push ad_2 to usr_1, max cnt: 5
[2023-12-15 07:09:56.087] Control frequency, can't push ad_2 to usr_1 , max cnt: 5
[2023-12-15 07:09:56.530] push ad_2 to usr_1 first time during 5 seconds
[2023-12-15 07:09:57.133] push ad_2 to usr_1 current cnt: 2
```

```
[2023-12-15 07:09:57.648] push ad_2 to usr_1 current cnt: 3
[2023-12-15 07:09:58.107] push ad_2 to usr_1 current cnt: 4
[2023-12-15 07:09:58.623] push ad_2 to usr_1 current cnt: 5
[2023-12-15 07:09:58.865] Control frequency, can't push ad_2 to usr_1, max cnt: 5
[2023-12-15 07:09:59.096] Control frequency, can't push ad_2 to usr_1, max cnt: 5
```

This section describes the features, usage, and application scenarios of the ExHash commands provided by GeminiDB Redis API. ExHash and native Redis hash have similar syntax but isolated mechanisms. ExHash allows you to specify expiration times and version numbers for fields. GeminiDB Redis API is dedicated to developing more easy-to-use enterprise-class features, helping you simplify O&M and improve development efficiency.

# 6.5 GeminiDB Redis API for Instant Messaging

# Context

Instant messaging (IM) works by connecting two or more people through a messaging platform over a network. Once connected, users can send text messages, files, even make voice and video calls. In the highly information-based mobile Internet era, IM products (such as WeChat and QQ) have become a must-have item in our life. The core of an IM system is a messaging system, which is used for synchronization, retrieval, and storage of messages.

- Message synchronization: Messages are completely and quickly sent from the sender to the receiver. The most important metrics of a message synchronization system are instantaneity, sequentiality, and integrity of transmitted messages, and the size of messages that can be supported.
- Message storage: The persistent storage of messages. Conventional message systems store messages on premises on a client, and data is not reliable. Modern message systems store messages on the cloud. This is the so-called "message roaming". You can log in to your account at any terminals to view all historical messages.
- Message retrieval: Messages are generally text. Therefore, full-text retrieval is also a mandatory capability. Conventional message systems usually create indexes based on local messages and support local retrieval. Modern message systems support online message storage and index creation while data is stored, providing comprehensive retrieval functions.

# **Application Scenarios**

IM systems can be used in many industries, such as chatting, gaming, and intelligent customer service. Different industries have different requirements on the cost, performance, reliability, and latency of IM systems. These requirements need to be considered to achieve balance in architecture design.



# **IM System Architecture**

The basic concepts involved in IM system architecture design are as follows .

• Comparison between conventional and modern architectures



Figure 6-6 Comparison between conventional and modern architectures

Conventional architecture:

- Messages are synchronized before being stored.
- Messages are synchronized online and cached offline.
- Servers do not persist messages or support message roaming.
   Modern architecture:
- Messages are stored before being synchronized.

- Messages are stored and synchronized in different libraries. The storage library stores all conversations and supports message roaming. The synchronization library stores synchronized messages by receiver.
- Full-text retrieval is supported.

### • Comparison between read fan-out and write fan-out

A suitable read/write model ensures message reliability and consistency and effectively reduces workloads of servers or clients, which is critical to an IM system. This section describes two models: read fan-out and write fan-out.

Read fan-out

Figure 6-7 Read fan-out



The recipient needs to pull messages from multiple mailboxes.

Messages from users A1, A2, and A3 are stored in three different mailboxes (an abstract data structure used to store messages) of user B. User B has to read new messages from all the mailboxes every time. In read fan-out mode, every two associated users have a mailbox.

Advantages of read fan-out:

- No matter a one-on-one chat or a group chat is initiated, messages need to be written into recipient's mailbox once.
- Each mailbox stores two users' chat records, which can be easily viewed and searched for.

Disadvantages of read fan-out:

- As the volume of read operations increases, the system may face challenges in scaling to handle the load efficiently.

# Figure 6-8 Write fan-out



Users B1, B2, and B3 read messages only from their own mailboxes. They write or send messages in different ways for one-on-one chat and group chats.

- One-to-one chat: A message is written into both a sender's and a recipient's mailboxes. To view the chat history, another message needs to be written.
- Group chat: A sender needs to write a message to mailboxes of all group members. The group chat works in write fan-out mode, which consumes enormous resources. Therefore, a chat group can hold a maximum of 500 members.

Advantages of write fan-out:

- Users only need to read their own mailboxes.
- It is convenient to synchronizing messages between multiple terminals. Disadvantages of write fan-out:
- The system is subjected to heavy write loads, especially for group chats.
- Comparison among the push, pull, and push-pull modes

Figure 6-9 Push, pull, and push-pull modes



In the IM system, messages can be obtained in the following modes:

- Push: The server instantly pushes a new message to all clients. A
  persistent connection needs to be established between the client and the
  server to ensure real-time performance. The client only needs to receive
  and process the message. However the server does not know the
  message processing capability of the client, which may cause a data
  backlog.
- Pull: The client requests messages from the frontend. This mode is used to obtain historical messages. The interval for the client to obtain new messages is not preset. If the interval is too short, a large number of connections may fail to obtain data. If the interval is too long, data cannot be received in time.
- Push-pull: This hybrid mode integrates advantages of push and pull systems. The server pushes a new message notification to the frontend. After receiving the notification, the frontend pulls the message from the server.

# IM Technology Challenges





Messages between the clients are forwarded by servers. Core functions of IM are implemented by the message storage and synchronization libraries, which have high requirements on storage layer performance.

- Massive data storage: If messages need to be stored permanently, the data volume will grow gradually. The message storage library must support unlimited capacity expansion to cope with the increasing data volume.
- Low storage cost: Messages contain both hot and cold data. Hot data is generated in most queries. The cold tier has lower storage costs against increasing data volumes.
- Data life cycle management: The life cycle must be defined for message data storage and synchronization. The storage library stores data online. Generally, a long retention duration needs to be specified. The synchronization library is used for online or offline push in the write fan-out mode, and data is stored for a short period.
- High write throughput: The write fan-out mode is used in most IM systems, so storage hardware must offer enhanced write throughput to cope with message floods.
- Low-latency read: The messaging system is usually used online with high realtime performance. The read latency must be as low as possible.

# Advantages of GeminiDB Redis in IM Scenarios

At the heart of the IM system lies the storage layer, whose performance directly affects user experience. Currently, there are many database products at the storage layer, such as HBase and open-source Redis, which can be selected based on the business scale, cost, and performance. GeminiDB Redis API is an in-house NoSQL database service. It can meet strict requirements of IM systems on the storage layer in terms of performance and scale, including massive data storage, low storage cost, lifecycle management, high write throughput, and low read latency.

With a cloud native distributed architecture, GeminiDB Redis API is compatible with Redis 5.0 and adopts decoupled storage and compute. In-house storage

systems ensure unlimited capacity expansion, strong consistency, and high reliability. The compute layer leverages LSM-based storage engines. A large number of random writes are converted into sequential writes, which greatly enhances write performance. In addition, read performance is greatly improved by read caches and Bloom filters.





Application Cases of GeminiDB Redis API in IM Scenarios

The following figure shows an IM system based on GeminiDB Redis API. A stream is used as a basic data structure. A Redis stream acts as a message container and allows data exchange between producers and consumers. A Redis stream provides basic functions of IM systems, such as message subscription, distribution, and adding consumers. Users can quickly build an IM system using GeminiDB Redis API. When a group chat is created, a stream queue is also created for the group chat on a GeminiDB Redis instance. Each sender adds messages to the stream queue in time sequence. A stream is a persistent queue that ensures no information loss.





GeminiDB Redis API uses a series of innovative technologies to improve read and write performance, scale up storage in seconds, and automatically back up data. A GeminiDB Redis API offers a storage layer of the IM system. Its excellent read and write performance and advanced features will greatly facilitate IM applications. In addition, GeminiDB Redis API balances performance and costs based on open-source Redis and can be widely used in fields such as smart healthcare, traffic control, and counter.

# 6.6 Implementing Distributed Locks Using Lua Scripts for GeminiDB Redis API

In a distributed system, distributed locks are used to ensure that only one process or thread can execute a specific code snippet at a time.

This section describes how to use Lua scripts to implement distributed locks.

# **Redis Distributed Locks**

Redis offers a basic locking mechanism that relying on atomic commands to implement distributed locks. One of the simplest ways to implement distributed locks is using the SETNX command. SETNX sets the value of a key only if the key does not exist. In this way, the key value is successfully set for the first process that obtains a lock, and subsequent processes that attempt to obtain the lock fail until the lock is released.

To prevent a lock from being released forever (for example, the process that holds the lock crashes), an expiration time is usually set for the lock using the EXPIRE command. In versions later than Redis 2.6.12, the EX and NX options are added to the SET command. You can set the expiration time when setting the key. This operation is atomic.

• Acquiring a lock

You can run the following command to acquire a lock:

SET resource\_name my\_random\_value NX PX 30000

The NX parameter is used to check whether the key exists. If it does not, that is, no one holds the lock, the lock is successfully acquired.

The PX parameter is mandatory and is used to set a lock validity period accurate to milliseconds. If a lock holder exits abnormally or a lock expires, the lock is automatically released to prevent deadlocks.

• Releasing a lock

Releasing a lock is more complex. A lock can be released only by its holder. To release multiple locks in sequence, you need to execute a Lua script.

```
Lua script:
```

```
if redis.call("get",KEYS[1]) == ARGV[1] then
  return redis.call("del",KEYS[1])
else
  return 0
end
```

The Lua script must be executed together with the EVAL command, for example:

EVAL 'if redis.call("get",KEYS[1]) == ARGV[1] then return redis.call("del",KEYS[1]) else return 0 end' 1 resource\_name my\_random\_value

This script ensures that only the holder can release the lock.

• Analysis

The preceding solution is easy to use but has the following disadvantages:

- The expired lock is released, but transactions are incomplete.
- Reentrant locks are not supported.
- No notification mechanism is available. Locks need to be preempted in polling mode, consuming a lot of CPU resources.

For production applications, the Redis distributed lock library is recommended to balance functions and performance.

The following uses Redisson as an example to describe how to use the Redis distributed lock library.

# Implementing Distributed Locks Using Redisson

Redisson is a Redis Java client and provides the distributed locking feature. A distributed lock is a mechanism used to synchronously access shared resources in a distributed system. Redisson implements distributed locks through atomic operations of Redis to ensure that only one client can access a resource at a time.

Distributed locks based on Redisson have the following features:

- High efficiency: Redis features high performance and memory storage, making distributed lock operations very fast.
- Easy to use: Various APIs allow developers to easily use distributed locks in Java applications.
- Reliability: Distributed locks based on Redisson are highly reliable. Even if a partition or node breaks down, the locks are not affected.

### Example

import org.redisson.Redisson; import org.redisson.api.RLock;

```
import org.redisson.api.RedissonClient;
import org.redisson.config.Config;
public class LockExamples {
   public static void main(String[] args) {
     // Creates a Redisson client.
     Config config = new Config();
     config.useSingleServer().setAddress("redis://127.0.0.1:7200");
     RedissonClient redisson = Redisson.create(config);
     // Obtains a distributed lock.
     RLock lock = redisson.getLock("myLock");
     try {
     // Acquires a lock.
        lock.lock();
        System.out.println("Lock acquired, executing critical section...");
       // Executes the code to acquire a lock.
        // ...
        System.out.println("Critical section executed, releasing lock...");
     } catch (Exception e) {
        e.printStackTrace();
      } finally {
       // Release a lock.
        lock.unlock();
     3
    // Closes the Redisson client.
     redisson.shutdown();
  }
}
```

# **Other Recommended Implementations of Distributed Locks**

Redisson is a Java client which has implemented distributed locks for various programming languages. Here are a few links to available implementations from the **Redis official website**:

- **Redlock-rb** (Ruby implementation). There is also a **fork of Redlock-rb** that adds a gem for easy distribution.
- **RedisQueuedLocks** (Ruby implementation).
- **Redlock-py** (Python implementation).
- **Pottery** (Python implementation).
- Aioredlock (Asyncio Python implementation).
- **RedisMutex** (PHP implementation with both **Redis extension** and **Predis library** clients support).
- **Redlock-php** (PHP implementation).
- cheprasov/php-redis-lock (PHP library for locks).
- rtckit/react-redlock (Async PHP implementation).
- **Redsync** (Go implementation).
- **Redisson** (Java implementation).
- **Redis::DistLock** (Perl implementation).
- **Redlock-cpp** (C++ implementation).
- **Redis-plus-plus** (C++ implementation).
- **Redlock-cs** (C#/.NET implementation).

- **RedLock.net** (C#/.NET implementation). Includes async and lock extension support.
- **ScarletLock** (C# .NET implementation with configurable datastore).
- **Redlock4Net** (C# .NET implementation).
- **node-redlock** (NodeJS implementation). Includes support for lock extension.
- **simple-redis-mutex** (Node.js implementation) Available as an **NPM package**.
- **Deno DLM** (Deno implementation)
- **Rslock** (Rust implementation). Includes async and lock extension support.

# 6.7 Suggestions on Alarm Rules of GeminiDB Redis 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 **4.13.2 Configuring Alarm Rules**.

This section describes recommended alarm rules of GeminiDB Redis instances.

| Metric ID                   | Metric<br>Name                                 | Di<br>me<br>nsi<br>on | Threshol<br>d (Raw<br>Value)<br>in Best<br>Practices                   | Al<br>ar<br>Se<br>ve<br>rit<br>y<br>in<br>Be<br>st<br>Pr<br>act<br>ice<br>s | Alarm Handling Suggestion                     |
|-----------------------------|--|-----------------------|--|---|---|
| redis688_qps<br>_send_total | Total<br>Traffic<br>Sent by<br>the<br>Instance | Inst<br>anc<br>e      | ≥<br>875,000,<br>000<br>bytes/s<br>for 3<br>consecuti<br>ve<br>periods | Ma<br>jor   | Comply with 5.1 Development<br>and O&M Rules. |

**Table 6-2** Suggestions on alarm rules of GeminiDB Redis instance metrics

| Metric ID                             | Metric<br>Name                                      | Di<br>me<br>nsi<br>on | Threshol<br>d (Raw<br>Value)<br>in Best<br>Practices                   | Al<br>ar<br>Se<br>ve<br>rit<br>y<br>in<br>Be<br>st<br>Pr<br>act<br>ice<br>s | Alarm Handling Suggestion   |  |
|---------------------------------------|---|-----------------------|--|---|---|--|
| redis689_qps<br>_receive_tot<br>al    | Total<br>Traffic<br>Receive<br>d by the<br>Instance | Inst<br>anc<br>e      | ≥<br>875,000,<br>000<br>bytes/s<br>for 3<br>consecuti<br>ve<br>periods | Ma<br>jor   | Comply with 5.1 Development<br>and O&M Rules.   |  |
| redis803_clu<br>ster_all_avg_<br>usec | Average<br>Latency                                  | Inst<br>anc<br>e      | ≥ 50,000<br>µs for 5<br>consecuti<br>ve<br>periods                     | Ma<br>jor   | <ul> <li>Check whether the instance<br/>has performance<br/>bottlenecks in CPU,<br/>memory, or connections. If<br/>yes, resolve the bottlenecks<br/>based on the related<br/>suggestions.</li> <li>If service optimization<br/>cannot be performed,<br/>increase the instance<br/>specifications or use<br/>specifications or use<br/>specifications with better<br/>disk performance. For<br/>details, see 4.6.4 Changing<br/>vCPUs and Memory and<br/>4.6.6.2 Adding Instance<br/>Nodes.</li> </ul> |  |

| Metric ID                             | Metric<br>Name   | Di<br>me<br>nsi<br>on | Threshol<br>d (Raw<br>Value)<br>in Best<br>Practices   | Al<br>ar<br>Se<br>ve<br>rit<br>y<br>in<br>Be<br>st<br>Pr<br>act<br>ice<br>s | Alarm Handling Suggestion   |
|---------------------------------------|------------------|-----------------------|--|---|---|
| redis804_clu<br>ster_all_p99_<br>usec | P99<br>Latency   | Inst<br>anc<br>e      | ≥<br>150,000<br>µs for 5<br>consecuti<br>ve<br>periods | Ma<br>jor   | <ul> <li>Check whether the instance<br/>has performance<br/>bottlenecks in CPU,<br/>memory, or connections. If<br/>yes, resolve the bottlenecks<br/>based on the related<br/>suggestions.</li> <li>If service optimization<br/>cannot be performed,<br/>increase the instance<br/>specifications or use<br/>specifications with better<br/>disk performance. For<br/>details, see 4.6.4 Changing<br/>vCPUs and Memory and<br/>4.6.6.2 Adding Instance<br/>Nodes.</li> </ul> |
| redis816_clu<br>ster_disk_us<br>age   | Storage<br>Usage | Inst<br>anc<br>e      | ≥ 80%<br>for 3<br>consecuti<br>ve<br>periods           | Ma<br>jor   | Scale up the storage space. For details, see <b>4.6.7.2 Manually</b> Scaling Up Storage Space.  |
| nosql001_cp<br>u_usage                | CPU<br>Usage     | No<br>de              | ≥ 85%<br>for 3<br>consecuti<br>ve<br>periods           | Ma<br>jor   | Upgrade instance specifications<br>or add nodes to reduce the<br>CPU load. For details, see 4.6.4<br>Changing vCPUs and Memory<br>and 4.6.6.2 Adding Instance<br>Nodes.   |
| nosql002_m<br>em_usage                | Memor<br>y Usage | No<br>de              | ≥ 85%<br>for 3<br>consecuti<br>ve<br>periods           | Ma<br>jor   | Upgrade memory<br>specifications. For details, see<br><b>4.6.4 Changing vCPUs and</b><br><b>Memory</b> .  |

| Metric ID                         | Metric<br>Name                          | Di<br>me<br>nsi<br>on | Threshol<br>d (Raw<br>Value)<br>in Best<br>Practices                  | Al<br>ar<br>Se<br>ve<br>rit<br>y<br>in<br>Be<br>st<br>Pr<br>act<br>ice<br>s | Alarm Handling Suggestion                     |
|-----------------------------------|---|-----------------------|---|---|---|
| redis669_co<br>nnection_us<br>age | Connect<br>ion<br>Usage                 | No<br>de              | ≥ 80%<br>for 3<br>consecuti<br>ve<br>periods                          | Ma<br>jor   | Comply with 5.1 Development<br>and O&M Rules. |
| gemini004_b<br>ytes_in            | Networ<br>k Input<br>Throug<br>hput     | No<br>de              | ≥<br>87,500,0<br>00<br>bytes/s<br>for 3<br>consecuti<br>ve<br>periods | Ma<br>jor   | Comply with 5.1 Development<br>and O&M Rules. |
| gemini003_b<br>ytes_out           | Networ<br>k<br>Output<br>Throug<br>hput | No<br>de              | ≥<br>87,500,0<br>00<br>bytes/s<br>for 3<br>consecuti<br>ve<br>periods | Ma<br>jor   | Comply with 5.1 Development<br>and O&M Rules. |

# 6.8 GeminiDB Redis API for Product Correlation Analysis

This section describes how to use Redis data structures (such as sets, sorted sets, and hash tables) to build an application for correlation analysis on e-commerce store items. You can use GeminiDB Redis API in the following scenarios.

# **Application Scenarios**

The correlation can be analyzed based on products added to the shopping cart of a user. The results are crucial for the e-commerce industry and can be used to analyze user's shopping habits. For example, merchants can:

- Recommend related items to the user who is browsing on the details page of a specific item.
- Recommend related items to a user who just added an item to the shopping cart.
- Display highly correlated items together.

# Data Structure Design

- Product browse records
  - Store IDs of products that users have browsed in a set of a Redis instance.
  - Key: user:<user\_id>:viewed
  - Value: set of product IDs

Example:

```
User 123 browsed products 1001, 1002, and 1003.
```

user:123:viewed -> {1001, 1002, 1003}

• Purchase records

Store IDs of products that users have purchased in a set.

- Key: user:<user\_id>:purchased
- Value: set of product IDs

Example:

User 123 purchased products 1001 and 1005. user:123:purchased -> {1001, 1005}

• Product correlation

Store co-occurrence times of products in a zset (sorted set) of a Redis instance.

- Key: product:<product\_id>:related
- Value: IDs of other related products and co-occurrence times

Example:

Other product ID related to product 1001 and their co-occurrence times product:1001:related -> {1002:5, 1003:3, 1005:2}

# **Data Collection**

• User browse behavior

Each time a user browses a product, add its ID to the user's browse records. The following is an example of C++ code:

```
void recordView(int user_id, int product_id) {
   std::cout << "User ID " << user_id << " has viewed Product ID " << product_id << std::endl;
   redis.sadd("user:" + std::to_string(user_id) + ":viewed", std::to_string(product_id));
}</pre>
```

• User purchase behavior

Each time a user purchases a product, add its ID to the user's purchase records. The following is an example of C++ code:

```
void recordPurchase(int user_id, int product_id) {
    std::cout << "User ID " << user_id << " has purchased Product ID " << product_id << std::endl;
    redis.sadd("user:" + std::to_string(user_id) + ":purchased", std::to_string(product_id));</pre>
```

# **Query API**

The following is an example of C++ code for querying other products most related to a specified product based on its ID:

```
std::vector<std::pair<std::string, double>> getRelatedProducts(int product_id) {
    std::vector<std::pair<std::string, double>> result;
    redis.zrevrange("product:" + std::to_string(product_id) + ":related", 0, -1, std::back_inserter(result));
    return result;
}
```

# Complete Example of C++ Code

The following is a complete example of C++ code, which is implemented using C+ + Redis client **redis++**.

```
#include <iostream>
#include <iterator>
#include <set>
#include <vector>
#include <string>
#include <utility>
#include "sw/redis++/redis++.h"
using namespace sw::redis;
auto redis = Redis("tcp://127.0.0.1:6379");
void recordView(int user_id, int product_id) {
  std::cout << "User ID " << user_id << " has viewed Product ID " << product_id << std::endl;
  redis.sadd("user:" + std::to_string(user_id) + ":viewed", std::to_string(product_id));
void recordPurchase(int user_id, int product_id) {
  std::cout << "User ID " << user_id << " has purchased Product ID " << product_id << std::endl;
  redis.sadd("user:" + std::to_string(user_id) + ":purchased", std::to_string(product_id));
void updateRelatedProducts(int user_id) {
  std::set<std::string> viewed_products;
  redis.smembers("user:" + std::to_string(user_id) + ":viewed", std::inserter(viewed_products,
viewed_products.end()));
  std::set<std::string> purchased_products;
  redis.smembers("user:" + std::to_string(user_id) + ":purchased", std::inserter(purchased_products,
purchased_products.end()));
  for (const auto& product_id : viewed_products) {
     for (const auto& related_product_id : viewed_products) {
        if (product_id != related_product_id) {
         //Sets the browse weight to 1.
           redis.zincrby("product:" + product_id + ":related", 1, related_product_id);
        }
     }
  }
  for (const auto& product_id : purchased_products) {
     for (const auto& related_product_id : purchased_products) {
        if (product_id != related_product_id) {
        //Higher weight of purchase behavior
           redis.zincrby("product:" + product_id + ":related", 2, related_product_id);
        }
     }
  }
```

std::vector<std::pair<std::string, double>> getRelatedProducts(int product\_id) {

```
std::vector<std::pair<std::string, double>> result;
  redis.zrevrange("product:" + std::to_string(product_id) + ":related", 0, -1, std::back_inserter(result));
  return result;
int main() {
  int user_id = 123;
  //Simulates user behavior.
  recordView(user_id, 1001);
  recordView(user_id, 1002);
  recordView(user_id, 1003);
  recordView(user id, 1004);
  recordView(user_id, 1005);
  recordPurchase(user_id, 1001);
  recordPurchase(user_id, 1005);
   //Updates product correlation information based on user behavior.
  updateRelatedProducts(user_id);
  //Queries products related to product 1001.
  int product_id = 1001;
  auto related_products = getRelatedProducts(product_id);
  std::cout << "Product ID " << product_id << "'s related product is: " << std::endl;
  for (const auto& [product_id, score] : related_products) {
    std::cout << "Product ID: " << product_id << ", Score: " << score << std::endl;
  }
  return 0;
```

Returned results:

| User  | ID  | 123  | has  | viewed  | Produc  | t ID  | 1001    |
|-------|-----|------|------|---------|---------|-------|---------|
| User  | ID  | 123  | has  | viewed  | Produc  | t ID: | 1002    |
| User  | ID  | 123  | has  | viewed  | Produc  | t ID: | 1003    |
| User  | ID  | 123  | has  | viewed  | Produc  | t ID: | 1004    |
| User  | ID  | 123  | has  | viewed  | Produc  | t ID: | 1005    |
| User  | ID  | 123  | has  | purcha  | sed Pro | oduct | ID 1001 |
| User  | ID  | 123  | has  | purcha  | sed Pro | oduct | ID 1005 |
| Produ | ıct | ID 1 | 1001 | 's rela | ted pro | oduct | is:     |
| P     | roc | duct | ID:  | 1005,   | Score:  | 3     |         |
| P     | roc | duct | ID:  | 1004,   | Score:  | 1     |         |
| P     | roc | duct | ID:  | 1003,   | Score:  | 1     |         |
| P     | roc | duct | ID:  | 1002,   | Score:  | 1     |         |

# 6.9 GeminiDB Redis API for Online Classroom

Redis can be used to easily build an online classroom application with functions such as user authentication, course management, real-time message push, and online interaction. The following provides a basic architecture and implementation ideas of an online classroom application based on Redis, including data structure design and function module development. GeminiDB Redis API is completely compatible with Redis. You can use a GeminiDB Redis instance to build the application.

# Data Structure Design

• User information

Store basic user information and authentication data.

- Key: user:<user\_id>
- Value: user information (HASH data structure of Redis)

Example:

user:123 -> {name: "ZhangSan", email: "zhangsan@example.com", password\_hash: "hashed\_password"}

• Course information

Store basic information and metadata of a course.

- Key: course:<course\_id>
- Value: course information (HASH data structure of Redis)

Example:

course:101 -> {title: "Introduction to Redis", description: "Learn the basics of Redis.", instructor: "LiSi"}

• Association between a user and a course

Record association between a user and a course.

- Key: user:<user\_id>:courses
- Value: IDs of courses that a user has attended (SET data structure of Redis)

### Example:

user:123:courses -> {101, 102}

- Key: course:<course\_id>:students
- Value: IDs of users who have attended a course (SET data structure of Redis)

Example:

course:101:students -> {123, 456}

Instant messages

Store instant messages and interactions in a course.

- Key: course:<course\_id>:messages
- Value: messages in a course (LIST data structure of Redis)

Example:

course:101:messages -> ["ZhangSan: Hello everyone!", "LiSi: Welcome to the class!"]

• User status

Record user's status.

- Key: user:<user\_id>:online
- Value: user status (STRING data structure of Redis: 1 indicates online and 0 offline.)

Example:

user:123:online -> 1

# **Functional Module Development**

The following is an example of C++ code, which is implemented using C++ Redis client **redis++**.

### User authentication

Store user and session information.

```
void registerUser(int user_id, const std::string& name, const std::string& email, const std::string& password_hash) {
```

```
redis.hset("user:" + std::to_string(user_id), "name", name);
redis.hset("user:" + std::to_string(user_id), "email", email);
redis.hset("user:" + std::to_string(user_id), "password_hash", password_hash);
}
bool loginUser(int user_id, const std::string& password_hash) {
```

```
std::string stored_password_hash = redis.hget("user:" + std::to_string(user_id), "password_hash").value();
return stored_password_hash == password_hash;
```

```
• Course management
```

}

Add and query course information.

```
void addCourse(int course_id, const std::string& title, const std::string& description, const std::string& instructor) {
```

```
redis.hset("course:" + std::to_string(course_id), "title", title);
redis.hset("course:" + std::to_string(course_id), "description", description);
redis.hset("course:" + std::to_string(course_id), "instructor", instructor);
}
std::string getCourseTitle(int course_id) {
return redis.hget("course:" + std::to_string(course_id), "title").value();
}
```

• Association between a user and a course

Manage association between a user and a course.

```
void enrollUserInCourse(int user_id, int course_id) {
  redis.sadd("user:" + std::to_string(user_id) + ":courses", std::to_string(course_id));
  redis.sadd("course:" + std::to_string(course_id) + ":students", std::to_string(user_id));
bool isUserEnrolledInCourse(int user_id, int course_id) {
  return redis.sismember("user:" + std::to_string(user_id) + ":courses", std::to_string(course_id));
}
     Instant messages
.
Push instant messages in a course.
void sendMessageToCourse(int course_id, const std::string& message) {
  redis.rpush("course:" + std::to_string(course_id) + ":messages", message);
std::vector<std::string> getCourseMessages(int course_id) {
  std::vector<std::string> output;
  redis.lrange("course:" + std::to_string(course_id) + ":messages", 0, -1, std::inserter(output, output.end()));
  return output;
}
     User status
Manage user's status.
void setOnlineStatus(int user id, bool online) {
  redis.set("user:" + std::to_string(user_id) + ":online", online ? "1" : "0");
```

```
bool isUserOnline(int user_id) {
    std::string status = redis.get("user:" + std::to_string(user_id) + ":online").value();
    return status == "1";
```

# 6.10 GeminiDB Redis API for Session Management in Web Applications

Redis is widely used for session management in web applications due to its efficiency, particularly in distributed systems. This section describes how to manage sessions using GeminiDB Redis API, which is completely compatible with Redis.

# **Application Scenarios**

- Distributed system: Traditionally, in a distributed system, server sessions like memory-based sessions, cannot be shared across servers. GeminiDB Redis API can address this issue.
- High performance: With a distributed and scalable architecture, GeminiDB Redis API can efficiently handle high throughput.
- Automatic expiration: You can enable automatic expiration for keys on your GeminiDB Redis instance to effectively manage session lifecycles.

# Advantages of GeminiDB Redis API in Session Management

- High availability: GeminiDB Redis API guarantees that session data remains intact even if there is a single point of failure.
- Scalability: Nodes can be added for cluster GeminiDB Redis instances to handle high throughput.
- Security: The access control and encryption functions enhance session data security.

# Prerequisites

You have purchased a GeminiDB Redis instance. For details, see **4.2 Buying a GeminiDB Redis Instance**.

# Procedure

# Step 1 Creating a Spring Boot Project

Use Spring Initializr to create a Spring Boot project and add the following dependencies:

```
<dependency>
<groupId>org.springframework.session</groupId>
<artifactId>spring-session-data-redis</artifactId>
</dependency>
<dependency>
<groupId>org.springframework.boot</groupId>
<artifactId>spring-boot-starter-data-redis</artifactId>
</dependency>
<dependency>
<groupId>org.springframework.boot</groupId>
<artifactId>spring-boot-starter-web</artifactId>
</dependency>
```

Use Spring Boot 3.0.x or later and JDK 17 or later.

### Project directory structure:



# Step 2 Configuring Redis and Enabling Redis Session Management

Configure connection information of a GeminiDB Redis instance in **application.properties**.

```
spring.data.redis.host=${ip}
spring.data.redis.port=${port}
spring.data.redis.password=${password}
server.servlet.session.timeout=30 #Session timeout interval (minutes)
```

# In the main class or configuration class of Spring Boot applications, add **@EnableRedisHttpSession** to enable Redis sessions.

```
package com.huawei.sessionmanagement;
import org.springframework.session.data.redis.config.annotation.web.http.EnableRedisHttpSession;
@EnableRedisHttpSession
public class SessionConfig {
//Uses Spring Session to automatically stores sessions in Redis instances.
}
```

# Step 3 Logging In and Out a User

Create a controller to process the login and logout logic.

```
package com.huawei.sessionmanagement;
import jakarta.servlet.http.HttpSession;
import org.springframework.web.bind.annotation.PostMapping;
import org.springframework.web.bind.annotation.RequestParam;
import org.springframework.web.bind.annotation.RestController;
@RestController
public class SessionController {
   @PostMapping("/login")
   public String login(@RequestParam String username, HttpSession session) {
      session.setAttribute("username", username);
      return "Login successful";
   }
   @PostMapping("/logout")
   public String logout(HttpSession session) {
      session.invalidate();
      return "Logout successful";
   }
}
```
#### Step 4 Obtaining Session Information and Specifying Session Storage

Obtain session information using **HttpSession** and add the following code to the SessionController class:

import org.springframework.web.bind.annotation.GetMapping;

```
@GetMapping("/getSessionInfo")
public String getSessionInfo(HttpSession session) {
   String username = (String) session.getAttribute("username");
   return "Username: " + username;
```

For more precise control, you can directly use RedisTemplate to store and manage sessions.

```
import org.springframework.data.redis.core.RedisTemplate;
import org.springframework.stereotype.Service;
@Service
public class SessionService {
  private final RedisTemplate<String, Object> redisTemplate;
  public SessionService(RedisTemplate<String, Object> redisTemplate) {
     this.redisTemplate = redisTemplate;
  }
  public void saveSession(String sessionId, Object sessionData) {
     redisTemplate.opsForValue().set(sessionId, sessionData, 30, TimeUnit.MINUTES);
  }
  public Object getSession(String sessionId) {
     return redisTemplate.opsForValue().get(sessionId);
  }
  public void deleteSession(String sessionId) {
     redisTemplate.delete(sessionId);
  }
```

By performing the preceding steps, you can efficiently use GeminiDB Redis API to manage sessions of web applications and improve application performance and scalability.

}

# 7 Performance White Paper

7.1 General Performance Data

7.2 Performance Data in RTA Scenarios

#### 7.1 General Performance Data

#### 7.1.1 Performance Test Methods

This section describes performance testing of GeminiDB Redis instances, including the test environment, tools, metrics, models, and procedure.

#### **Test Environment**

- Region: CN-Hong Kong
- AZ: AZ1
- Elastic Cloud Server (ECS): c6.4xlarge.2 with 16 vCPUs, 32 GB of memory, and CentOS 7.5 64-bit image
- Nodes per instance: 3
- Instance specifications: Specifications described in Table 7-1

#### Table 7-1 Instance specifications

| No.       | Specifications    |
|-----------|-------------------|
| Cluster 1 | 4 vCPUs x 3 nodes |
| Cluster 2 | 8 vCPUs x 3 nodes |

#### Test Tool

This test used a multi-thread load test tool, memtier\_benchmark, developed by Redis Labs. For details about how to use this tool, see **memtier\_benchmark**. The following describes some functions of memtier\_benchmark.

Usage: memtier\_benchmark [options]

A memcache/redis NoSQL traffic generator and performance benchmarking tool.

| Connection a | nd General | Options: |
|--------------|------------|----------|
|--------------|------------|----------|

| -s,server=ADDR<br>-p,port=PORT<br>-a,authenticate=PASSWORD<br>-o,out-file=FILE  | Server address (default: localhost)<br>Server port (default: 6379)<br>Authenticate to redis using PASSWORD<br>Name of output file (default: stdout)  |
|---|--|
| Test Options:   |  |
| -n,requests=NUMBER<br>-c,clients=NUMBER<br>-t,threads=NUMBER<br>ratio=RATIO<br>pipeline=NUMBER<br>distinct-client-seed<br>randomize | Number of total requests per client (default: 10000)<br>Number of clients per thread (default: 50)<br>Number of threads (default: 4)<br>Set:Get ratio (default: 1:10)<br>Number of concurrent pipelined requests (default: 1)<br>Use a different random seed for each client<br>Random seed based on timestamp (default is constant value) |
| Dbject Options:<br>-ddata-size=SIZE<br>-Rrandom-data  | Object data size (default: 32)<br>Indicate that data should be randomized  |
| Key Options:<br>key-prefix=PREFIX<br>key-minimum=NUMBER<br>key-maximum=NUMBER   | Prefix for keys (default: memtier-)<br>Key ID minimum value (default: 0)<br>Key ID maximum value (default: 10000000)   |
|   |  |

#### **Test Metrics**

 Table 7-2 Test metrics

| Metric Abbreviation | Description  |
|---------------------|--|
| QPS                 | Number of read and write operations executed per second.   |
| Avg Latency         | Average latency of read and write operations, in milliseconds.   |
| p99 Latency         | <ul> <li>p99 latency of read and write operations.</li> <li>99% of operations can be completed within this latency. Only 1% of operations have a latency longer.</li> <li>Unit: ms.</li> </ul> |

#### **Test Models**

• Workload model

#### Table 7-3 Workload models

| Workload Model | Description                        |
|----------------|------------------------------------|
| 100% Write     | 100% write operations (string set) |

| Workload Model     | Description  |
|--------------------|--|
| 100% Read          | 100% read operations (string get).<br>The even random access model is<br>used in strict performance tests. |
| 50% Read+50% Write | 50% read operations (string get)<br>plus 50% write operations (string<br>set)                              |

• Data model

#### Table 7-4 Data model description

| Data Model   | Description                                 |
|--------------|---|
| value length | A value in 100 bytes is generated randomly. |

#### **Test scenarios**

| Table | 7-5 | Test | scenario | description |
|-------|-----|------|----------|-------------|
|-------|-----|------|----------|-------------|

| Test Scenario                          | Description   |
|--|---|
| The data volume is less than memory.   | All data can be cached in memory.   |
| The data volume is larger than memory. | Some data can be cached in memory,<br>and some data can be accessed from<br>the DFV storage pool. |

#### **Test Procedure**

Use a DB instance with 3 nodes and 4 vCPUs for each node as an example:

Scenario 1: When the data volume is less than memory, data is written to and read data from the instance respectively and then concurrently, and record OPS, average latency, and P99 latency of each operation. Workload models and methods of testing performance metrics are as follows:

• Workload model: 100% write

Use 30 threads and 3 client connections for each thread. That is, 100-byte data is written 60,000,000 times on total 90 connections. The data is generated randomly by all the clients using different seeds within the range of [1, 60,000,000]. Based on the specified range of keys, the total size of data written this time is less than the memory of the database cluster.

./memtier\_benchmark -s \${ip} -a \${passwd} -p \${port} -c 3 -t 30 -n 1000000 --random-data -randomize --distinct-client-seed -d 100 --key-maximum=60000000 --key-minimum=1 --keyprefix= --ratio=1:0 --out-file=./output\_filename • Workload model: 100% read

Use 30 threads and 3 client connections for each thread. That is, data is randomly and concurrently read 60,000,000 times over 90 connections. The key is within the range of [1, 60,000,000].

./memtier\_benchmark -s \${ip} -a \${passwd} -p \${port} -c 3 -t 30 -n 1000000 --random-data -randomize --distinct-client-seed --key-maximum=60000000 --key-minimum=1 --key-prefix= -ratio=0:1 --out-file=./output\_filename

• Workload model: 50% read and 50% write

Use 30 threads and 3 client connections for each thread. That is, data is randomly and concurrently written and read 60,000,000 times over 90 connections. The key is within the range of [1, 60,000,000]. The write-read ratio is 1:1. Based on the specified range of keys, the total size of data written and read this time was less than the memory of the database cluster.

./memtier\_benchmark -s \${ip} -a \${passwd} -p \${port} -c 3 -t 30 -n 1000000 --random-data -randomize --distinct-client-seed -d 100 --key-maximum=60000000 --key-minimum=1 --keyprefix= --ratio=1:1 --out-file=./output\_filename

2. Scenario 2: When the data volume is larger than memory of the database cluster, use 30 threads and create 3 client connections for each thread. That is, 100-byte data is concurrently written 20,000,000 times over total 90 connections. The data is generated randomly by all the clients using different seeds within the range of [60,000,001, 780,000,000]. In addition, pipeline parameters were set to speed up data writes. Based on the specified range of keys and total writes, the total size of data written this time was larger than the memory of the database cluster.

./memtier\_benchmark -s \${ip} -a \${passwd} -p \${port} -c 3 -t 30 -n 20000000 --random-data -randomize --distinct-client-seed -d 100 --key-maximum=780000000 --key-minimum=60000001 -pipeline=100 --key-prefix= --ratio=1:0 --out-file=./output\_filename

3. When the data volume is larger than memory, data is written to and read from the database cluster respectively and then concurrently, and metrics OPS, average latency, and p99 latency of each operation were recorded. Workload models and methods of testing performance metrics are as follows:

• Test model: 100% write

Use 30 threads and 3 clients for each thread. That is, 100-byte data is written 500,000 times over total 90 connections. The data is generated randomly by all the clients within the range of [1, 780,000,000].

./memtier\_benchmark -s \${ip} -a \${passwd} -p \${port} -c 3 -t 30 -n 500000 --random-data -randomize --distinct-client-seed -d 100 --key-maximum=780000000 --key-minimum=1 --keyprefix= --ratio=1:0 --out-file=./output\_filename

• Test model: 100% read

Use 30 threads and 3 clients for each thread. That is, data is randomly and concurrently read 500,000 times over 90 connections. The key is within the range of [1, 780,000,000].

./memtier\_benchmark -s \${ip} -a \${passwd} -p \${port} -c 3 -t 30 -n 500000 --random-data -randomize --distinct-client-seed --key-maximum=780000000 --key-minimum=1 --key-prefix= -ratio=0:1 --out-file=./output\_filename

• Test model: 50% read and 50% write

Use 30 threads and 3 clients for each thread. That is, data is written and read 500,000 times over total 90 connections. The data is generated randomly by all the clients within the range of [1, 780,000,000].

./memtier\_benchmark -s \${ip} -a \${passwd} -p \${port} -c 3 -t 30 -n 500000 --random-data -randomize --distinct-client-seed -d 100 --key-maximum=780000000 --key-minimum=1 --keyprefix= --ratio=1:1 --out-file=./output\_filename

#### 7.1.2 Performance Test Results

This section describes performance metrics that are tested using different data and workload models in different scenarios. Only performance data of instances with small- and medium specifications is presented. To use higher specifications, you can scale out or up the instances.

- **Table 7-6** describes the test data used when the data volume is less than memory.
- **Table 7-7** describes the test data used when the data volume is greater than memory.

| Instanc<br>e<br>Specific<br>ations | Test Model                                   | Workload<br>Model            | QPS    | Average<br>Latency<br>(ms) | p99<br>Latency<br>(ms) |
|------------------------------------|--|------------------------------|--------|----------------------------|------------------------|
| 4 vCPUs<br>x 3<br>nodes            | value_length<br>= 100 bytes<br>clients = 90  | 100% write                   | 125590 | 0.66                       | 1.85                   |
|                                    | value_length<br>= 100 bytes<br>clients = 105 | 100% read                    | 139741 | 0.62                       | 1.51                   |
|                                    | value_length                                 | 50% read<br>and 50%<br>write | 125620 | Read: 0.56                 | Read: 1.32             |
|                                    | = 100 bytes<br>clients = 90                  |                              |        | Write: 0.55                | Write: 1.30            |
| 8 vCPUs<br>x 3<br>nodes            | value_length<br>= 100 bytes<br>clients = 128 | 100% write                   | 216392 | 0.62                       | 1.92                   |
|                                    | value_length<br>= 100 bytes<br>clients = 128 | 100% read                    | 202970 | 0.62                       | 1.89                   |
|                                    | value_length                                 | 50% read                     | 212052 | Read: 0.63                 | Read: 1.94             |
|                                    | clients = 128                                | write                        |        | Write: 0.63                | Write: 1.92            |

#### Table 7-6 Test data

| Instanc<br>e<br>Specific<br>ations | Test Model                                   | Workload<br>Model | QPS        | Average<br>Latency<br>(ms) | p99<br>Latency<br>(ms) |
|------------------------------------|--|-------------------|------------|----------------------------|------------------------|
| 4 vCPUs<br>x 3<br>nodes            | value_length<br>= 100 bytes<br>clients = 75  | 100% write        | 123942     | 0.62                       | 1.30                   |
|                                    | value_length<br>= 100 bytes<br>clients = 96  | 100% read         | 125351     | 0.63                       | 1.54                   |
|                                    | value_length 50% read                        | 122485            | Read: 0.64 | Read: 1.65                 |                        |
|                                    | = 100 bytes<br>clients = 96                  | and 50%<br>write  |            | Write: 0.64                | Write: 1.61            |
| 8 vCPUs<br>x 3<br>nodes            | value_length<br>= 100 bytes<br>clients = 120 | 100% write        | 196596     | 0.62                       | 2.02                   |
|                                    | value_length<br>= 100 bytes<br>clients = 120 | 100% read         | 187716     | 0.62                       | 1.90                   |
|                                    | value_length                                 | 50% read          | 197097     | Read: 0.62                 | Read: 1.94             |
|                                    | clients = 120                                | write             |            | Write: 0.62                | Write: 1.94            |

Table 7-7 Test data

**clients** indicates the number of connections, which is the product of fields **t** and **c** in the **memtier** command.

#### 7.2 Performance Data in RTA Scenarios

Real-time API (RTA) is a core technology for placing refined ads in real-time. As shown in the following figure, the ad platform asks advertisers first whether they want to participate in the bidding, then searches and recalls ads after receiving the bid responses.



#### 7.2.1 Performance Test Methods

#### Objectives

RTA-based advertising poses higher technical requirements for advertisers, including quick response from the media and lower costs in data storage. In recent years, GeminiDB Redis API is widely used as a key-value (KV) signature database in RTA scenarios and delivers good performance at low costs.

This section describes a pressure test of GeminiDB Redis instances in RTA scenarios, including the performance on data compression, QPS, bandwidth, and latency.

#### **Test Environment**

This test used GeminiDB Redis clusters and Elastic Cloud Servers (ECSs). The following table lists the specifications.

• GeminiDB Redis cluster specifications

| Region              | CN East-Shanghai1                      |
|---------------------|--|
| AZ type             | Deployment across AZ 1, AZ 2, and AZ 3 |
| vCPUs of nodes      | 16                                     |
| Nodes               | 20                                     |
| Total storage space | 2 ТВ                                   |

• ECS specifications

|--|

| Specifications        | c7.4xlarge.2, 3 PCS |
|-----------------------|---------------------|
| vCPUs                 | 16                  |
| Memory                | 32 GiB              |
| Operating System (OS) | CentOS 8.2 64-bit   |

#### Test Tool

This test used memtier\_benchmark, which is a multi-thread load test tool developed by Redis Labs. For details, see **memtier\_benchmark**.

#### **Test Metrics**

Service scale of the simulated RTA scenario: 1 TB of data, 1.6 million QPS, and 1.5 Gbit/s of bandwidth.

1. Data samples

Categories of data samples are as follows.

| Category | Key           | Value  |
|----------|---------------|--|
| Hash     | 34 characters | 10 field-value pairs. A field contains 10 characters and a value contains 20 to 80 characters. |
| String   | 68 characters | 32 random characters   |
| String   | 19 characters | 500 to 2,000 random characters   |

**Four billion** keys need to be stored in the Redis clusters. The proportion of each data category is about 2:7:1, and the frequently accessed data accounts for 50% of the total.

2. Metrics

Test metrics of database operations are as follows.

| Metric<br>Abbreviation | Description  |
|------------------------|--|
| QPS                    | Number of requests executed per second.  |
| Avg Latency (ms)       | Average request latency, indicating the overall performance of a GeminiDB Redis cluster.                                 |
| p99 Latency (ms)       | p99 latency of a request, indicating that 99% of the request execution time is shorter than the value of this parameter. |

| p9999 Latency (ms) | p9999 latency of a request, indicating that 99.99% of<br>the request execution time is shorter than the value<br>of this parameter |
|--------------------|--|
|                    | of this parameter.   |

#### Test Procedure

1. Inject test data.

Before the test, generate and inject test data. Configure the three categories of data as follows:

- a. Hash
  - A key consists of 34 characters in the format of string prefix + nine digits. The digits are consecutive from 100 million to 900 million. The key is used to control the total data volume and hot data distribution.
  - Inject 10 field-value pairs. A field contains 10 characters and a value contains 20 to 80 random characters. The average value of a fieldvalue is 50 characters.
  - Construct and inject 800 million keys. memtier\_benchmark -s \${ip} -a \$(passwd} -p \${port} -c 20-t20 -n7500000 -d 32 -key-maximum=3 800000000 -key-minimum =1000000000 --key-pr efix ='cefkljrithuir123894873h4523blj4b2jkjh2iw13b nfdhsbnkfhsdjkh' --key-pattern=P:P--ratio=1:0 -pipelire=100
- b. String
  - A key consists of 68 characters in the format of string prefix + 10 digits. The digits are consecutive from 1 billion to 3.8 billion. The key is used to control the total data volume and hot data distribution.
  - Inject 32 random characters for a value.
  - Construct and inject 2.8 billion keys. memtier\_benchmark -s \${ip} -a \${passwd} -p \${port} -c 20 -t 20 -n 2500000 -command='hset \_key\_ mendke398d \_data\_ mebnejkehe \_data\_ fmebejdbnf \_data\_ j3i45u8923 \_data\_ j43245i908 \_data\_ jhiriu2349 \_data\_ 21021034ji \_data\_ jh23ui45j2 \_data\_ jiu5rj9234 \_data\_ j23i045u29 \_data\_' -d 50 --keymaximum=900000000 --key-minimum=100000000 --keyprefix='ewfdjkff43ksdh41fuihikucl' --command-key-pattern=P --pipeline=100
- c. String
  - A key consists of 19 characters in the format of string prefix + 9 digits. The digits are consecutive from 100 million to 300 million. The key is used to control the total data volume and hot data distribution.
  - Inject 500 to 2,000 random characters for a value. The average value is 1,250 bits.
  - Construct and inject 400 million keys. memtier\_benchmark -s \${ip} -a \${passwd} -p \${port} -c 20 -t 20 -n 520000 -d 1250 --key-maximum=300000000 --key-minimum=1000000000 --key-prefix='miqjkfdjiu' --key-pattern=P:P --ratio=1:0 --pipeline=100

After data is injected, there were 3,809,940,889 (about 3.8 billion) keys. Obtain the total data volume on the GeminiDB Redis API console and calculate the data compression ratio. The compressed storage space was 155 GB, and the compression ratio was 13.8%.

#### 

- About 4 billion data records were generated by memtier\_benchmark of the current version. Data distribution among different categories is not affected.
- A random character string constructed by memtier\_benchmark contains many consecutive characters, so the compression ratio was low. The data compression ratio is about 30% to 50% in actual production.

#### 2. Pressure test commands

Perform pressure tests on GeminiDB Redis clusters deployed on three ECSs, separately. The pressure test tasks are as follows:

- a. On ECS 1, run the HGETALL command for hashes and set a range for keys to allow access to hot data only. memtier\_benchmark -s \${ip} -a \${passwd} -p \${port} -c 20 -t 30 --test-time 1200 --randomdata --randomize --distinct-client-seed --command='hgetall \_\_key\_' --keymaximum=600000000 --key-minimum=200000000 --key-prefix='ewfdjkff43ksdh41fuihikucl' -out-file=./output\_filename
- b. Run the GET command for type data 2 and set a range for keys to allow access to hot data only. memtier\_benchmark -s \${ip} -a \${passwd} -p \${port} -c 70 -t 30 --test-time 1200 --randomdata --randomize --distinct-client-seed --key-maximum=2400000000 --keyminimum=1000000000 --keyprefix='cefkljrithuin123894873h4523bhj4b2jkjh2iu13bnfdhsbnkfhsdjkh' --ratio=0:1 --out-file=./ output\_filename
- c. Run the GET command for type data 3 and set a range for keys to allow access to hot data only. memtier\_benchmark -s \${ip} -a \${passwd} -p \${port} -c 10 -t 30 --test-time 1200 --randomdata --randomize --distinct-client-seed --key-maximum=300000000 --key-minimum=100000000 --key-prefix='miqjkfdjiu' --ratio=0:1 --out-file=./output\_filename

The number of connections (the product of **c** and **t**) was adjusted to modify the number of clients and configuration of each instance, so as to achieve a QPS of 1,600,000 and a read request traffic of 1.5 Gbit/s. Remain the service volume unchanged and evaluate the performance of GeminiDB Redis API.

#### 7.2.2 Performance Test Results

Over 1 TB of data was injected for pressure testing. The test results are as follows:

• Data compression ratio

1.1 TB of data (about 3.8 billion data records) is written and the occupied space is 155 GB after compression. The data compression ratio is about 13.8%.

Performance

The total QPS reaches 1.6 million, the total read request traffic is 1.5 Gbit/s, and the CPU usage ranges from 60% to 70%.

The average latency is about 0.7 ms, and the p99 long-tail latency is about 1.77 ms.

The results show that GeminiDB Redis API has stable latency in large-scale RTA scenarios. With data compression and decoupled compute and storage architecture, GeminiDB Redis API is an ideal KV database for advertising services.

# **8** FAQs

- 8.1 About GeminiDB Redis API
- 8.2 Billing
- 8.3 Database Usage
- 8.4 Database Connection
- 8.5 Backup and Restoration
- 8.6 Regions and AZs
- 8.7 Data Migration
- 8.8 Memory Acceleration
- 8.9 Freezing, Releasing, Deleting, and Unsubscribing from Instances

#### 8.1 About GeminiDB Redis API

#### 8.1.1 What Are the Differences Between GeminiDB Redis API, Open-Source Redis, and Other Open-Source Redis Cloud Services?

Redis, an open-source in-memory data structure store, is used as a cache broker. GeminiDB Redis API, an enhanced version of open-source Redis, is an elastic KV database compatible with the Redis protocol, supports much larger capacity than memory, and delivers ultimate performance. Hot data is stored in memory, and full data is stored in a high-performance storage pool. GeminiDB Redis API features:

• Low stable latency

The average single-point read/write latency is shorter than 1 ms, and the P99 latency is shorter than 2 ms. By adopting a multi-thread architecture, GeminiDB Redis API allows for flexible QPS adjustment ranging from 10,000 to 10,000,000.

• High cost-effectiveness

30% lower comprehensive costs: Because no standby node is required and GeminiDB Redis API offers an ultra-high data compression ratio of 4:1, it is cheaper to scale out storage capacity.

• Higher O&M efficiency

2 GB to 100 TB more capacity can be added to storage devices without any impact on services. Point-in-Time Recovery (PITR) restores databases up to a specific moment in time.

• More enhanced features for enterprises

An expiration time can be specified for individual fields in a hash. A Bloom filter can be used. Data can be imported extremely fast. Memory acceleration can be enabled.

For details about comparison between GeminiDB Redis instances and open-source KV databases, see **Highlights**.

## 8.1.2 How Is the Performance of GeminiDB Redis API Compared with Open-Source Redis?

GeminiDB Redis API uses the multi-thread architecture. More CPUs can improve QPS (10,000–10,000,000).

Generally, the average latency of single-point access is less than 1 ms, and the p99 latency is less than 2 ms, similar to that of open-source Redis.

For details about performance data, see **Performance Test Results**.

## 8.1.3 What Redis Versions and Commands Are Compatible with GeminiDB Redis API? Whether Application Code Needs to Be Refactored for Connecting to a Redis Client?

GeminiDB Redis API is fully compatible with Redis 6.2 (including 6.2.x) and earlier versions, such as 5.0, 4.0, and 2.8. It is partially compatible with Redis 7.0.

You can migrate data of Redis 6.2 and earlier instances (such as 5.0, 4.0, and 2.8) to GeminiDB Redis instances, without the need of code modifications. Any Redis client can be connected to GeminiDB Redis instances.

## 8.1.4 Can Data Be Migrated from Open-Source Redis to GeminiDB Redis API? What Are the Precautions?

Yes. Take care with the version and specifications before migration:

- Version: If the version of the source database is 6.2 or earlier (including 6.2.x), data can be directly migrated. If the version of the source database is later than 6.2, you need to evaluate the migration project and then migrate data to GeminiDB Redis 6.2. In the upper right corner of the console, choose Service Tickets > Create Service Ticket and contact the customer service.
- Specifications: Configure proper specifications based on QPS and data volumes of the source instance.

#### 8.1.5 What Is the Availability of a GeminiDB Redis Instance?

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 a DB instance after you buy the instance. The total service duration refers to the total running time of the DB instance.

#### 8.1.6 Are Total Memory and Total Capacity of a GeminiDB Redis Instance the Same? What Is the Relationship Between Memory and Capacity?

No.

In an open-source Redis instance, all data is stored in memory, and the total capacity is the amount of memory that can be utilized.

In a GeminiDB Redis instance, all data is stored in a high-performance shared storage pool, and hot data is stored in memory. Generally, you only need to pay attention to the total capacity and usage of the instance. The CPU usage increases as QPS increases. In this case, you need to scale up the specifications.

## 8.1.7 How Do I Select Proper Node Specifications and Node Quantity When Purchasing a GeminiDB Redis Instance?

When purchasing a GeminiDB Redis instance, pay attention to QPS and data volume. You can select **Fast configure** or **Standard configure** for **Instance Creation Method**.

- **Fast configure**: If 16 GB of storage space is used for a cluster, you can select **16 GB** in the **Instance Specifications** area. If QPS does not meet requirements, select higher specifications.
- **Standard configure**: Select specifications of compute and storage resources separately. The node specifications and number of nodes determine instance QPS, and the total instance capacity determines the maximum storage space. After selecting the node specifications, number of nodes, and total instance capacity, you can view the QPS and number of connections of the selected instance next to **Specification Preview**.

#### 8.1.8 Is a Primary/Standby or Cluster Deployment Mode Preferred for GeminiDB Redis Instances with Several GB of Storage Space?

A cluster is preferred. At least 4 GB of storage space is recommended. Compared with the primary/standby architecture, the cluster has better scalability and higher QPS. The GeminiDB Redis cluster has the following advantages:

• All compute nodes in the GeminiDB Redis cluster support writes and reads, and the node resource utilization is 100%. In the primary/standby mode,

shards on the standby nodes do not support writes, resulting in low resource utilization.

• Both a single node and a cluster can access the GeminiDB Redis cluster, which is a proxy cluster.

## 8.1.9 How Does GeminiDB Redis API Persist Data? Will Data Be Lost?

Open-Source Redis persists data periodically, so there is a high probability that data loss occurs in abnormal scenarios. GeminiDB Redis API data is updated to a storage pool in real time, improving data security.

Similar to other NoSQL databases, backend processes on the GeminiDB Redis API instance write write-ahead logs (WALs) into OS buffers. The buffers immediately return and then are updated to the storage pool. Therefore, a small amount of data may be lost in the case of an unexpected power failure.

GeminiDB Redis API ensures data is not lost during routine O&M, such as changing specifications, upgrading versions, and adding nodes. Synchronous writes greatly reduce write performance. To achieve higher data reliability, you need to enable synchronous writes. You can choose **Service Tickets > Create Service Ticket** in the upper right corner of the console.

## 8.1.10 What Is the Memory Eviction Policy of GeminiDB Redis API?

If keys of an open-source Redis instance are evicted from the memory, the key values cannot be read later.

By default, GeminiDB Redis API supports a noeviction policy, that is, user keys are not evicted. All data is stored in a storage pool. Hot data evicted from the memory can be read from the storage pool. The data is reloaded to the memory after being accessed, and user keys are not deleted.

Therefore, GeminiDB Redis API users do not need to set or modify the **maxmemory-policy** parameter. If unnecessary data is stored, users need to add an expiration time to avoid dramatical increase in data volumes.

## 8.1.11 Does GeminiDB Redis API Support Modules Such as a Bloom Filter?

A Bloom filter can be used to check whether an element is in a large-size data set. It is suitable for scenarios such as web interceptors and anti-cache penetration.

GeminiDB Redis API supports Bloom filters.

In addition, you can set an expiration time for individual fields in a hash shard. Shards can be scanned in parallel. Data can be imported extremely fast using FastLoad.

#### 8.2 Billing

## 8.2.1 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 post payment mode, so you can start or stop an instance at any time. Pricing is listed on a per-hour basis, but bills are calculated based on the actual usage duration.

For details, see **2.2.1 Overview**.

## 8.2.2 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.

- For details about how to change the billing mode from yearly/monthly to a pay-per-use, see **2.5.3 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 2.5.2 Changing a Pay-per-Use Instance to Yearly/Monthly.

#### 8.3 Database Usage

#### 8.3.1 Why Is the Key Not Returned Using Scan Match?

#### Symptom

As shown in the following figure, the value of key is **test** and exists in the database. However, no data is returned using this scan match command.

139.9.177.148:6379> scan 1 match tes\* 1) "21" 2) (empty list or set) 139.9.177.148:6379> get test "abc" 139.9.177.148:6379>scan 0 match tes\* 1) "21" 2) (empty list or set) 139.9.177.148:6379>

#### **Possible Causes**

The MATCH command is used to iterate elements that only match a specified pattern. Pattern matching is performed after the command obtains elements from the data set and before the elements are returned to the client. If all the extracted elements do not match the pattern, no element is returned.

#### Solution

If multiple scans are performed, the iteration is complete when the returned cursor is 0. The cursor returned from the last scan is used for the next scan.

#### 8.3.2 How Do I Process Existing Data Shards After Migrating Workloads to GeminiDB Redis API?

GeminiDB Redis API uses decoupled compute and storage and allows adding data shards dynamically, making scaling smooth.

After an GeminiDB Redis instance is connected, data sharding is not required on the service side.

## 8.3.3 Does GeminiDB Redis API Support Fuzzy Queries Using KEYS?

Yes.

Fuzzy queries using KEYS may cause OOM and longer latency.

KEYS can be used only in a test environment. In a production environment, use SCAN and MATCH instead.

## 8.3.4 Does the GeminiDB Redis API Support Multiple Databases?

GeminiDB Redis API allows you to create multiple databases in an instance since March 2022. Instances created before March 2022 do not support this function and cannot be upgraded to support it.

This feature has the following constraints:

- The number of databases ranges from 0 to 999.
- The SWAPDB command is not supported.
- The result of the **dbsize** command is not updated in real time. The result does not decrease to 0 immediately after **flushdb** is executed, and will change to 0 after a while.
- Executing SELECT and FLUSHDB commands in LUA scripts is not supported.
- Executing SELECT and FLUSHDB commands in transactions is not supported.
- The MOVE command is not supported.

#### 8.3.5 Why the Values Returned by Scan Operations Are Different Between GeminiDB Redis API and Open-Source Redis 5.0?

GeminiDB Redis API may return values in a different sequence from open-source Redis, but they both comply with open-source document description requirements. This is because open-source Redis does not specify the sorting rules for:

- Returned values of SCAN/HSCAN/SSCAN operations
- Returned values of ZSCAN operations ZSET when its elements have the same score

## 8.3.6 Why Is the Cursor Length Returned by SCAN of GeminiDB Redis API Is Longer Than That of Open-Source Redis?

Possible causes:

- The encoding rule of cursors returned by GeminiDB Redis API is different from that of open-source Redis, so the cursor returned by GeminiDB Redis API is longer. The cursor usage and behavior of GeminiDB Redis API and open-source Redis are the same, so data can be correctly traversed by clients.
- As the number of keys increases, the Redis cursor length grows longer as well. Therefore, the cursor length returned by GeminiDB Redis API is acceptable. After a SCAN command is executed, the returned cursors must be parsed as 64-bit unsigned integers. Failing to do so may cause parsing failures or incorrect results. If a database receives these incorrect results, error message "invalid cursor" will be reported.

#### 8.3.7 Why Are Error Messages Returned by Some Invalid Commands Different Between GeminiDB Redis API and Open-Source Redis 5.0?

GeminiDB Redis API checks command syntax and checks for keys each time it executes a command. However, open-source Redis has no specific rules and returns the results for invalid commands in random.

Therefore, error messages returned by some invalid commands may be different.

## 8.3.8 How Do I Resolve the Error "CROSSSLOT Keys in request don't hash to the same slot"?

#### Scenarios

When multi-key commands are executed in a GeminiDB Redis instance, the error "CROSSSLOT Keys in request don't hash to the same slot" may be reported.

#### **Error Cause**

Commands involving multiple keys were executed across slots in a GeminiDB Redis cluster instance. For example, EVAL and BRPOPLPUSH were executed across slots.

#### Solution

- Change key names and use hashtags to ensure that the keys are in the same slot. Avoid data skew when you use hashtags. For more information, see
   8.3.10 Which Commands Require Hashtags on GeminiDB Redis Cluster Instances?
- When hashtags cannot be used, change the instance type to primary/standby. For details, see **1.5 Compatible API and Versions**.

## 8.3.9 How Many Commands Can Be Contained in a GeminiDB Redis Transaction?

It is recommended that a transaction contain a maximum of 100 commands.

Exercise caution when using commands with time complexity of O(N).

### 8.3.10 Which Commands Require Hashtags on GeminiDB Redis Cluster Instances?

#### Hashtag Overview and Usage

Multi-key commands in a Redis Cluster must comply with the **hashtag mechanism**. Keys with the same hashtag must be allocated to the same hash slot to ensure the atomicity and performance of the multi-key commands. Otherwise, error "CROSSSLOT Keys in request don't hash to the same slot" will be reported. Rules for using a hashtag are as follows:

#### 1. Basic format

If a key contains a "{}" pattern, only the substring between the braces is hashed to obtain the hash slot.

For example, the hashtags of {user:1000}.profile and {user:1000}.settings are both user:1000. Therefore, they are allocated to the same hash slot.

2. Location

{} can appear anywhere in a key.

For example, the hashtag of foo{user:1000}bar is still user:1000.

Only the first {} is valid.

If a key contains multiple braces, only the substring between the first braces is used as a hashtag.

For example, the hashtag of {user:1000}.{profile} is user:1000.

3. Scenarios

- Transactions: In a Redis Cluster, all operations within a MULTI/EXEC block must be performed on keys on the same node. A hashtag ensures that these keys are allocated to the same hash slot.
- Lua scripts: A hashtag ensures all keys used by Lua scripts are in the same slot.
- Multi-key operations: string (MSET and MGET), LIST (BLPOP, BRPOP, BRPOPLPUSH, and RPOPLPUSH), SET (SDIFF, SDIFFSTORE, SINTER, SINTERSTORE, SINTERCARD, SUNION, and SUNIONSTORE), and ZSET (ZINTER, ZINTERSTORE, ZINTERCARD, ZUNION, ZUNIONSTORE, ZDIFF, ZDIFFSTORE, and ZRANGESTORE), key management (DEL, EXISTS, UNLINK, TOUCH, RENAME, RENAMENX, and SORT), STREAM (XREAD and XREADGROUP), and BITOP

Example of using a cluster:

1. String: MSET/MGET

#### • Setting multiple keys (user data)

mset {user:1000}:name "Alice" {user:1000}:email "alice@example.com" {user:1000}:age 30

• Obtaining multiple keys

mget {user:1000}:name {user:1000}:email {user:1000}:age

2. Transaction: MULTI/EXEC

• Starting a transaction

MULTI SET {order:1234}:status "processing" EXPIRE {order:1234}:status 3600 EXEC

3. LUA script:

• Reducing inventory scripts and recording logs

EVAL "redis.call('DECR', KEYS[1]); redis.call('SET', KEYS[2], 'updated')" 2 {product:100}:stock {product:100}:log

#### Splitting Commands Supported by Proxy Cluster GeminiDB Redis Instances

A proxy cluster can route commands, balance loads, and perform failovers. It can simplify the client-side logic while providing advanced features such as handling connections to multiple databases. You do not need to bother with shard management. The proxy cluster is recommended because it is compatible with a standalone Redis node, Redis Sentinel, and Redis Cluster.

The proxy cluster can simplify the logic of some multi-key commands by splitting and routing them to different backend nodes. After being executed, the commands are aggregated on the proxy and then returned to the client. Proxy cluster GeminiDB Redis instances support the following splitting commands:

- Key management: DEL, EXISTS, UNLINK, and TOUCH
- String: MGET and MSET
- SET: SDIFF, SDIFFSTORE, SINTER, SINTERSTORE, SINTERCARD, SUNION, and SUNIONSTORE
- ZSET: ZINTER, ZINTERSTORE, ZINTERCARD, ZUNION, ZUNIONSTORE, ZDIFF, ZDIFFSTORE, and ZRANGESTORE
- Multiple commands in a transaction can be split. If a transaction contains multi-key commands that cannot be split, hashtags must be added to keys involved in these commands.

Other commands cannot be split. You are advised to use a hashtag in the cluster to ensure the atomicity and performance of multi-key commands. Key management and string multi-key commands are more efficient than SET and ZSET. After being executed on shards, the commands are aggregated on the proxy. The outputs are returned to the client. To execute SET and ZSET, each key needs to be read to the proxy before related logic operations are performed. SET and ZSET are not recommended for big keys which lead to slower access and increased memory usage.

#### **Scenarios**

When **SENTINEL** commands are executed on a GeminiDB Redis instance, the error message "ERR unknown command sentinel" may be displayed.

#### **Error Cause**

If the value of **CompatibleMode** of cluster GeminiDB Redis instances is not **3** or the value of **CompatibleMode** of primary/standby GeminiDB Redis instances is not **2**, **SENTINEL** commands are not allowed.

#### Solution

- 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, choose **Parameters**.
- **Step 5** Change the value of **CompatibleMode** and click **Save**.
  - For a cluster instance, set **CompatibleMode** to **3**.
  - For a primary/standby instance, set **CompatibleMode** to **2**.

#### Figure 8-1 Changing parameters

| Basic Information                         |                                  |                          |            |                    |   |  |  |  |
|---|----------------------------------|--------------------------|------------|--------------------|---|--|--|--|
| Backups & Restorations<br>Node Management | Parameters Change History        |                          |            |                    |   |  |  |  |
| Accounts                                  | Save Cancel Preview              | Compare                  |            |                    | Enter a parameter name. Q   |  |  |  |
| Slow Query Logs                           | Parameter Name                   | Effective upon Restart 👙 | Value      | Allowed Values     | Description   |  |  |  |
| Parameters                                | AuthFail.ockTime                 | No                       | 5          | 0-10,000           | The length of time, in second, that a suspicious $\ensuremath{\mathbb{P}}$ ad |  |  |  |
| Sessions                                  | <b>BigbeysQuantityLimitation</b> | No                       | 100        | 1-10,000           | string hashlisticsel/sel/sel/sel/sel/sel/sel/sel/sel/sel/                     |  |  |  |
| Diagnosis Analysis<br>Rename High-risk    | CompatibleMode                   | No                       | 3 -        | 0, 1, 2, 3         | Whether StackExchange Redis is available. Set this                            |  |  |  |
| Command<br>Tags                           | EnableAcIDbDirect                | No                       | 100 v      | yes, no            | Is the DB direct function enabled. The default is false.                      |  |  |  |
|   | MaxAuthFalTimes                  | No                       | 5          | 0-10.000           | Maximum failed access attempts. When this limit is re                         |  |  |  |
|   | ProxyTimeout                     | No                       | 0          | 0-100.000          | The length of time, in seconds, that a proxy-client con                       |  |  |  |
|   | bigkeys-composite-threshold      | No                       | 1024       | 1-2,147,483,647    | A key of the hashilistizsel/sel/stream type whose nu                          |  |  |  |
|   | bigkeys-string-threshold         | ys-string-threshold No   |            | 1-2,147,483,647    | If the value is greater than the value of a string key, th                    |  |  |  |
|   | databases                        | No                       | 1000       | 1-1,000            | Allow a limit on the number of supported DBs.                                 |  |  |  |
|   | maxmemory-policy                 | Yes                      | nonviction | noeviction         | Whether new keys can be saved when the storage sp                             |  |  |  |
|   | notity-keyspace-events           | No                       |            | -                  | The type of event that needs to be monitored. The de                          |  |  |  |
|   | slowlog-threshold                | No                       | 300000     | 80,000-100,000,000 | Maximum time in microseconds for executing a query                            |  |  |  |

----End

#### 8.3.12 Why Return Values of Blocking Commands Differ Between Primary/Standby GeminiDB Redis Instances and Open-Source Redis Instances?

A return value is not specified when a blocking command is executed to block an open-source Redis client until keys are written concurrently.

Although their return values are different, both of them meet requirements described in open-source Redis documentation.

#### 8.3.13 How Long Does It Take to Scale Up GeminiDB Redis Instance Storage? Will Services Be Affected?

GeminiDB Redis instance storage can be scaled up in seconds, and services are not affected.

For details about how to scale storage, see **4.6.7.2 Manually Scaling Up Storage Space**.

For details about automated scale-up, see **4.6.7.3 Automatically Scaling Up Storage Space**.

## 8.3.14 How Long Does It Take to Add GeminiDB Redis Nodes at the Same Time? What Are the Impacts on Services?

GeminiDB Redis nodes can be added at the same time, which can be completed within 5 minutes.

#### NOTICE

Shared storage is used. After nodes are added, data does not need to be migrated, but slots are rebalanced. A retry mechanism is needed to avoid service interruptions due to a few seconds of jitter or latency.

#### 8.3.15 What Are the Differences Between Online and Offline Specification Changes of GeminiDB Redis Nodes? How Long Will the Changes Take? What Are the Impacts on Services?

- Online change: Nodes are changed in rolling mode. The change duration is positively related to the number of nodes. Each node takes about 5 to 10 minutes. In addition, both primary/standby and cluster instances contain three internal management nodes, which are changed at the same time. For example, three worker nodes and three internal management nodes are created for a GeminiDB Redis instance. The online change takes about 30 to 60 minutes. While specifications are changed, the node is disconnected, and its slots become disabled and are taken over by a functional node. In addition to node disconnection , there are also other interruptions in several seconds, for example, access timeout and invisible data partitions, so a reconnection mechanism must be established. You are advised to change the node specifications during off-peak hours and keep the CPU and memory usage at a low level. This prevents exceptions such as heavy load on other nodes and process startup failures.
- Offline change: Specifications of all nodes are changed concurrently. During the change, services are interrupted for about 10 to 20 minutes. Offline change is applicable when services are stopped or no service is accessed. Exercise caution when performing this operation.

For your online production services, you are advised to perform the change online. For details, see **4.6.4 Changing vCPUs and Memory**.

#### 8.3.16 What Are the Differences Between Online and Offline Patch Installation of GeminiDB Redis Nodes? How Long Will the Upgrades Take? What Are the Impacts on Services?

- Online patch installation: Nodes are upgraded in rolling mode. The upgrade duration is positively related to the number of nodes. Each node takes about 2 to 5 minutes. Both primary/standby and cluster instances contain three internal management nodes, which are upgraded at the same time. For example, a GeminiDB Redis instance consists of six nodes, including three worker nodes and three internal management nodes. The online upgrade takes about 12 to 30 minutes. During the upgrade of a single node, services are affected due to a few seconds of jitter. Therefore, a reconnection mechanism is required. You are advised to upgrade the node during off-peak hours and keep the CPU and memory usage at a low level. This prevents exceptions such as heavy load on other nodes and process startup failures.
- Offline patch installation: All nodes are upgraded concurrently. During the upgrade, services are interrupted for about 10 to 20 minutes. Offline upgrade is applicable when services are stopped or no service is accessed. Exercise caution when performing this operation.

For details about how to install the patch of GeminiDB Redis API, see **4.6.1 Upgrading a Minor Version**.

#### 8.3.17 Can I Download Backups of a GeminiDB Redis Instance to a Local PC and Restore Data Offline?

Backups of a GeminiDB Redis instance differ from RDB files of an open-source Redis instance and cannot be used by users. Therefore, the backups cannot be downloaded to a local PC.

If instance data is corrupted, you can restore backup data to a new instance.

For details, see 4.7.1 Overview and 4.8.2 Restoring Data to a New Instance.

## 8.3.18 What Is the Data Backup Mechanism of GeminiDB Redis API? What Are the Impacts on Services?

To back up data of a GeminiDB Redis instance, snapshots need be created in seconds only for the storage layer, which does not affect compute nodes. Therefore, services are not affected as well.

When backup data is uploaded, a small amount of CPU and bandwidth resources are consumed, which may cause slight jitter.

GeminiDB Redis instances support automated and manual backup. For details, see **4.7.1 Overview**.

#### 8.3.19 Why Does the CPU Usage Remain High Despite Low Service Access Volume on a GeminiDB Redis Preferential Instance with 1 CPU and 2 Nodes?

GeminiDB Redis API collects metrics and reports monitoring data. The CPU usage of your nodes is high because of its small specifications.

A GeminiDB Redis Preferential instance with one CPU and two nodes is recommended in the test environment. A GeminiDB Redis instance with one CPU (standard) or two or more CPUs is recommended in the production environment.

For details about instance specifications, see **1.6 Instance Specifications**.

#### 8.3.20 Why Does the Number of Keys Decrease and Then Become Normal on the Monitoring Panel on the GUI of GeminiDB Redis API?

The number of keys is scanned and counted asynchronously by a GeminiDB Redis server to ensure final consistency.

When an instance process is restarted (due to node restart, instance fault, specification change, or version upgrade), the keys are counted again. In this case, the number of keys displayed decreases temporarily and becomes accurate gradually.

## 8.3.21 Why Is CPU Usage of GeminiDB Redis Instance Nodes Occasionally High?

There are many possible reasons, such as sudden spike in service traffic, big key operations, network jitter, data backup and garbage recycle tasks on a server.

If the CPU usage is occasionally high, just ignore it.

If there are other service reasons (excluding high QPS), you can choose **Service Tickets > Create Service Ticket** in the upper right corner of the console and contact the customer service.

#### 8.3.22 How Do I Upgrade GeminiDB Redis API from 5.0 to 6.2?

GeminiDB Redis API is compatible with Redis 7.0, 6.2 (including 6.2.*x*), 5.0, and earlier versions.

To upgrade an existing instance, choose **Service Tickets** > **Create Service Ticket** in the upper right corner of the console and contact the customer service to enable the whitelist.

For details about the upgrade operations, see **4.6.1 Upgrading a Minor Version**.

#### 8.3.23 When Does a GeminiDB Redis Instance Become Read-Only?

To ensure that the GeminiDB Redis instance can still run properly when the storage space is about to be used up, the database is set to read-only, and data

cannot be modified. If this happens, you can scale up the storage to restore the database status to read/write.

| Storage Capacity | Description  |
|------------------|--|
| < 600 GB         | • When the storage usage reaches 97%, the instance is read-only.   |
|                  | • When the storage usage decreases to 85%, the read-only status is automatically disabled for the instance.                            |
| ≥ 600 GB         | • When the remaining storage space is less than 18 GB, the instance is read-only.  |
|                  | • When the remaining storage space is greater than or equal to 90 GB, the read-only status is automatically disabled for the instance. |

| Table o-I Setting an instance status to read-only | Table 8-1 | Setting an | instance | status | to | read-only |
|---|-----------|------------|----------|--------|----|-----------|
|---|-----------|------------|----------|--------|----|-----------|

#### 8.3.24 How Is the Number of Keys Collected? Why Is the Number of Keys Monitored by GeminiDB Inaccurately Increasing After Migration?

- The number of keys is not updated in real time. Instead, it is calculated by asynchronously scanning multiple nodes and aggregating their data. If the growth rate of keys exceeds the scan rate, for example, when data is migrated or when a large volume of data is injected quickly, the statistics collection will be delayed. The key scan speed is determined by the **key-scan-batch** parameter. Since this speed is constant, the growth of the key quantity maintains steady.
- To speed up the scan, go to the instance details page, choose **Parameters** in the navigation pane, search for **key-scan-batch**, and increase the value. For details, see **Modifying Parameters of an Instance**. Setting **key-scan-batch** to a larger value will consume more CPUs. During migration, you can increase the value. After the migration is complete, you are advised to use the default value to prevent excessive CPU consumption.

#### Figure 8-2 Modifying key-scan-batch



#### 8.4 Database Connection

#### 8.4.1 How Do I Connect to a GeminiDB Redis Instance?

You can connect to a GeminiDB Influx instance using a private network, public network, load balancer IP address, DAS, or program code. For details, see **4.3.1** Connecting to a GeminiDB Redis Instance.

- You can connect to a GeminiDB Redis instance using a web-based console client.
- You can connect to a GeminiDB Redis instance through a private IP address, private domain name, or load balancer address.
- You can connect to a GeminiDB Redis instance through a **public domain name** or an **EIP**.
- You can connect to a GeminiDB Redis instance using different code. For details, see 5.3 Examples of Connecting to an Instance Using Programming Languages.

## 8.4.2 How Do I Use Multiple Node IP Addresses Provided by GeminiDB Redis API?

GeminiDB Redis API provides multiple IP addresses for you to access a cluster and achieve load balancing and disaster recovery.

You can use multiple IP addresses in any of the following ways:

- 1. Use the connection pool on the service side implement load balancing and fault detection.
- Choose Service Tickets > Create Service Ticket in the upper right corner of the console. Contact the customer service to configure Elastic Load Balance (ELB) and provide a unique IP address.
- Configure domain names for multiple proxy IP addresses. For details about how to connect to an instance through a private domain name, see 4.3.3.1 Connecting to an Instance Using a Load Balancer Address (Recommended).

#### 8.4.3 How Does Load Balancing Work in GeminiDB Redis API?

GeminiDB Redis API uses dedicated load balancers with scalable specifications and supports a maximum bandwidth of 10 Gbit/s. For details, see **Dedicated Load Balancer Overview**.

#### 8.4.4 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 and security group with the GeminiDB Redis 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*.

#### 8.4.5 Can I Change the VPC of a GeminiDB Redis Instance?

After a GeminiDB Redis 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 4.8.2 Restoring Data to a New Instance.

#### 8.4.6 Why Can't I Connect to the Instance After an EIP Is Bound to It?

#### **Possible Cause**

The current port has not been enabled in the inbound rules of the security group.

#### Handling Procedure

- **Step 1** Click the instance name to go to the **Basic Information** page.
- **Step 2** In the **Network Information** area, click the value of the **Security Group** parameter.

#### Figure 8-3 Network information

| Network Information   | I       |   |                                 |  |
|-----------------------|---------|---|---------------------------------|--|
| VPC                   |         |   | Security Group                  | default 🖉  |
| Subnet                |         |   |                                 |  |
| Load Balancer Address | t6379 🗇 | Access Control<br>your load balancer<br>ol allows any IP ad | does not support security group | rules, so you need to configure access control for the li.<br>your instance to access your instance using the load b |

**Step 3** Click the **Inbound Rules** tab and click **Add Rule**. Then, specify parameters as shown in the following figure.

#### Figure 8-4 Inbound rules

| Add R | ule Fas               | t-Add Rule Delete | Allow Common Ports | Inbound Rules: 5 Learn more abo | ut security group configuration. |             |                       |                             |
|-------|-----------------------|-------------------|--------------------|---------------------------------|----------------------------------|-------------|-----------------------|-----------------------------|
| Q Sp  | ecify filter criteria | 1.                |                    |                                 |                                  |             |                       |                             |
|       |                       |                   |                    |                                 |                                  |             |                       |                             |
|       | Priority ⑦            | Action ⑦          | Туре               | Protocol & Port (?)             | Source (?)                       | Description | Last Modified         | Operation                   |
|       | 1                     | Allow             | IPv4               | TCP : 6379                      | default (1)                      |             | Nov 16, 2023 09:52:40 | Modify   Replicate   Delete |

You can also specify parameters by referring to **Configuring Security Group Rules**.

----End

## 8.4.7 How Do I Access a GeminiDB Redis Instance from a Private Network?

You can access a GeminiDB Redis instance through a load balancer or a directly-connected node.

- Access through a load balancer (recommended): The load balancer is associated with a high-availability backend cluster, using an internal IP address that is accessible only to clients. Periodical health checks are performed on backend nodes to prevent single points of failure (SPOFs).
- Access through a directly-connected node: An agent installed on a GeminiDB Redis node enables you to connect to any node. Then you can access the entire cluster. To prevent SPOFs, this access mode is only recommended in test scenarios.

For details about how to connect to a GeminiDB Redis instance over a private network, see **4.3.3 Connecting to a GeminiDB Redis Instance Over a Private Network**.

## 8.4.8 Do I Need to Enable Private Network Access Control for a Load Balancer After Setting a Security Group?

You can access a GeminiDB Redis instance through a node or load balancer. Therefore, you need to configure both a security group and private network access control for a load balancer to ensure instance security.

- Security groups take effect only for nodes. It is a collection of access control rules for ECSs and GeminiDB Redis instances that have the same security requirements and are mutually trusted in a VPC. For details, see 4.3.5.1
   Setting Security Group Rules for a GeminiDB Redis Instance.
- Security groups cannot take effect for load balancers. If access control is disabled, all IP addresses that can access the VPC of the GeminiDB Redis instance also can access the instance using a load balancer IP address. Therefore, you need to configure access control properly. For details, see
   4.3.5.8 Configuring Private Network Access to a GeminiDB Redis Instance.

#### 8.4.9 What Should I Do If the Client Connection Pool Reports Error " Could not get a resource from the pool"?

#### Scenarios

A large number of GeminiDB Redis instance requests are suspended on the clients, and the following error is displayed in the abnormal stack:

- Jedis client: redis.clients.jedis.exceptions.JedisConnectionException: Could not get a resource from the pool
- Lettuce client:
   redis.connection.lettuce.LettucePoolingConnectionProvider.getConnection
- Go-redis client: redis: connection pool timeout You can get that error when there are no free connections in the pool for Options.PoolTimeout duration. If you are using redis.PubSub or redis.Conn, make sure to properly release PuSub/Conn resources by calling Close method when they are not needed any more. You can also get that error when Redis processes commands too slowly and all connections in the pool are blocked for more than PoolTimeout duration.

However, the instance information queried by following **4.13.4 Viewing Metrics** shows that the database QPS, latency, and number of connections are normal and no slow request is displayed.

#### **Possible Causes**

Generally, the preceding issue is caused by incorrect configurations of a client connection pool. The maximum number of the connection pools is limited for applications. If the QPS of an application exceeds the limit of the connection pool or connections are not released in time, the thread cannot obtain new connections and services are affected.

#### Solution

Check whether the QPS and traffic metrics increase sharply last two hours and whether connection pool parameters configured on the Redis clients (such as Jedis and Lettuce) meet service requirements.

#### **NOTE**

For details about how to configure the Redis clients, see **5.1 Development and O&M Rules** and **5.10 Configuring Parameters for a Client Connection Pool**.

#### 8.4.10 Common Client Errors and Troubleshooting Methods

#### Symptom 1

- The client displayed a message indicating that the network timed out for 10 seconds and the connection failed.
   CommonResponseAspect exception!Redis command timed out; nested exception is io.lettuce.core.RedisCommandTimeoutException: Command timed out after 10 second(s)
- Client: Lettuce
- Possible cause: The bandwidth of the client is used up.
- Solution: Check service resources and solve the resource bottleneck.

#### Symptom 2

- The client occasionally displayed a message indicating that the connection was unavailable. [redisClient=[addr=XXXX], channel=[id: 0x0a0d20bc, L:0.0.0/0.0.0:53192]] is not active!
- Client: Redisson
- Possible cause: The client reconnection mechanism is incomplete. This issue may occur after an HA switchover is triggered on servers.
- Solution: Restart the client.

#### Symptom 3

- Error "Could not get a resource from the pool" was displayed, and there were a large number of TCP connections in the **CLOSE\_WAIT** state on ECSs running the service program.
- Clients: Jedis and Lettuce
- Possible cause: The connection pool size configured for the client program is too small. As a result, connections to the Redis instance are insufficient when concurrent services increase dramatically.
- Solution: Check service code and configure sufficient size for the connection pool.

#### Symptom 4

• The connection pool of the client program timed out. The error information is as follows:

"redis: connection pool timeout You can get that error when there are no free connections in the pool for Options.PoolTimeout duration. If you are using redis.PubSub or redis.Conn, make sure to properly release PuSub/Conn resources by calling Close method when they are not needed any more. You can also get that error when Redis processes commands too slowly and all connections in the pool are blocked for more than PoolTimeout duration."

- Client: Go-redis
- Possible cause: The connection pool size configured for the client program is too small. As a result, connections to the Redis instance are insufficient when concurrent services increase dramatically.
- Solution: Check service code and configure sufficient size for the connection pool.

#### 8.5 Backup and Restoration

### 8.5.1 How Long Can a GeminiDB Redis Instance Backup Be Saved?

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 manual backups as needed.

For more backup information, see **4.7.2 Managing Automated Backups** and **4.7.3 Managing Manual Backups**.

#### 8.6 Regions and AZs

#### 8.6.1 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**.

#### 8.6.2 Can I Change the Region of a GeminiDB Redis Instance?

Not supported. After an instance is created, its region cannot be changed.

#### NOTICE

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.

#### 8.7 Data Migration

## 8.7.1 What Do I Do if the Error "ERR the worker queue is full, and the request cannot be executed" Is Displayed?

If the internal queue is full due to massive volumes of data being migrated, choose **Service Tickets > Create Service Ticket** in the upper right corner of the console and contact the customer service.

## 8.7.2 What Do I Do If the Error "ERR the request queue of io thread is full, and the request cannot be executed" Is Displayed?

If the internal queue is full due to massive volumes of data being migrated, choose **Service Tickets > Create Service Ticket** in the upper right corner of the console and contact the customer service.

## 8.7.3 What Do I DO If the Error "read error, please check source redis log or network" Is Displayed?

The sending buffer of the source end is too small. You need to modify the Redis parameter configuration of the source end. This parameter takes effect immediately after the value of **client-output-buffer-limit** is changed.

#### 8.7.4 What Do I Do If the Error "slaveping\_thread.cc-ThreadMain-90: error: Ping master error" Is Displayed?

When the **pika-port** command is executed, the specified IP address is **127.0.0.1**. You need to set it to another IP address, for example, the IP address of eth0.

## 8.7.5 What Do I Do If the Forward Migration Speed of the Synchronization Status Is Too Slow?

The value of **source.rdb.parallel** of redis-shake to adjust the migration concurrency. The default value is **0**, which is determined by the number of databases and shards.

#### 8.7.6 What Do i Do When the Forward Migration Speed of the Synchronization Status Is Too Fast, and the Error Message "ERR Server Reply Timeout, Some Responses May Lose, but Requests Have Been Executed" Is Displayed?

The value of **parallel** is changed to adjust the concurrency of RDB transmission in the full process. The default value is **32**.

## 8.7.7 Can Data Be Migrated from Self-Built Redis 4.0, 5.0, and 6.2 to GeminiDB Redis API?

Yes. GeminiDB Redis API is compatible with Redis 6.2 (including 6.2.x) and earlier versions, such as 5.0, 4.0, and 2.8.

## 8.7.8 How Do I Migrate Data from Self-Built Primary/Standby and Cluster Redis Instances to GeminiDB Redis Instances?

DRS can be used for online migration.

- For details about how to migrate data from a single Redis instance to primary/standby GeminiDB Redis instances, see From Redis to GeminiDB Redis.
- For details about how to migrate data from a Redis cluster to a GeminiDB Redis cluster, see From Cluster Redis to GeminiDB Redis.

For details about how to migrate RDB files to GeminiDB Redis instances, see **4.4.7** (Recommended) Importing Data to Restore RDB Files to a GeminiDB Redis Instance.

#### 8.7.9 Why Cannot DRS Migrate Data from Third-Party Redis Such as ApsaraDB for Redis and TencentDB for Redis?

Generally, there are the following possible causes:

- Some self-developed Redis-like databases are not compatible with the PSync protocol.
- Architecture restrictions: For many cloud vendors, the proxy component is added between users and Redis. PSync is not supported due to the proxy.
- Security restrictions: In native Redis, fork() is used over PSync, which causes memory expansion, user request delay increase, and even out of memory.
- Business strategy: A large number of users use RedisShake to migrate services from the cloud or change the cloud, so PSync is shielded.

Generally, you can use a data migration service suitable for the corresponding cloud service. For details, see **4.4.1 Migration Solution**.

#### 8.7.10 Which of the Following Factors Need to Be Considered When Data Is Migrated from Self-Built Primary/Standby Redis Instances to a GeminiDB Redis cluster?

GeminiDB Redis API is deployed in a proxy cluster and can be directly accessed by a single node or primary/standby nodes. No modification is required. Multi-key operations of primary/standby Redis instances are different from those of a Redis cluster, so services need to be modified.

All data of self-built single and primary/standby Redis instances is stored on the same node. Therefore, atomicity of multi-key operations, such as Lua, RPOPLPUSH, SDIFF, and SUNION, can be ensured. In a self-built Redis cluster, the modular hash function is executed for keys to determine which shard (node) will process the keys. Therefore, it is difficult to ensure atomicity when operations are performed on multiple keys across shards. To ensure atomicity of multi-key operations in a cluster, the Redis cluster uses a hashtag to ensure that multi-key operations are performed on the same node.

To use the hashtag, add the same strings to those multiple keys, for example, *{aaa}list1* and *{aaa}list2*. When processing the preceding keys, Redis identifies *{}* and calculates hash values only based on *aaa*. Therefore, multi-key operations are performed on the same node.

For details about commands for adding hashtags in a GeminiDB Redis cluster, see **8.3.10 Which Commands Require Hashtags on GeminiDB Redis Cluster Instances?**.

## 8.7.11 Only 20% to 30% of 100 GB of Data Was Migrated to GeminiDB Redis. Is the Migration Incomplete?

GeminiDB Redis API offers an ultra-high data compression ratio of 4:1. Migrated data of multiple users complies with this rule. You can verify data consistency to determine whether the migration is complete. For example, verify key samples and the number of keys.

#### 8.8 Memory Acceleration

#### 8.8.1 Will All Data Be Cached to GeminiDB Redis Instances After Memory Acceleration Is Enabled and MySQL Database Data Is Updated?

No. You need to specify conversion rules of the MySQL database tablespaces, table names, and fields of GeminiDB Redis instances on the GUI. After the configuration is complete, data that meets the rules is automatically synchronized to a GeminiDB Redis instance.

#### 8.8.2 If Memory Acceleration Is Enabled, GeminiDB Redis Instance Data Increases Continuously. Do I Need to Scale Out the Capacity? How Do I Manage Cached Data?

By default, each piece of data of GeminiDB Redis instances will expire in 30 days. You can adjust the expiration time. If the data volume keeps increasing, you need to scale out storage capacity of GeminiDB Redis instances in a timely manner.

#### 8.8.3 Is Memory Acceleration Recommended When Customers' Service Data Can Be Synchronized Between MySQL and Redis? In Which Scenarios Can Memory Acceleration Be enabled?

If customers' service data can be synchronized between MySQL and Redis, you are advised to migrate cache data to GeminiDB Redis instances. Memory acceleration is recommended for new services to simplify development.

## 8.8.4 How Long Is the Latency of Synchronization from RDS for MySQL to GeminiDB Redis API? What Factors Affect the Latency?

Data can be synchronized in real time. The latency may be affected by the following factors and needs to be measured:

- Physical distance between RDS for MySQL and GeminiDB Redis instances. It is recommended that the instances be in the same region.
- You are advised to set the CPU specifications of RDS for MySQL to GeminiDB Redis instances to the same value.

## 8.8.5 Will the Source MySQL Database Be Affected After Memory Acceleration Is Enabled?

Memory acceleration works based on MySQL binlogs, which has little impact on the source MySQL database.

#### 8.8.6 GeminiDB Redis Instances with Memory Acceleration Enabled Needs to Process a Large Number of Binlogs in a Short Period of Time. Will a Large Number of Resources Be Occupied and Online Services Be Affected?

If a large number of DDL operations are performed on the source MySQL database, a large number of GeminiDB Redis resources are consumed. You can query OPS (**dbcache\_ops\_per\_sec**) after memory acceleration is enabled. You are advised to configure basic resource alarms. For details, see **4.13.2 Configuring** Alarm Rules.

## 8.9 Freezing, Releasing, Deleting, and Unsubscribing from Instances

#### Why Are My GeminiDB Redis 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 **Service Suspension and Resource Release**.

#### Why Are My GeminiDB Redis 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 GeminiDB Redis 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 GeminiDB Redis instances are frozen because your account is in arrears, you can unfreeze them by renewing them or topping up your account. The frozen GeminiDB Redis instances can be renewed, released, or deleted. Expired yearly/ monthly instances cannot be unsubscribed from.

### 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 Redis 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 Redis 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 Redis 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 **4.5.4.4 Recycling a GeminiDB Redis 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 Redis Instance?

- To delete a pay-per-use instance, see **4.5.4.3 Deleting a Pay-per-Use Instance**.
- To delete a yearly/monthly instance, see **2.11.4 How Do I Unsubscribe from** a Yearly/Monthly Instance?.