Graph Engine Service

User Guide

 Issue
 01

 Date
 2025-01-16





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 GES Overview	1
2 Preparations	3
3 Permissions Management	4
3.1 Creating a User	4
3.2 Policy Permissions	5
3.2.1 Policy	6
3.2.2 System-Defined Policies	6
3.2.3 Custom Policies	9
3.2.4 Request Conditions	11
3.2.5 GES Resources	12
3.3 Role Permissions	12
3.4 Reducing Extensive Permissions of Cloud Service Agencies	16
4 Metadata Operations	
4.1 Graph Data Formats	
4.1.1 Static Graph	18
4.1.2 Dynamic Graph	
4.2 Importing a Metadata File	25
4.2.1 Preparing Metadata	26
4.2.2 Importing Data From a Local Path or OBS	
4.3 Creating a Metadata File	27
4.4 Copying a Metadata File	
4.5 Editing a Metadata File	
4.6 Searching for a Metadata File	
4.7 Deleting a Metadata File	30
5 Creating Graphs	32
5.1 Methods to Create a Graph	32
5.2 Creating a Graph Without Using a Template	32
5.3 Creating a Graph Using an Industry-Specific Template	40
5.4 Creating a Dynamic Graph	41
5.5 Starting a Graph	41
5.6 Stopping a Graph	
5.7 Accessing Graphs	43

5.8 Importing Incremental Data	
6 Managing Graphs	46
6.1 Graph Management Overview	46
6.2 Viewing a Failed Graph	47
6.3 Backing Up and Restoring Graphs	
6.3.1 Backing Up a Graph	
6.3.2 Restoring a Graph	49
6.3.3 Deleting a Backup	50
6.3.4 Exporting a Backup to OBS	51
6.3.5 Importing a Backup from OBS	52
6.4 Upgrading a Graph	53
6.5 Rolling Back a Graph	53
6.6 Exporting a Graph	54
6.7 Restarting a Graph	55
6.8 Resizing a Graph	55
6.9 Expanding a Graph	57
6.10 Binding and Unbinding an EIP	58
6.11 Clearing Data	59
6.12 Deleting a Graph	59
6.13 Viewing Monitoring Metrics	60
6.14 Querying Schema	61
6.15 Connecting GES to LTS	62
6.16 Changing a Security Group	65
6.17 Changing the Security Mode	
7 Data Migration	68
7.1 Functions	68
7.2 Creating a Data Source	70
7.3 Creating a Data Migration Task	72
8 Accessing and Analyzing Graph Data	77
8.1 Accessing an Infinite Graph	77
8.2 Graph Editor	81
8.3 Accessing the Graph Editor	
8.4 Dynamic Graphs	90
8.4.1 Timeline	
8.4.2 Community Evolution	
8.4.3 Temporal BFS	
8.4.4 Temporal Paths	96
8.5 Graph Exploration	
8.6 Multi-Graph Management (Database Edition)	101
8.7 HyG Graph Management (Database Edition)	
8.8 Adding Custom Operations	

8.9 Editing Schema	100
8.10 Hiding Sensitive Information About a Graph	116
8 11 Visual Ouerv	118
8.12 Canvas Snapshot	
8.13 Gremlin Ouerv	
8.14 Cypher Ouery	
8.15 DSL Query	
8.16 Analyzing Graphs Using Algorithms	
8.17 Managing Indexes	
8.18 Analyzing Graphs on the Canvas	
8.19 Graph Display in 3D View	
8.20 Filter Criteria	
8.21 Editing Properties	
8.22 Statistics Display	
8.23 Check Running Records	
8.24 Viewing Query Results	150
9 Viewing Graph Tasks	153
9.1 Graph Overview	153
9.2 Task Center	155
9.2.1 Management Plane Task Center	156
9.2.2. Service Plane Task Center	157
9.3 Viewing Monitoring Metrics	158
9.4 Managing Connections	
10 Configuring Permissions	160
10 Configuring Permissions	
10.1 Configuring Granular Permissions	
10.2 User Groups	
TU.3 User Details	
11 O&M Monitoring	175
11.1 Monitoring Metrics	
11.2 Graph Instance O&M Monitoring	
11.3 Monitoring	
11.3.1 Nodes	
11.3.2 Performance	
11.3.3 Real-Time Queries	
11.3.4 Historical Queries	
11.4 Monitoring Clusters Using Cloud Eye	
12 Package Management	
13 Algorithms	
13.1 Algorithm List	
13.2 PageRank	

13.3 PersonalRank	
13.4 K-core	203
13.5 K-hop	
13.6 Shortest Path	
13.7 All Shortest Paths	
13.8 Filtered Shortest Path	
13.9 SSSP	208
13.10 Shortest Path of Vertex Sets	
13.11 n-Paths	
13.12 Closeness Centrality	
13.13 Label Propagation	
13.14 Louvain	
13.15 Link Prediction	
13.16 Node2vec	
13.17 Real-time Recommendation	
13.18 Common Neighbors	219
13.19 Connected Component	
13.20 Degree Correlation	
13.21 Triangle Count	221
13.22 Clustering Coefficient	
13.23 Betweenness Centrality	222
13.24 Edge Betweenness Centrality	224
13.25 Origin-Destination Betweenness Centrality	226
13.26 Circle Detection with a Single Vertex	228
13.27 Common Neighbors of Vertex Sets	
13.28 All Shortest Paths of Vertex Sets	
13.29 Filtered Circle Detection	
13.30 Subgraph Matching	232
13.31 Filtered All Pairs Shortest Paths	233
13.32 Filtered All Shortest Paths	235
13.33 TopicRank	236
13.34 Filtered n-Paths	
13.35 Temporal Paths	



Graph Engine Service (GES) facilitates query and analysis of multi-relational graph data structures. It is particularly well suited for scenarios requiring analysis of rich relationships, including social network analysis, marketing recommendations, social listening, information distribution, and fraud detection.

This document describes how to operate and analyze graph data on the GES management console.

The following figure shows the procedure to use GES.

Figure 1-1 Use procedure

Preparations	Metadata import	Graph creation	Graph management	Graph analysis	Tasks
Registering with HUAWEI CLOUD	Importing data locally	Customizing graph creation	Managing graphs	Analyzing graph data	Dashboard
Granting GES permissions	Importing data from OBS	Creating a graph using a template			Task center

 Table 1-1 Process description

Task	Sub Task	Description	Instructions
Preparatio ns	Creating a Huawei ID	Before using GES, create a Huawei ID.	Creating a Huawei ID and Enabling Huawei Cloud Services
	Granting GES permissions	Grant GES permissions to a user group and add a user to the user group.	Granting GES permissions
Metadata import	Importing metadata from a local path	Import the metadata file to GES for graph creation.	Importing a metadata file from a local path

Task	Sub Task	Description	Instructions
	Importing metadata from OBS	Upload the prepared metadata file to an OBS bucket.	Importing metadata from OBS
Graph creation	Creating a graph	Create a graph without using a template.	Creating a graph
Graph creation	Creating a dynamic graph	Create a graph with the dynamic graph template.	Creating a dynamic graph
Graph Managem ent	Managing graphs	Back up, restore, resize, expand, and upgrade a graph.	Managing graphs
Graph analysis	Analyzing graph data	Use the Graph Editor to query and analyze graph data.	Analyzing graph data
Tasks	Dashboard	The overview page displays information about your resources, including basic graph information and billing details.	Dashboard
	Task center	The task center displays details about asynchronous tasks, such as creating, backing up, starting, and deleting graphs.	Task center

2 Preparations

Before using GES, create a Huawei ID.

Creating a Huawei ID and Enabling Huawei Cloud Services

Skip this step if you have created one.

- **Step 1** Log in to the **Huawei Cloud** official website.
- **Step 2** Click **Register** in the upper right corner to access the registration page.
- **Step 3** Complete the registration as instructed. For details, see Account Registration Process.

----End

3 Permissions Management

3.1 Creating a User

If you need to assign different permissions to employees in your enterprise to access GES resources, **Identity and Access Management**(IAM) is a good choice for fine-grained permissions management.

With IAM, you can:

- Create IAM users for your employees within your Huawei Cloud account based on your company's organizational structure. This allows each employee to have their own security credentials and access to GES resources.
- Grant users only the permissions required to perform a given task.
- Entrust an account of Huawei Cloud or cloud service to perform professional and efficient O&M on your GES resources.

If your Huawei Cloud account does not need individual IAM users, you may skip over this section.

Permission Type

Туре

- Roles: A type of coarse-grained authorization mechanism that defines permissions related to user responsibilities. There are only a limited number of roles. When using roles to grant permissions, you need to also assign dependency roles. However, roles are not an ideal choice for fine-grained authorization and secure access control.
- Policies: A type of fine-grained authorization mechanism that defines permissions required to perform operations on specific cloud resources under certain conditions. Policies allow for more flexible permissions control than roles. They allow you to meet requirements for more secure access control. For example, you can grant GES users only the permissions for managing a certain type of cloud servers. For the API actions supported by GES, see Permissions Policies and Supported Actions.

GES ReadOnlyAccess is a policy.

Procedure

This section describes how to use a group to grant permissions to a user. **Figure 3-1** shows the process.



Figure 3-1 Granting GES permissions

1. Create a user group and assign permissions.

Create a user group on the IAM console, and assign the **GES ReadOnlyAccess** policy to the group.

2. Create a user and add it to a user group.

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

3. Log in as the user you created and verify permissions.

Log in to the management console using the user your created and verify the user permissions.

- Choose Service List > Graph Engine Service to enter the GES management console, and click Create Graph in the upper right corner to create a graph. If you cannot create one, the GES ReadOnlyAccess policy has taken effect.
- Choose any other service in Service List. If a message appears indicating that you have insufficient permissions to access the service, the GES ReadOnlyAccess policy has taken effect.

3.2 Policy Permissions

3.2.1 Policy

IAM supports both system-defined and custom policies.

System-defined Policies

System-defined policies cover various common actions of a cloud service. Systemdefined policies can be used to assign permissions to user groups, but they cannot be modified.

The system-defined policies for GES include **GES FullAccess**, **GES Development**, and **GES ReadOnlyAccess**. These policies are recommended as they can cover most of the role assignments your will need in most scenarios. For details, see **GES System-defined Policy**.

Custom Policies

If the supplied system policies are unable to meet your needs, you can create custom policies for more refined control. You can create custom policies in the visual editor or using a JSON editor. For details, see **GES Custom Policy**.

3.2.2 System-Defined Policies

Policy Name	Description
GES FullAccess	Permissions for all operations on GES, including creating, deleting, accessing, and updating graphs.
	NOTE
	 Users with the permissions of this policy also need the following policy permissions granted: Tenant Guest, Server Administrator, and VPC Administrator.
	 To bind or unbind an EIP, you need the Security Administrator permission to create agencies. The Security Administrator role has extensive permissions and can be replaced with the following custom policies: iam:permissions:listRolesForAgen- cyOnD, iam:permissions:listRolesForAgency, iam:roles:listRoles, iam:permissions:listRolesForAgencyOn- Project, iam:agencies:listAgencies, iam:roles:createRole, iam:permissions:grantRoleToAgencyOnDomain, iam:agencies:getAgency, iam:agencies:createAgency, and iam:permissions:grantRoleToAgencyOnProject.
	• To use resources stored on OBS for other services, you need the OBS OperateAccess permission. OBS is a global service. You can find the corresponding OBS policy in the Global service project scope.
	 When granting GES FullAccess to an enterprise project, you need to configure the following permissions policies in IAM:
	 ecs:availabilityZones:list. For details, see AZ Management.
	 ecs:cloudServerNics:update. For details, see NIC Management.

 Table 3-1 GES system-defined policies

Policy Name	Description
GES Development	Operator permissions for all operations except creating, deleting, resizing, and expanding graphs.
	 To bind or unbind an EIP, you also need to have the Security Administrator role to create agencies. The Security Administrator role has extensive permissions and can be replaced with the following custom policies: iam:permissions:listRolesForAgencyOnD, iam:permissions:listRolesForAgency, iam:roles:listRoles, iam:permissions:listRolesForAgencyOnProject, iam:agencies:listAgencies, iam:roles:createRole, iam:permissions:grantRoleToAgencyOnDomain, iam:agencies:updateRole, iam:permissions:grantRoleToAgency, and iam:permissions:grantRoleToAgencyOnProject.
	• To use resources stored on OBS for other services, you need the OBS OperateAccess permission. OBS is a global service. You can find the corresponding OBS policy in the Global service project scope.
GES ReadOnlyAccess	Read-only permissions for viewing resources, such as graphs, metadata, and backup data.
	To use resources stored on OBS for other services, you need the OBS OperateAccess permission. OBS is a global service. You can find the corresponding OBS policy in the Global service project scope.

It takes about 13 minutes for an OBS role to take effect after being applied to a user or group. A policy takes about 5 minutes.

Table 3-2 Commor	operations	supported	by each	system-defined	policy
------------------	------------	-----------	---------	----------------	--------

Operation	GES FullAccess	GES Development	GES ReadOnlyAcc ess	Resource
Querying the graph list	Yes	Yes	Yes	-
Querying graph details	Yes	Yes	Yes	graphName
Creating graphs	Yes	No	No	graphName
Accessing graphs	Yes	Yes	No	graphName
Stopping graphs	Yes	Yes	No	graphName
Starting graphs	Yes	Yes	No	graphName

Operation	GES FullAccess	GES Development	GES ReadOnlyAcc ess	Resource
Deleting graphs	Yes	No	No	graphName
Importing Incremental data to graphs	Yes	Yes	No	graphName
Exporting graphs	Yes	Yes	No	graphName
Clearing graphs	Yes	Yes	No	graphName
Upgrading graphs	Yes	Yes	No	graphName
Resizing a graph	√	No	No	graphName
Expanding a Graph	\checkmark	No	No	graphName
Restarting a Graph	\checkmark	Yes	No	graphName
Binding EIPs	Yes	Yes	No	graphName
Unbinding an EIP	Yes	Yes	No	graphName
Querying backups of all graphs	Yes	Yes	Yes	-
Querying backups of a graph	Yes	Yes	Yes	-
Adding backups	Yes	Yes	No	backupName
Deleting a graph backup	Yes	Yes	No	backupName
Querying the metadata list	Yes	Yes	Yes	-
Querying metadata	Yes	Yes	Yes	metadataNa me
Verifying metadata	Yes	Yes	No	-
Adding metadata	Yes	Yes	No	metadataNa me
Deleting metadata	Yes	Yes	No	metadataNa me

Operation	GES FullAccess	GES Development	GES ReadOnlyAcc ess	Resource
Querying task statuses	Yes	Yes	Yes	-
Querying the task list	Yes	Yes	Yes	-
Configuring fine- grained permissions	\checkmark	Yes	No	-
Configuring user groups	\checkmark	Yes	No	-
Importing IAM users	\checkmark	Yes	No	-
Viewing user details	\checkmark	Yes	Yes	-

3.2.3 Custom Policies

In addition to the system-defined policies of GES, you can also create your own custom policies. For the actions supported for custom policies, see **Permissions Policies and Supported Actions**.

You can create custom policies using the visual editor or by editing a JSON file:

- Visual editor: Just select the relevant cloud services, actions, resources, and request conditions. You do not need to understand policy syntax.
- JSON: You can create a policy using a JSON file or edit the JSON file for an existing policy.

For details, see **Creating a Custom Policy**.

Examples

• Example 1: Allowing users to query and operate graphs

```
{
    "Version": "1.1",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ges:*:get*",
                "ges::list*",
                "ges:graph:operate"
        ]
     }
]
```

• Example 2: Preventing graph deletion

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

If you need to assign the **GES FullAccess** policy to a user but also forbid that user from deleting graphs, you can create a custom policy that blocks graph deletion, and then assign both policies to the group the user belongs to. The user will be granted full access based on the system policy, but the custom policy will then override the permission allowing graph deletion. The following is an example of a deny policy:

```
{
    "Version": "1.1",
    "Statement": [
        {
            "Effect": "Deny",
                "Action": [
                    "ges:graph:delete"
              ]
        }
]
```

• Example 3: Authorizing users to perform operations on graphs whose name prefix is **ges_project** (**ges_project** names are case insensitive) and access the graph list

```
"Version": "1.1",
"Statement": [
   {
      "Effect": "Allow",
      "Action": [
         "ges:graph:create",
         "ges:graph:delete",
         "ges:graph:access",
         "ges:graph:getDetail"
      ],
      "Resource": [
         "ges:*:*:graphName:ges_project*"
      ]
   },
   {
      "Effect": "Allow".
      "Action": [
         "ges:graph:list"
      1
  }
]
```

• Example 4: Authorizing users to operate only some graph resources, but allowing them to view all resources

The policy consists of the following two parts:

- Part 1: Authorizing users to perform operations on resources whose name prefix is ges_project. The resources include graphs and backups.
- Part 2: Authorizing users to query the graph, backups, tasks, and metadata lists, and view job details

```
{
    "Version": "1.1",
    "Statement": [
        {
            "Action": [
            "ges:backup:delete",
            "ges:graph:access",
            "ges:graph:operate",
            "ges:graph:
```

}



3.2.4 Request Conditions

Request conditions are useful in determining when a custom policy takes effect. A request condition consists of a condition key and operator. Condition keys are either global or service-level and are used in the Condition element of a policy statement. **Global condition keys** (starting with **g**:) are available for operations of all services, while service-level condition keys (starting with a service name such as **ges**) are available only for operations of a specific service. An operator is used together with a condition key to form a complete condition statement.

GES has a group of predefined condition keys that can be used in IAM. For example, to define an allow permission, you can use the condition key **hw:Sourcelp** to match requesters by IP address. The following table shows the request conditions that are used with GES.

Condition Key	Туре	Description
g:CurrentTime	Date and time	Time when an authentication request is received
		NOTE The time is in ISO 8601 format, for example, 2012-11-11T23:59:59Z.
g:MFAPresent	Boolean	Whether multi-factor authentication is used for user login
g:UserId	String	User ID used for current login
g:UserName	String	Username used for current login
g:ProjectName	String	Project of the current login
g:DomainName	String	Domain of the current login

Table 3-3 Request conditions

3.2.5 GES Resources

A resource is an object that exists within a service. On GES, you can select these resources by specifying their paths.

Table 3-4 GES resources and their paths

Resource	Resource Name	Path
graphName	GES graph name	graphName
backupNam e	GES backup name	backupName

3.3 Role Permissions

Roles can be used for fairly coarse-grained permissions control. They grant servicelevel permissions based on user responsibilities. GES does not support custom roles. The following system roles are available.

 Table 3-5
 System roles

Role Name	Description
Tenant Guest	Regular tenant usersPermissions: querying GES resourcesScope: project-level service
GES Administrator	 GES administrator Permissions: performing any operations on GES resources Scope: project-level service NOTE If you have the Tenant Guest, Server Administrator, and VPC Administrator permissions, you can perform any operations on GES resources. If you do not have the Tenant
	 Ges properly. If you need to bind or unbind an EIP, you need the Security Administrator permissions to create agencies.
	 If GES needs to interact with OBS, for instance, when creating and importing data, OBS permissions are required. For details, see Common GES operations supported by each OBS policy. When granting OBS permissions, specify the permission scope as global service resources.

Role Name	Description
GES Manager	GES manager
	 Permissions: performing any operations on GES resources other than creating, deleting, resizing, and expanding graphs
	Scope: project-level service
	NOTE If you have both Tenant Guest and Server Administrator permissions, you can perform any operations on GES resources except for creating and deleting graphs. If you do not have the Tenant Guest permission, you cannot use GES properly.
	Security Administrator and Server Administrator permissions.
	 If GES needs to interact with OBS, for instance, when importing data, OBS permissions are required. For details, see Common GES operations supported by each OBS policy. When granting OBS permissions, specify the permission scope as global service resources.
GES Operator	Regular GES users
	Permissions: viewing and accessing GES resources
	Scope: project-level service
	NOTE
	• If you have both the GES Operator and Tenant Guest permissions, you can view and access GES resources. If you do not have the Tenant Guest permissions, you cannot view resources or access graphs.
	 To interact with OBS, for instance, to view the metadata, you need the OBS permissions. For details, see Common GES operations supported by each OBS policy.

Table 3-6 Common GES operations supported by each role

Operation	GES Administrator	GES Manager	GES Operator	Tenant Guest
Creating graphs	Yes	No	No	No
Deleting graphs	Yes	No	No	No
Querying graphs	Yes	Yes	Yes	Yes
Accessing graphs	Yes	Yes	Yes	No

Operation	GES Administrator	GES Manager	GES Operator	Tenant Guest
Importing data	Yes	Yes	No	No
Creating metadata files	Yes	Yes	No	No
Checking metadata files	Yes	Yes	Yes	Yes
Copying metadata files	Yes	Yes	No	No
Editing metadata files	Yes	Yes	No	No
Deleting metadata files	Yes	Yes	No	No
Clearing data	Yes	Yes	No	No
Backing up graphs	Yes	Yes	No	No
Restoring graphs from backups	Yes	Yes	No	No
Deleting backups	Yes	Yes	No	No
Querying backups	Yes	Yes	Yes	Yes
Starting graphs	Yes	Yes	No	No
Stopping graphs	Yes	Yes	No	No
Upgrading graphs	Yes	Yes	No	No
Exporting graphs	Yes	Yes	No	No
Binding EIPs	Yes	Yes	No	No
Unbinding an EIP	Yes	Yes	No	No

Operation	GES Administrator	GES Manager	GES Operator	Tenant Guest
Checking results in the task center	Yes	Yes	Yes	Yes
Resizing a graph	\checkmark	No	No	×
Expanding a graph	\checkmark	No	No	×
Restarting a graph	\checkmark	Yes	No	×
Configuring fine-grained permissions	\checkmark	Yes	No	×
Configuring user groups	\checkmark	Yes	No	×
Importing IAM users	\checkmark	Yes	No	×
Checking user details	\checkmark	Yes	Yes	\checkmark

Table 3-7 Common GES operations supported by each OBS policy

GES Operation	Dependent OBS Permission
Viewing metadata	OBS Viewer policy or OBS Buckets Viewer role
Creating, importing, copying, editing, and deleting metadata	OBS Operator policy or Tenant Administrator role
Creating, importing, and exporting graphs	OBS Operator policy or Tenant Administrator role

Table 3-8 Common	GES	operations	supported	by each	IAM policy
				2	

GES Operation	Dependent IAM Permission
Importing IAM users	iam:users:listUsers (custom policy), IAM ReadOnlyAccess (system policy), or Server Administrator role

GES Operation	Dependent IAM Permission
Creating or editing a user group	iam:users:listUsers (custom policy), IAM ReadOnlyAccess (system policy), or Server Administrator role

3.4 Reducing Extensive Permissions of Cloud Service Agencies

In versions earlier than GES 2.4.6, an agency can be used in the following scenarios:

Table 3-9 Agency scenarios

Agency	Permission	Description
get_agency or	Server	Allows GES to call your VPC service. For
ges_agency_d	Administrator	example, in the event of a failover, GES uses
efault_ <i>{Regio</i>	or XX	this agency to bind your EIP to the primary
n ID}	FullAccess	GES load balancing instance.

Due to the limitations of IAM 1.0, which only had RBAC authorization, the agency permissions for these two scenarios were relatively large. In reality, GES did not require such extensive permissions.

To reduce agency permissions, GES provides a one-click reduction function on the console, which helps you easily remove unnecessary permissions delegated to GES.

Procedure

- 1. Log in to the GES management console. In the navigation pane on the left, choose **Graph Management**.
- 2. If excessive agency permissions are not reduced, you will see a notification to reduce agency permissions at the top of the console.
- 3. Click **Rectify**. The **Reduce Agency Permission** dialog box is displayed. See **Figure 3-2**.

Note: This dialog box will remind you that when using GES, some scenarios require an agency to authorize GES to access user resources. The system will create a custom policy called **ges_access_vpc_custom** and authorize it to **ges_agency**. It will also list high-risk agency permissions that need to be removed to enhance account security.

Cancel

ок

 \times

Figure 3-2 Reducing agency permissions

Reduce Agency Permission

When using GES, certain scenarios require an agency to grant GES access to user resources. The system will create a custom policy named ges_access_vpc_custom and grant authorization to ges_agency. Do you authorize this? Agency Custom Policy Action Description vpc:publiclps:update ges_agency ges_access_vpc_custom VPC Agency Permission Requir. vpc:publiclps:get The following high-risk agency permissions will also be revoked to enhance account security: Agency Permission Description GESAccessKMS KMS Administrator KMS Administrator Server Administrator Server Administrator ges_agency VPC FullAccess All permissions of VPC service. ges_agency test AAD FullAccess Full permissions for Advanced Anti-DDoS. To confirm deletion, enter "DELETE" below. Auto Enter DELETE

 Manually enter **DELETE** or click **Auto Enter** to reduce agency permissions. Once the operation is successful, the notification to reduce agency permissions will automatically disappear from the dialog box.

NOTE

If you do not have the permission to query agency permissions, the system cannot retrieve agency information using your authentication credentials. A notification to fix agency permissions will appear every time you log in to the console, urging you to inform the administrator to resolve the issue. You can also close the notification or select **Do not show again**.

4 Metadata Operations

4.1 Graph Data Formats

4.1.1 Static Graph

Before importing graph data, familiarize yourself with the graph data formats supported by GES.

- GES only supports the loading of raw graph data in the standard CSV format. If your raw data is not in this format, convert it to CSV.
- GES graph data consists of the vertex, edge, and metadata files.
 - Vertex files store vertex data.
 - Edge files store edge data.
 - Metadata is used to describe the formats of data in vertex and edge files.

Concept Description

Graph data is imported through a property graph model in GES, so you must learn the concept of the property graph.

A property graph is a directed graph consisting of vertices, edges, labels, and properties.

- A vertex is also called a node, and an edge is also called a relationship. Nodes and relationships are the most important entities.
- Metadata describes vertex and edge properties. It contains multiple labels, and each label consists of one or more properties.
- Vertices with the same label belong to a group or a set.
- Each vertex or edge can have only one label.

Metadata

The following figure shows the metadata structure.



Figure 4-1 Metadata structure

GES metadata is stored in an XML file and is used to define vertex and edge properties.

It contains labels and properties.

• Label

A label is a collection of properties. It describes formats of property data contained within a vertex or an edge.

NOTE

If the same property **name** is defined in different labels, the **cardinality** and **dataType** of the properties in different labels must be the same. Starting from version 2.3.18, this restriction no longer exists, meaning that properties with the same name under different labels can have different types.

• Property

A property refers to the data format of a single property and contains three fields.

 Property name: Enter 1 to 256 characters. Special characters (<>& and ASCII codes 14, 15, and 30) are not allowed.

NOTE

A label cannot contain two properties with the same name.

 cardinality: Indicates the composite type of data. Possible values are single, list, and set. single indicates that the data of this property has a single value, such as a digit or a character string.

NOTE

If **value1;value2** is of the **single** type, it is regarded as a single value.

- list and set indicate that data of this property consists of multiple values separated by semicolons (;).
 - **list**: The values are placed in sequence and can be repeated. For example, **1;1;1** contains three values.
 - **set**: The values are in random sequence and must be unique. Duplicate values will be overwritten. For example, **1;1;1** contains only one value (1).

NOTE

list and set do not support values of the char array data type.

- **dataType**: Indicates the data type of the property values. The following table lists the data types supported by GES.

 Table 4-1
 Supported
 data types

Туре	Description
char	Character
char array	Fixed-length string. Set the maximum length using the maxDataSize parameter.
	NOTE
	 You can set maxDataSize to limit the maximum length of the string. For details, see Metadata structure.
	Only single supports the data type.
	 If the property data is a string, you are advised to set dataType to char array. If the data type is set to string, the import is slower.
float	Float type (32-bit float)
double	Double floating point type (64-bit float point)
bool	Boolean type. Available values are 0/1 and true/false .
long	Long integer (value range: -2^63 to 2^63-1)
int	Integer (value range: -2^31 to 2^31-1)
date	Date. Currently, the following formats are supported:
	• YYYY-MM-DD HH:MM:SS
	• YYYY-MM-DD
	NOTE The value of MM or DD must consist of two digits. If the day or month number contains only one digit, add 0 before it, for example, 05/01.

Туре	Description
enum	Enumeration. Specify the number of the enumerated values and the name of each value. For details, see Metadata structure .
string	Variable-length string
	NOTE The data import efficiency can be very low if the string is too long. You are advised to use a char array instead.
	You can set the length of a char array as needed. It is recommended that the length be less than or equal to 32 characters.

The following figure shows a metadata example:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<PMML version="3.0"
 xmlns="http://www.dmg.org/PMML-3-0"
 xmlns:xsi="http://www.w3.org/2001/XMLSchema_instance" >
 <labels>
  <label name="default">
  </label>
  <label name="movie">
     <properties>
       <property name="ChineseTitle" cardinality="single" dataType="int" />
       <property name="Year" cardinality="single" dataType="string"/>
       <property name="Genres" cardinality="single" dataType="string"/>
     </properties>
  </label>
  <label name="user">
     <properties>
       <property name="ChineseName" cardinality="single" dataType="int" />
       <property name="Gender" cardinality="single" dataType="string"/>
       <property name="age" cardinality="single" dataType="enum" typeNameCount="7"</pre>
       typeName1="Under 18" typeName2="18-24" typeName3="25-34" typeName4="35-44"
typeName5="45-49'
        typeName6="50-55" typeName7="56+"/>
       roperty name="occupation" cardinality="single" dataType="enum" typeNameCount="21"
       typeName1="other or not specified" typeName2="academic/educator" typeName3="artist"
typeName4="clerical/admin" typeName5="college/grad student"
        typeName6="customer service" typeName7="doctor/health care" typeName8="executive/
managerial" typeName9="farmer" typeName10="homemaker'
        typeName11="K-12 student" typeName12="lawyer" typeName13="programmer"
typeName14="retired" typeName15="sales/marketing"
         typeName16="scientist" typeName17="self-employed" typeName18="technician/engineer"
typeName19="tradesman/craftsman" typeName20="unemployed"
         typeName21="writer"/>
       <property name="Zip-code" cardinality="single" dataType="char array" maxDataSize="12"/>
     </properties>
  </label>
  <label name="rate">
     <properties>
       <property name="Rating" cardinality="single" dataType="int" />
       <property name="Datetime" cardinality="single" dataType="string"/>
     </properties>
  </label>
</labels>
</PMML>
```

Vertex Files

A vertex file contains the data of each vertex. A vertex of data is generated for each behavior. The following is an example. **id** is the unique identifier of a set of vertex data.

id, label, property 1, property 2, property 3, ...

NOTE

- Name of the vertex ID. You are advised not to use hyphens (-) as it may impact Gremlin queries.
- You do not need to set the data type of the vertex ID. It is of the string type by default.
- Do not add spaces before or after a label. Use commas (,) to separate information. If a space is identified as a part of a label, the label may fail to be identified. In this case, the system may display a message indicating that the label does not exist.

Example:

Vivian, user, Vivian, F, 25-34, artist, 98133 Eric, user, Eric, M, 18-24, college/grad student, 40205

Edge Files

An edge file contains the data of each edge. An edge of data is generated for each behavior. The graph size in GES is defined by the quantity level of the edges, for example, one million edges. The following is an example. **id 1** and **id 2** are the IDs of the two endpoints (vertices) of an edge.

id 1, id 2, label, property 1, property 2, ...

Example:

Eric,Lethal Weapon,rate,4,2000-11-21 15:33:18 Vivian,Eric,friends

Note: To store edges with the same vertices and labels in a database edition graph, you need to include a sortKey column. This column should be placed after the property column, which should be the last column.

When importing, specify the **sortKey** parameter. If **sortKey** has a value, it will be correctly read based on the graph's sortKey type. If there is no value, add a comma at the end of the property. This will import an empty value, which will set **sortKey** to **NULL**.

id 1, id 2, label, property 1, property 2, ...,sortKey

Example:

Eric,Lethal Weapon,rate,4,2000-11-21 15:33:18, 5 Vivian,Eric,friends,

4.1.2 Dynamic Graph

In most real-life problems, entities and relationships change over time (such as disease transmission networks and transaction networks). The time sequence and changing information greatly affect the results. To predict these results, we use dynamic graphs to model, store, and analyze the dynamic data.





D NOTE

This section mainly describes the data format of dynamic graphs. For details about operations related to these graphs, see **Creating Dynamic Graphs** and **Using Dynamic Graphs**.

Data Model

A general property graph is a directed graph consisting of vertices, edges, labels, and properties.

A dynamic graph evolves over time. From static to dynamic, there are spatiotemporal graphs (STGs), discrete-time dynamic graphs (DTDGs), and continuoustime dynamic graphs (CTDGs) as shown in dynamic graphs. CTDGs are dynamic graphs that store more details about the vertices and edges.

GES allows you to create CTDGs. The following is an example.

Assume that a graph has three vertices: **Vivian**, **P1**, and **P2**, and three edges: (**Vivian**, **P1**), (**Vivian**, **P2**), and (**Vivian**, **Vivian**). There are two vertex types (lables): **Person** and **Place**, and relationship types (label): **Visited** and **Diagnosed**. The timestamp **[startTime, endTime]** indicates the duration of an event. For example, (**Vivian**, **P1**) indicates that Vivian visited P1 during 2021-11-21 12:05:21 to 2021-11-21 14:00:00, and (**Vivian**, **Vivian**) indicates that Vivian was infected with COVID-19 during 2021-11-25 23:00:00 to 2021-12-04 08:00:00. (Note: The vertex state changed, and the edge that starts and ends with this vertex represents this event.)





Metadata of Dynamic Graphs

Timestamps are important features of dynamic graphs. To describe dynamic graph data, you need to define timestamp-related properties such as **startTime** and **endTime** in metadata.

Note that **startTime** and **endTime** dynamic graph properties and are related to the life cycle of vertices and edges in the graph. The type must be **date** or **long**. The following is an example:

<pmml></pmml>
<labels></labels>
<label name="Person"></label>
<properties></properties>
<property cardinality="single" datatype="long" name="startTime"></property>
<property cardinality="single" datatype="long" name="endTime"></property>
<property cardinality="single" datatype="string" name="name"></property>
<property cardinality="single" datatype="int" name="age"></property>
<property cardinality="single" datatype="string" name="gender"></property>
<label name="Place"></label>
<properties></properties>
<property cardinality="single" datatype="string" name="type"></property>
<property cardinality="single" datatype="string" name="address"></property>
<property cardinality="single" datatype="float" name="longitude"></property>
<property cardinality="single" datatype="float" name="latitude"></property>
<label name="Visited"></label>
<properties></properties>
<property cardinality="single" datatype="long" name="startTime"></property>
<property cardinality="single" datatype="long" name="endTime"></property>
<label name="Dignosed"></label>
<properties></properties>
<property cardinality="single" datatype="long" name="startTime"></property>
<property cardinality="single" datatype="long" name="endTime"></property>
<property cardinality="single" datatype="string" name="risk"></property>

Vertices of Dynamic Graphs

• Dynamic vertex

For dynamic graphs, each line of the vertex file contains the data of a vertex. **id** uniquely identifies vertex data, **startTime** indicates the start time of the vertex lifecycle, and **endTime** indicates the end time of the vertex lifecycle.

id,label,startTime,endTime,property1,property2...

The following is an example:

Vivian, Person, 1991-02-03 08:00:00, 9999-12-31 24:00:00, Vivian, F, 25-34

• Static vertex

A vertex without specified **startTime** and **endTime** is a static vertex.

id,label,property1,property2...

The following is an example for static vertex:

Vivian, Person, Vivian, F, 25-34

P1,Place,residentialArea,xxxxxx,114.001494,22.554249

P2,Place,publicArea,xxxxx,114.074367,22.53492

Note

If a vertex changes over time in its lifecycle, for example, the health status information of a person changes in a certain period, the changes can be modeled as an edge. The edge data is stored in a line of the edge file, representing status changes of the vertex.

id,id,label,startTime,endTime,property...

The following is an example:

Vivian, Vivian, Diagnosed, 2021-11-25 23:00:00, 2021-12-04 08:00:00, Covid-9

Edges of Dynamic Graphs

• Dynamic edge

The following example shows the data of an edge in a dynamic graph. Each line in the edge file contains the data of an edge. **id 1** and **id 2** indicate the IDs of the start and end vertices of an edge, respectively. **startTime** indicates the start time of the edge lifecycle, and **endTime** indicates the end time of the edge lifecycle.

id 1, id 2, label, startTime, endTime, property 1, property 2, ...

The following is an example:

Vivian,P1,Visited,2021-11-21 12:05:21,2021-11-21 14:00:00

Vivian, P2, Visited, 2021-11-21 16:33:18, 2021-11-21 19:51:00

Static edge

An edge without the start time and end time is a static edge.

id 1, id 2, label, property 1, property 2, ...

Vertex and Edge Data File

Vertex data file

Each line in the file indicates a dynamic/static vertex. You can use more than one vertex file.

Vivian, Person, Vivian, F, 25-34

P1,Place,residentialArea,xxxxx,114.001494,22.554249

P2,Place,publicArea,xxxxx,114.074367,22.53492

• Edge data file

Each line in the file indicates a dynamic/static edge. You can use more than one edge file.

Vivian,P1,Visited,2021-11-21 12:05:21,2021-11-21 14:00:00 Vivian,P2,Visited,2021-11-21 16:33:18,2021-11-21 19:51:00

4.2 Importing a Metadata File

4.2.1 Preparing Metadata

Preparing Metadata on a Local PC

You need to prepare a metadata file on your PC and import the file to GES for subsequent graph analysis.

The metadata files you want to import must meet the following requirements:

- 1. A maximum of 50 metadata files can be imported.
- 2. The metadata files must be in XML format.

(Optional) Importing Metadata to OBS

You can upload a prepared metadata file to an OBS bucket to import it to GES.

The procedure is as follows:

- 1. Log in to the OBS console and create an OBS bucket. If you already have a bucket, ensure that the OBS bucket and GES are in the same region. For how to create a bucket and upload files, see **Using OBS Console**.
- 2. Upload the prepared file to the OBS bucket by referring to **Uploading a File**. The metadata file must be in XML format.

4.2.2 Importing Data From a Local Path or OBS

- 1. On the GES management console, click **Metadata Management** in the navigation tree on the left.
- 2. On the **Metadata Management** page, click **Import** in the upper left corner.
- 3. In the **Import** dialog box, select **Local** or **OBS** for **Type** to import a metadata file form a local path or OBS.
 - Import a metadata file from a local path.

Select Local File: Click Upload to select the metadata file.

NOTE

The file must be in the XML format.

Name: Enter a name for the metadata.

Storage Path: Select an OBS path for storing the metadata file.

Figure 4-4 Importing metadata from a local path

Import					×
★ Туре	Local	OBS			
* Select File				Upload	
* Name					
* Storage Path				Ð	
			Cancel	Ок	

- Import a metadata file from OBS.

Select File Path: Select the metadata file from OBS.

D NOTE

- The file must be in the XML format.
- Ensure that you have uploaded the metadata file to your OBS bucket.

Name: Enter a name for the metadata.

Figure 4-5 Importing metadata from OBS

Import				×
* Туре	Local	OBS		
* Select File Path				
* Name				
			Cancel	ОК

4. Click **OK** to import the metadata.

If the import is successful, the metadata file is displayed on the **Metadata Management** page.

4.3 Creating a Metadata File

If you currently have no metadata file, you can create metadata files on GES.

NOTE

A maximum of 50 metadata files can be created.

Procedure

- 1. On the **Metadata Management** page, click **Create Metadata File** in the upper right corner.
- 2. Configure the following parameters on the displayed page:
 - Name: Enter the metadata file name. The default file format is XML.
 - Storage Path: Select an OBS path for storing the metadata file. If you create metadata for the first time, you need to enable OBS. (You are advised to obtain user authorization and automatically create OBS buckets for the metadata.)
 - Definition: Metadata models can be built manually or in a visualized manner.

Manual: Click **Add Label**. Define the label name and label type. Click **Add** under the label name to add a property. You can also click **Up** or **Down** to sort properties. **Table 4-2** lists the property parameters. For details about other metadata information, see **Graph Data Formats**.

D NOTE

- 1. Multiple labels are allowed. Click **Add label** to add labels as needed.
- 2. There are three types of labels: vertex, edge, and general-purpose (both vertex and edge).

Figure 4-6 Manual

* Name	schema_3c14				
· Storage Path	8				
Definition (3)	Manual Visual				
	▲ Label Name	Type All v		Remove	
	Add Up Down				
	Property Name	Cardinality	Data Type		Operation
	C Input a property.	S	char v		Remove
	+ Add label				
	CIC Carol				

Visual:

- Adding a vertex label: Drag a circle to the canvas to add a vertex. Click the vertex in the canvas to define its name, description, and properties.
- Adding an edge label: Click a connection point on a vertex and drag it to the connection point of another vertex to create an edge. Define its name, description, source vertex, target vertex, and properties. Table 4-2 lists the property parameters.



Metadata Managemen	/ Create Metadata File								
* Name * Storage Path Definition ①	schema_524f Manusal Vozual	ß							
			Des a creb b becase to a de a veloc	•		•	Letel Name Type Descaling Descaling Securit Veter Type Type Type Typenty	Etter a blott.	+Add
	OK Cancel								

Table 4-2 Property parameters

Name	Description
Property Name	Property name. Enter 1 to 256 characters. Special characters (<>& and ASCII codes 14, 15, and 30) are not allowed.

Name	Description
Cardinalit	Composite type of data
У	• Single value : indicates that the property has a single value, such as a digit or a string.
	• Multiple values : indicates that the property has multiple values separated by semicolons (;). You can determine whether to allow repetitive values.
Data Type	Data type of the property values. Available values are char, float, double, bool, long, int, date, enum, string, and char array. For details, see Static Graph. NOTE
	only the single-value property supports the char array type.
Operation	Click Remove to delete a property.

3. Click **OK**. The created metadata file will be displayed on the **Metadata Management** page.

On the **Metadata Management** page, you can view the storage path, status, and modification time of the metadata.

4.4 Copying a Metadata File

If you edit a metadata file, the original metadata file will be overwritten. To avoid loss of the original metadata, you can sabe a copy of the file before editing it.

Procedure

- 1. GES provides two methods for you to copy a metadata file on the **Data Management** page.
 - Click the metadata file name. On the details page, click Copy.
 - Click **Copy** in the **Operation** column of the target metadata file.
- 2. Specify the metadata file name and storage path.

Name: Enter the name of the copied metadata file. The default file format is XML.

Storage Path: Enter an OBS path for storing the metadata file.

Figure 4-8 Copying a metadata file

Metadata Management	/ Сору	
★ Name		
★ Storage Path		Ð
	OK Cancel	
3. Click **OK**.

The copy of the metadata file will be displayed on the **Metadata Management** page.

4.5 Editing a Metadata File

If the metadata file you imported or created needs to be modified, you can directly modify its labels and properties.

NOTE

After the metadata file is edited, the original metadata file will be overwritten. To avoid data loss, you are advised to save a copy of the metadata file before editing it.

Procedure

- 1. GES provides two methods for you to edit a metadata file on the **Data Management** page.
 - Click the metadata file name. On the metadata details page, click Edit.
 - Click Edit in the Operation column of the target metadata file.

Figure 4-9 Clicking Edit

Metadata Management 🛞				Create Metado	ata File
You can create 14 more metadata files. Import Q: Select a property or enter a keyword.					
Name 🖯	Storage Path 🖯	Status 🖯	Last Modified 😜	Operation	
schema_Vectorindex	r schema.xml	Normal	Dec 25, 2024 11:10:31 GMT+08:00	Edit Copy Delete	

- 2. On the editing page:
 - On the **Manual** tab, you can add labels and properties, change label names, and sort properties by clicking **Up** and **Down**.
 - On the **Visual** tab, you can drag a vertex to the canvas to add a label, or click a vertex or edge to modify the label information.
- 3. After the modification is complete, click **OK**.

4.6 Searching for a Metadata File

On the **Metadata Management** page, enter the name of the metadata file you want to search.

Figure 4-10 Searching for a metadata file

Meladata Management 💿	Creale Metadala File
You can create 400 more invaluate thes.	
C. Select a property or enter a keywand.	() ()

4.7 Deleting a Metadata File

If a metadata file becomes invalid, locate it in the metadata file list on the **Metadata Management** page, click **More** in its **Operation** column, and select **Delete**.

NOTE

Deleted data cannot be recovered. Exercise caution when performing this operation.

Figure 4-11 Deleting a metadata file

Metadata Management 🛞				Create Metadata File
You can create 14 more metadata files.				
Q. Select a property or enter a keyword.				Q ()
Name O	Storage Path 😑	Status 0	Last Modified 🖗	Operation
schema_Vectorindex	schema ami	Normal	Dec 25, 2024 11:10:31 GMT+08:00	Edit Copy Delete

5 Creating Graphs

5.1 Methods to Create a Graph

The following content describes how to create a graph on GES console.

You can create a graph **using an industry-specific template** or **without any template**, or you can create a **dynamic graph**. No template is selected by default.

- Custom graph: This is a default graph creation method that fully meets your requirements.
- With an industry template: You can select a template you want, specify graph specifications, and add data to the template to create a graph.
- Dynamic graph: By default, the **dynamic graph analysis function** is enabled for graphs created in this mode.

NOTE

You must create a dynamic graph to use the function. This function cannot be enabled for custom graphs and template-based graphs.

5.2 Creating a Graph Without Using a Template

- 1. Log in to the GES console and click **Create Graph** in the upper right corner of the **Overview** page.
- 2. Select the **Region** where the cluster works from the drop-down list in the upper left corner of the page.
- 3. On the **Create Graph** page, click the **Customize Graph** tab and set the following parameters:
 - a. In the **Configure** step, set the graph name and software version.

Figure 5-1 Graph name and software version

★ Graph Name ⑦	ges_a53e
* GES Software Version	2.4.0

Parameter	Description	
Graph Name	You can set a name or use the default name. After a graph is created, its name cannot be changed.	
	The graph name must:	
	• Contain 4 to 50 characters and start with a letter.	
	Be case-insensitive.	
	• Contain only letters, digits, and underscores (_).	
GES Software Version	The system uses the latest version by default, and only the default version is available.	

b. Set network parameters, including VPC, Subnet, Security Group, Enterprise Project, and Public Network Access.



* VPC 🕜	vpc-2a53	✓ C View VPC
* Subnet ③	subnet-2a61 (192.168.0.0/24)	~
* Security Group ②	Learn how to configure a security group.	
	dws-ges_etl-8000	View Security Group
* Public Network Access	Do not use Buy now S	Specify
	A graph instance without an EIP cannot be access	sed over the Internet. However, the graph instance can be accessed through ECSs deployed on a private network.
Tag	TMS's predefined tag function is recommended for	or adding the same tag to different cloud resources.View Predefined Tags
	To add a tag, enter a tag key and a tag value belo	W.
	Enter a tag kay	
	Tans you can still add: 20	
	rugo you can sui uuu. 20	
Data Encryption	Data Encryption	
	Access to a graph instance will be encrypted. Ena	abling data encryption will affect the performance.
Cryptographic Algorithm 🕐	General encryption algorit 🗸	

Parameter	Description
VPC	A VPC is a secure, isolated, and logical network environment.
	Select the VPC for which you want to create the graph and click View VPC to view the name and ID of the VPC.
	NOTE If your account has VPCs, a VPC will be automatically selected. You can change it as needed. If no VPC is available, you need to create a VPC. After the VPC is created, it will be automatically selected.
Subnet	A subnet provides dedicated network resources that are logically isolated from other networks for network security.
	Select the subnet for which you want to create the graph to enter the VPC and view the name and ID of the subnet.
Security Group	A security group implements access control for ECSs that have the same security protection requirements in a VPC.
	• Click Learn how to configure a security group. to get instructions.
	 Click View Security Group to learn security group details.
Public Network Access	The public network access to the graph. Set this parameter as you need.
	Do not use : A graph instance without an elastic IP (EIP) cannot be accessed over the Internet. However, the graph instance can be accessed through ECSs deployed on a private network.
	Buy now : GES automatically allocates an EIP with exclusive bandwidth to the graph instance so that the graph instance can be accessed over the Internet using the EIP. In addition, GES uses the tenant permission to automatically create an agency with the prefix of ges_agency_default in the project to support EIP binding.
	Specify : Select an EIP to allow the graph instance to be accessed over the Internet.
	Click Create EIP to access the VPC management console and create an EIP.
Enterprise Project	Centrally manages cloud resources and members by project.
	Click Create Enterprise Project to go to the Enterprise Project Management page.

D	
Parameter	Description
Tag	Tags for a resource. Enter a tag key and value, and click Add to add the tag. You can view the added tag in the graph details and
	search for graphs by tag on the Graph Management page.
	Figure 5-3 Viewing tag details
	Graph Management (*) Statutes Parts Vision Vision results or results on the statute (*) (*)
	MI ADDay Manual Instrume ADDAy Manua Instrume ADDAy Manual Instrume ADDAy
	<text></text>
Security Mode	If you enable the security mode, communications will be encrypted when you access a graph instance, and only HTTPS can be used when you call APIs. This function affects GES performance.
Cryptographic Algorithm	 Available values are as follows: General cryptographic algorithms (SM series cryptographic algorithms not supported) are used by all components to store and transmit sensitive data. These algorithms that do not have special requirements. SM series commercial encryption algorithm
	 Sive series continencial encryption algorithm (compatible with the international general algorithm) is supported. Sensitive data of all components is stored using this algorithm. The SM series commercial encryption algorithm and international algorithm can be used for data transmission.

c. Set graph parameters.

Figure 5-4 Graph parameters

Cross-AZ HA	
* Purpose	Enterprise production Developer learning
	Supports high reliability and concurrency, suitable for enterprise production and large-scale application.
* Versions	Memory edition Database edition
	Storage and compute based on distributed key-value databases. This edition has higher performance and has unlimited capacity, but it supports only the Cypher queries.
* Vertex ID Type 💮	String (variable length) v
	In string (variable length) format, the length of the vertex ID is not limited. However, if the ID is too long, the read and write performance is affected. It is recommended that the length be less than 1 KB. The maximum length is 4 KB.
SortKey Type 💿	Integer v
	Integer type, saves space
* Compute Resource	ECS
	An Elastic Cloud Server (ECS) is a virtual server that runs in a secure and isolated environment.
* CPU Architecture	X86 Kurpeng
+ Grank Size /Edgard)	Tee million edus (add will) Unedeed million edus (add will)
* Giapii Gize (Euges)	reminioreuge (sou ou) numerorinioreuge (sou ou) chichedge remonioreuge one-numeroge

Parameter	Description	
Cross-AZ HA	Whether to support cross-AZ cluster. If this function is enabled, graph instances are distributed in different AZs to enhance reliability.	
Purpose	Purpose of the graph to be created.	
	Enterprise production : High reliability and concurrency are supported, suitable for production and large-scale applications.	
	Developer learning : A complete function experience is offered, suitable for developer learning.	
Versions	GES editions.	
	• Memory edition: The capacity is limited and a maximum of 10 billion edges are supported. Storage and compute based on memory storage. This edition is preset with a variety of algorithms, and Gremlin and Cypher query languages are supported.	
	 Database edition: The storage capacity is unlimited. Storage and compute based on distributed key-value databases. This edition has higher performance and has unlimited capacity, but it supports only the Cypher queries. 	

Parameter	Description
Vertex ID Type (This parameter is available only when you choose the database edition.)	The options include String (fixed length) , String (variable length), and Hash.
	 String (fixed length): Vertex IDs are used for internal storage and compute. Specify the length limit. If the IDs are too long, the query performance can be reduced. Specify the length limit based on your dataset vertex IDs. If you cannot determine the maximum length, set the ID type to hash. String (variable length): The length of the vertex
	IDs written by the user is not limited. However, if the IDs are too long, the read and write performance is affected. It is recommended that the length be within 1,000 bytes, with a maximum of 4,000 bytes.
	• Hash: Vertex IDs are converted into hash code for storage and compute. There is no limit on the ID length. However, there is an extremely low probability, approximately 10^(-43), that the vertex IDs will conflict.
	NOTE If you cannot determine the maximum length of a vertex ID, set this parameter to Hash .
SortKey Type (This parameter is available only when you choose the database edition.)	Different SortKey values are configured to distinguish duplicate edges (edges with the same source vertex, end vertex, and label). The options include:
	 Integer: The value is an integer, which saves space.
	• String (byte length less than or equal to 40)
	• String (variable length) : The length is not limited. However, if the IDs are too long, the read and write performance is affected. It is recommended that the length be within 1,000 bytes, with a maximum of 2,000 bytes.
Compute	Type of compute resources.
Kesource	An elastic cloud server (ECS) is a computer system that has complete hardware, an operating system (OS), and network functions and runs in a secure, isolated environment.
CPU Architecture	Currently, GES supports X86 and Kunpeng.

Parameter	Description
Graph Size	Available options based on your resource quota.
(Edges)	Different graph specifications are displayed for Enterprise production and Developer learning .
	 Development learning: Currently, there is only Ten-thousand-edge graphs are available for this purpose, regardless of the edition.
	 Enterprise production: The specifications vary depending on the edition.
	 Memory edition: The options are Million- edge, Ten-million-edge, Hundred-million- edge, Billion-edge, Billion-edge-pro, and Ten- billion-edge.
	 Database edition: The options are Billion- edge, Ten-billion-edge, and Hundred-billion- edge.
	NOTE Graph size, which is based on the number of edges. The value is not accurate. If there are a large number of vertices and properties, you are advised to apply for graphs with a larger size.

- d. Advanced Settings: Set this parameter to Default or Custom.
 - **Default**: Use the default values.
 - Custom:
 - When you set Versions to Memory edition, you need to set the following custom parameters: Fine-Grained Permission and Multiple labels.

Figure 5-5 Advanced settings for the memory edition



Parameter	Description
Fine-Grained Permission	Whether to enable fine-grained permission management. If this function is enabled, the traverse, read, and write permissions can be set for specific properties of a label.

Parameter	Description
Multiple labels	After this option is enabled, multiple labels can be set for the same vertex in the graph.
	NOTE
	1. Only the memory edition supports this function.
	2. Each label corresponds to a unique property. When the API for querying vertex details is called, information about all labels and corresponding properties on the vertex is returned. Property filtering queries filter different labels on the vertex.

• If you choose the database edition, you can enable or disable **HyG computing engine** and **Fine-Grained Permission**.

Figure 5-6 Advanced settings for the database edition



Parameter	Description
HyG computing engine	HyG is a high-performance distributed graph computing framework that supports many graph analysis algorithms. HyG engine is suitable for complex graph analysis.
Fine-Grained Permission	Whether to enable fine-grained permission management. If this function is enabled, the traverse, read, and write permissions can be set for specific properties of a label.

- 4. Click Next. The Confirm page is displayed.
- 5. Confirm the information and click **Submit** to create the graph.
- 6. After the submission is successful, the **Finish** tab page is displayed. You can click **Back to Task Center** to view the status and running result of the created graph.

5.3 Creating a Graph Using an Industry-Specific Template

- 1. Log in to the GES console and click **Create Graph** in the upper right corner of the **Overview** page.
- 2. Select the **Region** where a graph works from the drop-down list in the upper left corner of the page.
- 3. On the **Create Graph** page, click the **Use Industry-Specific Graph Template** tab and configure the following parameters:

In the **Configure** step, select a template and configure network and graph information:

 Select the desired template. Currently, Asset Management Graph Template and Power Distribution Management Template are available.

Figure 5-7 Selecting the template



- b. Set network information. Configure related parameters by referring to **Creating a Graph Without Using a Template**.
- 4. Click **Next**. On the **Confirm** page, confirm the specifications and click **Submit**. The system automatically creates the graph of the selected specifications and inserts the selected template data (schema and sample data).
- 5. After the submission is successful, the **Finish** tab page is displayed. You can click **Back to Task Center** to view the status and running result of the created graph.

NOTE

- You do not need to set the name for a graph created using a template. By default, the name of the template is used as the prefix of the created graph, for example, **assets_management**.
- After the graph is created, the name of the created graph is in **assets_management_***XXXX* format, where *XXXX* is the unique identifier automatically generated by the system and cannot be modified.

5.4 Creating a Dynamic Graph

- 1. Log in to the GES console and click **Create Graph** in the upper right corner of the **Overview** page.
- 2. On the displayed page, click the **Create Dynamic Graph** tab. The page for creating a dynamic graph is displayed.

Figure 5-8 Page for creating a dynamic graph



3. Set required parameters by referring to **Creating a Graph Without Using a Template**.

By default, the **Dynamic graph analysis capability** is enabled for dynamic graphs.

Temporal Graph Analysis

- 4. Click **next**. On the **Confirm** page that is displayed, confirm the information and click **Submit** to create the graph.
- 5. After the submission is successful, the **Finish** tab page is displayed. You can click **Back to Task Center** to view the status and running result of the created graph.
- 6. For details about how to use dynamic graphs, see **Dynamic Graphs**.

5.5 Starting a Graph

Scenario

You can start graphs in **Stopped** status in the graph list so that they can be accessed and analyzed again.

Graphs in **Running** status cannot be started.

Procedure

- **Step 1** Log in to the GES management console.
- **Step 2** In the navigation tree on the left, select **Graph Management**.
- **Step 3** Locate the target graph in the graph list and choose **More** > **Start** in the **Operation** column.
 - If the graph to be started has backups, a dialog box is displayed indicating that you can select either of the following methods to start the graph:
 - **Restore Last Graph**: Restart the graph that stopped running.

- **Start Backup**: Start the graph using the backup data.

After selecting a startup method, click **Yes**. The graph status becomes **Preparing** and the progress is displayed.

- If the graph to be started does not have backups, the graph status changes to **Preparing** and the progress is displayed after you click **Start**.
- **Step 4** After the graph is started, the status changes from **Preparing** to **Starting**. Wait several minutes. When the startup is successful, the graph status is switched to **Running**.

NOTE

If the startup fails, try again later. If the failure persists, fill in and submit a service ticket to contact the technical support.

----End

5.6 Stopping a Graph

Scenario

If you do not need to use a graph, you can stop it. After the graph is stopped, you cannot access it.

NOTE

- Stopping a graph does not release resources.
- After seven days of stopping a graph, the system will automatically restart the graph database instance to ensure it can keep up with the maintenance updates provided by the service.

Procedure

- **Step 1** Log in to the GES management console.
- Step 2 In the navigation tree on the left, select Graph Management.
- **Step 3** Locate the target graph in the graph list and choose **More** > **Stop** in the **Operation** column.

Figure 5-9 Stopping a graph

Gri	aph Management 💮 😋 0 Creation f	alled				•	reate Graph
	You can create 17 more graphs using 60	00 billion-edge quota.					
	 Select a property or enter a keywork 	rd.					
	Name.ID 😑	Running Status ()	Internal Access Address	External Access Addr Θ	Billing Mode Θ	Created Θ	Operation
	ges_showcase ce15ca46-8c18-4f8e-8301-761	Pad952cca D Running	192 168 0.222		Pay-per-use		Access Back
	ges_migtest 822#0#46-34b6-4295-bc6b-12	e1b7022eb0 Ø Running	192.168.0.125	100.95.153.38	Pay-per-use		Start
	0es_b00420760 8403c5e7-77b4-4e51-9b69-b0	5268e25f2c O Running	192.168.0.2	100.93.11.110	Pay-per-use		Stop Delete

Step 4 The graph status changes to **Stopping**. Wait several minutes. When the graph is successfully stopped, the graph status is switched to **Stopped**.

----End

5.7 Accessing Graphs

Scenario

On the **Graph Management** page, you can click **Access** to query and analyze a created graph.

Procedure

On the **Graph Management** page, view all created graphs and click **Access** in the **Operation** column of a target graph.

Figure 5-10 Accessing a graph

Gra	ph M	lanagement () O Creation failed					C	NAR Graph Pro	apay for Disc	ounts
	cou car	n create 16 more graphs using 6000 billion edge quot elect a property or enter a keyword.	u.						0	۲
		NameID 0	Running Status ()	Internal Access Address	External Access Addr	Billing Node 🖯	Created 0	Operation		
	۲	dynamic_graph_2h1 39382c38-383a-4b10-88a1-8271448a837d	O Running	192.168.0.186		Pay-par-use	Mar 12, 2024 20:41:48 GMT+08:08	Access Back Up Mo	n -	

5.8 Importing Incremental Data

Scenario

After you create a graph, you need to import graph data. If you need to add new graph data, you can import data to the graph.

NOTE

- Currently, only graphs of version 1.1.8 and later support this function.
- To prevent failures in restoring the imported graph data during system restart, do not delete the data stored on OBS when the graph is in use.
- The default separator of data columns is comma (,). You cannot define a separator.
- The size of a single file in the import directory or the size of a single file to be imported cannot exceed 5 GB. Or the import will fail. You are advised to split the file into multiple files smaller than 5 GB before importing.
- The total size of files imported at once (including vertex and edge datasets) cannot exceed 1/5 of the available memory. For details about the available memory, check the **Node Monitoring** area on the **O&M monitoring dashboard** for the minimum value of available memory for nodes with the suffix **ges-dn-1-1** and **ges-dn-2-1** (hover over the memory usage rate).

Procedure

- **Step 1** Log in to the GES management console. In the navigation pane on the left, choose **Graph Management**.
- **Step 2** In the graph list, locate the target graph, click **More** in the **Operation** column, and select **Import**.

Import		×
Metadata	Q Create Download	
Edge Data		Download
Vertex Data		Download
Log Storage Path ⊘		
* Edge Processing 🕜	Allow repetitive edges ⑦	
	O Ignore subsequent repetitive edges 💿	
	Overwrite previous repetitive edges	
	✓ Ignore labels on repetitive edges ⑦	
Import Type	Online import The import speed is slower, but the graph can be read (cannot be written).	
	Offline import The import speed is higher, but the graph cannot be read or written.	
	Cancel	ОК

Figure 5-11 Importing data

Step 3 In the displayed **Import** dialog box, set the following parameters:

- Graph Cluster (only available for database edition graphs): When a database edition graph is created, it is automatically upgraded to a multi-graph cluster. Such a cluster can contain multiple graph instances. For details, see Multi-Graph Management (Database Edition).
- Metadata: Select an existing metadata file or create one. For details, see Creating a Metadata File.
- Edge Data: Select the corresponding edge data set.
- Vertex Data: Select the corresponding vertex data set. If you leave it blank, the vertices in the Edge Data set are used as the source of Vertex Data.
- Log Storage Path: Stores vertex and edge data sets that do not comply with the metadata definition, as well as detailed logs generated during graph import. Storage on OBS may incur fees, so delete the data in time.
- **SortKey Included in Edge File** (only available for database edition graphs): Different SortKey values can be configured to distinguish duplicate edges (edges with the same source vertex, end vertex, and label).
- Edge Processing: Includes Allow repetitive edges, Ignore subsequent repetitive edges, Overwrite previous repetitive edges, and Ignore labels on repetitive edges.

Edge Processing: Repetitive edges have the same source and target vertices. When labels are considered, repetitive edges must have the same source and target vertices and the same labels.

- Allow repetitive edges: Multiple edges may exist between a source vertex and a target vertex.
- Ignore subsequent repetitive edges: If there are multiple edges between a source vertex and a target vertex, only the first edge read is retained.

- Overwrite previous repetitive edges: If there are multiple edges between a source vertex and a target vertex, only the last edge read is retained.
- Ignore labels on repetitive edges: If labels are ignored, edges with the same source vertex and target vertex are repetitive edges.
- Import Type: The value can be Online import or Offline import.

NOTE

- Graphs of the database edition support **multi-graph management**, and you need to select a graph name. **Import Type** is not supported.
- The edge and vertex data sets can only be stored in English paths and folders.
- Currently, you can import the edge and vertex data sets only from OBS. You need to store data files in an OBS bucket..
- The sequence of the properties and labels in the selected edge or vertex data set must be the same as the sequence in the selected metadata file. Otherwise, **The edge/vertex data file does not match the metadata file** is displayed in the upper right corner and the graph fails to be created. For details about the graph data format, refer to **Graph Data Formats**.
- You need to import the graph data (including the metadata file, and edge and vertex data sets) in the format specified in the template. The template contains a copy of movie information. You can click **Download** to download and import it.

Step 4 Click OK.

----End

6 Managing Graphs

6.1 Graph Management Overview

On the **Graph Management** page, you can view the name, running status, internal access address, external access address, billing mode, and creation time of a graph.

NOTE

To view the **internal access address** is the floating IP address for accessing the graph instance. You can click the IP address to view the list of physical IP addresses of the graph instance. To prevent service interruption caused by floating IP address switchover, poll the physical IP addresses to access the graph instance.

Gra	ph M	anagement ③ O Creation tailed					•	nala Goph Prop	wy for Disco	unts
	lou car	create 16 more graphs using 6000 billion-edge que elect a property or enter a keyward.	a.						Q (8
		Name1D 0	Running Status ()	Internal Access Address	External Access Addr 0	Billing Mode 🖯	Created 0	Operation		
		005_060861 822#3#46-3466-4285-bc5b-12#167822#60	C Running	132	-	Pay-par-use	Mar 11, 2024 15:29:37 GMT+08:00	Access Back Up More		

Method 1: Click A next to a graph name to view the graph information, including Graph ID, VPC, Subnet, Security Group, Graph Size (Edges), Vertex Data Set, Edge Data Set, Metadata, Graph Version, Cross-AZ HA, Full-Text Indexing, Created By, Enterprise Project, CPU Architecture, Multiple labels, and Vertex ID Type (only available for database edition graphs).

Figure 6-1 Graph details tab

1	Graph Ma	anagement 🕞	O 1 Creation failed							Cr	sate Graph	Prepay for Discoun	•
	You can	create 98 more gra	aphs using 139.12 million-edge q	uota.									
	Q 86	elect a property or e	enter a keyword.									0)
		Name/ID \varTheta		Running Status \varTheta	Internal Access Address	⊖ External Access Addr ⊖	Billing Mode		Created \varTheta		Operation		
	۲	ges_hyg b88	10028e	S Running	192.	-	Pay-per-use Created on Dec 28, 2024	18:12:34	Dec 28, 2024 1	8:32:16 GMT+08.00	Access Back	Up More ~	
		TenThousand Ch	aroino				Pay-per-use						
1	Selected: g	jes_hyg											×
	Sraph ID		b8i	a00028e	Internal Access Address	192.1		External Access	Address	-			
1	/PC		vpo-c _autotest		Subnet	subnet-		Security Group		ges_s		Change	
	Graph Stee (B	Edges)	One-hundred-billion-edge		Vertex Data Set 🕥	View Details		Edge Data Set	0	View Details			
Ċ	ifetadata 🤆	0	View Details		Graph Version	2.4.8		Cross-AZ HA		No			
	Created By		ei_pe: _01		Enterprise Project	default		Fine-Grained P	ermission	No			
	CPU Architec	cture	×86		Full-Text/Vector Indexing	No		HyG computing	engine	Yes			
1	Aertex ID Typ	pe 🕤	String (fixed length) (Length: :	128)	SortKey Type 💮	Integer		Tag		View Details			
1	Data Encrypt	tion			Cryptographic Algorithm	generalCipher		Versions		Database edition			
1	Compute Rei	source	ECS		Enable LTS	No							

• Method 2: Click a graph name to access the details page and check its details. In the upper right corner of the page, you can click **Access**, **Back Up**, or **More** to manage the graph.

Figure 6-2 Graph details page

Graph Management List / ges_byg							
ges_hyg Numme Crisin ID b8875 Billing Mode Pay-per-Use Crisited on Dec 28,	92368 🗷 2004 18:12.34 GMT+08:80			Created Dec 20, 2024 18:32:10 GMT+08:00		C	Access Beck Up More v
Graph Detail							
Graph ID	b887 26025e	Internal Access Address	192		External Access Address	-	
VPC	vpcautolest	Subnet	subnet-		Security Oroup	per_x	Change
Graph Size (Edges)	One-hundred-billion-edge	Vertex Data Set 🕥	View Details		Edge Data Set 🕥	View Details	
Metadata 🕤	Vew Details	Graph Version	2.4.8		Cross-AZ HA	No	
Created By	e_gs 2_01	Enterprise Project	default		Fine-Orained Permission	No	
CPU Architecture	x86	Full-Text/Vector Indexing	No		HyG computing engine	Yes	
Vertex ID Type 🛞	String (fixed length) (Length: : 128)	SortKey Type 💮	Integer		Ting	Vew Details	
Data Encryption		Cryptographic Algorithm	generalCiphor		Versions	Database edition	
Compute Resource	ECS	Enable LTS	No				

6.2 Viewing a Failed Graph

If the ECS quota is insufficient, graphs may fail to be created. You can view failed graphs on the **Graph Management** page.

Procedure

- **Step 1** In the navigation tree on the left, select **Graph Management**.
- **Step 2** In the upper left corner of the displayed page, view the number of graphs that fail to be created next to **Graph Management**.

Figure 6-3 Number of failed graphs

Graph Management ⑦ Creation failed		
You can create 96 more graphs using 188.53 b	llion-edge quota.	
Name/ID 🕀	Running Status 😝	Internal Access Address

Step 3 Click ¹ to view the name, running status, and creation time of the graph that fails to be created. You can also delete the failed graph.

Figure 6-4 Viewing the creation status

Graph Creat Status							×
					Enter a graph nar	ne. C	a Q
Graph Name \ominus	Runnin \ominus Ir	nternal Acces 🖨	External Acce \ominus	Created #	•	Operation	
dont_delete_failed_graph	🕗 Failed		-	Aug 14, 20	019 12:29:23 G	View Details	Delete.

Graphs that fail to be created will occupy quotas if they are not deleted.

Step 4 Click **View Details** in the **Operation** column to go to the **Task Center** page. View the start time, end time, failure cause, and job ID of the failed creation task.

Figure 6-5 Task details

Engine ce	1.000111010.0011.0011.00.0	Cente	CURRENTVERSION US							
24								Enter	a job ID.	Q
Inerage	jobID 0	Type 0	Original Request 0	Status 0	Progress 0	Start Time 🖕	End Time 0	Operations	Running Res	ult
aragement	032693a3-6101-4861-b0ee-ab91e3e9	ImportGraph	("graphName":"CURRENTV	Succeeded	100%	Aug 02, 2023 16:04:45 GMT	Aug 02, 2023 16:04:47 GMT	Suspend	View Details	Cause of
ar a construction of the second s	85931478-b728-4808-a8ab-4ac83c0b	ImportGraph	[graphName":"CURRENTV	Succeeded	100%	Aug 02, 2023 16:04:43 GMT	Aug 02, 2023 16:04:44 GMT	Suspend	View Details	Cause of
on Management										

NOTE

Asynchronous task details can be retained only for one month. You cannot view information about graphs created more than one month ago.

----End

6.3 Backing Up and Restoring Graphs

6.3.1 Backing Up a Graph

To ensure data security, back up the graph data so that you can restore it when faults occur.

Procedure

You can perform the backup operation on the **Graph Management** page or the **Backup Management** page.

- 1. Graph management operations
 - a. Log in to the GES management console. In the navigation tree on the left, select **Graph Management**.
 - b. Locate the target graph in the graph list and select **Back Up** in the **Operation** column.
 - c. In the dialog box displayed, click OK.

Figure 6-6 Graph backup

Create Backu	p		×
Associated Graph:	ges_migtest		
		\langle	Cancel

On the **Graph Management** page, the backup operation can be performed only on the selected graph. The associated graph cannot be changed.

d. In the navigation tree on the left, click **Backup Management**. You can view the backup task in the backup list.

If **Status** is **Backing up**, wait several minutes. When **Status** is switched to **Succeeded**, the backup is successful.

Figure 6-7 Backup management

Ba	ckup Management 🕤												Create	Backup		mport
	You can create 36 more backups.															
	C. Select a property or enter a keyword.														Q	8
	Name/ID 0	Backup Ty 🖯	Graph Name 🖯	Graph Sta 😣	Graph Size ()	CPU Archi 🖯	Status ()	Created @	Ended ()	Backup Si 🖯	Backup D	θ	Operation			
	get_migtest-20240313094403 9ed50055-e47a-4a0b-9694-e8b4ba0ef564	Manual beckup	ges_migtest	8 Running	Ten-thousand-e	X85	O Backing up	Mar 13, 2024 D	-		2	0	Restore I	Delete E	sport	
	data_migration_test_cch_20240306211626_auto ef1132c4-e485-435a-930f-e54b52845afb	Automatic Back	data_migration	O Deleted	Million-edge	Kunpeng	Succeeded	Mar 09, 2024 2	Mar 09, 2024 2			0	Restore I	Delete E	oport	
	g45_multi-20231215182634 952ca524-5b83-4c7a-9e5c-991dee5c627c	Manual backup	ges_multi	C Deleted	Million-edge	X86	Succeeded	Dec 15, 2023 1	Dec 15, 2023 1			50	Restore I	Delete E	oport	

- 2. Backup management operations
 - a. Log in to the GES management console. In the navigation tree on the left, select **Backup Management**.
 - b. In the upper right corner of the **Backup Management** page, click **Create Backup**.
 - c. In the **Create Backup** dialog box, set **Associated Graph** (a graph created by the current user) and click **OK** to start the backup.

Figure 6-8 Creating a backup



NOTE

You can select an **Associated Graph** for the backup. However, if there is only one graph, you cannot change the value of **Associated Graph**.

d. In the backup list, you can view the data being backup up or newly backed up.

If **Status** is **Backing up**, wait several minutes. When **Status** is switched to **Succeeded**, the backup is successful.

Figure 6-9 Backup management

Ba	ckup Management 💿											Creat	le Backup		Import
	You can create 36 more backups.													0	
	Name1D 🖯	Backup Ty 😣	Graph Name 😣	Graph Sta 🖯	Graph Size \varTheta	CPU Archi 😣	Status ⊖	Created 🖨	Ended 😣	Backup Si 🖯	Backup D 😣	Operatio	n		
	ges_migtest-20240313094403 9ed90055-e47a-4a0b-9694-e9b4be0ef564	Manual backup	ges_migtest	Running	Ten-thousand-e	X86	O Backing up	Mar 13, 2024 0	-	0	0	Restore	Delete E	Seport	
	data_migration_test_cch_20240309211626_auto ef1132c4-e485-436a-930f-e54b52845afb	Automatic Back	data_migration	O Deleted	Million-edge	Kunpeng	Succeeded	Mar 09, 2024 2	Mar 09, 2024 2	1	0	Restore	Delete E	Seport	

e. Go to the **Backup Management** page, view the backup name and type, name, status, and size of the associated graph, CPU architecture, creation time, end time, backup size, and backup duration.

6.3.2 Restoring a Graph

If the graph data being edited is incorrect, you can load the backup data to restore the graph data for analysis.

D NOTE

Ten-thousand-edge graphs and graphs of the database edition cannot be automatically backed up. You need to back up a graph and restore data from the manul backup. For graphs of other sizes, you can restore data from an automatic backup or manual backup.

The procedure is as follows:

- **Step 1** Log in to the GES management console and choose **Backup Management** from the navigation pane on the left.
- **Step 2** On the **Backup Management** page displayed, locate the row containing your desired backup and click **Restore** in the **Operation** column.
- Step 3 In the Restore dialog box, select This operation will overwrite the target graph. After the restoration starts, the target graph will be restarted using the backup. Then, click Yes.

Figure 6-10 Restoring data

Backup Name	Target Graph	Target Graph Status
ges_migtest-20240313094403	dynamic_gr 🗸	Running
ne name of the graph associated aph to be restored. If you want to dex.	with the backup is inconsiste use the index after restoration	ent with the name of the ion, you need to rebuild th

Step 4 After a message is prompted indicating that the restoration is successful, you can access the target graph and obtain the restored data on the **Graph Management** page.

----End

6.3.3 Deleting a Backup

If backup data is no longer used, you can delete it as needed.

The procedure is as follows:

- **Step 1** Log in to the GES management console and choose **Backup Management** from the navigation pane on the left.
- **Step 2** In the backup list, select your desired backup and click **Delete** in the **Operation** column.
- **Step 3** In the displayed dialog box, click **Yes** to delete the data.

D NOTE

- 1. Deleted data cannot be recovered. Exercise caution when performing this operation.
- 2. You cannot delete the automatic backup data of a graph that has not been deleted.

----End

6.3.4 Exporting a Backup to OBS

To migrate GES data across regions, you can export backup files to OBS.

NOTE

- Graphs of the database edition do not support this function.
- Only graphs of memory edition 2.3.16 or later support this function. To export graphs of an earlier version, you need to upgrade the graphs by referring to Upgrading a Graph, and then export the graphs.
- You need to back up the graph on the **Graph Management** page so that the graph can be displayed on the **Backup Management** page. For details, see **Backing Up a Graph**.
- On the **Backup Management** page, only graphs whose **Graph Status** is **Running** and **Status** is **Successful** can be exported to OBS. Otherwise, the **Export** button is unavailable.

Ba	ckup Management 🕤											Create Ba	ickup (Import	
	You can create 36 more backups.													0.0	
	Name/ID 0	Backup Ty 🖯	Graph Name 😣	Graph Sta 😣	Graph Stee 🖯	CPU Archi 😣	Status 🖯	Created 🖨	Ended (8	Backup S 😣	Backup D 😣	Operation			
	ges_migtest-20240313094403 9ed900555-e47a-4a05-5694-e854be0e75d4	Manual backup	ges_migtest	Running	Ten-thousand-e	X86	Succeeded	Mar 13, 2024 0	Mar 13, 2024 0	1	50	Restore De	Jata Expor		

- **Step 1** Log in to the GES management console and choose **Backup Management** from the navigation pane on the left.
- **Step 2** In the backup list, select the backup to be exported and click **Export** in the **Operation** column.
- **Step 3** In the dialog box that is displayed, verify that the backup information is correct and select an OBS path.

Note that the OBS export path can only be an empty directory, and after the export, the graph data files under that directory cannot be deleted, added, or modified. Otherwise, the backup will fail when importing from OBS to the graph.

Export		×
Backup Name	Graph	Status
ges_migtest-20240313094403	ges_migtest	ges_migtest
The OBS path must be an empty di deleted, added, or modified. Otherv	irectory. After the export, files vise, the backup will fail to be	in the directory cannot be imported.
* OBS Path:		Đ
		No Yes

Figure 6-11 Exporting a backup to OBS

Step 4 Click **OK** to back up the graph.

NOTE

Storing backup files in OBS will incur charges. For details, see OBS Billing.

Step 5 After the task is delivered, you can view its execution status on the **Task Center** page.

----End

6.3.5 Importing a Backup from OBS

You can import a backup file exported to OBS to a graph. After the import is successful, you can use the backup to restore the graph instance.

- Graphs of the database edition do not support this function.
- Only graphs of memory edition 2.3.16 or later support this function. To export graphs of an earlier version, you need to upgrade the graphs by referring to **Upgrading a Graph**, and then import the graphs.

The procedure is as follows:

- **Step 1** Log in to the GES management console and choose **Backup Management** from the navigation pane on the left.
- **Step 2** In the upper right corner of the page displayed, click **Import**.
- **Step 3** In the dialog box that is displayed, select the graph to be imported and the OBS path where the backup is stored, and click **OK** to import the backup.

Figure 6-12 Importing a backup

Import		×
Graph	Status	
ges_migtest	V O Running	
Select a directory to en	sure successful backup import.	
* OBS Path:		E
	No	Yes

NOTE

Select a directory (folder) to ensure successful backup import.

Step 4 After the task is delivered, you can view its execution status on the **Task Center** page.

----End

6.4 Upgrading a Graph

Because the GES software is upgraded continuously, graphs of earlier versions can also be upgraded to the new version.

NOTE

Currently, only graphs of version 1.0.3 and later can be upgraded.

The procedure is as follows:

- **Step 1** Log in to the GES management console and choose **Graph Management** from the navigation pane on the left.
- **Step 2** Locate the target graph in the graph list and choose **More** > **Upgrade** in the **Operation** column.
- **Step 3** In the displayed dialog box, select a version from the **Version List** and determine whether to select **Forcible Upgrade**.

If **Forcible Upgrade** is selected, all in-progress tasks will be interrupted. Exercise caution when performing this operation.

Step 4 Click **OK**. The graph status changes to **Upgrading**. Wait a few minutes and the status will become **Running** once the upgrade is successful.

If the upgrade fails, the graph automatically rolls back to the source version.

----End

6.5 Rolling Back a Graph

You can roll back a GES graph to its source version after a successful upgrade.

NOTE

- You can only roll back graphs of version 2.4.4 or later.
- You can only perform a rollback within 30 days after completing an upgrade.
- During the rollback, the graph will be unavailable and the data is rolled back to that before the upgrade.

The procedure is as follows:

- **Step 1** Log in to the GES management console. In the navigation pane on the left, choose **Graph Management**.
- **Step 2** Select the graph you want to roll back in the graph list, click **More** in the **Operation** column, and select **Roll Back**.

NOTE

The Roll Back button is not displayed for graphs that cannot be rolled back.

Step 3 In the dialog box that appears, view the graph version. If the version is correct, click **OK**. The status of the graph changes to **Rolling back**. Once the rollback is complete, the status changes to **Running**, meaning that the graph is successfully rolled back.

----End

6.6 Exporting a Graph

You can export graph data to a custom OBS directory.

NOTE

- Graph data of memory edition 1.0.3 or later can be exported.
- Graph data of database edition 2.3.14 or later can be exported.

The procedure is as follows:

- **Step 1** Log in to the GES management console and choose **Graph Management** from the navigation pane on the left.
- **Step 2** Locate the target graph in the graph list and choose **More** > **Export** in the **Operation** column.

Export				>
★ Vertex Data Set	ges_vertex_			
★ Edge Data Set	ges_edge_			
★ Metadata Name	ges_schema_		.xml	
* Export Path 📎				Enter a keyword. Q
Name		Modified	Туре	Size
🗎 1c			BUCKET	-
🖻 ao			BUCKET	
🖻 co	s	-	BUCKET	-
🖻 cy		-	BUCKET	-
🖿 de		-	BUCKET	-
Total Records: 32	5 🗸 < 1	>		
				Cancel OK

Figure 6-13 Exporting a graph

- **Step 3** In the lower part of the page that is displayed, select a storage path. (For a graph of the database edition, you also need to select the graph name.)
- **Step 4** Click **OK**. The graph status changes to **Exporting**. Wait several minutes, the status will become **Running** after the export is successful.

You can check whether the data is exported successfully in the selected OBS path.

----End

6.7 Restarting a Graph

You need to restart a graph in the following cases:

- 1. If you access a graph in the **Running**, **Importing**, **Exporting**, or **Clearing** status and an unknown exception occurs, you can restart the graph.
- 2. You can restart a graph that is stuck in a state. For example, if a graph stuck in the **Exporting** status for a long time because the data to be exported is too much. You can restart the graph to stop exporting.

The procedure is as follows:

- **Step 1** Logging In to the GES Management Console.
- Step 2 In the navigation pane on the left, choose Graph Management. On the displayed page, locate the graph to be restarted and choose More > Restart in the Operation column.

Figure 6-14 Restarting a graph

Graph Management				G	Natio Graph Prepay for Discounts
You can create 15 more graphs using 6000 billion-edge quota.					
Q. Select a property or enter a keyword.					0
NameID () Running Status ()	Internal Access Address 😁	External Access Addr Θ	Billing Mode 🖯	Created 🛞	Operation
set_migrast s224046-3466-4285 bc66-12e1b7022eb0 Ranning	192.198.0.125	-	Pay-per-use	Mar 11, 2024 15:29:37 GMT+08:00	Access Back Up More ~
Opartic_prefx_bit 303a2c35-353a-8b10-8ba1-827548bar37d OR Running	192.198.0.186	-	Pay-per-use	Mar 12, 2024 20:41:48 GMT+08:00	Start
per_b0013750 s402597-7754-4e01-0669-b05288e3502c Running	192.168.0.2	100.93.11.110	Pay-per-use	Mar 05, 2024 21 23:44 GMT+08:00	Stop Delete
ce15ca65-0c18-450-6301-7618ad852cca	192.168.0.222	-	Pay-per-use	Mar 12, 2024 09:32:36 GMT+08:00	Export
10 V Total Records: 4 - < 1 ->					Bind EIP Unbind EIP
					Clear Data View Metric
					Query Schema Restart Restart

Step 3 In the displayed dialog box, check the name of the graph to be restarted.

NOTE

Restarting a graph will forcibly terminate the running task. For an import task, only partial data can be imported.

Step 4 Click **OK**. The graph status changes to **Stopping**. After several minutes, the graph status changes to **Running**.

----End

6.8 Resizing a Graph

Resize a graph to meet your storage, compute, and service needs.

NOTE

- Currently, Ten-thousand-edge graphs cannot be resized.
- After the resize, you need to re-create all indexes, including composite and full-text indexes.

The procedure is as follows:

Step 1 Log in to the GES management console. In the navigation pane on the left, choose **Graph Management**.

Step 2 On the displayed page, locate the row containing the graph for which you want to resize, click **More** in the **Operation** column, and select **Change Size** from the drop-down list.

Figure	6-15	Selecting	Change	Size
			e	

Graph Management					•	Prepary for Discounts
You can create 16 more graphs using 5998 99 billion-ed	be cupta.					
C. Select a property or enter a keyword.						0
Name10 0	Running Status 🖯	Internal Access Address	External Access Addr 😣	Billing Mode 🖯	Created @	Operation
						Access Back Up More ~
000	Running	192.168.0.49	-	Pay-par-use Created on May 20, 2024 19:00:56	May 20, 2024 19:06:45 GMT+08:00	Access Back Up More -
						Import
						Stop
						Delete
Total Records: 4 10 V C 1 >						Expert
						Bind EP
						Unbind EIP
						Clear Data
						View Metric
						Query Schema
						Restart
						Change Size
						Expand
						Task Center
						Operational monitoring
						Enable LTS
						Disable LTS

Step 3 In the displayed dialog box, select the target graph size.

NOTE

• The size of the graph cannot be changed to its current size. You can only reduce the size of a graph by one level, but you can increase it across levels.

For example, a Ten-million-edge graph can only be downsized to a Million-edge graph, and a Million-edge graph can be upsized to a Ten-million-edge graph or a graph with more edges.

• When CPU Architecture is X86, you can change the size to Billion-edge-pro.

Figure 6-16 Changing the size of a graph

Change Size		
Graph Size (Edges)	Million-edge	Ten-million-edge
	Hundred-million-edge	e Billion-edge
	Ten-billion-edge	
Current Size	Million-edge	
urrent Price	¥0.25 /hour	
rice After Change	¥15 /hour	
Selecting the same siz While changing the siz	ze as the current size is not a ze, the graph is in read-only	allowed. state, and any write atter
		Cancel

Step 4 Click **OK**. The graph status changes to **Preparing to change size**. After a few minutes, the graph status changes to **Changing size**. Once the size is successfully changed, the graph status changes to **Running**.

Note: While changing the size, the graph is in read-only state, and any write attempts will fail.

----End

6.9 Expanding a Graph

Graph expanding increases the maximum number of concurrent read-only requests that can be processed, without changing the graph size.

NOTE

- Currently, Ten-thousand-edge graphs cannot be expanded.
- Graphs cannot be resized after expansion. If you want to resize and expand the graph, resize the graph before you expand it.

The procedure is as follows:

- **Step 1** Log in to the management console.
- **Step 2** In the navigation pane, choose **Graph Management**. On the displayed page, locate the target graph and choose **More** > **Expand** in the **Operation** column.

Figure 6-17 Expanding a graph

Management 🛞	0 0 Creation failed							Oxede Graph Propey for D
can create 5 more graph	s using 4900 billion edge quo	6.						
). Select a property or en	lar a keyword.							(c
Name1D 🖯		Running Status 🖯	Internal Access Address	\varTheta External Access Addr 🖯	Dilling Mode 🛞	Created @		Operation
011_2)1011 17501xd9-11b3-435	7-8026-8255cb147892	8 Running			Pay per use	Mar 11, 203	si 17:47:45 GMT+08:00	Access Back Up More ~
999_/1/10 92091402-c005-400	4-a4e4-ac4cc88150ba	O Running			Pay-par-use	Mar 14, 20	19 24 05 GMT+08 00	Import Start
005_10214 be55e525-cb44-42	01-af79-a97e5ead7318	Running			Pay-per-use	Mar 12, 20	14 11:08:44 GMT+08:00	Stop Delete
Total Records	3 (1)							Upgrade Experi
								Bind EIP
								Clear Date
d ges_zytest				=				View Metric
	17981949-1153-4357-8826-82	155cb147892	Internal Access Address			External Access Address		Restart
	vpc-		Subnet	submit-640		Security Group	default	Resize
ize (Edges)	Million-edge		Vertex Data Set	View Details		Edge Data Set	View Details	Expand
	View Details		Oraph Version	2.3.19		Cross-AZ HA	No	Task Center
By	ei_ges_		Fine Grained Permission	No		CPU Architecture	3086	Operational monitoring
rd .	No		Operation Audit	No		Teg	View Details	
ryption	Yes		Cryptographic Algorithm	generalCiphor		Versions	Memory edition	
	F00		Multiple Inheir	No.				

NOTE

Only a running graph can be expanded.



Figure 6-18 Select the number of nodes to expand

Expand	×
Nodes to expand	- 1 +
Price	¥6.25 /hour
Existing nodes	1
	Cancel

 \sim

Step 4 Click **OK**. The graph status changes to **Expanding**. Wait several minutes, the status will become **Running** after the expansion is successful.

----End

6.10 Binding and Unbinding an EIP

Binding an EIP

To access GES over the Internet, you can bind an Elastic IP Address (EIP) to your instance.

The procedure is as follows:

- **Step 1** Log in to the GES management console.
- **Step 2** In the navigation tree on the left, select **Graph Management**.
- **Step 3** Locate the target graph in the graph list and choose **More** > **Bind EIP** in the **Operation** column.
- **Step 4** On the displayed **Bind EIP** page, select an available EIP.

If no EIP is available, click Create EIP to create one. Then, click	to refresh the
list and select the created EIP.	

Figure 6-19 Binding an E	IP
Bind EIP	×
EIP	✓ Q View EIP
	Cancel

Step 5 Click OK.

----End

Unbinding an EIP

If you do not need to use the EIP, you can unbind the EIP to release network resources.

The procedure is as follows:

- **Step 1** Log in to the GES management console.
- **Step 2** In the navigation tree on the left, select **Graph Management**.
- **Step 3** Locate the target graph in the graph list and choose **More** > **Unbind EIP** in the **Operation** column.

Step 4 In the displayed dialog box, click Yes.

----End

6.11 Clearing Data

If unnecessary data is imported or the imported data volume exceeds the graph size, you can clear the data.

In addition, if you delete data by mistake using Gremlin or Cypher commands, you can clear the broken data and import the correct data again.

NOTE

This operation will clear all vertex and edge data of the graph. Exercise caution when performing this operation.

The procedure is as follows:

- **Step 1** Log in to the GES management console. In the navigation pane on the left, choose **Graph Management**.
- **Step 2** In the graph list, locate your desired graph, click **More** in the **Operation** column, and select **Clear Data**.

Figure 6-20 Selecting Clear Data

pri Management · · · Creation falled						•	Inter Cape	UID COUR
ou can create 5 more graphs using 4500 billion-edge	quala.							
Q: Select a property or enter a keyword.								3
Name1D ()	Running Status 🕀	Internal Access Address	♦ External Access Add	Billing Mode 😣	Created ()		Operation	
set_color1 1705645-163-4357-4826-8255c6147882	Running			Pay-par-ase	Mar 11, 202	17.47.45 GMT-02.00	Access Back Up More ~	
 Set.2vs 92091402-ctor-4064-8444-9c4ccl885088 	Running			Pay-per-use	Mar 14, 200	4 19:24:05 GBIT+08:00	Start .	
0 545,20740 54729025-0545-4221-4772-42745044735	During			Pay-per-use	Mar 12, 202	4 11.08.44 (MIT-98.90	5lop Delete	
10 V Tatal Recents: 3 ()							Upgrade Export	
							Dird EIP	
							Unbied EIP	
			-				Clear Data	
cted. ges_zytest			-				Query Scheme	
10 1798869-193-4357-482	d-8255cb147882	Internal Access Address			External Access Address		Rostat	
wpore.		Subnet	subnit-649		Security Group	celeut.	Pesize	
Size (Edges) Million-edge		Wefex Data Set	View Details		Edge Data Set	View Defails	Expand	
ata Wew Defails		Oraph Version	2.3.19		Cross-AZ MA	No	Task Center	
el_ges_		Fine-Onlined Permission	No		CPU Architecture	X86	Operational maniforing	
sted Na		Operation Audit	No		Tag	View Defails		
incryption Yes		Cryptographic Algorithm	generalCoher		Versions	Memory edition		
ule Resource ECS		Multiple labels	No					

Step 3 In the dialog box that is displayed, select or deselect **Clear the metadata in the graph**. (For a database edition graph, you need to select the graph name first.)

NOTE

- If you clear graph metadata, the graph will be reset, and all data and running tasks will be cleared.
- Deleted metadata cannot be recovered. Exercise caution when performing this operation.
- Step 4 Click Yes.

----End

6.12 Deleting a Graph

If you have analyzed the graph data, you can delete the graph to release resources.

D NOTE

Backups of a graph will be also deleted after the graph is deleted, and data cannot be recovered. Exercise caution when performing this operation.

The procedure is as follows:

- **Step 1** Log in to the GES management console.
- Step 2 In the navigation tree on the left, select Graph Management.
- **Step 3** Locate the target graph in the graph list and choose **More** > **Delete** in the **Operation** column.

Step 4 In the **Delete Graph** dialog box displayed, determine:

- Whether to delete the EIPs bound to the graph instance. If no EIPs are bound, this option is unavailable. EIPs that are not selected will continue to incur fees. If you do not select the EIPs, the EIPs are retained by default.
- Whether to delete graph backups. By default, one automated backup and two manual backups are retained, occupying the backup quota. If you do not select the backups, the backups are retained by default.

Figure 6-21 Deleting a graph

Delete Graph		
Delete the following 1 graphs? Deleting the following graphs will release resources and clear data, and the g	raphs cannot be restored.	
Graph Name \ominus	Running Status \ominus	
dynamic_graph_zhf	Running	
The following 2 associated resources will be deleted after being selected:		
You can choose to delete some or all of the EIPs associated with the g advised to delete them.	praph. EIPs that are not deleted will continue to incur fees. You are	×
Associated Resource Type	Name \ominus	
EIP	100.95.153.38	
If you cancel the deletion of the following associated backup resource retained by default, occupying the backup quota.). You can also click	s, the graph backup will be retained. (Two manual backups are view backups to manually delete the account.	×
☐ Associated Resource Type	Quantity \ominus	
Backup data	1	
To confirm deletion, enter "DELETE" below. Auto Enter		
DELETE		
	Cancel	ок

Step 5 Click OK.

----End

6.13 Viewing Monitoring Metrics

Cloud Eye monitors the running status of GES. You can view the monitoring metrics of GES on the Cloud Eye management console.

It takes a period of time for transmitting and displaying monitoring data. The GES status displayed in the Cloud Eye monitoring data is the status obtained 5 to 10 minutes before. You can view the monitoring data of a newly created graph 5 to 10 minutes later.

Prerequisites

- The created graph is running properly.
- The graph has been properly running for at least 10 minutes. For a newly created graph, you need to wait for a while before viewing its monitoring metrics.
- You can view monitoring data of graphs in the running, importing, exporting, and clearing states.

Viewing Monitoring Metrics

- **Step 1** Log in to the management console.
- **Step 2** In the navigation pane, choose **Graph Management**. In the **Operation** column, choose **More** > **View Metrics**. The Cloud Eye management console is displayed.
- **Step 3** On the monitoring page for GES, you can view the figures of all monitoring metrics.

Figure 6-22 Viewing monitoring metrics

9. 20. 528 52 72 🔘 Auto Referen 🕥				Select Matric
randa Raw data +			Drive a rec	itrio name. C
Memory Usage 💮	Co	nputing Resource Usage 🕥	IOPS ()	
s.	Max Min %	Max Min	Countin	Max Min
40	32 29,660 20	16.940 11.330	12	10.880 6.630
10 A	- 15	manni	1 · march march	<u> </u>
20	11		V V	
A			3	
0935 1040 1005 1010 1015 1020 1025 1030 1035 1040 10	545 10:50	0955 1050 1055 1030 1035 1020 1025 1030 1035 1040 1045 1050	0955 1060 1065 1010 1015 1020 1025 1030 103	5 1040 1045 1050
Network Input Throughput	Ne	work Output Throughput	Disk Usage 🕥	
100 a	Max Max	Max Me		Max Mm
1	2.558 0.026	791.400.46.250	4	2.510 2.500
,		. λ	3 🗸	<u> </u>
	Δ	$\sim M$	2	
	1	in an hard hard		
0 08.55 10.00 10.05 10.19 10.15 10.20 10.25 10.00 10.35 10.40 10.	0.45 10.50	08.55 10.00 90.05 10.10 90.15 10.20 10.25 90.00 10.35 10.40 10.45 10.50	0 09:55 18:00 18:05 10:10 10:15 10:20 10:25 18:30 18:2	5 10:40 10:45 10:50
Total Data Size (1)	DH	t Saace Used ①	Disk Read Throughout (7)	
-	Max Mo ob	Max Min		Max Min
120	00.000.000.000	3,810 3,800	000	675.540 675.540
90		*	600	
60	2		400	
30			200	
· · · · · · · · · · · · · · · · · · ·	0		0	
1953 1050 1053 1010 1013 1020 1023 1010 1033 1080 10		CASS NEW DATE TATE THEY THEY THEY THEY THEY THEY THEY	0103 000 1003 1019 1020 0023 1000 100	13 12/10 10/13 10/32
Disk Write Throughput 🛞	ka	rape Time per Disk Read 🛞	Average Time per Disk Write 🛞	
1893 Y	Max Mrs 4 78,490 78,400	Max Mir.		Max Mm 0.150 0.000
90			02	
60 Y	0.3			\sim
30	0.1			~
	0.3			
0955 1000 1005 1010 1015 1020 1025 1030 1035 1040 10	245 10:50	0955 1000 1005 1010 1015 1020 1025 1030 1035 1040 1045 1050	09:55 10:00 10:05 10:10 10:13 10:20 10:25 10:30 10:1	15 10:40 10:45 10:50

Step 4 The system allows you to select a fixed time range or use automatic refresh.

- 1. Fixed time ranges include **1h**, **3h**, **12h**.
- 2. The automatic refresh interval is 60s, which is used as the user monitoring period.

----End

6.14 Querying Schema

Query the metadata of a graph. The metadata contains labels and properties.

The procedure is as follows:

- **Step 1** Log in to the management console.
- Step 2 In the navigation pane, choose Graph Management. In the Operation column, choose More > Query Schema. A window is displayed, showing the labels contained in the metadata of the current graph.

Figure 6-23 Querying schema

Query	Schema	;
~	Label Name	rate
~	Label Name	user
~	Label Name	default
~	Label Name	movie
~	Label Name	DEFAULT

Step 3 To view the properties contained in a label, click \checkmark of each label.

Figure 6-24 Viewing properties in labels

Query Schema			×
∧ Label Name rate			
Property Name	Cardinality	Data Type	
Rating	Single value	int	
Datetime	Single value	string	



6.15 Connecting GES to LTS

Enable LTS to check service run logs.

NOTE

Multigraph clusters do not support interconnection with LTS.

Prerequisites

Before enabling LTS, you have created a log group and a log stream on the LTS console. For details, see **Creating a Log Group** and **Creating a Log Stream**.

Enabling LTS

The procedure is as follows:

- 1. Log in to the GES management console. In the navigation pane on the left, choose **Graph Management**.
- 2. On the displayed page, locate the row containing the graph whose service logs you want to check, click **More** in the **Operation** column, and select **Enable LTS** in the drop-down list.

Figure 6-25 Enabling LTS

Name1D A	Running Status A	Internal Access Address A	External Access Addr., 🖗	Dilling Mode A	Created A	Operation
01.	Running	192,168.0.136		Pay-per-use Created on May 17, 2024 15:32:35	May 17, 2024 15:30:26 OMT+08.00	Access Back Up More ~
						Import
						Start
						Delete
						Upgrade
						Export
						Under CIP
						Clear Data
						View Metric
						Restort
						0.001028
e807 10 V < 1 >						Expand

- 3. In the displayed dialog box, select a log group and log stream.
 - Log group: A log group is the basic unit for log management and needs to be created on the LTS management console. You can click View Log Group List to check existing log groups. If you have not created a log group, create one by referring to Log Groups.
 - Log stream: A log stream is the basic unit for reading and writing logs and belongs to a log group. For details about how to create a log stream, see Log Streams.

×

Figure 6-26 Selecting a log group and log stream

Enab	le LTS			
0	Storing logs to LTS is bille standards.	d. For details, see the LTS billing		
Log	l Group	View Log Group List	~	Q
Log) Stream		\sim	Q
		C	Cancel	ок

4. Confirm the information and click **OK**. The graph status changes to **Enabling LTS**. In the navigation pane on the left, choose **Task Center**. On the displayed page, locate the row containing the desired graph name and its corresponding task name **Enable LTS**. When the graph status changes from **Running** to **Succeeded**, LTS is successfully enabled.

Figure 6-27 LTS enabled



5. In the graph details, click the log group link next to the value of **Enable LTS**. The **Log Management** page of the LTS management console is displayed, facilitating your management of the log group.

Figure 6-28 Clicking the log group link

ges_large_dsta_ 4af4b3dc-b258-4	1w_2h1 O Running 8cb-b745-b64777bcd302	192.168.0.77	-	Pay-per-use Created on May 23, 2024 09:38:36	May 23, 2024	09:38:27 GMT+08:00 Access Back Up More ~
	mi 🔷 Duration	107 159 0 157		Pay-per-use	11 22 2024	18-14-20 GMT_400-05 Access Deck Lin Mass or
Selected: ges_large_data_	1w_zhf					
Graph ID	4#f4b3dc-b258-48cb-b745-b647f7bcd302	Internal Access Address		External A	cess Address	-
VPC	vpc-2a53	Sutnet	subnet-2a61	Security G	oup	dws-ges_ell-8000 Change
Graph Size (Edges)	Ten-thousand-edge	Vertex Data Set 💿	View Details	Edge Data	Set 💿	View Details
Metadata 💮	Wew Details	Graph Version	2.4.2	Cross-AZ	iA.	No
Created By	w00346654	Fine-Grained Permission	No	CPU Arch	ecture	X85
Encrypled	No	Vertex ID Type 💮	String (fixed length) (Length: : 30)	SortKey T	pe 🛞	Integer
Тад	Wew Details	Data Encryption	Yes	Cryptogra	hic Algorithm	generalCipher
Versions	Database edition	Compute Resource	ECS	Enable LT	1	Yes ges_lts_viwan_group

Disabling LTS

1. On the **Graph Management** page, locate the target graph, click **More** in the **Operation** column, and select **Disable LTS** from the drop-down list.

Figure 6-29 Disabling LTS

elect a property or enter a keyword.						
NameID	Running Status	Internal Access Address	External Access Address	Billing Mode	Created	Operation
pm_	O Running	192.168.0.156	-	Pay-per-use Created on May 2024 15 30 35	A	Access Back Up Mare ~
10033910-0900-490-980-09						Impart
						Stat
						Stop
						Delete
						Upgrada
						Export
	-					Bind EIP
						Unbind EIP
						Cine Deta
						Diary Schama
						Restart
						交更规则
iconds:7 10 √ (1 →						Expand
						Task Center

2. In the displayed dialog box, check the graph information and click **OK**.

Figure 6-30 Confirming information

Disable LTS		×
A Storing logs to LTS sure you want to di	is billed. For details, see the sable LTS for this graph?	ne LTS billing standards.Are you
Graph Name 🖨	Log Group	Log Stream \Leftrightarrow
ges_		
		Cancel OK

3. The graph status changes to **Disabling LTS**. In the navigation pane on the left, choose **Task Center**. On the displayed page, locate the row containing the desired graph name and its corresponding task name **Disable LTS**. When the graph status changes from **Running** to **Succeeded**, LTS is successfully disabled.

Figure 6-31 LTS disabled

Service										
Ovenniew	C. Select a property or e	nter a keyword.								00
Oraph Management	Туре	Task Name	Graph Name	Associated Graph	Start Time	End Time	Status	Job ID	Operation	
Backup Management	Graph Management	Disable LTS					Succeeded		View Details Cause	o of Failure
Netadata Nanagement Taak Center							•			
Connection Management										
Access Sandbox (2) Oranular Permissions										
Resource Packages 🗸 🤆										

6.16 Changing a Security Group

Change the security group for a created graph.

The procedure is as follows:

- **Step 1** Log in to the GES management console. In the navigation pane on the left, choose **Graph Management**.
- **Step 2** On the displayed page, select the graph for which you want to change the security group. The graph information is displayed in the lower part of the page.

Graph Management	O 1 Creation failed							Prepay for Discounts
You can create 16 more gr	aphs using 5999 billion-edge quo	ta.						
Q. Select a property or	enter a keyword.							Q 0
Name1D 🕀		Running Status 🖯	Internal Access Address	⊖ External Access Addr	Billing Mode 😣	Created 😣		Operation
944 b0909704-5e82-	#37-8e10-373ce9dc8579	Running		-	Pay-per-use	Apr 16, 2024	09.39.00 GMT+06:00	Access Back Up More ~
e31763a1-64c5-4	44e-8190-6e5305471335	8 Running		100.85.112.188	Pay-per-use	Apr 15, 2024	20.10.33 GMT+06:00	Access Back Up More ~
82240645-3466-	1285-0c60-12e107022e00	Runing		-	Pay-per-use	Mar 11, 2024	15:29:37 GMT+00.00	Access Back Up More ~
Total Records: 3 10	▼ < 1 →							
elected: ges_								
iraph ID	bb909704-5e82-4/37-8e10-37	3ce9dc8579	Internal Access Address	192.168.0.192		External Access Address	-	
PC	vpc-2a53		Subnet	subnet-2a61		Security Group	dws-ges_e6-6000 Ch	inge
raph Size (Edges)	Million-edge		Vertex Data Set	View Details		Edge Data Set	Wew Details	
etadata	View Details		Oraph Version	2.4.1		Crass-AZ HA	No	
sealed By	el_ges_c00451695_01		Fine-Grained Permission	No		CPU Architecture	X86	
ncrypted	No		Operation Audit	No		Tag	View Details	
ata Encryption	Yes		Cryplographic Algorithm	generalCipher		Versions	Memory edition	
ampute Resource	ECS		Multiple labels	No				

Step 3 In the graph details, you can check the security group of the current graph. Click **Change**. In the displayed **Change Security Group** slide-out panel, select another security group.

Figure 6-32 Graph details page
Change Security Gro	pup	×
You are currently swith enabled for service a service access failure	tching security groups. Make sure that ports 80 and 443 are ccess. Any incorrect security group configurations may result in e. So, exercise caution when performing this operation.	
Graph Name	ges_LTS_h30049850_million	
Security Group	default ~ Q	



Step 4 Click OK.

----End

6.17 Changing the Security Mode

After creating a graph, you can change the security mode of the graph on the graph details page.

NOTE

Only GES 2.4.4 or later graphs that are in the **Running** state support this feature.

The procedure is as follows:

- **Step 1** Log in to the GES management console. In the navigation pane on the left, choose **Graph Management**.
- **Step 2** On the displayed page, select the graph for which you want to change the security mode. The graph information is displayed in the lower part of the page.

-		-							
Sraph Engine Service	Graph Management	 O 0 Creation failed 						Create Graph Prepay for Disc	sunts
veniev	You can create 18 more	graphs using 6000 billion edge qu	sta.						
iraph Management	C. Select a property of	or enter a keyword.						Q	۲
lata Migration	Name10 0		Running Status 😣	Internal Access Address	e External Access Addr e	Billing Mode (8)	Created (6)	Operation	
lackup Management	O bab 15a3a-17d	1-4309-80xa-70052545863d	O Upgrading		-	Pay-per-use Created on Aug 02, 2024 09:45:44	Aug 02, 2024 09:45:34 GMT+08	00 Access Back Up More -	
asik Center	ex. a2r32b33-ec19	9-4aa6-953f-faee075ac3a2	O Running			Pay-per-use Created on Jul 29, 2024 16:20:41 G.	Jul 29, 2024 16:20:31 GMT+08	Access Back Up More +	
connection Management	Tatal Records: 2 1	• • • • •							
Instanular Permissions 🔍 🗌									
	Selected ges_zhf_secur	ed							×
	Graph ID	82c32b33-ec19-4se5-9531-te	ee075ad3a2	Internal Access Address	192.108.0.03	External Ac	coss Address -		
	VPC	vpc-2853		Subnet	subnet-2a61	Security Gr	roup divis divis naj 80	00 Change	
	Graph Size (Edges)	Ten-thousand-edge		Vertex Data Set	Vev Details	Edge Data	Set View Details		
	Metadata	View Details		Graph Version	2.4.4	Cross-AZ F	IA No		
	Created By	z00209327		Fine-Grained Permission	No	CPU Archit	lecture x86		
	Encrypted	No		Tag	Vev Details	Data Encry	ptice		
	Cryptographic Algorithm	generalCipher		Versions	Memory edition	Compute R	besource ECS		
	Multiple labels	No		Enable LTS	No				

Figure 6-33 Graph details page

- **Step 3** In the graph details, you can enable or disable the security mode.
- **Step 4** The **Data Encryption** slider is grayed out during the security mode change. You can view the task progress in the Task Center.

Figure 6-34 Viewing the task status

Graph Engine Service	Task Center ③									
Overview	Q. Select a property or e	inter a keyword.								
Graph Management	Туре	Task Name	Graph Name	Associated Graph	Start Time	End Time	Status	Job ID	Operation	
Backup Management	Graph Management	Update security mode	per_211	015_1	Aug 02, 2024 11:25:49	Aug 02, 2024 11:20:05	Succeeded	209081e39110e4b6019111114b210036	View Details Cause of Failure	
Metadata Management Task Center	Trap Response	100001-1100	$\{0,1,1^{(1)},1^{(2)},1^{(2)},1^{(2)}\}$	10.01	$\lambda \in [0,\infty) \to [0,\infty)$	$(a_{2},a_{1},a_{2},a_{3},a_{4},a_{5},a_{$	•			

----End

7 Data Migration

7.1 Functions

The GES data migration feature allows you to easily import data from common relational databases (MySQL, Oracle, and ShenTong MPP) and big data components (DWS, Hive) into a graph instance with just one click. You only need to preprocess the raw data into the vertex-edge table required by GES and then use the GUI to import these tables into the graph instance. This eliminates the previously tedious intermediate steps of generating metadata, exporting to CSV, uploading to OBS, and importing to GES, significantly facilitating the process of importing user data into the graph.

Precautions

- 1. To migrate data, all data from each table in the database will be imported into the graph instance as either vertex or edge data sets. Therefore, ensure that the tables in the database have been processed as either vertex or edge data.
- 2. For the data types supported by vertex and edge tables, see the property description in **Static Graph**.
- 3. Vertex table format: *Vertex ID column name, Vertex label column name, Vertex property column name 1, Vertex property column name 2...*

MySQL [test]> de	sc perso	on;			44
Field	Type	Null	Key	Default	Extra
vertex_id person_vertex firstname lastname gender birthday creationdate locationip browserused lang email	text text	YES YES YES YES YES YES YES YES YES YES		NULL NULL NULL NULL NULL NULL NULL NULL	

The following figure shows the data in a vertex table:

Figure 7-2 Data in a vertex table

MySQL [test]>	select * from person limit 1 \G;
*******	**************************************
vertex_id:	21990232556410
person_vertex:	Person
firstname:	Aa Ngurah
lastname:	Gallagher
gender:	male
birthday:	1989-08-27
creationdate:	utc datetime: 2011-09-09T03:02:45.579000, timezone_offset: 0
locationip:	36.95.74.186
browserused:	Chrome
lang:	NULL
email:	NULL
1 row in set (0.003 sec)

4. Edge table format: *Source vertex ID column name, Target vertex ID column name, Edge label column name, Edge property column name 1, Edge property column name 2...*

MySQL [test]> d	lesc know	s;	.		..
Field	Type	Null	Key	Default	Extra
source_id target_id knows_edge creationdate +	text text text text text	NO NO NO NO	++	NULL NULL NULL NULL	

Figure 7-3 Edge table format

The following figure shows the data in an edge table:

Figure 7-4 Data in an edge table

7.2 Creating a Data Source

Prerequisites

You have obtained the type, CIDR block, IP address, port, database name, and authentication information of the data source.

Procedure

- **Step 1** Log in to the GES management console. In the navigation pane on the left, choose **Data Migration**.
- **Step 2** On the displayed **Data Sources** tab, click **Create**.

Figure 7-5 Creating a data source

 Data Migration ③

 Data Sources
 Data Migration

 Create

Step 3 On the displayed page, set the parameters parameters as follows:

- **Data Source Name**: Enter a name. The value must contain 4 to 50 characters and start with a letter. Only letters (case-insensitivity), numbers, and underscores (_) are allowed.
- Data Source Type: Select a type. The options are MySQL, ShenTong database, Oracle, GaussDB(DWS), and Hive.
- **Graph Name**: Select the graph where you want to import data.
- CIDR Block: CIDR block of the subnet where the data source is.
- Access IP Address: IP address of the database of the data source.
- Access Port: port number of the data source database (not involved in Hive).
- Database Name: Name of the database of the data source.
- **Database Username**: Username for logging in to the database of the data source (unavailable for Hive).
- **Database Password**: Password for logging in to the database of the data source (unavailable for Hive).

- **Require Verification**: Check if Kerberos authentication is enabled for the MRS cluster where Hive is (only available for Hive).
- **MRS Cluster Username**: Username of the MRS cluster where Hive is (only available for Hive). This parameter is optional when Kerberos authentication is disabled.
- **MRS Cluster User Authentication Credential**: Authentication credential of the user of the MRS cluster where Hive is (only available for Hive). This parameter is optional when Kerberos authentication is disabled.
- MRS Cluster Hive Client File: Hive client file (only available for Hive).

Click **OK**.

		×
Create Data Source		
Data Source Name		
Data Source Type	MySQL	
Graph Name	ges_gj_new V Q	
CIDR Block	/ 0	
Access IP Address	· · ·	
Access Port		
Database Username		
Database Password	Ø	
Databasa Nama		
Database NdIII8		

Figure 7-6 Data source information



Step 4 Check the data source status and wait until the creation is complete.

----End

7.3 Creating a Data Migration Task

Prerequisites

The types of vertices and edges corresponding to each table in the database of the data source have been confirmed.

Procedure

- **Step 1** Log in to the GES management console. In the navigation pane on the left, choose **Data Migration**.
- **Step 2** On the displayed page, click the **Data Migration** tab.

Figure 7-7 Creating a data migration task

Data Migration 🧿	
Data Sources	Data Migration
Create	
Q Select a propert	y or enter a keyword.

Step 3 Set data source parameters.

- **Task Name**: Enter a name that is not already in use. The value must contain 4 to 50 characters and start with a letter. Only letters (case-insensitivity), numbers, and underscores (_) are allowed.
- Data Source: Select a data source you created.
- Associated Graph Name: It automatically appears once a data source is selected.

Figure 7-8 Data source configuration

Data Source Configur	ation	
Task Name		
Data Source	test001	~
Associated Graph Name	ges_yiwan_new	



- Vertex File Sources: Select the tables where vertex data is from the Available
 Files area and click > to add them to Selected Files.
- Edge File Sources: Select the tables where edge data is from the Available
 Files area and click > to add them to Selected Files.
- Schema File: When you create a migration task for the first time, generate a schema file by performing Step 5. Then, select the schema file. (In ECS/BMS +MRS mode, you need to select the storage path of the schema file.)

Figure 7-9 Metadata configuration

ertex File Sources	Available Files	1/5		Selected Files	0 / 0
	Q Enter a keyword.			Q Enter a keyword.	
	gaussdb.forum				
	gaussdb.has_tag				
	gaussdb.knows			No data available.	
	gaussdb.person				
	gaussdb.tag				
		< 1/1 >			
dge File Sources	Available Files	1/5		Selected Files	0 /
	Q Enter a keyword.			Q Enter a keyword.	
	gaussdb.forum				
	gaussdb.has_tag				
	gaussdb.knows		<	No dele succesione	
	gaussdb.person			NO GALA AVAIJADIE.	
	gaussdb.tag				

Step 5 Generate a schema file.

1. Click Generate Schema.

Figure 7-10 Schema file

testSchema	~	Q	Generate Schema
	testSchema	testSchema 🗸	testSchema V Q

2. In the displayed dialog box, set **Schema Name** and **Schema Storage Path** and click **OK**.

Figure 7-11	Creating a	metadata file
-------------	------------	---------------

Create Metadata	;	×
Schema Name	testSchema	
Schema Storage Path	ges-data-qjh/testSchema.xml	
ges-data-qjh	Enter a keyword. Q	
← Previous		
🖻 edge		
🖻 schemal		
🖻 vertex		



3. In the displayed dialog box, click **Go**. On the displayed **Data Migration** tab page, you can view the status of the metadata file creation task and wait until the task is successfully executed.



Data Obarces Data migration								
Create								
Q. Select a property or enter a keyword.								08
Task Name/D 😔	Execution Status ()	Task Type $ \Theta $	Started Θ	Ended ()	Request Parameter (9)	Associated Graph Name $ \Theta $	Operation	
testSchema 2c9080be923c7b40019274312f1b0116	Succeeded	Create metadata	Oct 10, 2024 10:10.33 GMT	Oct 10, 2024 10:10:51 GMT	{"schema" {{"lables" {{"colum	ges_viwan_new	View Details	Delete

Step 6 On the data migration task creation page, set the following import parameters:

- **Repetitive Edge Processing**: Select a repetitive edge processing policy. (You can only select **Ignore subsequent repetitive edges** or **Overwrite previous repetitive edges** for database edition graphs.)
- **Ignore Labels on Repetitive Edges**: Whether to ignore labels for repetitive edges (not involved in database edition graphs).
- **Offline Import**: Whether to enable offline import. (During offline import, graphs cannot be read or written. However, database edition graphs can still be properly used.)

Figure 7-14 Importing configurations

Import Configuration			
Repetitive Edge Processing	Ignore subsequent repetitive edges	Overwrite previous repetitive edges	Allow repetitive edges
Ignore Labels on Repetitive Edges			
Offline Import			

Step 7 Set storage path parameters.

- Vertex File Storage Path: stores vertex data exported from the database of the data source.
- **Edge File Storage Path**: stores edge data exported from the database of the data source.
- **Log Storage Path**: stores log files generated during data import.

Figure 7-15 Storage path configuration

Storage Path Configurat	ion	
Vertex File Storage Path	Ê	ť
Edge File Storage Path	È	9
Log Storage Path	È)

Step 8 Click **Create**. On the **Data Migration** tab page, view the migration task progress and result.

Figure 7-16 Viewing the migration result

Data	Migration ③								
D	ata Sources Data Migration								
	Create								
C	Q " Jadd titler								×Q®
	Task Name1D 😔	Execution Status 🖨	Task Type $ \Theta $	Started 😔	Ended 0	Request Parameter \varTheta	Associated Graph Name $ \Theta $	Operation	
	gestes100001 2c5080de923c7e9a0192710b0e2d02b8	Succeeded	Create migration task	Oct 09, 2024 19:30:22 GMT	Oct 09, 2624 19:30:37 GMT	("name":"gestest00001","offi	TenThousand_Charging	View Details	Derete Stop

You can click **View Details** in the **Operation** column of the migration task to view the task status of each vertex or edge dataset.

Figure 7-17 Task details

Data Migration ③												
Data Sources Data Migration	View Det	ails							×			
Create	Type \ominus	File Path	Status 🖯	Cause of Failure	Log	Total Impo	Row Impo	Successf				
Q Keyword: gestest00001 × Add filter	Vertex D		Succeeded	-	-	13750	0	13750				× Q 🛛
Task Name/ID 😔	Metadata		Succeeded	all labels are impor		2	2	0		Associated Graph Name 😔	Operation	
gestes100001 2c9080de923c7e9a0192710b0e2d02b8									,"offi	TenThousand_Charging	View Details	Delete Stop

----End

8 Accessing and Analyzing Graph Data

8.1 Accessing an Infinite Graph

To improve the display experience, you can use the infinite graph access mode to process and analyze graph data.

NOTE

Currently, the only way to access and analyze infinite graphs is using Cypher queries.

The procedure is as follows:

- 1. Log in to the GES management console. In the navigation pane on the left, choose **Graph Management**.
- 2. In the graph list, locate the row containing the graph you want to access and analyze, click **More** in the **Operation** column, and select **Access Infinite Graph**.

Figure 8-1 Accessing an infinite graph

n create 16 more graphs using 6000 bills	on-edge quota.					
Select a property or enter a keyword.						
Name(ID 🖯	Running Status \varTheta	Internal Access Address	External Access Addr 🛞	Billing Mode 😣	Created O	Operation
				No. of Concession, Name and Name	and 2010 101 101 101 101	
-						
				1000 cm	***	
per_an_	Running		-	Pay-per-use Created on Jun 21, 2024 16 50 58 G	Jun 21, 2024 16:50:47 GMT+08:00	Access Back Up More ~

3. If you use Cypher for the first time, click **Create Index** to create an index. (Skip this step if an index has been created.)

Figure 8-2 Creating an index

S Command Box	A 41
Create Index	10

If you are a first-time user of Cypher, create an index first.

4. Enter Cypher commands below and press **Enter** to execute them. The command output will be displayed on the canvas.





5. The vertices in the chart data are displayed in the upper right corner. You can click ⁽²⁾ to hide them from the infinite graph on the canvas.

Figure 8-4 Hiding a vertex



- 6. Check vertex and edge properties.
 - a. When hovering over a vertex or edge in the canvas, its ID and properties are displayed.



Figure 8-5 Checking vertex information

b. Double-click a vertex or edge in the canvas. In the displayed slide-out panel, you can check the properties of the vertex or edge.

Figure 8-6 Checking vertex properties



7. Analyze the graph using the function keys in the upper right corner.

Figure 8-7 Function keys

ges_zhf_1w 🛛 🗄 🖉 🖸 📩 🕲 🍳 🔯 Q 🗘 🗘 🗰 🕺 🔯 Bolated Vertices Neighbor vertices

Table 8-1 Function keys

Function Key	Description							
≣	Hides the function bar. Clicking it will collapse the function bar.							
R	Fixes the function bar. After dragging the function bar to any position, clicking this button will fix the function bar in place.							
昏	Clears all content on the canvas.							
Ð	Zooms in the graph. You can zoom in a graph to at most 600%.							
Q	Zooms out the graph. You can zoom out a graph to 5%.							
[0]	Adapts to the graph. Clicking it will restore the graph to its appropriate canvas size.							
Q	Searches for vertices and edges in the entire graph data or the currently running data.							
	Figure 8-8 Searching for data							
	All data ^ Enter a vertex ID or multiple IDs separate	E C						
	All data Double-click of ri	righ						
	Current data							
\$	Displays the current running results in a circular layout.							
ш	Displays the current running results in a grid layout.							
A	Displays the current running results in a hierarchical layout.							
24	Displays the current running results in an automatic segmentation layout.							
Isolated Vertices	An isolated vertex is a vertex that has no connection to other vertices and exists independently on the canvas.							
	Clicking this button will highlight all isolated vertices on the canvas. If there are no isolated vertices, clicking this button will have no effect							

Function Key	Description
Neighbor vertices	Select any vertex on the canvas and click this button to view all vertices associated with the currently selected vertex.

8.2 Graph Editor

The graph editor consists of several sections: analysis section (including exploration, operation, schema, algorithm, and index sections), canvas, query text box, result display pane, and filtering and property tabs.



	o Graph analysis					∣ක්ඵම€	≊∣QQI,II	> o = ×	$V \otimes H$
949	Explore Operations Schema Algorithms Indexes				ex from the shortcut menu to Show/Hirle Sen	add a vertex.	Fillering		Statistics
	A Start weber D								
									¥)
	Search Criteria (D)								
	1869								
	• · · · · · · · ·							Filter	
									area
						146 Vertex			
		Running Record Quer			Created Popol				
						<u> </u>			
						0			
						le enter			
						to-enter			
		Total Records: 2 $10 \sim 1$							

Table 8-2 Graph editor

Area	Description
Exploratio n pane	Graph exploration tools, for example, path expansion. For details about the functions, see Exploring Graphs .
Operation s	Operations executed by API calls. For details, see Adding Custom Operations .
Schema	Metadata operations, such as adding, hiding, importing, and exporting data. For details, see Editing Schema .
Algorithm s	Algorithms supported by GES. You can set the properties of each algorithm in this area. Table 8-3 describes the functions of the algorithm library. NOTE After you select an algorithm in the algorithm library and execute it, the canvas displays the sampling sub-graph that contains the key result. The
	execution result is incomplete. To obtain the complete returned result, call the corresponding API.
Index area	The index management function is added to the graph access page to facilitate index addition, deletion, and search.

Area	Description			
Canvas	Graph structure of data. Shortcut operations are preset in the drawing area for you to easily analyze the graph data. Table 8-4 describes the functions of the drawing area.			
Query box	 Gremlin query statements Cypher query statements DSL query statements 			
Result display pane	 There are two tab pages: Running Record where you can Check Running Records. Query Result where you can Viewing Query Results. 			
Filter and Property area	On the canvas, select a vertex and right-click it. Then, choose View Property from the shortcut menu to view the Filter and Property area.			
	 The Filtering tab page allows you to set properties and conditions to filter the data for analysis. For details, see Filter Criteria. 			
	• The Property tab page displays the property information about a vertex or an edge.			
	 The statistics tab displays the number of labels and vertex weights of the selected vertices and edges. For details, see Statistics Display. 			

Figure 8-10 Algorithm Library





Interface Element	Description
Enter an algorithm name. Q	Enter the algorithm name to quickly find it.
\sim	Expand the algorithm parameter configuration area.
\odot	Run the algorithm.
∧	Set the properties of an algorithm. Different algorithms have different properties. For details, see Algorithms .

Figure 8-11 Canvas



Table 8-4 Canvas description

Interface Element	Description
13 /886813 Vertex 9 /892773 Edge	Row 1: 13 indicates the number of vertices displayed on the current canvas and 886813 indicates the total number of vertices in the entire graph.
	Row 2: 9 indicates the number of edges displayed on the current canvas and 892773 indicates the total number of edges in the entire graph.
Isolated Vertices	An isolated vertex is a vertex that is not an endpoint of any edge.
	 To display isolated vertices in a selected area, press Ctrl and click and drag to select an area on the canvas, and then click Isolated Vertices.
	 To display all isolated vertices in the canvas, click Isolated Vertices.
Neighbor vertices	Select a vertex in the canvas and click neighbor vertices to view all vertices associated.
Undo	Cancel the previous operation.
Redo	Redo the canceled previous operation.
	Select All data or Current data.
	• All data indicates all data of a graph.
	• Current data indicates the data rendered on the canvas.
ि 🕁 Theme 🗸 🗸	You can change the theme of the graph editor. Three themes are supported: light, dark, and system.

Interface Element	Description
Enter a vertex ID or multiple IDs separate Q	 After you select All data or Current data, enter the node ID in the search box, for example, 2. Press Enter or click the query icon to search for the corresponding vertex and render it to the canvas. NOTE Currently, only a single vertex ID can be entered. If you choose Current data from the drop-down list, vertices on the current canvas are highlighted.
Е́а	Click Clear to clear all content on the canvas.
仚	Export the canvas content as a TXT file (snapshot or vertex and edge file of the current canvas).
	 Keyboard shortcuts Ctrl+E: Select an associated entity. Ctrl+'-': Zoom in. Ctrl+'-': Zoom out Ctrl+Z: Undo an operation. Ctrl+A: Select all. Ctrl+Delete: Clear the canvas. Delete: Hide vertices. Ctrl+Click: Select multiple vertices and edges.
0	You can create a snapshot for the graph shown on the canvas and then restore it from the snapshot. For more information on this feature, refer to the snapshot section.
Q	Zoom in the graph. You can zoom in a graph to at most 600%.
Q	Zoom out the graph. You can zoom out a graph to 5%.
1:1	Automatic screen adaptation When the displayed graph data is too large (cannot be completely displayed) or too small, you can click this button to quickly adjust it based on the screen size.

Interface Element	Description
≫ ◇ Ⅲ Ҳ Ѧ थ ҡ	Quick layout switchover. From left to right: Force directed, Circle, Grid, Radial-tree, Hierarchical, CoSE, and Double-core. Figure Force directed shows how the graph looks on the canvas. NOTE The Double-core takes effect only when two nodes are selected.
Legend were color: Size: Color: Size: Color:	Click a vertex to select the color and size, which is a good way to mark data.
7id 78 label user occupation academic/educator gender M Zip-code 85718 userid 32 age 56+	Vertex details. Move the cursor to a non- virtualized vertex. The ID, label, and properties of this vertex are displayed. NOTE A maximum of six properties of a vertex can be displayed in the pop-up window. When the number of properties is greater than six, you can view all of them in the filter and property tab as shown in Editor page.
Shortcut operations in the drawing area	Box-select: Shift + Left-click and drag All vertices in the box are selected, as illustrated in the following figure.

Interface Element	Description
	Multi-select: Ctrl + Left-click and drag All vertices in the box are selected and highlighted, as illustrated in the following figure.
	Select/Deselect: Ctrl + Left-click Press Ctrl and left-click a vertex or an edge to select and highlight it. Press Ctrl and left-click the vertex or edge again to deselect it.
	Select all: Ctrl + A Select and highlight all vertices and edges.
	Select associated vertices and edges: Ctrl + E Select a vertex and press Ctrl + E to highlight all vertices and edges associated with it.
	Hide: Delete Quickly hide a vertex or an edge.
	Adaptation: Ctrl + F Automatically zoom in or out all vertices and edges based on the current screen width and height.
	Zoom out: - Press the - key on the keyboard to zoom out the graph.
	Zoom in: = (+) Press the + key on the keyboard to zoom in the graph.
	Deselect: Esc Deselect all selected and highlighted vertices and edges.

Interface Element	Description
	Zoom in and zoom out: Scroll the mouse wheel forwards and backwards.
	Scroll the mouse wheel to zoom in or out the graph.

Figure 8-12 Force directed



Figure 8-13 Circle



Figure 8-14 Grid





Figure 8-15 Radial-tree

Figure 8-16 Hierarchical



Figure 8-17 CoSE



Figure 8-18 Double-core



8.3 Accessing the Graph Editor

You can use the graph editor to query and analyze graphs. It has extensive built-in algorithms for customers to use in different scenarios of different fields. In addition, it is compatible with the Gremlin and Cypher query languages and supports open APIs. GES is easy to use even for zero-based users.

The procedure is as follows:

- 1. Log in to the GES management console and choose **Graph Management** from the navigation pane on the left.
- 2. On the **Graph Management** page, select the graph to be accessed and click **Access** in the **Operation** column.

Figure 8-19 shows the graph editor page. You can analyze the graph data on the graph editor. For details, see **Graph Editor**.

Ţ	enTho	ousan	d_Charging				bolated Vertices	Neighbor vertices	Undo Redo	Alt data	← Center a verte	t ID or multiple IDs seperate . Q	rhene ∼ 📩 🖞 🖾 🖄	I Q Q HI 🌶 🔶	$\mathfrak{m} \times \mathbb{V} \overset{\otimes}{\otimes} \mathbb{H}$
916			perations Schema Alg	orithms Indexe	20730									Vertex from the shortcut Show	menu to add a vertex. Histo Sanstive Cotom
5.Aze				<u>a</u>)											
Grap				~						a Sugalan Sustan		. 🚨 🚊 🚊			
è.			PageRank	00								. 💂 🔎 👷			 novie
ouel Que			PersonalRank	๏ ⊙											
ŝ,				00								Lancarda Way Lan			
shot				00	4					• •	• • •	• • •			
Snep			Shorlest Path	00						• •		• • •			
			All Shortest Paths	00					•	• • •	• • •	• • •			
				00					•		• • •				
		e ji	Closeness Centrality	0 0					Nin 10	• 10°4 14°4					
			Label Propagation	00							Now tion the				
		12		⊚ ⊙								angen anna agus			100 / 146 Vertex 0 / 1659 Edge
			Unk Prediction	00											
			Node2vec	00											
			Real-time Recomme.		Running P	Record									
			Common Neighbors	⊚ ⊙	Q Select	a property or enter a									
		·		00	name 🖯										
			Connected Compone	10 Om	Cypher										
			Degree Correlation												

Figure 8-19 Graph editor

8.4 Dynamic Graphs

8.4.1 Timeline

If you want to view vertex and edge changes over time, a timeline is required to convert a static graph into a dynamic graph. This also allows you to get dynamic analysis result.

D NOTE

To use this function, you need to create a dynamic graph. For details, see **Creating a Dynamic Graph**.

Setting a Timeline

- 1. Log in to the GES console and choose **Graph Management** from the navigation pane on the left. On the displayed page, locate the dynamic graph and click **Access** in the **Operation** column.
- 2. On the displayed graph editor page, set the following parameters in the **Timeline Settings** dialog box:

NOTE

The parameters set here will be synchronized to those in **Community Evolution** and **Temporal BFS**.

- Start Time Property: Name of the start time property that is a property of the imported or created metadata. The default value is startTime. The name must be of the date, long, or int type.
- **Start**: Start time of the dynamic graph. The start time must be earlier than or equal to the end time.
- End Time Property: Name of the end time property that is a property of the imported or created metadata. The default value is endTime. The name must be of the date, long, or int type.
- End: End time of the dynamic graph.
- Advanced Settings: Use Default settings or Custom settings.
 - **Default**: Use the default settings.
 - **Custom**: Set the display duration of vertices and edges in the graph and the display priority of labels.
 - Vertex/Edge Display: How long the vertices and edges in an algorithm result will be displayed on the canvas. This function is supported for Temporal BFS only. The value must be a timestamp in seconds. The default value is 604800 (7 days).

This function is used to the returned vertex and edge data that contains the start time only.

 Label Display Priority: This parameter is available only for temporal graphs with multiple labels. For details about how to create such a graph, see Creating a Graph. You can select multiple labels. When two vertex labels have identical start and end times, the label on the left will be displayed first.

Figure 8-20 Setting a timeline

Timeline Settings		×
* Start Time Property ⑦		
* Start		
★ End Time Property ⑦		
* End		
Advanced Settings	Default Custom	
Vertex/Edge Display ⑦	604800	
Label Display Priority 🔇		
		ок

3. Click OK.

NOTE

If you want to modify the timeline parameters, click in the lower left corner of the canvas.

8.4.2 Community Evolution

The community evolution algorithm generates a dynamic graph that shows structure changes of a community over time. The procedure to use this algorithm is as follows:

- 1. Set parameters in the **Community Evolution** drop-down list in the **Temporal** tab of the **Graph Analysis** area on the left of the graph editor page.
 - Set the start time, end time, and their properties. For details see **Setting**

a Timeline. To modify the parameters, click in the lower left corner of the canvas.

- **Vertices**: IDs of vertices in the community. You can enter a maximum of 100,000 vertex IDs. Use commas (,) to separate them.

Figure 8-21 Community evolution



2. Click On the right of **Community Evolution**. The running result is displayed on the canvas.

Figure 8-22 Dynamic graph



UI Element	Description
٢	Start playback.
Forward	Playback direction of the dynamic graph. If you toggle on this switch, the playback will be forward. If you toggle off this switch, the playback will be backward.

UI Element	Description
Double slider	Whether the playback uses the double slider
	• Toggled on (by default): Two sliders are used for playback. The start and end sliders move forward or backward at the same time, and the length of the time window represented by the distance between the sliders remains unchanged.
	 Toggled off: Only the one slider is used for playback.
	 If the playback is forward, the start slider is fixed and end slider moves froward on the timeline.
	 If the playback is backward, the end slider is fixed and start slider moves backwards on the time line.
All data	Whether data displayed on the canvas contains static data. If you toggle on this switch, only dynamic data is displayed.
	static data refers to the data that does not change over time.
Numerals O	 Whether the timeline uses dates or timestamps. By default, this switch is toggled on, which means that you need to enter timestamps to specify the duration.
	 If you toggle this switch off, you enter dates and time to specify the duration.
Start End	Stin t time and end time of the duration you want to view graph data changes
<u>نې</u>	Timeline settings. For details about how to set the parameters, see Setting a Timeline .
1646092800 1648511999 Start: 2022-03-01 08:00:00 End: 2022-03-29 07:3	Step length : Length of each step that the slider ^{Ren} oves on the timeline
	Interval: Interval between two steps
	e e

8.4.3 Temporal BFS

Temporal breadth-first search (BFS) algorithm searches for associated vertices based on temporal message passing and temporal BFS algorithms, and outputs

the visit time of each vertex and the distance from the vertex to the source start vertex. The procedure to use this algorithm is as follows:

- 1. In the **Temporal** tab of the **Graph Analysis** area on the left of the graph editor page, click **Temporal BFS**, and set the parameters in the drop-down list.
 - Set the start time, end time, and their properties. For details see **Setting**

a Timeline. To modify the parameters, click in the lower left corner of the canvas.

- Start Vertex: ID of the start vertex
- k: Traversal depth, indicating the maximum number of vertices in a traversal. The value ranges from 1 to 100. The default value is 3.
- **Direction**: Whether the traversal is performed along the directions of edges in the graph. The value can be **true** (default value) or **false**.
 - **true**: Traversal is performed along edge directions.
 - **false**: Edge directions will not be considered in the traversal.

Figure 8-23 Temporal BFS

A 🚯 Temporal BFS ⑦ O	
* Start Vertex ⑦	
Enter a vertex ID.	
к (?)	
Direction ⑦	
true ~	
★ Start Time ⑦	
1646092800	
★ End Time ⑦	
1648511999	
★ Start Time Property ⑦	
startTime	
* End Time Property ⑦	
endTime	

2. Click On the right of **Temporal BFS**. The running result is displayed on the canvas. In this algorithm, a single slider is used for playback. As shown in **Figure 8-24** and **Figure 8-25**, the vertices in the dynamic graph are increases over time.

Figure 8-24 Execution result 1



Figure 8-25 Execution result 2



8.4.4 Temporal Paths

Temporal paths that start from a vertex to the target node show the trend of increment (or non-decrement) of vertices and edges over time on the canvas. The paths follow the order of information transmission on dynamic graphs, the passing time of an edge on a path must be later than or the same as that of the previous edge.

For this feature, you can use the **strategy** parameter to adjust whether the temporal path with the shortest distance or the temporal path that reaches the target node as early as possible is searched for. The procedure is as follows:

- 1. In the **Temporal Paths** tab of the **Graph Analysis** area on the left of the graph editor page, click **Temporal BFS**, and set the parameters in the drop-down list.
 - Set the start time, end time, and their properties. For details see Setting

a Timeline. To modify the parameters, click in the lower left corner of the canvas.

- **source**: ID of the start node
- **targets**: set of end node IDs. Multiple end node IDs can be configured.
- k: Traversal depth, indicating the maximum number of vertices in a traversal. The value ranges from 1 to 100. The default value is 3.
- strategy: execution strategy of the algorithm. The value can be shortest or foremost.
 - shortest: the temporal path with the shortest distance is returned
 - foremost: the temporal path that reaches the target node as early as possible is returned
- directed: Whether the traversal is performed along the directions of edges in the graph. The value can be true (default) or false.

- **true**: Traversal is performed along edge directions.
- **false**: Edge directions will not be considered in the traversal.

Figure 8-26 Temporal paths

🔺 🌐 Temporal Paths 🛛 🕐
* source ⑦
∗ targets ⑦
k (?)
strategy ⑦
shortest ~
directed ⑦
true 🗸
* Start Time ⑦
* End Time ⑦
* Start Time Property ⑦
* End Time Property ⑦
endTime

2. Click on the right of **Temporal Paths**. The execution results are displayed on the canvas. As shown in **Figure 8-27** and **Figure 8-28**, the vertices in the dynamic graph change over time.

Figure 8-27 Execution result 1



Figure 8-28 Execution result 2



8.5 Graph Exploration

Handful graph exploration tools facilitate your analysis.

D NOTE

Multi-label graphs do not support graph exploration.

Path Extension

Filters are added to query APIs to search for the desired k-hop vertices or edges. For details about APIs for filtered queries, see **Filtered Query V2**.

In the **Path Extension** area on the left of the GES graph editor, set the following parameters:

- **Start Vertex**: IDs of start vertices. You can use any of the following methods to query the vertices:
 - a. Press and hold **Shift** and drag a rectangle using the left mouse button to select desired vertices, right-click a vertex, and choose **Set as Path Start** from the shortcut menu. The **Path Extension** will be displayed. The IDs of the selected vertices are automatically filled in the **Start Vertex** box. In this box, you can add or delete vertex IDs. After you finish selecting, click

💽. The query result is displayed on the canvas.

Explore Operations Schema Algorithms		Double-clic
General ~		
🗠 😥 Path Extension 🛛 💿	Place00009	
★ Start vertex ⑦	Place00006	Place00003
Person00051 × Random	View Property	The stress of th
Search Criteria ⑦	Place00043 Search by Association	Place00016
	Export	
0 5 10 15 20	Search by Path	Person000
bothV None	Shortest Path of the Verte	ex Sets
Advanced Settings	Place00007 Common Neighbors of the	e Vertex Sets
	Sub Graph	
	Add Edge	Alt + A
	Place Hide	
	Set as Path Start	
		A

Figure 8-29 Selecting start vertices

b. Random selection: Click **Random** next to the start vertex box. The system automatically selects vertices in the graph and enters vertex IDs. You can

add or delete vertex IDs in the box. After you finish selecting, click . The query result is displayed on the canvas.

- c. Specifying one start vertex: Enter the ID of a vertex in the text box and press **Enter**.
- d. Specifying a batch of start vertices: Enter IDs of desired vertices in the text box and separate them with commas (,). Then, press **Enter**. A window is displayed when you enter many vertex IDs so you can view them clearly.

D NOTE

Do not enter the same vertex ID repeatedly or an empty value. If the entered vertex ID name contains commas (,), replace the commas with ",".



• **Search Criteria**: Each row in the list corresponds to a query type and criterion of each hop. If there are more hops than criteria, the criteria will be repeated.

Figure 8-30 Search criteria



Refer to the following description to set the search criteria:

- Hop count: Number of search criteria.
- Search criterion: Each hop has a search criterion. Click a search statement text box. The Search Settings window is displayed. Enter a search statement.

The following search criteria operators are available:

has: A property key or the value of a property key must be contained.

hasLabel: The label value must be one of the specified values.

and: Conditions A and B (can be nested) must be met.

or: Either condition A or B (can be nested) must be met.

Figure 8-31 Search settings

Search Setting	s ⑦	
bothV 🗸 Ent	er a vertex criterion, for example, has(key, value).	
bothV 🗸 Ent	er a vertex criterion, for example, has(key, value).	
bothV 🗸 Ent	er a vertex criterion, for example, has(key, value).	
	Cancel	ОК

NOTE

1. To view a sample criterion, double-click a blank text box. Regular search statements are as follows:

has(PropertyName): Search for a vertex that has PropertyName.

has(PropertyName, PropertyValue): Search for a vertex that has a property whose name is **PropertyValue**.

hasLabel(LabelName1,LabelName2): Search for a vertex that has a label
whose value is LabelName1 or LabelName2

or(has('name', 'peter'), has('age', '30')): Search for a vertex whose name is Peter or age is 30.

and(has('person'),or(has('name','peter'),has('age','30')): Search for a vertex whose name is peter and age is 30.

- 2. If there is only one search criterion, the delete, up, and down buttons are grayed out. The first criterion cannot be upshifted, and the last criterion cannot be downshifted. The maximum number of search criteria is 20 (that is, the maximum number of hops).
- **Show path process**: Whether the vertices that are not on the final path will be displayed. This is disabled by default.
- Advanced Settings: You can set the expansion strategy here.

Currently the following traversal methods are available for graph expansion:

- ShortestPath: This method traverses all the shortest paths from the start vertex to every vertex in the graph. This effectively suppresses the exponential growth of the query volume in multi-hop queries.
- Walk: Duplicate vertices are not filtered during traversal.

NOTE



As shown in the figure, the third-hop neighbor of vertex **a** is queried.

If you use the walk method, the paths are: **a->c->a->b**, **a->c->d->f**, **a->c->d->c**, and **a->c->a->c**.

Vertices **a** and **c** appear repeatedly in the paths such as **a->c->a->b** and **a->c->d->c**. Using **ShortestPath** can reduce duplicate paths, speed up the query process, and reduce the number of queries in this process.

For **ShortestPath**, the query process generates the **a->c->d->f** path only.

8.6 Multi-Graph Management (Database Edition)

When you create a database graph, it is automatically upgraded to a multi-graph cluster. This cluster can have multiple graph instances, each allocated with different data. This allows you to analyze multiple graphs simultaneously.
In the graph engine editor, you can manage the graph instances in the graph cluster by clicking the dropdown menu next to the cluster name in the upper left corner of the page to switch between graph instances.

Figure 8-32 Multi-graph management



Adding or Deleting a Graph

- 1. After the database graph cluster is created, the graph engine editor page is displayed. For details, see Accessing the Graph Editor.
- 2. In the upper left corner of the page, click **Add Graph**. In the displayed dialog box, enter the graph name, vertex ID type, and SortKey type.

Figure 8-33 Adding a graph

add graph		×
* Graph Name		
* ID type	String (fixed length) Fixed-length string: Vertex IDs are stored internally and used for compute. If the IDs are to long, the query performance can be reduced. Specify the length limit based on your datase vertex IDs. If you cannot determine the maximum length, set the ID type to Hash.	o et
SortKey Type 🕜	Integer Integer Cancel OK	

- Vertex ID Type: The options include String (fixed length), String (variable length), and Hash.
 - String (fixed length): Vertex IDs are used for internal storage and compute. Specify the length limit. If the IDs are too long, the query performance can be reduced. Specify the length limit based on your dataset vertex IDs. If you cannot determine the maximum length, set the ID type to Hash. If you select String (fixed length), you also need to enter the vertex ID length.
 - String (variable length): The length of the vertex IDs written by the user is not limited. However, if the IDs are too long, the read and write performance is affected. It is recommended that the length be within 1,000 bytes, with a maximum of 4,000 bytes.
 - Hash: Vertex IDs are converted into hash code for storage and compute. There is no limit on the ID length. However, there is an extremely low probability, approximately 10⁽⁻⁴³⁾, that the vertex IDs will conflict.

NOTE

If you cannot determine the maximum length of a vertex ID, set this parameter to **Hash**.

- SortKey Type: SortKey type. Different SortKey values are configured to distinguish duplicate edges (edges with the same source vertex, end vertex, and label). The options include:
 - Integer: The value is an integer.
 - String (byte length less than or equal to 40): Importing a SortKey greater than 40 bytes will result in an error.
 - String (variable length): The length is not limited. However, if the IDs are too long, the read and write performance is affected. It is recommended that the length be within 1,000 bytes, with a maximum of 2,000 bytes.
- 3. After setting the parameters, click **OK**.
- 4. To delete a graph instance, click **Delete Graph**.

8.7 HyG Graph Management (Database Edition)

Create a HyG graph in the GES editor and import data into the graph.

NOTE

- Only graphs of version 2.4.2 or later support this function.
- When creating a graph, set the product type to database edition and enable the HyG computing engine. For details, see **Creating a Custom Graph**.

Creating a HyG Graph

- 1. After the database graph cluster is created, the graph engine editor page is displayed. For details, see Accessing the Graph Editor.
- 2. On the HyG tab, click Create HyG graph.

Figure 8-34 Creating a HyG graph



- 3. In the displayed dialog box, set **Policy** (only **oec** currently available) and **Include Incoming Edge**, and click **OK**.
 - **Policy**: graph splitting policy. **oec** (out edge cut) indicates outgoing edge splitting. Retain the default value.
 - **Include Incoming Edge**: whether the graph contains incoming edges. If set to **Yes**, the data synchronization performance will be affected.

Figure 8-35 Setting parameters

Create HyG graph		
Policy	oec	
include the inEdge	● Yes ○ No	
		Cancel OK

- 4. Once the creation is successful, you can import or synchronize data.
 - Data import: You can import new vertex and edge data. For details, see **Importing Data**.
 - Data synchronization: Synchronize your existing vertex and edge data in the graph database to the compute engine. For details, see Synchronizing Data.

	Delete HyG	graph
HyG graph		
Import Data	Data sync	
	oec	
	true	
	ertex ID true	

Figure 8-36 Importing data

5. To delete a HyG graph, click **Delete HyG Graph**. In the displayed dialog box, enter **DELETE** and click **OK**.

Figure 8-37 Deleting a HyG graph

Delete HyG graph	
Delete the graphs?	
To confirm deletion, enter "DELETE" below. Auto Enter	
DELETE	
Cancel OK	

Importing Data

- 1. Click Import Data. In the displayed dialog box, set the following parameters:
 - AccessKey: user's access key ID
 - SecretKey: secret key used together with the access key ID
 - Vertex Dataset: vertex file directory or vertex file name. CSV and TXT files can be imported.
 - **Edge Dataset**: edge file directory or edge file name. CSV and TXT files can be imported.
 - Metadata: OBS path of the metadata file of the new data
 - Log Storage Path: directory for storing graph import logs, used to store failed data imports and detailed error causes
 - Field Delimiter: field delimiter in a CSV file. The default value is comma (,).
 - Field Enclosure Symbol: field enclosure symbol in a CSV file, which is used to enclose a field, such as when the field contains a delimiter or line break. The default value is double quotes (").
 - Vertex Properties: List of vertex properties. The specified properties must belong to the schema file. If the list is empty, vertex properties will not be imported.
 - Edge Properties: List of edge properties. The specified properties must belong to the schema file. If the list is empty, edge properties will not be imported.

Figure 8-38 Importing data

Import Data Into HyG Gra	ph ×
AccessKey	
SecretKey	<u> </u>
Vertex Dataset	B
	Verlex file directory or name. The files must be in CSV or TXT format.
Edge Dataset	e
	Edge file directory or name. The files must be in CSV or TXT format.
Metadata	Q
(Optional) Log Storage Path	e
	Directory where logs of imported graphs are stored. It is used to store data that to import and the detailed causes of errors.
(Optional) Field Delimiter	,
	By default, the delimiter is a comma (,).
(Optional) Field Enclosure Symbol	(•
	By default, the enclosure symbol is double quotes ("), which is used to enclose that contain delimiters or line breaks.
(Optional) Vertex Properties	⊕ Add
	The property to add must belong to the properties in the schema file. If the pro text box above is empty, the vertex property will not be added.
(Optional) Edge Properties	⊕ Add
	The property to add must belong to the properties in the schema file. If the pro text box above is empty, the edge property will not be added.
	Cancel OK

2. Click **OK**. The imported data is displayed in the HyG graph details.

Figure 8-39 HyG graph details



Data Synchronization

1. Click **Synchronize Data**. In the displayed dialog box, specify the vertex and edge properties.

During the initial data synchronization, the vertex and edge parameters will be applied. For subsequent synchronizations, these parameters will default to the values specified during the first synchronization.

Figure 8-40 Data synchronization

Data sync	×
Vertex Properties	
Edge Properties	
	Cancel OK

2. Click **OK** to synchronize data. Once the synchronization is complete, the data is displayed in the HyG graph details.

Figure 8-41 HyG graph details



8.8 Adding Custom Operations

You can add custom operations executed by calling APIs. You can create shortcut operation sets.

Procedure

1. In the **Operations** tab on the left of the graph editor, click **Edit** . The **Add Operation** button is displayed.



Figure 8-42 Adding an operation

- 2. Click **Add Operation** and set the following parameters in the displayed dialog box:
 - **Name**: Enter a name for the custom operation.
 - API Type: cypher, gremlin, algorithm, and path_query are supported.
 - **Request Body**: Enter the request body for the calling the API.
 - **Description**: Add a description for the operation.

Figure 8-43 Adding a custom operation

Add Operation		×
* Name	Fraud ring statistics for the past w	
★ API Type	gremlin ~	
* Request Body	{ "command":"g.V().limit(1)"]	
* Description	Calculates potential fraud rings w Cancel	ок

- 3. Click **OK**. These parameters cannot be changed after the operation is added.
- 4. The new custom operation is displayed in the **Operations** tab. You can click the run button to execute the operation and view the results on the canvas.

Figure 8-44 Custom operations

Explore	Operations	Schema	Algorithms
All	~	۵	Edit
Private			
Fraud	ring statistics f	for the pa	
Calculat	tes potential frau	d rings wit	€

5. To delete the operation, click **Edit**. Then click isplayed in the upper right corner of the operation.

8.9 Editing Schema

In the metadata analysis area of the graph editor, you can perform the following operations:

- 1. Adding a Label
- 2. Counting Vertices and Edges
- 3. Modifying a Label
- 4. Hiding a Label
- 5. Importing and Exporting Labels
- 6. Deleting a Label

Adding a Label

In the metadata list on the left of the graph editor, click 📩 to add a label.

- Label Name: name of the label to be added.
- **Type**: You can select a label type (vertex, edge, or general-purpose). General-purpose indicates that a label can represent either a vertex or an edge.
- **Custom vertex style**: You can define the color and mark of a label to distinguish vertices.
- Add properties. By default, only the first added property is displayed on the canvas. You can manually adjust the property to be displayed. The canvas will respond in real time.

Figure 8-45 Adding a label



Counting Vertices and Edges

On the **Schema** tab of the graph editor, click **Refresh Vertex and Edge Count**. The system counts the total number of vertices and edges in the current graph. You can also view the last count time.

Explore Operations Schema Algorithms Export Import + Enter a label. Q Last Counted: 2024-03-19 11:00:00 Refresh Vertex and Edge Count • NOWS • • Refresh Vertex and Edge Count • NOWS • • Refresh Vertex and Edge Count • NOWS • • Refresh Vertex and Edge Count • NOWS • • Refresh Vertex and Edge Count • NOWS • • Refresh Vertex and Edge Count • Nows • • Post • • Post • • Post • • Person • • Person • • Person • • Person • • Person

Figure 8-46 Counting vertices and edges

Modifying a Label

NOTE

This function is only available on graph version 2.3.18 or later.

In the metadata file list, click the metadata file for which you want to modify the label. The metadata label details page is displayed.

- You can modify the label's property name, cardinality, and data type.
- To hide or delete a property, click the hide or remove button in the **Operation** column.
- If you accidentally deleted or incorrectly modified a property, click the reset button to restore to the last saved data.

Confirm the modification and click **Save**.

Figure 8-47 Modifying a label

Explore Operati	ions Schema Algoi	ithms Indexes					
			Label - All - mo	vie 🛛 🗞			×
Export							
+ Enter a la			 Custom vertex : 	style (for vertices with this label)			
Last Counted: 202	4-12-24 15:01:16		Property Name	Cardinality	Data Type	Operation	
Refresh Vertex			ChineseTitle	Single value V		99	
		© 🗇		Single value 🗸 🗸			
				Muttinla values			
			Genres	Allow duplicate values.		0 Q	
			Add a property				
				Save	Clear Cancel		
		© Û					

Hiding a Label

• Hide all vertices and edges of a label.

In the metadata list on the left of the graph editor, click the eye button next to metadata to hide all vertices and edges of the metadata in the analysis result.

Figure 8-48 Hiding a label



• Hiding a label: hide the current label on the canvas.

On the **Schema** tab of the graph editor, click the metadata file you want to edit. On the label details page that appears, click a next to the label to hide it from the canvas.

Figure 8-49 Hiding a label



• Hide the vertices and edges of a selected label

On the canvas, click any vertex in the graph. The selected vertex is displayed



- is a label-based hide button. You can click this button next to a label to hide the vertices and edges of the selected label. That is, these vertices and edges are not displayed on the canvas.
- **I** is a label-based display button. You can click the button to display the vertices and properties of the label.

Importing and Exporting Labels

You can import the metadata, edge data, and vertex data of a graph to or export them from an OBS bucket.

- Import: Click Import in the metadata list. In the dialog box that is displayed, set Metadata, Edge Data, Vertex Data, Log Storage Path, Edge Processing, and Import Type, and click OK to import the data from the OBS bucket to a graph.
 - Log Storage Path: Stores vertex and edge data sets that do not comply with the metadata definition, as well as detailed logs generated during graph import.
 - Edge Processing: Includes Allow repetitive edges, Ignore subsequent repetitive edges, Overwrite previous repetitive edges, and Ignore labels on repetitive edges. Repetitive edges have the same source vertex and target vertex. When labels are considered, repetitive edges must have the same source and target vertices and the same labels.

Figure 8-50 Importing metadata

Import		
Metadata		
Edge Data		
Vertex Data		
Log Storage Path		
★ Edge Processing ⑦	 Allow repetitive edges ⑦ Ignore subsequent repetitive edges ⑦ Overwrite previous repetitive edges ⑦ Ignore labels on repetitive edges ⑦ 	
Import Type	Online import The import speed is slower, but the graph can be read (cannot be written). Offline import The import speed is higher, but the graph cannot be read or written.	
	Cancel	ок

• Export: Click **Export** in the metadata list. In the dialog box that is displayed, set **Metadata Name**, **Vertex Data Set**, **Edge Data Set**, and **Export Path**, and click **OK** to export the data to the OBS bucket.

Export If you choose to expo files in a text editor. If be parsed into formul 1. Do not select Enab 2. Do not select Enab	ort CSV files to your le f the data contains sp las by the software. T le Dynamic Data Excl le or Yes if a dialog b	ocal host, the files are opened usin ecial characters such as plus sign o ensure system security, pay atte hange Server Launch (not recomm ox indicating a security issue is dis	g the spreadsheet software by defa s (+), minus signs (-), equal signs (ntion to the following when openin ended). splayed.	ault. You are advised to open the -), and at signs (@), the data will g such files:	
★ Vertex Data Set	ges_vertex_1710404	1815138			
* Edge Data Set	ges_edge_1710404	315143			
* Metadata Name	ges_schema_17104	04815144			
* Export Path (?)					
Name		Modified	Туре	Size	
Cab8d6	a-90e3-4679-915				
🗎 3571fe9f	15dc415d99ade18				
🖿 aa-test-lit					
🗎 aaaa-c-te					
🗖 aaaa-c-te	estaaa				
5 🗸 Total Re					

Figure 8-51 Exporting metadata

Deleting a Label

NOTE

- 1. After this API is called, all data associated with the label will be deleted. Exercise caution when performing this operation.
- 2. If the graph version is earlier than 2.2.18, schema labels cannot be deleted.
- 3. Schema labels cannot be deleted from graphs of the database edition.
- 4. The default label **_DEFAULT**_ cannot be deleted.



To delete a label, do the following:

1. To delete a label, click the deletion icon next to the schema on the **Schema** tab on the left of the graph engine editor.

Figure 8-52 Deleting a label



2. In the dialog box that is displayed, read the message carefully, confirm the name of the label to be deleted, enter **DELETE** in the text box, and click **OK**.

Figure 8-53 Confirming the deletion



3. During the deletion, the result of deleting the label algorithm is displayed in the result display pane below the canvas.

Figure 8-54 Results display



During the deletion, the **Filter** button on the **Filtering** tab is grayed out and becomes unavailable.

8.10 Hiding Sensitive Information About a Graph

You can control whether to show sensitive information about a graph.

Procedure

- 1. Log in to the graph editor. For details, see Accessing the Graph Editor.
- 2. Hide all sensitive information:
 - a. In the upper right corner of the canvas, click an ext to **Show/Hide Sensitive Data** to hide sensitive information.

Figure 8-55	Before	hiding
-------------	--------	--------



Figure 8-56 After hiding



b. When sensitive information is hidden, the status of the visibility icon in the **Operation** column for each property in the **Schema** tab's editing area matches the status of an the canvas.

Figure 8-57 Label and property hiding



3. To show sensitive information about a single label after hiding it, do the following:

In the upper part of the **Schema** tab's editing area, click the visibility icon on the right to unhide the label on the canvas, and then click the visibility icon on the left to show sensitive information about the label.

Note: If you only click the visibility icon on the left and not on the right, like

sensitive information about the property will not be shown.

Figure 8-58 Showing sensitive information about a label

Label - All - mo	2 1 ovie ⊙ ⊙			×
 Custom vertex 	style (for vertices with this label)			
Property Name	Cardinality	Data Type	Operation	

4. To show sensitive information about a property under a label after hiding it, do the following:

In the **Operation** column of a property in the **Schema** tab's editing area, click the visibility icon on the right to cancel the hiding of the property on the canvas, and then click the visibility icon on the left to show sensitive information about the property. Once the settings are complete, click **Save**.

Note: If you only click the visibility icon on the left and not on the right, like sensitive information about the property will not be shown.

Figure 8-59 Showing sensitive information about a single property

	Label - All - movie 💩 🕸						
	✓ Custom vertex :	style (for vertices with this label)					
	Property Name	Cardinality	Data Type	Operatio	n		
	movieid	Single value V	int v	4 1 @@.			
•	title	Single value V	string ~	ଡ଼ <i>ଭ ଭ</i>			
	genres	Single value V	string ~	(? ଜ୍ଜ୍ ୧			
	Add a property						
		Save	Clear Cancel				

8.11 Visual Query

In the graph editor, you can create graph query statements by dragging and dropping vertices and edges, and preview the query results without writing any code.

Procedure

1. In the left pane of the graph editor, click the **Visual Query** tab.

Vertex Pattern
Filter

Image: Constraint of the stop

Image: Constraint of the st_vertex

Figure 8-60 Visual query

- 2. Add a vertex to the canvas.
 - a. In the **Add Vertex Pattern** tab, all vertex labels and edge labels of the graph are displayed. Each label is displayed as a card that can be dragged to the canvas. Select a vertex label and drag it to the canvas.

The Cypher query statement below changes with your operations.

D NOTE

These vertex labels and edge labels are the same as those in the metadata list in **Editing Schema**.

b. Drag the labels you want to use for the query to the canvas and click **Execute Query**. The graph result is displayed on the right of the canvas.

You can view the running records of the Cypher query statement in the **Running Record** tab below the canvas. Click **Query Result** to view the result.





D NOTE

Query results can be displayed only when there is only one submap pattern on the canvas. If there are multiple disconnected submaps or isolated vertices, you must add edges to connect the submaps or isolated vertices. You can also set multiple labels to reconstruct your query mode. Otherwise, when you click **Query**, the system displays a message indicating that there are multiple submap patterns.



				~	
8	There is more than one subgra subgraphs or isolated vertices.	ph pattern. Con	nect the	×	ery
	P	Place			
	•				
P	erson				

3. Add a vertex filter.

Click a vertex in the canvas. The **Filter** tab page is displayed in the left pane. On the **Filter** tab, specify labels, vertex ID, and property search criteria to search for the vertex labels you want to view on the canvas.

Vertex Pattern	Filter
Vertex v1	
Label	
route × user ×	
Vertex ID	
Constraints	
Prop V Oper	Value Ū
•	
Del	ete

Figure 8-63 Adding a vertex search criterion

- Vertex V1: Cypher variable ID (vertex identifier in the Cypher query statement below the canvas), which is named based on the sequence in which vertices are dragged to the canvas, for example, V1, V2, and more alike.
- **Label**: Set one or more labels to search for target vertices. The logical operator between each two labels is OR.
- Vertex ID: It is equivalent to a filter criterion. After adding a vertex ID to a vertex label, you can click **Query** to query the vertex labels with the same vertex ID.
- Constraints: Specify a property contained in the vertex label. Currently, a property with multiple values is not supported.
 - **Property**: Property contained in the label.
 - Operator: Comparison operators (>,>=,<,<=,=,<>), null judgment operators (is null, is not null), and string comparison operators (starts with, ends with, contains) are supported.

starts with searches for a property that starts with a specified string; **ends with** searches for a property that ends with a specified string; contains searches for a property that contains a specified string.

- Value: Property value. The attribute value type must be the same as that in the metadata. If the attribute value is of the character type, you need to use single quotation marks (").
- Delete the constraint.
- + button: Add a criterion.
- **Delete**: Delete the added criterion.

Click **Execute Query** in the canvas again. The query result is displayed on the right of the canvas.

4. Add an edge (connect two vertices on the canvas):

Double-click a vertex. After the border of the vertex turns gray (do not move the cursor out of the gray border), click and drag a line from the vertex to another vertex.

The Cypher query statement below changes with your operations.

Figure 8-64 Gray border of a vertex



Figure 8-65 Adding an edge



5. Add an edge filter.

Click an edge in the canvas. The **Filter** tab page is displayed in the left pane. On the **Filter** tab, specify labels, direction, hops, and property search criteria to search for the edge labels you want to view on the canvas.

Vertex Pattern	Filter
Edge e1	
Label	
Single click to select the	label 🗸
Direction Directed	•
Alt	ter
Hops	
1 (default), Range input:	minHopsmaxHops
Constraints	
Prop V Oper	Value Ū
Del	ete

Figure 8-66 Adding an edge filter

- **Edge e2**: Cypher variable ID, which is named based on the sequence in which edges are added to the canvas, for example, e1, e2, and more alike.
- **Label**: Set one or more labels to search for target edges. The logical operator between each two labels is OR.
- Direction: Select the direction contained in the edge label.
 When the slider is toggled on, the edge is a directed one. When the slider is toggled off, the edge is undirected (or called bidirectional).

If the edge is directed, the arrow on the canvas indicates the direction of the edge. You can click **Change Direction** to change the direction of the selected edge on the canvas.

- **Hops**: The default value is **1**. The value range is [0, 20). You can specify a number or a range.
 - If you enter an integer, it will be used as the number of hops in the edge pattern.

- If you enter two integers in the format of *minHops..maxHops*, for example, 2..3, the number of hops in the edge pattern is within the range of [2,3].
- Constraints: Specify a property contained in the edge label. Currently, a
 property with multiple values is not supported.
 - Property: Property contained in the label.
 - Operator: Comparison operators (>,>=,<,<=,=,<>), null judgment operators (is null, is not null), and string comparison operators (starts with, ends with, contains) are supported.

NOTE

starts with searches for a property that starts with a specified string; **ends with** searches for a property that ends with a specified string; contains searches for a property that contains a specified string.

- Value: Property value. The attribute value type must be the same as that in the metadata. If the attribute value is of the character type, you need to use single quotation marks (").
- **Ū**: Delete the constraint.
- + button: Add a criterion.

If there is more than one criterion, click and next to AND to set the logical operator (AND or OR).



Figure 8-67 Selecting a logical operator

D NOTE

The priority of AND is higher than OR. The suggested calculation sequence is as follows:

- 1. Arrange all AND operations first.
- 2. Then, perform all OR operations.

In the following example, the edge search criterion is **userid** < **100 AND gender** = 'male' OR userid > 50 AND age = '18-24'.

The operation sequence is:

(userid < 100 AND gender = 'male') and (userid > 50 AND age = '18-24') are operated first, and result1 and result2 are recorded respectively.

Then, **result1 OR result2** is operated.

Constrai	ints				
userid				100	Ū
		AND			
gender				'male'	ট
		OR			
userid				50	Ū
		AND			
age				'18-24'	Ū
		Dele	te		

Delete: Delete the added criterion.

Click **Execute Query** in the canvas again. The query result is displayed on the right of the canvas.

8.12 Canvas Snapshot

In the graph editor, you can use the snapshot feature to create a snapshot for a graph displayed on the canvas and then restore it from the snapshot.

Creating a Snapshot

- 1. In the graph editor, click in the upper right corner of the canvas to create a snapshot for the graph displayed on the canvas.
- 2. Once the snapshot is created, the system will show a message as depicted in the figure below:

Figure 8-68 Snapshot generated



D NOTE

- 1. Since the data is stored in the browser cache, switching browsers will result in the snapshot data being cleared.
- 2. The more snapshots you generate, the more likely your browser is to lag, so please use snapshots in moderation.

Checking a Snapshot

1. On the left of the graph editor, click the **Snapshot** tab. The snapshot information is displayed.

Figure 8-69 Snapshot page



- **Thumbnail**: snapshot thumbnail. When you hover over the thumbnail, the snapshot is automatically zoomed in.
- Name/ID: snapshot name and ID. The name can be changed, but the ID cannot. The system generates the ID to differentiate between saved files when importing the snapshot. If files with the same ID are imported, they will be overwritten.
- Graph Information: displays the graph data saved in the current snapshot.
- **Theme Color**: theme color of the canvas when the snapshot is saved.
- **Created** and **Modified**: creation time and modification time of the snapshot, respectively.
- The **Operation** column offers the following functions:
 - View: Check the snapshot on the canvas. If the current theme color does not match that when the snapshot is saved, the system will display a prompt message. Clicking Yes will switch to the theme used when the snapshot was saved, while clicking No will keep using the current canvas theme color.

Figure 8-70 Theme change



 Delete: Delete the snapshot. Confirm the deletion information, enter DELETE in the field box (or click Auto Enter), and click OK.

Figure 8-71 Deleting a snapshot

Delete Snaps	shot					
Selected for deletion	c 1					
Snapshot 🖨		ID 🖨	l	Graph 😝		
ges	_index-202	54s0,		ges_:	_test_index	
To confirm deletion, DELETE	enter "DELETE" I	pelow. Auto Enter				
					01	10

- More: includes Download and Download Thumbnail.
 - 1) **Download**: Download the snapshot as a JSON file and save it to the local host.
 - 2) **Download Thumbnail**: Download the snapshot as an image and save it to the local host.

NOTE

Deleting a graph will also delete the stored snapshots. Download the snapshots before deleting the graph.

Importing a Snapshot

Import downloaded snapshots one by one.

1. In the upper left corner of the snapshot page, click **Import**. The **Import Snapshot** page is displayed on the right.

Figure 8-72 Importing a snapshot

_						Import Snapshot
•	1. Snapshot data is stor	ed in the browser and will not affect database data	2. Clearing the browser cache will delete saved co	ervas data. You are advised to download the dat	a before clearing the cache to avoid data loss. 3. D	You are advised to import downloaded snapshot files and avoid
Do		Import				changing fields to prevent import failures caused by field changes
(Q :						
	Thumbnail	Namet0 0	Graph Information	Theme Color @	Created 😜	
-		ges_2ht_baiwan_lest_index-2024-05-15 17 ubmr3/hm11g56842xjami2h875el	Graph Name ges_zH_balwon_lest_index Oraph ID 0x508225-378a-45229ax534852155 Vertices:590 Edges:1611			
Tel						

2. Click Select File, select the snapshot's JSON file, and click OK.

D NOTE

When importing a JSON file with the same snapshot ID, a prompt message similar to the one shown in **Figure 8-73** will appear. You will need to determine whether or not to overwrite the existing data.

Figure 8-73 Snapshot prompt



Batch Downloading or Deleting Snapshots

Download or delete the created snapshots in batches.

Batch download:

On the snapshot page, select multiple snapshots and click **Download** in the upper left corner.

Figure 8-74 Batch download

-							
1	0 1. Snapshot data is st	ored in the brawser and will not affect database dat	a. 2. Clearing the browser cache will delete saved	canvas data. You are advised to download the dat	ta before clearing the coche to avoid data loss. 3. D	eleting a graph will also delete the stored snapshots	. Download the snapshats before deleting the graph.
C	Download Delate						
ſ							
	🗹 Thursbool						
			Cruph F Graph F Writous Edges: f				
			Graph N				

- Batch deletion:
 - a. On the snapshot page, select multiple snapshots and click **Delete** in the upper left corner.
 - b. In the displayed dialog box, confirm the deletion information, enter **DELETE** (or click **Auto Enter**), and click **OK**.

Figure 8-75 Batch deletion

Delete Snapshot			
Selected for deletion: 2			
Snapshot 🖨	ID 🔶	Graph 🖨	•
ges_;index-20	2 abnm	875ell ges_:	L_index
ges_z index-20	2 ttotsn	id4rg ges_	_index
To confirm deletion, enter "DELE	TE" below. Auto Enter		
			Cancel OK

8.13 Gremlin Query

Gremlin is a graph traversal language in the open source graph calculation framework of Apache TinkerPop. You can use Gremlin to query, modify, and traverse graph data as well as filter properties.

The procedure is as follows:

- 1. Log in to the GES graph editor. For details, see Accessing the Graph Editor.
- 2. In the graph data query area, click the drop-up button to choose **Gremlin**. Enter a query statement and press **Enter** to run the statement.

Figure 8-76	Switching	to Gremlin	query
-------------	-----------	------------	-------

Running Reco	Query Result	
Q Select a pro	perty or enter a keyword.	
Cypher	status ⊜	required parameter 😝
Gremlin		
DSL		
Gremlin ^		

D NOTE

Multi-label graphs do not support Gremlin query.

Gremlin Statement

Typical query commands are as follows:

• Querying vertices

g.V().limit(100): This command is used to query all vertices and return only 100 vertices. You can also use the **range (x, y)** operator to obtain vertices within the specified quantity.

g.V().hasLabel('movie'): This command is used to query vertices whose label value is **movie**.

g.V('11'): This command is used to query the vertex whose ID is 11.

D NOTE

- 1. The **g.V** () is not recommended because the query result cannot be completely displayed if the vertex scale is large.
- 2. To prevent query timeout due to a large data volume, add the **limit** parameter and set it less than **1,000**.
- Querying edges

g.E(): This command is used to query all edges. You are not advised using this command without filter criteria or limit to the returned results.

g.E('55-81-5'): This command queries the edge whose ID is 55-81-5.

g.E().hasLabel('rate'): This command queries edges whose label value is **rate**.

g.V('46').outE('rate'): This command queries the edge whose ID is **46** and all its labels are **rate**.

• Querying properties

g.V().limit(3).valueMap(): This command is used to query all properties of a vertex. (You can specify a parameter to query only one vertex. All properties of the vertex will be displayed in one row.)

g.V().limit(1).label(): This command is used to query the label of a vertex.

g.V().limit(10).values('userid'): This command is used to query the **name** property of a vertex. (You can leave the parameter blank to query all properties. Each property value is displayed in one row, without the key).

• Adding a vertex

g.addV('user').property(id,'600').property('age','18-24'): This command adds a vertex whose label is **user**, ID is **600**, and age ranges from **18** to **24**.

• Deleting a vertex

g.V('600').drop(): This command deletes the vertex whose ID is 600.

• Adding an edge

g.addV('user').property(id,'501').property('age','18-24')

g.addV('movie').property(id,'502').property('title','love')

g.addE('rate').property('Rating', '4').from(V('501')).to(V('502'))

The preceding commands add two vertices and an edge. The two vertex IDs are 501 and 502.

• Deleting an edge

g.E('501-502-0').drop(): This command deletes the edge whose ID is **501-502-0**.

NOTE

- 1. You can press the up and down arrow keys in the text box to view historical query commands.
- 2. When you enter a syntax keyword, the system automatically displays historical statements with the same keyword.

Figure 8-77 Historical queries

Running Record	Query Result		
Q Select a propert	y or enter a keyword.		
name 🖨	status ♦	required parameter 😝	jobid 😝
gremlin	SUCCESS	{"command":"g.V().limit(10).val	
gremlin	g.V().limit(3).valueMap() g.V().limit(3).valueMap() g.V().limit(100)	{"command":"ɑ.V().limit(3).valu	
Gremlin ∨ s			

- 3. Keywords in the text box are displayed in different colors.
 - Reserved words in gray

Note: A reserved word is predefined in the syntax system of a programming language. Reserved words vary depending on programming languages.

- String values in orange
- Delimiters in red. Regular delimiters including square brackets [], curly brackets {}, parenthesis (), commas (,), and semicolons (;).
- Variables in green

Figure 8-78 Gremlin keywords



Gremlin Syntax Optimization

GES integrates the OLTP function of Gremlin, enhances some features, and optimizes the strategy.

• Enhanced Text Predicate

g.V().has('name', Text.textSubString('xx'))

Predicate	Description
textSubString	Substring
textClSubString	Substring that ignores cases
textFuzzy	Fuzzy match
textPrefix	Prefix query
textRegex	Regular expression match

NOTE

When specifying a schema, do not name the attributes **id**, **label**, **property**, or **properties**.

When you do Gremlin queries with many steps, the results will be converted into a map. Two identical keys are not allowed in a map structure. If multiple identical keys are inserted into a map, the key value will be overwritten or this operation is canceled. If you set an attribute name to **id**, **label**, **property**, or **properties**, the returned results will be incomplete because in many queries the graph ID is returned together with the attribute ID.

Reference

Table 8-5 shows how Gremlin in GES differs from open source Gremlin.

Difference	Description
Vertex and Edge IDs	An edge ID consists of the source vertex ID, target vertex ID, and index that distinguishes duplicate edges. The three parts are connected by hyphens (-), for example, sid-tid-index. Edge and vertex IDs must be the string type.
User Supplied IDs	Users can only provide vertex IDs without hyphens (-).
Vertex Property IDs	Both edge and vertex properties do not have IDs. The returned IDs are vertex IDs.
Vertex and Edge Property	Vertex and edge properties are defined by metadata files in GES. Therefore, you cannot add or delete properties, but you can use property() and remove() to modify property values. The value set by property() is determined by the corresponding parameter. remove() converts string properties into empty strings, digital properties into 0, and list properties into empty lists.
Variables	The GES graph structure does not support the variables feature.
Cardinality	GES supports the single and list cardinality. The value type of a vertex property is defined by the metadata file. Therefore, no new property is added when you set the property value.
Transactions	During GES Gremlin implementation, transactions are not explicitly used.

Fable 8-5 G	ES Gremlin	differences
Table 8-5 G	ES Gremun	anterences

You can use the **feature** function to view the supported Gremlin features. If **false** is displayed, GES does not support the feature. If **true** is displayed, GES supports the feature. For details about the features, visit the **Gremlin official website**.

gremlin> graph.features() ==>FEATURES

D NOTE

Currently, the following step commands are not supported:

- tryNext()
- explain()
- tree()

8.14 Cypher Query

Cypher is a declarative graph query language. You can use Cypher statements to obtain query result and modify data in GES.

The procedure is as follows:

- 1. Access the GES graph editor. For details, see Accessing the Graph Editor.
- 2. Use label-based vertex and edge indexes during Cypher query.

If this is your first time using Cypher, click **Create Index** in the upper right corner of the result display area. You do not need to perform this operation in subsequent operations.

Figure 8-79 Creating an index



3. In the graph data query area, press **Enter** before entering your query statement.

Cypher Statements

The following are typical query statements.

• Querying a vertex

match (n:movie) return n: Query the vertex whose label is movie.

match (n) return n limit 100: Query details about 100 vertices.

match (n{Occupation:'artist'**}) return id(n), n.Gender limit 100**: Query the first 100 vertices whose **Occupation** is **artist**, and return their IDs and genders.

match (n) where id(n)='Vivian' return n: Query the vertex whose ID is **Vivian**.

match (n) return n skip 50 limit 100: Query all vertices of a graph. Skip the first 50 vertices, and return a total of 100 vertices.

• Querying an edge

match (n)-[r]->(m) return r, n, m: Query all edges. Return the edges and vertices at both ends.

match (n)-[r:rate]->(m) return r, n, m: Query the edges whose label is rate.

match (n)-[r:rate|:friends]-(m) where id(n)='Vivian' return n,r,m: Query all edges whose start vertex is Vivian and edge label is rate or friends.

• Searching by path

match p=(n:user)--(m1:user)--(m2:movie) return p limit 100: Query the paths whose start vertex is **user**, first-hop end vertex is **user**, and second-hop end vertex is **movie**. Returns the first 100 paths.

• Aggregating and deduplicating based on groups

match (n) return count(*): Query the number of all vertices in a graph.

match (n:user) return n.Gender, count(n): Collect statistics on the number of **user** vertices in every gender.

match (n:user) return distinct n.Occupation: Return deduplicated occupations of all **user** vertices.

Sorting

match (n:user) return id(n) as name order by name: Change IDs of all user vertices to name, and sort the vertices by name.

• Creating a vertex

create(n:movie{_ID_:'The Captain', Year:2019})return n: Create a vertex
whose ID is The Captain, label is movie, and Year is 2019. Return the vertex.

create(n:movie{_ID_:'The Captain', Year:2019})-[r:rate]->
(m:movie{_ID_:'The Climbers',Title: 'The Climbers', Year:2019}) return r:
Create two vertices and their associated edges.

• Creating an edge

match (n),(m) where id(n)= 'The Captain' and id(m)= 'Lethal Weapon'
create (n)-[r:rate]->(m) return r : Create an edge whose label is rate
between two vertices with specified IDs. (You are advised to use this query in
2.2.21 and later versions.)

• Modifying properties

match (n) where id(n)= 'The Captain' set n.Title= 'The Captain' return n: Search for the vertex whose ID is The Captain and change the attribute Title to Ji Zhang.

• Deleting a vertex

match (n) where id(n)=' The Captain' delete n: Search and delete the vertex whose ID is **The Captain**.

match (n) where id(n)=' "detach delete n": Search for the vertex whose ID is **The Captain**. Delete the vertex and its edges.

• Querying a schema

If you call **db.schema()** independently, only the schema metadata of the vertices is returned. Multiple isolated vertices are displayed on the canvas.

NOTE

- 1. You can press the up and down arrow keys in the text box to view historical query commands.
- 2. When you enter a syntax keyword, the system automatically displays historical statements with the same keyword.

Figure 8-80 Historical queries

Running Record	d Query Result		
Q Select a prope			
name 😝	status ⊜	required parameter 😝	jobid 🔶
cypher	success	{"statements":[{"statement":"m	
cypher	match (n) return count(*) match (n) return n skip 50 limit 100 match (n) return n skip 50 limit 100 match (n) return n skip 50 limit 100	{"statements":[{"statement":"m	
Cypher \vee	match		

- 3. Keywords in the text box are displayed in different colors.
 - Reserved words in gray

Note: A reserved word is predefined in the syntax system of a programming language. Reserved words vary depending on programming languages.

- String values in orange
- Key-value pairs in purple. They are of the non-string type in the *key:value* format.
- Delimiters in red. Regular delimiters including square brackets [], curly brackets {}, parenthesis (), commas (,), and semicolons (;).
- Variables in green

Figure 8-81 Cypher keywords



8.15 DSL Query

DSL is a graph query language. You can use DSL statements to query and compute graphs, helping you design and run algorithms at low costs. This function applies only to graphs of 2.3.14 or later.

The procedure is as follows:

- 1. Log in to the GES graph editor. For details, see Accessing the Graph Editor.
- 2. In the graph data query area, click the drop-up button to choose **DSL**. Enter a query statement and press **Enter** to run the statement.

Figure 8-82 Switching to DSL query

Running Re	ecord	Query Result		
Cypher	operty or e	nter a keyword.		
Gremlin		etatue 🛆	roquired parameter	iobld 🛆
DSL				
Cypher ^				

NOTE

Multi-label graphs do not support DSL query.

Common DSL Query Statements

The following are typical query statements.

- Querying a vertex
 Match<Vertex> v(['Vivian','Eric']);return v: Query vertices whose IDs are Vivian and Eric.
- Querying neighbor vertices in *N* hops

Match<Vertex> v(['Vivian']);v.repeat(bothV()).times(2).emit();return v: Query all neighbor vertices in two hops in both directions of a vertex whose ID is Vivian.

• Returning a subgraph

Match<Vertex> v(['Vivian','Eric']); return v.subgraph(): Return vertices Vivian and Eric and the edge set between them.

• Other statements

Match<Vertex> v(); v.pick(1); return v: Randomly match and return one vertex.

Match<Vertex> v(); v.pattern('match (n:user) return n'); return v: // Use Cypher statements to query and return the vertex set.

NOTE

- 1. You can press the up and down arrow keys in the text box to view historical query commands.
- 2. When you enter a syntax keyword, the system automatically displays historical statements with the same keyword.
- 3. Keywords in the text box are displayed in different colors.
 - Reserved words in gray

Note: A reserved word is predefined in the syntax system of a programming language. Reserved words vary depending on programming languages.

- String values in orange
- Key-value pairs in purple. They are of the non-string type in the *key:value* format.
- Delimiters in red. Regular delimiters including square brackets [], curly brackets {}, parenthesis (), commas (,), and semicolons (;).
- Variables in green

Figure 8-83 DSL keywords



8.16 Analyzing Graphs Using Algorithms

You can analyze graphs using basic graph algorithms, graph analysis algorithms, and graph metric algorithms.

Procedure

- 1. Log in to the GES graph editor. For details, see Accessing the Graph Editor.
- 2. In the algorithm library area, you can select an algorithm and set its parameters.

Algorithm List shows the algorithms supported by GES and **Algorithms** describes the algorithm details.

Figure 8-84 Setting algorithm parameters

∧ ♠ PageRank	?	⊘
alpha 🕐		
0.85		
convergence ⑦		
0.00001		
max_iterations ⑦		
1000		
directed ⑦		
true		~

Algorithms such as PersonalRank, K-hop, and Shortest Path that use the **source** (vertex ID) and **target** parameters for query now support querying vertices by property. However, this feature is currently only available for memory edition graphs.

Figure 8-85 Querying a vertex

	۹							
	~							
	00							
 B PersonalRank 	00							
	I Verlex							
		FIGERO MADE						
			Enter a boolean v	salaa, enther (0/1) or (bu	antaise).		_	
			If the aftri	bute has multiple value	s, you can enter only	one value for que	v ×	
			-					
					chararray	boolean	string	double
	×							
	00							
	00							
	⊙ ⊙							
	⊚ ⊙							
	⊚ ⊙		_		_			
	⊚ ⊙							
$\sim - \mathbf{v}_{\mathbf{f}}^{\prime}$ Label Propagation	⊚ ⊙		1 ×					
	⊚ ⊙		After sole deplaying	cting the Label, propert pup to 2,000 entries.	ly, and inputting the p	roperty value, the	vertex list will autor	natically refresh.
	•							
	⊚ ⊙							

3. Run the algorithm by clicking \bigcirc . You can view the query result after the analysis is complete.

NOTE

1. Only the results of 500 vertices are displayed due to the size of the result display area. If you want to view the complete query results of global iterative algorithms, such as the PageRank algorithm, you can call the algorithm APIs. For details, see Algorithm APIs.

Take the sample movie data in the template as an example. The following figure shows the PageRank values.

Figure 8-86 Viewing the analysis result

Run	ning Record Query Result
	"runtime": 0.000275,
	"results": [
4	
5	"id": "Comedy",
	"pagerank": 0.03127640523990175
	"id": "Action",
10	"pagerank": 0.021366796120714945
11	

a. Adjust the parameters, and run the algorithm again. PageRank value is different this time, but the top ranking does not change.

Figure 8-87 Adjusting parameters



b. Perform association prediction to obtain the association degree of the two movies. The association degree is 0.029, indicating that only a small group of people have watched both movies.




Figure 8-89 Association analysis result



8.17 Managing Indexes

The index management function is added to the graph access page to facilitate index addition, deletion, and search.

Creating an Index

- 1. Log in to the graph editor. For details, see Accessing the Graph Editor.
- 2. On the **Indexes** tab of the graph editor, click **Create**.

Figure 8-90 Creating an index

sis	Explore Operations Schema Algorithms Indexes
aph Analy	Create
Gri	Q Select a property or enter a keyword.
lueny	Type 🖨 🛛 Name 🖨 🛛 Attribute 🖨
Visual Q	Global edge cypher_edg –
apshot	Global verte cypher_vert –

- 3. In the **Create** dialog box that appears, set the following parameters:
 - **Name**: Enter a custom index name.
 - **Type**:
 - Memory edition: The options are Global vertex index and Global edge index.
 - Database edition: The options are Global vertex index, Global edge index, Local vertex index, and Local edge index.
 - **Label**: You can enable or disable this toggle.
 - Label Name: This parameter is available only when **Type** is Local vertex index or Local edge index.
 - **Attribute**: Only when there is a single attribute, its name will be displayed here. If there are multiple attributes, their names are not displayed.



Figure 8-91 Parameters for creating an index

NOTE

For a database edition graph, you can create any number of indexes. For a memory edition graph, you can create up to 10 indexes.

Click OK. A dialog box appears, where you can choose whether to go to the 4. task center to monitor the index creation progress.

Figure 8-92 Monitoring the creation progress



Once the creation is successful, the new index is displayed on the Indexes tab. 5.

Deleting an Index

- On the Indexes tab of the graph editor, locate the index you want to delete 1. and slide the scroll bar from left to right.
- 2. Click **Delete** in the **Operation** column.

Figure 8-93 Deleting an index



3. In the **Delete Index** dialog box that appears, confirm the index information, manually enter **DELETE** or click **Auto Enter**, and click **OK**.

Figure 8-94 Deleting an index

Delete Inde	ex			
The following ind	exes will be delet	ed:		
Туре	Name	Attribute	Label included	
Global verte	test01		true	
To confirm deleti	on, enter "DELET	E" below . Auto Er		
DELETE				
			Cancel	ок

8.18 Analyzing Graphs on the Canvas

The canvas intuitively displays the graph data. You can also edit and analyze data in this area. For details about the shortcut keys and interface elements on the canvas, see **Table 8-4**.

The procedure is as follows:

- **Step 1** Log in to the GES graph editor. For details, see Accessing the Graph Editor.
- **Step 2** On the canvas, right-click a vertex or an edge, and perform the following operations:

Figure 8-95 Shortcut menu



• View Property



Figure 8-96 View Property



• Search by Association

You can select **OUT**, **IN**, and **ALL** to expand vertices related to the current vertex.

- **OUT**: Query the vertices using this vertex as the source vertex.

- **IN**: Query the vertices using this vertex as the target vertex.
- ALL: Query all vertices of OUT and IN.

• Export

Export the graph or data displayed on the canvas.

• Search by Path

Query paths between two vertices. All possible paths are listed.

Procedure: Hold down **Ctrl** and click two vertices. The first is the source vertex and the second is the target vertex. Then, Right-click and choose **Search by Path** from the shortcut menu.

NOTE

This option is valid only when two vertices are selected. Otherwise, it is dimmed.

After this function is executed, the canvas is cleared, and then the queried vertex and edge data is returned and rendered in the canvas. A path is formed based on the selected two vertices.

Figure 8-97 Search by path



- Shortest Path of the Vertex Sets
 - a. Hold down **Shift** and box-select a group of vertices (a single vertex or multiple vertices).
 - b. Hold down **Shift** and box-select another group of vertices (a single vertex or multiple vertices).
 - c. Right-click in the selection box and choose **Shortest Path of the Vertex Sets** from the shortcut menu.
 - d. In the dialog box that is displayed, you can edit the selected two sets of vertices and click + to quickly add vertices.
 - e. Click **Run**. The shortest paths between two vertex sets are returned.
- Common Neighbors of Vertex Sets
 - Function

By box-selecting the common neighbors of two vertex sets, you can intuitively discover the objects associated with the two sets.

- Procedure
 - i. Hold down Shift and box-select two vertex sets.



ii. Right-click a vertex set and choose **Common Neighbors of Vertex Sets** from the shortcut menu.



Figure 8-99 Common Neighbors of Vertex Sets

iii. In the displayed dialog box, confirm the vertices in the vertex set. You can add or delete vertices and determine whether to carry additional parameters. Then, click **Run**.

Common Neighbors of the Vertex Sets	×
Source Vertex Set	
Target Verlex Set	
Carry Additional Constraints ③	
Cancel	Run

Figure 8-100 Confirming the vertices in the vertex sets

The **Carrying additional constraints** option allows you to limit the result set:

- If this option is not selected, the found common neighbors are the intersection of the neighbors corresponding to the source vertex set and target vertex set.
- If this option is selected, the found common neighbors are not only the intersection of the neighbors corresponding to the source vertex set and target vertex set, but each vertex in the common neighbor set has at least two neighboring vertices in the source vertex set and target vertex set.
- iv. Display the result.

Figure 8-101 Graph



Figure 8-102 Query result



- **Sub Graph**: Press and hold **Ctrl** and select some vertices. The edges between those vertices and the selected vertices form a new graph.
- Add Edge: Add an edge.

To add an edge between any two vertices on the canvas, hold down **Ctrl**, select two vertices on the canvas, right-click the selected vertices, and choose **Add Edge**. By default, the vertex selected first is the source vertex, and that selected later is the target vertex. After the edge is added, you can select the label of the edge and set the edge properties.

- Hide: Hide the selected vertex.
- **Delete**: Delete a vertex, an edge, multiple vertices, and multiple edges, or delete edges and vertices in batches.
 - To delete a vertex /edge, select the vertex/edge and delete it.
 - To delete multiple vertices/edges, press Ctrl to select the vertices/edges and delete them.
 - To delete vertices and edges in batches, hold down Shift and drag the left key of the mouse to select multiple vertices and edges and delete them.

After you click **Delete**, a confirmation dialog box is displayed. Confirm information about the vertices and edges you want to delete and click **OK**.

Figure 8-103 Deleting vertices and edges

Delete v	vertices and	edges? Del	eted vertices	and edges ca	innot be rest	ored.
Vertices			"		"	[*]
1101 ×	3101 ×	7101 ×				
(1101) ->	(3101) ×	(1101)>	(7101) ×			

D NOTE

The vertices and edges will be permanently deleted and cannot be restored. Exercise caution when performing this operation.

Step 3 View the details about a vertex.

Move the cursor to a non-virtualized vertex. The ID, label, and properties of this vertex are displayed.

Figure 8-104 Details

7 ^{id}	78
label	user
occupation	academic/educator
gender	M
Zip-code	85718
userid	32
	50.

NOTE

A maximum of six properties of a vertex can be displayed in the pop-up window. When the number of properties is greater than six, you can view all of them in the filter and property tab as shown in **Editor page**.

----End

8.19 Graph Display in 3D View

The 3D view of a graph provides you intuitive analysis experience.

NOTE

Restrictions: The 3D view for displaying graphs is currently only supported by the PageRank algorithm, the PersonalRank algorithm, Cypher queries, and Gremlin queries. Other algorithms or functions can only analyze graphs in the 2D view.

Displaying a Graph in 3D View

The following example shows how to view results of the PagePank algorithm in the 3D view graph:

- 1. In the algorithm area on the left of the graph editor, select the PagePank algorithm and set required parameters.
- 2. Run the algorithm. After the analysis is complete, you can view the result in the canvas.
- 3. Click in the upper left corner of the canvas to switch to the 3D view.

Figure 8-105 3D view of the result



8.20 Filter Criteria

You can set filter criteria to filter graph data.

The procedure is as follows:

- 1. Log in to the GES graph editor. For details, see Accessing the Graph Editor.
- 2. Click on the right of the canvas, or select a vertex on the canvas, rightclick it , and choose **View Property**, to display the **Filtering and Property** page.
- 3. In the **Filtering and Property** area, set the filtering conditions and click **Filter**.
 - Match: Vertex is selected by default. Possible values are Vertex and Edge.
 - **Type**: **All types** is selected by default. You can select the vertex or edge type from the drop-down list. The type is defined by the metadata file you upload.
 - Add filtering condition: Click Add filtering condition to select a property and choose a condition (Less than, Greater than, Equal to, Not equal to, In range, Existent, Non-existent, Greater than or equal to, or Less than or equal to). Properties are defined by the metadata file you upload. You can add multiple filtering conditions or click Delete to delete set conditions.

Filtering	Property	Statistics
Match		
Vertex		
Туре		
All types		
Property		
Add filtering con		
	Filter	

Figure 8-106 Setting filtering conditions

4. After the execution is complete, the filtering result is displayed in the drawing area and result area.

8.21 Editing Properties

The **Property** tab displays information about the properties of the selected vertices and edges. You can edit the properties of a single vertex or edge.

The procedure is as follows:

1. Right-click a vertex/edge on the canvas and choose **View Property** from the shortcut menu. The **Property** tab is displayed on the right, showing the properties of the selected vertex/edge.

If the selected vertex has multiple labels, you can click the drop-down box next to the label to view the properties of other labels.

Figure 8-107 Properties



2. Click \blacksquare next to the property to edit it.



Click **Edit All** at the bottom of the property area to edit all the displayed properties. Click **Save All**.

3. Click 🔽 after you finish editing.

NOTE

In the **Property** tab, only the properties of a single vertex or edge can be edited. In the **Schema** tab of the metadata area, you can add or delete all properties of a tag, as described in section **Editing Schema**.

8.22 Statistics Display

To view the number of tags and vertex weights of specified vertices and edges, you can select the vertices and edges on the canvas. For details about the concepts of vertices and edges, see **Graph Data Formats**.

To display statistics, perform the following steps:

- 1. Log in to the GES graph editor. For details, see Accessing the Graph Editor.
- 2. Click on the right side of the canvas. The **Filter**, **Property**, and **Statistics** tabs are displayed. Click the **Statistics** tab.
 - **Tags**: Statistics on all tags, and the number of vertices and edges of each tag on the current canvas
 - **Top 10 Vertex Weight**: Top 10 vertices with the largest number of edges in the current graph

In the following example, there are seven tags. There are five vertices tagged with **FUND_PRODV** and three vertices tagged with **FIN_PRODV**.

In the example graph, the vertex whose ID is 1101 has the largest weight. There are five edges in total. The vertex ranked No. 10 is vertex 1103. There is one edge in total.

Figure 8-108 Tag statistics

Filtering	Property	Statistics
∧ Taos		4/14 Selected
		2/5
		213
		2/3
CHANNELV		0/2
RISK_LEVELV		0/1
FUND_CORPV		0/1
TRUST_ORGV		0/1
FUND_MGRV		0/1
A Top 10 Verte	ex Weight	
1101		0/5
6101		0/5
3101		0/2
5101		0/2
7101		0/2
1102		0/2
2101		0/2
4101		0/1
3102		0/1
1103		0/1

3. Press **Shift** and drag the left key of the mouse to select vertices and edges in the graph. The tags of the selected vertices and edges are displayed along with the top 10 vertices with the highest weights among the selected vertices.

8.23 Check Running Records

The system logs your operations in a table, allowing you to review the execution progress and completion time when analyzing data.

The procedure is as follows:

- 1. Log in to the GES graph editor. For details, see Accessing the Graph Editor.
- After executing a Gremlin/Cypher/DSL query or algorithm analysis, the operation record name, status, request parameters, job ID, start time, and duration will be displayed under the **Running Record** tab. Clicking the **Query Result** tab will take you to the query results page, where you can view the complete results of the operation.

Figure 8-109 Running Record tab

Running Record				
Q Select a property or enter a				
name 🖨				
Gremin				
Gremin				
PageRank				
Total Records: 3 10 🗸				
Cypher v Input a Dyp				

- 3. The **Operation** column offers the following functions:
 - a. To stop the execution of an algorithm while it is running, click **Stop** in the **Operation** column.
 - b. To rerun Gremlin, Cypher, or DSL query requests without having to reenter them in the query area, click **Resend** in the **Operation** column.
 - c. To modify a previously executed Gremlin, Cypher, or DSL query request, click **Re-enter** in the **Operation** column and the query statement will be re-entered in the query area.
- 4. To export the running record, click **Export** in the upper right corner and select the desired export format.
 - Cypher queries support two export formats: JSON and TXT.
 - Gremlin and DSL queries only support the JSON export format.

8.24 Viewing Query Results

After data analysis is complete, you can directly view the result on the canvas or on the **Query Result** tab page.

The procedure is as follows:

- 1. Log in to the GES graph editor. For details, see Accessing the Graph Editor.
- 2. Perform a Gremlin/Cypher/DSL query or algorithm analysis and check the query results on the **Query Result** tab page.

If the returned results are too large to be fully displayed on the canvas and result area, you can click the export button in the upper right corner to download the analysis results. Currently, three export formats are supported: **json**, **csv**, and **excel**.

 Run a Gremlin command. The command output is quickly displayed. For example, if you run the g.V().limit(100) command, the result is as follows:



 Run a Cypher command. The command output is quickly displayed. For example, if you run the **match (n) return n limit 100** command, the result is as follows:

Figure	8-111	Cypher	output
--------	-------	--------	--------



Run a DSL command to display its execution result. For example, if you enter the query command Match<Vertex> v(); v.pick(1); return v, the query result is as follows:

Figure 8-112 DSL output



- Run an algorithm. The running time and result are displayed. For example, if you run PageRank, the result is as follows:

Figure 8-113 Algorithm output



9 Viewing Graph Tasks

9.1 Graph Overview

Log in to the GES management console. In the navigation pane on the left, choose **Overview**. The **Overview** page displays the following function modules:

- **Process Flow**: help you understand how to use the service.
- My Resources: includes graph data information like status, size, and backup.
- Industry-Specific Graph Templates: shows the supported templates. Clicking a template takes you to the Use Industry-Specific Graph Template tab page where you can create a chart based on the template.



Figure 9-1 Overview

NOTE

To hide a function module, click $^{\odot}$ next to the module name.

Figure 9-2 Hiding a function module



Graph Status

The **Graph Status** pane displays the number of graphs in different statuses. Currently, the system supports the following statuses.

 Table 9-1 Graph statuses

Status	Description
Running	Running graphs. Graphs in this status can be accessed.
Preparing	Graphs whose ECSs are being created or started
Starting	Graphs being started
Stopping	Graphs being stopped
Upgrading	Graphs being upgraded
Importing	Graphs being imported
Exporting	Graphs being exported
Rolling back	Graphs being rolled back
Clearing	Graphs being cleared
Preparing to change size	Change in graph size being created or initiated
Changing size	Graph size being changed
Rolling back to previous size	Rolling back the graph to its previous size
Preparing for expansion	Graphs preparing for expansion
Expanding	Graphs being expanded
Stopped	Stopped graphs. Graphs in this status cannot be accessed, but can be restarted.
Frozen	Frozen graphs. The user account that created these graphs are frozen. NOTE After a user account is frozen, only deletion operations are allowed.
Abnormal	Abnormal graphs. Graphs in this status cannot be accessed.
Failed	Graphs failed to be created
Rolling back	Graphs being rolled back

Graph Size

The **Graph Size** pane displays the number of graphs in different sizes. Currently, the system supports the following eight sizes.

Only graph names and the number of graphs are displayed.

Table 9-2 Graph sizes

Size	Description
Ten-thousand- edge	Indicates that the number of edges of a graph cannot exceed 10 thousand.
Million-edge	Indicates that the number of edges of a graph cannot exceed 1 million.
Ten-million- edge	Indicates that the number of edges of a graph cannot exceed 10 million.
Hundred- million-edge	Indicates that the number of edges of a graph cannot exceed 100 million.
Billion-edge	Indicates that the number of edges of a graph cannot exceed 1 billion.
Billion-edge- pro	Indicates that the number of edges of a graph cannot exceed 2 billion.
Ten-billion- edge	Indicates that the number of edges of a graph cannot exceed 10 billion.
Hundred- billion-edge	Indicates that the number of edges of a graph cannot exceed 100 billion.

Graph Backup

You can back up graphs to prevent data loss. The **Graph Backup** pane displays the numbers of graphs with and without backups.

 Table 9-3
 Backup statuses

Backup Status	Description
Backed up	Indicates the number of graphs that are backed up.
Non-backed up	Indicates the number of graphs that are not backed up.

Payment Details

This part displays the billing mode, number of instances, and graph expiration time.

9.2 Task Center

9.2.1 Management Plane Task Center

If you want to view details about creating, backing up, starting, backing up, importing, exporting, and upgrading tasks, you can go to the **Task Center** page.

The procedure is as follows:

- 1. In the navigation pane on the left, click **Task Center**.
- 2. On the **Task Center** page, view the task type, task name, graph name, associated graph, start time, end time, status, and running result.

Figure 9-3 Task center

Task Center ③								
C Belect a property or en	ter a keyword.							00
Туре	Task Name	Graph Name	Associated Graph	Start Time	End Time	Status	Job ID	Operation
Graph Management	Import Graph	controller_movie	controller_movie	Mar 13, 2024 11:50:10 G	Mar 13, 2024 11:50:15 G	Succeeded	2c9000d00e35x0f0010e35ee054902f2	View Details Cause of Failure
Graph Management	Import Graph	controller_movie	controller_movie	Mar 13, 2024 11:49:52 G	Mar 13, 2024 11:49:57 G	Succeeded	2c0080d08e35a0f8018e35ee3c1b02ef	View Details Cause of Failure
Graph Management	Import Graph	controller_movie	controller_movie	Mar 13, 2024 11:48:45 G	Mar 13, 2024 11:48:50 G	Partially successful	2c9080ba8e35a3ec018e35ed2e820077	View Details Cause of Failure
Graph Management	Clear Graph	movie_gesoperator	movie_gesoperator	Mar 13, 2024 11:47:07 0	Mer 13, 2024 11:50:21 G	Succeeded	2c9000d08e35a0f8018e35ebc1d902ec	View Details Cause of Failure

3. In the **Running Result** column, click **View Details** to view the detailed information. You can also click **Cause of Failure** or **Job ID**.

Figure 9-4 Viewing details

iew Deta	ils						
īype ⊜	File Path	Status \ominus	Cause of Failure	Log	Total Impo	Row Impo	Successf
/ertex D	north7-gesbucket/Logs.txt	Succeeded	-		0	0	0
/ertex D	north7-gesbucket/aikv/	Succeeded			146	0	146
/ertex D	north7-gesbucket/aikv/r	Failed	File data does not				
/ertex D	north7-gesbucket/movi	Failed	File data does not				
/ertex D	north7-gesbucket/movi	Succeeded			146	0	146

If the status of a data import task is **Partially successful**, you can click **View Details** to view information such as the type of data that fails to be imported and the number of rows that fail to be imported. To view the cause of failure, check the log path (optional) specified when you import the graph because failure logs are uploaded to the path.

Figure 9-5 Partially successful task

View Deta	View Details ×								
Type ↓Ξ	File Path	Status ↓Ξ	Cause of Failure	Log	Total Impor	Row Import	Successfull		
Vertex D	north7-gesdata/auData	Succeeded			146	0	146		
Edge Dat	north7-gesdata/auData	Partially suc	-	View	1659	450	1209		

4. On the **Task Center** page, search for a task in any of the following ways:

Figure 9-6 Searching for a task

4 ت	elect a property or enter a	keyword.								Q	8
тур	API Filtering	10	Graph Name	Associated Graph	Start Time	End Time	Status	Job ID	Operation		
Grat	Type Task Name	aph	controller_movie	controller_movie	Mar 13, 2024 11:50:10 G	Mar 13, 2024 11:50:15 G	Succeeded	2c9080458e35e01010e35ee85490212	View Details	Cause of Failur	-
Grat	Associated Graph Status	aph	controller_movie	controller_movie	Mar 13, 2024 11:49:52 G	Mar 13, 2024 11:49:57 G	Succeeded	2c9000d00e35e01010e35ee3c1b02ef	View Details	Cause of Failur	10
Gray	Job ID Dates	aph	controller_movie	controller_movie	Mar 13, 2024 11:40:45 G	Mar 13, 2024 11:40:50 G	Partially successful	2c9000ba0e35x3ec010e35ec2e02007	View Details	Cause of Failur	10

- a. Selecting the task type
- b. Selecting the task name
- c. Entering an associated graph
- d. Entering a task status
- e. Entering a task ID
- f. Setting the time

9.2.2 Service Plane Task Center

Task Center @

The task center allows you to view details about the historical tasks and asynchronous tasks that are being executed.

The procedure is as follows:

 In the navigation pane, choose Graph Management. On the displayed page, locate the target graph and choose More > Task Center in the Operation column.

Figure 9-7 Query task center

	a 0
Enter a jui D.	
HolD 0 Type 0 Original Request 0 Status 0 Progress 0 Start Time 0 Directions Running Ress	t
Sano, Micagement 24/50185-dot-4b7+453-dot3895894 Dukleheler. (IndexTyper'GlobalCompext Succeeded 100%. Mer 13, 2004 18:19:57 GUT+ Mer 13, 2004 18:19:57 GUT+ Mer 13, 2004 18:19:57 GUT+	Cause of Failure
Tail Center 7 #12/25b 25b6-1306-1206-1206-1206-1206-1206-1206-1206-12	Cause of Pallure
2044/58/92-022-4685-4502-4603-4502-4603-4502-4603-4502-4502-4502-4502-4502-4502-4502-4502	Cause of Pallare
Accell and Company Provides Company Prov	Cause of Patture
2007/965-0083-4002-2468-00072542 DetectFiler (VOISer."V", Visualized" "Tou"	Cause of Failure
73050446.0692.4675.8050-742840044 DetectlyFile ("vertexeelPart": fronth?gesd Bacceceded 100% Mar 13, 2024 16 16 20 0MT+ Mar 13, 2024 16 16 21 0MT+ Bacterial View Details	Cause of Failure
1ca03e64 9913-4265-dsfe9 99369693a Deleterabel 98cceeded 109% Mar 13, 2024 18:16:05 GMT+ Mar 13, 2024 18:16:05 GMT+ Starperst View Details	Cause of Failure

D NOTE

- The query task center is available for graphs of version 2.2.23 and later.
- You can access the query task center of graphs that are in the running, importing, exporting, or clearing states only.
- 2. In the upper left corner of the **Task Center** page, select a node from the drop-down list to view details about the asynchronous tasks that are being executed or have been executed. The following task information is displayed:
 - **Job ID**: Job ID of an asynchronous task
 - Graph Name: name of graphs of the database edition
 - Task Type: Type of the asynchronous task, including ImportGraph and VertexQuery
 - Original Request: Original request body sent by the user
 - Status: Task status, which can be Suspended, Running, Succeeded, or Failed
 - **Progress**: Progress of the task

- **Start Time**: Time when the task starts. If the task does not start, the start time is empty.
- End Time: Time when the task ends. If the task does not end, the end time is empty.
- **Operation**: You can suspend the task.
- **Running Result**: You can view the task details. If the task fails, you can view the failure cause.
- 3. To view details about an asynchronous task, search the task by its job ID using the search box in the upper right corner of the page.

9.3 Viewing Monitoring Metrics

Cloud Eye monitors the running status of your graphs. You can view the monitoring metrics of GES on the Cloud Eye management console.

It takes a period of time for transmitting and displaying monitoring data. The GES status displayed in the Cloud Eye monitoring data is the status obtained 5 to 10 minutes before. You can view the monitored data of a graph 5 to 10 minutes after it is created.

Prerequisites

- The created graph is running properly.
- The graph has been properly running for at least 10 minutes. For a newly created graph, you need to wait for a while before viewing its monitoring metrics.
- You can view monitoring data of graphs in the running, importing, exporting, and clearing states. The monitoring metrics can be viewed after the real-time service starts or recovers.

Viewing Metrics

- 1. Log in to the GES management console.
- In the navigation pane, choose Graph Management. In the Operation column, choose More > View Metrics. The Cloud Eye management console is displayed.
- 3. On the monitoring page for GES, you can view the figures of all monitoring metrics.

Figure 9-8 Viewing monitoring metrics

55 26 528 56 76 🔯 Auto Raheah 🌑			Salect Mario	
Period Raw data *			Driter a metric name.	0
Memory Usage 🕥		Computing Resource Usage ③	KOPS 🛞	
*	Max Min 32 29:550	% Max Min 16.660 11.333	Countin Hac 10 800	5.500
30 A 20 10			m	
09255 10:00 10:05 10:10 10:15 10:20 10:25 10:30 10:35 10:40 10:	45 10.50	0855 1000 1085 1010 1015 1020 1025 1030 1835 1040 1045 1850	0925 1030 1025 1010 1015 1020 1025 1030 1025 1040 1045	0.50
Network Input Throughput		Network Output Throughput	Disk Usage (1)	
2 A	Max Min 2.556 0.036	000 Mar. Mar. Mar. Mar. Mar. 101 450 4289 000 44289 000 44280 0000 44280 000 44280 000 44280 000 44280 000 44280 000 44280 000 44280 0000 44280 0000 442800 0000 440000000000	% Main 380	Min 3.800
2 1 • • • • • • • • • • • • • • • • • • •	A5 10.50	00 0 0000 1000 1010 1010 1020 1020 1030 103	1 Y 2 1 	10:50
Total Dak Size 💮		Disk Space Used 💿	Disk Read Throughput	
08 20 20 20 20 20 20 20 20 20 20 20 20 20	Max Min Ie 850 ee 850	08 Mar Mn 2250 200 2 V	Syles Max 800 075 540 60	5.940
0 30 0 00:55 10:00 10:05 10:10 10:15 10:20 10:25 10:20 10:35 10:40 10:	45 10:50	* 1 0 0055 1060 1085 1010 1015 1020 1025 1030 1035 1048 1048 1059	0 0 0 0055 1000 1005 1010 1015 1020 1025 1020 1025 1040 1045	10.50
Disk Write Throughput		Average Time per Disk Read (1)	Average Time per Disk Write (1)	
NOP N 00 ▲ 00 ▲ 00 ▲ 00 ▼ 00 ▼	Max Min 16.480 78.400	t Buckton 0 0 00 00 03		5.000
0		0	0	

- 4. The system allows you to select a fixed time range or use automatic refresh.
 - a. Fixed time ranges include **1h**, **3h**, **12h**.
 - b. The automatic refresh interval is 60s, which is the user monitoring period.

Metrics

This chapter describes metrics reported by GES to Cloud Eye as well as their namespaces, lists, and dimensions. You can use the management console and APIs provided by Cloud Eye to query the metric and alarm information generated for GES.

NOTE

The namespace of the metrics reported by GES to Cloud Eye is SYS.GES.

Metric ID	Metric	Description	Value Range	Monitore d Object	Monitor ing Period (Origin al Metric)
ges001_vert ex_util	Vertex Capacity Usage	Capacity usage of vertices in a graph instance. The value is the ratio of the number of used vertices to the total vertex capacity. Unit: %	0-100 Value type: Float	GES instance	1 minute

Table 9-4 GES metrics

Metric ID	Metric	Description	Value Range	Monitore d Object	Monitor ing Period (Origin al Metric)
ges002_edg e_util	Edge Capacity Usage	Capacity usage of edges in a graph instance. The value is the ratio of the number of used edges to the total edge capacity. Unit: %	0-100 Value type: Float	GES instance	1 minute
ges003_ave rage_impor t_rate	Average Import Rate	Average rate of importing vertices or edges to a graph instance Unit: count/s	0-400000 Value type: Float	GES instance	1 minute
ges004_req uest_count	Request Quantity	Number of requests received by a graph instance Unit: count	≥ 0 Value type: Int	GES instance	1 minute
ges005_ave rage_respon se_time	Average Response Time	Average response time of requests received by a graph instance Unit: ms	≥ 0 Value type: Int	GES instance	1 minute
ges006_min _response_t ime	Minimum Response Time	Minimum response time of requests received by a graph instance Unit: ms	≥ 0 Value type: Int	GES instance	1 minute
ges007_ma x_response_ time	Maximum Response Time	Maximum response time of requests received by a graph instance Unit: ms	≥ 0 Value type: Int	GES instance	1 minute

Metric ID	Metric	Description	Value Range	Monitore d Object	Monitor ing Period (Origin al Metric)
ges008_rea d_task_pen ding_queue _size	Length of the Waiting Queue for Read Tasks	Length of the waiting queue for read requests received by a graph instance. This metric is used to view the number of read requests waiting in the queue. Unit: count	≥ 0 Value type: Int	GES instance	1 minute
ges009_rea d_task_pen ding_max_ti me	Maximum Waiting Duration of Read Tasks	Maximum waiting duration of read requests received by a graph instance Unit: ms	≥ 0 Value type: Int	GES instance	1 minute
ges010_pen ding_max_ti me_ read_task_t ype	Type of the Read Task That Waits the Longest	Type of the read request that waits the longest in a graph instance. Refer to Table 9-6 to find the corresponding task name.	≥ 1 Value type: Int	GES instance	1 minute
ges011_rea d_task_runn ing_queue_ size	Length of the Running Queue for Read Tasks	Length of the running queue for read requests received by a graph instance. This metric is used to view the number of running read requests. Unit: count	≥ 0 Value type: Int	GES instance	1 minute

Metric ID	Metric	Description	Value Range	Monitore d Object	Monitor ing Period (Origin al Metric)
ges012_rea d_task_runn ing_max_ti me	Maximum Running Duration of Read Tasks	Maximum running duration of read requests received by a graph instance Unit: ms	≥ 0 Value type: Int	GES instance	1 minute
ges013_run ning_max_ti me_ read_task_t ype	Type of the Read Task That Runs the Longest	Type of the read request that runs the longest in a graph instance. You can find the corresponding task name in GES documentation.	≥ 1 Value type: Int	GES instance	1 minute
ges014_writ e_task_pen ding_queue _size	Length of the Waiting Queue for Write Tasks	Length of the waiting queue for write requests received by a graph instance. This metric is used to view the number of write requests waiting in the queue. Unit: count	≥ 0 Value type: Int	GES instance	1 minute
ges015_writ e_task_pen ding_max_ti me	Maximum Waiting Duration of Write Tasks	Maximum waiting duration of write requests received by a graph instance Unit: ms	≥ 0 Value type: Int	GES instance	1 minute

Metric ID	Metric	Description	Value Range	Monitore d Object	Monitor ing Period (Origin al Metric)
ges016_pen ding_max_ti me_ write_task_t ype	Type of the Write Task That Waits the Longest	Type of the write request that waits the longest in a graph instance. Refer to Table 9-6 to find the corresponding task name.	≥ 1 Value type: Int	GES instance	1 minute
ges017_writ e_task_runn ing_queue_ size	Length of the Running Queue for Write Tasks	Length of the running queue for write requests received by a graph instance. This metric is used to view the number of running write requests. Unit: count	≥ 0 Value type: Int	GES instance	1 minute
ges018_writ e_task_runn ing_max_ti me	Maximum Running Duration of Write Tasks	Maximum running duration of write requests received by a graph instance Unit: ms	≥ 0 Value type: Int	GES instance	1 minute
ges019 _running_m ax_time_ write_task_t ype	Type of the Write Task That Runs the Longest	Type of the write request that runs the longest in a graph instance. You can find the corresponding task name in GES documentation.	≥ 1 Value type: Int	GES instance	1 minute

Metric ID	Metric	Description	Value Range	Monitore d Object	Monitor ing Period (Origin al Metric)
ges020_co mputer_res ource_usag e	Computin g Resource Usage	Computing resource usage of each graph instance Unit: %	0-100 Value type: Float	GES instance	1 minute
ges021_me mory_usage	Memory Usage	Memory usage of each graph instance Unit: %	0-100 Value type: Float	GES instance	1 minute
ges022_iops	IOPS	Number of I/O requests processed by each graph instance per second Unit: count/s	≥ 0 Value type: Int	GES instance	1 minute
ges023_byt es_in	Network Input Throughp ut	Data input to each graph instance per second over the network Unit: byte/s	≥ 0 Value type: Float	GES instance	1 minute
ges024_byt es_out	Network Output Throughp ut	Data sent to the network per second from each graph instance Unit: byte/s	≥ 0 Value type: Float	GES instance	1 minute
ges025_disk _usage	Disk Usage	Disk usage of each graph instance Unit: %	0-100 Value type: Float	GES instance	1 minute
ges026_disk _total_size	Total Disk Size	Total data disk space of each graph instance Unit: GB	≥ 0 Value type: Float	GES instance	1 minute

Metric ID	Metric	Description	Value Range	Monitore d Object	Monitor ing Period (Origin al Metric)
ges027_disk _used_size	Disk Space Used	Used data disk space of each graph instance Unit: GB	≥ 0 Value type: Float	GES instance	1 minute
ges028_disk _read_throu ghput	Disk Read Throughp ut	Data volume read from the disk in a graph instance per second Unit: byte/s	≥ 0 Value type: Float	GES instance	1 minute
ges029_disk _write_thro ughput	Disk Write Throughp ut	Data volume written to the disk in a graph instance per second Unit: byte/s	≥ 0 Value type: Float	GES instance	1 minute
ges030_avg _disk_sec_p er_read	Average Time per Disk Read	Average time used each time when the disk of a graph instance reads data Unit: second	≥ 0 Value type: Float	GES instance	1 minute
ges031_avg _disk_sec_p er_write	Average Time per Disk Write	Average time used each time when data is written to the disk of a graph instance Unit: second	GES instance	GES instance	1 minute
ges032_avg _disk_queue _length	Average Disk Queue Length	Average I/O queue length of the disk in a graph instance Unit: count	≥ 0 Value type: Int	GES instance	1 minute

Dimensions

Table 9-5 Dimensions

Кеу	Value
instance_id	GES instance

Mapping Between Task Types and Names

Туре	Name
100	Querying a vertex
101	Creating a vertex
102	Deleting a vertex
103	Modifying a vertex property
104	Adding a vertex label
105	Deleting a vertex label
200	Querying an edge
201	Creating an edge
202	Deleting an edge
203	Modifying an edge property
300	Querying schema details
301	Adding a Label
302	Modifying a Label
303	Querying a Label
304	Modifying a property
400	Querying graph details
401	Clearing graphs
402	Incrementally importing graph data online
403	Creating graphs
405	Deleting graphs
406	Exporting graphs

Туре	Name
407	filtered_khop
408	Querying path details
409	Incrementally importing graph data offline
500	Creating a graph backup
501	Restoring a graph from a backup
601	Creating an index.
602	Querying an index
603	Updating an index
604	Deleting an index
700	Running the algorithm
800	Querying an asynchronous task

9.4 Managing Connections

After you create a graph instance, you can download the required SDK and driver and view the connection information of the graph.

Log in to the GES management console. In the navigation pane on the left, choose **Connection Management**.

Connection Manag	ement		
SDK and Driver	SDK and Driver $ \ensuremath{\mathfrak{D}}$	1985 V Download Heatory	al Vessel
Connection Inform	nation		
	Available Instance	CURRENTVERSION_GRAPH_x v	
	Private Network Address ③	192.168.0.77	
	Public Access Address	100.93.4.237	
	JDBC URL (Private Network)	http(s)://192.168.0.77/ges/v1.0/357	_x86_64_5/action?action_id=execute-cypher-query
	JDBC URL (Public Network)	http(s)://100.93.4.237/ges/v1.0/357	_x86_64_5/action?action_id=execute-cypher-query

Figure 9-9 Connection management page

Downloading SDK and Driver

Figure 9-10 SDK and driver

SDK and Driver					
	SDK and Driver	x86	V Downloa	d	Historical Version

Select the CPU architecture that the cluster supports and click **Download** to download the SDK.

- Download an SDK and driver
 - The SDK encapsulates the service plane APIs. You are advised to use the SDK to access graph instances.
 - You need to download the Cypher-JDBC driver for Cypher API access. For details, see Using Cypher JDBC Driver to Access GES.
- Select the CPU architecture supported by the cluster. Currently, **x86** and **Arm** are available. Click **Download** to download the SDK.
- Click **Historical Version** to view historical SDK and driver versions and CPU architecture of the driver. You can click **Download** in the **Operation** column to download the historical driver.

Connection Information

Figure 9-11 Instance information

	Available Instance	CURRENTVERSION_GRAPH_X V	
0	Private Network Address	192.168.0.77	
	Public Access Address ③	100.93.4.237	
	JDBC URL (Private Network)	http(s)://192.168.0.77/ges/v1.0/357	_GRAPH_x86_64_5/action?action_id+execute-cypher-query
	JDBC URL (Public Network)	http(s)://100.93.4.237/ges/v1.0/3571	_GRAPH_x86_64_5/action?action_id=execute-cypher-query

Select the name of a created graph instance to view the following information:

- **Private Network Address**: ECSs in the same private network can connect to the graph instance using the private network address.
- **Public Access Address**: You can use the public access address (EIP) to access the graph instance through the Internet. You can bind an EIP to or unbind one from a graph instance.
- JDBC URL (Private Network): Configure this parameter when the JDBC driver executor and the graph instance are in the same private network.
- JDBC URL (Public Network): Configure this parameter when the JDBC driver executor can access the graph instance (with an EIP bound) through the Internet.

10 Configuring Permissions

10.1 Configuring Granular Permissions

GES graph instances support granular permission control. You can set the traverse, read, and write permissions for specific properties of specific labels. You are allowed to manage these permissions of a specific label or property of a graph and grant them to a user group.

NOTE

- This function allows you to set granular permissions for memory edition graphs of version 2.2.21 or later and database edition graphs of version 2.4.0 or later. You can **upgrade a graph** of an earlier version to 2.2.21 or a later version and then set granular permissions.
- Configuring fine-grained permissions for the graph requires IAM user viewing permissions and **GES Manager** or higher permissions. If there is no IAM user viewing permission, refer to **User Details** to import IAM users.

Procedure

- 1. Before setting granular permissions, configure the user group first. For details, see **Configuring a User Group**.
- 2. In the navigation pane, choose **Granular Permissions > Permission Configuration**.
- 3. On the **Permission Configuration** page, you can view the graph name, permission status, latest enabling time, and operations that can be performed on a graph in the **Running** state.

Figure 10-1 Configuring granular permissions

Graph Engine Service	Permission Configuration ③			
Overview	Q. Select a property or enter a keyword.			
Graph Management	Graph Name 😣	Permission Status 🖯	Last Open Time \ominus	Operation
Data Migration Backup Management	assets_management_F48Kw7	Close	-	Enable Set
Metadata Management	ges_migtest	Close	Apr 02, 2024 18:47:49 GMT+08:00	Enable Set
Connection Management	power_supply_eJ3LCo	Close	-	Enable Set
Access Sandbox (2) Granular Permissions	ges_datamig_0403	Close	-	Enable Set
Permission Configuration	<pre>ges_datamig_arm_baiwan</pre>	Close	-	Enable Set
User Groups	Total Records: 5 10 V (1)			

D NOTE

- 1. Only graphs in the **Running** status are displayed on this page.
- 2. You can search for graphs by their names in the upper right corner of the page.
- 4. Select the graph for which you want to set permission and click **Set** in the **Operation** column. The **Set Permission** page is displayed. You can create metadata permissions and granular permissions on this page.

Figure 10-2 Setting permissions

URRENTVEI _5 / Set Permission				Drable
Metadata Write Permission ③				
Status Not create Create				
Creation Time -				
Users -				
Granular Permission Policy ③				
Create Policy			Pet	y Name v Q
Policy Name	Users	Creation Time	Operation	

5. Click **Create** under **Metadata Write Permission** to create permission. After the metadata write permission is created, all labels of the metadata can be modified.

Figure 10-3 Creating permission

CURRENT	H_x86_64_5 / Set Permission			
Metadata Wri	• Not create Create		Create The metadata write permission allows modification of all	×
Creation Time Users			metadata file labels. Cancel OK	
Granular Per	mission Policy ③			
Policy Name		Users	Creation	Time

- 6. Click **Create Policy** under **Granular Permission Policy** to set granular permissions for a graph. You can set label- and property-level graph permissions and grant them to user groups.
 - **Policy Name**: You can set a name or use the default name.
 - View: You can configure permissions in form or code view.
 - Permissions: You can select labels whose traversal permission will be granted to a certain group of users. You can set read and write permissions of the label properties.

NOTE

Figure 10-4 Configuring permissions

7. Click Save. The Set Permission page is displayed. You can view the created permission policy in the Granular Permission Policy pane.

Figure 10-5 Created policies

ranular Permission Policy () Create Palicy				Policy Name V	0
Policy Name	Users	Creation Time	Operation		
defxulf_20244613094632	0	Mar 13, 2024 09-48-36 GMT+08-00	Associate User Group Edit D	elete View	

 \times

8. Click Set in the Operation column to associate the created granular permission with a user group.

Figure 10-6 Associating with a user group

Associate User Group	Associate User Group						
Select user groups you want to associa	te permissions with.						
Enter a user group. V)	Select all					
<pre><script>alert(1)</script></pre>	1						
default_2023830164610	2						
default_1011	1						
<pre> <script>alert('11111')</script></pre>	1						
default_gjtest	2						
default_20211124174125	2						
default_mbc	1						
default_20211112153853	4						
default_2021112593350	2						
default_gjnew	1						
10 V Total Records: 17	< 1 2 >						

9. Click **OK**. On the **Granular Permission Policy** pane, you can view the number of users who have been granted the permission.

Cancel OK

Figure 10-7 Users granted the permission



10.2 User Groups

You can create and manage user groups, and check whether a user group has been associated with permissions.

The procedure is as follows:

- Before creating a user group, you need to understand the concept of the User Group.
- 2. On the **User Groups** page, click **Create User Group** in the upper right corner. The **Create User Group** page is displayed.

Figure 10-8 User groups

18	er Groups 💿					Create User Group
	C. Select a property or enter a keyword.					
	User Group (B	Granular Permission B	Users O	Creation Time B	Operation	
	(k_20240308_1	(i) No	1	Mar 08, 2024 19:00:28 GMT+08:00	Edit Delete	
	default_mic	S No	1	Nov 27, 2023 17:10:56 GMT+08:00	Edit Delete	
	default_1011	© No	1	Oct 11, 2023 08:59:16 GMT+08:00	Edit Delete	

- 3. Set the user group name and add group members.
 - Name: Set a name for the user group or use the default name.
 - Members: All IAM users created under your account are displayed in this area. Select members you want to add to the user group. The selected members are displayed on the right.
 - Click local next to User name/ID to select or deselect all the members on the current page.

D NOTE

If the IAM user is not found due to insufficient permissions, manually import the IAM user by referring to **User Details**.

Figure 10-9 Creating a user group

Create User Group				
Name	default_ges			
wenters	Anallable : 94	Enter a keyword. Q	Available : 10	Enter a keyword. Q.
	User nome/ID		User name/ID	Operation
	cover587938_1 00.666/207710005941102v0116	87982092	0xe587938_1 0xe6c2d77100e594192z	×
	201_02 0b1730556b006404118c01163	340a6c7	2ml_02 0b1790356b0044041185c0	×
	Otly_graphint_user 05b61a7b6a00xCaa105c01tc	4e930c3	only_graphint_user 05261a126660022a115c0	×
	ges_guest_resource 074889514380055218501100	8342414	ges_guest_resource 0740895143000512185c0	×
	ges_superviser_sesource_RB 074be1bef20042201t2c0103	MC lic01elic	ges_superviser_resource 074be1bef200d2201f2c0	×
	ges_schnin_without_ges_sch 021716476d2548100d34000	sin 422590	ges_schnin_without_ges 021719647ed25481008340	×
	ges_operator_without_ges_o 66147ce8222214c4c8bf7104a6	per 2040-02a	ges_operator_without_ge 66147cs82291464c9b779	×
	10 v Total Records: 104		only_metadatagetdetai_u	×
	< 1 2 3 4	1 5 6 11	10 V Total Records: 10 C	1 >

4. Click **Save** in the lower right corner. The user group is created. The created user group is displayed on the **User Groups** page. You can edit or delete the user group.

Figure 10-10 Available operations

User Groups 🕥				Create User Group
Q Select a property or enter a keyword.				G 0
User Group \varTheta	Granular Permission	Users O Creation Time O	Operation	
delault_ges	No	10 Mar 13, 2824 18.45.41 GMT+88.00	Edit Delete	
(p_20240300_1	© No	1 Mar 08, 2024 19:00 20 GMT+88:00	Edit Delete	
default_mbc	© No	1 Nev 27, 2023 17:10:56 GMT+08:00	Edit Delete	

NOTE

You are not allowed to delete user groups that have been associated with granular permissions.

10.3 User Details

You can view the granular permissions of all IAM users created within your account.

The procedure is as follows:

1. On the **User Permissions** page, click v next to the target username to view its fine-grained permissions.

Figure 10-11 Viewing granular permissions

User Permissions 💿				G	port IAM User
If you have the IAM user view permission, the data in	the following lable is obtained from the IAM user. If the IAM user data is not syn	chonized due to insufficient permissions, click Import IAM User in the upper rig	it corner to manually order the IAM user data.	User Name	v O
User Name	ID ID	User Group	Granular Permission	Number of Granular Permissions	
∧ el_ges,	764e94131a5542c98b590cca32cb3419	 -script-sleet(11111)-/script- default_0728 default_002111231 	Associated	2	
Graph : TenThousand_Charging defaut_20213112153138 TenThousand_Charging_schema					

2. Click the permission name to view the details.

Figure 10-12 Permission details

User Permissions ⑦			default_20232811112847			
If you have the IAM user view permission, the	a data in the following table is obtained from the IAM us	er. If the IAM user data is not synchronized due terrisufficien	Label-specific traverse permissions	Property-specif	ic permissions	
User Name	ID	User Group		Property	Read	Write
∧ ges admin new	0c13b1f13e00d3e81f8fc01f2d6eb4f9	default 1011 default 20221111151934 defaul	rate	userid	YES	YES
			user	gender	YES	YES
Graph : ges_arm_ Grap	h : ges_gi_multilabel Graph : ges_arm_bi	aiwan Graph : TenThousand_Charging	default	909	VES	VES
			movie	090	120	120
default 20232811112847		DEFAULT	occupation	YES	YES	
				Zip-code	YES	YES

3. If you do not have such permission, you can click Import IAM User in the upper right corner to manually import IAM users.

Figure 10-13 Importing IAM users

User Permissions ()					et VAN User
If you have the IAM user view permission, the data is	n the following table is obtained from the IAM user. If the IAM user data is not synchro	onized due to insufficient permissions, click import IAM User in the upper rig	tit corner to manually enter the I2M user data.	User Name	v a
User Name	10	User Group	Granular Permission	Number of Granular Permissions	
×	754e94181a5542c98c590cca52cb3419	<pre><script-alert(11111)< pre=""> default_0728 default_202111231</script-alert(11111)<></pre>	 Associated 	2	
v	0b179/356b00d4041ffbc01f6340a6c7	default_20221111151934 default_gen	None	0	

In the **Import IAM User** dialog box, enter the ID and username of the IAM user to be added and click **OK**. The system will add the IAM user to GES so that the IAM user can be selected in the user group.
Figure 10-14 Entering IAM user information

Import IAM U	ser	×
* IAM User ID		
	Contact your account administrator to obtain the ID of the IAM user to be added.	
★ IAM User Name		
	Contact your account administrator to obtain the IAM username to be added.	
	Cancel OK	

11 O&M Monitoring

11.1 Monitoring Metrics

By using the O&M monitoring function of the graph instance, you can check the instance status, available resources, and real-time resource consumption.

 Table 11-1 lists the monitoring metrics for GES.

Table 11-1 GES monitoring	metrics

Monitor ed Object	Metric	Description	Value Range	Monitor ing Period (Origin al Metric)		
Instance overview	Cluster Information	Size and CPU architecture	String	-		
metrics –	Cluster Capacity	Total and used vertices and edges, and usage	≥ 0	Real- time		
	Cluster Node	Node type, available quantity, and total quantity	≥ 0	Real- time		
	Cluster Request Statistics (only available for the memory edition)	Number of waiting and running read and write requests on an instance	≥ 0	Real- time		
Instance workloa d metrics	QPS	Number of requests processed by an instance per second	≥ 0	5 min		

Monitor ed Object	Metric	Description	Value Range	Monitor ing Period (Origin al Metric)
Resourc e	CPU Usage (%)	CPU usage of the active node	0%–100%	5 min
consum ption metrics	Memory Usage	Average memory usage of the active node	0%–100%	5 min
of graph instance s	Disk Usage	Average disk usage of the active node	0%–100%	5 min
	Disk I/O (KB/S)	Average disk I/O value of the active node	≥ 0KB/s	5 min
	Network I/O	Average network I/O value of the active node	≥ 0KB/s	5 min
Overvie	Node Name	Name of a node	String	-
W	CPU Usage (%)	CPU usage of a node	0%–100%	5 min
	Memory Usage (%)	Memory usage of a node	0%–100%	5 min
	Average Disk Usage (%)	Disk usage of a node	0%–100%	5 min
	IP Address	Service IP address of a node	String	5 min
	Disk I/O (KB/S)	Disk I/O of a node, in KB/s	≥ 0KB/s	5 min
	TCP Protocol Stack Retransmissio n Rate (%)	Retransmission rate of TCP packets per unit time	0%-100%	5 min
	Status	Status of a node	Running/ Faulty	5 min
Disks	Node Name	Name of a node	String	5 min
	Disk Name	Name of a disk on a node	String	5 min

Monitor ed Object	Metric	Description	Value Range	Monitor ing Period (Origin al Metric)		
	Disk Type	Type of the disk on the node	System disk/Data disk/Log disk/Swap partition disk/Backup disk/Storage disk/HyG storage disk	5min		
	Disk Capacity (GB)	Capacity of a disk on a node, in GB	≥ 0 GB	5 min		
	Disk Usage (%)	Disk usage of a node	0%-100%	5 min		
	Disk Read Rate (KB/S)	Disk read rate of a node, in KB/s	≥ 0KB/S	5 min		
	Disk Write Rate (KB/S)	Disk write rate of a node, in KB/s	≥ 0KB/S	5 min		
	I/O Wait Time (ms)	Average waiting time for each I/O request, in ms	≥ 0 ms	5 min		
	I/O Service Time (ms)	Average processing time for each I/O request, in ms	≥ 0 ms	5 min		
	I/O Usage (%)	Disk I/O usage of a host	0%–100%	5 min		
Network	Node Name	Name of a node	String	5 min		
S	NIC Name	Name of the NIC on a node	String	5 min		
	NIC Status	NIC status	Online/ Offline	5 min		
	NIC Speed	Working rate of a NIC, in Mbit/s	≥ 0	5 min		
	Received Packets	Number of packets received by a NIC	≥ 0	5 min		
	Transmitted Packets	Number of packets transmitted by a NIC	≥ 0	5 min		
	Lost Received Packets	Number of lost packets received by a NIC	≥ 0	5 min		

Monitor ed Object	Metric	Description	Value Range	Monitor ing Period (Origin al Metric)		
	Receive Rate (KB/S)	Number of bytes received by a NIC per unit time, in KB/s	≥ OKB/s	5 min		
	Transmit Rate (KB/S)	Number of bytes transmitted by a NIC per unit time, in KB/s	≥ OKB/s	5 min		
Perform ance	Cluster CPU Usage	Average CPU usage of the active node	0%–100%	5 min		
	Cluster Memory Usage	Average memory usage of the active node	0%–100%	5 min		
	Cluster Disk Usage	Average disk usage of the active node	0%–100%	5 min		
	Cluster Disk I/O	Average disk I/O of the active node	≥ 0KB/s	5 min		
	Cluster Network I/O	Average network I/O of the NIC of the active node	≥ 0KB/s	5 min		
	Tomcat Connection Usage	HTTP connection usage of the active node	0%–100%	5 min		
	Cluster Swap Disk Usage (only available for the memory edition)	Swap partition disk usage of the active node	0%-100%	5 min		
	JVM Heap Memory Usage	JVM heap memory usage of the active node	0%–100%	5 min		
	Read Requests in Running Queue	Number of running read requests on the current instance	≥ 0	5 min		
	(only available for the memory edition)					

Monitor ed Object	Metric	Description	Value Range	Monitor ing Period (Origin al Metric)		
	Read Requests in Blocked Queue (only available for the memory edition)	Number of blocked read requests on the current instance	≥ 0	5 min		
Real- Time	Request ID	ID of the current query request	String	Real- time		
Queries	Job Name	Name of the current query job	String	Real- time		
	Request Parameters	Request parameters for the current query	String	Real- time		
	Progress (only available for the memory edition)	Progress of the current query	0%–100%	Real- time		
	Blocking Duration (S) (only available for the memory edition)	Blocking duration of the current query, in seconds	≥ 0	Real- time		
	Started	Start time of the current query	String	Real- time		
	Ended	End time of the current query	String	Real- time		
	Running Duration	Running duration of the current query, in seconds	≥ 0	Real- time		
Historica l	Job ID	ID of a historical query job	String	Real- time		
Queries	Туре	Type of a historical query job	String	Real- time		
	Original Request	Original request for a historical query	String	Real- time		

Monitor ed Object	Metric	Description	Value Range	Monitor ing Period (Origin al Metric)		
	Status	Status of a historical query job	String	Real- time		
	Progress	Execution progress of a historical query job	0%–100%	Real- time		
	Start Time	Start time of a historical query job	String	Real- time		
	End Time	End time of a historical query job	String	Real- time		
	Running Result	Execution results of a historical query job	String	Real- time		

11.2 Graph Instance O&M Monitoring

GES offers a multi-dimensional O&M monitoring interface that guarantees the smooth operations of graph instances. This feature gathers, monitors, and analyzes disk, network, and OS metrics utilized by graph instances, along with key cluster performance metrics. It promptly identifies significant database faults and performance issues and provides recommendations to optimize and resolve them.

NOTE

- The graph instance O&M monitoring dashboard supports only memory edition graphs of version 2.3.17 or later and database edition graphs of version 2.4.8 or later.
- The ten-thousand-edge size is for development learning and does not support the O&M monitoring dashboard.

O&M Monitoring Page

- 1. Log in to the GES management console. In the navigation pane on the left, choose **Graph Management**.
- 2. In the graph list, locate the target graph instance, click **More** in the **Operation** column, and select **View Metric** to access the **Instance Overview** page. For details about monitoring metrics, see **Monitoring Metrics**.

Figure 11-1 Instance Overview page

nstance Overview																
Graph Cluster Sta	tus								Instance Resources	5						
Cluster Information			10	Junter Node					CPU asam 157%		NR UD 68 2369/5	Disk Usage 1.8%	DHR 1N	ate: 33.535	Network UD	A 12KBR
Туре	50	pecification		Туре		Availat	lato?/ald		CPU usee					CPU snapeTOP5		
Size (edges)	8	lion edge		CN		2/2								Node Name	CPU at	Hager(%)
CPU architecture	>5	10_64		ON		3/3			5					ges_00085364.ges.	18.89	
				DISPATCH		2/2			12					gen_c00095364-gen.	16.13	
Claster Capacity	Hanna (%)								2				-	gen_c00095364-gen.	15.9	
Costs stream	0								6					ges_00095364.ges.	9.53	
									2					ges_00095364.ges.	9.16	
Work Load																
000																
025																
1																
0.6																
0.4																
0.2																
2024.12.16 20.04.42	2024.12.17 203 00:58:42 05	N.12.17 2024.12.17 154542 11.04.42	2024.12.17 15:58:42	2024.12.17 20:54:42	2024.12.18 01:54:42	2024.12.18 05:40:42	2024.12.18 11:44:42	2024.12.10								

Instance Overview

On the **Instance Overview** page of a graph instance, you can check the graph instance status, real-time resource consumption, and service workload. The functions of these areas are as follows:

• Graph Cluster Status

Owner Olivertee Otertee

In this area, you can check the basic information, cluster capacity, and number of requests of the current graph instance.

- a. Cluster Information: includes graph size and CPU architecture.
- b. **Cluster Capacity**: includes the number of used and total vertices and edges, as well as the usage.
- c. Cluster Node: includes the number of available/total CNs/DNs.
- d. **Cluster Request Statistics** (only available for the memory edition): includes the number of waiting read requests, running read requests, waiting write requests, and running write requests.

Figure 11-2 Graph Cluster Status

Gr	aph Cluster Status									
L	Cluster Information			Cluster Capacity						
	Туре	Specification		Туре	Used	Used			Usage (%)	
	Size (edges)	Million-edge		Vertex	146		1200000		0	
	CPU architecture	x86_64	Edge 1659		1659	1200000			0	
Ē	Cluster Node	L	Cluster Request Statistics							
	Туре	Available/Total		Туре		Waiting	Run		ning	
	CN	3/3		Read request		0		0		
	DN	2/2		Write request		0		0		

• Instance Resources

In this area, you can check the resource usage of the current instance, including the CPU usage, disk I/O, disk usage, memory usage, and network I/O. You can click a resource metric to view its change trend in the last 72

hours and the top 5 nodes with the highest usage of the resource at the current time.

Figure 11-3 Instance Resources

Instance Resources				
CPU usage 9.53%	Disk I/O 66.42KB/S	Disk Usage 1.79%	Disk Usage 32.5	4%
Network I/O 5.37KB/S				
CPU usage			CPU usageTOP5	
			Node Name	CPU usage(%)
%			ges_c00895364	18.89
12			ges_c00895364	16.13
9			ges_c00895364	15.9
6			ges_c00895364	9.53
0	20241217 20241217	20241210	ges_c00895364	9.16
20:04:42 05:08:42	14:28:42 23:38:42	08:44:42		

Workload

In this area, you can check the change trend of the database service load metric QPS in the last 72 hours.

Figure 11-4 Workload



11.3 Monitoring

11.3.1 Nodes

In the navigation pane on the left of the O&M monitoring page, choose **Monitoring** > **Nodes**. The node monitoring page is displayed, showing the real-time consumption of nodes, memory, disks, disk I/O, and network I/O.

Overview

On the **Overview** page, you can check the key resources of a specified node based on the node name, including the node name, CPU usage (%), memory usage (%), average disk usage (%), IP address, disk I/O (KB/s), TCP protocol stack retransmission rate (%), network I/O (KB/s), node status, and node monitoring status.

Figure 11-5 Overview page

nstance	Overview																			
Gray	ph Cluster Sta	tus								Inst	ance Resourc	e5								
10	luster Information			110	Junter Node						CPU usage # 515		Dist 10 44 2365 5	Disk D	ann 18%	DHR 1N	ate 31.515	Netwo	0.00 6.32682	8
	Туре	Spe	ification		Туре		Availab	le/Total		CPU	UNICE						CPU seaseTOP5			
	Size (edges)	Bille	n-edge		CN		2/2										Node Name	(PU usage(%)	
	CPU architecture	\$55_	<u>64</u>		ON		3/3				5						ges_c00885364.pr	s. 1	8.89	
	and an and an and an				DISPATCH		2/2				12						ges_c00005364-ge	n. 1	6.13	
	Туре	Usago (%)									2					-	ges_c00005364-ge	n. 1	5.9	
	Oraph storage	0									-						ges_c00895364.ge	s 8	.53	
											0						ges_c00095364.ge	s	16	
											20104.42	4.08.42	12:28:42 25:3	4.42 04.44	42 12.48.42	,				
Wor	* Load																			
	OPS																			
q#s																				
	1																			
	0.0																			
	0.6																			
	0.4																			
	0.2																			
	2024.12.16 20.04.42	2024.12.17 2024 00:58:42 05:5	12.17 2024.12.17 642 11.04.42	2024.12.17 15:58:42	2024.12.17 20:54:42	2024.12.18 01:54:42	2024.12.18 06:40:42	2024.12.18 11:44:42	2024.12.10 16:44:42											

You can click **Monitor** on the right of the row where a specified node is located to access the monitoring overview page and check the performance metric topology of the node in a specified period.

The period options are **Last 1 hour**, **Last 3 hour**, **Last 12 hour**, **Last 24 hour**, and **Last 3 days**. If you stay on the page for a long time, you can click **Refresh** in the upper right corner to update the monitoring data.

Figure 11-6 Node monitoring page

). Select a property or enter a	a keyword.									0
ode Name 🖯	CPU Usage (%) 🖯		Memory Usage (%) 8	Average Disk Usage (%) Θ	IP Address (8)	Disk I/O (KB/S) 0	TCP Protocol Stack Re 0	Network I/O (KB/S) 8	Status 0	Monitor
geronen	-	8.81%	- 9.17N		192	65.45	•	1.94	Running	Monitor
n 💼 ges-dispat	-	9.53%	33.46%		192	66.1		5.11	Bunning	Monitor
s,ges-ds-1-1	-	16.13%	- 13.94%		192	153.48	+	170.15	8 Running	Monitor
garda 3-1	_	18.82%	- 13.5%	1.20%	192	129.42		209.99	8 Running	Monitor
ges-dispat	-	8.93%	32.99%	1.76%	192	54.79		4.32	8 Running	Monitor
garanan	-	9.95%	- 9.15%	1.95%	192	60.47	•	2.66	O Running	Monitor
ges-ds-2-1	_	15.9%	- 13.83%	1.29%	192	133.75		134.02	Running	Monitor

Disks

On the **Disks** tab page, you can check the real-time disk usage of a node based on the node name and disk name. The metrics include **Node Name**, **Disk Name**, **Disk Type**, **Disk Capacity (GB)**, **Disk Usage (%)**, **Disk Read Rate (KB/s)**, **Disk Write Rate (KB/s)**, **I/O Wait Time (ms)**, **I/O Service Time (ms)**, **I/O Usage (%)**, and **Monitor**.

The disk types include system disk, data disk, log disk, swap partition disk, backup disk, storage disk, and HyG disk.

Figure 11-7 Disks tab page

Not	ies												
	Overview	Disks	Networks										
	Q. Select a p	roperty or ent	er a keyword.										0
	Node Name	9	Disk Name \varTheta	Disk Type \varTheta	Disk Capacity (GB) 🖯	Disk Usage (%) 🕀	Disk Read Rate (KB θ	Disk Write Rate (KB 0	ND Wait Time (ma) 🕀	1/O Service Time (ma) 🕀	DO Usage (%) 🕀	Monitor	
	995, III III	1095-60	vda	System disk	50	· 5.47%	1.92	\$7.13	10.24	0	0.06	Monitor	
	905,	201-01	vdb	Data disk	99.95	0.21%	0.02	6.41	2.38	0	0.09	Monitor	
	pri,	pada	vda	System disk	50	· 5.59%	1.95	52.52	5.74	a	0.06	Monitor	
	995_	yes-48	vdb	Log disk	99.95	0.17%	0.03	0.95	1.09	0	0.09	Monitor	
	901.	ges-dis	vdc	Data disk	99.95		0.02	10.6	11.8	0	0.09	Monitor	
	993,	201-da	vda	System disk	50	• 5.4PS	1.96	37.47	5.85	0	0.14	Monitor	
	90.	pasda	vdb	Data dak	99.95	4.30%	0.02	35.39	472	0	0.12	Monitor	
	995_	yes-da	vdo	Storage disk	129.94	0.28%	0.05	13.87	0.81		0.09	Monitor	
	90.	gen-dn	vdd	Storage disk	129.94	0.27%	0.04	13.47	0.74	0	0.05	Monitor	
	94. H	pesde	vde	Storage disk	129.94	0.27%	0.04	13.47	0.77	0	0.00	Monitor	
	Total Records: 2	8 10 %	< 1 2 3 →										

You can click **Monitor** on the right of the row where a specified node is located to access the monitoring overview page and check the performance metric topology of the disk in a specified period.

The options are **Last 1 hour**, **Last 3 hour**, **Last 12 hour**, **Last 24 hour**, and **Last 3 days**. If you stay on the page for a long time, you can click **Refresh** in the upper right corner to update the monitoring data.

Figure 11-8 Disks page

ick Canadity (CP)	Dick (Jongo (%))
(B)	Disk osage (%)
2	- 6
** 	4
17	3
20	2
10	
k Read Rate (KB/S)	Disk Write Rate (KB/S)
KE/S	12/5
3	70
25	60
2	- 40
1.5	
	20
	10
2024.12.18 05:0642 2024.12.18 07:34:42 2024.12.18 09:5642 2024.12.18 12:24:42 2024.12.18 14:464542	2024.12.18.050642 2024.12.18.07.5442 2024.12.18.09.5642 2024.12.18.12.2642 2024.12.18.14.4642
Wait Time (ms)	I/O Usage (%)
ms	%
14	0.05

D NOTE

According to the disk usage displayed on the page, the sum of the used disk space and available disk space is not equal to the total disk space. This is because a small amount of space is reserved in each default partition for system administrators to use. Even if common users have run out of space, system administrators can log in to the system and use their space required for solving problems.

The disk capacity is collected by running the **df** command on Linux. The following is an example:

[Ruby@host	10-0-16-43 8 1	0]# df ·	-x tmpfs -x	devtmpfs
Filesystem	1K-blocks	Used	Available	Use% Mounted on
/dev/sda4	569616888	5757444	540228616	2% /
/dev/sda2	999320	107584	822924	12% /boot
/dev/sdal	204580	8368	196212	5% /boot/efi
/dev/sdd	3513495364	390076	3513105288	1% /var/chroot/DWS/datal
/dev/sde	3513495364	274192	3513221172	1% /var/chroot/DWS/data2
/dev/sdb	3513495364	34224	3513461140	1% /var/chroot/DWS/data3
/dev/sdc	3513495364	34224	3513461140	1% /var/chroot/DWS/data4
Dubuchast	10 0 16 42 0 1	01#		

/dev/sda4: Used(5757444) + Available(540228616) != Total(569616888)

The parameters are as follows:

- **Filesystem**: path name of the device file corresponding to the file system. Generally, it is a hard disk partition.
- IK-blocks: number of data blocks (1,024 bytes) in a partition.
- Used: number of data blocks used by the disk.
- Available: number of available data blocks on the disk.
- **Use%**: percentage of the space used by common users. Even if the space is used up, the partition still reserves the space for system administrators.
- Mounted on: mount point of the file system.
- Networks

On the **Networks** tab page, you can check the real-time network resource consumption of a node based on the node and NIC name. The metrics include **Node Name**, **NIC Name**, **NIC Status**, **Lost Received Packets**, **Receive Rate** (KB/S), Transmit Rate (KB/S), and **Monitor**.

Figure 11-9 Networks tab page

Overview Disks N	etworks						
C. Select a property or enter a l	izyword.						00
Node Name Θ	NIC Name @	NIC Status ()	Lost Received Packets ()	Receive Rate (KB/S) 0	Transmit Rate (KB/S) 🕀	Monitor	
per,	eth0	celles	0	0	4	Monitor	
per,	eth1	celine	0	0	0	Monitor	
989.	ett2	celline	0			Monitor	
pes,	emo	coline	0		0	Monitor	
per,	eth1	celine	0	0	0	Monitor	
pes,	em2	celine	0	2	9	Monitor	
per,	eth0	celine	0	0	4	Monitor	
pro,	eth1	celine	0	0	0	Monitor	
ges,	em2	online	0	64	42	Monitor	
per,	otto	celine	0	0	4	Monitor	

You can click **Monitor** on the right of the row where a specified node is located to access the monitoring overview page and check the performance metric topology of the network in a specified period.

The options are Last 1 hour, Last 3 hour, Last 12 hour, Last 24 hour, and Last 3 days. If you stay on the page for a long time, you can click **Refresh** in the upper right corner to update the monitoring data.

Figure 11-10 Networks page

NIC Status			Received Packets ↑			
0 2024.12.18 16:18:42	2024.12.18 16:34:42 2024.12.18 16:48:42	2024.12.18 17/04-42	2024.12.18 16:18:42	2024,12,18 16:34-42	2024.12.13 164842	2024.12.18 17:04:42
Transmitted Packets 5,000,000 4,000,000 2,000,000 1,000,000 0 2,000,000 0 2,000,000 0 2,000,000 0 2,000,000 0 0 0 0 0 0 0 0	20412.01 10.0442 20412.11 1046.02	222412.18 17:04-12	Lost Received Packets	2024 12:18 10:34-62	2024.12.18 16-48-42	202412.18 17.0442
Receive Rate (KB/S) 12 10 8 6 6 4 2		22241218 170442	Transmit Rate (KB/S) 12 13 14 15 14 15 16 16 16 1 1 1 1 1 1 1 1 1 1 1 1 1 1	20041210 10:04-02	202412.30 19:44-42	20041218170442

11.3.2 Performance

In the navigation pane on the left of the O&M monitoring page, choose **Monitoring** > **Performance**. The performance monitoring page displays the trends of the following performance metrics:

- CPU Usage (%)
- Memory Usage (%)
- Disk Usage (%)
- Disk I/O (KB/s)
- Network I/O (KB/s)
- Tomcat Connection Usage (%)
- Swap Disk Usage (only available for the memory edition)
- JVM Heap Memory Usage
- Read Requests in Running Queue (only available for the memory edition)
- Read Requests in Blocked Queue (only available for the memory edition)

You can select a time range to check the performance trends within this range.

The options are Last 1 hour, Last 3 hour, Last 12 hour, Last 24 hour, and Last 3 days. If you stay on the page for a long time, you can click **Refresh** in the upper right corner to update the monitoring data.

Figure 11-11 Performance page



11.3.3 Real-Time Queries

In the navigation pane on the left of the O&M monitoring page, choose **Monitoring** > **Real-Time Queries**. The **Real-Time Queries** page is displayed, showing the real-time information about all queries running on the instance. The information includes **Request ID**, **Job Name**, **Request Parameters**, **Progress** (only available for the memory edition), **Blocking Duration (S)** (only available for the memory edition), **Started**, **Ended**, and **Running Duration**.

NOTE

The **Real-Time Queries** page for the database edition displays only Cypher queries.

Figure 11-12 Real-Time Queries page

 Read-Times Guerties
 v

 C. Solid s property or aller a largeout.
 C. Bits a property or aller a largeout.

 Regreent D 0
 Jub Rear 0
 Regreent D Properts 0
 Based 0
 Ended 0
 Burring Deation 0

11.3.4 Historical Queries

In the navigation tree on the left of the O&M monitoring page, choose **Monitoring** > **History Queries**. The **History Queries** page is displayed, showing details about historical asynchronous tasks running on the graph instance (the same as those displayed in the task center on the service plane).

Figure 11-13 Historical Queries page

Task Center gm_		v						
							Enter a job ID.	QC
jobID :	Type :	Original Request :	Status :	Progress : Start Time :	End Time :	Operations	Running Result	
				<u>(-)</u>				
				No records found.				

11.4 Monitoring Clusters Using Cloud Eye

This section describes metrics reported by GES to Cloud Eye as well as their namespaces, lists, and dimensions. You can use APIs provided by Cloud Eye to query the metric information generated for GES.

Namespace

SYS.GES

Monitoring Metrics

Metric ID	Metric	Description	Value Range	Monitored Object
ges001_vertex_ util	Vertex Capacity Usage	Vertex usage in a graph instance. The value is the ratio of used vertices to the total vertices. Unit: %	0–100 Type: float	GES instance
ges002_edge_ut il	Edge Capacity Usage	Edge usage of a graph instance. The value is the ratio of the used edges to the total edges. Unit: %	0–100 Type: float	GES instance
ges003_average _import_rate	Average Import Rate	Average rate of importing vertices or edges to a graph instance Unit: count/s	0– 400000 Type: float	GES instance
ges004_request _count	Request Quantity	Number of requests received by a graph instance Unit: count	≥ 0 Type: integer	GES instance
ges005_average _response_time	Average Response Time	Average response time of requests received by a graph instance Unit: ms	≥ 0 Type: integer	GES instance

Table 11-2 GES metrics

Metric ID	Metric	Description	Value Range	Monitored Object
ges006_min_res ponse_time	Minimum Response Time	Minimum response time of requests received by a graph instance Unit: ms	≥ 0 Type: integer	GES instance
ges007_max_res ponse_time	Maximum Response Time	Maximum response time of requests received by a graph instance Unit: ms	≥ 0 Type: integer	GES instance
ges008_read_ta sk_pending_que ue_size	Length of the Waiting Queue for Read Tasks	Length of the waiting queue for read requests received by a graph instance. This metric is used to view the number of read requests waiting in the queue. Unit: count	≥ 0 Type: integer	GES instance
ges009_read_ta sk_pending_ma x_time	Maximum Waiting Duration of Read Tasks	Maximum waiting duration of read requests received by a graph instance Unit: ms	≥ 0 Type: integer	GES instance
ges010_pending _max_time_ read_task_type	Type of the Read Task That Waits the Longest	Type of the read request that waits the longest in a graph instance. You can find the corresponding task name in GES documents.	≥ 1 Type: integer	GES instance
ges011_read_ta sk_running_que ue_size	Length of the Running Queue for Read Tasks	Length of the running queue for read requests received by a graph instance. This metric is used to view the number of running read requests. Unit: count	≥ 0 Type: integer	GES instance
ges012_read_ta sk_running_max _time	Maximum Running Duration of Read Tasks	Maximum running duration of read requests received by a graph instance Unit: ms	≥ 0 Type: integer	GES instance

Metric ID	Metric	Description	Value Range	Monitored Object
ges013_running _max_time_ read_task_type	Type of the Read Task That Runs the Longest	Type of the read request that runs the longest in a graph instance. You can find the corresponding task name in GES documentation.	≥ 1 Type: integer	GES instance
ges014_write_ta sk_pending_que ue_size	Length of the Waiting Queue for Write Tasks	Length of the waiting queue for write requests received by a graph instance. This metric is used to view the number of write requests waiting in the queue. Unit: count	≥ 0 Type: integer	GES instance
ges015_write_ta sk_pending_ma x_time	Maximum Waiting Duration of Write Tasks	Maximum waiting duration of write requests received by a graph instance Unit: ms	≥ 0 Type: integer	GES instance
ges016_pending _max_time_ write_task_type	Type of the Write Task That Waits the Longest	Type of the write request that waits the longest in a graph instance. You can find the corresponding task name in GES documents.	≥ 1 Type: integer	GES instance
ges017_write_ta sk_running_que ue_size	Length of the Running Queue for Write Tasks	Length of the running queue for write requests received by a graph instance. This metric is used to view the number of running write requests. Unit: count	≥ 0 Type: integer	GES instance
ges018_write_ta sk_running_max _time	Maximum Running Duration of Write Tasks	Maximum running duration of write requests received by a graph instance Unit: ms	≥ 0 Type: integer	GES instance

Metric ID	Metric	Description	Value Range	Monitored Object
ges019 _running_max_t ime_ write_task_type	Type of the Write Task That Runs the Longest	Type of the write request that runs the longest in a graph instance. You can find the corresponding task name in GES documentation.	≥ 1 Type: integer	GES instance
ges020_comput er_resource_usa ge	Computin g Resource Usage	Compute resource usage of each graph instance Unit: %	0–100 Type: float	GES instance
ges021_memor y_usage	Memory Usage	Memory usage of each graph instance Unit: %	0–100 Type: float	GES instance
ges022_iops	IOPS	Number of I/O requests processed by each graph instance per second Unit: count/s	≥ 0 Type: integer	GES instance
ges023_bytes_in	Network Input Throughp ut	Data input to each graph instance per second over the network Unit: byte/s	≥ 0 Type: float	GES instance
ges024_bytes_o ut	Network Output Throughp ut	Data sent to the network per second from each graph instance Unit: byte/s	≥ 0 Type: float	GES instance
ges025_disk_us age	Disk Usage	Disk usage of each graph instance Unit: %	0–100 Type: float	GES instance
ges026_disk_tot al_size	Total Disk Size	Total data disk space of each graph instance Unit: GB	≥ 0 Type: float	GES instance
ges027_disk_us ed_size	Disk Space Used	Used data disk space of each graph instance Unit: GB	≥ 0 Type: float	GES instance
ges028_disk_rea d_throughput	Disk Read Throughp ut	Data volume read from the disk in a graph instance per second Unit: byte/s	≥ 0 Type: float	GES instance

Metric ID	Metric	Description	Value Range	Monitored Object
ges029_disk_wri te_throughput	Disk Write Throughp ut	Data volume written to the disk in a graph instance per second Unit: byte/s	≥ 0 Type: float	GES instance
ges030_avg_dis k_sec_per_read	Average Time per Disk Read	Average time per disk read for a graph instance Unit: second	≥ 0 Type: float	GES instance
ges031_avg_dis k_sec_per_write	Average Time per Disk Write	Average time per disk write for a graph instance Unit: second	≥ 0 Type: float	GES instance
ges032_avg_dis k_queue_length	Average Disk Queue Length	Average I/O queue length of the disk in a graph instance Unit: count	≥ 0 Type: integer	GES instance

Dimensions

Кеу	Value
instance_id	GES instance

Mapping Between Task Types and Names

Table 11-3 Mapping between task types and names

Туре	Name
100	Querying vertices
101	Creating a vertex
102	Deleting a vertex
103	Modifying a vertex property
104	Adding a vertex label
105	Deleting a vertex label
200	Querying edges
201	Creating an edge

Туре	Name
202	Deleting an edge
203	Modifying an edge property
300	Querying schema details
301	Adding a label
302	Modifying a label
303	Querying a label
304	Modifying a property
400	Querying graph details
401	Clearing graphs
402	Incrementally importing graph data online
403	Creating a graph
405	Deleting a graph
406	Exporting graphs
407	filtered_khop
408	Querying path details
409	Incrementally importing graph data offline
500	Creating a graph backup
501	Restoring a graph from a backup
601	Creating an index
602	Querying indexes
603	Updating an index
604	Deleting an index
700	Running an algorithm

Viewing Instance Monitoring Information

- 1. Log in to the GES management console and choose **Graph Management**.
- 2. In the graph list, locate the row that contains the target graph, choose **More**, and select **View Metric** to access the Cloud Eye management console. By default, the graph instance monitoring information is displayed.

You can select a monitoring metric name and time range to check the performance curve.

Transferring Data to OBS

On Cloud Eye, raw metric data is only stored for two days. However, if you subscribe to OBS, you can synchronize the raw data and extend the storage period.

12 Package Management

GES offers package management, allowing you to easily access the page for purchasing, managing, or renewing your current package and learn about its usage.

The procedure is as follows:

1. Log in to the GES management console. In the navigation pane on the left, choose **Resource Packages**.

Figure 12-1 Package management

Graph Engine Service	Overview ③		
Overview	Process Flow		
Graph Management	-(1)	-(2)	(3)
Data Migration	Prepare	Import Metadata	Create Graph
Backup Management	Before using GES, you must create an account. Assign the GES policy to a	Get your metadata ready and then import the files into GES.	Use GES to create a graph and import data.
Metadata Management	user group and add a user to the group to give the user corresponding GES		
Task Center	permissions.		
Connection Management			
Access Sandbox 🕑			
Granular Permissions \sim	My Resources @		
Resource Packages			
Package Purchase	Creat Otatura C		Creat Olar C
Package Management 🖸	Graph Status ()	Preparing	Graph Size ()
Renewal Management 🕑		Preparing 0 expansion 0 Starting 0 Expanding 0 Stopping 0 Stopped 0 Upgrading 0 Frozen 0	Counts

2. To purchase a package, click **Package Purchase**. On the displayed page, set **CPU Architecture**, **Graph Specification (Edge Count)**, **Purchase Duration**, and **Quantity to Purchase** as needed and click **Next Step**.

Figure 12-2 Buying a package

014 < E	eh Management Ust / I kuy Graph Engine	buy Graph Engine Service Discount Packag	D4				Configuration	- (2) Configuration Confirmation
Bas Regi Pac CPU Crup (Edg	ic Configuration on Hage Specifications Architecture A Specification a Court	Cristen-Unreadd Cris	e same CPU antibiotes packages for each e same CPU antibiotes. Please devoise are <u>Manual CPU antibiotes devoises are</u> <u>Manual CPU antibiotes devoises are</u> technicates of the million edges can be obtained of the million edges.	ngue any tase effect for resources when to take	Integras Prese choice candida according to Control of the second s	No bodies of your resources.	The Malos edge The parkage for the Malos edges can reached be in the Malos edges.	
Dur	ation and Quantity							
Purc	hase Duration titly to Purchase	1 Month 2 Months	3 Months 4 Months 5	Months 6 Months 7 Ma	oths 8 Months 9 Manth	s 1 Year		
nce:	0							Next Step

- 3. On the configuration confirmation page, click **Proceed to Payment**. On the displayed payment page, select a payment method and complete the payment.
- 4. To manage existing packages, choose **Package Management**. On the displayed **Resource Packages** page, check the list of purchased resource packages, remaining quotas, and usage details.

Figure 12-3 Billing Center

Billing Center		Resource Packages				
Overview		1. If the required pay-genuse resource package is not found, the service does not provide such a package, or more than 18 months have package to the package expired. 2 For a backage expired. This is backage expired.				
Orders	۳	3. If your pay-per-use resource package expired, Huavie Coud charges you only for the resources that are actually used.				
Resource Packages		4. Lost Lenter prepares your resource package usage data or the last 12 months. Use Now				
Funds Management	Ŧ					
Billing	v	Resource Packages Remaining Quotas Usage Details				

5. To renew a package, choose **Renewals**. On the displayed **Renewals** page, renew your packages.

Figure 12-4 Renewals

Billing Center	1	Renewals									
Overview Orders Unpaid Orders		If you want to continue using any resources about to expire, refer to How Do I Renew Resources? Cannot Find the Resources to Be Renewel? and How Do I Change the Billing Mode from Yeany/Monthly to Pay-per-Use? Renewals or changes to pay-per-use will be applied after the current subcription form ends. If you want to an winh to hopped and the resource expire, refer to How Do I Renewals or Anteresto Pay-per-Use? If you want to mean winh to hopped and the resource expire, refer to Automatically Renewing a Resource all Setting a Renewal Date									
Renewals											
My Orders		Expires	Expires in 30 days	Expires in 15 days	Expires in 7 days	Custom	Status	Provisioned	Expired	Frozen	
Unsubscriptions and Returns/Exchanges		Do not	show resources that have o	orders pending payment	Do not show resource pa	ackages v					

13 Algorithms

13.1 Algorithm List

To meet the requirements of various scenarios, GES provides extensive basic graph algorithms, graph analytics algorithms, and graph metrics algorithms. The following table lists the algorithms:

Algorithm	Description
PageRank	PageRank, also known as web page ranking, is a hyperlink analysis algorithm used to rank web pages (nodes) based on their search engine results. PageRank is a way of measuring the relevance and importance of web pages (nodes).
PersonalRank	PersonalRank is also called Personalized PageRank. It inherits the idea of the classic PageRank algorithm and uses the graph link structure to recursively calculate the importance of each node. However, unlike the PageRank algorithm, to ensure that the access probability of each node in the random walk can reflect user preferences, the PersonalRank algorithm returns each hop to the source node at a (1-alpha) probability during random walk. Therefore, the relevance and importance of network nodes can be calculated based on the source node (the higher the PersonalRank value, the higher the correlation/ importance of the source node).
K-core	K-core is a classic graph algorithm used to calculate the number of cores of each node. The calculation result is one of the most commonly used reference values for determining the importance of a node so that the propagation capability of the node can be better understood.

Algorithm	Description
K-hop	K-hop is an algorithm used to search all nodes in the k layer that are associated with the source node through breadth-first search (BFS). The found sub-graph is the source node's ego-net. The K-hop algorithm returns the number of nodes in the ego- net.
Shortest Path	The Shortest Path algorithm is used to find the shortest path between two nodes in a graph.
All Shortest Paths	The All Shortest Paths algorithm is used to find all shortest paths between two nodes in a graph.
Filtered Shortest Path	This algorithm searches for the shortest path that meets the filter criteria between vertices. If there are multiple shortest paths, any one of them is returned.
SSSP	The SSSP algorithm finds the shortest paths from a specified node (source node) to all other nodes.
Shortest Path of Vertex Sets	The Shortest Path of Vertex Sets algorithm finds the shortest path between two vertex sets. It can be used to analyze the relationships between blocks in scenarios such as Internet social networking, financial risk control, road network transportation, and logistics delivery.
n-Paths	The n-Paths algorithm is used to find the <i>n</i> paths between two vertices on the k layer of a graph. It applies to scenarios such as relationship analysis, path design, and network planning.
Closeness Centrality	Closeness centrality is the average distance from a node to all other reachable nodes. It can be used to measure the time for transmitting information from this node to other nodes. A small Closeness Centrality within a node corresponds to a central location of the node.
Label Propagation	The Label Propagation algorithm is a graph-based semi- supervised learning method. Its basic principle is to predict the label information about unlabeled nodes using that of the labeled nodes. This algorithm can create graphs based on the relationships between samples. Nodes include labeled data and unlabeled data, and the edge indicates the similarity between two nodes. Node labels are transferred to other nodes based on the similarity. Labeled data is like a source used to label unlabeled data. Greater node similarity corresponds to an easier label propagation.
Louvain	Louvain is a modularity-based community detection algorithm with high efficiency and effect. It detects hierarchical community structures and aims to maximize the modularity of the entire community network.

Algorithm	Description
Link Prediction	This algorithm is used to calculate the similarity between two nodes and predict their relationship based on the Jaccard measurement method.
Node2vec	By invoking the Word2vec algorithm, the Node2vec algorithm maps nodes in the network to the Euclidean space, and uses vectors to represent the node characteristics. The Node2vec algorithm generates random steps from each node using the rollback parameter P and forward parameter Q . It combines BFS and DFS. The rollback probability is proportional to 1/P, and the forward probability is proportional to 1/Q. Multiple random steps are generated to reflect the network structures.
Real-time Recommenda tion	The Real-time Recommendation algorithm is based on the random walk model and is used to recommend nodes that are similar (have similar relationships or preferences) to the input node. This algorithm can be used to recommend similar products based on historical purchasing or browsing data or recommend potential friends with similar preferences.
Common Neighbors	Common Neighbors is a basic graph analysis algorithm that obtains the neighboring nodes shared by two nodes and further speculate the potential relationship and similarity between the two nodes. For example, it can intuitively discover shared friends in social occasions or products that interest both nodes in the consumption field.
Connected Component	A connected component stands for a sub-graph, in which all nodes are connected with each other. Path directions are involved in the strongly connected components and are not considered in the weakly connected components. NOTE This algorithm generates weakly connected components.
Degree Correlation	The Degree Correlation algorithm calculates the Pearson correlation coefficient between the source vertex degree and the target vertex degree of each edge. It is used to indicate whether the high-degree nodes are connected to other high- degree nodes in a graph.
Triangle Count	The Triangle Count algorithm counts the number of triangles in a graph without considering the edge directions. More triangles mean higher node association degrees and closer organization relationships.
Clustering Coefficient	The clustering coefficient is a measure of the degree to which nodes in a graph tend to cluster together. Evidence suggests that in most real-world networks, and in particular social networks, nodes tend to create tightly knit groups characterized by a relatively high density of ties.

Algorithm	Description
Betweenness Centrality	Betweenness centrality is a measure of centrality in a graph based on shortest paths. The Betweenness Centrality algorithm calculates shortest paths that pass through a vertex.
Edge Betweenness Centrality	The Edge Betweenness Centrality algorithm calculates shortest paths that pass through an edge.
Origin- Destination Betweenness Centrality	The Origin-Destination Betweenness Centrality algorithm calculates shortest paths that pass through a (an) vertex/edge, with the origin and destination specified.
Circle Detection with a Single Vertex	This algorithm solves a classic graph problem: detecting loops in a graph. Vertices on looped paths reflect the importance of the vertices. This algorithm is suitable for transportation analysis and financial risk control.
Common Neighbors of Vertex Sets	This algorithm obtains vertex set neighbors, that are, the intersection of two vertex sets (groups). They are objects that are associated with both sets, for example, common friends, common products of interest, and persons contacting with both communities. You can use neighbors to further speculate potential relationships and the degree of the connection between two vertices.
All Shortest Paths of Vertex Sets	This algorithm is used to discover all shortest paths between two vertex sets. It can be used to analyze the relationships between blocks in scenarios such as social networking, financial risk control, road networks and transportation, and logistics delivery.
Filtered Circle Detection	This algorithm searches for all circles that meet a specified filter criteria in the graph. It is applicable to scenarios such as detecting round-trip transfers and anti-money laundering for financial risk control, locating abnormal links in network routes, and risk identification in enterprise finance guarantee.
Subgraph Matching	This algorithm is used to find all subgraphs of a given small graph that is isomorphic to a given large graph. This is a basic graph query operation and is intended to explore important substructures of a graph.
Filtered All Pairs Shortest Paths	This algorithm is used to search for the shortest path between any two vertices in the graph that meets the condition. In a specific application scenario, you need to set a start vertex set (sources) and end vertex set (targets) as input for this algorithm. This algorithm returns the required shortest paths between the start and the end vertex sets.
Filtered All Shortest Paths	This algorithm allows you to search query results of the Shortest Path algorithm for the paths that meet the conditions between two vertices in a graph.

Algorithm	Description
TopicRank	The TopicRank algorithm is one of commonly used algorithms for ranking topics by multiple dimensions. For example, this algorithm is applicable to rank complaint topics obtained through a government hotline.
Filtered n- Paths	The filtered n-Paths algorithm is used to find no more than n k- hop loop-free paths between the source and target vertices. The start vertex (source), end vertex (target), number of hops (k), number of paths (n), and filter criteria (filters) are the parameters for the algorithm.
Temporal Paths	Different from path analysis on static graphs, the Temporal Paths algorithm combines the order of information transmission on dynamic graphs. The passing time of an edge on a path must be later than or the same as that of the previous edge, showing the increment (or non-decrement) of time.

13.2 PageRank

Overview

PageRank, also known as web page ranking, is a hyperlink analysis algorithm used to rank web pages (nodes) based on their search engine results. PageRank is a way of measuring the relevance and importance of web pages (nodes).

- If a web page is linked to many other web pages, the web page is of great importance. That is, the PageRank value is relatively high.
- If a web page with a high PageRank value is linked to another web page, the PageRank value of the linked web page increases accordingly.

Application Scenarios

This algorithm applies to scenarios such as web page sorting and key role discovery in social networking.

Parameter Description

Parameter	Mandat ory	Description	Туре	Value Range	Default Value
alpha	No	Weight coefficient (also called damping coefficient)	Double	A real number between 0 and 1 (excluding 0 and 1)	0.85

 Table 13-2
 PageRank algorithm parameters

Parameter	Mandat ory	Description	Туре	Value Range	Default Value
convergen ce	No	Convergence	Double	A real number between 0 and 1 (excluding 0 and 1)	0.00001
max_iterati ons	No	Maximum iterations	Int	1–2000	1000
directed	No	Whether an edge is directed	Bool	true or false	true

D NOTE

- **alpha** determines the jump probability coefficient, also called damping coefficient, which is a computing control variable in the algorithm.
- **convergence** indicates the upper limit of the sum of each absolute vertex change between an iteration and the last iteration. If the sum is less than the value of this parameter, the computing is considered converged and the algorithm stops.
- When the convergence is set to a large value, the iteration will stop quickly.

Precautions

When the convergence is set to a large value, the iteration will stop quickly.

Example

Select the algorithm in the algorithm area of the graph engine editor. For details, see **Analyzing Graphs Using Algorithms**.

Set parameters **alpha** to **0.85**, **coverage** to **0.00001**, **max_iterations** to **1,000**, and **directed** to **true**. The sub-graph formed by top nodes in the calculation result is displayed on the canvas. The size of a node varies with the PageRank values. The JSON result is displayed in the query result area.

13.3 PersonalRank

Overview

PersonalRank is also called Personalized PageRank. It inherits the idea of the classic PageRank algorithm and uses the graph link structure to recursively calculate the importance of each node. However, unlike the PageRank algorithm, to ensure that the access probability of each node in the random walk can reflect user preferences, the PersonalRank algorithm returns each hop to the source node at a **(1-alpha)** probability during random walk. Therefore, the relevance and importance of network nodes can be calculated based on the source node. (The higher the PersonalRank value, the higher the correlation/importance of the source node.)

Application Scenarios

This algorithm applies to fields such as product, friend, and web page recommendations.

Parameter Description

Paramet er	Mandato ry	Descriptio n	Туре	Value Range	Default Value
source	Yes	Node ID	String	-	-
alpha	No	Weight coefficient	Doubl e	A real number between 0 and 1 (excluding 0 and 1)	0.85
converge nce	No	Convergen ce	Doubl e	A real number between 0 and 1 (excluding 0 and 1)	0.00001
max_iter ations	No	Maximum iterations	Int	1–2000	1000
directed	No	Whether an edge is directed	Bool	true or false	true

Table 13	3-3	PersonalRank	alg	orithm	paramet	ters
----------	-----	--------------	-----	--------	---------	------

D NOTE

- **alpha** determines the jump probability coefficient, also called damping coefficient, which is a computing control variable in the algorithm.
- **convergence** defines the sum and upper limit of absolute values of each vertex in each iteration compared with the last iteration. If the sum is less than the value, the computing is considered to be converged and the algorithm stops.

Precautions

When the convergence is set to a large value, the iteration will stop quickly.

Example

Select the algorithm in the algorithm area of the graph engine editor. For details, see **Analyzing Graphs Using Algorithms**.

Set **source** to **Lee**, **alpha** to **0.85**, **convergence** to **0.00001**, **max_iterations** to **1000**, and **directed** to **true**. The sub-graph formed by top nodes in the calculation result is displayed on the canvas. The size of a node varies with the PersonalRank values. The JSON result is displayed in the query result area.

13.4 K-core

Overview

K-core is a classic graph algorithm used to calculate the number of cores of each node. The calculation result is one of the most commonly used reference values for determining the importance of a node so that the propagation capability of the node can be better understood.

Application Scenarios

This algorithm applies to scenarios such as community discovery and finance risk control.

Parameter Description

Parame	Mandat	Description	Тур	Value	Default
ter	ory		е	Range	Value
k	Yes	Number of cores The algorithm returns nodes whose number of cores is greater than or equal to k.	Int	Greater than or equal to 0	-

Table 13-4 K-core algorithm parameters

Precautions

None

Example

Set parameter \mathbf{k} to **10**. The sub-graph formed by nodes whose number of cores is greater than or equal to 10 in the calculation result is displayed on the canvas. The color of a node varies with the number of cores. The JSON result is displayed in the query result area.

13.5 K-hop

Overview

K-hop is an algorithm used to search all nodes in the k layer that are associated with the source node through breadth-first search (BFS). The found sub-graph is the source node's **ego-net**. The K-hop algorithm returns the number of nodes in the ego-net.

Application Scenarios

This algorithm applies to scenarios such as relationship discovery, influence prediction, and friend recommendation.

Parameter Description

Parame ter	Mandat ory	Description	Туре	Value Range	Default Value
k	Yes	Number of hops	Integer	1-100	-
source	Yes	Node ID	String	-	-
mode	No	 Direction: OUT: Hop from the outgoing edges. IN: Hop from the incoming edges. All: Hop from edges in both directions. 	String	OUT, IN, ALL	OUT

Table 13-5 K-hop a	gorithm parameters
--------------------	--------------------

Precautions

- A larger k value indicates a wider node coverage area.
- According to the six degrees of separation theory, all people in social networks will be covered after six hops.
- BFS searches information based on edges.

Example

Select the algorithm in the algorithm area of the graph engine editor. For details, see **Analyzing Graphs Using Algorithms**.

Calculate the sub-graph formed by the three hops starting from the Lee node.

Set parameters **k** to **3**, **source** to **Lee**, and **mode** to **OUT**. The sub-graph is displayed on the canvas, and the JSON result is displayed in the query result area.

13.6 Shortest Path

Overview

The Shortest Path algorithm is used to find the shortest path between two nodes in a graph.

Application Scenarios

This algorithm applies to scenarios such as path design and network planning.

Parameter Description

Paramet er	Mandat ory	Description	Туре	Value Range	Defau lt Value
source	Yes	Enter the source ID of a path.	String	-	-
target	Yes	Enter the target ID of a path.	String	-	-
directed	No	Whether an edge is directed	Bool	true or false	false
weight	No	Weight of an edge	String	 Empty or null character string Empty: The default weight and distance are 1. Character string: The attribute of the corresponding edge is the weight. When the edge does not have corresponding attribute, the weight is 1 by default. NOTE The weight of an edge must be greater than 0. 	-

Table	13-6	Shortest	Paths	algorithm	parameters
		011011000	1 010110	augorierini	parameters

Paramet er	Mandat ory	Description	Туре	Value Range	Defau lt Value
timeWin dow	No	Time window used for time filtering	Json	For details, see Table 13-7. NOTE timeWindow does not support the shortest path with weight. That is, parameters timeWindow and weight cannot be both specified.	-

Table 13-7 timeWindow parameters

Parame ter	Man dator y	Description	Тур е	Value Range	Def ault Valu e
filterNa me	Yes	Name of the time attribute used for time filtering	Stri ng	Character string: The attribute on the corresponding vertex/ edge is used as the time.	-
filterTy pe	No	Filtering by vertex or edge	Stri ng	V: Filtering by vertex E: Filtering by edge BOTH: Filtering by vertex and edge	BOT H
startTi me	No	Start time	Stri ng	Date character string or timestamp	-
endTim e	No	End time	Stri ng	Date character string or timestamp	-

Precautions

This algorithm only returns one shortest path.

Example

Calculate the shortest path from the Lee node to the Alice node.

Set parameters **source** to **Lee**, **target** to **Alice**, **weight** to **weights**, and **directed** to **false**. The shortest path is displayed on the canvas, and the JSON result is displayed in the result area.

13.7 All Shortest Paths

Overview

The All Shortest Paths algorithm is used to find all shortest paths between two nodes in a graph.

Application Scenarios

This algorithm applies to scenarios such as path design and network planning.

Parameter Description

Paramet er	Mandato ry	Description	Туре	Value Range	Default Value
source	Yes	Enter the source ID of a path.	String	-	-
target	Yes	Enter the target ID of a path.	String	-	-
directed	No	Whether an edge is directed	Bool	true or false	false

Table 13-8 All Shortest Paths algorithm parameters

Precautions

None

Example

Set parameters **source** to **Lee**, **target** to **Alice**, and **directed** to **false**. The calculation result is displayed on the canvas and the JSON result is displayed in the query result area.

13.8 Filtered Shortest Path

Overview

The Filtered Shortest Path algorithm is used to search for the shortest path that meets the filtering criteria between two vertices. If there are multiple shortest paths, any one of them is returned.

Application Scenarios

This algorithm applies to path design and network planning. It generates the shortest path based on vertex and edge filtering criteria.

Parameter Description

Paramet er	Mandat ory	Туре	Description
source	Yes	String	Enter the source vertex ID of a path.
target	Yes	String	Enter the target vertex ID of a path.
directed	No	Boole an	Whether to consider the edge direction The default value is false .

Table 13-9 Filtered Shortest Path algorithm parameters

Precautions

This algorithm only returns one shortest path.

13.9 SSSP

Overview

The SSSP algorithm finds the shortest paths from a specified node (source node) to all other nodes.

Application Scenarios

This algorithm applies to scenarios such as path design and network planning.

Parameter Description

Paramet er	Mandatory	Description	Туре	Value Range	Default Value
source	Yes	Node ID	Strin g	-	-
directed	No	Whether to consider the edge direction	Bool	true or false	true

 Table 13-10 SSSP algorithm parameters

Example

Calculate the shortest paths from the Lee node to other nodes.

Set parameters **source** to **Lee** and **directed** to **true**.

13.10 Shortest Path of Vertex Sets

Overview

The Shortest Path of Vertex Sets algorithm finds the shortest path between two vertex sets.

Application Scenarios

This algorithm applies to block relationship analysis in Internet social networking, financial risk control, road network transportation, and logistics delivery scenarios.

Parameter Description

Parame ter	Mandato ry	Descripti on	Туре	Value Range	Defa ult Value
sources	Yes	Source vertex ID set	String	The value is in the standard CSV format. IDs are separated by commas (,), for example, Alice, Nana . The maximum ID number is 100,000.	-
targets	Yes	Target vertex ID set	String	The value is in the standard CSV format. IDs are separated by commas (,), for example, Alice, Nana . The maximum ID number is 100,000.	-
directed	No	Whether an edge is directed	Bool	true or false	false
timeWin dow	No	Time window used for time filtering	Json	For details, see Table 13-12 .	-

 Table 13-11
 Shortest Path of Vertex Sets algorithm parameters
Parame ter	Man dator y	Description	Тур e	Value Range	Def ault Valu e
filterNa me	No	Name of the time attribute used for time filtering	Stri ng	Character string: The attribute on the corresponding vertex/ edge is used as the time.	-
filterTy pe	No	Filtering by vertex or edge	Stri ng	V: Filtering by vertex E: Filtering by edge BOTH: Filtering by vertex and edge	BOT H
startTi me	No	Start time	Stri ng	Date character string or timestamp	-
endTim e	No	End time	Stri ng	Date character string or timestamp	-

Table 13-12 timeWindow parameters

D NOTE

If a vertex ID contains commas (,), add double quotation marks to it. For example, when **Paris, je taime** and **Alice** IDs are used as sources, the ID set is "**Paris, je taime**",**Alice**".

Example

Set parameters **directed** to **true**, **sources** to "**Alice**,**Nana**", and **targets** to "**Lily**,**Amy**". The JSON result is displayed in the query result area.

13.11 n-Paths

Overview

The n-Paths algorithm is used to find the *n* paths between two nodes within the layers of relationships in a graph.

Application Scenarios

This algorithm applies to scenarios such as relationship analysis, path design, and network planning.

Parameter Description

Table 13-13 n-F	Paths algorithm	parameters
-----------------	-----------------	------------

Paramet er	Mandator y	Description	Туре	Value Range	Default Value
source	Yes	Enter the source ID of a path.	String	-	-
target	Yes	Enter the target ID of a path.	String	-	-
directed	No	Whether an edge is directed	Bool	true or false	false
n	No	Number of paths	Int	1-100	10
k	No	Number of hops	Int	1-10	5

Example

Set parameters **source** to **Lee**, **target** to **Alice**, **n** to **10**, **k** to **5**, and **directed** to **false**. The calculation result is displayed on the canvas and the JSON result is displayed in the query result area.

13.12 Closeness Centrality

Overview

Closeness centrality of a node is a measure of centrality in a network, calculated as the reciprocal of the sum of the length of the shortest paths between the node and all other reachable nodes in a graph. It can be used to measure the time for transmitting information from this node to other nodes. The bigger the node's **Closeness Centrality** is, the more central the location of the node will be.

Application Scenarios

This algorithm is used in key node mining in social networking.

Paramet er	Mandato ry	Description	Туре	Value Range	Default Value
source	Yes	Enter the ID of the node to be calculated.	String	-	-

Table 13-14 Closeness Centrality algorithm parameters

Example

Set parameter **source** to **Lee** to calculate the closeness centrality of the Lee node. The JSON result is displayed in the query result area.

13.13 Label Propagation

Overview

The Label Propagation algorithm is a graph-based semi-supervised learning method. Its basic principle is to predict the label information about unlabeled nodes using that of the labeled nodes. This algorithm can create graphs based on the relationships between samples. Nodes include labeled data and unlabeled data, and the edge indicates the similarity between two nodes. Node labels are transferred to other nodes based on the similarity. Labeled data is like a source used to label unlabeled data. The greater the node similarity is, the easier the label propagation will be.

Application Scenarios

This algorithm applies to scenarios such as information propagation, advertisement recommendation, and community discovery.

Paramete r	Mandato ry	Descripti on	Туре	Value Range	Default Value
convergen ce	No	Converge nce	Double	A real number between 0 and 1 (excluding 0 and 1)	0.00001
max_itera tions	No	Maximum iterations	Int	1–2000	1000

 Table 13-15
 Label Propagation algorithm parameters

Paramete r	Mandato ry	Descripti on	Туре	Value Range	Default Value
initial	No	Name of the property used as the initializati on label on a vertex	String	 Null or string Null: Each vertex is allocated with a unique initialization label. This method is applicable to scenarios where no vertex label information exists. 	-
				 Character string: The value of the property field corresponding to each vertex is used as the initialization label (the type is string, and the initialization label field is set to null for a vertex with unknown labels). This method is applicable to scenarios where some vertex labels are marked to predict unknown vertex labels. 	

Paramete r	Mandato ry	Descripti on	Туре	Value Range	Default Value
				NOTE If the value of initial is not null, the number of vertices with initialization labels must be greater than 0 and less than the total number of vertices.	

Precautions

Label Propagation uses IDs as labels by default.

Example

Set parameters **coverage** to **0.00001** and **max_iterations** to **1,000**, the sub-graphs with different labels are displayed on the canvas. The color of a node varies with labels. The JSON result is displayed in the query result area.

13.14 Louvain

Overview

Louvain is a modularity-based community detection algorithm with high efficiency and effect. It detects hierarchical community structures and aims to maximize the modularity of the entire community network.

Application Scenarios

This algorithm applies to scenarios such as community mining and hierarchical clustering.

Parameter	Mandat ory	Description	Туре	Value Range	Default Value
convergen ce	No	Convergence	Doubl e	A real number between 0 and 1 (excluding 0 and 1)	0.00001

Table 13-16 Louvain algorithm parameters

Parameter	Mandat ory	Description	Туре	Value Range	Default Value
max_iterat ions	No	Maximum iterations	Int	1–2000	100
weight	No	Weight of an edge	String	 Empty or null string Empty: The default weight and distance are 1. Character string: The attribute of the correspondin g edge is the weight. When the edge does not have correspondin g attribute, the weight is 1 by default. NOTE The weight of an edge must be greater than 0. 	weight

Precautions

This algorithm generates only the final community result and does not save the hierarchical results.

Example

Set parameters **coverage** to **0.00001** and **max_iterations** to **100**, the sub-graphs of different communities are displayed on the canvas. The color of a node varies with communities. The JSON result is displayed in the query result area.

13.15 Link Prediction

Overview

This algorithm is used to calculate the similarity between two nodes and predict their relationship based on the Jaccard measurement method.

Scenario

This algorithm applies to scenarios such as friend recommendation and relationship prediction in social networks.

Parameter Description

Paramet er	Mandator y	Description	Туре	Value Range	Default Value
source	Yes	Enter the source ID.	String	-	-
target	Yes	Enter the target ID.	String	-	-

 Table 13-17 Link Prediction algorithm parameters

Example

Set parameters **source** to **Lee** and **target** to **Alice** to calculate the association between two nodes. The JSON result is displayed in the query result area.

13.16 Node2vec

Overview

By invoking the Word2vec algorithm, the Node2vec algorithm maps nodes in the network to the Euclidean space, and uses vectors to represent the node characteristics.

The Node2vec algorithm generates random steps from each node using the rollback parameter **P** and forward parameter **Q**. It combines BFS and DFS. The rollback probability is proportional to 1/P, and the forward probability is proportional to 1/Q. Multiple random steps are generated to reflect the network structures.

Application Scenarios

This algorithm applies to scenarios such as node function similarity comparison, structural similarity comparison, and community clustering.

Parameter Description

Table	13-18	Node2vec	algorithm	parameters

Parame ter	Mandato ry	Description	Туре	Value Range	Defa ult Valu e
Ρ	No	Rollback parameter	Doubl e	-	1
Q	No	Forward parameter	Doubl e	-	1
dim	No	Mapping dimension	Int	1 to 200, including 1 and 200	50
walkLen gth	No	Random walk length	Int	1 to 100, including 1 and 100	40
walkNu mber	No	Number of random walk steps of each node.	Int	1 to 100, including 1 and 100	10
iteration s	No	Number of iterations	Int	1 to 100, including 1 and 100	10

Precautions

None

Example

Set parameters **P** to **1**, **Q** to **0.3**, **dim** to **3**, **walkLength** to **20**, **walkNumber** to **10**, and **iterations** to **40** to obtain the three-dimensional vector display of each node.

13.17 Real-time Recommendation

Overview

The Real-time Recommendation algorithm is based on the random walk model and is used to recommend nodes that are similar (have similar relationships or preferences) to the input node.

Application Scenarios

This algorithm can be used to recommend similar products based on historical purchasing or browsing data or recommend potential friends with similar preferences.

It is applicable to scenarios such as e-commerce and social networking.

Parame ter	Mandat ory	Description	Туре	Value Range	Defa ult Value
sources	Yes	Node ID. Multiple node IDs separated by commas (,) are supported (standard CSV input format).	Strin g	The number of source nodes cannot exceed 30.	-
alpha	No	Weight coefficient. A larger value indicates a longer step.	Dou ble	A real number between 0 and 1 (excluding 0 and 1)	0.85
N	No	Total number of walk Int 1–200,000 steps		1–200,000	10,00 0
nv	No	Parameter indicating that the walk process ends ahead of schedule: minimum number of access times of a potential recommended node NOTE If a node is accessed during random walk and the number of access times reaches nv , the node will be recorded as the potential recommended node.	Int	1-10	5
np	No	Parameter indicating that the walk process ends ahead of schedule: number of potential recommended nodes NOTE If the number of potential recommended nodes of a source node reaches np , the random walk for the source node ends ahead of schedule.	Int	1-2000	1000

Table	13-19	Real-time	Recommendat	tion algorith	nm parameters

Parame ter	Mandat ory	Description	Туре	Value Range	Defa ult Value
label	No	 Expected type of the vertex to be output. NOTE Expected type of the vertex to be output. If the value is null, the original calculation result of the algorithm is output without considering the vertex type. If the value is not null, vertices with the label are filtered from the calculation result. 	Strin g	Node label	-
directed	No	Whether to consider the edge direction	Bool	true or false	true

NOTE

alpha determines the jump probability coefficient, also called damping coefficient, which is a computing control variable in the algorithm.

Precautions

In the end conditions, the smaller the values of **nv** and **np**, the faster the algorithm ends.

Example

Set parameters **sources** to **Lee**, **alpha** to **0.85**, **N** to **10,000**, **nv** to **5**, **np** to **1,000**, **directed** to **true**, and **label** to null.

The sub-graph formed by top nodes in the calculation result is displayed on the canvas. The size of a node varies with the final scores. The JSON result is displayed in the query result area.

13.18 Common Neighbors

Overview

Common Neighbors is a basic graph analysis algorithm that obtains the neighboring nodes shared by two nodes and further speculate the potential relationship and similarity between the two nodes. For example, it can intuitively discover shared friends in social occasions or products that interest both nodes in the consumption field.

Application Scenarios

This algorithm applies to scenarios such as e-commerce and social networking.

Parameter Description

Parame ter	Mandat ory	Description	Туре	Value Range	Default Value				
source	Yes	Enter the source ID.	String	-	-				
target	Yes	Enter the target ID.	String	-	-				

Table 13-20 Common Neighbors algorithm parameters

Precautions

None

Example

Set parameters **source** to **Lee** and **target** to **Alice**. The calculation result is displayed on the canvas and the JSON result is displayed in the query result area.

13.19 Connected Component

Overview

A connected component stands for a sub-graph, in which all nodes are connected with each other. Path directions are involved in the strongly connected components and are not considered in the weakly connected components. This algorithm generates weakly connected components.

Parameter Description

None

Example

Run the algorithm to calculate the connected component to which each node belongs. The JSON result is displayed in the query result area.

13.20 Degree Correlation

Overview

The Degree Correlation algorithm calculates the Pearson correlation coefficient between the source vertex degree and the target vertex degree of each edge. It is used to indicate whether the high-degree nodes are connected to other highdegree nodes in a graph.

Application Scenarios

This algorithm is often used to measure the structure features of a graph.

Parameter Description

None

Example

Run the algorithm to calculate the degree correlation of a graph. The JSON result is displayed in the query result area.

13.21 Triangle Count

Overview

The Triangle Count algorithm counts the number of triangles in a graph. More triangles mean higher node association degrees and closer organization relationships.

Application Scenarios

This algorithm is often used to measure the structure features of a graph.

Parameter Description

Paramet er	Manda tory	Description	Туре	Value Range
statistics	No	Whether to export only the total statistical result.	Boolea n	true or false . The default value is
	 true: Export only the statistical result. false: Export the number of triangles corresponding to each vertex. 			true.

Instructions

The edge direction and multi-edge situation are not considered.

Example

Enter **statistics = true**. The JSON result is displayed in the query result area.

13.22 Clustering Coefficient

Overview

The clustering coefficient is a measure of the degree to which nodes in a graph tend to cluster together. Evidence suggests that in most real-world networks, and in particular social networks, nodes tend to create tightly knit groups characterized by a relatively high density of ties. This algorithm is used to calculate the aggregation degree of nodes in a graph.

Application Scenarios

This algorithm is often used to measure the structure features of a graph.

Parameter Description

None

Instructions

The multi-edge situation is not considered.

Example

Run the algorithm to calculate the clustering coefficient of a graph. The JSON result is displayed in the query result area.

13.23 Betweenness Centrality

Overview

Betweenness centrality is a measure of centrality in a graph based on shortest paths. This algorithm calculates shortest paths that pass through a vertex.

Application Scenarios

The Betweenness Centrality algorithm can be used for tracing man-in-the-middle in social networks and risk control networks and identifying key vertices in transportation networks. This algorithm is widely used for social networking, financial risk control, transportation networking, and city planning.

Parame ter	Manda tory	Descriptio n	Туре	Value Range	Default Value
directed	No	Whether an edge is directed	Boolean	The value can be true or false .	true
weight	No	Weight of an edge	String	The value can be an empty string. If this parameter is left blank, the weight and distance of this edge are 1 by default. You can set this parameter to a property of the edge, and the property value will be the weight. If the edge does not have the specified property, the weight is 1 by default. NOTE The weight of an edge must be greater than 0 .	-
seeds	No	Vertex ID	String	If the graph is large, betweenness calculation can be slow. You can set seeds to the sampling nodes for approximate calculation. The more seeds nodes, the closer results to the accurate calculation. The number of vertices cannot be greater than 100,000.	-
k	No	Number of samples	Integer	If the graph is large, betweenness calculation can be slow. You can set k to randomly select k sampling vertices from the graph. The larger value, the closer results to the accurate calculation. The value cannot be greater than 100,000.	-

 Table 13-21
 Algorithm parameters

D NOTE

When you perform approximate betweenness calculation, either **seeds** or **k** must be specified. If both are specified, **seeds** vertices will be sampled by default and **k** will be ignored.

Precautions

None

Example

Set **weight="length"**, **directed=true**, **seeds ="Lee,Alice"** and view the result.

13.24 Edge Betweenness Centrality

Overview

The Edge Betweenness Centrality algorithm calculates shortest paths that pass through an edge.

Application Scenarios

The Edge Betweenness Centrality algorithm can be used for key relationship mining. It is applicable to social networking, financial risk control, transportation networking, and city planning.

Parame ter	Manda tory	Descriptio n	Туре	Value Range	Default Value
directed	No	Whether an edge is directed	Boolean	The value can be true or false .	true

Table 13-22 Algorithm parameters

Parame ter	Manda tory	Descriptio n	Туре	Value Range	Default Value
weight	No	Weight of an edge	String	The value can be an empty string. If this parameter is left blank, the weight and distance of this edge are 1 by default. You can set this parameter to a property of the edge, and the property value will be the weight. If the edge does not have the specified property, the weight is 1 by default. NOTE The weight of an edge must be greater than 0 .	
seeds	No	Vertex ID	String	If the graph is large, betweenness calculation can be slow. You can set seeds to the sampling nodes for approximate calculation. The more seeds nodes, the closer results to the accurate calculation. The number of vertices cannot be greater than 100,000.	-
k	No	Number of samples	Integer	If the graph is large, betweenness calculation can be slow. You can set k to randomly select k sampling vertices from the graph. The larger value, the closer results to the accurate calculation. The value cannot be greater than 100,000.	-

NOTE

When you perform approximate edge-betweenness calculation, either **seeds** or **k** must be specified. If both are specified, **seeds** vertices will be sampled by default and **k** will be ignored.

Precautions

None

Example

Set **weight="length"**, **directed=true**, **seeds ="Lee,Alice"** and view the result.

13.25 Origin-Destination Betweenness Centrality

Overview

The Origin-Destination Betweenness Centrality algorithm calculates shortest paths that pass through a vertex/edge, with the origin and destination (OD) specified.

Application Scenarios

OD Betweenness Centrality can be used for tracing man-in-the-middle in social networks and risk control networks and identifying key vertices in transportation networks. It is suitable for simulating busy transportation sections during peak hours. It is also widely used for social networking, financial risk control, transportation networking, and city planning.

Parameter Description

Parame ter	Manda tory	Descriptio n	Туре	Value Range	Default Value
directed	No	Whether an edge is directed	Boole an	The value can be true or false .	true
weight	No	Weight of an edge	String	The value can be an empty string. If this parameter is left blank, the weight and distance of this edge are 1 by default. You can set this parameter to a property of the edge, and the property value will be the weight. If the edge does not have the specified property, the weight is 1 by default. NOTE The weight of an edge must be greater than 0 .	-

 Table 13-23
 Algorithm parameters

Parame ter	Manda tory	Descriptio n	Туре	Value Range	Default Value
OD_pai rs	No	Pairs of OD vertices	String	The value must be in the standard CSV format. The start vertex (origin) and end vertex (destination) are separated by commas (,), and the start and end vertex pairs are separated by newline characters (\n), for example, Alice,Nana\nLily,Amy .	-
seeds	No	ID of the hot spot vertex	String	Data that will be imported when the data of OD vertex pairs is unknown. The value is in the standard CSV format. IDs are separated by commas (,), for example, Alice, Nana . A maximum of 30 IDs are allowed.	-
modes	No	Hot spot vertex type	String	 IN: The hot spot vertex ID is used as the start vertex ID. OUT: The hot spot vertex ID is used as the end vertex ID. 	-
attende es	No	Number of participant s at each hot spot in seeds	String	The value is in the standard CSV format. Numbers are separated by commas (,), for example, 10,20 . The value ranges from 1 to 1,000,000.	-

NOTE

When you perform approximate OD-Betweenness calculation, either **OD_pairs** or **seeds** must be specified. If both are specified, the **OD_pairs** vertices will be used for calculation by default and **seeds** will be ignored.

Precautions

None

Example

Ser **weight=length**, **directed=true**, **OD = Alice,Nana\nLily,Amy** and view the result.

13.26 Circle Detection with a Single Vertex

Overview

This algorithm solves a classic graph problem: detecting loops in a graph. The vertices on a loop (circle) are import.

Application Scenarios

This algorithm is widely used for transportation networking and financial risk control.

Parameter Description

Parameter	Man dato ry	Description	Туре	Value Range	Default Value
source	Yes	ID of the given vertex	String	-	-
min_circle_le ngth	No	Minimum circle length	Int	[3,15]	3
max_circle_le ngth	No	Maximum circle length. The value must be bigger than min_circle_len gth .	Int	[3,15]	10
limit_circle_n umber	No	Maximum number of circles you want to search for	Int	[1,100000]	100

Table 13-24 Algorithm parameters

13.27 Common Neighbors of Vertex Sets

Overview

The Common Neighbors of Vertex Sets algorithm can find common neighbors of two vertex sets, and intuitively discover an object jointly associated with both sets, for example, a common friend in a social occasion, a product that is of common interest, a person who has been contacted by community groups. In this way, the algorithm infers the potential relationship and degree of association between the vertex sets.

Application Scenarios

This algorithm applies to graph analysis such as relationship mining and product/ friend recommendations.

Parameter Description

Parameter	Mand atory	Descripti on	Туре	Value Range	Default Value
sources	Yes	Source vertex ID set	String	The value is in the standard CSV format. IDs are separated by commas (,), for example, Alice , Nana . The maximum ID	-
				number is 100,000.	
targets	Yes	Target vertex ID set	String	The value is in the standard CSV format. IDs are separated by commas (,), for example, Mike,Amy .	-
				The maximum ID number is 100,000.	

 Table 13-25
 Common Neighbors of Vertex Sets algorithm parameters

Precautions

None

Example

Enter **sources=Alice,Nana** and **targets=Mike,Amy**. The calculation result is displayed on the canvas and the JSON result is displayed in the query result area.

13.28 All Shortest Paths of Vertex Sets

Overview

The Shortest Path of Vertex Sets algorithm finds the shortest path between vertex sets.

Application Scenarios

This algorithm can be used to analyze relationships between blocks in scenarios such as Internet social networking, financial risk control, road network traffic, and logistics delivery.

Parameter Description

Param eter	Man dato ry	Descripti on	Туре	Value Range	Default Value
sources	Yes	Source vertex ID set	Strin g	The value is in the standard CSV format. IDs are separated by commas (,), for example, Alice, Nana .	-
				The maximum ID number is 100,000.	
targets	Yes	Target vertex ID set	Strin g	The value is in the standard CSV format. IDs are separated by commas (,), for example, Alice, Nana . The maximum ID number is 100,000.	-
directe d	No	Whether to consider the edge direction	Boole an	true or false . It is a Boolean value.	false

Table 13-26 All Shortest Paths of Vertex Sets algorithm parameters

Precautions

If a vertex ID contains commas (,), add double quotation marks to it. For example, when **Paris, je taime** and **Alice** IDs are used as sources, the ID set is "**Paris, je taime**",**Alice**".

Example

Set parameters **directed** to **true**, **sources** to "**Alice**,**Nana**", and **targets** to "**Lily**,**Amy**". The JSON result is displayed in the query result area.

13.29 Filtered Circle Detection

Overview

The Filtered Circle Detection algorithm finds all circles that meet the filter criteria.

Application Scenarios

The Filtered Circle Detection algorithm is applicable to scenarios such as cyclic transfer detection and anti-money laundering in financial risk control, abnormal connection detection in network routing, and loan risk identification in enterprise guarantee circles.

Paramet er	Ma nda tor y	Description	Туре	Value Range	Default Value
sources	No	Set of source vertex IDs to be queried	Strin g	-	The value is in the standard CSV format. IDs are separated by commas (,), for example, Alice, Nana .
n	No	Upper limit of the number of enumerated circles that meet the filter criteria	Int	[1,100000]	100
statistics	No	Whether to export the number of circles that meet the filter criteria	Bool ean	true or false	false
batch_nu mber	No	Number of source vertices for batch processing	Int	[1,1000]	10

Table 13-27 Falameter description	Table	13-27	Parameter	description
-----------------------------------	-------	-------	-----------	-------------

Paramet er	Ma nda tor y	Description	Туре	Value Range	Default Value
output_f ormat	No	Output format	Strin g	vertexId , edgeId , or edgeObject	edgeObject
filters	Yes	Filter criteria. Each element in the array corresponds to the filter criteria of each layer.	Json	-	-

13.30 Subgraph Matching

Overview

The subgraph matching algorithm is used to find all subgraphs of a given small graph that is isomorphic to a given large graph. This is a basic graph query operation and is intended to explore important substructures of a graph.

Application Scenarios

This algorithm is applicable to fields such as social network analysis, bioinformatics, transportation, crowd discovery, and anomaly detection.

Parameter Description

 Table 13-28
 Subgraph matching parameters

Name	Manda tory	Description	Туре	Value Range
edges	Yes	Edge set of the subgraph to be matched. The vertex ID must be a non- negative integer.	String	The value is in standard CSV format. The start and end vertices of an edge are separated by a comma (,), and edges are separated by a newline character (\n). For example, 1,2\n2,3 .

Name	Manda tory	Description	Туре	Value Range
vertices	Yes	Label of each vertex on the subgraph to be matched.	String	The value is in standard CSV format. Vertices and their labels are separated by commas (,), and labels are separated by newline characters (\n). For example, 1,BP \n2,FBP\n3,CP.
directed	No	Whether the graph is directed	Bool	The value can be true or false . The default value is true .
n	No	Maximum number of subgraphs to be searched for	Int	The value range is [1,100000]. The default value is 100 .
batch_num ber	No	Number of queries processed in batches each time	Int	The value range is [1,1000000]. The default value is 10000 .
statistics	No	Whether to display the number of all subgraphs that meet the conditions	Bool	The value can be true or false . The default value is false .

13.31 Filtered All Pairs Shortest Paths

Overview

The Filtered All Pairs Shortest Paths algorithm is used to search for the shortest path between any two vertices in the graph that meets the condition. In a specific application scenario, you need to set a start vertex set (**sources**) and end vertex set (**targets**) as input for this algorithm. This algorithm returns the required shortest paths between the start and the end vertex sets.

Application Scenarios

This algorithm applies to relationship mining, path planning, and network planning.

Name	Mand atory	Description	Туре	Value Range	Default
sources	Yes	Set of start vertex IDs. The value is in the standard CSV input format, that is, multiple vertex IDs are separated by commas (,).	Strin g	The number of source vertices cannot exceed 10,000. -	-
targets	Yes	Set of end vertex IDs. The value is in the standard CSV input format, that is, multiple vertex IDs are separated by commas (,).	Strin g	The number of target vertices cannot exceed 10,000. -	-
directed	No	Whether the edges are directed	Bool	The value can be true or false .	-
cutoff	No	Maximum length	Int	1-100	6
path_lim it	No	Maximum number of paths	Int	 For synchronous tasks: The value ranges from 1 to 100000. The default value is 100000. For asynchronous tasks: The value ranges from 1 to 1000000. The default value is 1000000. The default value is 1000000. 	100000/10 00000

Example

Configure the parameters as follows: **directed=true**, **sources="Alice,Vivian"**, **targets="Jay,Bonnie"**, and set the edge search condition **labelName=friends**. The shortest paths between each pair of start and end vertices are returned in JSON format.

13.32 Filtered All Shortest Paths

Overview

The Filtered All Shortest Paths algorithm allows you to search query results of the Shortest Path algorithm for the paths that meet the conditions between two vertices in a graph.

Application Scenarios

This algorithm applies to scenarios such as relationship mining, path planing, and network planning.

Parameter Description

Paramete r	Mand atory	Descrip tion	Туре	Value Range	Default Value
source	Yes	Source vertex ID	String	-	-
target	Yes	Target vertex ID	String	-	-
directed	No	Whethe r an edge is directed	Bool	The value can be true or false .	false

Table 13-30 Filtered All Shortest Paths algorithm parameters

Example

Configure the parameters as follows: **directed=true**, **source="Alice"**, **target="Jay"**, and set the search condition to **labelName=friends**. The results are returned in JSON format.

13.33 TopicRank

Overview

TopicRank algorithm is one of commonly used algorithms for ranking topics by multiple dimensions.

Application Scenarios

This algorithm is applicable to rank hot topics. For example, it can be used to rank complaint topics obtained through a government hotline.

Name	Ma nda tor y	Description	Туре	Value Range	Default
sources	Yes	Vertex ID. You can specify multiple IDs in CSV format and separate them with commas (,).	Strin g	Currently, a maximum of 100000 IDs are allowed.	-
actived_p	No	Initial weight of the source vertices	Dou ble	The value ranges from 0 to 100000.	1
default_p	No	Initial weight of a non-source vertices	Dou ble	The value ranges from 0 to 100000.	1
filtered	No	Whether to filter results	Bool ean	The value can be true or false .	false
only_neig hbors	No	Whether to display only the neighboring vertices of the sources	Bool ean	The value can be true or false .	false
alpha	No	Weight coefficient	Real num ber	A real number between 0 and 1	0.85
converge nce	No	Convergence	Real num ber	A real number between 0 and 1	0.00001

 Table 13-31
 TopicRank parameters

Name	Ma nda tor y	Description	Туре	Value Range	Default
max_iter ations	No	Maximum iterations	Posit ive integ er	The value ranges from 1 to 2000.	1000
directed	No	Whether the edges are directed	Bool ean	The value can be true or false .	true
num_thr ead	No	Number of threads	Posit ive integ er	1-40	4

Example

Specify

sources="20190110004349,20190129023326,20190107003294,20190129023391
", filtered = true, only_neighbors=true, alpha=0.85, converage=0.00001,
max_iterations=1000, directed=true, and label="Topic" to obtain the topic
ranking result.

13.34 Filtered n-Paths

Overview

The filtered n-Paths algorithm is used to find no more than n k-hop loop-free paths between the source and target vertices. The start vertex (source), end vertex (target), number of hops (k), number of paths (n), and filter criteria (filters) are the parameters for the algorithm.

Application Scenarios

Any network

Parameter Description

 Table 13-32 filtered_n_paths parameters

Parameter	Mandato ry	Descripti on	Туре	Value Range	Default Value
source	Yes	Source vertex	String	Internal vertices	None

Parameter	Mandato ry	Descripti on	Туре	Value Range	Default Value
target	Yes	Target vertex	String	Internal vertices	None
k	Yes	Number of hops	Int	[2,6]	2
n	Yes	Number of paths	Int	[1,1000]	1

13.35 Temporal Paths

Overview

Different from path analysis on static graphs, the Temporal Paths algorithm combines the order of information transmission on dynamic graphs. The passing time of an edge on a path must be later than or the same as that of the previous edge, showing the increment (or non-decrement) of time.

- Temporal paths do not meet transitivity: If there is one temporal path from the vertex i to the vertex j, and there is one temporal path from the vertex j to the vertex k, it does not indicate that there is one temporal path from the vertex i to the vertex k. So, in terms of solving a problem, solving a path on a dynamic graph is more complex than on a static graph, and the calculation is much more difficult. However, temporal path analysis is widely used in actual life, for example, calculating a travel route and simulating/searching for an information propagation path.
- Temporal Paths can be classified into Shortest, Foremost, and Fastest Temporal Paths based on the problem-solving objective.
 - Shortest Temporal Paths: indicates the temporal path with the shortest distance.
 - **Foremost Temporal Paths**: indicates the temporal path that reaches the target node as early as possible.
 - Fastest Temporal Paths: indicates the temporal path that takes the shortest time.

Application Scenarios

It is applicable to scenarios such as epidemic or disease transmission source tracing, information transmission and public opinion analysis, time sequence-based path planning, and fund circulation path.

 Table 13-33
 Temporal Paths parameters

Parameter	Mand atory	Descriptio n	Туре	Value Range	Default Value
source	Yes	Source vertex ID	String	-	-
targets	Yes	Target vertex ID set	String	The value is in CSV format. IDs are separated by commas (,), for example, Alice,Nana . The number of IDs cannot exceed 100,000.	1000
directed	No	Whether an edge is directed	Boolea n	The value can be true or false .	false
k	No	Maximum depth	Integer	1 to 100, including 1 and 100	3
strategy	No	Algorithm policy	String	 The value can be shortest, foremost, or fastest. (Note: fastest is not supported currently.) shortest: Runs the shortest temporal paths algorithm to return the temporal path with the shortest distance. foremost: Runs the foremost temporal paths algorithm to return the temporal path salgorithm to return the temporal path that reaches the target node as early as possible. fastest: Runs the fastest temporal paths algorithm to return the temporal path that reaches the target node as early as possible. fastest: Runs the fastest temporal paths algorithm to return the temporal path that takes the shortest time. 	shortes t

Paramete r	Mandat ory	Descriptio n	Туре	Value Range	Defau lt Value
start	Yes	Start time for dynamic analysis	Date/Integer	-	-
end	Yes	End time for dynamic analysis	Date/Integer	-	-
time_prop s	Yes	Time properties for dynamic analysis	Object	-	-

Table 13-34 dynamicRange description

Table 13-35 time_props description

Paramete r	Mand atory	Description	Туре	Value Range	Defau lt Value
stime	Yes	Name of the start time property	String	-	-
etime	Yes	Name of the end time property	String	-	-

Precautions

Temporal path analysis needs to be performed on dynamic graphs.

Example

Select the algorithm in the algorithm area of the graph engine editor. For details, see **Analyzing Graphs Using Algorithms**.

- 1. To set the dynamic time range parameters, run the following command: start=1646092800, end =1646170716, stime="startTime", etime="endTime"
- 2. Set the parameters of the temporal paths algorithm. source="Person00014"

targets="Person00055,Person00058,Person00052,Person00061,Person00060,Pl ace00032,Place00016,Place00026,Place00015,Place00043" directed="false"

unecteu- Tais

k="5"

3. Select the algorithm search policy **shortest** or **foremost**. Click **Run** to run the temporal paths algorithm. The graph engine calculates and returns the temporal analysis path based on the selected algorithm search policy. The path dynamically extends with the time axis until it reaches the target node. The JSON results are displayed in the query result area.