# Cloud Monitor

## Product Documentation

# Contents


Cloud Monitor ..... 5
- Getting Started ..... 5
  - Monitoring Overview ..... 5
  - Instance Grouping ..... 7
  - Cloud Services Monitoring ..... 8
- Operation Guide ..... 9
  - Obtaining Monitoring Views ..... 9
    - Metric Data Use Limits ..... 9
    - Getting Monitoring Views and Reports ..... 10
    - Obtaining Monitoring Data for Specific Metrics ..... 12
  - Dashboard ..... 13
    - Dashboard Overview ..... 13
    - Creating Dashboards ..... 14
    - Creating Chart ..... 15
    - Viewing Dashboards ..... 17
    - Template Variables ..... 19
    - Link Management ..... 20
    - JSON ..... 22
  - Grouping Instances ..... 28
  - Alarm Service ..... 31
    - Alarm Overview ..... 31
    - Alarm Policies ..... 33
      - Creating  Alarm Policies ..... 33
      - Trigger Condition Templates ..... 35
      - Associating and Dissociating Objects ..... 39
    - Alarm Notifications ..... 40
      - Creating and Managing Alarm Recipient Groups ..... 40
      - Alarm Callback ..... 43
      - Receiving Alarm Notifications Through a WeCom Group ..... 45
      - Notification Template ..... 47
      - Custom Callback Content Templates ..... 49
    - Viewing Alarm Records ..... 52
  - Event Center ..... 54
    - Event Center Overview ..... 54
    - Product Event List ..... 57
  - Cloud Product Monitoring ..... 61
    - Cloud Product Monitoring Metrics Overview ..... 61
    - CVM ..... 63
      - CVM Monitoring Metrics ..... 63
      - Installing the CVM Monitoring Component ..... 65
      - Common Causes and Solutions for Monitoring Components Going Offline ..... 66
    - Monitoring Metrics for TKE ..... 67
    - Monitoring Metrics for BMS ..... 75
    - Monitoring Metrics for CLB ..... 78
    - Monitoring Metrics for VPC ..... 79
    - Monitoring Metrics for DC ..... 82
    - Monitoring Metrics for CBS ..... 83
    - Monitoring Metrics for COS ..... 84
    - Monitoring Metrics for CSP ..... 87
    - Monitoring Metrics for CFS ..... 89
    - Monitoring Metrics for Redis® ..... 91
    - Monitoring Metrics for TDSQL for MySQL ..... 96
    - Monitoring Metrics for MongoDB ..... 102
    - Monitoring Metrics for CMQ ..... 105
    - Monitoring Metrics for Ckafka ..... 106
    - Monitoring Metrics for TSF ..... 111
    - Monitoring Metrics for TDMQ Pulsar ..... 113
    - Monitoring Metrics for TI-ONE Platform ..... 115
      - TI-ONE Task-Based Modeling ..... 115
      - TI-ONE Development Machine ..... 136
      - TI-ONE Online Services ..... 144
      - TI-ONE Resource Management ..... 164
  - Querying Monitoring Metrics: Guide for Calling the GetMonitorData API ..... 171
  - Querying Monitoring Metrics: Guide for Calling the GetMonitorDat ..... 179
  - Shipping Monitoring Metrics to Kafka ..... 186
- Best Practices ..... 191
  - Application of Instance Grouping in Alarms ..... 191
  - Best Practices for Monitoring Scenarios ..... 193
  - Configuring CVM Metric Alarms ..... 196
- Ops Management Guide ..... 198
  - Product Architecture ..... 198
    - Core Feature Components ..... 198
    - Product Architecture Diagram ..... 201
    - Deployment Architecture Diagram ..... 205
    - Business Flow Diagram ..... 207
    - Core Logic Overview ..... 208
  - Product Dependencies ..... 209
  - Troubleshooting ..... 213
    - Impact Scope of Failures ..... 213
    - Failure Recovery ..... 214
      - Scenario 1: Flink Task Exception Impacting Monitoring Data Continuity ..... 214
      - Scenario 2: ES Exception Impacting Monitoring Data Continuity ..... 218
  - Daily Inspection ..... 220
    - Cloud Product Inspection ..... 220
      - Viewing Inspection Items ..... 220
      - Viewing Inspection Tasks ..... 231
      - Viewing Health Check Results ..... 237
    - Cloud Monitor Inspection Item List ..... 240
    - Inspection Handling ..... 242

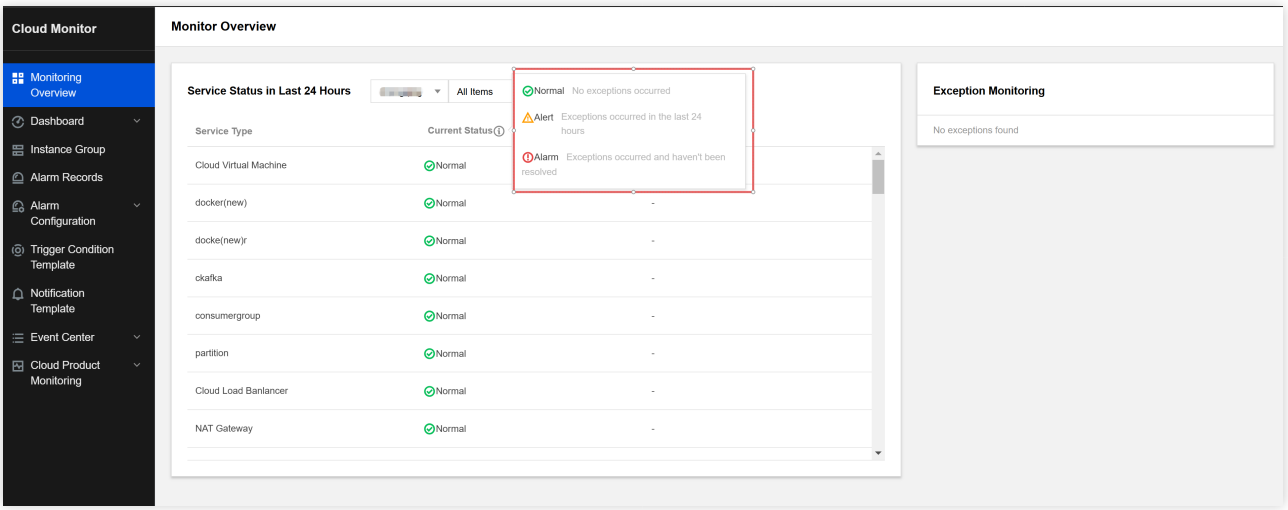
- Inspection of NWS Data Reception 242
- Inspection of Container Running Status 243
- Daily Monitoring 244
- Common Monitoring Metrics 244
- Alarm Handling 246
- Status of flink 246
- Emergency Response Plan 248
- Best Practices 250
- Node Restart 257
- Backup and Restoration 260
- Appendix A: Common Operations 262
- References 262
- Monitoring Tool Query and Usage 271
- Obtaining Operation Logs 272
- Log File 274
- User Guide 277
- Getting Started 277
- Monitoring Overview 277
- Instance Grouping 279
- Cloud Product Monitoring 280
- Operation Guide 281
- Obtaining Monitoring Views 281
- Metric Data Use Limits 281
- Obtaining Monitoring Views and Reports 282
- Obtaining Monitoring Data for Specific Metrics 284
- Alarm Service 285
- Alarm Overview 285
- Alarm Policies 287
- Creating Alarm Policies 287
- Trigger Condition Templates 289
- Associating and Dissociating Objects 293
- Alarm Notifications 294
- Creating and Managing Alarm Recipient Groups 294
- Alarm Callback 297
- Receiving Alarm Notifications Through a WeCom Group 299
- Notification Templates 301
- Custom Callback Content Templates 303
- Viewing Alarm Records 306
- Dashboard 308
- Dashboard Overview 308
- Creating Dashboards 309
- Creating Charts 310
- Viewing Dashboards 313
- Template Variables 315
- Link Management 316
- JSON 318
- Grouping Instances 324
- Event Center 327
- Event Center Overview 327
- Product Event List 330
- Cloud Product Monitoring 334
- Cloud Product Monitoring Metrics Overview 334
- CVM 336
- CVM Monitoring Metrics 336
- Installing the CVM Monitoring Component 338
- Common Causes and Solutions for Monitoring Components Going Offline 339
- Monitoring Metrics for TKE 340
- Monitoring Metrics for BMS 348
- Monitoring Metrics for CLB 351
- Monitoring Metrics for VPC 352
- Monitoring Metrics for DC 355
- Monitoring Metrics for CBS 356
- Monitoring Metrics for COS 357
- Monitoring Metrics for CSP 360
- Monitoring Metrics for CFS 362
- Monitoring Metrics for Redis® 364
- Monitoring Metrics for TDSQL for MySQL 369
- Monitoring Metrics for MongoDB 375
- Monitoring Metrics for CMQ 378
- Monitoring Metrics for Ckafka 379
- Monitoring Metrics for TSF 384
- Monitoring Metrics for TDMQ Pulsar 386
- Monitoring Metrics for TI-ONE Platform 388
- TI-ONE Task-Based Modeling 388
- TI-ONE Development Machine 409
- TI-ONE Online Services 417
- TI-ONE Resource Management 437
- Querying Monitoring Metrics: Guide for Calling the GetMonitorData API 444
- Shipping Monitoring Metrics to Kafka 452
- Best Practices 457
- Application of Instance Grouping in Alarms 457
- Best Practices for Monitoring Scenarios 459
- Configuring CVM Metric Alarms 462
- Product White Paper 464
- Product Overview 464
- Product Strengths 465
- Use Cases 466
- Product Architecture 467
- Features 468
- API Documentation 470
- Cloud Monitor(monitor) 470

- version(2018-07-24) ... 470
  - API overview ... 470
  - CallingMethod ... 473
    - signaturev1 ... 473
    - signaturev3 ... 481
    - request ... 490
    - result ... 491
    - PublicParameter ... 494
  - Alarm APIs ... 496
    - CreateAlarmPolicy ... 496
    - DescribeAlarmEvents ... 498
    - DescribeAlarmMetrics ... 499
    - DescribeAlarmPolicy ... 500
    - DescribeMiniDashboardAlarmInfo ... 502
    - ModifyAlarmPolicyInfo ... 504
    - ModifyAlarmPolicyStatus ... 506
  - Cloud Monitor API ... 508
    - BindingPolicyObject ... 508
    - CreateAlarmNotice ... 510
    - CreateNoticeContentTmpl ... 512
    - CreatePolicyGroup ... 514
    - DeleteAlarmNotices ... 515
    - DeleteNoticeContentTmpls ... 517
    - DeletePolicyGroup ... 519
    - DescribeAbnormalObjects ... 522
    - DescribeAccidentEventList ... 524
    - DescribeAgentStatusHistory ... 525
    - DescribeAlarmHistories ... 527
    - DescribeAlarmNotice ... 528
    - DescribeAlarmNoticeCallbacks ... 530
    - DescribeAlarmNotices ... 532
    - DescribeAlarmPolicies ... 533
    - DescribeAllNamespaces ... 535
    - DescribeBaseMetrics ... 536
    - DescribeBasicAlarmList ... 538
    - DescribeBindingPolicyObjectList ... 540
    - DescribeMonitorProducts ... 542
    - DescribeNoticeContentTmpl ... 544
    - DescribePolicyConditionList ... 546
    - DescribePolicyGroupInfo ... 549
    - DescribePolicyGroupList ... 553
    - DescribeProductEventList ... 556
    - DescribeProductHealthStatusList ... 557
    - DescribeProductList ... 559
    - DescribeStatisticData ... 561
    - DescribeUnifyDashboards ... 564
    - GetMonitorData ... 567
    - ModifyAlarmNotice ... 568
    - ModifyAlarmPolicyCondition ... 570
    - ModifyAlarmPolicyNotice ... 572
    - ModifyAlarmPolicyTasks ... 574
    - ModifyAlarmReceivers ... 575
    - ModifyNoticeContentTmpl ... 576
    - PutMonitorData ... 578
    - SetDefaultAlarmPolicy ... 580
    - UnBindingAllPolicyObject ... 582
    - UnBindingPolicyObject ... 585
  - Other APIs ... 587
    - DescribeLogTopics ... 587
  - Data structure ... 588
  - ErrorCode ... 619
- version(2017-03-12) ... 623
  - API overview ... 623
  - CallingMethod ... 624
    - signaturev1 ... 624
    - signaturev3 ... 632
    - request ... 641
    - result ... 642
    - PublicParameter ... 645
  - Cloud Monitor API ... 647
    - DescribeBaseMetrics ... 647
    - GetMonitorData ... 649
  - Data structure ... 651
  - ErrorCode ... 652

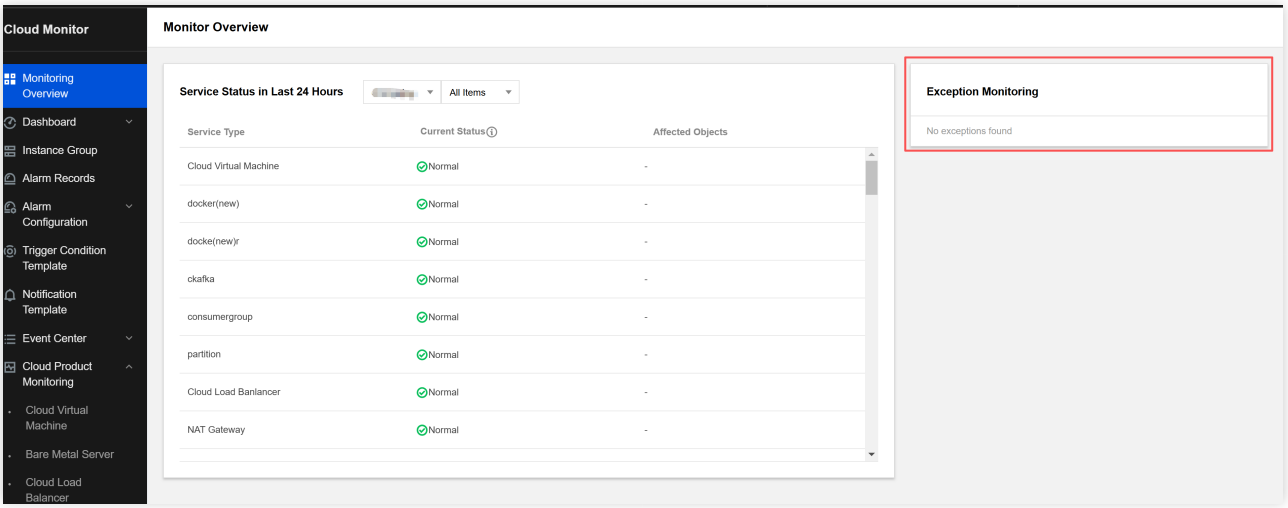
# Getting Started
# Monitoring Overview

The monitoring overview module provides an entry for viewing the overall monitoring status of cloud products. You can view the service health status from the past 24 hours, monitoring timeline from the past 7 days, exception monitoring status, and other information in this module.
Operation steps:

1. Log in to the cloud platform console and click Cloud Monitor to go to the Monitoring Overview page by default.
2. The Monitoring Overview page contains the following four modules.
    i. Service Health Status in the Last 24 Hours
    It displays the current three statuses of cloud resources, including Normal, Reminder, and In Alarm. The number of objects affected by abnormal cloud resources is also displayed.
    - Normal: no resource exceptions.
    - Reminder: Exceptions occurred in resources in the last 24 hours.
    - In Alarm: There are currently resources in an abnormal status that have not been restored.
    You can hover over the abnormal status icon to view the alarm exception records, and select the region where the cloud resource is located in the drop-down list.

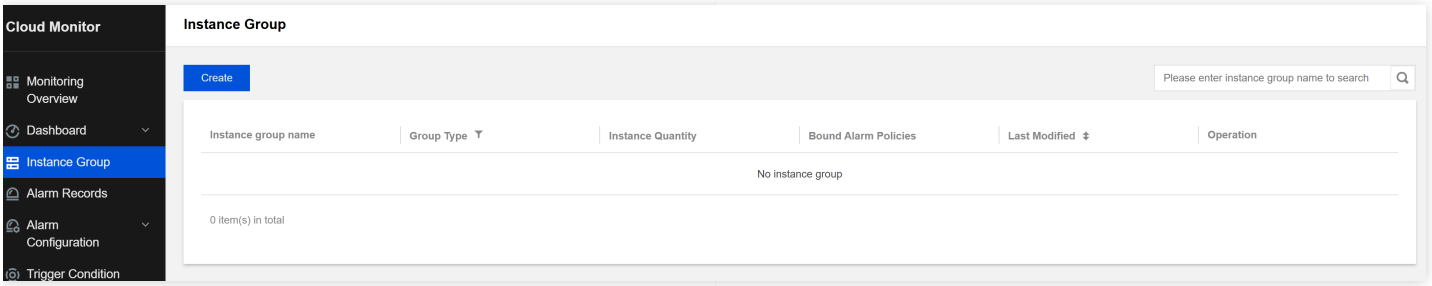



    ii. Exception Monitoring
    It displays the number of policies for which no alarm receiving group is configured under the current account. You can click Configure Reception Group to go to the alarm policy list for configuration.


Cloud Monitor
Monitor Overview
Monitoring Overview
Dashboard
Instance Group
Alarm Records
Alarm Configuration
Trigger Condition Template
Notification Template
Event Center
Cloud Product Monitoring
Cloud Virtual Machine
Bare Metal Server
Cloud Load Balancer
Service Status in Last 24 Hours
All Items
Service Type
Current Status
Affected Objects
Cloud Virtual Machine
Normal
docker(new)
Normal
docke(new)r
Normal
ckafka
Normal
consumergroup
Normal
partition
Normal
Cloud Load Banlancer
Normal
NAT Gateway
Normal
Exception Monitoring
No exceptions found

# Instance Grouping

You can group instances within the same cloud service across regions for unified management. When creating a large number of instances, you can create groups as needed, making it easier to manage instances, set alarm policies, and improve operational efficiency.



# Cloud Services Monitoring

Cloud Services Monitoring is a unified display platform within the Cloud Monitor console that aggregates the most detailed monitoring information. Users can view the list of cloud resources purchased under their account in Cloud Services Monitoring, as well as the corresponding monitoring and alarm details. Compared to the monitoring data presented in the business console, Cloud Services Monitoring integrates the instance's health status with alarm data. It provides a feature for users to quickly locate exception instances and supports the troubleshooting of issues by combining specific data and error information.

Monitoring details are displayed based on the cloud products you have purchased. Unpurchased cloud products will not be displayed. Currently supported Cloud Product Monitoring includes Cloud Virtual Machine (CVM), Cloud Load Balancer (CLB) , etc.

# Operation Guide
# Obtaining Monitoring Views
# Metric Data Use Limits

Metrics represent the performance data of user systems and are a core concept of Cloud Monitor. Metric data refers to the specific monitoring data of various cloud services, denoting a time-ordered collection of data points sent to Cloud Monitor. Statistical data about these data points can be retrieved as a time series. Currently, Cloud Monitor offers 1-minute and 5-minute data granularities, and the data storage durations for the two granularities are different.

| Granularity | Storage Duration |
| --- | --- |
| 1min | 30 days |
| 5min | 30 days |

Storage duration refers to the period from when Cloud Monitor collects the monitoring data until the data is terminated after reaching the corresponding period.

> Note:
>
> If a user's cloud resources are reclaimed or terminated, the monitoring data generated by the cloud resources will not be immediately cleared. Cloud Monitor continues to follow a unified storage duration to store and manage the monitoring data of terminated instances.

Users can view or retrieve the monitoring data generated by cloud resources under their accounts through the frontend console or API calls. If the resource object to be viewed has not been reclaimed or terminated, users can view the monitoring data in the frontend console. If the target instance no longer exists, users need to retrieve the corresponding monitoring data through API calls.

# Getting Monitoring Views and Reports

1. Viewing Monitoring Chart
   Cloud Monitor offers customers various channels to view monitoring data.

2. Viewing detailed monitoring data for a single instance

   i. Viewing Through Cloud Services Monitoring
   Cloud Service Monitoring is a unified display platform that gathers the most detailed monitoring information in the Cloud Monitor console. Users can view the list of cloud resources under their accounts and the corresponding monitoring and alert details.
   Compared to the monitoring data displayed on the business console, Cloud Service Monitoring integrates the health status and alarm data of instances. It provides users with the functionality to quickly locate abnormal instances and supports users in troubleshooting issues by combining specific data and exceptions.

   Cloud Product Monitoring currently supports displaying monitoring details for the following cloud services: CVM, CLB, etc.
   2. View Through Console
   Users can also log in to each cloud service's corresponding console to view the monitoring details of an instance.
   Using CVM as an example:

   - Users can quickly browse the performance data of each instance through the monitoring pop-up window in the list of cloud servers.
   - Users can also click on the CVM ID in the list to further view the instance's monitoring details.

3. Batch View of Cloud Service Monitoring Data

   i. View Through Dashboard
   View cross-instance and cross-product monitoring data through the dashboard, providing users with the feature to view multiple instance data at once.
   - Users can subscribe to the monitoring metrics they are most concerned about through the dashboard and keep track of the performance status of key instances through regular inspections or large screen displays of the dashboard.
   - Users can subscribe to the overall performance status of a business/cluster through the dashboard. Intuitively understand the overall situation of resources, avoiding the efficiency problem of checking monitoring data one by one.
     For specific instructions on using the dashboard, please refer to: Create New Dashboard

   ii. Viewing Through Traffic Monitoring

Traffic Monitoring provides users with the feature of monitoring the public network traffic of CVM, showing the aggregated data of the public network outbound bandwidth of all cloud servers under the user's account.Log in to the Cloud Observability Platform console, click Traffic Monitoring, and enter the management page.You can view real-time traffic, or switch the display time span through the time selector to view the traffic status of the last 7 days, last 15 days, last 30 days, or any specific day.

4. Obtaining Data Report

i. Exporting from Dashboard

Log in to the TCOP console, and navigate to Cloud Monitor Console > Dashboard > Dashboard List.

Click the corresponding dashboard.

On the dashboard page, find the monitoring item you want to export data for and click ... above it to export the corresponding instance monitoring detail list.

# Obtaining Monitoring Data for Specific Metrics

Cloud Monitor provides the following methods to support users in obtaining the specific metric data of a monitoring object at a specific time.

1. Obtain the specific metric data of a monitoring object at a specific time from the console.
    i. Log in to the cloud platform console and navigate to the corresponding product console where you want to view monitoring data.
    ii. Select the specific object you want to monitor and click the object ID to enter the monitoring details page. Alternatively, click the monitoring icon in the object list to view the data through the monitoring pop-up window.
    iii. Locate the target metric on the monitoring details page or in the pop-up window.
    iv. Use the time selector and granularity selector to adjust the view and display the specific content of the data. This allows you to see the specific metric data of the monitoring object at a specific time.
2. Obtain the specific metric data of a monitoring object at a specific time through an API.
   See Query Monitoring Metrics: GetMonitorData API Call Usage Guide.

# Dashboard
# Dashboard Overview

Dashboards are intelligent tools provided by Cloud Monitor for monitoring data of cloud product metrics, featuring visualization and analysis capabilities.
You can create a dashboard for cloud service metrics, and the dashboard will automatically present the monitoring data in a visually appealing chart format on the monitoring panel. This makes the data more intuitive and assists you in analyzing metrics through trends and exception values.

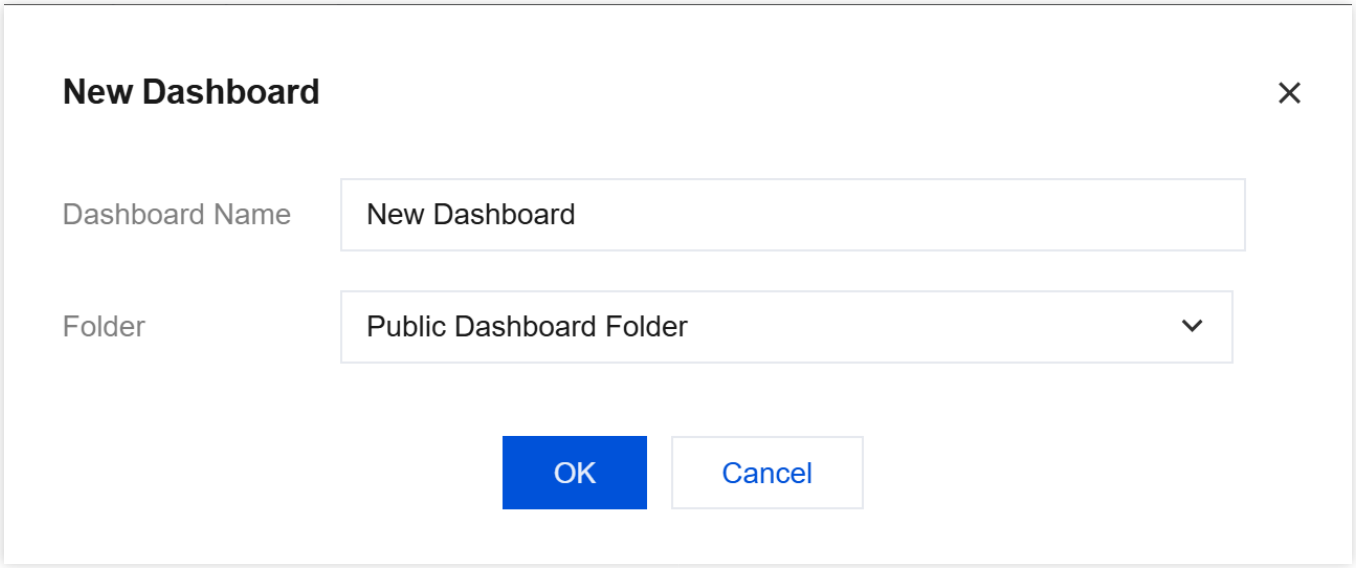
1. Use Cases
    - When receiving a metric alarm, you can use the dashboard to analyze the cause of the alarm.
    - After a new feature is released, you can use the dashboard to check for resource exceptions.
    - Through real-time metric display, you can optimize performance during business peaks.
    - View the loads through the dashboard to determine whether resource scaling is needed.
2. Limitations

| Category | Upper Limitation |
| --- | --- |
| Custom dashboard | 30 |
| Number of charts per dashboard | 20 |
| Number of instances that can be associated with each chart | 50 |

# Creating Dashboards

Before creating monitoring charts, you need to create a dashboard first. You can refer to the following steps to create a dashboard.

1. Log in to the cloud platform console and choose Cloud Monitor > Dashboard to enter the dashboard list.
2. Click Create Dashboard in the upper-left corner of the dashboard list to go to the dashboard creation page.
3. Click Create New Chart in the empty chart to go to the chart editing page. For details, see Creating a Chart.
4. Configure chart parameters. You can click <- in the upper-left corner to return to the dashboard creation page to add a chart.
5. After configuring the chart, click Save in the upper-right corner, and then fill in the dashboard name and choose the folder it belongs to in the pop-up box.




6. Click OK to create the dashboard.

# Creating Chart

You can follow the steps below to create a new chart.

1. Add New Metric

   i. Log in to the Cloud Platform console and click Cloud Monitor > Dashboard > Dashboard List.
   ii. Select the corresponding dashboard, and click Create New > Create New Chart.
   iii. Configure the metric information as follows:
   - Metric: Select the product type and metric.
   - Filtering: Choose the filtering conditions to filter and display the data on the chart that meets the criteria.
     Instance: The chart will display the monitoring data of the selected instances.
     Template Variables: The chart will display instances of template variable filters. See the template variable documentation for the configuration of template variables.
   - Comparison: Supports sequential comparison (the same period yesterday), year-on-year comparison (same period last week), and custom time comparison. When both options are selected, the chart will display the monitoring curves for the selected instance during the same time period of yesterday and last week, providing convenience for you to compare the data.
   - Left Y-axis, Right Y-axis: Supports adjusting the placement of the Y-axis on either side.
   - More configurations:
     Alias: Supports one-click renaming of all instance aliases. If you need different instances to have different aliases, you can create multiple metrics and enter an alias under each.
     Enable Sorting Feature: The instances bound to the chart will be sorted according to the sorting rules and display quantity, used to implement the high and low load features of a large number of monitoring machines.
     Sorting rules: Supports various methods of sorting metrics, which can be used to filter instances based on the sorting results.
     Display quantity: Display the number of instances. For example, set the collation rule as "maximum; descending order", and display quantity as 10. This means the chart will display the top 10 instances with the maximum value in descending order.

   iv. Configure chart information as follows:
   - Basic Information: Includes the chart name and chart annotation.
   - Chart Type: Select from line chart, bar chart, gauge chart, pie chart, numerical statistic chart, heatmap, or table. Customize the chart display method based on each chart type.

   v. After configuration, click the top-left [<-] to return to the Dashboard page to add the new chart configuration.

2. Manage multiple metrics

- You can click [Add Metric] or the "Copy" icon button to display multiple metrics in the same chart for cross-instance metric data comparison.
- You can click the up and down arrows to adjust the metric order.
- You can click the "Delete" icon button to remove metrics.

3. Editing a chart

i. Click the [...] in the upper right corner of the chart you need to edit.
ii. Click [Edit] in the popup to enter the chart editing page.

# Viewing Dashboards

You can refer to the following steps to view a dashboard.

1. Operation Steps
   i. Log in to the cloud platform console and choose Cloud Monitor > Dashboard to enter the dashboard list.
   ii. By clicking the corresponding dashboard, you can view the dashboard overview. The system allows you to view a dashboard in full-screen mode, customize the time span, and bookmark panels.
2. View a chart in full-screen mode.
   Click the full screen icon in the upper-right corner of a chart to view it in full-screen mode.
   To exit full-screen mode, press ESC or click the X icon in the upper-right corner.
3. Zoom and move a chart.
   - Chart zooming: You can scale a chart by hovering your mouse pointer over the lower-right corner of the chart until a straight-angle icon appears, as shown below.
   - Chart moving: You can move a chart by hovering your mouse pointer over the name of the chart until a movement icon appears, as shown below.
4. View monitoring data at a specific time.
   You can hover your mouse pointer over the monitoring chart to view the monitoring data at a specific time.
5. Adjust the time span of charts to view monitoring data.
   By default, dashboards display the data of the last 12 hours.
   You can adjust the data interval and granularity for all charts in the dashboard by using the time selection control in the upper-right corner of the dashboard. This feature allows users to review historical monitoring data and troubleshoot issues.
   Comparison table of time periods and granularities in charts

| Time Range | Default Statistical Granularity |
| --- | --- |
| <=1 hour | 1 minute |
| (1 hour, 12 hours] | 1 minute |
| (12 hours, 3 days] | 5 minutes |
| (3 days, 30 days] | 1 hour |
| (30 days, 186 days] | 1 day |

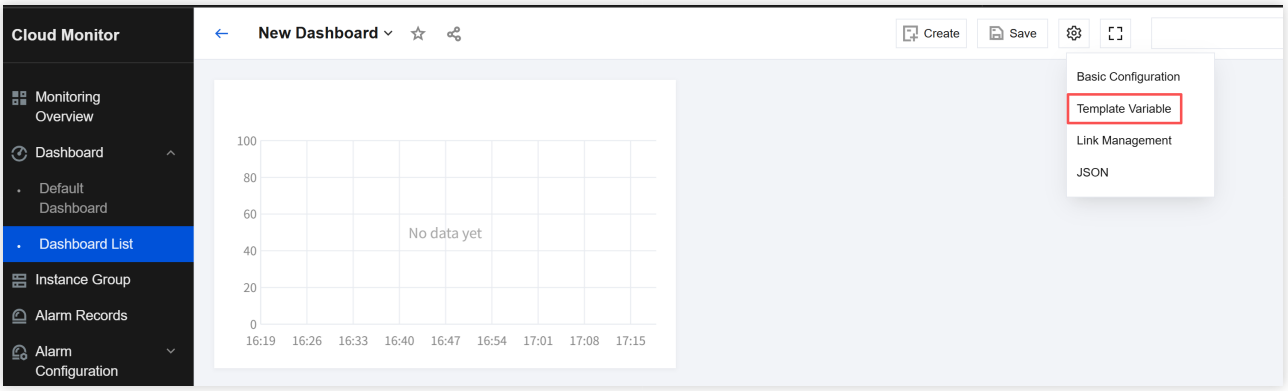
6. Switch dashboard.

Click the inverted triangle button on the right of the current dashboard name to switch between panels and view by current folder, recently visited, favorites, or all.

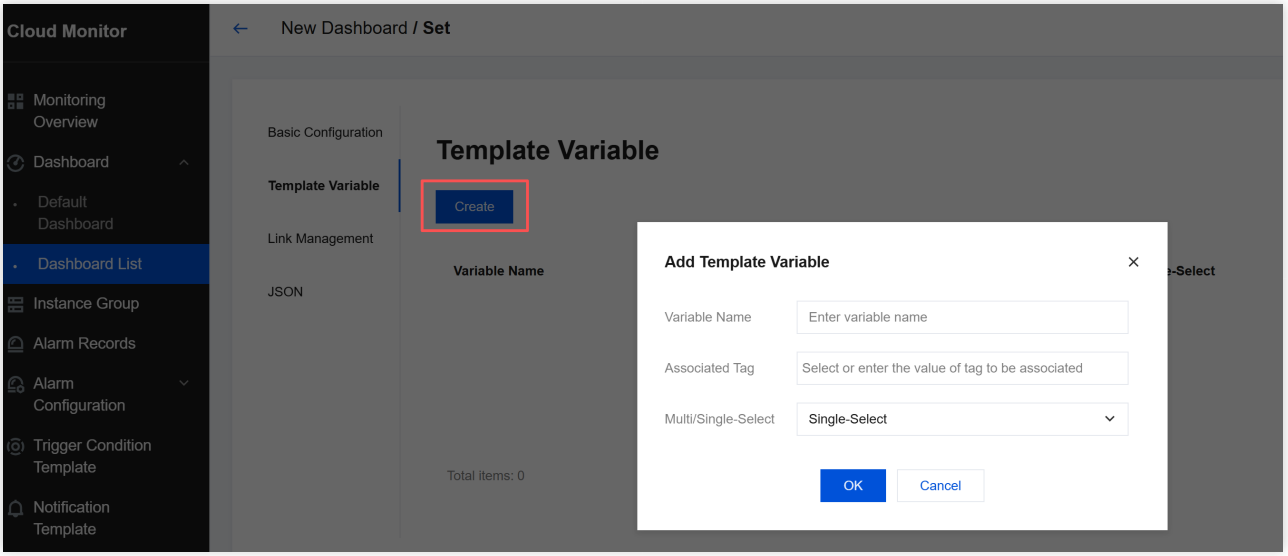
- Current folder: It displays the dashboards in the folder where the current dashboard is located.
- Recently visited: It displays the dashboards you have recently visited.
- Favorites: It displays your favorite dashboards.
- All: It displays all the dashboards you have customized.

# Template Variables

1. Configure template variables.
   i. Log in to the console and choose Cloud Monitor > Dashboard to go to the dashboard list.
   ii. Choose the corresponding dashboard, click the settings icon, and select Template Variables on the Dashboard Management page.



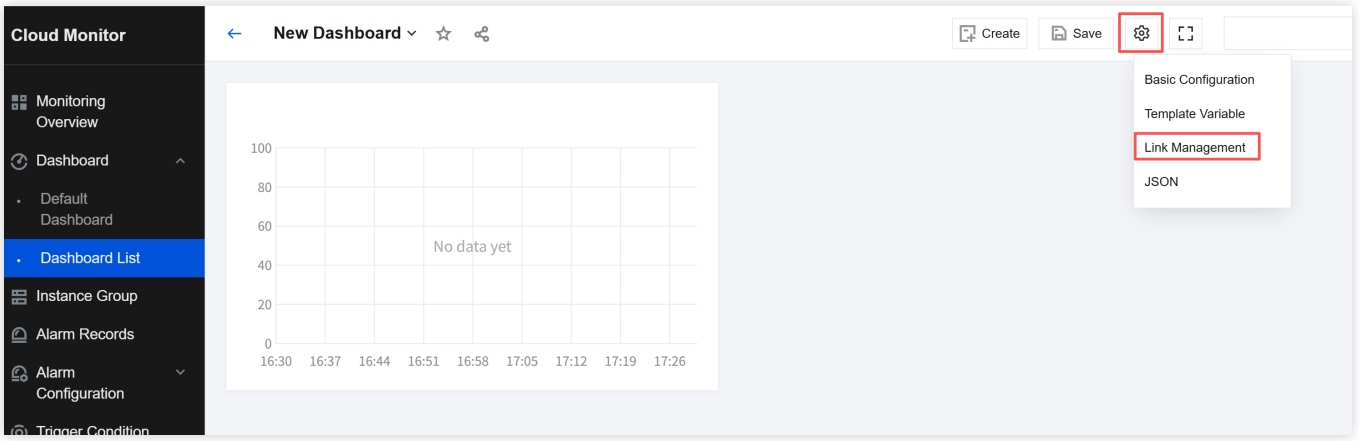

   iii. Template variables are used for customizing dashboard filter conditions. On the Dashboard Management page, you can use template variables to quickly filter instances.
   iv. Click Create on the template management page, and then click OK to create a template variable.



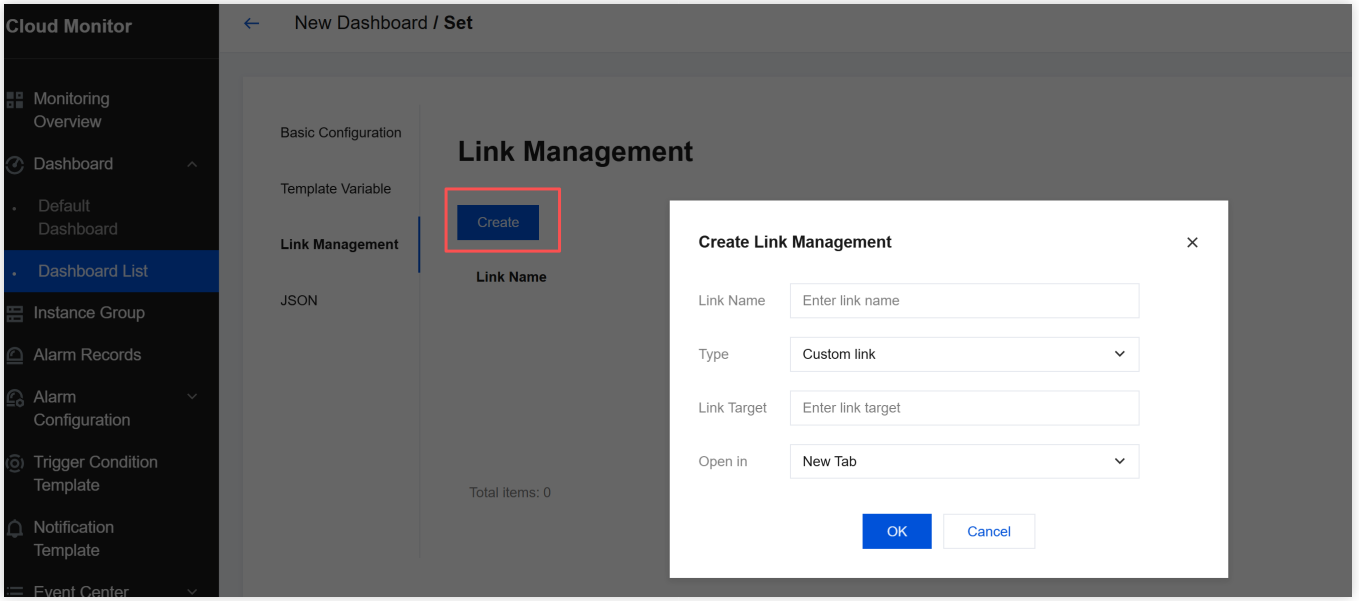

2. Use template variables.
   After template variables are created, they can be used as quick selectors for dashboards and monitoring charts.
   - Use in dashboards: Template variables can be used on the Dashboard Management page to filter the dashboard data.
   - Use in monitoring charts: On the Chart Editing page, quickly select tags to group and aggregate the instance data to be displayed.

# Link Management

1. Log in to the cloud platform console and choose Cloud Monitor > Dashboard to enter the dashboard list.
2. Choose the corresponding dashboard, click the settings icon, and select Link Management on the dashboard management page.




3. Link management is used for customizing quick links to dashboards. To create a link, specify the link name, type, link target, and parameter,




as described below:

- Link Name: custom link name.

  Type
- Custom Link: All links are supported.
- Other Panels: links to other panels.
- Link Target: link address or link panel.

  Parameter
- Time: whether the destination page synchronizes the panel's time during link redirection.

Open Mode

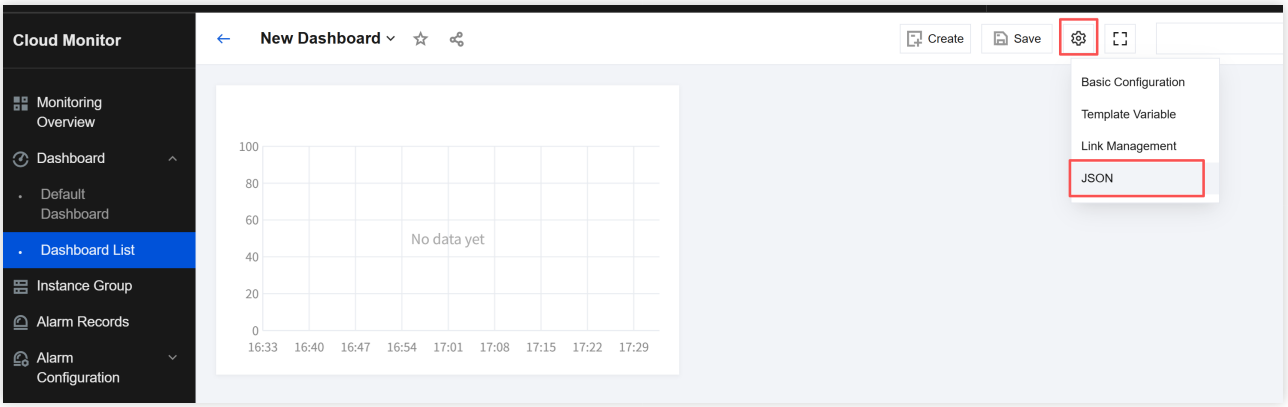
- New Tab: opens the link in a new tab.
- Current Page: opens the link in the current tab.

4. Use links.

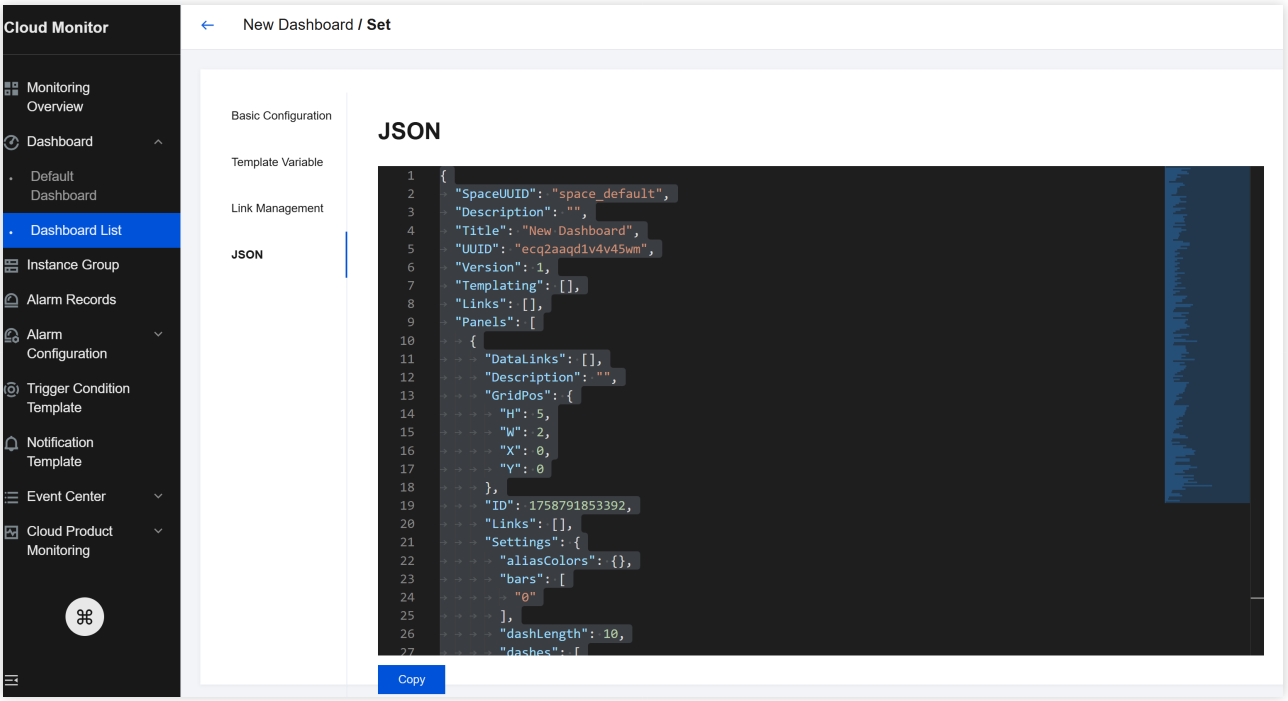
   On the dashboard management page, you can quickly navigate to your custom links for troubleshooting or monitoring data comparison.

# JSON

1. View and copy JSON data.
   i. Log in to the cloud platform console and choose Cloud Monitor > Dashboard to enter the dashboard list.
   ii. Choose the corresponding dashboard, click the settings icon, and select JSON on the dashboard management page.




   iii. You can copy the JSON content to deploy the dashboard to your self-built system.




   The JSON template includes the dashboard's attributes, template variables, and panel query fields. For the specific field description, see the following text.
2. JSON field description.

```
{
    "Description": "",
    "Refresh": "close",
    "Title": "JSONTEST",
```

```
    "UUID": "jdq4joy56is4w60q",
    "Version": 1,
    "Templating": [],
    "Links": [],
    "Panels": [],
    "Time": {
            "From": "now-12h",
            "To": "now"
     },
}
```

| Name | Description |
| --- | --- |
| Description | Current dashboard remarks. |
| Refresh | Auto-refresh time interval. |
| Title | Current dashboard name. |
| UUID | Unique ID of the dashboard. |
| Version | Dashboard version, which increases with each save. |
| Templating | Dashboard template variables. For more information, see Templating. |
| Link | Dashboard link. For more information, see Links. |
| Panels | Chart configuration. See Panels. |
| Time | Time range of the dashboard. |

1. Templating

```
"Templating": [ // Template variables.
      {
         "Label": "CVM instance name", // Template variable alias.
         "Multi": true, // Whether multiple selection is allowed.
         "Name": "cvm", // Tag.
         "Selected": [], // Template variable value.
         "Type": "monitor", // Template variable type.
         "TemplatingType": "basics", // Tag type: basic monitoring or custom monitoring.
    "TemplatingType": '123', // Variable ID.
      }
  ]
```

2. Link

```
"Links": [ // Dashboard link.
      {
         "IncludeVars": true, // Whether to include template variables in the link parameters.
         "KeepTime": true, // Whether to include time variables in the link parameters.
```

```
            "TargetBlank": true, // Whether to open the link in a new tab.
            "Title": "xxx", // Link name.
            "Type": "other", // Link type: custom links or links to other dashboards.
            "Url": "/monitor/dashboard2/dashboards/d/130618223492100096hh64oj49rru3ctk/dashboard2-convergecl
oud_user_dir",  // Link address.
            "ID": 0 // ID
        }
    ]
```

3. Panels

```
"Panels": [ // Panel configuration.
        {
            "Collapsed": false,  // Whether the chart group is collapsed.
            "Datasource": null, // Data source.
            "GridPos": { // Chart position.
                "H": 1,
                "W": 6,
                "X": 0,
                "Y": 0
            },
            "ID": 1595472129140, // Chart ID.
            "Panels": [], // Panel is a field for chart groups, containing sub-panels.
            "Title": "Default Chart Group", // Title.
            "Type": "row" // Chart type.
        },
        {
            "DataLinks": [], // Data links configured for the chart.
            "Description": "", // Remarks for the chart.
            "GridPos": {
                "H": 5,
                "W": 6,
                "X": 0,
                "Y": 1
            },
            "ID": 1595471392817,
            "Links": [], // Chart links.
            "Settings": { // Visualization configuration for the chart.
                "aliasColors": {},
                "bars": false,
                "dashLength": 10,
                "dashes": false,
                "datasource": null,
                "decimals": 2, // Legend precision.
                "fieldConfig": {
                    "defaults": {
                        "custom": {}
                    },
                    "overrides": []
                },
                "fill": "0.8", // Chart fill transparency.
```

```
"fillGradient": 0,
"gridPos": {
  "h": 8,
  "w": 12,
  "x": 0,
  "y": 0
},
"hiddenSeries": false,
"id": "1595471392817", // ID of the chart panel.
"legend": {// Legend settings.
  "alignAsTable": [ // Whether the legend is displayed in table form.
    "1"
  ],
  "avg": [// Whether to display the average value. ['1'] for true; ['0'] for false.
    "1"
  ],
  "current": [// Whether to display the current value. ['1'] for true; ['0'] for false.
    "1"
  ],
  "max": [// Whether to display the maximum value. ['1'] for true; ['0'] for false.
    "1"
  ],
  "min": [// Whether to display the minimum value. ['1'] for true; ['0'] for false.
    "1"
  ],
  "rightSide": [// Whether to place the legend on the right. ['1'] for true; ['0'] for false.
    "1"
  ],
  "show": [// Whether to display the legend. ['1'] for true; ['0'] for false.
    "1"
  ],
  "total": [// Whether to display the cumulative value. ['1'] for true; ['0'] for false.
    "1"
  ],
  "values": false
},
"lines": [// Whether to display the curve. ['1'] for true; ['0'] for false.
  "1"
],
"linesType": true, // Whether to display a smooth curve. ['1'] for true; ['0'] for false.
"linewidth": "2", // Curve width.
"markline": { // Mark settings.
  "marklineMax": {
    "max": [// Whether to display the maximum value. ['1'] for true; ['0'] for false.
      "1"
    ]
  }
},
"nullPointMode": "1", // Display method for null values in the data. 0: connects null data; 1: does not fill i
n the gaps caused by null values; 2: automatically replaces null values with zeros.
"options": {
  "dataLinks": [] // Array of data links.
```

```
},
"percentage": false,
"pointradius": 2,
"points": false,
"renderer": "flot",
"seriesOverrides": [],
"spaceLength": 10,
"stack": [// Whether to display data in a stacked format. ['1'] for true; ['0'] for false.
    "1"
],
"steppedLine": false,
"targets": [
    {
        "refId": "A",
        "scenarioId": "random_walk"
    }
],
"thresholds": [],
"timeFrom": null,
"timeRegions": [],
"timeShift": null,
"title": "New Chart",
"tooltip": {
    "shared": true,
    "sort": 0,
    "value_type": "individual"
},
"type": "graph",
"xaxis": {
    "buckets": null,
    "mode": "time",
    "name": null,
    "show": true,
    "values": []
},
"yaxes": [
    {
        "decimals": 2, // Left y-axis precision.
        "format": "%", // Left y-axis label unit.
        "label": null,
        "logBase": 1,
        "max": 2, // Maximum value of the left y-axis coordinates.
        "min": 0, // Minimum value of the left y-axis coordinates.
        "show": [// Whether to display the left y-axis. ['1'] for true; ['0'] for false.
            "1"
        ]
    },
    {
        "decimals": 2,
        "format": "",
        "label": null,
        "logBase": 1,
```

```
                "max": null,
                "min": null,
                "show": [
                    "1"
                ]
            }
        ],
        "yaxis": {
            "align": false,
            "alignLevel": null
        }
    },
    "Targets": [ // Metric configuration.
        {
            "Aggregate": "", // Statistical method.
            "CompareLastWeek": false, // Week-on-week comparison.
            "CompareYesterday": false, // Day-on-day comparison.
            "Conditions": [ // Filter conditions.
                {
                    "Dimension": [
                        "{\"InstanceId\":\"ins-19827u5b\"}",
                        "{\"InstanceId\":\"ins-xxooxxoo\"}",
                        "{\"InstanceId\":\"ins-19719mfp\"}"
                    ],
                    "Region": "region-name",
                    "Type": "normal"
                }
            ],
            "ConfigId": "cvm",
            "Datasource": "DS_QCEMetric", // Product type.
            "DimensionKey": [
                "InstanceId"
            ],
            "GroupBy": [ // groupby
                "InstanceId"
            ],
            "MetricNames": [ // Metric names.
                "BaseCpuUsage"
            ],
            "Namespace": "QCE/CVM", // Namespace.
            "Period": 60 // Granularity.
        },
    ],
    "Title": "Single Metric - Default Configuration", // Chart name.
    "Type": "graph" // Chart type.
  },
]
```

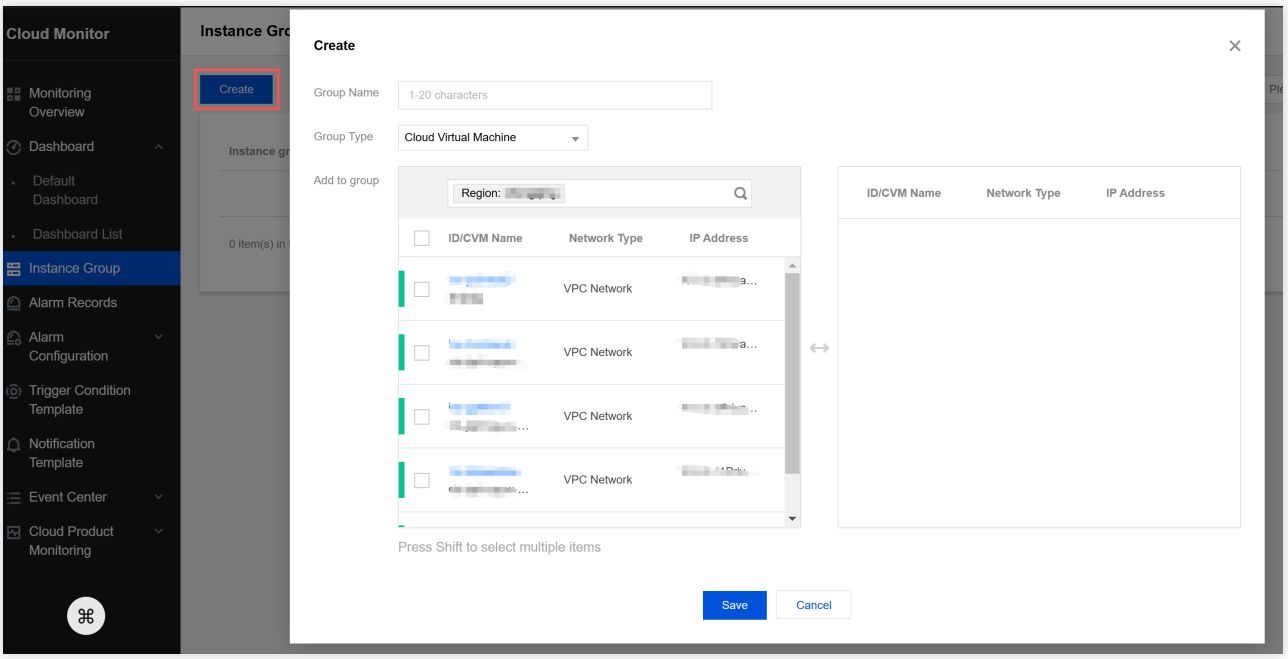
# Grouping Instances

1. Creating an instance group

   i. Log in to the cloud platform console and choose Cloud Monitor > Instance Group.

   ii. Click Create and configure the following options.

   - Group Name: Enter an instance group name.
   - Group Type: Select an option as needed.
   - Add to Group: Select the instances to be added to the group.




   iii. Click Save to create the instance group.

2. Viewing an instance group

   i. Log in to the cloud platform console and choose Cloud Monitor > Instance Group.

   ii. On the instance list page, you can view information such as the instance group name, number of instances, and number of associated alarm policies. You can also click the number of alarm policies to view the instances associated with the group.

3. Editing an instance group

> Note:
>
> - If an instance group is already bound to an alarm policy, instances added to this instance group will automatically be bound to this alarm policy.
> - If an instance group is already bound to an alarm policy, instances removed from this instance group will no longer be bound to this alarm policy.

   i. Log in to the cloud platform console and choose Cloud Monitor > Instance Group.

   ii. Click the name of the instance group that needs to be edited to go to the Instance Management page. On

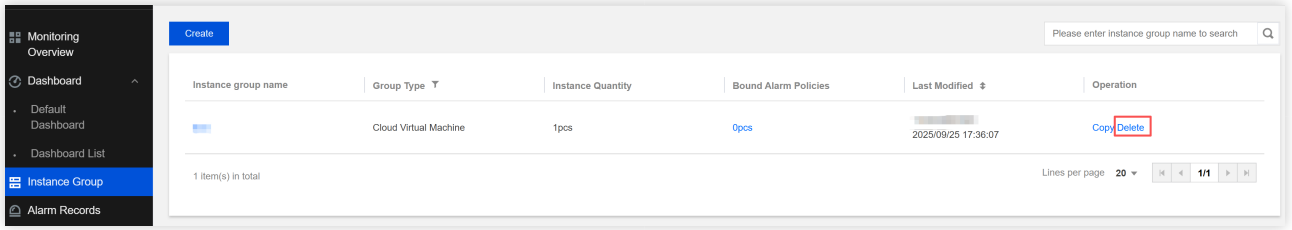
this page, you can edit (add or delete) the instances of this instance group.

- Adding an instance
  - Click Create.
  - In the pop-up configuration box, select the instance that needs to be added to this instance group.
  - Click OK to add the instance to this instance group.
- Deleting an instance
  - Select the instance you want to remove and click Remove.
  - In the pop-up confirmation dialog box, click Confirm to Remove Selected Instance to remove the instance from the instance group.

4. Deleting an instance group

> Note:
>
> If an instance group is already bound to an alarm policy, deleting the instance group will cause the bound alarm policy to become invalid.

i. Log in to the cloud platform console and choose Cloud Monitor > Instance Group.
ii. Find the instance group you want to delete, and click Delete in its right-side operation column.

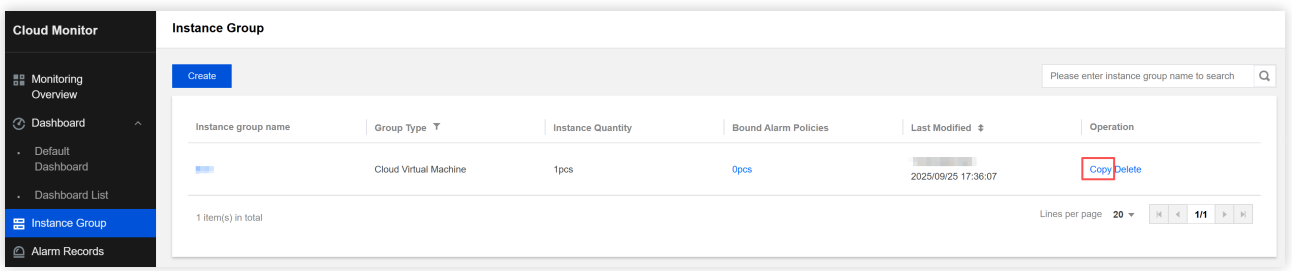



iii. In the pop-up confirmation dialog box, click Confirm Deletion to delete the instance group.

5. Copying an instance group

> Note:
>
> When an instance group is copied, only the instances contained within it will be copied, and the binding relationship between this instance group and the alarm policy will not be copied.

i. Log in to the cloud platform console and choose Cloud Monitor > Instance Group.
ii. Find the instance group you want to delete, and click Copy in its right-side operation column.




iii. In the pop-up confirmation dialog box, click Confirm to Copy to successfully copy the instance group.

# Alarm Service
# Alarm Overview

When the status of certain products changes, you can create alarms to promptly notify you to take action. Alarms monitor specific metrics over a certain period and send alarms at configured intervals based on configured thresholds.
An alarm consists of the following components:

- Alarm trigger condition (under what conditions will the alarm be sent)
- Alarm object (which object will send an alarm)
- Alarm recipient group (who receives the alarm).
- Alarm receiving method (how the alarm is received)
  This document describes how to create alarms for one or more objects, and select the objects to receive alarms.

1. Basic Concepts

| Term | Definition |
| --- | --- |
| Alarm policy type | Policy type identifies policy category and corresponds to specific cloud products, with a distinct layer of granularity under cloud products. For example, when you choose Cloud Virtual Machine > Basic Monitoring policy, you can customize alarm metrics such as CPU utilization and disk utilization; when you choose Cloud Virtual Machine > Storage Monitoring policy, you can customize alarm metrics such as disk I/O wait time and disk read traffic. |
| Alarm rule | Alarm rules correspond to the trigger conditions that specific monitoring metrics should satisfy. For example, when disk utilization exceeds 80% per minute and lasts for 3 minutes, an alarm is triggered once every hour. |
| Relationship between policy types and alarm rules | An alarm type contains multiple alarm rules and is a collection of a series of alarm rules. For example, when you choose Cloud Virtual Machine > Basic Monitoring policy, the policy includes disk utilization alarms, ping unreachable alarms, CPU utilization alarms, and more. |

2. Alarm statuses

| Alarm Status | Description |
| --- | --- |
| Unresolved | Alarms that have not been handled or are being handled, but the resources are still in an abnormal status. |

| Alarm Status | Description |
| --- | --- |
| Resolved | Alarms have been resolved. |
| Insufficient Data | - The alarm policy that triggered an alarm has been deleted.<br>- The CVM has been migrated from one project to another.<br>- No data is reported because the Cloud Monitor control has not been installed or has been uninstalled. |
| Expired | - The alarm policy has changed.<br>- The most recent alarm time has not been updated for more than 24 hours. |

# Alarm Policies
# Creating Alarm Policies

When users need to set alarms for a specific product status, they need to first create an alarm policy. The alarm policy consists of six required components: policy name, monitoring type, policy type, alarm object, trigger condition, and notification template. You can create an alarm policy according to the following instructions:

1. Log in to the Cloud Platform console and choose Cloud Monitor > Alarm Configuration > Alarm Policy.

2. Click Create and configure the alarm according to the following instructions.

| Configuration Type | Configuration Item | Description |
|---|---|---|
| Basic information | Policy name | Enter a custom policy name. |
| | Remarks | Enter custom policy remarks. |
| | Monitoring type | Only cloud service monitoring is supported. |
| | Policy type | Choose the cloud product policy type you need to monitor, for example, Cloud Virtual Machine > Basic Monitoring. |
| Alarm rule configuration | Alarm object | ◦ If you select an instance ID, the alarm policy will be associated with the selected instance.<br>◦ If you select an instance group, the alarm policy will be associated with the selected instance group.<br>◦ If you select all objects, the alarm policy will be associated with all instances under the current account. |
| | Manual configuration (metric alarm) | Alarm trigger condition: It is a semantically meaningful condition composed of metrics, comparison relation, thresholds, statistical granularity, and continuous N monitoring data points. You can set up alarm thresholds based on the trend of metric changes in the chart. For example, when the metric is CPU utilization, the comparison relation is >, the threshold is 80%, the statistical granularity is 5 minutes, and the number of continuous monitoring data points is 2, the condition means to collect CPU utilization data every 5 minutes, and |

| Configuration Type | Configuration Item | Description |
| --- | --- | --- |
| | | if the CPU utilization of a CVM is greater than 80% for two consecutive times, an alarm will be triggered.<br>Alarm frequency: You can set a repeated notification policy for each alarm rule. This way, an alarm notification will be sent repeatedly at a specified frequency when an alarm is triggered.<br>Options: No repetition, 5 minutes, 10 minutes, periodic exponential increasing interval, and other repetition frequencies.<br>An exponentially increasing interval means that alarm information is sent to you when an alarm is triggered the first time, second time, fourth time, eighth time, ... and 2 raised to the power of N times. In other words, the alarm information will be sent less and less frequently as time goes on to reduce the disturbance caused by repeated alarms.<br>Default logic for repeated alarms: An alarm notification will be sent to you at the configured frequency within 24 hours after an alarm is triggered. After 24 hours, the alarm notification will be sent once every day. |
| | Manual configuration (event alarm) | When an exception occurs in cloud product resources or underlying infrastructure services, you can create an event alarm to promptly notify you to take action. For details, see the Event Center document. |
| | Template selection | Click the template selection button, and select the configured template from the dropdown list. For specific configurations, see the Trigger Condition Template document. If the created template is not displayed, you can click Refresh on the right side of the dropdown list to refresh the template selection list. |
| Alarm notification configuration | Alarm notification | Supports choosing a system preset notification template or a user-defined notification template. Each alarm policy can be associated with up to three notification templates. For details, see the Notification Template document. |

3. After configuring the above information, click Complete to create the alarm policy.

# Trigger Condition Templates

When the same alarm trigger conditions are required for different alarm policies, users can configure an alarm trigger condition template, which can be reused with one click when an alarm policy is created, avoiding repeated configurations. For alarm policies using the alarm trigger condition template, users can modify the template trigger conditions, and the changes will automatically apply to the corresponding alarm policies, improving the Ops efficiency.

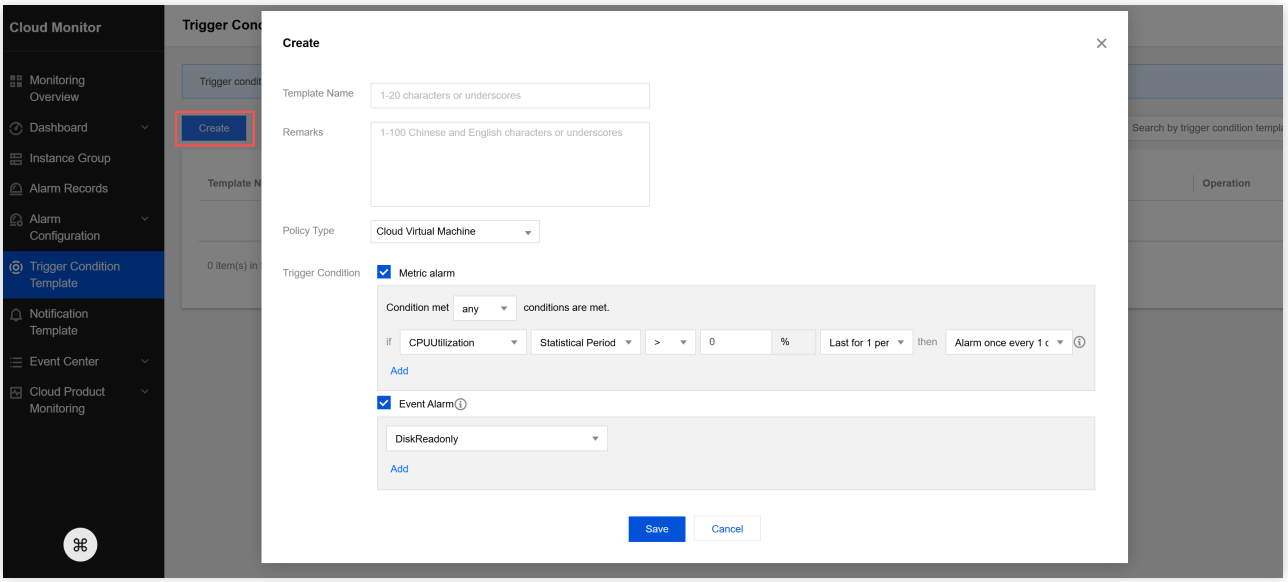
> Note:
>
> One trigger condition template is used to set up trigger condition rules for one cloud product.

1. Create a trigger condition template.

   i. Log in to the cloud platform console and choose Cloud Monitor > Trigger Condition Template.

   ii. Click Create, and configure the trigger conditions in the pop-up window as follows.

   - Template name: Enter a template name.
   - Note: Enter a note for the template.
   - Policy type: Choose the service to be monitored, such as Cloud Virtual Machine > Basic Monitoring.
   - Trigger condition: It is divided into metric alarm and event alarm. Below it, click Add to set up multiple alarm items.




   iii. After completing the configuration, click Save.

2. Edit a trigger condition template.

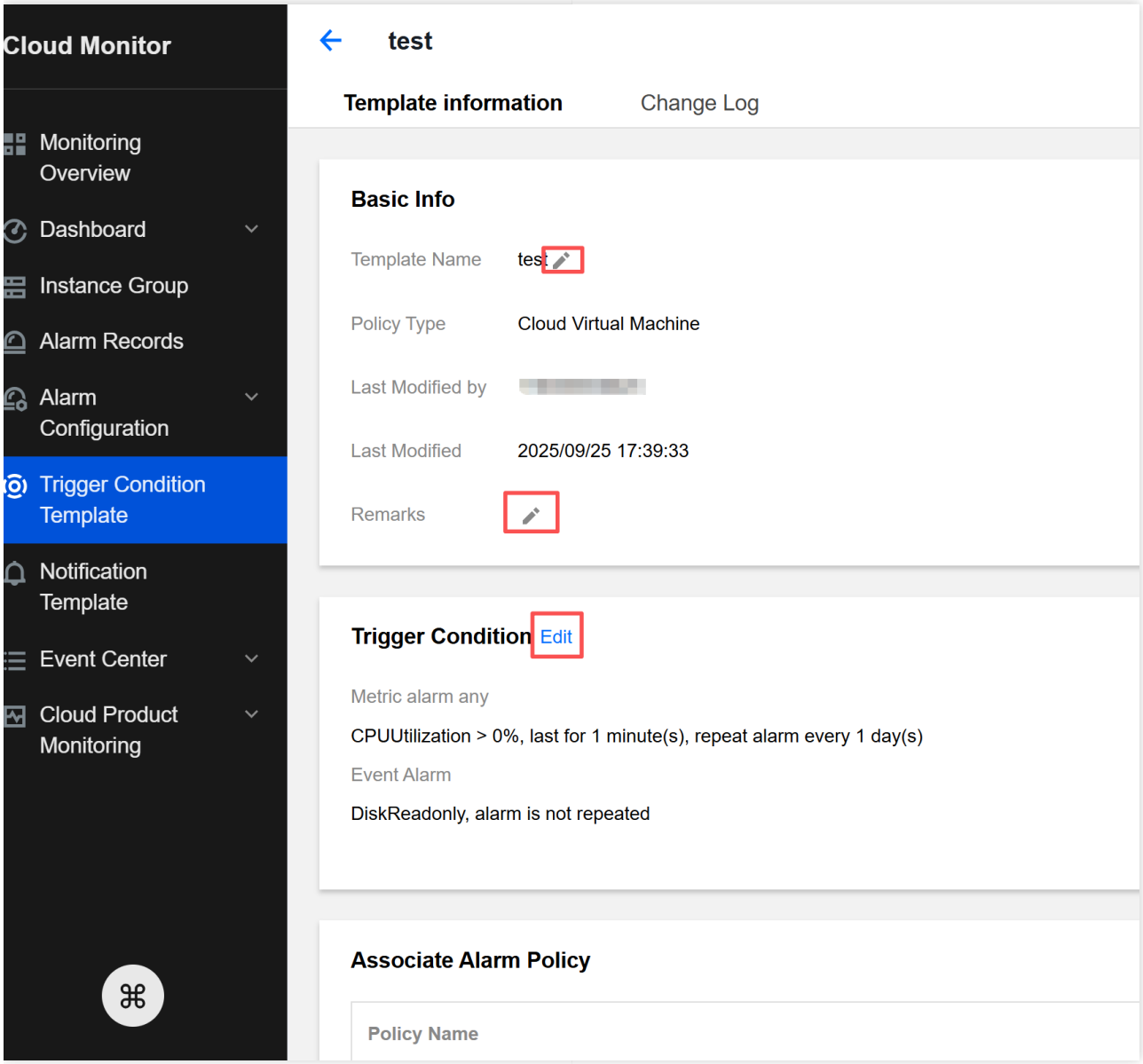
> Note:
>
> When a trigger condition template is already associated with alarm policies, if the trigger condition template is edited, the modification will be applied to all associated alarm policies.

   i. Log in to the cloud platform console and choose Cloud Monitor > Trigger Condition Template.

   ii. In the template list, click the name of the template that needs to be edited to go to the template details

page.

iii. In the Template Information Tab, you can click different buttons to edit and modify different information items.

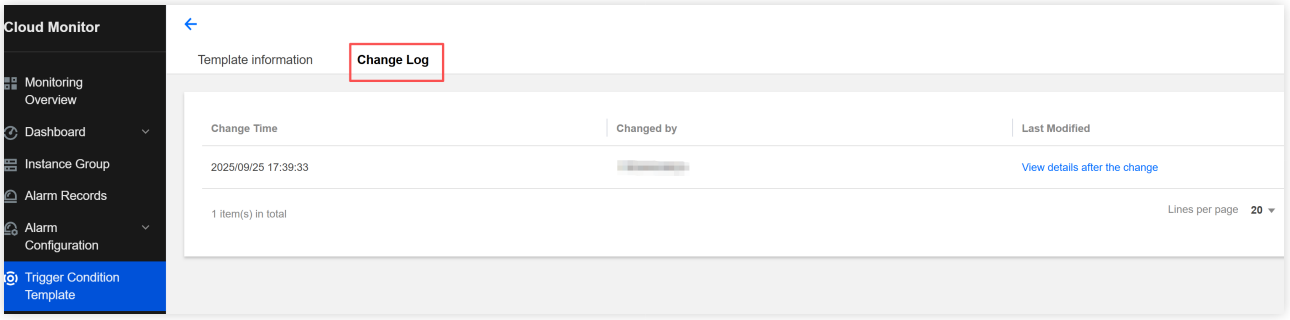



3. View the change records of the trigger condition template.

1. Log in to the cloud platform console and choose Cloud Monitor > Trigger Condition Template.

2. In the template list, click the name of the template for which you need to view the change records to go to the template details page.

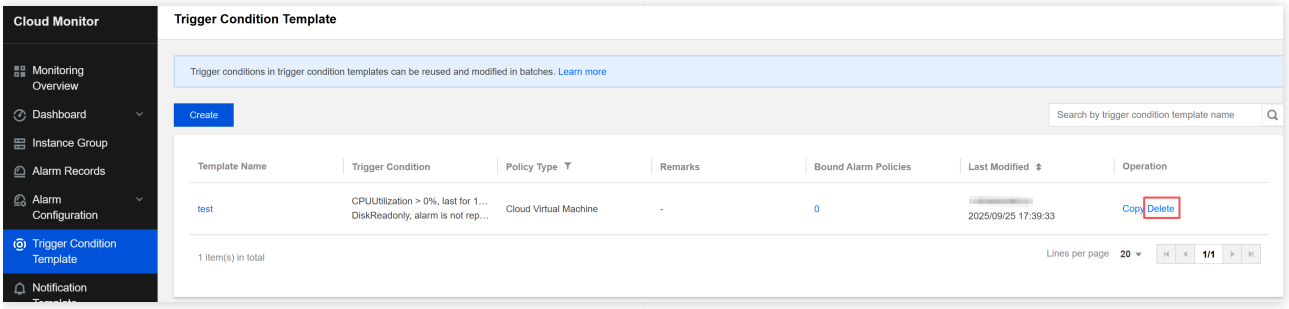
3. In the Change Log tab, you can view the change time, updater, and change result of historical change operations.




4. Delete a trigger condition template.

> Note:
>
> When the trigger condition template is already associated with alarm policies, deleting the trigger condition template will cause the alarm policies associated with the template to become invalid.

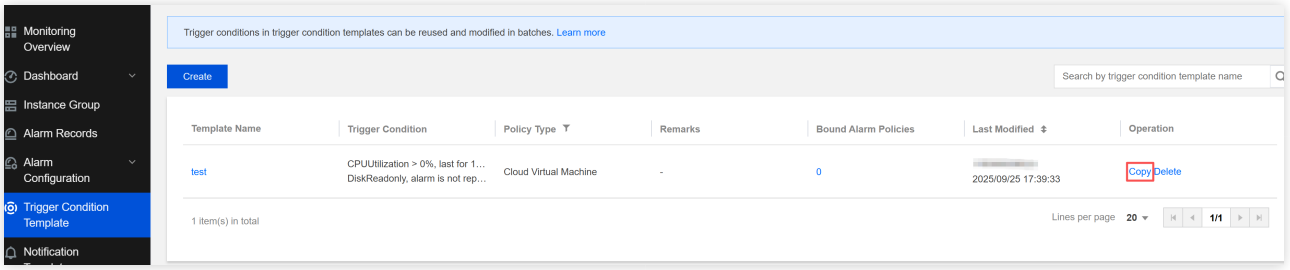
i. Log in to the cloud platform console and choose Cloud Monitor > Trigger Condition Template.
ii. Locate the template you need to modify in the template list, and click Delete in the operation column on the right to delete the template.




5. Copy a trigger condition template.

> Note:
>
> Copying a trigger condition template means that only the trigger condition rules of the template will be copied, while the association between the template and alarm policies will not be copied.

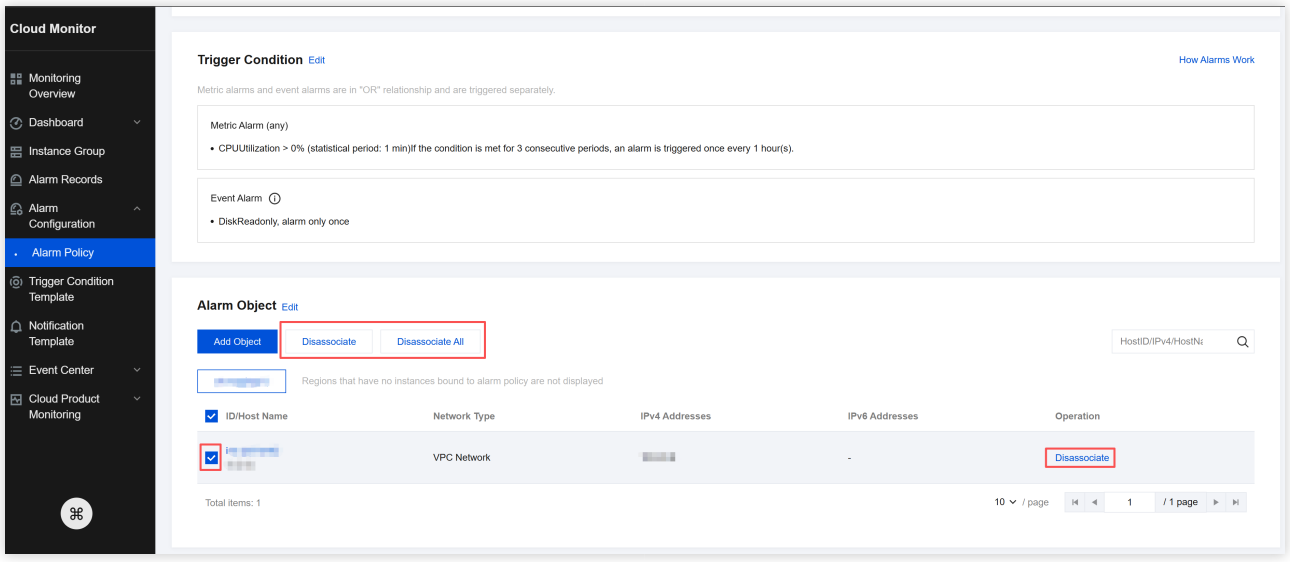
1. Log in to the cloud platform console and choose Cloud Monitor > Trigger Condition Template.
2. On the trigger condition template list page, locate the template you need to copy, and click Copy in the operation column on the right.
3. In the pop-up dialog box, click Confirm Copy to copy the trigger condition template.


Monitoring Overview
Dashboard
Instance Group
Alarm Records
Alarm Configuration
Trigger Condition Template
Notification
Trigger conditions in trigger condition templates can be reused and modified in batches. Learn more
Create
Search by trigger condition template name
Template Name
Trigger Condition
Policy Type
Remarks
Bound Alarm Policies
Last Modified
Operation
test
CPUUtilization > 0%, last for 1…
DiskReadonly, alarm is not rep…
Cloud Virtual Machine
-
0
2025/09/25 17:39:33
Copy Delete
1 item(s) in total
Lines per page 20
1/1

# Associating and Dissociating Objects

After creating an alarm policy, you need to associate some alarm objects with it to indicate which instance objects will trigger an alarm when they meet the alarm trigger conditions.

1. Add an associated object.
    i. Log in to the Cloud Platform console and choose Cloud Monitor > Alarm Configuration > Alarm Policy.
    ii. In the alarm policy list, click on the name of the alarm policy you want to associate objects with to enter the policy details page.
    iii. In the Alarm Object module, click Edit.
    iv. In the pop-up box for alarm object editing, users can select instance ID, instance group, or all object types, and then choose the instance objects or instance groups available for that type accordingly.
    v. Click OK to establish the association between the alarm policy and the alarm object.
2. Remove the association.
    i. Log in to the Cloud Platform console and choose Cloud Monitor > Alarm Configuration > Alarm Policy.
    ii. In the alarm policy list, click on the name of the alarm policy you want to associate objects with to enter the policy details page.
    iii. In the Alarm Object module:
    - If the alarm policy is already associated with an alarm object, click Remove after selecting the instances for which you want to remove the association, and then click Remove in the pop-up confirmation dialog box. If you need to remove the association between the alarm policy and all instances, click Remove All, and then click Remove All in the pop-up confirmation dialog box to remove the association between the selected instances and the alarm policy.

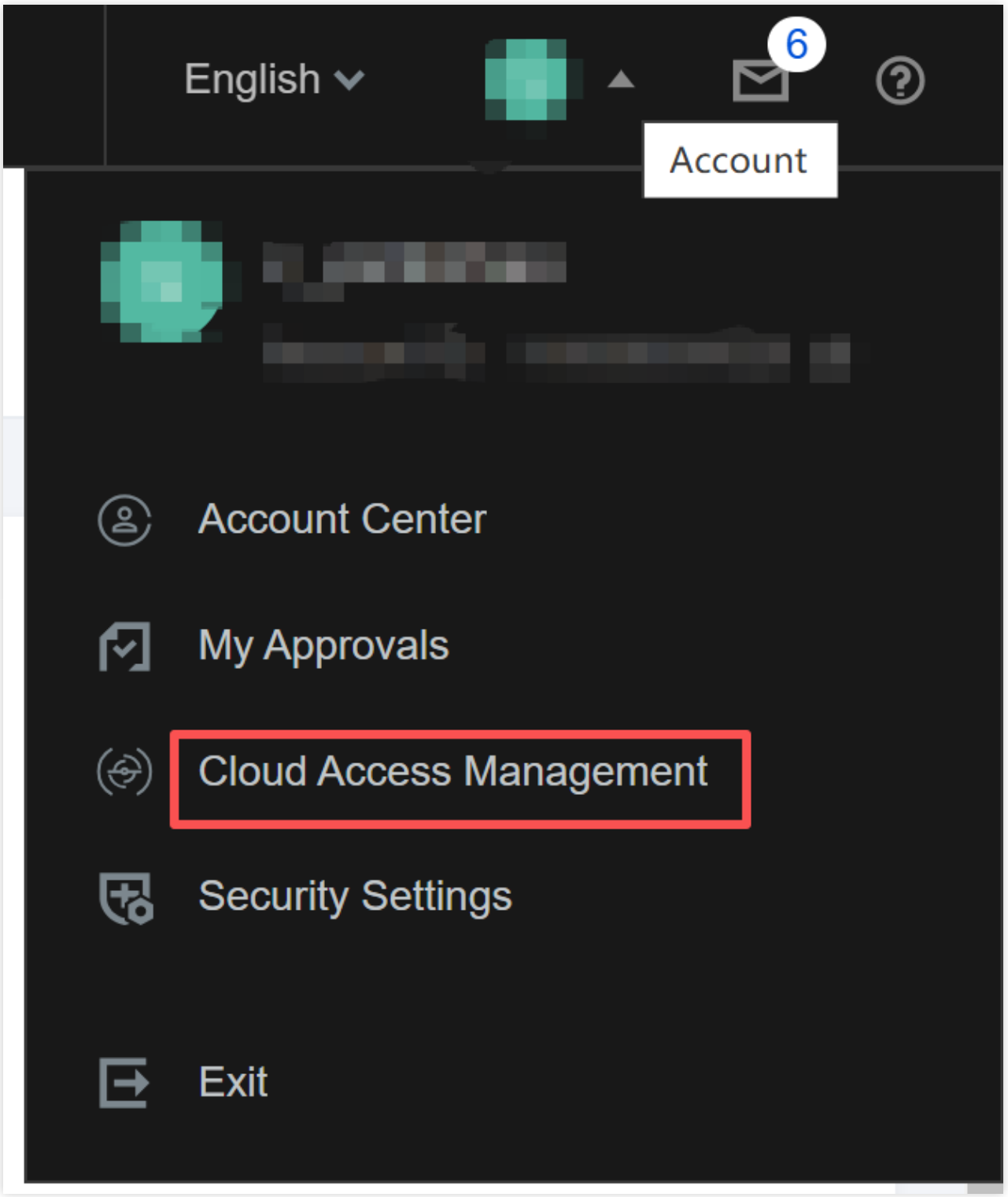



    - If the alarm policy is already associated with an instance group, click Unbind on the right side of the instance group for which you want to remove the association. In the pop-up confirmation dialog box, click Confirm Unbind to remove its association with the alarm policy.

# Alarm Notifications
# Creating and Managing Alarm Recipient Groups

Alarm recipient groups determine who can receive alarm notifications. You can group people concerned with the same alarms into a group, so when an alarm is triggered, everyone in the group receives the corresponding alarm notification.

1. Create an alarm recipient group.
    i. Log in to the cloud platform console, move the mouse pointer over your account avatar, and click Access Management.




ii. Choose User Management > User Group.

iii. Click Create New User Group, fill in the user group name and remarks, and then click Submit to create it successfully.

iv. In the operation column on the right side of the created user group, click Add User. Alternatively, click the user group name to go to the user group details page, choose the Added Users tab, and click Add User.

v. In the pop-up Add User window, select the users you want to add to the user group, and then click

OK.

2. You can also manage and create alarm recipient groups when configuring the alarm notification template.

   i. Log in to the Cloud Platform console and choose Cloud Monitor > Alarm Configuration > Alarm Policy.
   ii. In the alarm policy list, click the name of the alarm policy you want to associate objects with to enter the Details page.
   iii. In the Alarm Notification module, click Create New Template.
   iv. In the pop-up Create Notification Template box, when a user group is selected as the recipient, click Add User Group to redirect to the CAM console to configure user groups.

3. Manage an alarm recipient group.

   i. Log in to the cloud platform console, move the mouse pointer over your account avatar, and click Access Management.
   ii. Choose User Management > User Group.
   iii. If you want to delete a user group, locate it in the user group list, click Delete in the operation column on the right, and then click Confirm to Delete.
   iv. If you want to manage user members in a user group, click the name of the user group you want to manage to go to the user group details page, and choose the Added Users tab.
   v. If you want to add a user, click Add User. If you want to remove a user, select the user you want to remove, and click Remove User or Remove from Group.

# Alarm Callback

By using API callbacks, you can directly receive alarm notifications from the Cloud Platform (TCOP) on your self-built system. API callbacks can push alarm information to URLs that are accessible over the public network through HTTP POST requests. You can take further actions based on the alarm information pushed by API callbacks.

1. Usages
   - Callback API: Users need to provide a URL that is capable of receiving HTTP POST requests and is accessible over the public network as the callback address.
   - Callback trigger: the trigger logic is similar to that of alarm SMS and emails. When an alarm policy created by the user is triggered or the alarm is resolved, the alarm messages will be pushed through the API callbacks. API callbacks also support repeated alarms.
   - Binding API Callbacks: Users can configure API callbacks during the API Callback step of creating an alarm, or add API callbacks on the alarm policy details page. A single alarm policy group can only be bound to one alarm callback URL.
   - Returned content: After sending an alarm notification to the user's bound URL, we need to receive the following returned content to confirm that the user has successfully received the information; otherwise, we will resend the alarm notification, up to three times.

     > Note:
     >
     > sessionId, used to authenticate the callback request.
     >
     > retCode, to determine whether the request was successfully sent.

```
{
  sessionId: "xxxxxxxx",
  retCode: 0
}
```

2. Callback Parameters

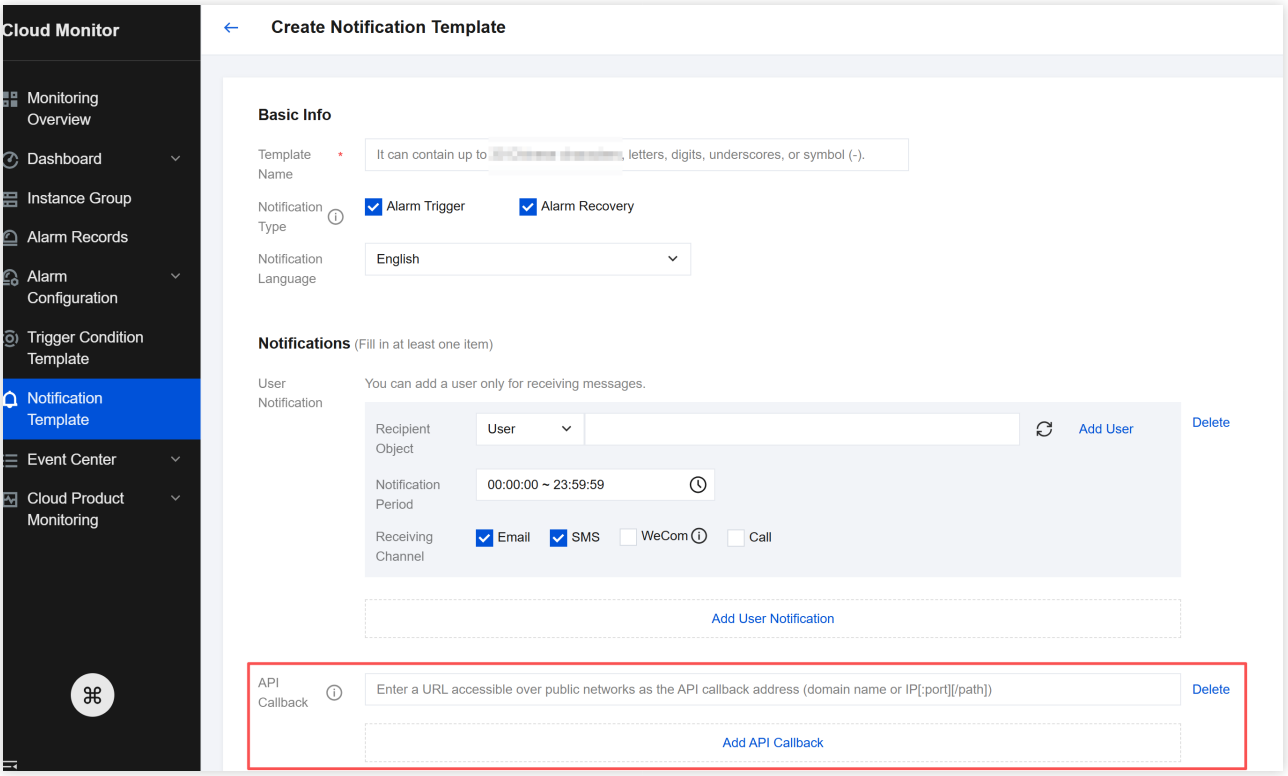
   The callback API will send data in JSON format through HTTP POST requests. The parameters are as follows:

```
{
    "sessionId": "xxxxxxxx",
    "alarmStatus": 1,
    "alarmObjInfo": {
        "region": "CAX01",  // This field will not be returned for products that are not region-specific.
```

```
        "namespace": "qce/cvm",      // Product namespace
        "dimensions": {              // Content in the  dimensions field varies by service.
           "unInstanceId": "ins-o9p3rg3m",
           "objId":"xxxxxxxxxxxx",
        }
    }
    "alarmPolicyInfo": {
           "policyId": "policy-n4exeh88",   // ID of the alarm policy group.
           "policyType": "cvm_device",     // Alarm policy type.
           "policyName": "test",      // Name of the alarm policy group.
           "conditions": {
              "metricName": "cpu usage",         // Metric name.
              "metricShowName": "CPU usage",       // Displayed metric name.
              "calcType": ">>",              // This field will not be returned for metrics without a threshold.
              "calcValue": "90",            // This field will not be returned for metrics without a threshold.
              "currentValue": "100",       // This field will not be returned for metrics without a threshold.
              "unit": "%",                // This field will not be returned for metrics without a threshold.
              "period": "60",              // This field will not be returned for metrics without a threshold.
              "periodNum": "1",            // This field will not be returned for metrics without a threshold.
              "alarmNotifyType": "continuousAlarm",    // Whether repeated alarms are supported, this fi
eld will not be returned for metrics without a threshold.
              "alarmNotifyPeriod": 300                 // Frequency of repeated alarms. This field will not be r
eturned for metrics without a threshold.
           }
           "firstOccurTime": "2017-03-09 07:00:00",     // TTime when the alarm is triggered for the first
time.
           "durationTime": 500,       // Alarm duration in seconds.
           "recoverTime": "0"     // Time when the alarm is resolved in seconds. If it is not resolved, the v
alue is 0.
       }
}
```

# Receiving Alarm Notifications Through a WeCom Group

1. Limitations
   Regarding sending WeCom group messages, the number of messages sent by each bot cannot exceed 20 per minute. If you have many alarm policies, it is recommended to create multiple bots and associate alarm policies with different bots. Otherwise, multiple alarm policies may trigger alarms simultaneously, and you may fail to receive some alarm notifications as a result.
2. Step 1: Add a bot on WeCom.
   i. WebCom for PC
      On WeCom for PC, find the target WeCom group for receiving alarm notifications.
      Select and right-click the WeCom group. In the pop-up box, click Add Group Robot.
      In the pop-up box, click Create a New Robot.
      In the pop-up box, enter a custom bot name and click Add Robot.
      Copy the webhook URL and configure the API callback by following Step 2.
   ii. WebCom for Web
      On WeCom for Web, open the target WeCom group for receiving alarm notifications.
      Click the group settings icon in the upper-right corner.
      On the Group Settings page, choose Group Bot > Add Robot.
      On the Add Bot page, enter a custom name for the new bot.
      Click Add, copy the webhook URL, and configure the API callback by following Step 2.
      WeCom alarm messages are in markdown format.
3. Step 2: Configure the alarm API callback.
   i. Log in to the cloud platform console and choose Cloud Monitor > Notification Template.
   ii. Click Create to enter the Create Notification Template page.
   iii. After configuring the basic information, paste the copied webhook URL in the API callback field.




iv. On the Alarm Policy List page, click the name of the alarm policy to be associated with an alarm callback to enter the Manage Alarm Policy page. Associate a notification template on the Alarm Policy page. After successful configuration, when an alarm policy is triggered or recovered, you can receive alarm notifications sent by the group bot in the WeCom group.

# Notification Template

1. Creating Notification Template
   i. Log in to the Cloud Platform console, and click Cloud Monitor > Notification Template.
   ii. Click Create and enter relevant information in Create Notification Template.

   - Template Name: Enter a custom template name.
   - Notification Type:
   - Alarm Trigger: A notification will be sent when an alarm is triggered.
   - Alarm Recovery: A notification will be sent when an alarm is resolved.
   - User Notification:
   - Recipient: Optional users or user groups. For creating an alert receiving group, please refer to Create and Manage Alert Receiving Groups.
   - Notification Period: Define the time period for receiving alarms.
   - Receiving Channel: Supports three alarm channels: Email, and SMS . You can modify the user's phone, and email information in CAM to decide how the user receives the alert information.
   - API Callback: You can enter up to three URLs accessible over the public network as the callback API addresses, and Cloud Monitor will push alarm messages to them promptly. If the HTTP response returns code 200, the verification is successful. For more information on alarm callback fields, see Alarm Callback Parameters.
   - After you save the callback URL, the system will automatically verify your URL once. The timeout threshold for this verification is 5 seconds. When an alarm policy created by the user is triggered or the alarm is resolved, the alarm messages will be pushed through the API callbacks. An alarm message can be pushed up to three times, and the timeout threshold for each request is 5 seconds.
   - When an alarm policy created by the user is triggered or the alarm is resolved, the alarm messages will be pushed through the API callbacks. API callbacks also support repeated alarms.
   - The outbound IP of the Cloud Monitor callback API is dynamically and randomly allocated, so

no specific IP information can be provided to you, but the IP port is fixed at 80. We recommend you configure a weighted opening policy in the security group based on port 80.

iii. Click Complete to create a notification template.

2. Default Notification Template

The system automatically creates a default notification template for you as detailed below:

| Feature | Default Configuration |
| --- | --- |
| Template Name | Preset notification template |
| Notification Type | Alarm trigger, alarm recovery |
| Alarm Recipient | Root account administrator |
| Notification Period | 00:00:00–23:59:59 (all day) |
| Receipt Channel | Email, SMS |

3. You can also create a new notification template when configuring the alarm policy.
   i. Log in to the Cloud Platform console, click Cloud Monitor > Alarm Configuration > Alarm Policy.
   ii. In the alarm policy list, click on the name of the alarm policy associated with the object to enter its details page.
   iii. In the alarm notification module, click Create Template.
   iv. In the pop-up Create Notification Template dialog, enter the template name, select the recipient and the receiving channel, and then click Confirm to complete creating the notification template.

# Custom Callback Content Templates

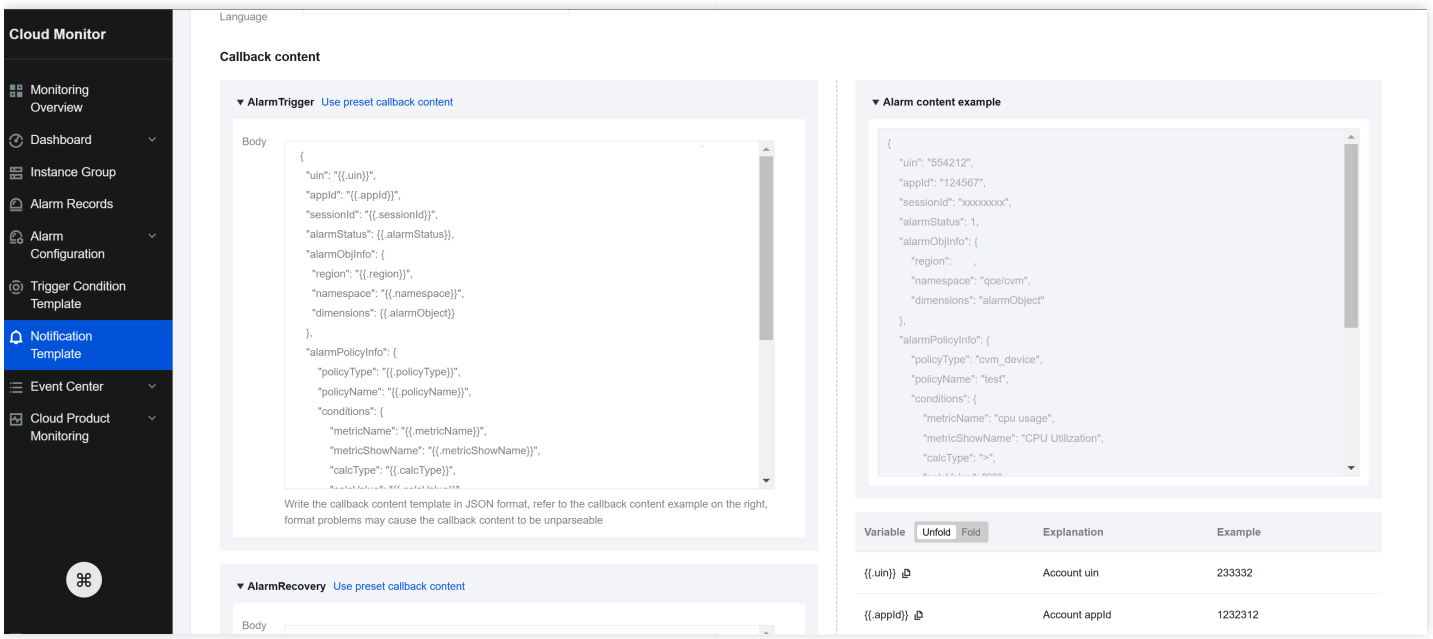
## Scenarios

Through a custom callback content template, you can customize the content and field mapping of the callback message body. This document describes how to create a custom callback content template.

## Operation Steps

1. Log in to the cloud platform console, choose Cloud Monitor > Notification Template, and select the Callback Content Template tab.
2. Click Create Callback Content Template, and configure the content and field mapping of the callback message body on the page.

> Note:
> Write the callback content template in JSON format. For details, see the callback content example on the right side of the page. Format issues may cause the callback content to fail to parse.
>
> 1. Users can click Use the Preset Callback Content to automatically fill in the preset callback content and perform customization based on the preset callback content.
>
> 2. The list on the right of the callback content shows the currently supported variable placeholders. Users can directly click to copy and use them.




3. Click OK to complete the creation of the callback content template.
4. When you choose Create Alarm Policy > Configure Alarm Notifications, if the notification template includes a callback method, you can select a custom callback content template. After the configuration is completed, the callback content will be sent based on the template.

> Note:
> Users can only select a callback content template with the same language as the notification template.

# Variable Description

Currently, only the following variables can be used as placeholders for custom notification content. If an invalid variable is used, the field of the callback content may have no data in it.

| Variable | Description | Example |
| --- | --- | --- |
| {{.uin}} | Account UIN. | 233332 |
| {{.appId}} | Account AppID. | 1232312 |
| {{.sessionId}} | Alarm sessionId. | LxePI1H4oYQCeg9BN1WZ74JG |
| {{.alarmStatus}} | Alarm status. | 0: alarm recovered; 1: alarm triggered; -: no status notification |
| {{.region}} | Region. | Cavite |

| | | |
|---|---|---|
| {{.namespace}} | Namespace. | convergecloud/cvm |
| {{.alarmObject}} | Cloud product dimension information. | {"unInstanceId": "ins-o9p3rg3m", "objId": "xxxxxxxxxxxx"} |
| {{.policyType}} | Alarm policy type. | cvm_device |
| {{.policyName}} | Alarm policy name. | CVM alarm |
| {{.metricName}} | Metric name. | cpu usage |
| {{.metricShowName}} | Metric display name. | CPU utilization |
| {{.calcType}} | Calculation type. | > |
| {{.calcValue}} | Threshold. | 90 |
| {{.currentValue}} | Current value. | 92 |
| {{.unit}} | Unit. | % |
| {{.period}} | Statistical granularity (unit: s). | 5 |
| {{.periodNum}} | Number of lasted data points. | 5 |
| {{.alarmNotifyType}} | Whether repeated alarms are supported. singleAlarm: single alarm; exponentialAlarm: exponential period alarm; continuousAlarm: continuous alarm. For metrics without thresholds, this field is not returned. | singleAlarm |
| {{.alarmNotifyPeriod}} | Alarm notification frequency (unit: s). | 300 |
| {{.firstOccurTime}} | First occurrence time. | 2024-07-09 14:35:00 (UTC+08:00) |
| {{.durationTime}} | Alarm duration (unit: s). | 500 |
| {{.recoverTime}} | Recovery time (0 when not recovered). | 0 |

# Viewing Alarm Records

Cloud Monitor provides you with the alarm records feature, allowing you to trace and view alarm records from the last six months. You can also quickly subscribe to alarm policies on the Alarm Records page.

1. View alarm records.

   i. Log in to the cloud platform console, and choose Cloud Monitor > Alarm Records.

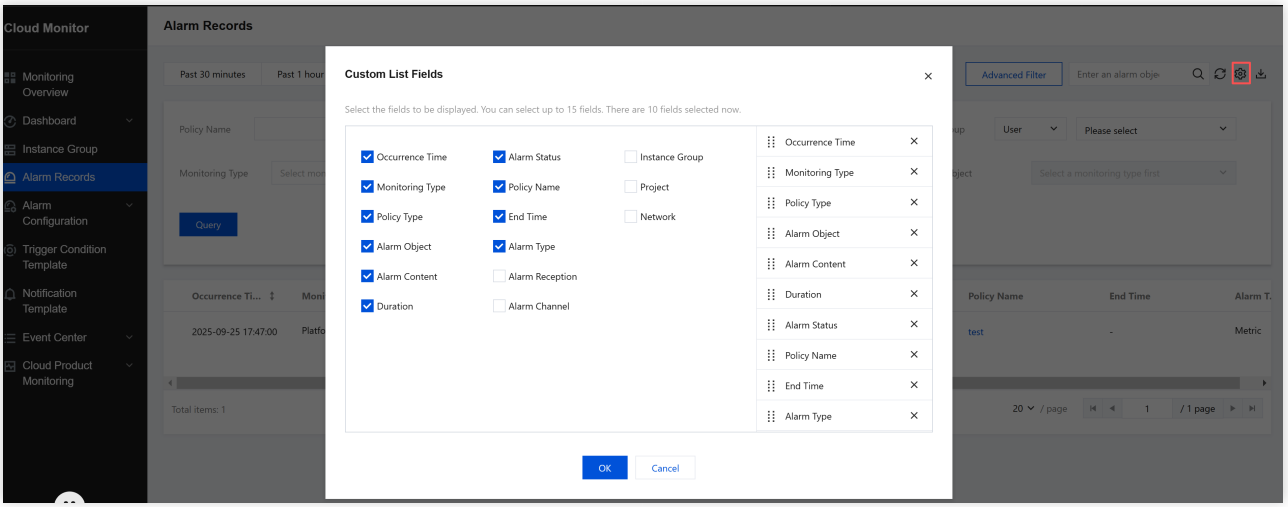
   ii. (Optional) To view alarm records for a certain time period, click the time filter button in the upper-left corner. You can filter alarms generated today, yesterday, and in the last 7 days or 30 days, and you can also select a custom time period. You can view the alarm records in the last six months at most.

   iii. (Optional) You can enter the information of an alarm object (such as instance name, public IP address, and private IP address) in the Alarm Object search box to search for corresponding records.

   iv. (Optional) You can click Advanced Filter to search for matching alarm records based on policy name, alarm content, user information, monitoring type, policy type, and alarm object.

   v. (Optional) You can click the alarm policy name in the list of alarm records to redirect to the detailed alarm policy information.

2. Clear filter conditions.
   After successfully filtering alarm records, click Clear Filter Conditions in the list.
3. Customize list fields.
   i. Log in to the cloud platform console, and choose Cloud Monitor > Alarm Records.
   ii. Click the settings icon in the upper-right corner. You can select the fields to be displayed in the left section of the pop-up box and drag field names in the right section to reorder them, as shown below.



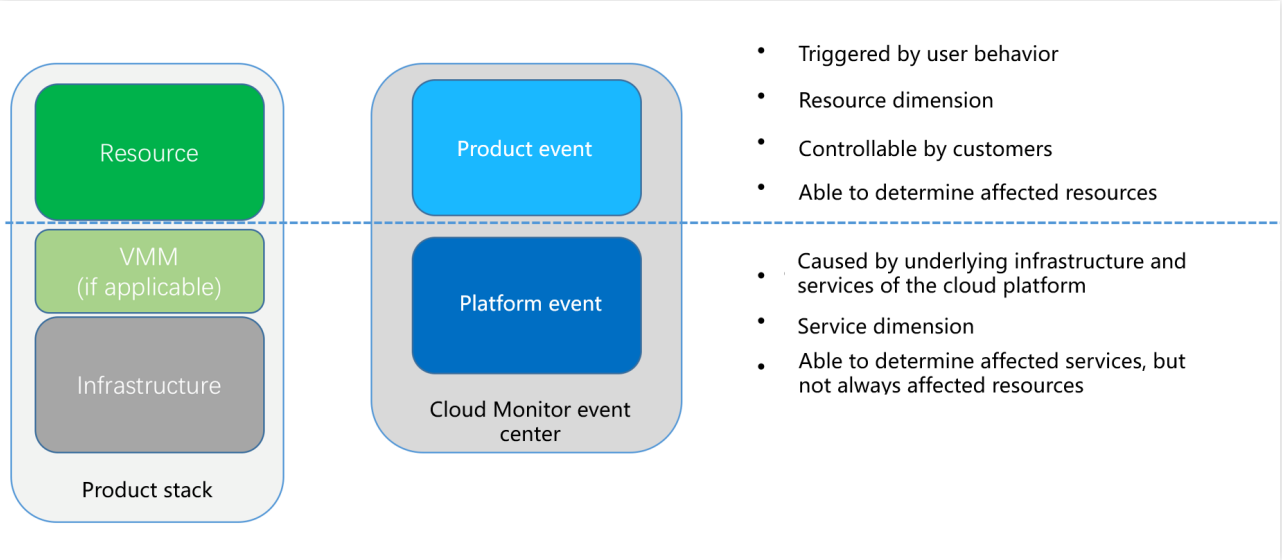

4. Download the list of alarm records.
   i. Log in to the cloud platform console, and choose Cloud Monitor > Alarm Records.
   ii. Click the download icon in the upper-right corner to save the alarm records as a local .csv file.

# Event Center
# Event Center Overview

1. Event overview
   An event is a medium that carries information, describing a non-persistent change of a specific object at a particular moment, associated with a unique moment and a unique object. For example, a computer changing from running to shutdown status, the start and end of a program run, or a power outage in an office building. An event is a record of the change at a moment between two different states of an object.
   The purpose of monitoring services is to assist in the discovery, location, and resolution of issues, ensuring the overall stability and performance of systems and services. Information and data are the core drivers of Ops and monitoring.
   Metrics, as the primary source of monitoring information, describe the numerical values and trends of specific variables of particular objects over time in a threshold-based, periodic, persistent form. They reflect the usage and health status of resources and environments. For example, the CPU utilization of a CVM and the number of slow queries in a database instance.
   Metrics can cover the vast majority of scenarios where monitoring identifies issues. However, using metrics alone as the monitoring information source has the following drawbacks:
   - Information transmission latency: Changes in metric thresholds may only become apparent (or not at all) some time after a specific event has occurred.
   - Low accuracy and distinguishability of transmitted information: It becomes more challenging to determine and locate real events or the causes of issues based only on changes in thresholds.
   - Limited dimensions of information: Metrics record information only in threshold form, lacking the capability to provide additional supplementary and ancillary information through textual means.
     Events are introduced as a carrier of monitoring information to more accurately and directly describe the operational status of resources and underlying infrastructure services, facilitating the discovery, location, and resolution of issues in a more efficient manner.
     The advantages of introducing events include improving information accuracy in a result-oriented manner, reducing latency, conveying more information, and enhancing the dimensionality of monitoring information.
2. Event Center introduction
   As the event information center of the cloud platform, Cloud Monitor's Event Center manages the important event information during the lifecycle and running of all product resources on the cloud platform. It builds a complete event consumption channel and process to support users' monitoring and Ops on the cloud.
   The event information provided by Event Center is obtained from various product modules and underlying infrastructure services on the cloud platform. It is then aggregated, determined, condensed, and finally displayed. The information sources originate from the underlying system logs and monitoring items of each module, ensuring the accuracy and value of the information for users.
3. Event Type
   Based on the information source, cause, characteristics, and form of events, the events covered by Event Center are divided into two main categories: product events and platform events, each supported by two separate product pages and consumption processes.
   - Product events: They originate from the products and resource instances (such as a CVM instance) used by

users on the cloud. They are directly or indirectly triggered by users' operational behaviors during use, and belong to the dimension specific to particular resource instances. Users have control over the events and the ability to handle them themselves. The impact and associated resource instances of the events can be identified.

- Platform events: They are generated in the underlying infrastructure and services that support the cloud platform (such as the virtualization layer VMM, underlying physical servers, networks, and storage modules that support CVM). They are generated or caused by the underlying infrastructure and services that support the cloud platform. Since the root causes of events are not user behaviors and events belong to the service dimension, users can not control these events, and only the cloud platform has the handling capability. Services and product modules affected by and associated with events can be identified, but the affected and associated resource instances may not be identified.



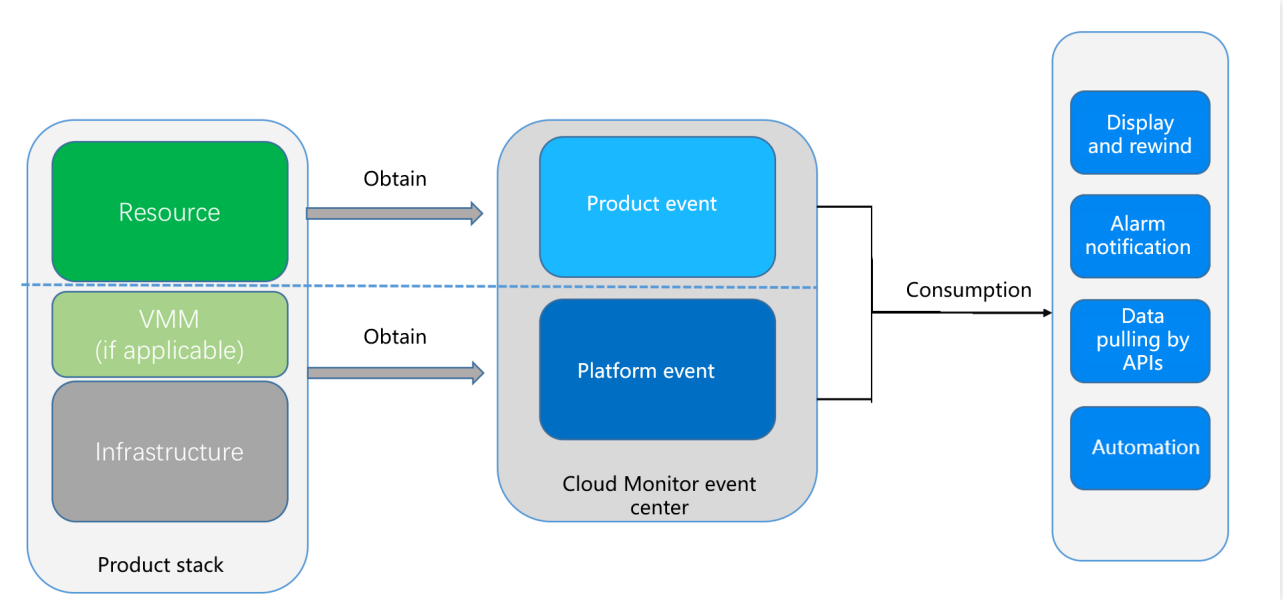

- Event comparison

| Comparison Item | Product Event | Platform Event |
| --- | --- | --- |
| Event source | Products and resource instances used by users on the cloud. | Underlying infrastructure services of the cloud platform. |
| Event cause | Directly or indirectly triggered by users' operational behaviors. | Generated or caused by the underlying cloud infrastructure and services. |
| Event dependency dimension | Resource instances. | Underlying services. |
| Event handling capability | Users have control over the events and can handle them themselves. | Users have no control over the events and need the cloud platform to handle them. |
| Event-related object granularity | The impacted and associated resource instances can be | The impacted and associated underlying services or product modules can be confirmed, while the |

| | | |
|---|---|---|
| | clearly confirmed. | impacted resource instances upstream may not be confirmed. |

4. Event Center use cases

   Based on event information, Event Center plans and provides comprehensive event consumption channels and capabilities.




   - Presentation and tracing: Records all event information related to the account, with the console providing the ability to view, globally filter, search, and trace event information.
   - Alarm notifications: Enables users to receive alarms or subscribe to notifications for monitored events and resources, ensuring timely communication with the owner.
   - Retrieval with APIs: Provides APIs to retrieve event information for integration with self-built monitoring systems.
   - Automation: Subsequent updates will support configuring linked invocation operations with events as triggers, enabling users to achieve automated Ops.

5. Benefits
   - It enhances the monitoring information dimensions of cloud resources, providing more comprehensive data support for monitoring and Ops.
   - It provides event information consumption channels, aiding in the transformation of monitoring information into value.
   - It converges determination logic to more efficiently and directly pinpoint impacted resources and causes.
   - It allows traceable review of event information, ensuring awareness of significant changes in resource and platform lifecycle events.
   - It can trigger automatic responses to specific resource and environment changes.

# Product Event List

The Event Center of Cloud Monitor currently provides the following monitoring information for product events:

1. CVM

| Event Name | Event English Name | Event Type | Event Dimension | Recovery Concept Involved | Event Description | Handling Method and Suggestion |
|---|---|---|---|---|---|---|
| Kernel Failure | GuestCoreError | Exception | CVM instance | No | An operating system kernel bug or driver issue causes a critical error in the operating system kernel. | 1.Check whether any kernel driver modules are loaded into the system other than those provided by the kernel. Try not to load these modules and observe the operating status of the system. 2. Monitor bug reports of the kernel and operating system distributions, and consider upgrading the kernel to resolve the bugs. 3. By default, kdump is enabled for CVM instances. When a panic occurs, system memory dump information will be generated in the /var/crash directory. Use the crash tool to analyze the information. |
| Out of Memory (OOM) | GuestOom | Exception | CVM instance | No | System memory usage is overloaded. | 1.Check whether the memory capacity configured in the current |

| Event Name | Event English Name | Event Type | Event Dimension | Recovery Concept Involved | Event Description | Handling Method and Suggestion |
|---|---|---|---|---|---|---|
| | | | | | | system meets business requirements. If additional capacity is required, it is recommended to upgrade the CVM memory configuration. 2. View processes that are terminated during OOM based on system logs (such as dmesg and /var/log/messages) to check whether the memory used by the processes meets the expectations. Use tools such as valgrind to analyze whether memory leakage occurs. |
| Disk read-only | DiskReadonly | Exception | CVM instance | Yes | Data cannot be written into the disk. | 1.Check whether the disk storage is full. 2. For Linux operating systems, check whether inodes are exhausted (df -i). 3. Check whether the file system is corrupted. |
| Server Restart | GuestReboot | Status change | CVM instance | Yes | The CVM instance is restarted. | This event is triggered when the CVM instance is restarted. Check whether the status |

| Event Name | Event English Name | Event Type | Event Dimension | Recovery Concept Involved | Event Description | Handling Method and Suggestion |
|---|---|---|---|---|---|---|
| | | | | | | change meets the expectations based on the actual situation. |
| Packet Loss Due to Outbound Bandwidth Limit of the Public Network Exceeded | PacketDroppedByQosWanOutBandwidth | Exception | CVM instance | Yes | The outbound public network bandwidth of the CVM instance exceeds the upper limit, causing packet loss. | Increase the upper limit of the public network bandwidth. If the maximum purchase limit has been reached, reduce the bandwidth usage of the instance through means such as load balancing. |
| Packet Loss Due to Connection Limit Exceeded | PacketDroppedByQosConnectionSession | Exception | CVM instance | Yes | An excessive number of connections to the CVM instance leads to packet loss. Taking the 8255C CPU model as an example, it is generally recognized that if the number of connections exceeds 500,000, packet loss may occur. | Contact after-sales service personnel. |

2. VPN Gateway

| Event Name | Event English Name | Event Type | Event Dimension | Recovery Concept Involved | Event Description | Handling Method and Suggestion |
|---|---|---|---|---|---|---|
| Packet Loss | PacketDroppedByQosWanOutBandwidth | Exception | VPN | Yes | The | Increase |

| Event Name | Event English Name | Event Type | Event Dimension | Recovery Concept Involved | Event Description | Handling Method and Suggestion |
|---|---|---|---|---|---|---|
| Due to Outbound Bandwidth Limit of the Public Network Exceeded | | | gateway instance | | outbound public network bandwidth of the VPN gateway instance exceeds the upper limit, causing packet loss. | the upper limit of the public network bandwidth. |
| Packet Loss Due to Connection Limit Exceeded | PacketDroppedByQosConnectionSession | Exception | VPN gateway instance | Yes | The number of connections to the VPN gateway instance exceeds the limit, causing packet loss. | Contact after-sales service personnel. |

# Cloud Product Monitoring
# Cloud Product Monitoring Metrics Overview

This section describes the monitoring metrics for various products in Cloud Monitor.

## Computing

- For CVM, see CVM.
- For Kubernetes Engine (TKE), see Monitoring Metrics for TKE.
- For Bare Metal Server (BMS), see Monitoring Metrics for BMS.

## Network

- For Cloud Load Balancer (CLB), see Monitoring Metrics for CLB.
- For Virtual Private Cloud (VPC), see Monitoring Metrics for VPC.
- For Direct Connect (DC), see Monitoring Metrics for DC.

## Storage

- For Cloud Block Storage (CBS), see Monitoring Metrics for CBS.
- For Cloud Object Storage (COS), see Monitoring Metrics for COS.
- For Cloud Storage Private (CSP), see Monitoring Metrics for CSP.
- For Cloud File Storage (CFS), see Monitoring Metrics for CFS.

## Database

- For CloudDB for Redis® (Redis®), see Monitoring Metrics for Redis®.
- For TDSQL for MySQL, see Monitoring Metrics for TDSQL for MySQL.
- For Cloud MongoDB (MongoDB), see Monitoring Metrics for MongoDB.

## Middleware

- For TDMQ for CMQ (CMQ), see Monitoring Metrics for CMQ.
- For TDMQ for CKafka (Ckafka), see Monitoring Metrics for Ckafka.
- For Service Framework (TSF), see Monitoring Metrics for TSF.
- For TDMQ for Apache Pulsar (TDMQ Pulsar), see Monitoring Metrics for TDMQ Pulsar.

## TI-ONE

- For TI-ONE Platform, see Monitoring Metrics for TI-ONE Platform.

# CVM
# CVM Monitoring Metrics

CVM monitoring metrics need to be collected through the CVM monitoring component. Users can automatically install the monitoring component by selecting Cloud Monitor when creating machines, or manually install it based on the Monitoring Component Installation Guide.
Cloud Monitor provides the following monitoring metrics for CVM instances:

| Metric/Event ID | Metric/Event Name | Unit | Metric Description |
|---|---|---|---|
| CPUUsage | CPU utilization | % | Real-time CPU utilization during machine operation. |
| BaseCpuUsage | Basic CPU utilization | % | Basic CPU utilization is collected and reported by the host, and data can be viewed without the need to install the monitoring component. Data can still be collected and reported even when the instance is under high load. |
| MemUsed | Memory usage | MB | Amount of memory actually used by the user, excluding the memory occupied by buffers and system cache. |
| MemUsage | Memory utilization | % | Amount of memory actually used by the user, excluding the memory occupied by buffers and system cache. |
| CvmDiskUsage | Disk utilization | % | Disk utilization monitoring. |
| DiskIOAwait | Disk I/O wait | ms | Average wait time per I/O operation for disk partitions. |
| DiskReadtraffic | Disk read traffic | KB/s | Average data read volume from disk to memory per second. |
| DiskWritetraffic | Disk write traffic | KB/s | Average data write volume from memory to disk per second. |
| LanIntraffic | Inbound private network bandwidth | MB/s | Average inbound traffic per second on private NIC. |

| Metric/Event ID | Metric/Event Name | Unit | Metric Description |
|---|---|---|---|
| LanOuttraffic | Outbound private network bandwidth | MB/s | Average outbound traffic per second on private NIC. |
| LanInpkg | Number of inbound packets on the private network | Counts/ second | Average inbound packet volume per second for private NIC. |
| LanOutpkg | Number of outbound packets on the private network | Counts/ second | Average outbound packet volume per second for private NIC. |
| WanIntraffic | Inbound public network bandwidth | MB/s | Average inbound traffic per second on public NIC. |
| WanOuttraffic | Outbound public network bandwidth | MB/s | Average outbound traffic per second on public NIC. |
| WanInpkg | Number of inbound packets on the public network | Counts/ second | Average inbound packet volume per second for public network NIC. |
| WanOutpkg | Number of outbound packets on the public network | Counts/ second | Average outbound packet volume per second for public network NIC. |
| TcpCurrEstab | Number of TCP connections | Count | Monitoring of TCP connections. |

# Installing the CVM Monitoring Component

If users need to use Cloud Monitor to view the CVM metric data and trigger alarms, they need to correctly install the monitoring component on the CVM, as CVM metric data collection depends on the monitoring component.
Note: To ensure the normal reporting of monitoring data, the user's CVM needs to allow tcp dport 80.

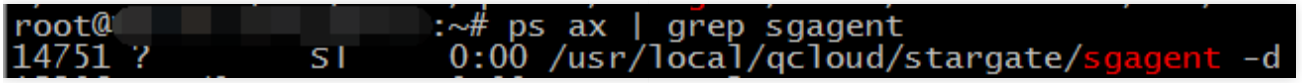
## Linux Installation Guide

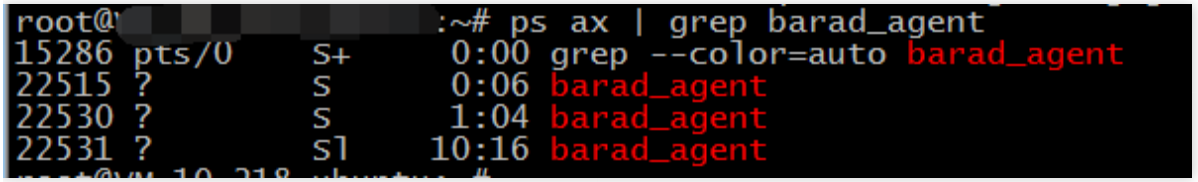
After logging in to the Linux instance, users can execute the following commands for installation:

```
wget http://update2.agent.convergecloud.com/update/linux_stargate_installer
chmod +x linux_stargate_installer
./linux_stargate_installer
```

If the installation is successful, the following content will be displayed:




## Windows Installation Guide

1. After logging in to the Windows instance, users can access http://update2.agent.tencentyun.com/update/windows-stargate-installer.exe via the private network to download the installation package windows-stargate-installer.exe .
2. Run this program to perform automated installation.

# Common Causes and Solutions for Monitoring Components Going Offline

Offline criteria for monitoring components:

> When the monitoring component of a CVM fails to report data for 5 consecutive minutes, the platform considers the monitoring component of this CVM to be offline.

The following are the top causes for monitoring components going offline and the corresponding troubleshooting methods.

## 1. User Operations on the CVM Through the Console or Command Line

When a CVM is shut down through operations, it enters a shutdown status, causing the monitoring component to go offline and no data available.

When users operate through the CVM console or log into the CVM to perform common CVM Ops such as restarting, upgrading the CVM, reinstalling, or creating images, it will result in a timeout in CVM monitoring data reporting, causing the monitoring component to go offline.

Troubleshooting method: You can check whether there are relevant Ops on the CVM based on the specific time point. You can also view operation logs by clicking Operation Log on the CVM Details page.

## 2. High Load on the CVM

High CPU load, full memory, and full bandwidth on CVM can all lead to abnormal data reporting by the monitoring component.

Troubleshooting method: You can log in to the CVM or view the monitoring views to check whether CPU, memory, and bandwidth utilization reach 100% . If so , you can scale out the CVM based on the actual situation.

## 3. Configuration Error of CVM's Private DNS

The configuration error of CVM's private DNS will prevent the monitoring component from reporting data.

Troubleshooting method: For private DNS configuration of the Cloud Platform, see Private DNS Access and Configuration.

# Monitoring Metrics for TKE

Cloud Monitor provides monitoring metrics for TKE in the following dimensions:

## Monitoring Metrics for Clusters

| Metric ID | Metric Name | Unit |
| --- | --- | --- |
| K8sClusterNodeTotal | Number of nodes | Count |
| K8sClusterPodsUsedTotal | Number of Pods | Count |
| K8sClusterAllocatablePodsTotal | Number of allocatable Pods | Count |
| K8sClusterCpuCoreTotal | Total CPU configuration | Cores |
| K8sClusterRateCpuCoreRequest | CPU allocation percentage | % |
| K8sClusterCpuCoreUsed | CPU usage | Cores |
| K8sClusterEksCpuCoreUsed | CPU usage (elastic containers) | Cores |
| K8sClusterRateCpuCoreUsed | CPU utilization | % |
| K8sClusterMemoryTotal | Total memory | GB |
| K8sClusterRateMemRequestBytes | Memory allocation rate | % |
| K8sClusterMemUsageBytes | Memory usage | MB |
| K8sClusterRateMemUsageBytes | Memory utilization | % |
| K8sClusterMemNoCacheBytes | Memory usage (excluding cache) | MB |
| K8sClusterRateMemNoCacheBytes | Memory utilization (excluding cache) | % |
| K8sClusterEksMemUsageBytes | Memory usage (elastic containers) | MB |
| K8sClusterEksMemNoCacheBytes | Memory usage (elastic containers, excluding cache) | MB |
| K8sClusterNetworkReceiveBytes | Inbound network traffic | MB |
| K8sClusterNetworkTransmitBytes | Outbound network traffic | MB |
| K8sClusterNetworkReceiveBytesBw | Inbound network bandwidth | MB/s |

| Metric ID | Metric Name | Unit |
| --- | --- | --- |
| K8sClusterNetworkTransmitBytesBw | Outbound network bandwidth | MB/s |
| K8sClusterNetworkReceivePackets | Number of inbound network packets | Counts/ second |
| K8sClusterNetworkTransmitPackets | Number of outbound network packets | Counts/ second |
| K8sClusterFsReadTimes | Number of reads from the block device | Times |
| K8sClusterFsWriteTimes | Number of writes on the block device | Times |
| K8sClusterFsReadBytes | Size of data read from the block device | MB |
| K8sClusterFsWriteBytes | Size of data written on the block device | MB |
| K8sClusterEtcdDbTotalSizeBytes | etcd storage capacity | MB |
| K8sClusterGpuTotal | Total GPU | Cards |
| K8sClusterRateGpuRequest | GPU allocation rate | % |
| K8sClusterGpuUsed | GPU usage | Cards |
| K8sClusterRateGpuUsed | GPU utilization | % |
| K8sClusterGpuMemoryTotalBytes | Total GPU memory | GB |
| K8sClusterRateGpuMemoryRequest | GPU memory allocation rate | % |
| K8sClusterGpuMemoryUsedBytes | GPU memory usage | MB |
| K8sClusterRateGpuMemoryUsed | GPU memory utilization | % |

# Monitoring Metrics for Nodes

| Metric ID | Metric Name | Unit |
| --- | --- | --- |
| K8sNodeStatus | Node status | - |
| K8sNodePodRestartTotal | Number of Pod restarts | Times |
| K8sNodeCpuCoreRequest | CPU allocation | Cores |
| K8sNodeCpuUsage | CPU utilization | % |

| Metric ID | Metric Name | Unit |
| --- | --- | --- |
| K8sNodeMemRequestBytes | Memory allocation | MB |
| K8sNodeMemUsage | Memory utilization | % |
| K8sNodeLanIntraffic | Inbound private network bandwidth | MB/s |
| K8sNodeLanOuttraffic | Outbound private network bandwidth | MB/s |
| K8sNodeWanIntraffic | Inbound public network bandwidth | MB/s |
| K8sNodeWanOuttraffic | Outbound public network bandwidth | MB/s |
| K8sNodeTcpCurrEstab | Number of TCP connections | Count |
| K8sNodeGpuUsed | GPU usage | Cards |
| K8sNodeRateGpuUsed | GPU utilization | % |
| K8sNodeGpuMemoryUsedBytes | GPU memory usage | MB |
| K8sNodeRateGpuMemoryUsed | GPU memory utilization | % |
| K8sNodeEniipRequestNode | eniip allocation on the node | Count |
| K8sNodeDirecteniRequestNode | direct_eni allocation on the node | Count |

# Monitoring Metrics for Workloads

| Metric ID | Metric Name | Unit |
| --- | --- | --- |
| K8sWorkloadAnomaly | Workload exception | - |
| K8sWorkloadPodsUsedTotal | Number of Pods | Count |
| K8sWorkloadPodRestartTotal | Number of Pod restarts | Times |
| K8sWorkloadCpuCoreUsed | CPU usage | Cores |
| K8sWorkloadRateCpuCoreUsed | CPU utilization | % |
| K8sWorkloadRateEksCpuCoreUsedPod | CPU utilization (elastic containers, as % of Pod specification) | % |
| K8sWorkloadMemUsageBytes | Memory usage | MB |
| K8sWorkloadRateMemUsed | Memory utilization | % |

| Metric ID | Metric Name | Unit |
| --- | --- | --- |
| K8sWorkloadMemNoCacheUsedBytes | Memory usage (excluding cache) | MB |
| K8sWorkloadRateMemNoCacheUsed | Memory utilization (excluding cache) | % |
| K8sWorkloadRateEksMemUsedPod | Memory utilization (elastic containers, as % of Pod specification) | % |
| K8sWorkloadRateEksMemNoCacheUsedPod | Memory utilization (elastic containers, as % of Pod specification, excluding cache) | % |
| K8sWorkloadNetworkReceiveBytesBw | Inbound network bandwidth | MB |
| K8sWorkloadNetworkTransmitBytesBw | Outbound network bandwidth | MB |
| K8sWorkloadNetworkReceiveBytes | Inbound network traffic | MB |
| K8sWorkloadNetworkTransmitBytes | Outbound network traffic | MB |
| K8sWorkloadNetworkReceivePackets | Number of inbound network packets | Counts/ second |
| K8sWorkloadNetworkTransmitPackets | Number of outbound network packets | Counts/ second |
| K8sWorkloadGpuUsed | GPU usage | Cards |
| K8sWorkloadRateGpuUsed | GPU utilization | % |
| K8sWorkloadRateEksGpuUsed | GPU utilization (elastic containers) | % |
| K8sWorkloadGpuMemoryUsedBytes | GPU memory usage | MB |
| K8sWorkloadRateGpuMemoryUsed | GPU memory utilization | % |
| K8sWorkloadEksGpuVideoMemoryUsedBytes | GPU memory usage (elastic containers) | MB |
| K8sWorkloadRateEksGpuVideoMemoryUsed | GPU memory utilization (elastic containers) | % |
| K8sWorkloadFsReadTimes | Number of reads from the block device | Times |
| K8sWorkloadFsWriteTimes | Number of writes on the block device | Times |
| K8sWorkloadFsReadBytes | Size of data read from the block device | MB |
| K8sWorkloadFsWriteBytes | Size of data written on the block device | MB |

# Monitoring Metrics for Pods

| Metric ID | Metric Name | Unit |
|---|---|---|
| K8sPodRestartTotal | Number of Pod restarts | Times |
| K8sPodReadyStatus | PodReady status | - |
| K8sPodCpuCoreUsed | CPU usage | Cores |
| K8sPodRateCpuCoreUsedNode | CPU utilization (% of node) | % |
| K8sPodRateCpuCoreUsedRequest | CPU utilization (% of request) | % |
| K8sPodRateCpuCoreUsedLimit | CPU utilization (% of limit) | % |
| K8sPodEksCpuCoreUsedPod | CPU utilization (elastic containers, as % of Pod specification) | % |
| K8sPodMemUsageBytes | Memory usage | MB |
| K8sPodMemNoCacheBytes | Memory usage (excluding cache) | MB |
| K8sPodRateMemUsageNode | Memory utilization (% of node) | % |
| K8sPodRateMemUsageRequest | Memory utilization (% of request) | % |
| K8sPodRateMemUsageLimit | Memory utilization (% of limit) | % |
| K8sPodRateMemNoCacheNode | Memory utilization (% of node, excluding cache) | % |
| K8sPodRateMemNoCacheRequest | Memory utilization (% of request, excluding cache) | % |
| K8sPodRateMemNoCacheLimit | Memory utilization (% of limit, excluding cache) | % |
| K8sPodRateEksMemUsagePod | Memory utilization (elastic containers, as % of Pod specification) | % |
| K8sPodRateEksMemNoCachePod | Memory utilization (elastic containers, as % of Pod specification, excluding cache) | % |
| K8sPodNetworkReceiveBytesBw | Inbound network bandwidth | MB |
| K8sPodNetworkTransmitBytesBw | Outbound network bandwidth | MB/s |
| K8sPodNetworkReceiveBytes | Inbound network traffic | MB |

| Metric ID | Metric Name | Unit |
| --- | --- | --- |
| K8sPodNetworkTransmitBytes | Outbound network traffic | MB |
| K8sPodNetworkReceivePackets | Number of inbound network packets | Counts/second |
| K8sPodNetworkTransmitPackets | Number of outbound network packets | Counts/second |
| K8sPodGpuRequest | GPU application volume | Cards |
| K8sPodGpuUsed | GPU usage | MB |
| K8sPodRateGpuUsedNode | GPU utilization (% of node) | % |
| K8sPodRateGpuUsedRequest | GPU utilization (% of request) | % |
| K8sPodRateEksGpuUsed | GPU utilization (elastic container) | % |
| K8sPodGpuMemoryRequest | GPU memory application volume | MB |
| K8sPodGpuMemoryUsed | GPU memory usage | MB |
| K8sPodRateGpuMemoryUsedNode | GPU memory utilization (% of node) | % |
| K8sPodRateGpuMemoryUsedRequest | GPU memory utilization (% of request) | % |
| K8sPodEksGpuVideoMemoryUsed | GPU memory usage (elastic containers) | MB |
| K8sPodRateEksGpuVideoMemoryUsed | GPU memory utilization (elastic containers) | % |
| K8sPodFsReadBytes | Block device read bandwidth | MB/s |
| K8sPodFsWriteBytes | Block device write bandwidth | MB/s |
| K8sPodFsReadTimes | Number of reads from the block device | Times |
| K8sPodFsWriteTimes | Number of writes on the block device | Times |
| K8sPodRootfsSize | rootfs space size | MiB |
| K8sPodRootfsUsed | rootfs space usage | GB |

# Monitoring Metrics for Containers

| Metric ID | Metric Name | Unit |
| --- | --- | --- |

| Metric ID | Metric Name | Unit |
| --- | --- | --- |
| K8sContainerCpuCoreUsed | CPU usage | Cores |
| K8sContainerRateCpuCoreUsedNode | CPU utilization (% of node) | % |
| K8sContainerRateCpuCoreUsedRequest | CPU utilization (% of request) | % |
| K8sContainerRateCpuCoreUsedLimit | CPU utilization (% of limit) | % |
| K8sContainerMemUsageBytes | Memory usage | MB |
| K8sContainerRateMemUsageNode | Memory utilization (% of node) | % |
| K8sContainerRateMemUsageRequest | Memory utilization (% of request) | % |
| K8sContainerRateMemUsageLimit | Memory utilization (% of limit) | % |
| K8sContainerMemNoCacheBytes | Memory usage (excluding cache) | MB |
| K8sContainerRateMemNoCacheNode | Memory utilization (% of node, excluding cache) | % |
| K8sContainerRateMemNoCacheRequest | Memory utilization (% of request, excluding cache) | % |
| K8sContainerRateMemNoCacheLimit | Memory utilization (% of limit, excluding cache) | % |
| K8sContainerFsReadBytes | Block device read bandwidth | MB/s |
| K8sContainerFsWriteBytes | Block device write bandwidth | MB/s |
| K8sContainerFsReadTimes | Number of writes on the block device | Times |
| K8sContainerFsWriteTimes | Number of reads from the block device | Times |
| K8sContainerGpuUsed | GPU usage | MB |
| K8sContainerRateGpuUsedNode | GPU utilization (% of node) | % |
| K8sContainerRateGpuUsedRequest | GPU utilization (% of request) | % |
| K8sContainerGpuMemoryUsedBytes | GPU memory usage | MB |
| K8sContainerRateGpuMemoryUsedNode | GPU memory utilization (% of node) | % |
| K8sContainerRateGpuMemoryUsedRequest | GPU memory utilization (% of request) | % |
| K8sContainerNetworkReceiveBytes | Inbound network traffic | MB |

| Metric ID | Metric Name | Unit |
| --- | --- | --- |
| K8sContainerNetworkTransmitBytes | Outbound network traffic | MB |
| K8sContainerNetworkReceiveBytesBw | Inbound network bandwidth | MB/s |
| K8sContainerNetworkTransmitBytesBw | Outbound network bandwidth | MB/s |
| K8sContainerNetworkReceivePackets | Number of inbound network packets | Counts/ second |
| K8sContainerNetworkTransmitPackets | Number of outbound network packets | Counts/ second |
| K8sContainerRootfsUsed | rootfs space usage | GB |

# Monitoring Metrics for Cluster Components

| Metric Name | Unit |
| --- | --- |
| Whether kube-apiserver is normal | - |
| Whether kube-controller-manager is normal | - |
| Whether etcd is normal | - |
| Whether etcd is normal | - |

# Monitoring Metrics for BMS

Cloud Monitor provides monitoring metrics for BMS in the following dimensions:

## Metrics for Basic Monitoring

| Metric ID | Metric/Event Name | Unit | Metric Description |
| --- | --- | --- | --- |
| cpu_usage | CPU utilization | % | CPU utilization |
| cpu_loadavg | 1-minute average load of CPU | - | Average CPU load within 1 minute, calculated using the data of the first column in /proc/loadavg (this metric is not available on Windows operating systems), dependent on the monitoring component installation for collection |
| MemUsed | Memory usage | MB | Memory usage |
| MemUsage | Memory utilization | % | Memory utilization |
| TcpCurrEstab | Number of TCP connections | Count | Number of TCP connections |

## Metrics for Network Monitoring

| Metric ID | Metric/Event Name | Unit | Metric Description |
| --- | --- | --- | --- |
| WanIntraffic | Inbound public network bandwidth | Mb/s | Average inbound traffic per second on public NIC |
| WanOuttraffic | Outbound public network bandwidth | Mb/s | Average outbound traffic per second on public NIC |
| WanInpkg | Number of inbound packets on the public network | Counts/ second | Average inbound packets per second on public NIC |
| WanOutpkg | Number of outbound packets on the public network | Counts/ second | Average outbound packets per second on public NIC |

# Metrics for Disk Monitoring

| Metric ID | Metric/Event Name | Unit | Metric Description |
|---|---|---|---|
| CvmDiskUsage | Disk utilization | % | Disk utilization monitoring. |
| DiskIOAwait | Disk I/O wait | - | Disk I/O wait time |
| DiskReadtraffic | Disk read traffic | Kb/s | Average disk read traffic per second |
| DiskWritetraffic | Disk write traffic | Kb/s | Average disk write traffic per second |
| DiskReadIops | Disk read IOPS | Count | Average disk write traffic per second |
| DiskWriteIops | Disk write IOPS | Count | Average disk read I/O operations per second |

# Metrics for Private Network Monitoring

| Metric/Event Name | Unit |
|---|---|
| bond1 inbound packet loss on the private network | Count |
| bond1 inbound error packets on the private network | Count |
| bond1 inbound packets on the private network | Count/s |
| bond1 inbound bandwidth on the private network | Byte/s |
| bond1 outbound packet loss on the private network | Count |
| bond1 outbound error packets on the private network | Count |
| bond1 outbound packets on the private network | Count/s |
| bond1 outbound bandwidth on the private network | Byte/s |
| eth0 inbound packet loss on the private network | Count |
| eth0 inbound error packets on the private network | Count |
| eth0 inbound packets on the private network | Count/s |
| eth0 inbound bandwidth on the private network | Byte/s |
| eth0 outbound packet loss on the private network | Count |
| eth0 outbound error packets on the private network | Count |

| Metric/Event Name | Unit |
| --- | --- |
| eth0 outbound packets on the private network | Count/s |
| eth0 outbound bandwidth on the private network | Byte/s |
| eth1 inbound packet loss on the private network | Count |
| eth1 inbound error packets on the private network | Count |
| eth1 inbound packets on the private network | Count/s |
| eth1 inbound bandwidth on the private network | Byte/s |
| eth1 outbound packet loss on the private network | Count |
| eth1 outbound error packets on the private network | Count |
| eth1 outbound packets on the private network | Count/s |
| eth1 outbound bandwidth on the private network | Byte/s |

# Monitoring Metrics for CLB

## Monitoring Metrics for Listeners on Private Network

| Metric/Event ID | Metric Name | Unit |
| --- | --- | --- |
| intraffic | Inbound bandwidth | Mbps |
| outtraffic | Outbound bandwidth | Mbps |
| inpkg | Number of inbound packets | Counts/second |
| outpkg | Number of outbound packets | Counts/second |

Cloud Monitor provides monitoring metrics for CLB in the following dimensions:

## Monitoring Metrics for Listeners on Public Network

| Metric/Event ID | Metric Name | Unit |
| --- | --- | --- |
| intraffic | Inbound bandwidth | Mbps |
| outtraffic | Outbound bandwidth | Mbps |
| inpkg | Number of inbound packets | Counts/second |
| outpkg | Number of outbound packets | Counts/second |

## Monitoring Metrics for Server Ports

| Metric/Event ID | Metric Name | Unit |
| --- | --- | --- |
| re_port_status | Abnormal server port status | - |
| health_status | Number of normal RS for health check | Count |

# Monitoring Metrics for VPC

Cloud Monitor provides monitoring metrics for VPC in the following dimensions:

## Monitoring Metrics for the NAT Gateway

| Metric/Event ID | Metric Name | Unit | Metric Description |
|---|---|---|---|
| WanOutBandwidth | Outbound public network bandwidth | MB/s | Average outbound traffic per second on the NAT gateway |
| WanInBandwidth | Inbound public network bandwidth | MB/s | Average inbound traffic per second on the NAT gateway |
| Outpkg | Number of outbound packets on the public network | Counts/ second | Average outbound packets per second on the NAT gateway |
| Inpkg | Number of inbound packets on the public network | Counts/ second | Average inbound packets per second on the NAT gateway |
| WanConnCount | Public network connections | Count | Real-time concurrent connections on the NAT gateway |
| Droppkg | Number of dropped packets | Count | Number of lost data packets |

## Monitoring Metrics for the DC Gateway

| Metric/Event ID | Metric Name | Unit | Metric Description |
|---|---|---|---|
| OutBandwidth | Outbound network bandwidth | MB/s | Average outbound traffic per second on the DC gateway |
| InBandwidth | Inbound network bandwidth | MB/s | Average inbound traffic per second on the DC gateway |
| Outpkg | Number of outbound packets | Counts/ second | Average outbound packets per second on the DC gateway |
| Inpkg | Number of inbound packets | Counts/ second | Average inbound packets per second on the DC gateway |

# Monitoring Metrics for the Network Probe

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| TimeOut | Latency | ms | Network latency |
| RateDroppkg | Packet loss rate | % | Ratio of lost network data packets to total data packets |

# Monitoring Metrics for PC

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| OutBandwidth | Outbound network bandwidth | MB/s | Average outbound traffic per second on peering connection (PC) |
| InBandwidth | Inbound network bandwidth | MB/s | Average inbound traffic per second on PC |
| Outpkg | Number of outbound packets | Counts/ second | Average outbound packets per second on PC |
| Inpkg | Number of inbound packets | Counts/ second | Average inbound packets per second on PC |
| RateDroppkg | Packet loss rate | % | Ratio of packets dropped due to bandwidth limit to total packets on PC |

# Monitoring Metrics for the VPN Gateway

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| OutBandwidth | Outbound network bandwidth | MB/s | Average outbound traffic per second on the VPN gateway |
| InBandwidth | Inbound network bandwidth | MB/s | Average inbound traffic per second on the VPN gateway |
| Outpkg | Number of outbound packets | Counts/ second | Average outbound packets per second on the VPN gateway |

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| Inpkg | Number of inbound packets | Counts/ second | Average inbound packets per second on the VPN gateway |

# Monitoring Metrics for EIP

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| VipOutBandwidth | Outbound bandwidth | MB/s | Outbound bandwidth of EIP |
| VipInBandwidth | Inbound bandwidth | MB/s | Inbound bandwidth of EIP |
| VipOutpkg | Number of outbound packets | Counts/second | Outbound packets of EIP |
| VipInpkg | Number of inbound packets | Counts/second | Inbound packets of EIP |

# Monitoring Metrics for DC

Cloud Monitor provides the following monitoring metrics for DC:

## Connection Monitoring Metrics

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| InBandwidth | Inbound network bandwidth | MB/s | Inbound bandwidth data measured from the access point AR to the VPC, collected every 1 or 5 minutes |
| OutBandwidth | Outbound network bandwidth | MB/s | Outbound bandwidth data measured from the VPC to the access point AR, collected every 1 or 5 minutes |

## DC Tunnel Monitoring Metrics

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| InBandwidth | Inbound network bandwidth | MB/s | Average inbound traffic per second on the physical dedicated line |
| OutBandwidth | Outbound network bandwidth | MB/s | Average outbound traffic per second on the physical dedicated line |

# Monitoring Metrics for CBS

Cloud Monitor provides the following monitoring metrics for CBS:

## Monitoring Metrics for CBS

| Metric/Event ID | Metric Name | Unit | Metric Description |
|---|---|---|---|
| disk_read_iops | Disk read IOPS | Times | Average disk read operations per second |
| disk_write_iops | Disk write IOPS | Times | Average disk write operations per second |
| disk_read_traffic | Disk read traffic | KB/s | Average amount of data read from disk to memory per second |
| disk_write_traffic | Disk write traffic | KB/s | Average amount of data written from memory to disk per second |
| disk_await | Disk I/O wait time | ms | Average wait time per disk I/O operation |
| disk_svctm | Disk I/O service time | ms | Average time taken per disk I/O operation |
| disk_util | Busy time ratio of disk I/O | % | The percentage of time taken by I/O operations on the disk out of the total time |

## Monitoring Metrics for CBS Snapshots

| Metric/Event ID | Metric Name | Unit |
|---|---|---|
| snap_capacity_usage | Capacity used by CBS snapshots | GB |
| snap_count_usage | Number of used CBS snapshots | Count |

# Monitoring Metrics for COS

Cloud Monitor provides the following types of monitoring metrics for COS:

## Monitoring Metrics for Status Codes

| Metric/Event ID | Metric Name | Unit | Metric Description |
|---|---|---|---|
| 2xxResponse | 2xx status code | Count | Number of requests returned with 2xx status code |
| 2xxResponseRate | Percentage of 2xx status code | % | Percentage of requests returned with 2xx status code out of total requests |
| 3xxResponse | 3xx status code | Count | Number of requests returned with 3xx status code |
| 3xxResponseRate | Percentage of 3xx status code | % | Percentage of requests returned with 3xx status code out of total requests |
| 4xxResponse | 4xx status code | Count | Number of requests returned with 4xx status code |
| 4xxResponseRate | Percentage of 4xx status code | % | Percentage of requests returned with 4xx status code out of total requests |
| 400Response | 400 status code | Count | Number of requests returned with 400 status code |
| 400ResponseRate | Percentage of 400 status code | % | Percentage of requests returned with 400 status code out of total requests |
| 403Response | 403 status code | Count | Number of requests returned with 403 status code |
| 403ResponseRate | Percentage of 403 status code | % | Percentage of requests returned with 403 status code out of total requests |
| 404Response | 404 status code | Count | Number of requests returned with 404 status code |
| 404ResponseRate | Percentage of 404 status code | % | Percentage of requests returned with 404 status code out of total requests |
| 5xxResponse | 5xx status code | Count | Number of requests returned with 5xx status code |

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| 5xxResponseRate | Percentage of 5xx status code | % | Percentage of requests returned with 5xx status code out of total requests |
| 500Response | 500 status code | Count | Number of requests returned with 500 status code |
| 500ResponseRate | Percentage of 500 status code | % | Percentage of requests returned with 500 status code out of total requests |
| 501Response | 501 status code | Count | Number of requests returned with 501 status code |
| 501ResponseRate | Percentage of 501 status code | % | Percentage of requests returned with 501 status code out of total requests |
| 502Response | 502 status code | Count | Number of requests returned with 502 status code |
| 502ResponseRate | Percentage of 502 status code | % | Percentage of requests returned with 502 status code out of total requests |
| 503Response | 503 status code | Count | Number of requests returned with 503 status code |
| 503ResponseRate | Percentage of 503 status code | % | Percentage of requests returned with 503 status code out of total requests |
| 504Response | 504 status code | Count | Number of requests returned with 504 status code |

# Monitoring Metrics for Requests

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| StdReadRequests | Read requests for standard storage | Times | Number of read requests for the standard storage type, calculated based on the number of request commands sent |
| StdWriteRequests | Write requests for standard storage | Times | Number of write requests for the standard storage type, calculated based on the number of request commands sent |

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| LFReadRequests | Infrequent storage read requests | Times | Number of read requests for infrequent storage type, calculated based on the number of request commands sent |
| LFWriteRequests | Infrequent storage write requests | Times | Number of write requests for infrequent storage type, calculated based on the number of request commands sent |

# Monitoring Metrics for Traffic

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| UploadFlow | Upload traffic | Byte | Traffic from local to remote |
| DownloadFlow | Download traffic | Byte | Traffic from remote to local |

# Monitoring Metrics for Capacity

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| BucketCapacity | Bucket capacity | MB | Bucket capacity |
| BucketsUsedRate | Percentage of bucket quota used by the tenant | % | Percentage of used buckets out of total buckets |
| CapacityUsedRate | Percentage of capacity quota used by the tenant | % | Percentage of used capacity out of total capacity |
| ObjectsUsedRate | Percentage of object quota used by the tenant | % | Percentage of used objects out of total objects |
| ObjCapacityUsedRate | Percentage of used bucket capacity quota | % | Percentage of used bucket capacity out of total bucket capacity |
| ObjNumsUsedRate | Percentage of used bucket object quota | % | Percentage of used bucket objects out of total bucket objects |

# Monitoring Metrics for CSP

Cloud Monitor provides the following monitoring metrics for CSP:

| Metric/Event ID | Metric Name | Unit | Metric Description |
|---|---|---|---|
| 2xx_response | 2xx status code | Count | Number of requests returned with 2xx status code |
| 2xx_response_pct | Percentage of 2xx status code | % | - |
| 3xx_response | 3xx status code | Count | Number of requests returned with 3xx status code |
| 3xx_response_pct | Percentage of 3xx status code | % | - |
| 400_response | 400 status code | Count | Number of requests returned with 400 status code |
| 400_response_pct | Percentage of 400 status code | % | - |
| 403_response | 403 status code | Count | Number of requests returned with 403 status code |
| 403_response_pct | Percentage of 403 status code | % | - |
| 404_response | 404 status code | Count | Number of requests returned with 404 status code |
| 404_response_pct | Percentage of 404 status code | % | - |
| 4xx_response | 4xx status code | Count | Number of requests returned with 4xx status code |
| 4xx_response_pct | Percentage of 4xx status code | % | - |
| 500_response | 500 status code | Count | Number of requests returned with 500 status code |
| 500_response_pct | Percentage of 500 status code | % | - |

| Metric/Event ID | Metric Name | Unit | Metric Description |
|---|---|---|---|
| 501_response | 501 status code | Count | Number of requests returned with 501 status code |
| 501_response_pct | Percentage of 501 status code | % | - |
| 502_response | 502 status code | Count | Number of requests returned with 502 status code |
| 502_response_pct | Percentage of 502 status code | % | - |
| 503_response | 503 status code | Count | Number of requests returned with 503 status code |
| 503_response_pct | Percentage of 503 status code | % | - |
| 5xx_response | 5xx status code | Count | Number of requests returned with 5xx status code |
| 5xx_response_pct | Percentage of 5xx status code | % | - |
| bucket_storage | Bucket capacity | Bytes | Capacity of the bucket |
| bucket_obj | Number of bucket objects | Count | Number of objects in the bucket |
| inbound_traffic | Upload traffic | Bytes | - |
| outbound_traffic | Download traffic | Bytes | - |
| std_read_requests | Read requests for standard storage | Count | Number of read requests for the standard storage type, calculated based on the number of request commands sent |
| std_write_requests | Write requests for standard storage | Count | Number of write requests for the standard storage type, calculated based on the number of request commands sent |

# Monitoring Metrics for CFS

Cloud Monitor provides the following types of monitoring metrics for CFS:

## Monitoring Metrics for Bandwidth

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| DataReadIoBytes | Read bandwidth | KB/s | Average amount of data read from a file system per second |
| DataWriteIoBytes | Write bandwidth | KB/s | Average amount of data written to a file system per second |

## Monitoring Metrics for Read/Write Counts

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| DataReadIoCount | Read IOPS | Times/second | Read operations per second on the file system |
| DataWriteIoCount | Write IOPS | Times/second | Write operations per second on the file system |

## Monitoring Metrics for Storage

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| Storage | Storage capacity | GB | File system storage capacity at the current time |

## Monitoring Metrics for Latencies

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| DataReadIoLatency | Read latency | ms | Average read latency on the file system |
| DataWriteIoLatency | Write latency | ms | Average write latency on the file system |

# Other Monitoring Metrics

| Metric/Event ID | Metric Name | Unit | Metric Description |
|---|---|---|---|
| ConnClientsNums | Number of connected clients | Count | Number of connected clients |
| HealthCheckSuccessRate | Success rate of client health monitoring | % | Percentage of successful client health check attempts |

# Monitoring Metrics for Redis®

Cloud Monitor provides monitoring metrics for Redis® in the following dimensions:

## Monitoring Metrics for Community Edition

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| InFlow | Inbound traffic | MB | - |
| InFlowUtil | Inbound traffic utilization | % | Ratio of actual inbound traffic usage to maximum traffic |
| OutFlow | Outbound traffic | MB | - |
| OutFlowUtil | Outbound traffic utilization | % | Ratio of actual outbound traffic usage to maximum traffic |
| CPUUsedUtilMax | Maximum CPU utilization of cache shards | % | Peak CPU utilization of shards within the time window |
| MemUsedUtilMax | Maximum memory utilization of shards | % | Peak memory utilization of shards within the time window |
| MemUsed | Memory usage | MB | Memory capacity actually used, including data and cache |
| Connections | Number of connections | Count | Number of TCP connections to the instance |
| ConnectionsUtil | Connection utilization | % | Ratio of actual TCP connections to maximum connections |
| Commands | Total number of requests | Times | Total number of commands executed |
| CmdRead | Number of read requests | Times/ second | Number of read command executions per second |
| CmdWrite | Number of write requests | Times/ second | Number of write command executions per second |
| CmdValue | Number of large-value requests | Times/ second | Number of large-value command executions per second |

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| CmdOther | Number of other requests | Times/ second | Number of executions of commands other than read and write commands per second |
| CmdSLow | Number of slow queries | Count | Number of commands with the execution latency greater than the value of slowlog-log-slower-than |
| KeyNum | Total number of keys | Count | Total number of keys stored in an instance (first-level keys) |
| KeyExpired | Number of expired keys | Count | Number of keys evicted within the time window, corresponding to the evicted_keys metric generated by the info command |
| KeyEvicted | Number of evicted keys | Count | Number of keys evicted within the time window, corresponding to the expired_keys metric generated by the info command |
| CmdMiss | Read request misses | Count | Number of read requests where keys do not exist, corresponding to the keyspace_misses metric generated by the info command |
| CmdErr | Number of execution errors | Count | Number of command execution errors, such as nonexistent commands and parameter errors |

# Monitoring Metrics for Memory Edition (5-Second Granularity)

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| InFlow | Inbound traffic | MB/s | - |
| InFlowUtil | Inbound traffic utilization | % | Ratio of actual inbound traffic usage to maximum traffic |
| InFlowLimit | Throttling-triggered inbound traffic | Count | Number of times when inbound traffic triggers throttling |
| OutFlow | Outbound traffic | MB/s | - |

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| OutFlowUtil | Outbound traffic utilization | % | Ratio of actual outbound traffic usage to maximum traffic |
| OutFlowLimit | Throttling-triggered outbound traffic | Count | Number of times when outbound traffic triggers throttling |
| MemUsed | Memory usage | MB | Memory capacity actually used, including data and cache |
| MemUsedUtil | Memory utilization | % | Ratio of actual memory usage to total memory requested |
| CpuUsedUtil | CPU utilization | % | Percentage of CPU in non-idle status, calculated from /proc/stat data |
| MemUsedUtilMax | Maximum memory utilization of nodes | % | Peak memory utilization within the time window |
| CpuUsedUtilMax | Maximum node CPU utilization | % | Peak CPU utilization within the time window |
| Commands | Total number of requests | Times | Total number of commands executed |
| CmdRead | Number of read requests | Times/ second | Number of read command executions per second |
| CmdWrite | Number of write requests | Times/ second | Number of write command executions per second |
| CmdKey | Number of key requests | Times/ second | Number of key requests executed per second |
| CmdMget | Number of MGET requests | Times/ second | Number of MGET requests executed per second |
| CmdValue | Number of large-value requests | Times/ second | Number of large-value command executions per second |
| CmdOther | Number of other requests | Times/ second | Number of executions of commands other than read and write commands per second |
| CmdHits | Read request hits | Times | Number of read requests where keys exist, corresponding to the keyspace_hits metric generated by the info command |

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| CmdMiss | Read request misses | Count | Number of read requests where keys do not exist, corresponding to the keyspace_misses metric generated by the info command |
| CmdHitsRatio | Read request hit rate | % | Key hits/(Key hits + Key misses), which can reflect the cache miss status |
| CmdSLow | Number of slow queries | Count | Number of commands with the execution latency greater than the value of slowlog-log-slower-than |
| CmdErr | Number of execution errors | Count | Number of command execution errors, such as nonexistent commands and parameter errors |
| Connections | Number of connections | Count | Number of TCP connections to the instance |
| ConnectionsUtil | Connection utilization | % | Ratio of the actual number of TCP connections to the maximum number of connections |
| KeyNum | Total number of keys | Count | Total number of keys stored in an instance (first-level keys) |
| KeyEvicted | Number of evicted keys | Count | Number of keys evicted within the time window, corresponding to the expired_keys metric generated by the info command |
| KeyExpired | Number of expired keys | Count | Number of keys evicted within the time window, corresponding to the evicted_keys metric generated by the info command |
| LatencyRead | Average read latency | ms | Average execution latency of read commands from Proxy to the Redis® server |
| latency_write | Average write latency | ms | Average execution latency of write commands from Proxy to the Redis® server |
| LatencyAvgProxy | Average execution latency | ms | Average execution latency of requests from Proxy to the Redis® server |
| LatencyMax | Maximum execution latency | ms | Maximum execution latency of requests from Proxy to the Redis® server |
| LatencyOther | Average latency of other commands | ms | Average execution latency of commands other than read and write commands from |

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| | | | Proxy to the Redis® server |

# Monitoring Metrics for TDSQL for MySQL

Cloud Monitor provides the following monitoring metrics for TDSQL for MySQL cloud database instances:

## Monitoring Metrics for Nodes

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| DBNosurviveNode | Number of inactive DB nodes | - | Number of inactive DB nodes |
| ProxyNosurviveNode | Number of inactive proxy nodes | - | Number of inactive proxy nodes |

## Monitoring Metrics for Instances

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| ActiveThreadCount | Number of active threads | Count | A monitoring metric for instances, calculated by summing active threads on the primary and standby nodes of all shards |
| BinlogMinDiskAvailable | Minimum remaining Binlog disk space | GB | A monitoring metric for instances, calculated by summing the minimum values of the BinlogDiskAvailableShard metric across all shards |
| BinlogUsedDisk | Binlog disk space | GB | A monitoring metric for instances, calculated by summing the used Binlog disk space on the primary nodes of all shards |
| ConnUsageRate | Maximum DB connection utilization | % | A monitoring metric for instances, calculated as the maximum value of the DB connection utilization on the primary and standby nodes of all shards in the instance |

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| CpuUsageRate | Maximum CPU utilization | % | A monitoring metric for instances, calculated as the maximum value of the CPU utilization on the primary nodes of all shards in the instance |
| DataDiskAvailable | Available data disk space | GB | A monitoring metric for instances, calculated by summing the available data disk space on the primary nodes of all shards |
| DataDiskUsedRate | Maximum data disk space utilization | % | A monitoring metric for instances, calculated as the maximum data disk space utilization on the primary nodes of all shards in the instance |
| DeleteTotal | Number of DELETE requests | Times/ second | A monitoring metric for instances, calculated by summing the number of DELETE requests on the primary nodes of all shards in the instance |
| IOUsageRate | Maximum I/O utilization | % | A monitoring metric for instances, calculated as the maximum I/O utilization on the primary nodes of all shards in the instance |
| InnodbBufferPoolReads | Number of InnoDB disk page reads | Times | A monitoring metric for instances, calculated by summing the read page counts in the InnoDB disk on all primary and standby nodes of all shards in the instance |
| InnodbBufferPoolReadAhead | Number of buffer pool page reads ahead in InnoDB | Times | A monitoring metric for instances, calculated by summing the number of page reads ahead in the InnoDB buffer pool on all primary and standby nodes of all shards in the instance |

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| InnodbBufferPoolReadRequests | Number of page reads in the InnoDB buffer pool | Times | A monitoring metric for instances, calculated by summing the number of page reads in the InnoDB buffer pool on all primary and standby nodes of all shards in the instance |
| InnodbRowsDeleted | Number of deleted rows in InnoDB | Rows | A monitoring metric for instances, calculated by summing the number of deleted rows in InnoDB on the primary nodes of all shards in the instance |
| InnodbRowsInserted | Number of rows inserted in InnoDB | Rows | A monitoring metric for instances, calculated by summing the number of inserted rows in InnoDB on the primary nodes of all shards in the instance |
| InnodbRowsRead | Number of rows read in InnoDB | Rows | A monitoring metric for instances, calculated by summing the number of rows read in InnoDB on all primary and standby nodes of all shards in the instance |
| InnodbRowsUpdated | Number of rows updated in InnoDB | Rows | A monitoring metric for instances, calculated by summing the number of updated rows in InnoDB on the primary nodes of all shards in the instance |
| InsertTotal | Number of INSERT requests | Times/ second | A monitoring metric for instances, calculated by summing the number of INSERT requests on the primary nodes of all shards in the instance |
| LongQueryCount | Number of slow queries | Times | A monitoring metric for instances, calculated by summing the number of slow queries on the primary nodes of |

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| | | | all shards in the instance |
| MemAvailable | Available cache space | GB | A monitoring metric for instances, calculated by summing the available cache space on the primary nodes of all shards in the instance |
| MemHitRate | Cache hit rate | % | A monitoring metric for instances, calculated as the minimum value of the cache hit rate on the primary nodes of all shards in the instance. |
| ReplaceSelectTotal | Number of REPLACE_SELECT requests | Times/ second | A monitoring metric for instances, calculated by summing the number of REPLACE_SELECT requests on the primary nodes of all shards in the instance. |
| ReplaceTotal | Number of REPLACE requests | Times/ second | A monitoring metric for instances, calculated by summing the number of REPLACE requests on the primary nodes of all shards in the instance. |
| RequestTotal | Total number of requests | Times/ second | A monitoring metric for instances, calculated by summing the number of requests on all primary and standby nodes in the instance. |
| SelectTotal | Number of SELECT requests | Times/ second | A monitoring metric for instances, calculated by summing the number of SELECT requests on all primary and standby nodes of all shards in the instance. |
| SlaveDelay | Secondary latency | Seconds | A monitoring metric for instances. The standby delay of an instance is the maximum standby delay of all shards in the instance, with the standby delay |

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| | | | of each shard calculated first. The standby delay of a shard is the minimum standby delay of all the standby nodes of the shard. |
| UpdateTotal | Number of UPDATE requests | Times/ second | A monitoring metric for instances, calculated by summing the number of UPDATE requests on the primary nodes of all shards in the instance. |
| ThreadsConnected | Number of open connections | Times | A monitoring metric for instances, calculated by summing the number of current open connections on all primary and standby nodes of all shards in the instance. |
| ConnMax | Maximum number of connections | Count | A monitoring metric for instances, calculated by summing the maximum connections on all primary and standby nodes of all shards in the instance. |
| ClientConnTotal | Total number of client connections | Count | A monitoring metric for instances, calculated by summing the number of connections on all proxies in the instance. This metric accurately indicates how many of your clients are connected to the database instance. |
| SQLTotal | Total number of SQL queries | Count | A monitoring metric for instances, indicating the number of SQL statements sent to the database instance. |
| ErrorSQLTotal | Number of SQL errors | Count | A monitoring metric for instances, indicating the number of SQL statements executed with errors. |

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| SuccessSQLTotal | Number of successful SQL queries | Count | A monitoring metric for instances, indicating the number of successfully executed SQL statements. |
| TimeRange0 | Number of requests with execution time less than 5 ms | Times/ second | A monitoring metric for instances, indicating the number of requests with execution time less than 5ms. |
| TimeRange1 | Number of requests with execution time between 5 ms and 20 ms | Times/ second | A monitoring metric for instances, indicating the number of requests with execution time between 5 ms and 20 ms |
| TimeRange2 | Number of requests with execution time between 20 ms and 30 ms | Times/ second | A monitoring metric for instances, indicating the number of requests with execution time between 20 ms and 30 ms |
| TimeRange3 | Number of requests with execution time more than 30 ms | Times/ second | A monitoring metric for instances, indicating the number of requests with execution time more than 30 ms |
| MasterSwitchedTotal | Number of primary-secondary switches | Times | A monitoring metric for instances, indicating the number of primary-secondary switchovers that occurred for the instance |

# Monitoring Metrics for MongoDB

Cloud Monitor provides the following dimensions of monitoring metrics for MongoDB:

## Monitoring Metrics for the Replica Sets

| Metric Name | Unit | Metric Description |
| --- | --- | --- |
| Cache dirty data percentage | % | Percentage of dirty data volume to total data volume in the cache |
| Cache utilization | % | Percentage of used cache space to total cache space |
| Disk utilization | % | - |
| Cache hit rate | % | Percentage of cache hits to total access times |
| Oplog retention time | Hour | - |
| Average primary/secondary node latency per unit of time | s | Average latency of primary/secondary nodes per unit of time |

## Monitoring Metrics for Nodes

| Metric Name | Unit | Metric Description |
| --- | --- | --- |
| CPU utilization | % | - |
| Memory utilization | % | - |
| Number of read requests in the waiting queue | Count | - |
| Number of write requests in the waiting queue | Count | - |
| Inbound network traffic | MB/s | Average inbound traffic per second |
| Outbound network traffic | MB/s | Average outbound traffic per second |
| Number of connections | Count | - |
| Node disk usage | MB | Disk space size used by a single node |

| Metric Name | Unit | Metric Description |
| --- | --- | --- |
| WT engine's ActiveRead | Count | Number of active read operations in the WT engine |
| WT engine's ActiveWrite | Count | Number of active write operations in the WT engine |
| Number of data entries deleted after TTL is triggered | Count | - |
| Number of TTL running rounds | Count | Number of rounds of data cleanup operations triggered based on the TTL index |

# Monitoring Metrics Instances

| Metric Name | Unit | Metric Description |
| --- | --- | --- |
| Number of write requests | Times | Number of write requests initiated |
| Number of read requests | Times | Number of read requests initiated |
| Number of update requests | Times | Number of update requests initiated |
| Number of deletion requests | Times | Number of deletion requests initiated |
| Number of count requests | Times | Number of count requests initiated |
| Number of aggregation requests | Times | Number of aggregation requests initiated |
| Number of command requests | Times | Number of command requests initiated |
| Number of successful requests | Times | Number of requests that succeeded and received the expected response |
| Disk utilization | % | - |
| Number of requests with unit time latency (10 ms to 50 ms) | Times | Number of requests with latency between 10 ms and 50 ms |
| Number of requests with unit time latency (50 ms to 100 ms) | Times | Number of requests with latency between 50 ms and 100 ms |
| Number of requests with unit time latency (over 100 ms) | Times | Number of requests with latency over 100 ms |
| Connection utilization | % | - |
| Number of connections | Count | - |

# Monitoring Metrics for Instances (Replica Set Instance for Versions Above MongoDB 4.0)

| Metric Name | Unit | Metric Description |
| --- | --- | --- |
| Number of write requests | Times | - |
| Number of read requests | Times | - |
| Number of update requests | Times | - |
| Number of deletion requests | Times | - |
| Number of successful requests | Times | - |
| Disk utilization | % | - |
| Number of requests with unit time latency (10 ms to 50 ms) | Times | Number of requests with latency between 10 ms and 50 ms |
| Number of requests with unit time latency (50 ms to 100 ms) | Times | Number of requests with latency between 50 ms and 100 ms |
| Number of requests with unit time latency (over 100 ms) | Times | Number of requests with latency over 100 ms |
| Number of connections | Count | - |
| Connection percentage | % | - |

# Monitoring Metrics for CMQ

Cloud Monitor provides the following monitoring metrics for CMQ:

| Metric/Event ID | Metric Name | Unit |
| --- | --- | --- |
| msg_heap_num | Number of heaped messages | Count |
| msg_product_rate | Message production rate | Count/s |
| msg_product_flow | Message production traffic | Byte/s |
| msg_consume_rate | Message consumption rate | Count/s |
| msg_consume_flow | Message consumption traffic | Byte/s |
| invisible_msg_Limit_count | Upper limit of invisible messages | Count |
| batch_delete_message_rate | Batch deletion request rate | Count/s |
| batch_product_message_rate | Batch production rate | Count/s |
| batch_consume_message_rate | Batch consumption request rate | Count/s |
| delete_message_rate | Deletion request rate | Count/s |
| product_message_rate | Production request rate | Count/s |
| consume_message_rate | Consumption request rate | Count/s |

# Monitoring Metrics for Ckafka

Cloud Monitor provides the following monitoring metrics for Ckafka:

## Monitoring Metrics for Topics

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| CtopicProFlow | Topic produced traffic | MB | Topic production traffic (excluding traffic generated by replicas), summed statistically according to the selected time granularity |
| CtopicProFlowMax | Maximum traffic of topic production | MB | Peak traffic of topic production |
| CtopicProCount | Number of messages produced for the topic | Count | The actual number of messages produced for the topic, summed statistically according to the selected time granularity |
| CtopicConFlow | Topic consumption traffic | MB | Topic consumption traffic (excluding traffic generated by replicas), summed according to the selected time granularity |
| CtopicConFlowMax | Maximum traffic of topic consumption | MB | Peak traffic of topic consumption |
| CtopicConCount | Number of messages consumed for the topic | Count | The number of messages consumed for the topic, summed statistically according to the selected time granularity |
| CtopicMsg | Total disk usage of messages for the topic | MB | Total disk usage of messages for the topic |
| CtopicMsgCount | Total number of messages written to the disk for the topic | Count | Total number of messages written to the disk for the topic (excluding replicas), calculated as the latest value according to the selected time granularity |

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| CtopicMsgRequestProCount | Number of requests produced for the topic | Count | The actual number of requests produced for the topic |
| CtopicMsgRequestConCount | Number of requests consumed for the topic | Count | The actual number of requests consumed for the topic |

# Monitoring Metrics for Instances

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| InstanceProCount | Number of messages produced for the instance | Count | The actual number of messages produced for the instance, summed statistically according to the selected time granularity |
| InstanceConCount | Number of messages consumed for the instance | Count | The number of messages consumed for the instance, summed statistically according to the selected time granularity |
| InstanceProTrafficMax | Peak bandwidth of messages produced for the instance | MB/s | - |
| RequestProCount | Number of requests produced for the instance | Count | The actual number of requests produced for the instance |
| RequestConCount | Number of requests consumed for the instance | Count | The actual number of requests consumed for the instance |
| InstanceMsgCount | Total number of messages written to disk for the instance | Count | The number of messages written to the disk for the instance (excluding replicas), calculated as the latest value according to the selected time granularity |
| InstanceProFlow | Instance production traffic | MB/ min | Instance production traffic (excluding traffic generated by replicas), summed statistically according to the selected time |

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| | | | granularity |
| InstanceMaxProFlow | Maximum production traffic | MB/s | The peak traffic of messages produced for the instance (excluding bandwidth generated by replica production) |
| InstanceProTrafficRate | Instance production bandwidth percentage | % | - |
| InstanceProLimitNums | Instance production throttling frequency | Count | - |
| InstanceConFlow | Instance consumption traffic | MB/ min | Instance consumption traffic (excluding traffic generated by replicas), summed statistically according to the selected time granularity |
| InstanceMaxConFlow | Maximum consumption traffic | MB/s | The peak traffic of messages consumed for the instance (no concept of replicas during consumption) |
| InstanceConTrafficRate | Instance consumption bandwidth percentage | % | - |
| InstanceConLimitNums | Instance consumption throttling frequency | Count | The throttling frequency of messages consumed for the instance |
| InstanceConGroupNums | Number of consumer groups for the instance | Count | Number of consumer groups |
| InstanceConGroupNumsRate | Consumer group percentage | % | - |
| InstanceDiskUsage | Disk usage percentage | % | The percentage of current disk usage to the total disk capacity of the instance specification |
| InstanceMsgHeap | Disk usage of the instance | MB | Disk usage of the instance (including replicas), calculated as |

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| | | | the latest value according to the selected time granularity |
| InstanceConnectionCount | Number of instance connections | Count | The current total number of connections of the instance |
| InstanceConnectionRate | Instance connection percentage | % | The ratio of the current total number of connections to the total number of instances |
| PartitionNums | Number of instance partitions | Count | - |
| PartitionNums | Instance partition percentage | % | - |
| TopicNums | Number of instance topics | Count | - |
| TopicNums | Instance topic percentage | % | - |

# Monitoring Metrics for Consumer Groups

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| CgroupTopicMaxOffset | Maximum offset of the consumer group under the current topic | Count | Maximum offset of the consumer group under the current topic |
| CgroupTopicMsgOffset | Current consumption offset of the consumer group | Count | Current offset of the consumer group |
| CgroupTopicUnconsumeMsgCount | Number of unconsumed messages under the consumer group | Count | Number of unconsumed messages under the consumer group |
| CgroupTopicMsgRate | Topic consumption rate | Count/min | Message consumption speed under the consumer group |
| CgroupTopicUnconsumeMsgSize | Size of unconsumed messages under the | MB | Size of unconsumed messages under the |

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| | consumer group | | consumer group |
| CgroupPartMaxOffset | Maximum offset of the consumer group | Count | Maximum offset under the consumer group |
| CgroupPartOffset | Partition offset | Count | Offset under the partition |
| CgroupPartUnconsumeMsgCount | Number of unconsumed messages in the partition | Count | Number of unconsumed messages under the partition |
| CgroupPartMsgRate | Partition consumption speed | Count/ min | Message consumption speed under the partition |
| CgroupPartUnconsumeMsgSize | Backlog of unconsumed messages in the partition | MB | Backlog of unconsumed messages under the partition |

# Monitoring Metrics for Partitions

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| CpartitionProFlow | Partition production traffic | MB | Partition production traffic (excluding traffic generated by replicas), summed statistically according to the selected time granularity |
| CpartitionConFlow | Partition consumption traffic | MB | Partition consumption traffic (excluding traffic generated by replicas), summed statistically according to the selected time granularity |

# Monitoring Metrics for TSF

Cloud Monitor provides the following monitoring metrics for TSF:

## Monitoring Metrics for Services

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| RequestAverageTime | Average time consumed for received requests | ms | The average processing time consumed by the requested service |
| RequestErrorPercentage | Failure rate of received requests | % | Percentage of abnormal results from the requested service |
| RequestCount | Volume of received requests | Times | Number of times the service was requested |
| Response4xx | HTTP 4xx response code | Count | Number of requests with 4xx response code |
| Response5xx | HTTP 5xx response code | Count | Number of requests with 5xx response code |

## Monitoring Metrics for Deployment Groups

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| HealthyInstancePercentage | Health rate of deployment group nodes | % | Percentage of healthy nodes in a TSF component |

## Monitoring Metrics for APIs

| Metric/Event ID | Metric Name | Unit |
| --- | --- | --- |
| ReqCount | Number of requests | Count |
| TimeSpentAvg | Average duration | ms |
| FailureRate | Failure rate | % |

# Monitoring Metrics for Logs

| Metric/Event ID | Metric Name | Unit |
| --- | --- | --- |
| KeyNums | Keyword occurrence count | Count |

# Monitoring Metrics for TCT

| Metric/Event ID | Metric Name | Unit |
| --- | --- | --- |
| FailureTimes | Number of execution failures | Count |

# Monitoring Metrics for Instances

| Metric/Event ID | Metric Name | Unit |
| --- | --- | --- |
| full_gcEventNums | FULL_GC event statistics metric | Count |

# Monitoring Metrics for TDMQ Pulsar

Cloud Monitor provides the following dimensions of monitoring metrics for TDMQ Pulsar:

## Monitoring Metrics for the Subscription

| Metric Name | Unit | Metric Description |
| --- | --- | --- |
| Number of heaped messages | count | Number of heaped messages |
| Percentage of quota used by unacknowledged messages | % | Ratio of unacknowledged messages to the total quota |
| Message consumption rate | count/s | - |
| Amount of consumed traffic | Byte/s | - |
| Number of unacknowledged messages | Count | Number of unacknowledged messages |
| Number of delayed messages | Count | Number of delayed messages |
| Number of consumers | Count | - |

## Monitoring Metrics for Topics

| Metric Name | Unit | Metric Description |
| --- | --- | --- |
| Message production rate | count/s | - |
| Message production traffic | Byte/s | - |
| Message consumption rate | count/s | - |
| Message consumption traffic | Byte/s | - |
| Message storage size | Count | - |
| Number of consumers | Count | - |
| Number of producers | Count | - |

| Metric Name | Unit | Metric Description |
| --- | --- | --- |
| Quota usage percentage of message production traffic | % | The ratio of message production traffic to the total quota |
| Quota usage percentage of message consumption traffic | % | The ratio of message consumption traffic to the total quota |
| Quota usage percentage of message production count | % | The ratio of produced message count to the total quota |
| Quota usage percentage of message consumption count | % | The ratio of consumed message count to the total quota |
| Number of heaped messages | Count | - |
| Heaped message size | Byte | - |
| Producer count throttling times | Count | - |
| Quota usage percentage of heaped messages | % | The ratio of heaped messages to the total quota |

# Monitoring Metrics for Clusters

| Metric Name | Unit | Metric Description |
| --- | --- | --- |
| Tenant-level message production rate | count/s | - |
| Tenant-level message consumption rate | count/s | - |
| Peak cluster production bandwidth | count/s | - |
| Peak cluster consumption bandwidth | Byte/s | - |
| Tenant-level heaped message size | Byte | - |
| Transactions per second (TPS) of the Pulsar cluster | Count/s | The number of messages processed per second by the cluster |

# Monitoring Metrics for TI-ONE Platform
# TI-ONE Task-Based Modeling

| Metric English Name | Metric Description | Unit | Dimension | Statistical Granularity |
|---|---|---|---|---|
| CfsClientDataReadBandwidth | Instance-dimension TuroCFS single-node server read bandwidth | KBytes/s | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| CfsClientDataReadBandwidth | Instance-dimension TuroCFS single-node server read bandwidth | KBytes/s | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| CfsClientDataWriteBandwidth | Instance-dimension TuroCFS single-node server write bandwidth | KBytes/s | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| CfsDataReadIoBytes | Instance-dimension CFS server read bandwidth | KBytes/s | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| CfsDataReadIoLatency | Instance-dimension CFS read latency | ms | AppId<br><br>InstanceId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg |

| | | | | |
|---|---|---|---|---|
| | | | SubUin | ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| CfsDataWriteIoBytes | Instance-dimension CFS server write bandwidth | KBytes/s | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| CfsDataWriteIoLatency | Instance-dimension CFS write latency | ms | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| CfsStrageUsageGb | Instance-dimension CFS storage data capacity | GBytes | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Cpuutil | Task-dimension CPU utilization | % | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DcgmFiDevFbUsed | GPU memory usage in the task instance-GPU card dimension | MBytes | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ] |

| | | | | |
|---|---|---|---|---|
| | | | | [ 86400s, avg ] |
| DcgmFiDevGpuUtil | GPU utilization in the task instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DcgmFiDevMemCopyUtil | GPU memory utilization in the task instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DiskIoUtil | Instance-dimension disk I/O utilization | % | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DiskIoWait | Instance-dimension disk I/O wait time | % | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DiskReadByte | Instance-dimension disk read bandwidth | MBytes/s | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| DiskReadIops | Instance-dimension disk read IOPS | Count | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DiskUsageRadio | Instance-dimension system disk partition utilization | % | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DiskWriteByte | Instance-dimension disk write bandwidth | MBytes/s | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DiskWriteIops | Instance-dimension disk write IOPS | Count | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Fp16EngineActivity | Instance-dimension FP16 active time ratio | % | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Fp32EngineActivity | Instance-dimension FP32 | % | AppId<br>InstanceId | [ 10s, avg ]<br>[ 60s, avg ] |

| | | | | |
|---|---|---|---|---|
| | active time ratio | | SubUin | [ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Fp64EngineActivity | Instance-dimension FP64 active time ratio | % | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuFp16EngineActivity | FP16 active time ratio in the task instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuFp32EngineActivity | FP32 active time ratio in the task instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuFp64EngineActivity | FP64 active time ratio in the task instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Gpumemutil | Task-dimension GPU memory utilization | % | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, |

| | | | | |
|---|---|---|---|---|
| | | | | avg ]<br>[ 86400s, avg ] |
| Gpumemvalue | Task-dimension GPU memory usage | MBytes | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuNvlinkBandwidth | NVLink transmission rate in the task instance-GPU card dimension | Bytes/s | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuPcieBandwidth | PCIe bus transmission rate in the task instance-GPU card dimension | Bytes/s | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuSmActivity | SM active time ratio in the task instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| GpuTensorActivity | Tensor active time ratio in the task instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Gpuutil | Task-dimension GPU utilization | % | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Instancecpuutil | Instance-dimension CPU utilization | % | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Instancegpumemutil | Instance-dimension GPU memory utilization | % | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Instancegpumemvalue | Instance-dimension GPU memory usage | MBytes | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Instancegpuutil | GPU utilization in the task | % | AppId<br>InstanceId | [ 10s, avg ]<br>[ 60s, avg ] |

| | | | | |
|---|---|---|---|---|
| | instance dimension | | SubUin | [ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Instancememutil | Instance-dimension memory utilization | % | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Instancememvalue | Instance-dimension memory usage | MBytes | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Memutil | Task-dimension memory utilization | % | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Memvalue | Task-dimension memory usage | MBytes | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| NvlinkBandwidth | Instance-dimension NVLink transmission rate | Bytes/s | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| PcieBandwidth | Instance-dimension PCIe bus transmission rate | Bytes/s | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| RdmaInpkt | Instance-dimension RDMA NIC inbound packet count | pps | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| RdmaIntraffic | Instance-dimension RDMA NIC receive bandwidth | Mbps | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| RdmaOutpkt | Instance-dimension RDMA NIC outbound packet count | pps | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| RdmaOuttraffic | Instance-dimension | Mbps | AppId<br>InstanceId | [ 10s, avg ]<br>[ 60s, avg ] |

| | | | | |
|---|---|---|---|---|
| | RDMA NIC transmit bandwidth | | SubUin | [ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| SmActivity | Instance-dimension SM active time ratio | % | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskCfsClientDataReadBandwidth | Task-dimension TuroCFS single-node server read bandwidth | KBytes/s | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskCfsClientDataWriteBandwidth | Task-dimension TuroCFS single-node server write bandwidth | KBytes/s | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskCfsDataReadIoBytes | Task-dimension CFS server read bandwidth | KBytes/s | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskCfsDataReadIoLatency | Task-dimension CFS read latency | ms | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, |

| | | | | |
|---|---|---|---|---|
| | | | | avg ]<br>[ 86400s, avg ] |
| TaskCfsDataWriteIoBytes | Task-dimension CFS server write bandwidth | KBytes/s | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskCfsDataWriteIoLatency | Task-dimension CFS write latency | ms | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskCfsStrageUsageGb | Task-dimension CFS storage data capacity | GBytes | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskDiskIoUtil | Task-dimension disk I/O utilization | % | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| TaskDiskIoWait | Task-dimension disk I/O wait time | % | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskDiskReadByte | Task-dimension disk read bandwidth | MBytes/s | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskDiskReadIops | Task-dimension disk read IOPS | Count | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskDiskUsageRadio | Task-dimension system disk partition utilization | % | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskDiskWriteByte | Task-dimension disk write bandwidth | MBytes/s | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskDiskWriteIops | Task-dimension disk write IOPS | Count | AppId<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ] |

| | | | | |
|---|---|---|---|---|
| | | | SubUin | [ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskFp16EngineActivity | Task-dimension FP16 active time ratio | % | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskFp32EngineActivity | Task-dimension FP32 active time ratio | % | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskFp64EngineActivity | Task-dimension FP64 active time ratio | % | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskNvlinkBandwidth | Task-dimension NVLink transmission rate | Bytes/s | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskPcieBandwidth | Task-dimension PCIe bus transmission rate | Bytes/s | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, |

| | | | | |
|---|---|---|---|---|
| | | | | avg ]<br>[ 86400s, avg ] |
| TaskRdmaInpkt | Task-dimension RDMA NIC inbound packet count | pps | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskRdmaIntraffic | Task-dimension RDMA NIC receive bandwidth | Mbps | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskRdmaOutpkt | Task-dimension RDMA NIC outbound packet count | pps | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskRdmaOuttraffic | Task-dimension RDMA NIC transmit bandwidth | Mbps | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| TaskSmActivity | Task-dimension SM active time ratio | % | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskTensorActivity | Task-dimension Tensor active time ratio | % | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TensorActivity | Instance-dimension Tensor active time ratio | % | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuDecUtil | Instance-dimension GPU decoder utilization | % | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuEncUtil | Instance-dimension GPU encoder utilization | % | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuMemoryClock | Instance-dimension GPU | s | AppId<br>InstanceId | [ 10s, avg ]<br>[ 60s, avg ] |

| | | | | |
|---|---|---|---|---|
| | memory frequency | | SubUin | [ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuMemoryFree | Instance-dimension available GPU memory | MBytes | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuMemoryUtil | Instance-dimension GPU memory utilization | % | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuNvlinkRxMb | Instance-dimension NVLink received data volume | Mbps | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuNvlinkTxMb | Instance-dimension NVLink transmitted data volume | Mbps | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuPcieRxMb | Instance-dimension PCIe received data volume | Mbps | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, |

| | | | | |
|---|---|---|---|---|
| | | | | avg ]<br>[ 86400s, avg ] |
| GpuPcieTxMb | Instance-dimension PCIe transmitted data volume | Mbps | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuSmClock | Instance-dimension SM clock frequency | s | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskGpuDecUtil | Task-dimension GPU decoder utilization | % | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskGpuEncUtil | Task-dimension GPU encoder utilization | % | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| TaskGpuMemoryClock | Task-dimension GPU memory frequency | s | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskGpuMemoryFree | Task-dimension available GPU memory | MBytes | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskGpuMemoryUtil | Task-dimension GPU memory bandwidth utilization | % | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskGpuNvlinkRxMb | Task-dimension NVLink received data volume | Mbps | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskGpuNvlinkTxMb | Task-dimension NVLink transmitted data volume | Mbps | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskGpuPcieRxMb | Task-dimension PCIe received | Mbps | AppId<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ] |

| | | | | |
|---|---|---|---|---|
| | data volume | | SubUin | [ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskGpuPcieTxMb | Task-dimension PCIe transmitted data volume | Mbps | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskGpuSmClock | Task-dimension SM clock frequency | s | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskGpuDecUtilGpu | GPU decoder utilization in the task instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskGpuEncUtilGpu | GPU encoder utilization in the task instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| TaskGpuMemoryClockGpu | GPU memory frequency in the task instance-GPU card dimension | s | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskGpuMemoryFreeGpu | Available GPU memory in the task instance-GPU card dimension | MBytes | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskGpuMemoryUtilGpu | GPU memory bandwidth utilization in the task instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskGpuNvlinkRxMbGpu | NVLink received data volume in the task instance-GPU card dimension | Mbps | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskGpuNvlinkTxMbGpu | NVLink transmitted data volume in the task instance-GPU card dimension | Mbps | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskGpuPcieRxMbGpu | PCIe received data volume in | Mbps | AppId<br>InstanceGpuNum | [ 10s, avg ]<br>[ 60s, avg ] |

| | | | | |
|---|---|---|---|---|
| | the task instance-GPU card dimension | | SubUin | [ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskGpuPcieTxMbGpu | PCIe transmitted data volume in the task instance-GPU card dimension | Mbps | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskGpuSmClockGpu | SM clock frequency in the task instance-GPU card dimension | s | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

# TI-ONE Development Machine

| Metric English Name | Metric Description | Unit | Dimension | Statistical Granularity |
|---|---|---|---|---|
| CfsClientDataReadBandwidth | Instance-dimension TuroCFS single-node server read bandwidth | KBytes/s | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| CfsClientDataWriteBandwidth | Instance-dimension TuroCFS single-node server write bandwidth | KBytes/s | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| CfsDataReadIoBytes | Instance-dimension CFS server read bandwidth | KBytes/s | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| CfsDataReadIoLatency | Instance-dimension CFS read latency | ms | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| CfsDataWriteIoBytes | Instance-dimension CFS server write bandwidth | KBytes/s | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| CfsDataWriteIoLatency | Instance-dimension CFS write latency | ms | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| CfsStrageUsageGb | Instance-dimension CFS storage data capacity | GBytes | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DiskIoUtil | Instance-dimension disk I/O utilization | % | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DiskIoWait | Instance-dimension disk I/O wait time | % | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DiskReadByte | Instance-dimension disk read bandwidth | MBytes/s | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DiskReadIops | Instance-dimension disk read IOPS | Count | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| DiskUsageRadio | Instance-dimension system disk partition utilization | % | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DiskWriteByte | Instance-dimension disk write bandwidth | MBytes/s | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DiskWriteIops | Instance-dimension disk write IOPS | Count | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Instancecpuutil | Instance-dimension CPU utilization | % | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Instancegpumemutil | Instance-dimension GPU memory utilization | % | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Instancegpumemvalue | Instance-dimension GPU memory usage | MBytes | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| Instancegpuutil | Instance-dimension GPU utilization | % | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Instancememutil | Instance-dimension memory utilization | % | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Instancememvalue | Instance-dimension memory usage | MBytes | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuFp16EngineActivity | FP16 active time ratio in the instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuFp32EngineActivity | FP32 active time ratio in the instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuFp64EngineActivity | FP64 active time ratio in the instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| NvlinkBandwidth | NVLink transmission rate in the instance-GPU card dimension | Bytes/s | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| PcieBandwidth | PCIe bus transmission rate in the instance-GPU card dimension | Bytes/s | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuSmActivity | SM active time ratio in the instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TensorActivity | Tensor active time ratio in the instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Dcgmfidevfbused | GPU memory usage in the instance-GPU card dimension | MBytes | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DcgmFiDevGpuUtil | GPU utilization in the instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| DcgmFiDevMemCopyUtil | GPU memory utilization in the instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuMemoryClockGpu | GPU memory frequency in the instance-GPU card dimension | s | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuMemoryFreeGpuv | Available GPU memory in the instance-GPU card dimension | MBytes | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuNvlinkRxMb | NVLink received data volume in the instance-GPU card dimension | Mbps | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuNvlinkTxMb | NVLink transmitted data volume in the instance-GPU card dimension | Mbps | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuPcieRxMb | PCIe received data volume in the instance-GPU card dimension | Mbps | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| GpuPcieTxMb | PCIe transmitted data volume in the instance-GPU card dimension | Mbps | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuSmClock | SM clock frequency in the instance-GPU card dimension | s | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| PodDiskLimit | Instance-dimension total disk size | GBytes | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| PodDiskValue | Instance-dimension disk usage | GBytes | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| NodeDiskLimit | Instance-dimension total node disk size | GBytes | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| NodeDiskValue | Instance-dimension node disk usage | GBytes | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| RdmaInpkt | Instance-dimension RDMA NIC inbound packet count | pps | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| RdmaOutpkt | Instance-dimension RDMA NIC outbound packet count | pps | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| RdmaIntraffic | Instance-dimension RDMA NIC receive bandwidth | Mbps | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| RdmaOuttraffic | Instance-dimension RDMA NIC transmit bandwidth | Mbps | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

# TI-ONE Online Services

| Metric English Name | Metric Description | Unit | Dimension | Statistical Granularity |
|---|---|---|---|---|
| Apicallerrortotal | Number of failed API calls | Count | ServiceGroupId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Apicalllimittotal | Total number of API calls throttled | Count | ServiceGroupId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Apicallsuccesstotal | Total number of successful API calls | Count | ServiceGroupId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Apicalltotal | Total number of API calls | Count | ServiceGroupId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Apiresponsetime | Average response time | ms | ServiceGroupId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ] |

| | | | | |
|---|---|---|---|---|
| | | | | [ 86400s, avg ] |
| CfsClientDataReadBandwidth | TuroCFS single-node server read bandwidth in the task instance dimension | KBytes/s | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| CfsClientDataWriteBandwidth | TuroCFS single-node server write bandwidth in the task instance dimension | KBytes/s | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| CfsDataReadIoBytes | CFS server read bandwidth in the task instance dimension | KBytes/s | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| CfsDataReadIoLatency | CFS read latency in the task instance dimension | ms | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| CfsDataWriteIoBytes | CFS server write bandwidth in the task instance dimension | KBytes/s | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| CfsDataWriteIoLatency | CFS write latency in the task instance dimension | ms | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| CfsStrageUsageGb | CFS storage data capacity in the task instance dimension | GBytes | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Cpuutil | Task-dimension CPU utilization | % | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DiskIoUtil | Disk I/O utilization in the task instance dimension | % | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DiskIoWait | Disk I/O wait time in the task instance dimension | % | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DiskReadByte | Disk read bandwidth in the task | MBytes/s | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg |

| | | | | |
|---|---|---|---|---|
| | instance dimension | | | ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DiskReadIops | Disk read IOPS in the task instance dimension | Count | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DiskUsageRadio | System disk partition utilization in the task instance dimension | % | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DiskWriteByte | Disk write bandwidth in the task instance dimension | MBytes/s | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DiskWriteIops | Disk write IOPS in the task instance dimension | Count | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Gpumemutil | Task-dimension GPU memory utilization | % | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, |

| | | | | |
|---|---|---|---|---|
| | | | | avg ] |
| Gpuutil | Task-dimension GPU utilization | % | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Instancecpuutil | CPU utilization in the task instance dimension | % | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Instancegpumemutil | GPU memory utilization in the task instance dimension | % | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Instancegpuutil | GPU utilization in the task instance dimension | % | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Instancehttpqps | HTTP requests per second in the task instance dimension | Count/s | InstanceId<br>Source<br>SubUin | [ 10s, max ]<br>[ 60s, max ]<br>[ 300s, max ]<br>[ 3600s, max ]<br>[ 86400s, max ] |

| | | | | |
|---|---|---|---|---|
| Instancehttpqpslimit | HTTP requests per second limited in the task instance dimension | Count/s | InstanceId<br>Source<br>SubUin | [ 10s, max ]<br>[ 60s, max ]<br>[ 300s, max ]<br>[ 3600s, max ]<br>[ 86400s, max ] |
| Instancememutil | Memory utilization in the task instance dimension | % | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Instancememvalue | Memory usage in the task instance dimension | MBytes | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Instancenetworkibytes | Inbound network traffic in the task instance dimension | MBytes | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Instanceready | Number of running instances in the task dimension | Count | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, last ]<br>[ 60s, last ]<br>[ 300s, last ]<br>[ 3600s, last ]<br>[ 86400s, last ] |

| | | | | |
|---|---|---|---|---|
| InstanceTiemsCurrentRequests | Number of concurrent requests in the task instance dimension | Count | InstanceId<br>Source<br>SubUin | [ 10s, max ]<br>[ 60s, max ]<br>[ 300s, max ]<br>[ 3600s, max ]<br>[ 86400s, max ] |
| Instancetotal | Number of instances in the task dimension | Count | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, last ]<br>[ 60s, last ]<br>[ 300s, last ]<br>[ 3600s, last ]<br>[ 86400s, last ] |
| Memutil | Task-dimension memory utilization | % | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Memvalue | Task-dimension memory usage | MBytes | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Networkreceivebytes | Inbound network traffic in the task dimension | MBytes | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| ServiceCfsClientDataReadBandwidth | Task-dimension TuroCFS single-node server read bandwidth | KBytes/s | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| ServiceCfsClientDataWriteBandwidth | Task-dimension TuroCFS single-node server write bandwidth | KBytes/s | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| ServiceCfsDataReadIoBytes | Task-dimension CFS server read bandwidth | KBytes/s | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| ServiceCfsDataReadIoLatency | Task-dimension CFS read latency | ms | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| ServiceCfsDataWriteIoBytes | Task-dimension CFS server write bandwidth | KBytes/s | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| ServiceCfsDataWriteIoLatency | Task-dimension CFS write | ms | AppId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg |

| | | | | |
|---|---|---|---|---|
| | latency | | TaskId | ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| ServiceCfsStrageUsageGb | Task-dimension CFS storage data capacity | GBytes | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| ServiceDiskIoUtil | Task-dimension disk I/O utilization | % | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| ServiceDiskIoWait | Task-dimension disk I/O wait time | % | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| ServiceDiskReadByte | Task-dimension disk read bandwidth | MBytes/s | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| ServiceDiskReadIops | Task-dimension disk read IOPS | Count | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, |

| | | | | |
|---|---|---|---|---|
| | | | | avg ] |
| ServiceDiskUsageRadio | Task-dimension system disk partition utilization | % | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| ServiceDiskWriteByte | Task-dimension disk write bandwidth | MBytes/s | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| ServiceDiskWriteIops | Task-dimension disk write IOPS | Count | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Servicehttpqps | HTTP requests per second in the task dimension for the service | Count/s | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, max ]<br>[ 60s, max ]<br>[ 300s, max ]<br>[ 3600s, max ]<br>[ 86400s, max ] |
| Servicehttpqpslimit | Rate-limited HTTP requests per second in the task dimension for the service | Count/s | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, max ]<br>[ 60s, max ]<br>[ 300s, max ]<br>[ 3600s, max ]<br>[ 86400s, |

| | | | | |
|---|---|---|---|---|
| | | | | max ] |
| ServiceTiemsCurrentRequests | Number of concurrent requests in the task dimension | Count | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, max ]<br>[ 60s, max ]<br>[ 300s, max ]<br>[ 3600s, max ]<br>[ 86400s, max ] |
| ServiceGpuMemValue | Task-dimension GPU memory usage | MBytes | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| ServiceEmsTokenThroughput | Number of tokens processed per minute in the task dimension | Count | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| ServiceEmsTokenThroughputInput | Number of input tokens processed per minute in the task dimension | Count | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| ServiceEmsTokenThroughputOutput | Number of output tokens processed per minute in the task dimension | Count | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| ServiceEmsFirstTokenLatency | Task-dimension first token latency | s | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| ServiceEmsNonFirstTokenLatency | Task-dimension non-first token latency | s | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| ServiceEmsProcessingRequestCount | Number of requests in processing in the task dimension | Count | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, sum ]<br>[ 60s, sum ]<br>[ 300s, sum ]<br>[ 3600s, sum ]<br>[ 86400s, sum ] |
| ServiceEmsQueuingRequestCount | Number of queued requests in the task dimension | Count | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, sum ]<br>[ 60s, sum ]<br>[ 300s, sum ]<br>[ 3600s, sum ]<br>[ 86400s, sum ] |
| ServiceEmsTotalProcessedTokens | Total number of tokens processed in the task dimension | Count | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, sum ]<br>[ 60s, sum ]<br>[ 300s, sum ]<br>[ 3600s, sum ] |

| | | | | |
|---|---|---|---|---|
| | | | | [ 86400s, sum ] |
| ServiceEmsTotalProcessedTokensInput | Total number of output tokens in the task dimension | Count | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, sum ]<br>[ 60s, sum ]<br>[ 300s, sum ]<br>[ 3600s, sum ]<br>[ 86400s, sum ] |
| ServiceEmsTotalProcessedTokensOutput | Total number of output tokens in the task dimension | Count | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, sum ]<br>[ 60s, sum ]<br>[ 300s, sum ]<br>[ 3600s, sum ]<br>[ 86400s, sum ] |
| ServiceEmsAverageLengthInput | Average input length in tokens in the task dimension | Count | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| ServiceEmsAverageLengthOutput | Average output length in tokens in the task dimension | Count | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuMemValue | GPU memory usage in the task instance dimension | MBytes | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ] |

| | | | | |
|---|---|---|---|---|
| | | | | [ 86400s, avg ] |
| EmsTokenThroughput | Number of tokens processed per minute in the task instance dimension | Count | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| EmsTokenThroughputInput | Total number of input tokens processed per minute in the task instance dimension | Count | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| EmsTokenThroughputOutput | Total number of output tokens processed per minute in the task instance dimension | Count | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| EmsFirstTokenLatency | First token latency in the task instance dimension | s | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| EmsNonFirstTokenLatency | Non-first token latency in the task instance dimension | s | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| EmsProcessingRequestCount | Number of requests in processing in the task instance dimension | Count | InstanceId<br>Source<br>SubUin | [ 10s, sum ]<br>[ 60s, sum ]<br>[ 300s, sum ]<br>[ 3600s, sum ]<br>[ 86400s, sum ] |
| EmsQueuingRequestCount | Number of queued requests in the task instance dimension | Count | InstanceId<br>Source<br>SubUin | [ 10s, sum ]<br>[ 60s, sum ]<br>[ 300s, sum ]<br>[ 3600s, sum ]<br>[ 86400s, sum ] |
| EmsTotalProcessedTokens | Total number of tokens processed in the task instance dimension | Count | InstanceId<br>Source<br>SubUin | [ 10s, sum ]<br>[ 60s, sum ]<br>[ 300s, sum ]<br>[ 3600s, sum ]<br>[ 86400s, sum ] |
| EmsTotalProcessedTokensInput | Total number of input tokens processed in the task instance dimension | Count | InstanceId<br>Source<br>SubUin | [ 10s, sum ]<br>[ 60s, sum ]<br>[ 300s, sum ]<br>[ 3600s, sum ]<br>[ 86400s, sum ] |
| EmsTotalProcessedTokensOutput | Total number of output tokens | Count | InstanceId<br>Source<br>SubUin | [ 10s, sum ]<br>[ 60s, sum ] |

| | | | | |
|---|---|---|---|---|
| | processed in the task instance dimension | | | [ 300s, sum ]<br>[ 3600s, sum ]<br>[ 86400s, sum ] |
| EmsAverageLengthInput | Average input length (tokens) in the task instance dimension | Count | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| EmsAverageLengthOutput | Average output length (tokens) in the task instance dimension | Count | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Fp16EngineActivity | FP16 active time ratio in the service instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Fp32EngineActivity | FP32 active time ratio in the service instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| Fp64EngineActivity | FP64 active time ratio in the service instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| NvlinkBandwidth | NVLink transmission rate in the service instance-GPU card dimension | Bytes/s | AppId<br>InstanceGpuNum<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| PcieBandwidth | PCIe bus transmission rate in the service instance-GPU card dimension | Bytes/s | AppId<br>InstanceGpuNum<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| SmActivity | SM active time ratio in the service instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TensorActivity | Tensor active time ratio in the service instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DcgmFiDevFbUsed | GPU memory usage in the | MBytes | AppId<br>InstanceGpuNum<br>Source | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg |

| | | | | |
|---|---|---|---|---|
| | service instance-GPU card dimension | | SubUin | ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DcgmFiDevGpuUtil | GPU utilization in the service instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DcgmFiDevMemCopyUtil | GPU memory utilization in the service instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuDecUtil | GPU decoder utilization in the service instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuEncUtil | GPU encoder utilization in the service instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| GpuMemoryClock | GPU memory frequency in the service instance-GPU card dimension | S | AppId<br>InstanceGpuNum<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuNvlinkRxMb | NVLink received data volume in the service instance-GPU card dimension | Mbps | AppId<br>InstanceGpuNum<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuNvlinkTxMb | NVLink transmitted data volume in the service instance-GPU card dimension | Mbps | AppId<br>InstanceGpuNum<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuPcieRxMb | PCIe received data volume in the service instance-GPU card dimension | Mbps | AppId<br>InstanceGpuNum<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuPcieTxMb | PCIe transmitted data volume in the service instance-GPU card dimension | Mbps | AppId<br>InstanceGpuNum<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuSmClock | SM clock frequency in the service | S | AppId<br>InstanceGpuNum<br>Source | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg |

| | | | | |
|---|---|---|---|---|
| | instance-GPU card dimension | | SubUin | ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

# TI-ONE Resource Management

| Metric English Name | Metric Description | Unit | Dimension | Statistical Granularity |
|---|---|---|---|---|
| GroupCpuUsage | CPU utilization of the resource group | % | ResourceGroupId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GroupGpuUtil | GPU utilization of the resource group | % | ResourceGroupId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GroupLanInTraffic | Inbound private network bandwidth of the resource group | Mbps | ResourceGroupId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GroupLanOutTraffic | Outbound private network bandwidth of the resource group | Mbps | ResourceGroupId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GroupMemUsage | Memory utilization of the resource group | % | ResourceGroupId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GroupWanInTraffic | Inbound public network bandwidth of the resource group | Mbps | ResourceGroupId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| GroupWanOutTraffic | Outbound public network bandwidth of the resource group | Mbps | ResourceGroupId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| InsCpuUsage | CPU utilization of the resource (node) | % | ResourceId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| InsGpuUtil | GPU utilization of the resource (node) | % | ResourceId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| InsMemUsage | Memory utilization of the resource (node) | % | ResourceId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| InsLanInTraffic | Inbound private network bandwidth of the resource (node) | Mbps | ResourceId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| InsLanOutTraffic | Outbound private network bandwidth of the resource (node) | Mbps | ResourceId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| InsWanInTraffic | Inbound public network bandwidth of the resource (node) | Mbps | ResourceId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| InsWanOutTraffic | Outbound public network bandwidth of the resource (node) | Mbps | ResourceId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GroupWanOutratio | Public network bandwidth utilization of the resource group | % | ResourceGroupId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GroupFp16EngineActivity | FP16 active time ratio of the resource group | % | ResourceGroupId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GroupFp32EngineActivity | FP32 active time ratio of the resource group | % | ResourceGroupId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GroupFp64EngineActivity | FP64 active time ratio of the resource group | % | ResourceGroupId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| GroupGpuDecUtil | GPU decoder utilization of the resource group | % | ResourceGroupId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GroupGpuEccPersistent | Persistent ECC errors of the resource group | Count | ResourceGroupId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GroupGpuEccVolatile | Volatile ECC errors of the resource group | Count | ResourceGroupId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GroupGpuEncUtil | GPU encoder utilization of the resource group | % | ResourceGroupId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GroupGpuPowerUsage | GPU power consumption of the resource group | W | ResourceGroupId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GroupGpuRetiredPages | Disabled memory pages of the resource group | Count | ResourceGroupId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| GroupGpuTemperature | GPU temperature of the resource group | °C | ResourceGroupId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GroupGpuXidErrors | GPU XID count of the resource group | Count | ResourceGroupId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| InsWanOutratio | Public network bandwidth utilization of the resource (node) | % | ResourceId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| InsGpuDecUtil | GPU decoder utilization of the resource (node) | % | ResourceId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| InsGpuEccPersistent | Persistent ECC errors of the resource (node) | Count | ResourceId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| InsGpuEccVolatile | Volatile ECC errors of the resource (node) | Count | ResourceId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| InsGpuEncUtil | GPU encoder utilization of the resource (node) | % | ResourceId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| InsGpuPowerUsage | GPU power consumption of the resource (node) | W | ResourceId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| InsGpuRetiredPages | Disabled memory pages of the resource (node) | Count | ResourceId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| InsGpuTemperature | GPU temperature of the resource (node) | °C | ResourceId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| InsGpuXidErrors | GPU XID count of the resource (node) | Count | ResourceId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| InsFp16EngineActivity | FP16 active time ratio of the resource (node) | % | ResourceId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| InsFp32EngineActivity | FP32 active time ratio of the resource (node) | % | ResourceId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| InsFp64EngineActivity | FP64 active time ratio of the resource (node) | % | ResourceId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

# Querying Monitoring Metrics: Guide for Calling the GetMonitorData API

## API Call Guide

1. API description

| API Name | Version | API Feature |
| --- | --- | --- |
| GetMonitorData | 2018-07-24 | Pulls the monitoring data of cloud product metrics. |

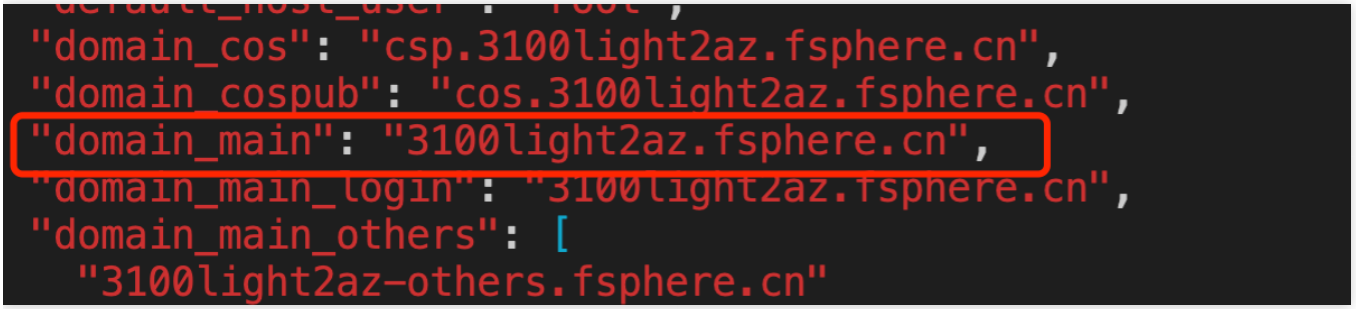
2. Input parameters

| Parameter Name | Required | Type | Description |
| --- | --- | --- | --- |
| Namespace | Yes | String | Namespace, such as qce/cvm.<br>Example value: qce/cvm. |
| MetricName | Yes | String | Metric name, such as cpu_usage. Only single-metric pulling is supported.<br>Example value: cpu_usage |
| Dimensions | Yes | Array | Dimensions of instance objects, in the form of a collection of key-value pairs. This field varies with the instance types. For example, this field is [{"unInstanceId": "ins-miltleuc"}] for a CVM instance. Each request supports batch retrieval of monitoring data for up to 10 instances.<br>Example value: [{"unInstanceId": "ins-miltleuc"}, {"unInstanceId": "ins-4xny0cbt"}] |
| Period | Yes | Integer | Monitoring statistical period, such as 60, in seconds. The statistical period supported by each metric may not be the same. For the statistical periods supported by each cloud product, see the appendix of this document. The maximum number of data points of a single request is 1,440.<br>Example value: 300. |
| StartTime | Yes | Time | Start time, for example, 2018-09-22T19:51:23+08:00.<br>Example value: 2019-03-24T10:51:23+08:00. |
| EndTime | Yes | Time | End time, such as 2018-09-22T20:51:23+08:00. The current time is used by default. EndTime cannot be earlier than StartTime.<br>Example value: 2019-03-24T20:51:23+08:00. |

3. Request domain name

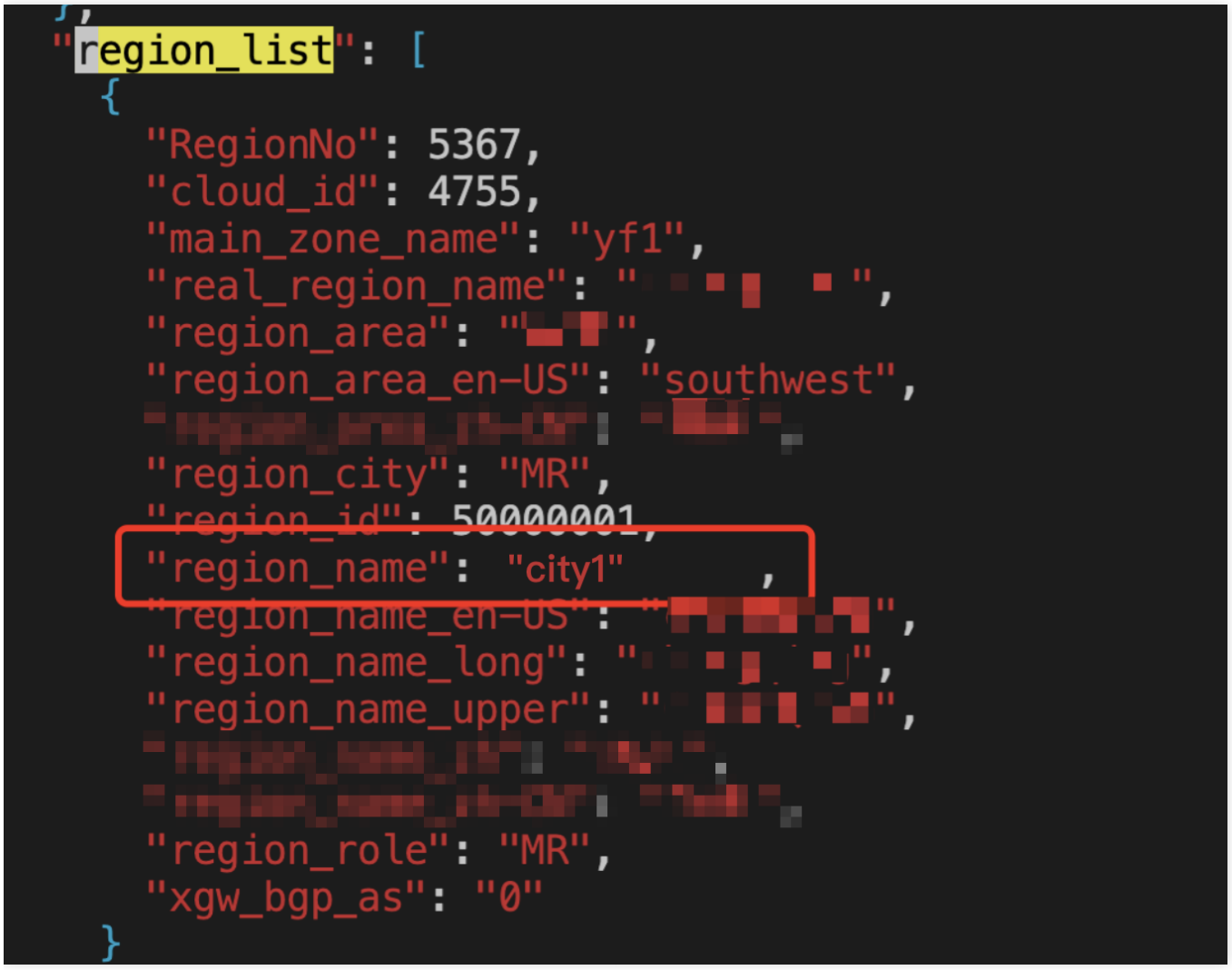
API request domain name: monitor.api3.${domain_main}, such as monitor.api3.3100light2az.fsphere.cn.

> Note:
>
> ${domain_main} can be obtained by logging into any Pod and searching for the domain_main keyword in /tce/conf/cm/local.json.




4. Request region

Since monitoring data is reported and stored by region, it is necessary to explicitly specify the region where the instance to be pulled is located when an API is called. The method to obtain region information is as follows: Log in to any Pod, open /tce/conf/cm/local.json, search for the region_list keyword, and the region_name indicates available regions.




5. Calling frequency limit

Default API request frequency limit: 20 times/second.

6. Output parameters

| Parameter Name | Type | Description |
| --- | --- | --- |
| MetricName | String | Metric name.<br>Example value: cpu_usage. |
| Period | Integer | Monitoring statistical period.<br>Example value: 60. |
| DataPoints[].Dimensions | Json | Instance object dimension group.<br>Example value: {"unInstanceId": "ins-miltleuc"}. |
| DataPoints[].Points | Array | Data point array. |
| StartTime | Time | Start time.<br>Example value: 2019-03-24T10:50:00+08:00. |

| Parameter Name | Type | Description |
| --- | --- | --- |
| EndTime | Time | End time.<br>Example value: 2019-03-24T20:50:00+08:00. |
| RequestId | String | Unique request ID. It is generated by the server and is returned for each request. (No RequestId is returned if the request is not received by the server due to certain reasons.) RequestId is required for locating an issue. |

# API Call Example

Scenario: Retrieve CPU utilization monitoring data for a CVM instance over a specified time range, with a 5-minute statistical period.

1. Input example

```
POST HTTP/1.1
Host: monitor.api3.3100light2az.fsphere.cn
Content-Type: application/json
Action: GetMonitorData
{
    "Namespace": "qce/cvm",
    "MetricName": "CPUUsage",
    "Period": 300,
    "Dimensions": [
        {
            "unInstanceId": "ins-miltleuc"
        }
    ],
    "StartTime": "2024-03-24T10:51:23+08:00",
    "EndTime": "2024-03-24T20:51:23+08:00"
}
```

2. Output example

```
{
    "Response": {
        "StartTime": "2019-03-24 10:50:00",
        "EndTime": "2019-03-24 20:50:00",
        "Period": 300,
        "MetricName": "cpu_usage",
        "DataPoints": [
            {
                "Dimensions": {
                        "unInstanceId": "ins-miltleuc"},
```

```
            "Points": [
                2.566,
                2.283,
                6.316,
                2.816,
                2.7,
                2.35
            ]
        }
    ],
    "RequestId": "d96ec542-6547-4af2-91ac-fee85c1b8b85"
  }
}
```


# SDK Code Call

Pull CPU utilization monitoring data with a 5-minute statistical period for a CVM instance over a period.

1. Golang example

```
package main

import (
    "fmt"

    "github.com/tcecloud/tcecloud-sdk-go/tcecloud/common"
    "github.com/tcecloud/tcecloud-sdk-go/tcecloud/common/errors"
    "github.com/tcecloud/tcecloud-sdk-go/tcecloud/common/profile"
    monitor "github.com/tcecloud/tcecloud-sdk-go/tcecloud/monitor/v20170312"
)

func main() {
    secretId := "AKIDxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx"        // Calling credentials. Refer to the appendix.
    secretKey := "eyzi69sGK3SWFeiblb8NKwD9l2yZplNf"          // Calling credentials. Refer to the appendix.

    credential := common.NewCredential(secretId, secretKey)
    cpf := profile.NewClientProfile()
    cpf.HttpProfile.Scheme = "HTTP"
    cpf.HttpProfile.ReqTimeout = 10
    cpf.HttpProfile.Endpoint = "monitor.api3.3100light2az.fsphere.cn"      // Calling domain name. Refer to 3. R
equest Domain Name in this chapter.
    client, _ := monitor.NewClient(credential, "city1", cpf)               // Region where the monitoring data of clou
d product instances is located. Refer to 4. Request Region in this chapter.

    request := monitor.NewGetMonitorDataRequest()
```

```
    request.Namespace = common.StringPtr("qce/cvm")
    request.MetricName = common.StringPtr("cpu_usage")
    request.Period = common.Uint64Ptr(60)
    request.StartTime = common.StringPtr("2024-10-17T15:10:42+08:00")
    request.EndTime = common.StringPtr("2024-10-17T16:10:42+08:00")

    // Build the Dimensions field.
    dimensions := []string{
        "{\"unInstanceId\": \"ins-miltleuc\"}",
    }
    request.Dimensions = common.StringPtrs(dimensions)
    response, err := client.GetMonitorData(request)

    if _, ok := err.(*errors.CloudSDKError); ok {
        fmt.Printf("An API error has returned: %s", err)
        return
    }
    // Non-SDK exception, direct failure. Other processing can be added to the actual code.
    if err != nil {
        panic(err)
    }
    // Print the returned json string
    fmt.Printf("%s", response.ToJsonString())
}
```

2. Python example

```
# -*- coding: utf8 -*-
#
import json
from tcecloud.common import credential
from tcecloud.common.exception.cloud_sdk_exception import CloudSDKException
from tcecloud.monitor.v20170312 import monitor_client, models
from tcecloud.common.profile.client_profile import ClientProfile
from tcecloud.common.profile.http_profile import HttpProfile
import ssl

ssl._create_default_https_context = ssl._create_unverified_context

try:
    secret_id = "AKIDxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx"          # Calling credentials. Refer to the appendix.
    secret_key = "1qVuzkBH3WgWhX3F7LJyJJyIJ0NP77rM"             # Calling credentials. Refer to the appendi
x.
    cred = credential.Credential(secret_id, secret_key)

    httpProfile = HttpProfile()
    httpProfile.endpoint = "monitor.api3.3100light2az.fsphere.cn"          # Calling domain name. Refer to 3. Re
quest Domain Name in this chapter.
```

```
    clientProfile = ClientProfile()
    clientProfile.httpProfile = httpProfile
    client = monitor_client.MonitorClient(cred, "city1", clientProfile)     # Region where the monitoring data of cloud product instances is located. Refer to 4. Request Region in this chapter.

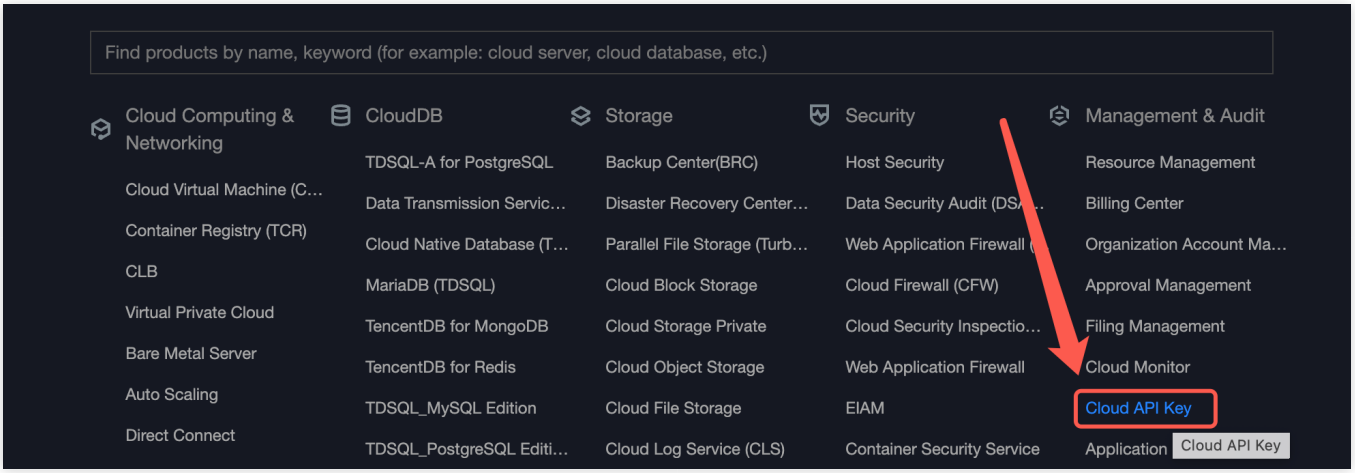
    req = models.GetMonitorDataRequest()
    params = {
        "Namespace": "qce/cvm",
        "MetricName": "cpu_usage",
        "Period": 60,
        "StartTime": "2024-10-17T15:10:42+08:00",
        "EndTime": "2024-10-17T16:10:42+08:00",
        "Dimensions": ['''{"unInstanceId": "ins-miltleuc"}''']
    }
    req.from_json_string(json.dumps(params))
    resp = client.GetMonitorData(req)

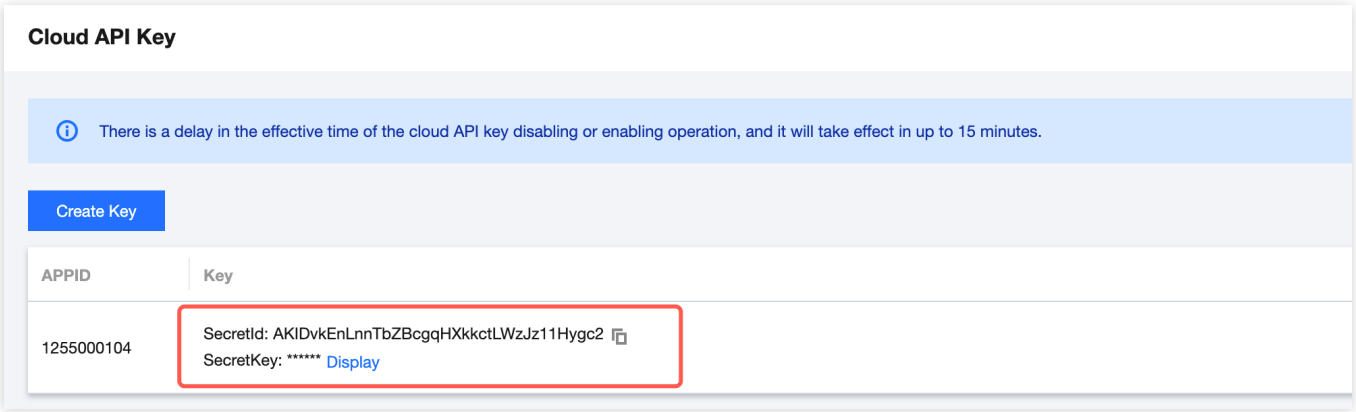
    # Output json format string back package
    print(resp.to_json_string(indent=2))

except CloudSDKException as err:
    print(err)
```

# Appendix: Calling Credentials Creation Process

1. Calling credentials: used for calling authentication, which means that the called AppID and UIN can be identified based on secretId and secretKey.
2. Creation process:




Cloud API Key
There is a delay in the effective time of the cloud API key disabling or enabling operation, and it will take effect in up to 15 minutes.
Create Key
APPID
Key
1255000104
SecretId: AKIDvkEnLnnTbZBcgqHXkkctLWzJz11Hygc2
SecretKey: ****** Display

# Querying Monitoring Metrics: Guide for Calling the GetMonitorDat

1. API description

| API Name | Version | API Feature |
| --- | --- | --- |
| GetMonitorData | 2017/3/12 | Pulls the monitoring data of cloud product metrics. |

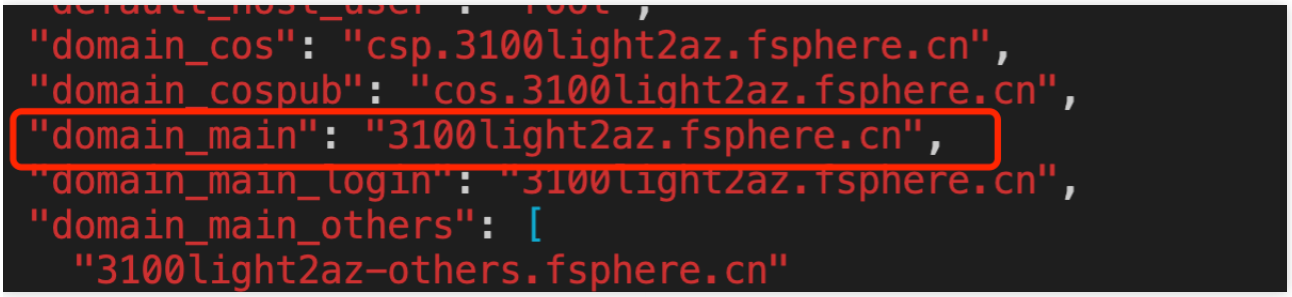
2. Input parameters

| Parameter Name | Required | Type | Description |
| --- | --- | --- | --- |
| Namespace | Yes | String | Namespace, such as qce/cvm. For detailed namespaces of each cloud product, see the appendix of this document.<br>Example value: qce/cvm. |
| MetricName | Yes | String | Metric name, such as cpu_usage. Only single-metric pulling is supported. For detailed metric descriptions of each cloud product, see the appendix of this document.<br>Example value: cpu_usage. |
| Dimensions | Yes | Array | The dimension group of instance objects, in the form of a collection of key-value pairs. The fields of different instance types are completely different. For example, the field of CVM is [{"unInstanceId": "ins-miltleuc"}]. For the dimensions of each cloud product, see the appendix of this document. A single request supports pulling monitoring data for up to 10 instances in batches.<br>Example value: [{"unInstanceId": "ins-miltleuc"}, {"unInstanceId": "ins-4xny0cbt"}] |
| Period | Yes | Integer | Monitoring statistical period, such as 60, in seconds. The statistical period supported by each metric may not be the same. For the statistical periods supported by each cloud product, see the appendix of this document. The maximum number of data points of a single request is 1,440.<br>Example value: 300. |
| StartTime | Yes | Time | Start time, for example, 2018-09-22T19:51:23+08:00.<br>Example value: 2019-03-24T10:51:23+08:00. |
| EndTime | Yes | Time | End time, such as 2018-09-22T20:51:23+08:00. The current time is used by default. EndTime cannot be earlier than StartTime.<br>Example value: 2019-03-24T20:51:23+08:00. |

3. Request domain name

API request domain name: monitor.api3.${domain_main}, such as monitor.api3.3100light2az.fsphere.cn.

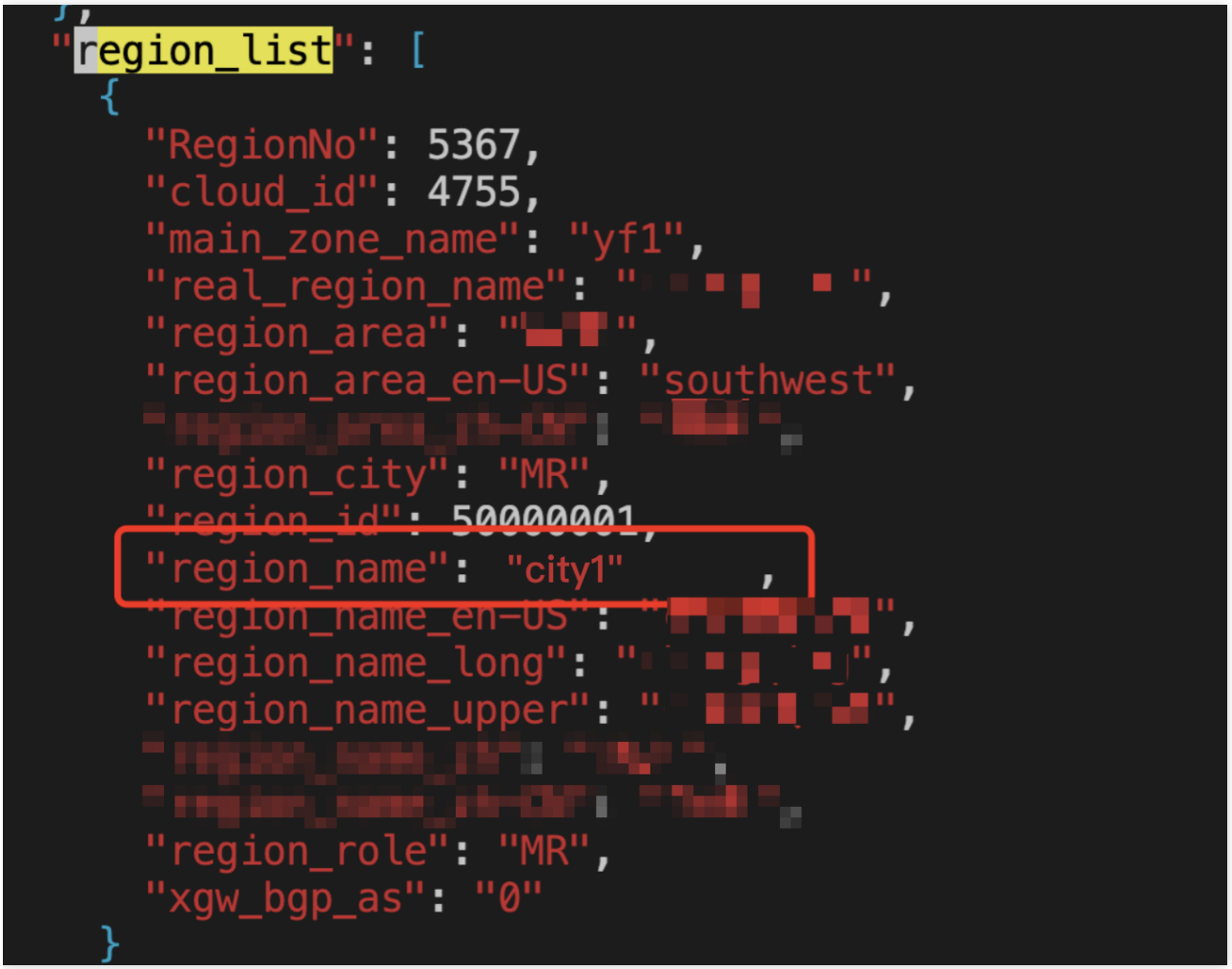
> Note:
> ${domain_main} can be obtained by logging into any Pod, opening /convergecloud/conf/cm/local.json, and searching for the domain_main keyword.




4. Request region

Since monitoring data is reported and stored by region, it is necessary to explicitly specify the region where the instance to be pulled is located when an API is called. The method to obtain region information is as follows:

Log in to any Pod, open /convergecloud/conf/cm/local.json, search for the region_list keyword, and the region_name indicates available regions.




5. Calling frequency limit

Default API request frequency limit: 20 times/second.

6. Output parameters

| Parameter Name | Type | Description |
| --- | --- | --- |
| MetricName | String | Metric name.<br>Example value: cpu_usage. |
| Period | Integer | Monitoring statistical period.<br>Example value: 60. |
| DataPoints[].Dimensions | Json | Instance object dimension group.<br>Example value: {"unInstanceId": "ins-miltleuc"}. |
| DataPoints[].Points | Array | Data point array. |
| StartTime | Time | Start time.<br>Example value: 2019-03-24T10:50:00+08:00. |
| EndTime | Time | End time.<br>Example value: 2019-03-24T20:50:00+08:00. |
| RequestId | String | Unique request ID. It is generated by the server and is returned for each request. (No RequestId is returned if the request is not received by the server due to certain reasons.) RequestId is required for locating an issue. |

7. Examples

Pull CPU utilization monitoring data with a 5-minute statistical period for a CVM instance over a period.

Input example

POST HTTP/1.1 Host: monitor.api3.3100light2az.fsphere.cn Content-Type: application/json Action: GetMonitorData { "Namespace": "qce/cvm", "MetricName": "CPUUsage", "Period": 300, "Dimensions": [ { "unInstanceId": "ins-miltleuc" } ], "StartTime": "2024-03-24T10:51:23+08:00", "EndTime": "2024-03-24T20:51:23+08:00" }

Output example

{ "Response": { "StartTime": "2019-03-24 10:50:00", "EndTime": "2019-03-24 20:50:00", "Period": 300, "MetricName": "cpu_usage", "DataPoints": [ { "Dimensions": { "unInstanceId": "ins-miltleuc" }, "Points": [ 2.566, 2.283, 6.316, 2.816, 2.7, 2.35 ] } ], "RequestId": "d96ec542-6547-4af2-91ac-fee85c1b8b85" } }

8. SDK code calling

Pull CPU utilization monitoring data with a 5-minute statistical period for a CVM instance over a period.

i. Golang example

```
package main

import (
 "fmt"
 "github.com/convergecloud/convergecloud-sdk-go/convergecloud/common"
 "github.com/convergecloud/convergecloud-sdk-go/convergecloud/common/errors"
 "github.com/convergecloud/convergecloud-sdk-go/convergecloud/common/profile"
```

```
 monitor "github.com/convergecloud/convergecloud-sdk-go/convergecloud/monitor/v20170312"
)

func main() {
 secretId := "AKIDxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx"
 secretKey := "eyzi69sGK3SWFeiblb8NKwD9l2yZplNf"

 credential := common.NewCredential(secretId, secretKey)
 cpf := profile.NewClientProfile()
 cpf.HttpProfile.Scheme = "HTTP"
 cpf.HttpProfile.ReqTimeout = 10
 cpf.HttpProfile.Endpoint = "monitor.api3.3100light2az.fsphere.cn"
 client, _ := monitor.NewClient(credential, "city1", cpf)

 request := monitor.NewGetMonitorDataRequest()

 request.Namespace = common.StringPtr("qce/cvm")
 request.MetricName = common.StringPtr("cpu_usage")
 request.Period = common.Uint64Ptr(60)
 request.StartTime = common.StringPtr("2024-10-17T15:10:42+08:00")
 request.EndTime = common.StringPtr("2024-10-17T16:10:42+08:00")

 // Building the Dimensions field
 dimensions := []string{
 "{\"unInstanceId\": \"ins-miltleuc\"}",
 }
 request.Dimensions = common.StringPtrs(dimensions)
 response, err := client.GetMonitorData(request)

 if _, ok := err.(*errors.CloudSDKError); ok {
 fmt.Printf("An API error has returned: %s", err)
 return
 }
 if err != nil {
 panic(err)
 }
 fmt.Printf("%s", response.ToJsonString())
}
```

> Note:
> Lines 13 to 14: secretId and secretKey are calling credentials used for calling authentication, which means that through secretId and secretKey, the called AppID and UIN can be identified. secretId and secretKey are created in the following ways:

Line 20: Endpoint is the calling domain name. For how to obtain it, see 3. Request Domain.
Line 21: "city1" is the region where the monitoring data of cloud product instances is located. For how to obtain it, see 4. Request Region.

2. Python example

```
# -*- coding: utf8 -*-
import json
from convergecloud.common import credential
from convergecloud.common.exception.cloud_sdk_exception import CloudSDKException
from convergecloud.monitor.v20170312 import monitor_client, models
from convergecloud.common.profile.client_profile import ClientProfile
from convergecloud.common.profile.http_profile import HttpProfile
import ssl

ssl._create_default_https_context = ssl._create_unverified_context

try:
 secret_id = "AKIDxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx"
 secret_key = "1qVuzkBH3WgWhX3F7LJyJJyIJ0NP77rM"
 cred = credential.Credential(secret_id, secret_key)

 httpProfile = HttpProfile()
 httpProfile.endpoint = "monitor.api3.3100light2az.fsphere.cn"

 clientProfile = ClientProfile()
 clientProfile.httpProfile = httpProfile
 client = monitor_client.MonitorClient(cred, "city1", clientProfile)

 req = models.GetMonitorDataRequest()
 params = {
 "Namespace": "qce/cvm",
 "MetricName": "cpu_usage",
 "Period": 60,
 "StartTime": "2024-10-17T15:10:42+08:00",
 "EndTime": "2024-10-17T16:10:42+08:00",
 "Dimensions": ['''{"unInstanceId": "ins-miltleuc"}''']
 }
 req.from_json_string(json.dumps(params))
 resp = client.GetMonitorData(req)

 print(resp.to_json_string(indent=2))

except CloudSDKException as err:
 print(err)
```

> Note:
> Lines 14 to 15: secretId and secretKey are calling credentials used for calling authentication, which means that through secretId and secretKey, the called AppID and UIN can be identified. secretId and secretKey are created in the following ways:

Line 19: Endpoint is the calling domain name. For how to obtain it, see 3. Request Domain.

Line 23: "city1" is the region where the monitoring data of cloud product instances is located. For how to obtain it, see 4. Request Region.

9. Appendix

| Namespace | Metric Name | Time Granularity | Dimension Group | Dimension Group Example |
|---|---|---|---|---|
| qce/cvm | cpu_usage | 60/300/3600/86400 | unInstanceId | "unInstanceId":"ins-096makkh" |
| | cpu_loadavg | 60/300/3600/86400 | unInstanceId | "unInstanceId":"ins-096makkh" |
| | CPULoadAvg5m | 60/300/3600/86400 | unInstanceId | "unInstanceId":"ins-096makkh" |
| | CPULoadAvg15m | 60/300/3600/86400 | unInstanceId | "unInstanceId":"ins-096makkh" |
| | base_cpu_usage | 60/300/3600/86400 | unInstanceId | "unInstanceId":"ins-096makkh" |
| | gpu_mem_total | 60/300/3600/86400 | unInstanceId | "unInstanceId":"ins-096makkh" |
| | GpuMemUsage | 60/300/3600/86400 | unInstanceId | "unInstanceId":"ins-096makkh" |
| | gpu_mem_used | 60/300/3600/86400 | unInstanceId | "unInstanceId":"ins-096makkh" |
| | GpuPowDraw | 60/300/3600/86400 | unInstanceId | "unInstanceId":"ins-096makkh" |
| | GpuPowLimit | 60/300/3600/86400 | unInstanceId | "unInstanceId":"ins-096makkh" |
| | GpuPowUsage | 60/300/3600/86400 | unInstanceId | "unInstanceId":"ins-096makkh" |
| | GpuTemp | 60/300/3600/86400 | unInstanceId | "unInstanceId":"ins-096makkh" |
| | GpuUtil | 60/300/3600/86400 | unInstanceId | "unInstanceId":"ins-096makkh" |
| | lan_outtraffic | 60/300/3600/86400 | unInstanceId | "unInstanceId":"ins-096makkh" |
| | lan_intraffic | 60/300/3600/86400 | unInstanceId | "unInstanceId":"ins-096makkh" |
| | lan_outpkg | 60/300/3600/86400 | unInstanceId | "unInstanceId":"ins-096makkh" |
| | lan_inpkg | 60/300/3600/86400 | unInstanceId | "unInstanceId":"ins-096makkh" |
| | tcp_curr_estab | 60/300/3600/86400 | unInstanceId | "unInstanceId":"ins-096makkh" |
| | mem_used | 60/300/3600/86400 | unInstanceId | "unInstanceId":"ins-096makkh" |
| | mem_usage | 60/300/3600/86400 | unInstanceId | "unInstanceId":"ins-096makkh" |
| qce/ lb_private | outtraffic | 60/300/3600/86400 | vip | "vip":"111.111.111.11" |
| | intraffic | 60/300/3600/86400 | vip | "vip":"111.111.111.11" |
| | outpkg | 60/300/3600/86400 | vip | "vip":"111.111.111.11" |
| | inpkg | 60/300/3600/86400 | vip | "vip":"111.111.111.11" |
| | connum | 60/300/3600/86400 | vip | "vip":"111.111.111.11" |

| Namespace | Metric Name | Time Granularity | Dimension Group | Dimension Group Example |
|---|---|---|---|---|
| | new_conn | 60/300/3600/86400 | vip | "vip":"111.111.111.11" |

# Shipping Monitoring Metrics to Kafka

## I. Feature Description

Cloud Monitor data can be forwarded to a specified Kafka, supporting:

- Full data forwarding, such as forwarding all metric data of CVMs.
- Partial forwarding. For example, metric data of a CVM instance or metric data of all CVM instances under a specific AppID is forwarded, or specified metric data of specified instances is forwarded.
  The above features are configured through forwarding rules.

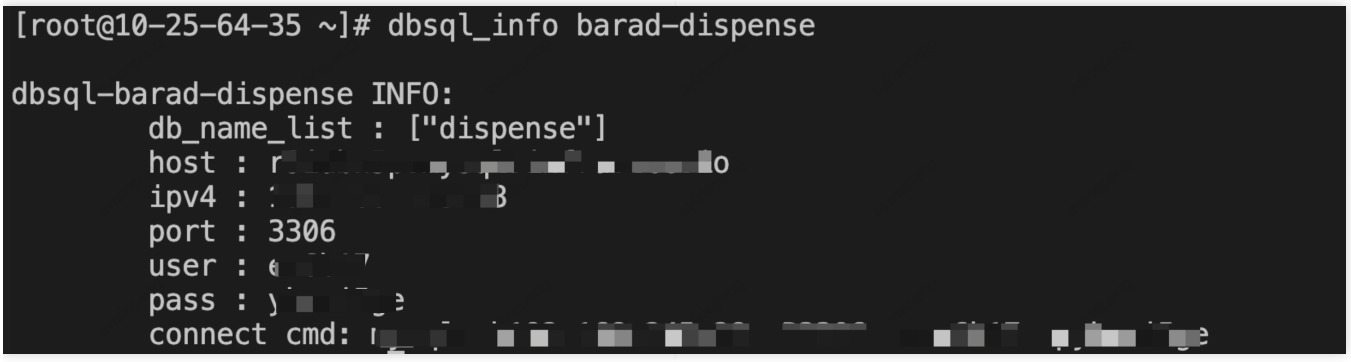
> Note:
>
> The tcloud-barad-dispense application must be deployed in the environment.

## II. Forwarding Configuration

Currently, forwarding rules need to be configured via CLI-based operations, that is, by inserting forwarding rules into the database. The steps are as follows:

1. Connect to the forwarding database: Set the database name to dispense and the corresponding service_id to dbsql-barad-dispense , and run the following command on the master node:

```
dbsql_info barad-dispense        // This is barad-dispense, not dbsql-barad-dispense.
```




> Note:
>
> The connect cmd is the command to connect to the dispense database.

2. Configure forwarding rules.

```
-- Create a rule.
insert into t_rule (RuleName, Status) values ('forwarding rule name', 1);
// Note:
// RuleName: rule name.
// Status: whether the rule is enabled. 1: enabled, 0: disabled.

-- Configure the metrics that need to be forwarded.
insert into t_metric (RuleId, Metric, Namespace, Measurement, Periods) values (10, 'cpu_usage', 'qce/cvm', 'cv
m_device', '60');
// Note:
// RuleId: ID of the new rule specified by t_rule.
// Metric: metric name.
// Namespace: Namespace to which the metric belongs.
// Measurement: view of Cloud Monitor. To obtain the view, refer to Appendix 1.
// Periods: time granularity of the metric. Multiple granularities can be specified, such as (60, 300), or the aste
risk (*) can be used to indicate all supported granularities.
// If you need to know the namespace to which the metrics belong, the measurement, as well as the support
ed time granularities, refer to the appendix.

-- Configure filter conditions (Optional. Configure this if you only want to forward data of a specific instance.
There are 10 tables in total. You can obtain the information about which table ${n} resides on by performing
a ruleId % 10 calculation.)
insert into t_condition_${n} (RuleId, Filters) values (10, '[{"Key":"vm_uuid","Values":["c7fc0b4c-0059-4318-bb0
c-dc4a31a071da"],"Expression":"="},{"Key":"__appid__","Values":["251005942", "1"],"Expression":"in"}]');
// Note:
// RuleId: ID of the new rule specified by t_rule.
// Filters: forwarding rule filter, which is of a JSON array with an and relationship between rules.

-- Configure the forwarding target Kafka.
insert into t_target (RuleId, Target, ProtoType) values (10, '{"brokers":"11.150.215.105:9092","topic":"Nowrit
e","user":"","password":""}', 1);
// Note:
// RuleId: ID of the new rule specified by t_rule.
// Target: target Kafka address.
// ProtoType: message types forwarded to Kafka. 1: JSON, 2: Protobuf.
```

# III. Message Protocol

The message protocol forwarded to Kafka is as follows:

```
// json
{
    "Namespace": "qce/cvm",
    "MetricName": "cpu_usage",
    "Statistic": "MAX",
```

```
    "Period": 60,
    "Dimensions": [{
        "Name": "appid",
        "Value": "1"
    }, {
        "Name": "projectid",
        "Value": "0"
    }, {
        "Name": "vm_uuid",
        "Value": "4f7639f1-f156-41b4-9072-f7e27af97d3b"
    }],
    "Timestamp": 1737107700000,
    "Value": 22.916666666666668
}
```

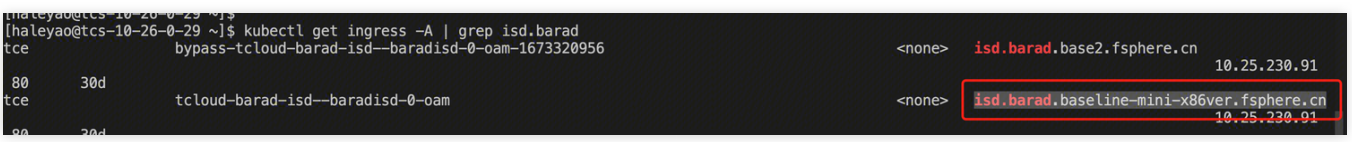

# IV. Forwarding Table Structure Description

| Table Name | Meaning | Description |
| --- | --- | --- |
| t_rule | Forwarding a rule table | Includes rule metadata such as rule ID, rule name, and configuration user.<br>`Id: 38230`<br>`RuleName: 1118-Phase 2-cvm-cpu // Forwarding rule name.`<br>`AppId: 1300260700 // User AppID for configuring rules, can be empty.`<br>`SubUin: 0 // User UIN for configuring rules, can be empty.`<br>`Status: 1 // Rule enabling status, 1: enabled; 0: disabled.`<br>`CreateTime: 2024-11-20 11:14:23`<br>`UpdateTime: 2024-11-20 11:14:23`<br>`IsAllMetric: 0 // Can be empty.` |
| t_metric | Forwarding a metric table | Information about the metrics to be forwarded.<br>Note:<br>If forwarding is performed within a private network, `ExtNamespace` and `ExtMetric` are empty.<br><br>` Id: 3732457`<br>` RuleId: 38230`<br>` ExtMetric: CpuUtil // External metric name, can be empty.`<br>`ExtNamespace: QCE/TI_MODEL // External namespace, can be empty.`<br>` Measurement: model_svc // View name.`<br>` Periods: * // * indicates all granularities. If you only want to forward granularities 60 and 300, configure it as 60,300.`<br>` Metric: cpu_util // Internal metric name.`<br>` Namespace: qce/ti_model // Internal namespace.`<br>` ENMeaning: cpu_util // Metric English name, which can be empty.`<br>` CName: `<br>` Unit: % // Metric unit, which can be empty.` |

| | | |
|---|---|---|
| | | ` CreateTime: 2024-12-12 18:30:00`<br>` UpdateTime: 2024-12-12 18:30:00` |
| t_condition_${n} | Forwarding a condition table | • There are 10 tables (numbered 0 to 9). The specific table for forwarding conditions can be determined by ruleId % 10.<br>• When the current table is empty, it indicates that unconditional forwarding is configured in the private network.<br>• The Expression in Filters can only be set to `=` or `in`. When it is set to `=`, only a single value is contained in Values. When it is set to `in`, multiple values may be contained in Values.<br><br>`Id: 12`<br>`RuleId: 38230`<br>`Filters: [{"Key":"vm_uuid","Values":["c7fc0b4c-0059-4318-bb0c-dc4a31a071da"],"Expression":"="},{"Key":"__appid__","Values":["251005942", "1"],"Expression":"in"}]`<br>`CreateTime: 2024-11-19 17:49:19`<br>`UpdateTime: 2024-11-19 17:49:19` |
| t_target | Forwarding a target table | • TargetType must be set to `Kafka`.<br>• Valid values of ProtoType:<br>• 1: `JSON`<br>• 2: `Protobuf`<br><br>` Id: 34539`<br>` RuleId: 38230`<br>`TargetType: Kafka`<br>` Target: {"brokers":"11.150.215.105:9092","topic":"Nowrite","user":"","password":""}`<br>` ProtoType: 1`<br>`CreateTime: 2024-11-19 16:48:31`<br>`UpdateTime: 2024-11-19 16:48:31` |

# 5. Appendix: Viewing Metrics, Namespace, Measurement, and Period

1. Run kubectl get ingress -A | grep isd.barad on the master node.



2. Access the domain name through a browser.

# Best Practices
# Application of Instance Grouping in Alarms

This document introduces how to implement unified settings of alarm policies in batch for cross-instance and cross-region projects of the same product according to workload classification.

1. Overview
   As your business grows, the number of cloud resources increases, making efficient daily monitoring a challenge for Ops. With the instance grouping feature in Cloud Monitor, you can classify a large batch of cloud resources by workload. This allows you to apply unified alarm policies to all instances within a group, helping you stay informed about their usage and operational status, and significantly improving Ops efficiency.

- Use cases
  Suppose you have five CVMs, three of which (CVM-1, CVM-2, and CVM-3) are used for Game A, while the remaining two are assigned to other workloads.
  - CVM-1 is located in Region A and assigned to Project A.
  - CVM-2 is located in Region B and assigned to Project B.
  - CVM-3 is located in Region C and assigned to Project C.
    To promptly learn about the usage and operation status of the three CVMs for Game A, the following configurations are required.

  i. Create an instance group. Add CVM-1, CVM-2, and CVM-3 to the same instance group.
  ii. Create an alarm policy, which includes metrics such as CPU utilization, memory utilization, disk utilization, and outbound bandwidth of the public network (for monitoring traffic usage). Alarm notifications are sent automatically when metrics of any instance reach a certain threshold.

2. Grouping instances
   i. Log in to the Cloud Platform console and choose Cloud Monitor > Instance Group.
   ii. Click Create and configure the instance group in the pop-up window as follows.
   - Group name: The name is "Game A".
   - Group type: Choose Cloud Virtual Machine > Basic Monitoring.
   - Add to the group: Select regions City A, City B, and City C respectively, and add CVM-1, CVM-2, and CVM-3 from each region to the instance group.

   iii. After the configuration is completed, click Save.
3. Creating alarms
   i. Log in to the Cloud Platform console and choose Cloud Monitor > Alarm Configuration >

Alarm Policy.

ii. Click Create, and perform configurations according to the prompts on the page.

- Policy name: Cutomized by you.
- Policy type: Choose Cloud Virtual Machine > Basic Monitoring.
- Alarm object: Choose Instance Group > Game Business A.
- Alarm trigger conditions.

  CPU utilization, statistical period 1 minute, >, 85%, lasts for 1 data point, and alarm once per day.

  Memory utilization, statistical period 1 minute, >, 85%, lasts for 1 data point, and alarm once per day.

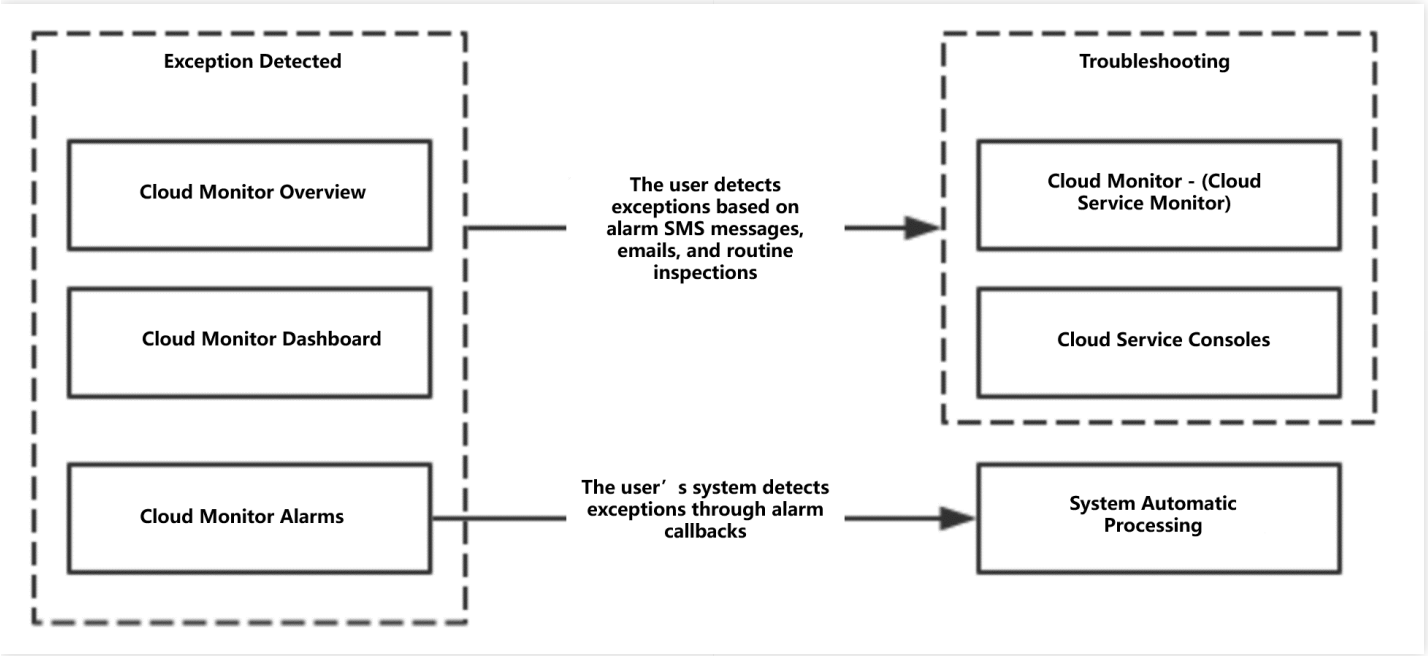
  Disk utilization, statistical period 1 minute, >, 85%, lasts for 1 data point, and alarm once per day.

  Outbound bandwidth, statistical period 1 minute, >, 100Mbps, lasts for 1 data point, and alarm once per day.
- Notification template: The system default is set to preset notification template. You can also click Select Template to choose an existing one you've created, or click Create New Template to create one based on your needs.

iii. After the configuration is completed, click Complete.

# Best Practices for Monitoring Scenarios

Cloud Monitor provides multiple methods to help users identify resource exceptions and send exception information to users through various channels as quickly as possible.




1. Locating exceptions
    i. Detecting exceptions via monitoring and alarms
    Monitoring and alarms enable the cloud platform to detect exceptions promptly and send proactive notifications, allowing users to identify exceptions passively. This ensures timely awareness of exceptions under all circumstances. Users can log in to the Cloud Platform console and go to the Cloud Monitor console to configure appropriate alarm policies for the resources they care about. For details, see Creating an Alarm Policy.
    Key performance metrics and events configured as alarm rules will trigger multiple notification methods to promptly reach users and their systems in the event of an exception.
    Alarm policies configured with designated recipients or recipient groups will notify users promptly via SMS, email, and other channels. Features such as repeated alarms and alarm convergence are supported to help ensure critical alarms are not missed while minimizing unnecessary disturbances.
    Users can also configure the callback API feature in alarm notifications to deliver exception alarm information to their systems for further aggregation and processing.
    ii. Detecting exceptions via the monitoring dashboard
    Locating exceptions via the monitoring dashboard is a user-driven approach that relies on analyzing average performance trends and historical data. It requires users to proactively identify exceptions. For exceptions that are not covered by alarm policies or are difficult to

detect through existing alarm policies, they can still be identified during daily inspections. Compared with alarms, the monitoring dashboard enables users to assess the overall impact scope of resource exceptions. By subscribing key resources to the dashboard and configuring charts appropriately, users can highlight exception information under various scenarios. For details, see Create New Dashboard.

For individual instances, users can subscribe to detailed instance views to easily compare performance trends on the dashboard panel.

For resource clusters, users can subscribe to aggregated data within the same cluster to conveniently view the overall monitoring dashboard on the dashboard panel and compare it with the trend of individual instances within the cluster. For details, see Monitoring Scenarios for Large Volumes.

Exceptions identified through the monitoring dashboard can be further analyzed using the dashboard's sorting and ranking features, enabling users to locate the affected resources and assess the scope of impact for in-depth troubleshooting.

2. Exception troubleshooting

   Locating exception objects via the monitoring overview page

   During daily inspections or upon receiving alarm notifications, users can log in to the Cloud Platform console and choose Cloud Monitor > Monitoring Overview.

   i. Check the Service Status in Last 24 Hours module to view exception statuses of resources across different regions.

      Use the Exception Information Overview feature to get a preliminary view of recent exceptions.

   ii. Click the number of exception objects to be redirected to the specific exception product menu page under Cloud Product Monitoring.

      The specific product list page under Cloud Product Monitoring will automatically filter and display the exception resource objects for the user.

   iii. Click the ID of a specific object to be redirected to its monitoring details page, which provides historical status rollback to assist in detailed exception troubleshooting.

   - The exception timeline provides users with access to both current and historical information about the exception object. By reviewing historical alarms and status changes, users can be assisted in troubleshooting the current exception.
   - The monitoring data for resource performance provides users with the most comprehensive resource performance data. It allows for week-over-week and day-over-day comparisons of current and historical data for the same metric and facilitates troubleshooting by comparing data changes in different metrics over the same time period.

3. Locating exception objects via the dashboard

   Log in to the Cloud Platform console and choose Cloud Monitor > Dashboard > Dashboard List to go to the corresponding dashboard page.

   i. When an abnormal trend appears in the monitoring chart, click Expand the Instance List, and then a sorted list of corresponding instances will be displayed below the curved line graph.

Through the list, the specific objects that are causing the exceptions can be located.

ii. By clicking the object name in the sorted list, users can be redirected to the monitoring details page of the object, where historical status information is provided to assist users in locating exceptions.

- The exception timeline provides users with access to both current and historical information of the exception object. By reviewing past alarms and status changes, users can be assisted in troubleshooting the current exception.
- The monitoring data for resource performance provides users with the most comprehensive resource performance data. It allows for week-over-week and day-over-day comparisons of current and historical data for the same metric and facilitates troubleshooting by comparing data changes in different metrics over the same time period.

# Configuring CVM Metric Alarms

Scenarios

The following example illustrates how to configure an alarm: Suppose you want to receive an SMS alarm at the phone number 12345678888 when the CPU utilization of CVM instance ins-12345678 (located in Region A) exceeds 80% for two consecutive five-minute periods.

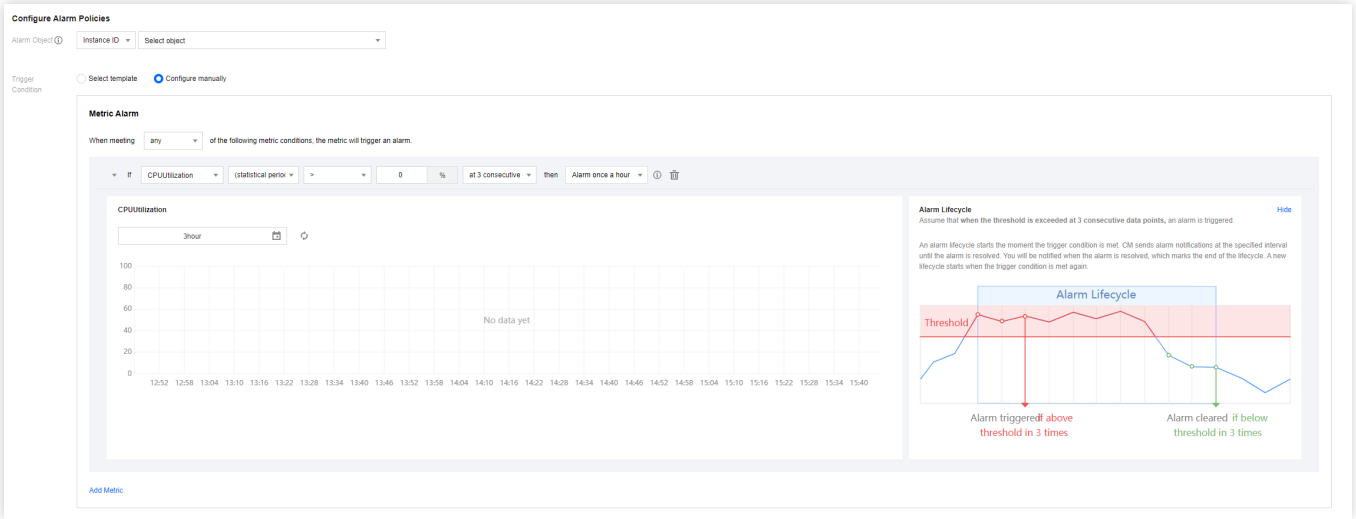
Operation Steps

1. Log in to the Cloud Platform console and choose Cloud Monitor > Alarm Configuration > Alarm Policy.
2. Click Create and configure the following options.
3. Configure the basic information.

- Policy name: CPU alarm.
- Policy type: Choose Cloud Virtual Machine > Basic Monitoring.

4. Configure the alarm object. Under the Alarm Object module, select instance ID, and then choose the CVM instance from the drop-down list on the right.
5. Configure trigger conditions. Under the Trigger Conditions module, configure the following conditions.

- Select Manual Configuration.
- In the Metric Alarm module, select CPU utilization, statistical granularity of 5 minutes, >, 80%, for 2 consecutive data points, and trigger an alarm every 15 minutes.




6. Configure the alarm notification template. Click Create New Template, enter a template name, select the recipient with the phone number 12345678888, and choose SMS as the notification channel. After completion, click Confirm.

7. Click Complete to complete the alarm configuration.

8. At this point, when the instance's CPU utilization exceeds 80% for two consecutive 5-minute periods, an SMS alarm will be sent by Cloud Monitor to the phone number 12345678888.

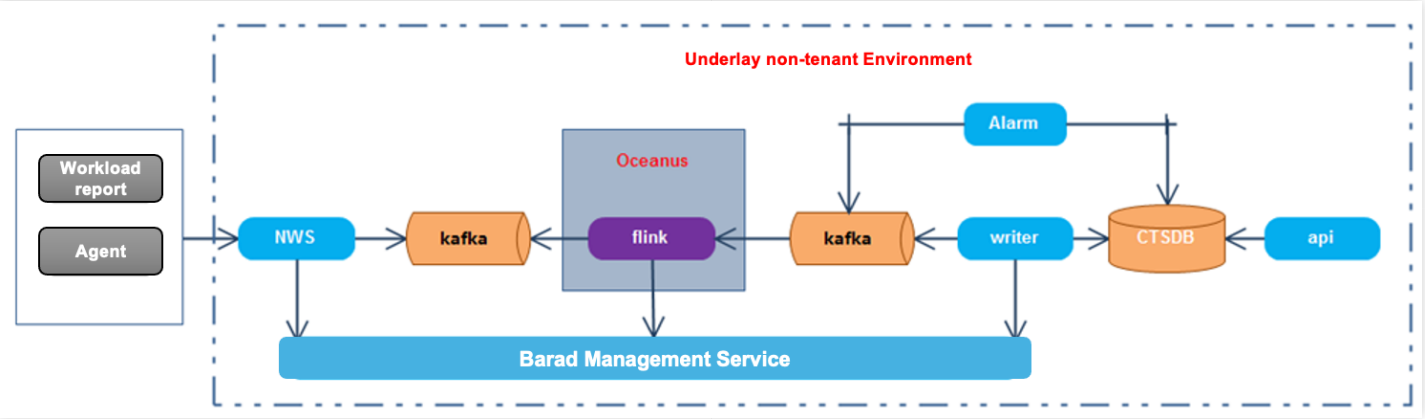
# Ops Management Guide
# Product Architecture
# Core Feature Components

| Component ID | Component Feature | Purpose |
| --- | --- | --- |
| es-barad | Data storage | Cloud Monitor CTSDB. |
| tcloud-barad-update2 | agent | The Stargate server provides services for the Stargate agent (ps aux \| grep sgagent) on the CVM, mainly used to record and update CVM status information as well as distribute BARAD agents. |
| tcloud-barad-script | Data reporting | Contains 6 bypass services:<br>1. barad_sync_server automatically synchronizes CVM information from CCDB to the translation table for NWS to perform dimension translation.<br>1. cm_php_script is used to synchronize alarm policy information of the primary database to the regional database.<br>1. dict_access, a server for reporting dimension translation information for other products, caches data into TDMQ for Kafka. The Topic list is the table of tDictionary_* in the StormDictionary library. It is recommended to create only two partitions for each topic. Otherwise, the container will continuously restart because of too many processes.<br>1. dict_sync consumes dimension translation information from TDMQ for Kafka, and then stores it in the database (to form a complete service together with dict_access).<br>1. The role of getAppTraffic is to obtain traffic data of the view under qce/lb in each region, aggregate and calculate the total inbound/outbound traffic of each tenant to qce/lb_total, and display it in the traffic monitoring column on the tenant side.<br>1. DROP |
| tcloud-barad-query-api | Data query | Query results from ES storage data via API. |
| tcloud-barad-nws-tke | Data reporting | Receives data reported by TKE. |
| tcloud-barad-nws | Data reporting | Receives the reported data. |

| Component ID | Component Feature | Purpose |
| --- | --- | --- |
| tcloud-barad-monitor-dashboard | Data query | Dashboard query. |
| tcloud-barad-manager-api | Data query | A middle platform that manages API services. |
| tcloud-barad-isd | Data query | Contains multiple backend management services and scripts, overlay instance agent report timeout detection service (to be removed soon), and Cloud Monitor configuration center configuration pull service. Generally, multiple instances are not required.<br>Other containers generally have ConfClient components to cache the configuration of the BARAD configuration center. |
| tcloud-barad-event | Data query | Receives and processes events. It is required to send alarm notifications to AMP. |
| tcloud-barad-config-center | Data query | Configures services for the monitoring data middle platform. |
| tcloud-barad-ctsdb-manager | Data query | Reads the access configuration in the database and creates a metric template in CTSDB. |
| tcloud-barad-api-go | Data query | Cloud API. |
| tcloud-barad-api | Data query | API. |
| tcloud-barad-alarm-amp | Data alarm | Unified alarm sending. |
| tcloud-barad-alarm-consumption | Data alarm | Alarm history query. |
| tcloud-barad-alarm-detector | Data alarm | Alarm detection. |
| tcloud-barad-alarm-policy | Data alarm | CRUD operations for alarm policies. |
| tcloud-barad-alarm-synchronizer | Data alarm | Alarm synchronization component. |

| Component ID | Component Feature | Purpose |
| --- | --- | --- |
| tcloud-barad-skywalker | Flink task | Contains Flink stream computation logic components and their management scripts. |
| tcloud-barad-metric-writer | Full | Consumes data from Kafka and writes it to ES. |
| tcloud-oceanus-ctcw1 | Flink task | Executes tasks and submits them to the yarn cluster. |
| tcloud-oceanus-galileo1 | Flink task | The image is used to provide external exposure interfaces to create job tasks. |
| tcloud-oceanus-servant1 | Flink task | Real-time data sampling for syntax checking and jar package detection. |

# Product Architecture Diagram

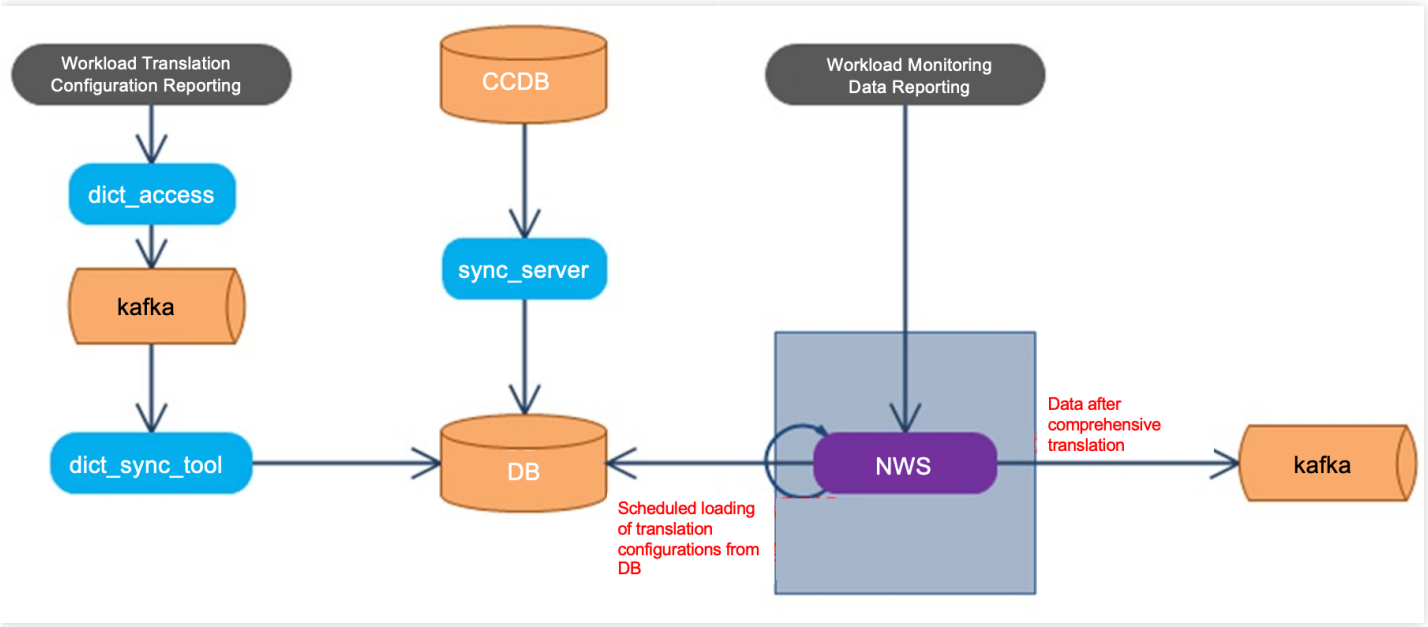



Main module division:

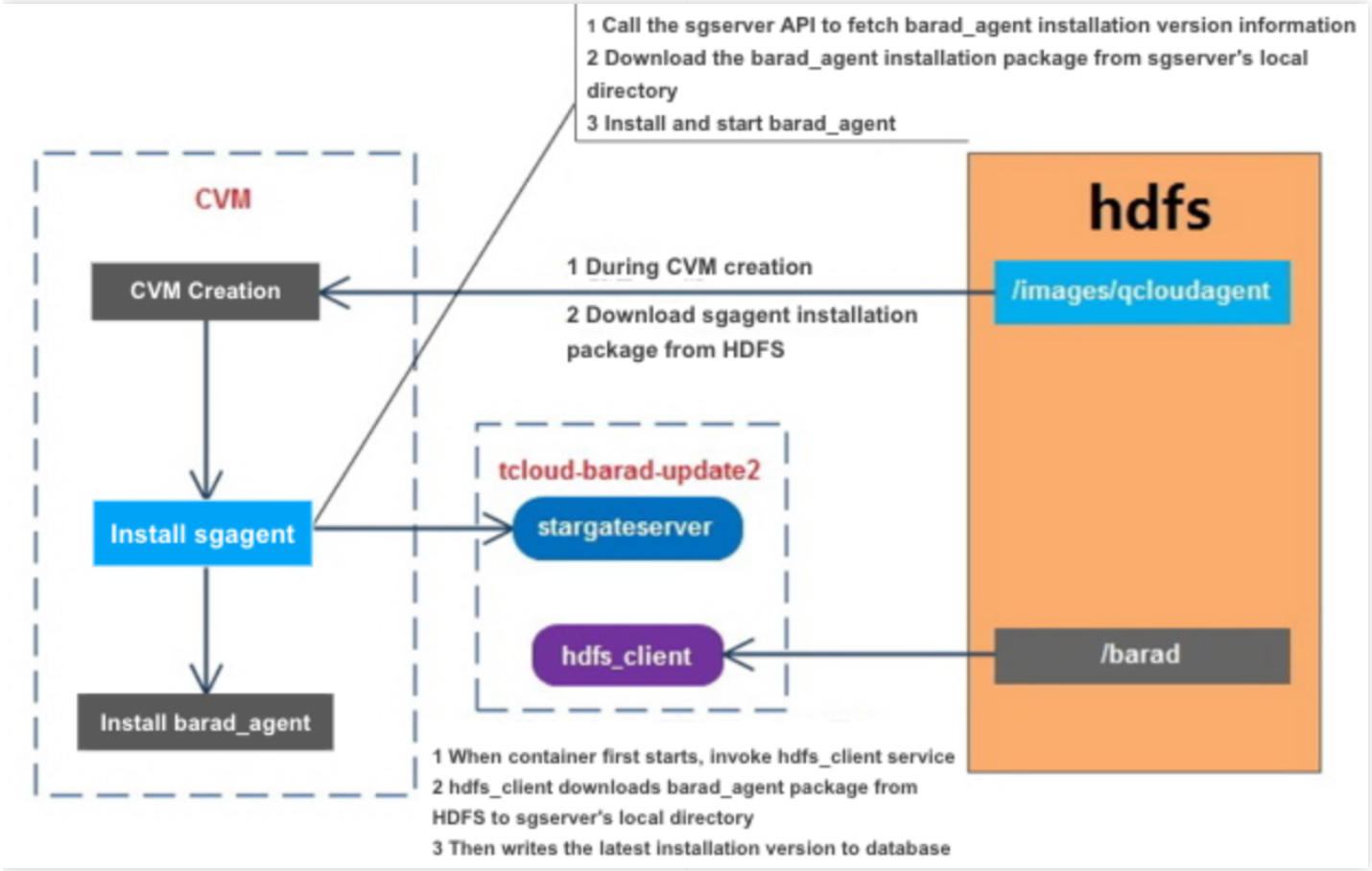
| Module | Description |
| --- | --- |
| Reporting part | The collection and reporting are mainly completed by the workload reporting module, dcos, as well as the Agents deployed on the instances and the host. Both the instances and the host have access paths to the Cloud Monitor access server. All reports are sent to the unified domain name and CGI provided by the Cloud Monitor. The HTTP protocol is used for reporting. |
| Data processing | Message buffering is provided by the stream processing cluster built by Flink, front access server-NWS, and Kafka. |
| Data storage | CTSDB is used for storage and is maintained and upgraded by a specialized team. |
| Exception handling module | Includes an alarm center and an event center, which respectively handle alarms with thresholds and durations after statistical calculations, as well as simple triggering alarms. |
| Message sending module | Primarily responsible for integrating with the backend platform message sending platform, recording alarm streams, and translating some required fields in alarm displays. |
| Custom message module | A module that provides the tenant side with the feature of directly sending messages. It does not perform convergence processing. |
| Api | It is divided into two layers: customapi and baradapi; the former directly connects to Cloud API Gateway and can be called by the tenant side; the latter directly connects to storage, blocks all storage-related features from the frontend, and provides some metric configuration management features. |

| Module | Description |
| --- | --- |
| Self-service access system | It is used on the operation side. It allows Ops personnel to dynamically add and delete dimension aggregation and statistical configurations of metrics. These configurations will affect the metric features for viewing and configuring alarms on the tenant side and the operation side. |
| Other bypass systems | Provides some configuration synchronization features. More supplementary information will be given in the section about the module's detailed architecture later. |

Access layer NWS service




barad_Agent installation process




1. If the user chooses to enable Cloud Monitor when creating a CVM.
2. After the CVM is successfully created, it will download the latest version of the Stargate installation package from the /images/qcloudagent directory of HDFS (generally the stargate_linux_install_v1.2.6.tar.gz version).
3. Install and start the sgagent service.
4. The container (tcloud-barad-update2) where sgserver is located will execute the hdfs_client service command when it is first started.
5. The hdfd_client will download the latest version of the barad_agent installation package from the /barad directory of HDFS (generally /barad/baradagent_installer), place it in the local /data/www/update2.agent.yun.com/update directory, and write the version information to the database.
6. After sgagent starts, it will call the sgserver API to query the installed version information of barad_agent.
7. Download the barad_agent installation package from the /data/www/update2.agent.yun.com/update directory of sgserver.
8. Install and start the barad_agent service.
9. In the whole process:
    - The barad_agent installation package is provided by the CVM team, which ensures that the latest installation package information is uploaded to the /barad directory of HDFS.
    - The barad_agent installation package and configuration are maintained by the CVM team.

sgagent is merely responsible for download, installation, and start.

# Deployment Architecture Diagram

Cloud Monitor involves many components, all of which feature containerized deployment. Its services are all global and regional services and none are physical services. The AZ level is not involved.
The global components are marked here:
Global:
DB:
dbsql-barad-manager
dbsql-barad-master-alert_management_platform_new
dbsql-barad-master-barad_assistant
dbsql-barad-master-CloudAlarmV2
dbsql-barad-master-CloudUser
dbsql-barad-master-ConfigCenter
dbsql-barad-master-CustomAlarm
dbsql-barad-master-CustomMonitorAlarm
dbsql-barad-master-dev-barad_assistant
dbsql-barad-master-dev-CloudAlarmV2
dbsql-barad-master-dev-StormCloudConf
dbsql-barad-master-domain_policy
dbsql-barad-master-monitor_dashboard
dbsql-barad-master-StormCloudConf
dbsql-barad-master-web
dbsql-oceanus-netcenter
dbsql-oceanus-online_clusteradmin
dbsql-oceanus-online_clustermaster
dbsql-oceanus-online_galileo
dbsql-oceanus-online_watchdog
dbsql-oceanus-online_taskcenter

Pod:
tcloud-barad-event

Frontend, CAM, and Cloud API:
product-frontend-imgcache-tcloud-monitor
product-frontend-imgcache-tcloud-monitor-alarm
product-frontend-imgcache-tcloud-monitor-clb
product-frontend-imgcache-tcloud-monitor-dashboard
product-frontend-imgcache-tcloud-monitor-overview
product-frontend-imgcache-tcloud-monitor-sdk

product-frontend-imgcache-tcloud-monitor-sdk2

preset-barad-cam-ocloud-tbarad

preset-barad-cam-tcloud-monitor

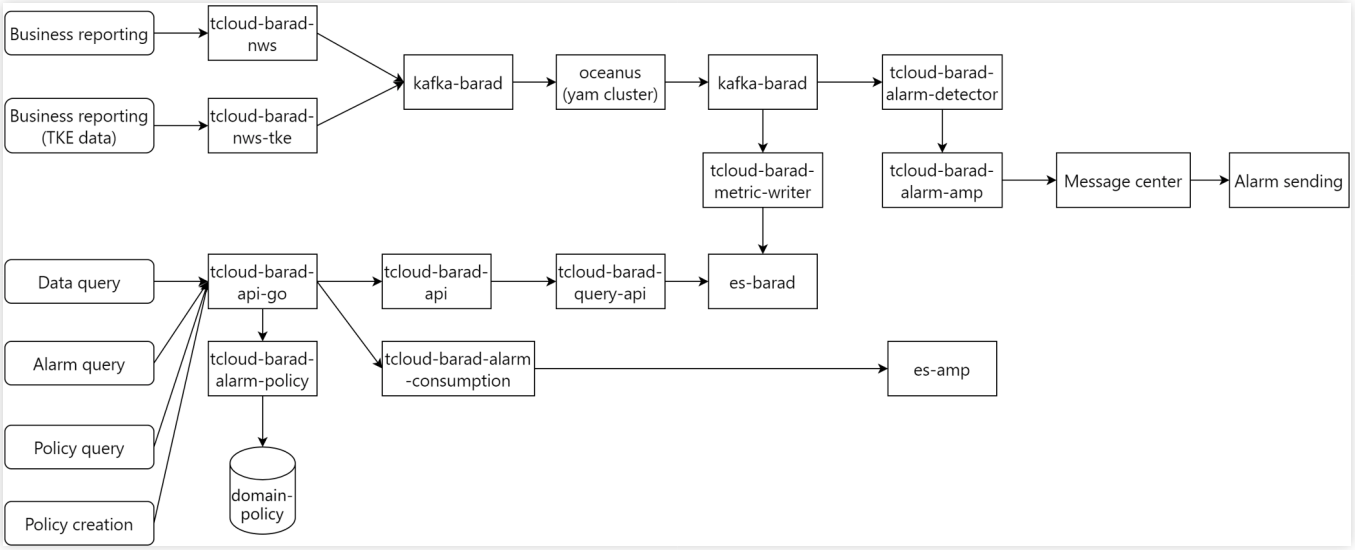
preset-barad-message-svr-tcloud-BARAD

yunapi-barad

# Business Flow Diagram

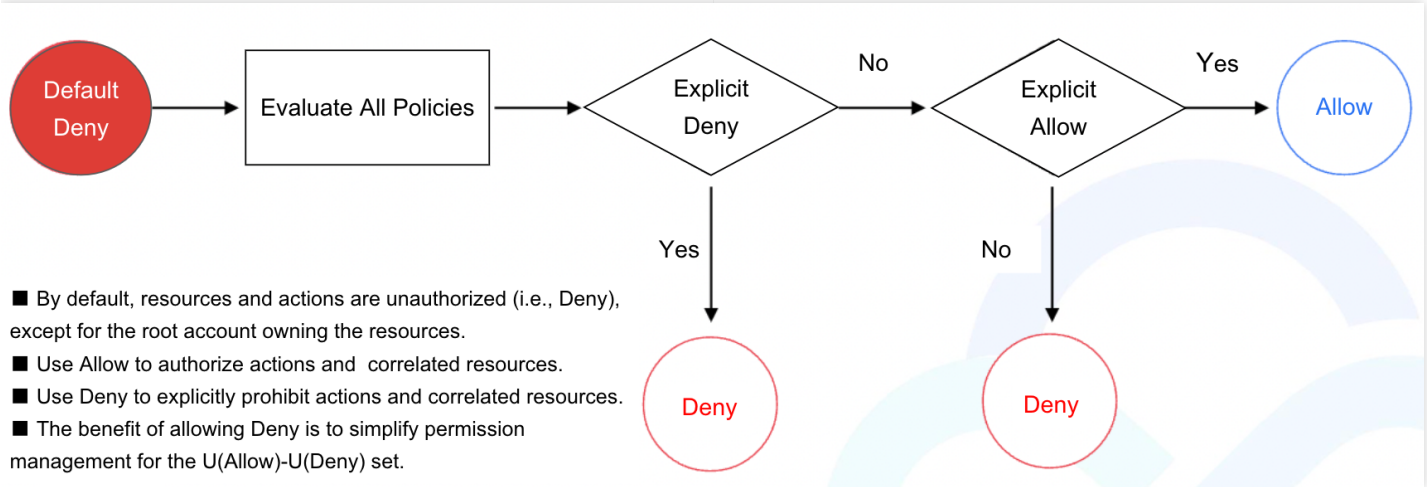
The monitoring system is built through Kafka, ES, and Oceanus (Finlk):

- Kafka is used to report queue buffers and write to ES data buffers.
- Oceanus (Finlk) is used to report data computing and processing.
- ES is used for the on-site storage of post-processing monitoring data.
- tcloud-barad-XXX, a monitoring microservice, features containerized deployment. All data will eventually be stored in ES.
- tcloud-barad-alarm-XXX is responsible for monitoring alarm microservices. Its main features are alarm policy maintenance, alarm check, and alarm history maintenance.



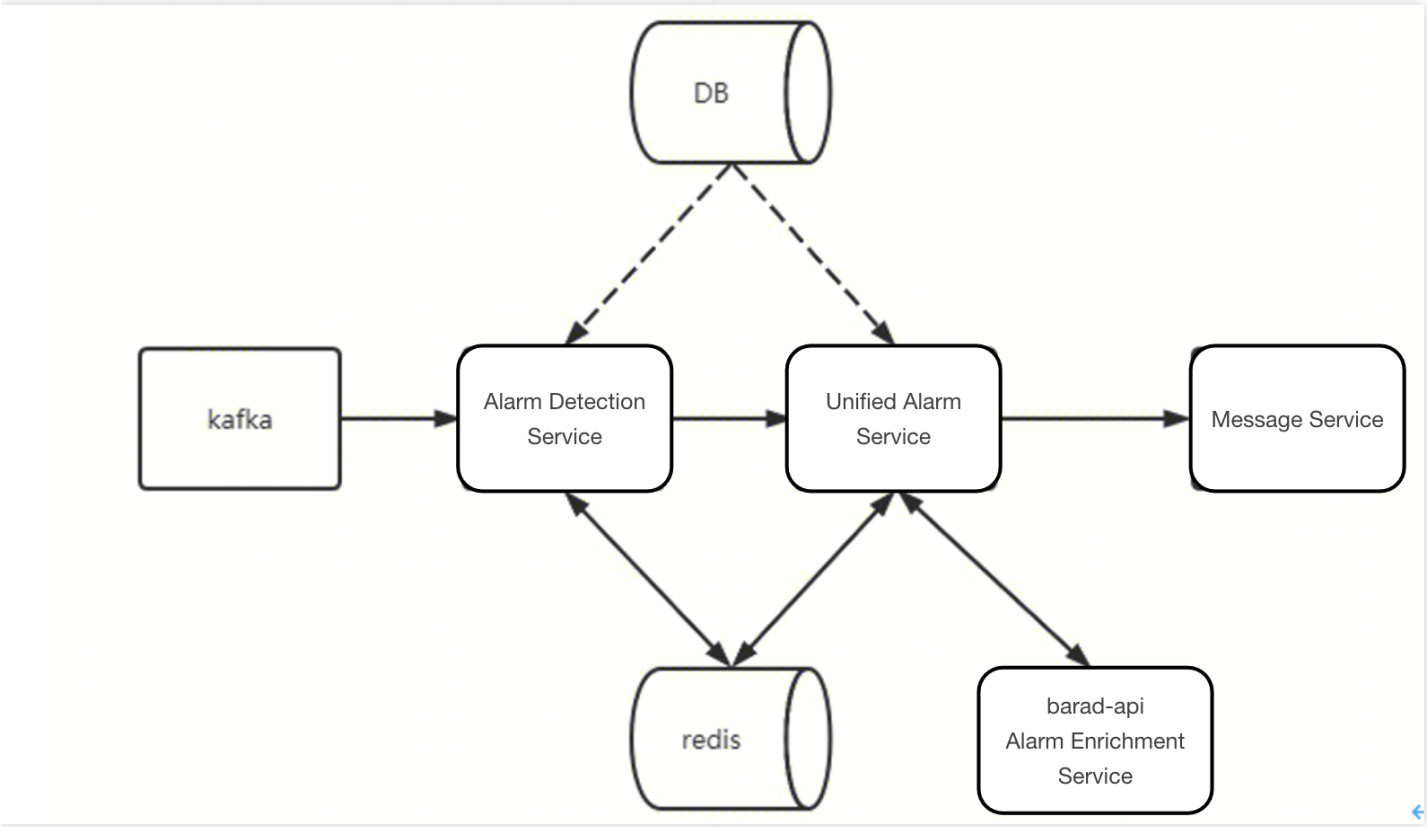
# Core Logic Overview

## Authentication logic




■ By default, resources and actions are unauthorized (i.e., Deny), except for the root account owning the resources.
■ Use Allow to authorize actions and correlated resources.
■ Use Deny to explicitly prohibit actions and correlated resources.
■ The benefit of allowing Deny is to simplify permission management for the U(Allow)-U(Deny) set.

## Alarm linkage



# Product Dependencies

| Dependent Product Name | Dependency Type | Dependency Relationship Description | Impact Level | Remarks |
|---|---|---|---|---|
| Supporting TDSQL | Database | dbsql-barad-AlarmManager -- ["AlarmManager"]<br>dbsql-barad-barad_assistant -- ["barad_assistant"]<br>dbsql-barad-Barad_DV -- ["Barad_DV"]<br>dbsql-barad-Barad_TgwData -- ["Barad_TgwData"]<br>dbsql-barad-BaradNwsStat -- ["BaradNwsStat"]<br>dbsql-barad-ClientKey -- ["ClientKey"]<br>dbsql-barad-config-center -- ["config-center"]<br>dbsql-barad-DayData -- ["DayData"]<br>dbsql-barad-IdcTraffic -- ["IdcTraffic"]<br>dbsql-barad-LogMonitor -- ["LogMonitor"]<br>dbsql-barad-manager -- ["manager"]<br>dbsql-barad-master-alert_management_platform_new -- ["alert_management_platform_new"]<br>dbsql-barad-master-barad_assistant -- ["barad_assistant"]<br>dbsql-barad-master-CloudAlarmV2 -- ["CloudAlarmV2"]<br>dbsql-barad-master-CloudUser -- ["CloudUser"]<br>dbsql-barad-master-ConfigCenter -- ["ConfigCenter"]<br>dbsql-barad-master-CustomAlarm -- ["CustomAlarm"]<br>dbsql-barad-master-CustomMonitorAlarm -- ["CustomMonitorAlarm"]<br>dbsql-barad-master-dev-barad_assistant -- ["barad_assistant"]<br>dbsql-barad-master-dev-CloudAlarmV2 -- ["CloudAlarmV2"]<br>dbsql-barad-master-dev-StormCloudConf -- ["StormCloudConf"] | High | For the detailed design of the database, see Product Architecture Design. |

| Dependent Product Name | Dependency Type | Dependency Relationship Description | Impact Level | Remarks |
|---|---|---|---|---|
| | | dbsql-barad-master-domain_policy -- ["domain_policy"]<br>dbsql-barad-master-monitor_dashboard -- ["monitor_dashboard"]<br>dbsql-barad-master-StormCloudConf -- ["StormCloudConf"]<br>dbsql-barad-master-web -- ["web"]<br>dbsql-barad-QComponentMonitor -- ["QComponentMonitor"]<br>dbsql-barad-QComponentMonitor_dErrorDetail -- ["QComponentMonitor_dErrorDetail"]<br>dbsql-barad-StargateUpdater -- ["StargateUpdater"]<br>dbsql-barad-StormCloudConf -- ["StormCloudConf"]<br>dbsql-barad-StormDataStat -- ["StormDataStat"]<br>dbsql-barad-StormDictionary -- [""]<br>dbsql-oceanus-netcenter -- ["netcenter"]<br>dbsql-oceanus-online_clusteradmin -- ["online_clusteradmin"]<br>dbsql-oceanus-online_clustermaster -- ["online_clustermaster"]<br>dbsql-oceanus-online_galileo -- ["online_galileo"]<br>dbsql-oceanus-online_watchdog -- ["online_watchdog"]<br>dbsql-oceanus-online_taskcenter -- ["online_taskcenter"] | | |
| Supporting Kafka | Cloud message queue | Monitoring data processing<br>Related topics<br>- Barad_Cdb<br>- Barad_Comm<br>- Barad_Device<br>- Barad_Hb<br>- Barad_Host<br>- Barad_Mongo<br>- Barad_Nat<br>- Barad_Ping | High | |

| Dependent Product Name | Dependency Type | Dependency Relationship Description | Impact Level | Remarks |
|---|---|---|---|---|
| | | - Barad_Rlb<br>- Barad_sdn<br>- Barad_tgw_1<br>- barad_shipper<br>- cm_event<br>- BaradUpdateConf<br>- SelfMonitor<br>- UpdateConf<br>- NWS_CAT1<br>- cm_platform<br>- configManagementToBarad<br>- FlinkToWriter_Barad_Comm<br>- taw_flink_to_writer<br>- taw_collector_metric<br>- taw_runtime_metric<br>- taw_collector_tracing<br>Monitoring data translation<br>Related topics<br>- tDictionary_1<br>- tDictionary_2<br>- tDictionary_4<br>- tDictionary_vsr<br>- tDictionary_bmtgw<br>- tDictionary_bmtgwfee<br>- tDictionary_bmtgwisp<br>- tDictionary_vsg<br>- tDictionary_cross_region<br>- tDictionary_bmlbtype<br>- tDictionary_tmp<br>- tDictionary_vipinfo<br>- tDictionary_cbs_cell<br>- tDictionary_bms<br>- tDictionary_bms2<br>- tDictionary_lb_rs_unhealth<br>- tDictionary_lb_rs_unhealth_vpcid | | |
| Supporting ES | Monitoring data and alarm record storage | es-amp<br>es-barad | High | |
| Supporting Redis® | Alarm record | ckv-barad | High | |

| Dependent Product Name | Dependency Type | Dependency Relationship Description | Impact Level | Remarks |
| --- | --- | --- | --- | --- |
| Supporting HDFS | Agent storage | stargate_agent storage | High | |
| Supporting yarn | Monitoring data of Flink processing of the running environment of the task | Monitoring data processing | High | |

# Troubleshooting
# Impact Scope of Failures

| No. | Scenario | Impact on the Cloud Product Data Plane | Impact on the Platform Control Plane |
|---|---|---|---|
| 1 | Flink task exception | Monitoring data breakpoints or data interruptions | No |
| 2 | ES exception | Monitoring data breakpoints or data interruptions | No |

# Failure Recovery
# Scenario 1: Flink Task Exception Impacting Monitoring Data Continuity

## Scenario Description of Failure Symptoms

The operator task of the Flink task stops or the entire task is interrupted. At this point, the monitoring data breakpoints or interruptions may occur. The topic consumer group corresponding to Kafka has a data consumption backlog (Barad_Comm).

## Impact Scope of the Failure

The monitoring data cannot be queried, and the alarm on the cloud product side cannot be triggered normally.
No impact on the platform's other cloud product services.

## Failure Location and Analysis

Enter the supporting yarn deployment node to check whether the corresponding yarn cluster and Flink tasks are running normally.
The following are common operation instructions:

```
##yarn resource display
yarn top

##yarn instance display
yarn application --list
yarn application -list -appStates ALL

##yarn task execution details
yarn application -status <application ID>

##yarn rm status
yarn rmadmin -getAllServiceState

##yarn node display
yarn node -list
```

Scenario 1: If the basic yarn cluster environment is abnormal, you need to contact our supporting queue colleagues for handling.
Scenario 2: Use df -h to check the disk utilization of the yarn cluster. If the disk utilization of the yarn machine service exceeds 80%, it will start cleaning up the services in the cluster. (Due to the mixed deployment of machines and HDFS, there is TDSQL cold backup data on HDFS. For similar issues, you can clean up a small amount of corresponding cold backup files first and then start monitoring Flink tasks. Meanwhile, contact our supporting TDSQL queue colleagues to handle the retention time of cold backup files.)
Scenario 3: The cluster is normal. If the Flink task stops due to the connection of the monitoring component, handle it according to the emergency steps. Just restart the task.

# Failure Emergency Response Steps

Steps to restart Flink:

1. Restart the oceanus-ctcw1 container (to prevent the connection exception of the component):

   ```
   tcloud-oceanus-ctcw1
   kubectl get po -A  |grep tcloud-oceanus-ctcw1 | awk '{print $2}' | xargs kubectl delete po -n tce
   ```

2. Check the status of job tasks (5 represents stopping tasks, 3 represents starting, and 4 represents normal operation):

   ```
   mysql22001
   ```

select Status from online_galileo.Job;

If it is 4, execute stop-job.sh first. Do not execute run-job.sh until the Status field of the database is 5. If it is 5, execute run-job.sh.

3. Enter the tcloud-barad-skywalker container:

   cd /usr/local/services/barad-skywalker/job/

4. Stop the Flink task:

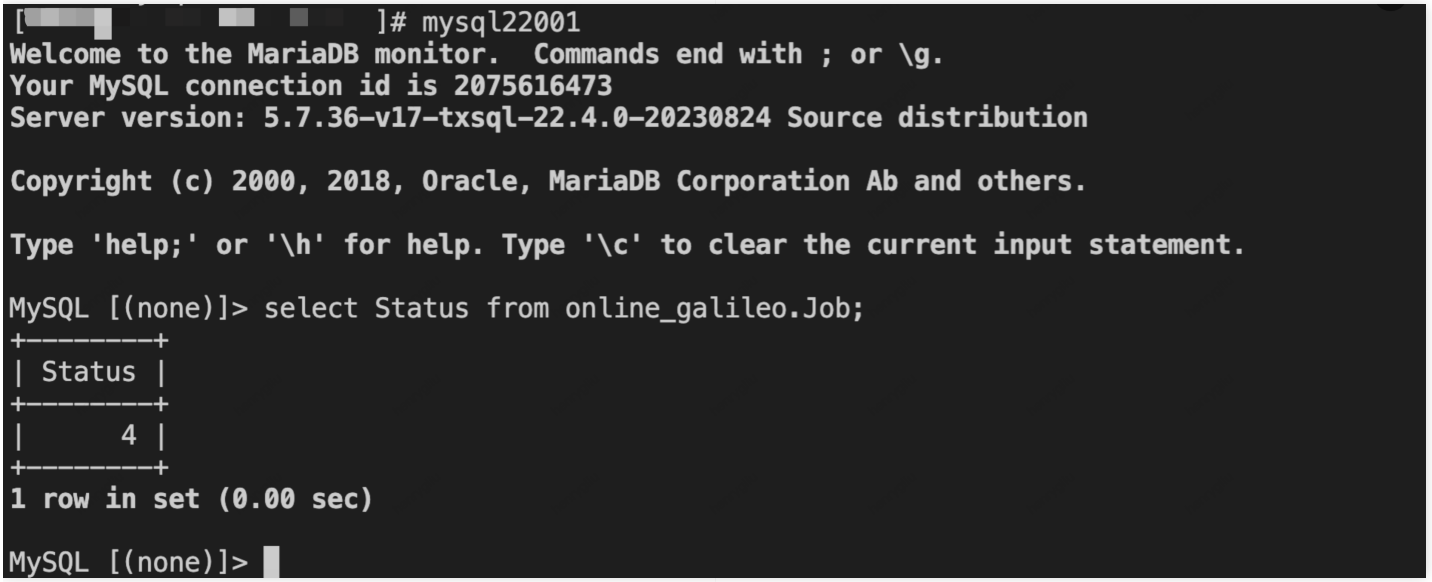
   bash stop-job.sh

5. Start the Flink task:

   bash run-job.sh

# Failure Recovery Verification

Check the status of job tasks (5 represents stopping tasks, 3 represents starting, and 4 represents normal operation):
mysql22001
select Status from online_galileo.Job;



Wait for 2-3 minutes, then check whether the monitoring metrics of the tenant-side CVM have recovered to normal.

# Scenario 2: ES Exception Impacting Monitoring Data Continuity

## Scenario Description of Failure Symptoms

ES write exception. At this point, the monitoring data breakpoints or interruptions occur, and the topic consumer group corresponding to Kafka has a consumption backlog (FlinkToWriter_Barad_Comm).

## Impact Scope of the Failure

The monitoring data cannot be queried, and the alarm on the cloud product side cannot be triggered normally.
No impact on the platform's other cloud product services.

## Failure Location and Analysis

Check the container logs written to ES:
tcloud-barad-metric-writer
Enter the container and run tail -1000f /usr/local/services/metric-writer/log/metric-writer.log | grep bulk.
Observe here whether the exception content returned in the log indicates that the ES cluster is under great pressure or the ES service prohibits writing (read-only).

## Failure Emergency Response Steps

Scenario 1: If the ES cluster is under great pressure, you can appropriately clean up part of the monitoring data and then contact our supporting ES queue colleagues for ES cluster scale-out processing.
Scenario 2: If ES is read-only, you can appropriately clean up part of the monitoring data to recover the writable status, and then contact our supporting ES queue colleagues for ES cluster scale-out processing.

Common checks and cleanup operations for ES:

```
##Sort by index size in the current cluster.
curl  -s -H "Content-Type: application/json" -u $es_user:$es_pass $es_ip:$es_port/_cat/indices?v\&s=st
ore.size:desc | more

##Selectively clean up by index size.
```

```
curl  -s -XDELETE -H "Content-Type: application/json" -u $es_user:$es_pass $es_ip:$es_port/xxx

##The cluster recovers to the writable status.
curl -X PUT  -H "Content-Type: application/json" -u $es_user:$es_pass $es_ip:$es_port/_cluster/settings?pretty  -d '{"persistent":{"cluster.blocks.read_only_allow_delete":null}}'

curl -X PUT  -H "Content-Type: application/json" -u $es_user:$es_pass $es_ip:$es_port/_cluster/settings?pretty -d '{"index.blocks.read_only_allow_delete": false}'
```

# Failure Recovery Verification

Check the container logs written to ES:

tcloud-barad-metric-writer

Enter the container and run tail -1000f /usr/local/services/metric-writer/log/metric-writer.log | grep bulk.

Check whether the newly written log reports an error.

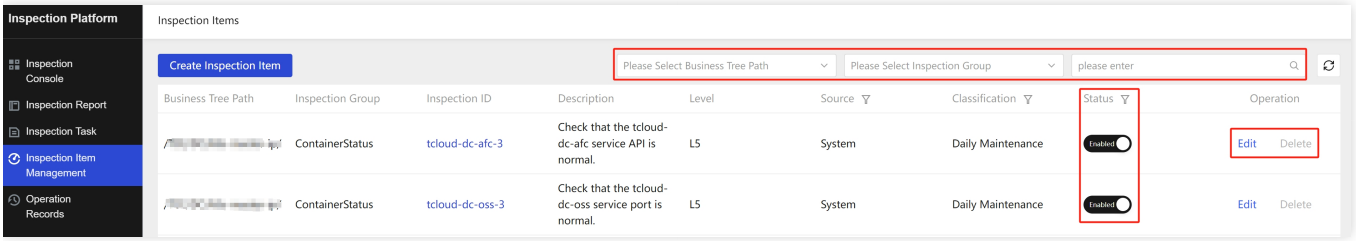
Wait for 2-3 minutes, then check whether the monitoring metrics of the tenant-side CVM have recovered to normal.

# Daily Inspection
# Cloud Product Inspection
# Viewing Inspection Items

Navigate to Inspection Platform > Inspection Item Management, where inspection items are already built-in for cloud products, and you can use these inspection items to check the status of cloud products.

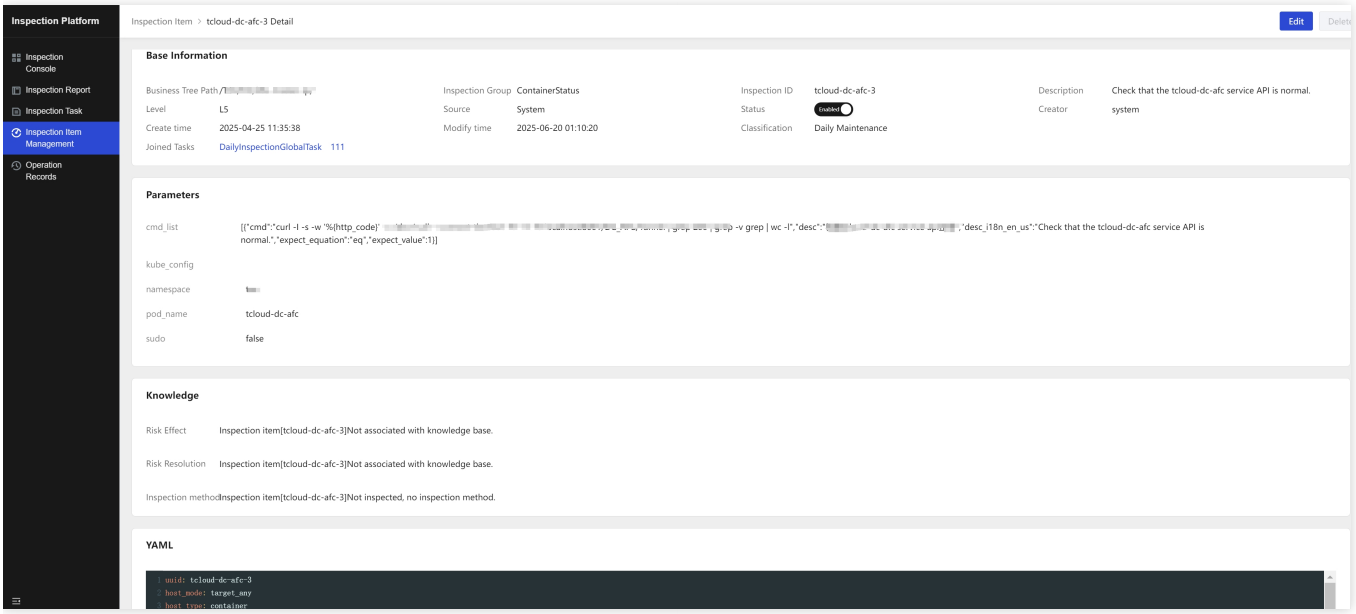
## 1 Inspection Item List

This module summarizes the list of inspection items on the current platform, allowing you to enable/disable, edit, add, and delete inspection items. By using the dropdown list to select the business tree path or inspection item group, you can filter the inspection items. Additionally, you can search for inspection items using keywords such as the inspection item description or inspection item group.



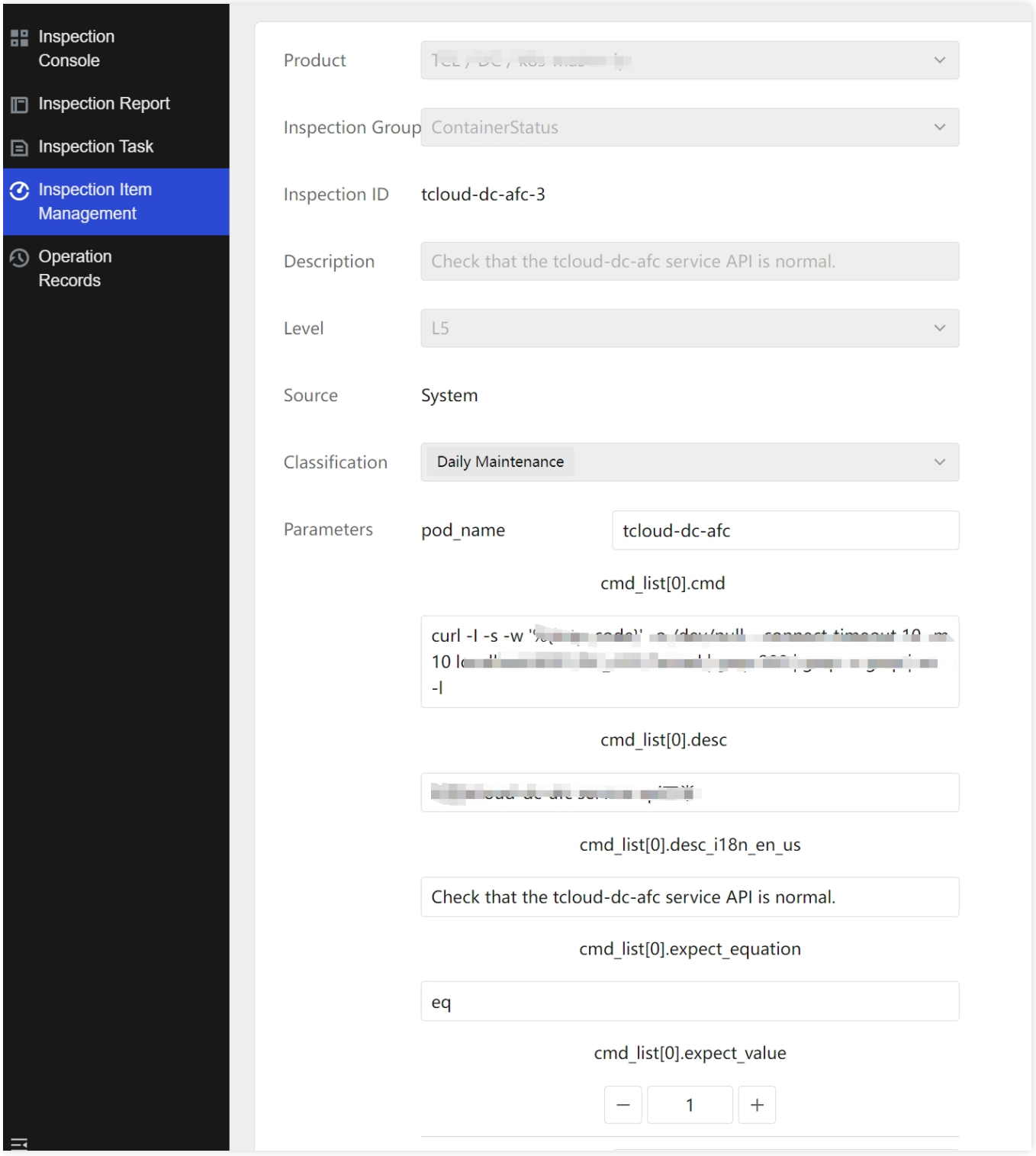

## 2 View Inspection Item Detailed Information

In the inspection item list, click the Inspection Item ID you want to view to enter the inspection item details page. The page displays inspection item information, including basic information, parameters, knowledge base, and YAML.




# 3 Edit Inspection Items

On the inspection item management page, click Edit to perform this operation.

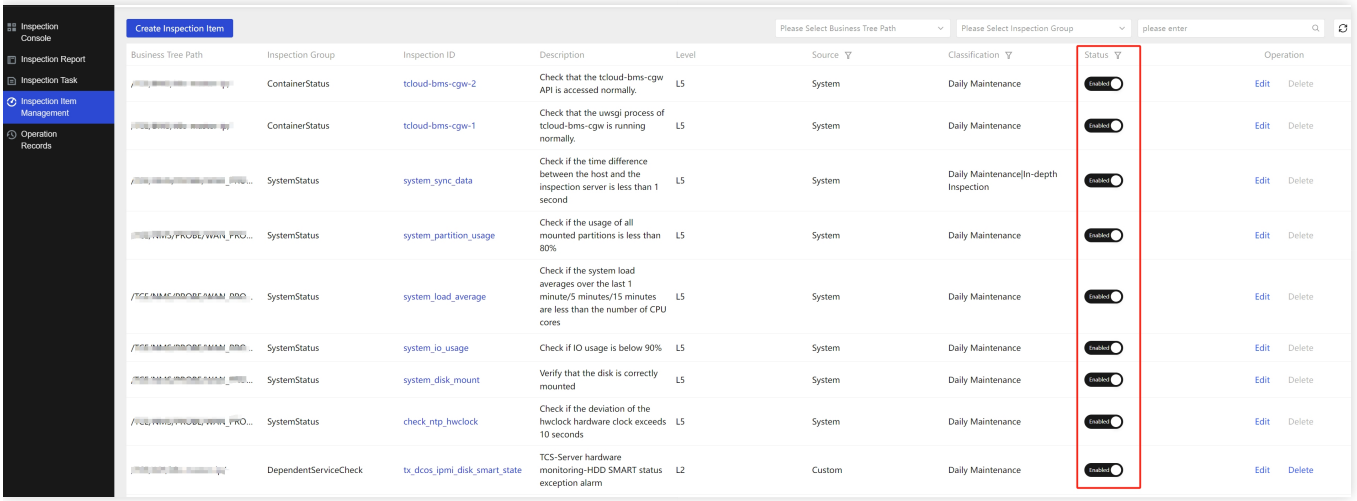



# 4 Enable/Disable Inspection Items

Enabling: Click the button switch in the disabled status to enable an inspection item that has been previously disabled. Once enabled, the inspection item will be executed during the next run of associated inspection tasks it has been added to.

Disabling: Click the button switch in the enabled status to disable an inspection task that has been previously

enabled. Once disabled, the inspection item will not be executed during the next run of associated inspection tasks it has been added to, and related abnormal items will be automatically cleared.

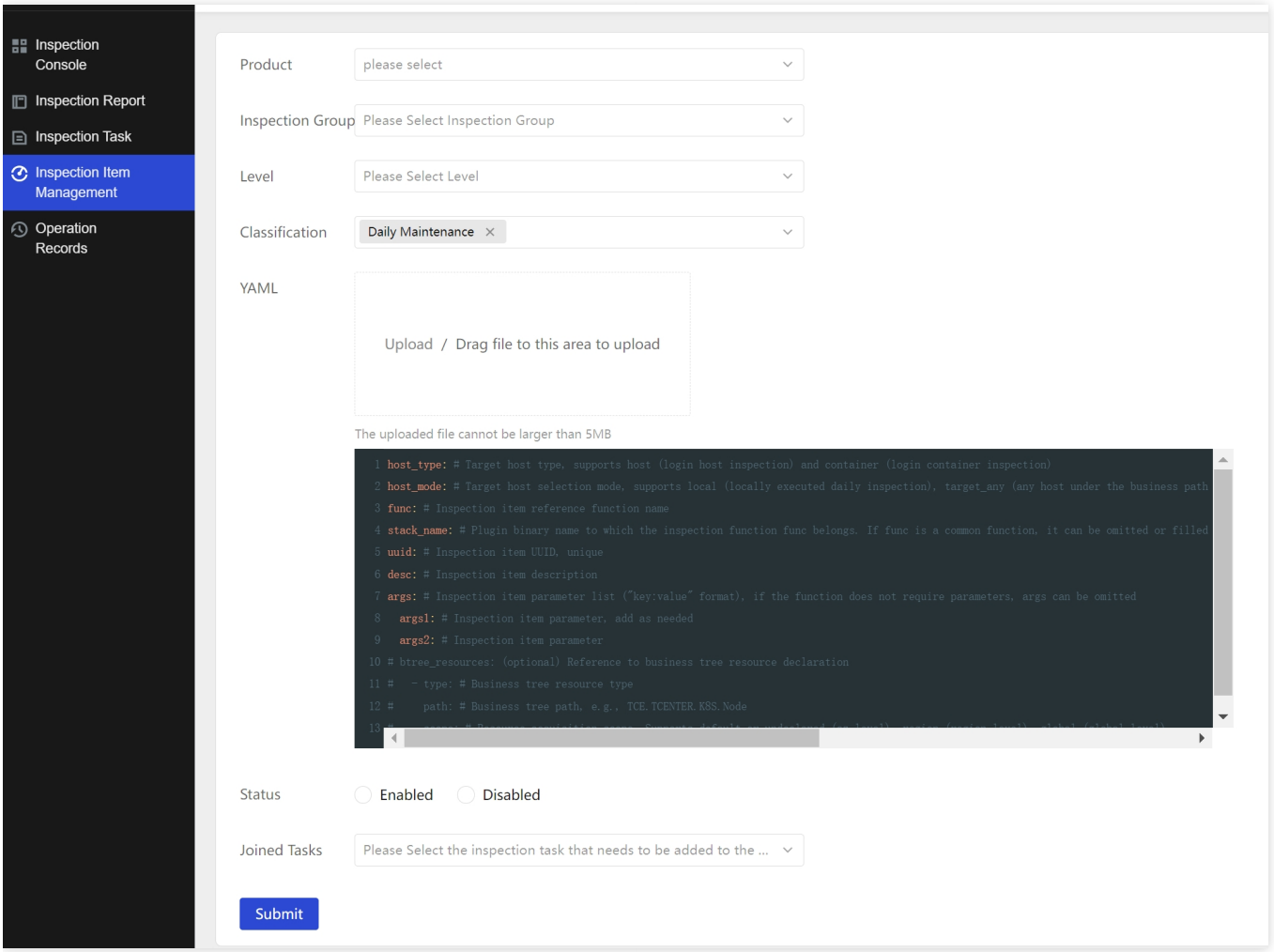



# 5 Add Custom Inspection Items

Automated management of inspection items is supported. Below is the guide for creating a custom inspection item.

Starting Adding Custom Inspection Items

Navigate to Inspection Platform > Inspection Item Management, and click Create Inspection Item.




The process for writing inspection items is as follows:

1. Inspection item product: Select the inspection target for the current inspection item.
2. Inspection item group: Select the group name for the inspection item.
3. Inspection item level: Select the inspection item level.
4. Category: Select the category (the default is Daily Ops).
5. Configuring the YAML File
    i. Find the appropriate inspection plugin and inspection function based on the inspection method.
        i. For remote command execution, typically use the system plugin's system_exec_command function.
        ii. For SQL execution, typically use the MySQL plugin's check_field_value function.
        iii. ……
    ii. Declare parameters for the inspection function.
    iii. Declare the description information and UUID (try to use meaningful English names for the UUID).
6. Enable inspection items and add them to the inspection task.

Configure the YAML file.

(1) Basic format

The YAML file contains the following fields:

host_type: host  # Target host type. It can be set to host (for logging in to a host for inspections) or container (for logging in to a container for inspections). When the current path is k8s-master-ip or tianxun, host_type s hould be set to container.
host_mode: target_all  # Selection mode for target hosts. It supports local (execute locally), target_any (select any host in the workload path of the current inspection item for execution), and target_all (select all hosts in the workload path of the current inspection item for execution).
func: MysqlExec  # Name of the function referenced by the inspection item.
stack_name: checks-base  # Binary name of the plugin of the inspection function func. If func is a general fun ction, this field can be omitted or filled in as checks-base. If it is a technology stack-specific function, the bin ary name of the technology stack should be filled in, such as tce-compute.
uuid: tx_cvm_podcheck_1 # (Required) UUID of the inspection item.
desc: Inspection item description # (Required) Description of the inspection item.
args: # Inspection item parameter list ("key: value" format). If func does not require any parameters, args can be omitted.
  database: yhimage
  sql: show tables;
# btree_resources: (Optional) Declaration for reference of the workload tree resources.
#  - type: host  # Type of the workload tree resources.
#    path: TCE.TCENTER.K8S.Node  # Path of the workload tree.
#    scope: region  # Resource acquisition scope. It supports the default value or undeclared (for AZ level), re gion (for region level), and global (for global level).

(2) Reference to workload tree resources (btree_resources)

host_type: host
host_mode: local
func: ExecuteShResultNum
btree_resources: # Here is the declaration for the host resources within the current region in the path TCE.TC ENTER.K8S.Node for the workload tree.
  - type: host
    path: TCE.TCENTER.K8S.Node
    scope: region
args:
  sh_list:
    - sh_name: "tgwadm_whitelist_check.sh"
      params: '{{ .btree_resources.host | map(.ip) | join(" ") }}' # Here, .btree_resources.host is used to referenc e the declared resource. The syntax is in jq format.
      expect_equation: eq
      expect_value: 1

(3) Common function formats

- Execute the shell command and compare the output result with the expected number.

  stack_name: checks-base
  func: SystemCmdResultNum
  args:
    cmd_list:

```
    cmd: "uptime | awk '{print $3/365}'"
    desc: "Whether the server has been running for more than 3 years."
    expect_equation: 'le'
    expect_value: 3
```

- Execute the shell command and compare the output result with the expected string.

```
stack_name: checks-base
func: SystemCmdResultStr
args:
  cmd_list:
    cmd: kubectl get secret -n cert-manager cert-manager-webhook-tls -o json | jq '.data["tls.crt"]' | x
args echo | base64 -d | openssl x509  -checkend 3888000
    desc: "Check whether the certificate expiration time for cert-manager is 45 days later."
    expect_equation: 'eq'
    expect_value: 'Certificate will not expire'
```

- Check whether the pod is alive.

```
stack_name: checks-base
func: PodActiveCheck
args:
  pod_name: 'tcloud-argus2-adp-synchronizer'
```

- Check whether the pod is alive, and execute shell commands inside the pod.

```
stack_name: checks-base
func: PodActiveCheck
args:
  cmd_list:
    - cmd: ps aux | grep 'gw-controller-monitor' | grep -v grep | awk '{print $11}' |sort | wc -l
      desc: Check whether the ocloud-vpc-nfv-gw-controller gw-controller-monitor process is running
normally.
      expect_equation: ge
      expect_value: 1
  pod_name: ocloud-vpc-nfv-gw-controller
```

- Copy the shell script and execute it, then check the execution results.

```
stack_name: checks-base
func: ExecuteShResultNum
args:
  sh_list:
    - sh_name: "tgwadm_whitelist_check.sh"
```

```
params: '"ip1" "ip2"'
expect_equation: eq
expect_value: 1
```

(4) Complete function list

- checks-base

```
CheckOsVersion
ClusterAllocationCheck
CpuFrequency
CpuUsage
CrontabGrep
DiskCapacity
DiskMount
DiskUsage
ElasticsearchCheck
EtcdCheck
ExecuteShResultNum
ExecuteShResultNumProxy
ExecuteShResultStr
ExecuteShResultStrProxy
GetK8sCert
GetOcloudCert
GetServerCert
GetTcloudCert
HttpApiResult
HttpApiResultNum
HttpApiResultStr
IOUsage
LoadAverage
MemSwapUsage
MemUsage
MysqlCheck
MysqlCheckDemo
MysqlCheckFieldValue
MysqlExec
PartitionUsage
PingCheck
PodActiveCheck
PodImageCheck
PodLogCheck
PodNodeCheck
PodReplicaCheck
PodRestartCheck
PodUsageCheck
PortCheck
ProcessCheck
```

PvcUsageCheck
RedisConfigCheck
RedisExecCheck
ServiceActiveCheck
SyncDate
SysLogCheck
SystemCmdResult
SystemCmdResultNum
SystemCmdResultNumProxy
SystemCmdResultStr
SystemCmdResultStrProxy
SystemModeCheck
ZookeeperCheck

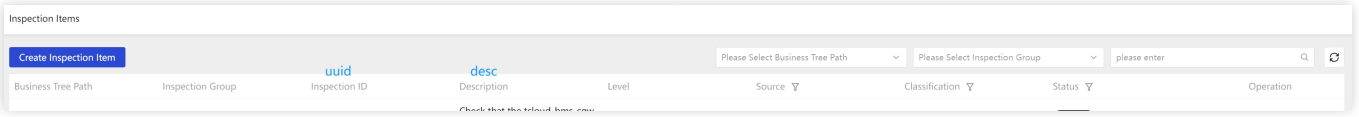
(5) Examples

Example 1: tx_tcenter_check_cert-manager_expire.

```
uuid: tx_tcenter_check_cert-manager_expire
host_mode: target_all
host_type: host
stack_name: tce-base
func: SystemCmdResultStr
args:
   cmd_list:
      - cmd: kubectl get secret -n cert-manager cert-manager-webhook-tls -o json | jq '.data["tls.crt"]' | xargs
echo | base64 -d | openssl x509  -checkend 3888000
        desc: Check whether the certificate expiration time for cert-manager is 45 days later.
        expect_equation: eq
        expect_value: Certificate will not expire
desc: Check whether the certificate expiration time for cert-manager is 45 days later.
btree_resources: []
```

Take the above inspection item as an example:

- This inspection item's UUID is set to tx_tcenter_check_cert-manager_expire, and its desc is set to 'Check whether the certificate expiration time for cert-manager is 45 days later'. The UUID and desc will be displayed on the inspection list page.




- This inspection item's host_mode is set to target_all, indicating that this inspection item will ultimately be executed on all machines in the associated business tree path. If it were set to target_any, this inspection item would be executed on any machine in the associated business tree path.
- This inspection item's host_type is set to host, indicating that the target of this inspection item is a host. In addition to host, the inspection platform also supports a container type for host_type, indicating that the

target is a container or special node.

- This inspection item's stack_name is set to tce-base, and its func is set to SystemCmdResultStr. This indicates that the implementation of the inspection is located in a binary file named tce-base, and the function being called is named SystemCmdResultStr.
- The subsequent args field contains parameters passed to the SystemCmdResultStr function. Similar to section 2.2, the SystemCmdResultStr function supports executing multiple shell commands, with each command specified using cmd. The expected output for these commands is 'Certificate will not expire'.

Example 2: tx_cbs_pod_processor

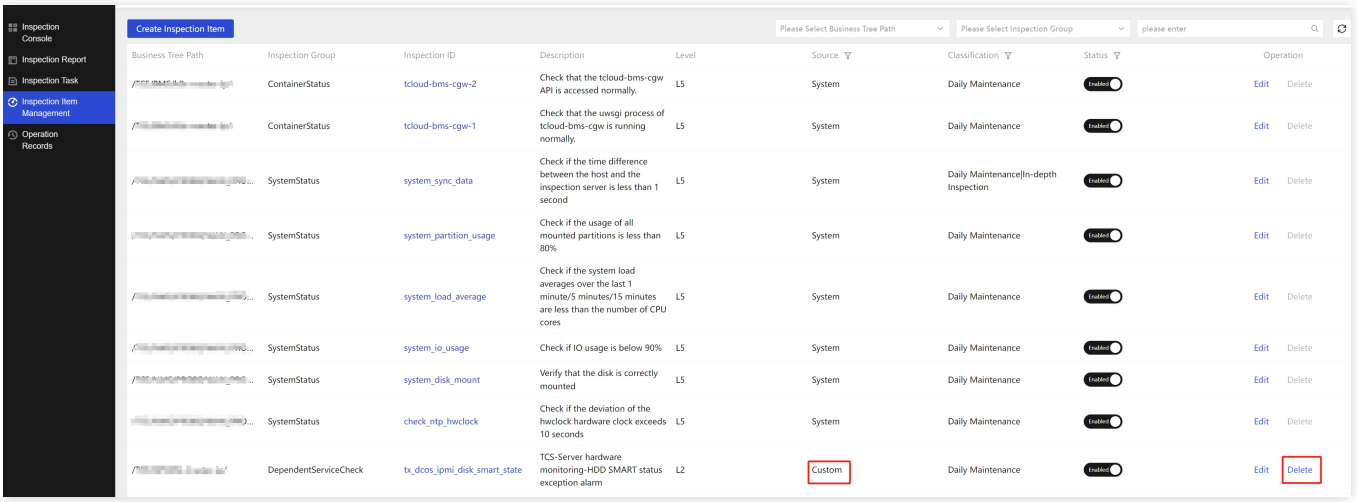
```
uuid: tx_cbs_pod_processor
host_mode: target_any
host_type: container
stack_name: tce-storage
func: PodActiveCheck
args:
  cmd_list:
    - cmd: df -h | grep '/' |tr -d '%' | awk '{if ($5>80)print}' | wc -l
      desc: Check whether the number of mounted partitions whose pod container disk utilization is greater
than 80% is 0.
      expect_equation: eq
      expect_value: 0
  pod_name: ocloud-cbs-processor
desc: The pod ocloud-cbs-processor container status is running, and the disk utilization is less than 80%.
btree_resources: []
```

Here, we only list the parts that differ between this inspection item and the previous one:

- The current inspection item's host_mode is set to target_any, and its host_type is set to container. This indicates that the inspection needs to be executed only once and should run inside the container (pod).
- The current inspection item's func is set to PodActiveCheck, and its args is set to the parameters passed to PodActiveCheck. This function also supports the simultaneous declaration of multiple shell commands.

# 6. Deleting Inspection Items

Click delete to delete a custom inspection item. System inspection items cannot be deleted. If you do not need to use them, you can disable them.

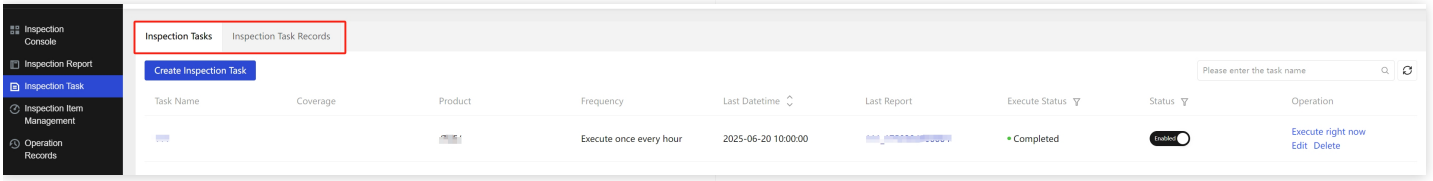

Inspection Console
Inspection Report
Inspection Task
Inspection Item Management
Operation Records
Create Inspection Item
Please Select Business Tree Path
Please Select Inspection Group
please enter
Business Tree Path
Inspection Group
Inspection ID
Description
Level
Source
Classification
Status
Operation
ContainerStatus
tcloud-bms-cgw-2
Check that the tcloud-bms-cgw API is accessed normally.
L5
System
Daily Maintenance
Enabled
Edit
Delete
ContainerStatus
tcloud-bms-cgw-1
Check that the uwsgi process of tcloud-bms-cgw is running normally.
L5
System
Daily Maintenance
Enabled
Edit
Delete
SystemStatus
system_sync_data
Check if the time difference between the host and the inspection server is less than 1 second
L5
System
Daily Maintenance|In-depth Inspection
Enabled
Edit
Delete
SystemStatus
system_partition_usage
Check if the usage of all mounted partitions is less than 80%
L5
System
Daily Maintenance
Enabled
Edit
Delete
SystemStatus
system_load_average
Check if the system load averages over the last 1 minute/5 minutes/15 minutes are less than the number of CPU cores
L5
System
Daily Maintenance
Enabled
Edit
Delete
SystemStatus
system_io_usage
Check if IO usage is below 90%
L5
System
Daily Maintenance
Enabled
Edit
Delete
SystemStatus
system_disk_mount
Verify that the disk is correctly mounted
L5
System
Daily Maintenance
Enabled
Edit
Delete
SystemStatus
check_ntp_hwclock
Check if the deviation of the hwclock hardware clock exceeds 10 seconds
L5
System
Daily Maintenance
Enabled
Edit
Delete
DependentServiceCheck
tx_dcos_ipmi_disk_smart_state
TCS-Server hardware monitoring-HDD SMART status exception alarm
L2
Custom
Daily Maintenance
Enabled
Edit
Delete

# Viewing Inspection Tasks

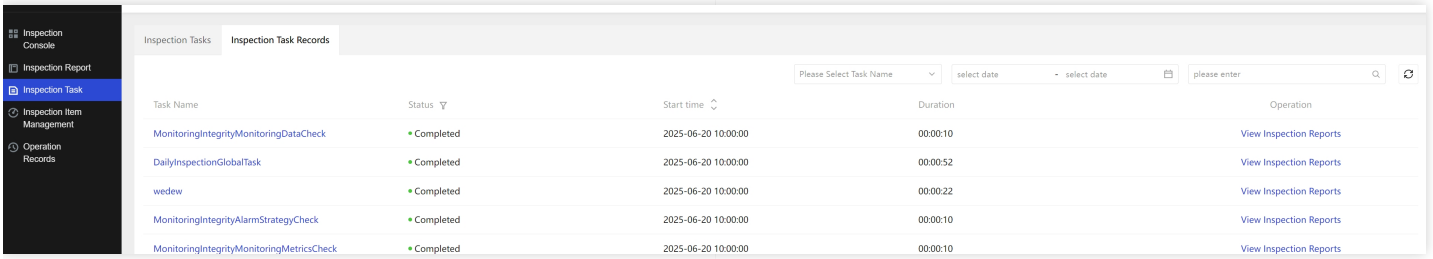
Navigate to Inspection Platform > Inspection Tasks. It includes both the system built-in inspection tasks and user-defined inspection tasks.

## 1. List of Inspection Tasks

Manages all inspection tasks and inspection task execution records on the platform and supports visualized management of inspection task execution. On the interface, you can directly view the inspection task execution records and perform operations such as inspection task creation, enabling/disabling, immediate execution, editing, and deletion.




">



2. Inspection Task Details Page

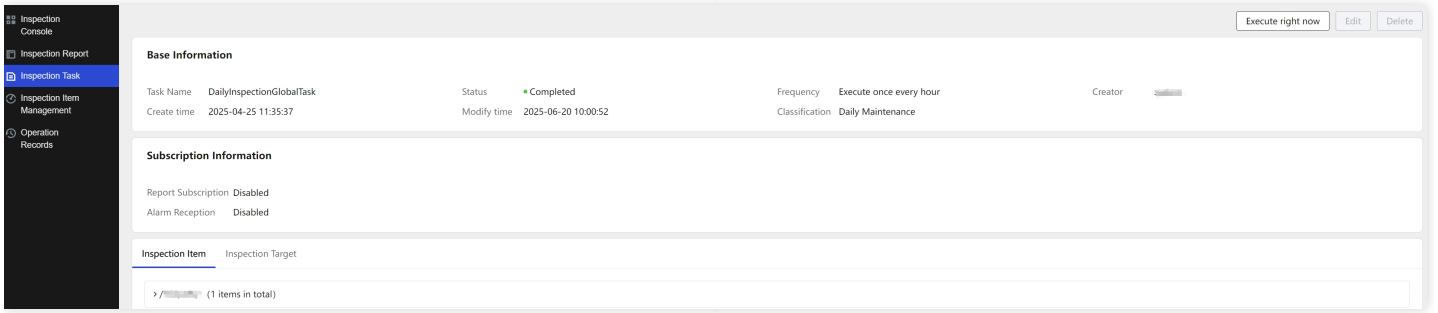
In the inspection task list, click the Task Name you want to view to enter the inspection task detail page.

(1) Basic Information displays the task name, status, inspection frequency, creation time, and other information of the inspection task.

(2) Subscription Information displays the report subscription and alarm subscription configuration status of the inspection task.

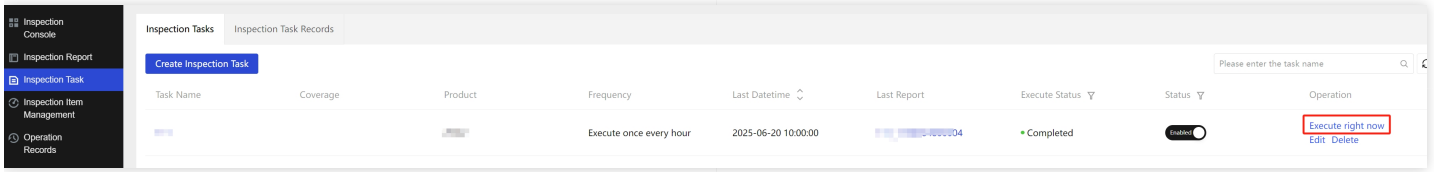
(3) Inspection Item List displays inspection items in the task based on the product node.

(4) Inspection Target List displays inspection targets in the task based on the product node.



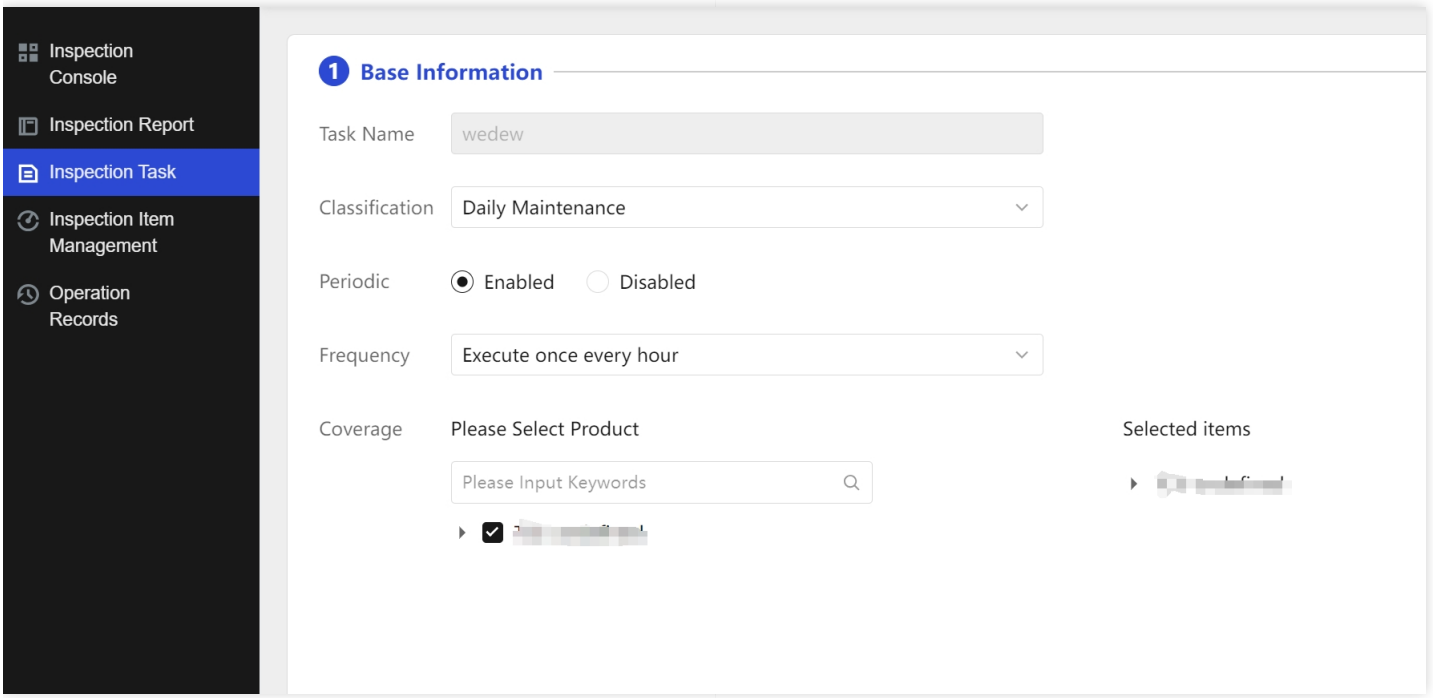

# 3 Execute inspection tasks

Click the immediate execution button for the inspection task. After confirmation, the execution of the inspection task will be triggered immediately. After execution, the inspection results will be displayed in the inspection console and the inspection report module.




# 4 Edit Inspection Task

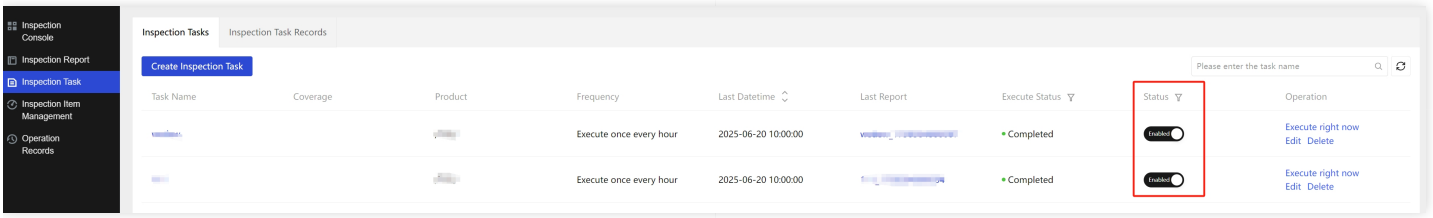
Click the Edit button for the inspection task to edit/modify the task.




When submitting the modifications, select Execute Task Immediately. If you select Yes, the modified inspection task will be executed immediately and will follow the updated information; if you select No, the modified inspection task will take effect the next time it is executed.

# 5 Enable/disable inspection task

Enabling: Click the toggle button in the disabled status to enable a disabled inspection task. The inspection task will be executed according to the inspection frequency.
Disabling: Click the toggle button in the enabled status to disable an enabled inspection task. The inspection task will no longer be executed.




# 6 Create Inspection Task

Support visualized management of inspection tasks. The guidelines for creating a custom inspection task are shown below.

Access it via Inspection Platform - Inspection Task and click Create Inspection Task.

(1) Fill in the Basic Information of the Inspection Task

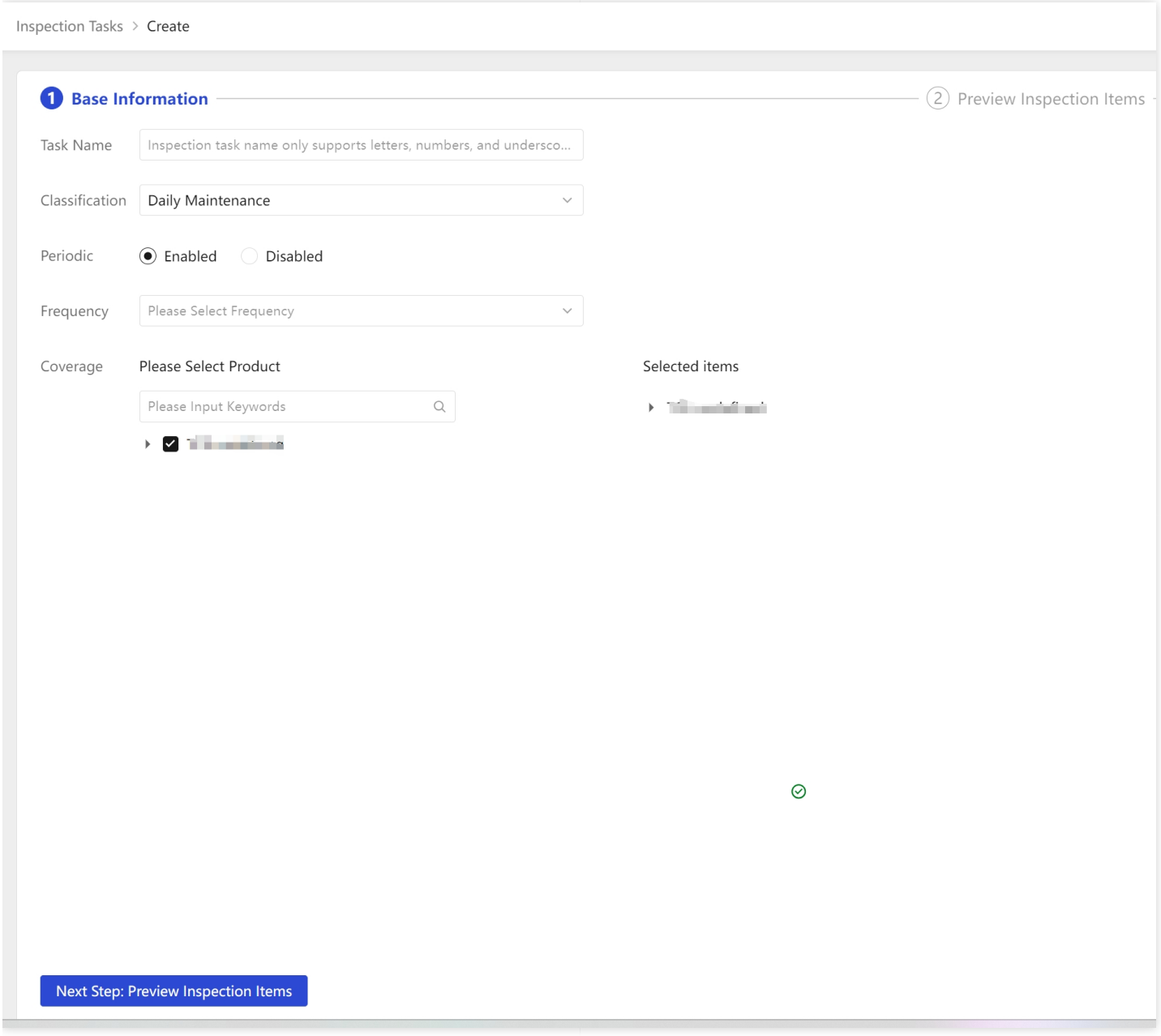
Task name: Enter the name of the inspection task.

Category: Select in the dropdown list.

Scheduled automatic inspections: Select whether to enable scheduled inspections.

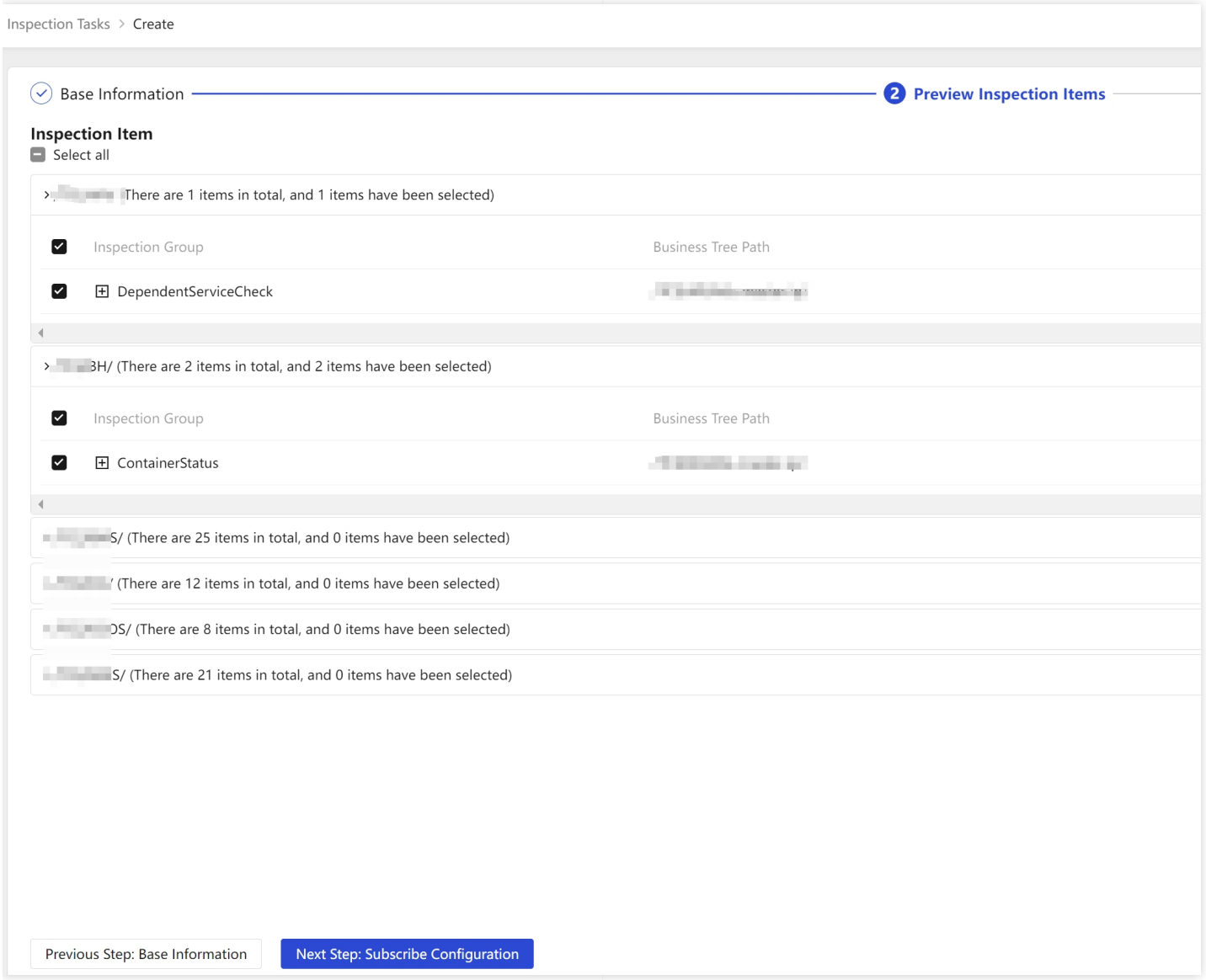
Inspection frequency: Select the frequency of automatic inspections for the inspection task in the dropdown list.

Inspection scope: Select the business tree scope for this inspection task according to information such as the product cluster and product node. You need to select business tree nodes that contain machines.



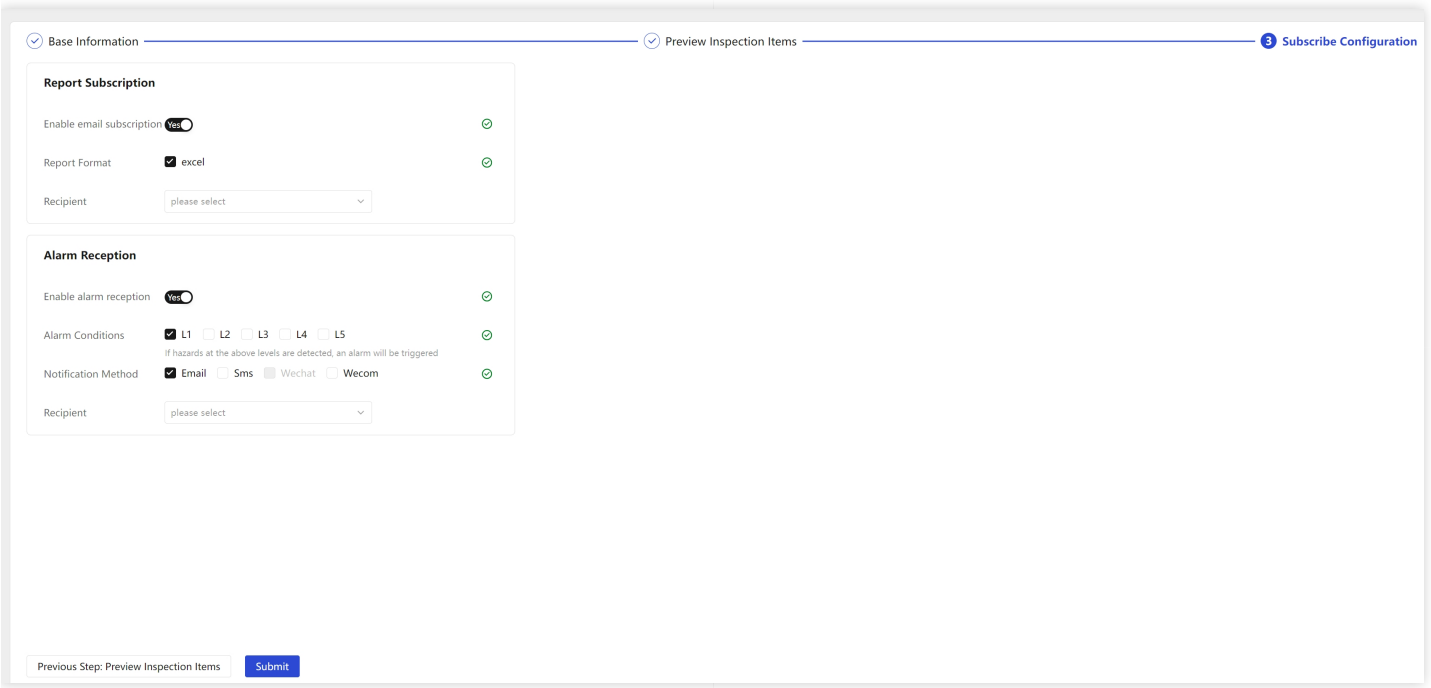

(2) Select Inspection Items

Check the inspection items to be executed.




(3) Setting the Subscription

Set the information for Report Subscription and Alarm Reception according to your needs.

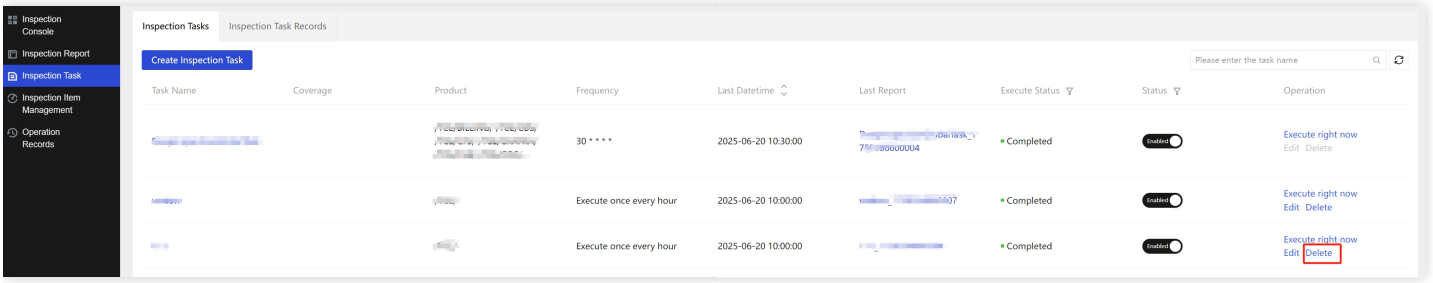



(4) Submitting an Inspection Task

When submitting the inspection task, you can select whether to execute it immediately. After selecting immediate execution and submitting it, the inspection task will be executed immediately.

# 7 Delete Task

Click the Delete button for the inspection task to delete the selected inspection task. Once deleted, the task cannot be recovered. Therefore, proceed with caution. Inspection reports that are generated by the executed inspection tasks will not be deleted. (Note: Default inspection tasks initially provided by the system cannot be deleted.)

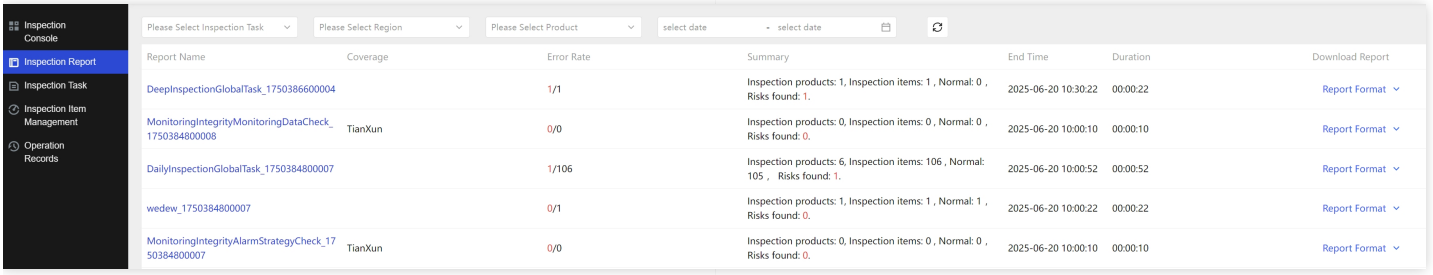


# Viewing Health Check Results

Choose Inspection Platform > Inspection Report. The execution results of inspection tasks will be displayed in the form of inspection reports. You can directly view the inspection results on the frontend page, or download inspection reports to review the inspection results.

## 1 Inspection Report List

You can view the inspection report list data, including the report name, inspection coverage, hazard detection ratio, result summary, completion time, and inspection duration. On the page, you can query data by inspection task, site, inspected product, and time.

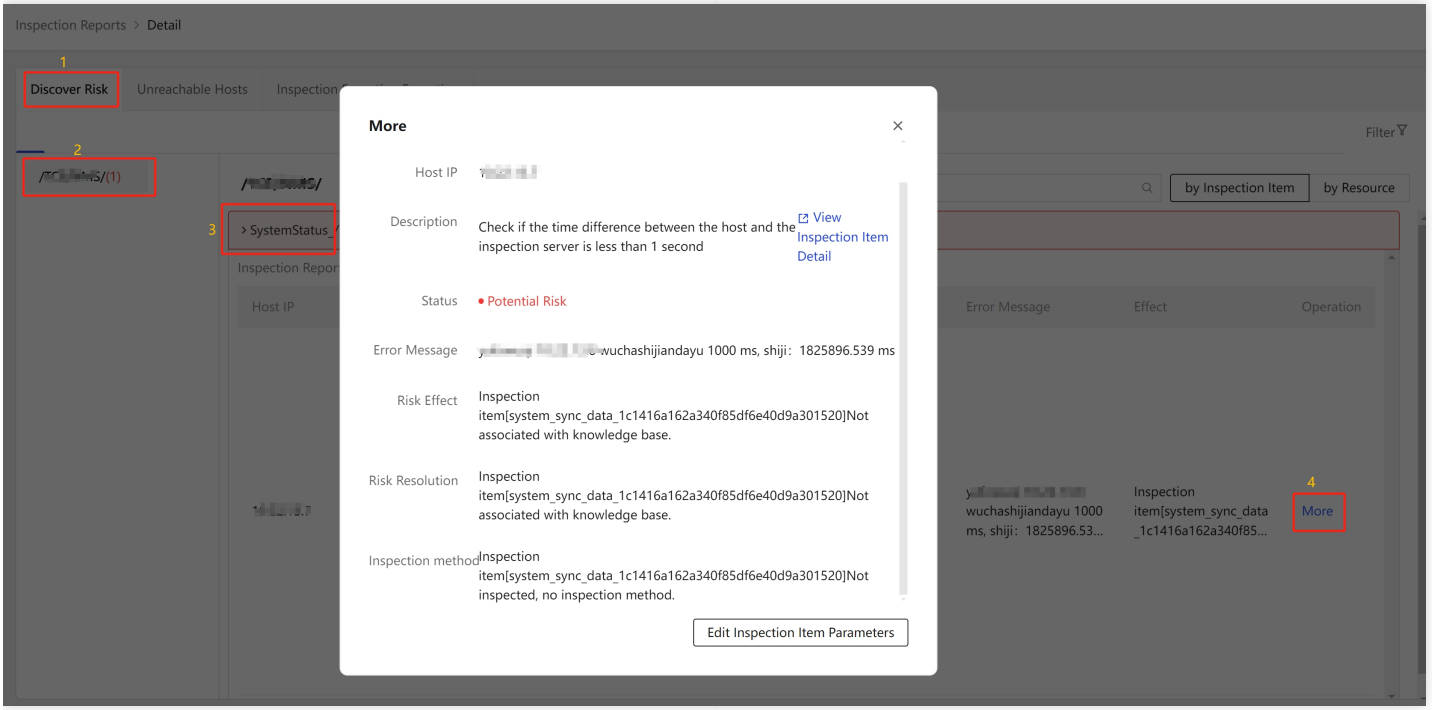



## 2 Details Page of Inspection Report

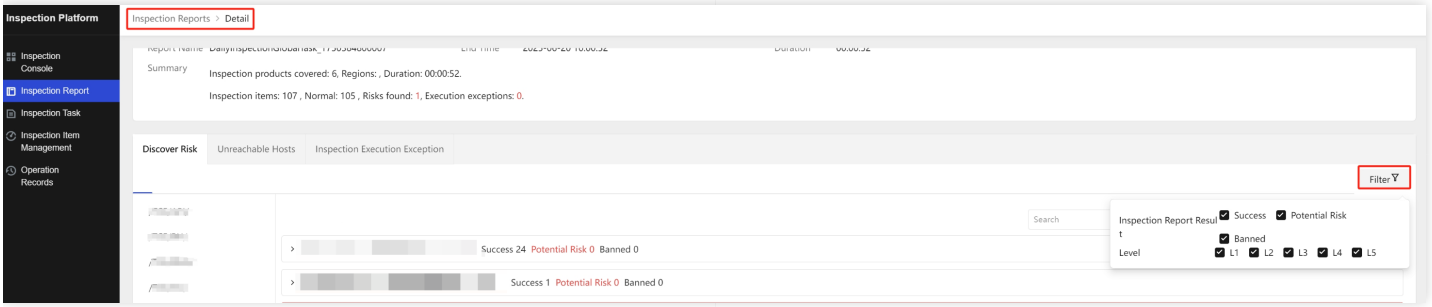
Click Report Name in the inspection report list to enter the inspection report details page, where the current inspection summary, identified hazards, unreachable host list, and abnormal inspection executions are displayed.

(1) Identified Hazards

If a red number appears after the product name on the current page and it is not 0, it indicates that the current inspection task has identified hazards in the inspected product. You can click the corresponding Product to expand the red inspection group, and then click More for inspection items with abnormal status to view detailed inspection results.

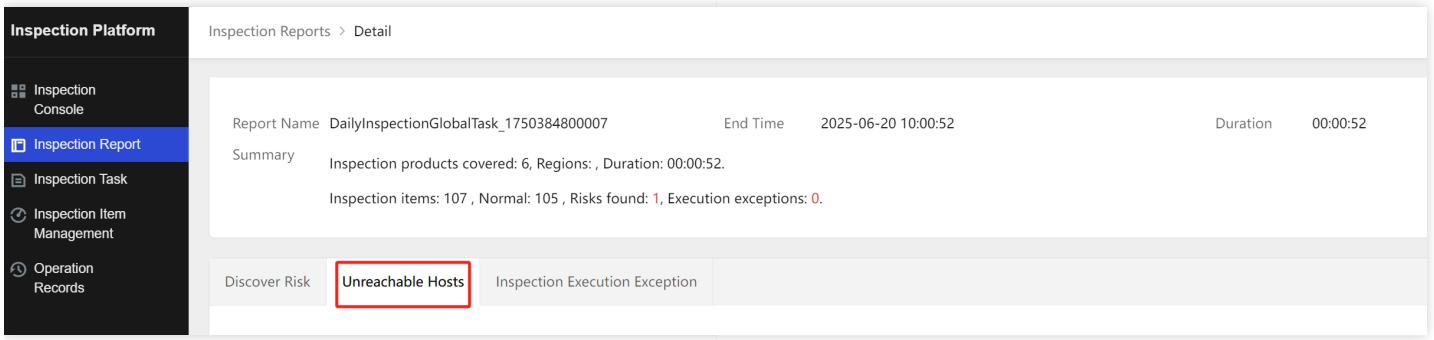



On the Inspection Report Details page, you can filter by condition by checking the Inspection Result Status and Inspection Item Grading.



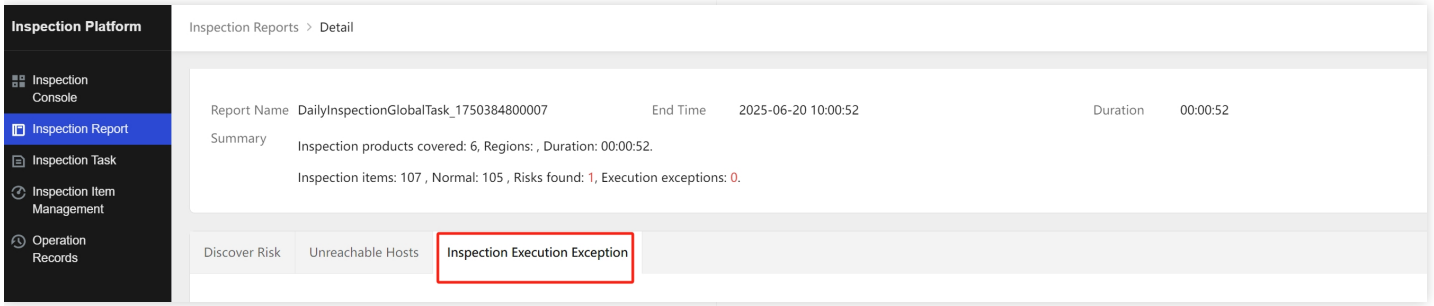

(2) Unreachable Host List

Click Unreachable Host List on the inspection report details page. You can see the information of hosts that failed to connect during the current inspection task. Use the error messages in the list to troubleshoot issues such as network connectivity or password-free configuration settings.



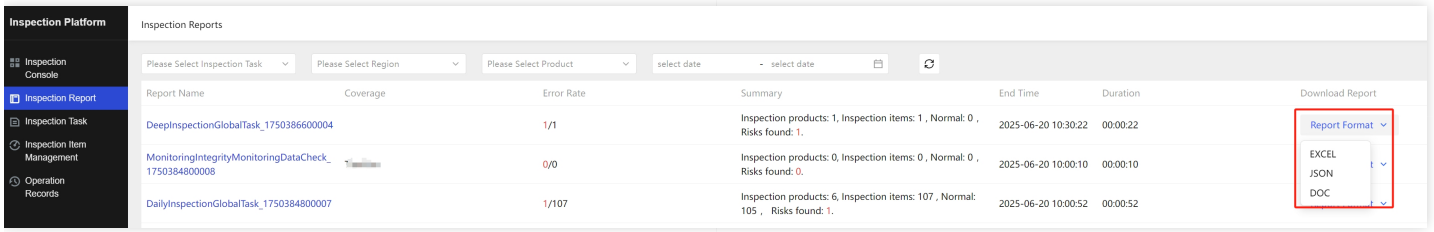

(3) Abnormal Inspection Executions

Click Inspection Execution Anomaly Items on the inspection report details page. You can see a summary of inspection items where host retrieval failed or inspection execution failed within the current inspection report.




# 3 Download the Inspection Report

Go to the Inspection Report page and click Report Format to download the inspection report in the corresponding format.



# Cloud Monitor Inspection Item List

| Inspection Item ID | Inspection Item Description | Inspection Item Group | Category | Business Tree Path |
|---|---|---|---|---|
| tx_barad_mysql_check_code | All code fields in the most recent 25 nws reporting statistics are 0. | Dirty data check | Daily Ops | BARAD/ tianxun |
| tx_barad_pod_alarm | Check that the container tcloud-barad-alarm is in the running state. | Container check | Daily Ops | BARAD/ k8s-master-ip |
| tx_barad_pod_api | Check that the container barad-api is in the running state. | Container check | Daily Ops | BARAD/ k8s-master-ip |
| tx_barad_pod_customize | Check that the container ocloud-barad-customize is in the running state. | Container check | Daily Ops | BARAD/ k8s-master-ip |
| tx_barad_pod_event | Check that the container tcloud-barad-event is in the running state. | Container check | Daily Ops | BARAD/ k8s-master-ip |
| tx_barad_pod_isd | Check that the container tcloud-barad-isd is in the running state. | Container check | Daily Ops | BARAD/ k8s-master-ip |
| tx_barad_pod_nws | Check that the container tcloud-barad-nws is in the running state. | Container check | Daily Ops | BARAD/ k8s-master-ip |
| tx_barad_pod_osp_dcos_barad | Check that the container ocloud-osp-dcos-barad is in the running state. | Container check | Daily Ops | BARAD/ k8s-master-ip |
| tx_barad_pod_script | Check that the container tcloud-barad-script is in the running | Container check | Daily Ops | BARAD/ k8s-master-ip |

| | | | | |
|---|---|---|---|---|
| | state. | | | |
| tx_barad_pod_update2 | Check that the container tcloud-barad-update2 is in the running state. | Container check | Daily Ops | BARAD/ k8s-master-ip |
| tx_barad_pod_waf_barad | Check that the container tcloud-waf-barad is in the running state. | Container check | Daily Ops | BARAD/ k8s-master-ip |

# Inspection Handling
# Inspection of NWS Data Reception

## Scenarios

tx_barad_mysql_check_code
The product side reports that the data reception is abnormal.

## Prerequisites

The tcloud-barad-nws container is running normally.

## Exception Handling

Scenario 1:
Cloud product access configuration is not updated:
Enter any tcloud-barad-isd container and execute the action to update product access configuration:
sh /data/release/product.pkg/tools/sync_product.sh xxx
Take the CVM as an example:
sh /data/release/product.pkg/tools/sync_product.sh qce/cvm

Scenario 2:
For cloud product configuration issues, you need to submit a ticket for consultation and handling.

# Inspection of Container Running Status

## Scenarios

tx_barad_pod_xxx
The container runs abnormally.

## Prerequisites

No

## Exception Handling

Enter the corresponding container and check the health check script:
cat /tce/healthchk.sh
Analyze the service exception according to the content of the corresponding container's detection script.
Try restarting for recovery. If it fails to recover, submit a ticket to our queue colleagues for handling.

# Daily Monitoring
# Common Monitoring Metrics

| Metric/Event Name | Metric Meaning | Type | Unit | Threshold | Contingency Plan | |
|---|---|---|---|---|---|---|
| count_over_time({tcs_product="barad",tcs_type="tcs_barad",event_name="FlinkJobStatus"}[1m]) | Flink job status detection | | | 0 | Restart the Flink job | |
| barad_access_message_delay_histogram_bucket | Reporting latency | histogram | dtdurations | | | |
| barad_access_http_request_count_total_increase | Number of requests | gauge | count | | | |
| barad_access_http_request_duration_seconds_p95 | Request duration (p95) | histogram | dtdurationms | | | |
| barad_access_http_request_size_bytes_p95 | Request body size (p95) | histogram | bytes | | | |
| barad_access_http_request_count_total_increase | Number of requests | gauge | count | | | |
| barad_access_http_request_duration_seconds_p95 | Request duration (p95) | histogram | dtdurationms | | | |
| barad_access_http_request_size_bytes_p95 | Request body size (p95) | histogram | bytes | | | |
| metric_writer_fields_count_increase | Number of written fields | counter | count | | | |
| metric_writer_parse_fail_count_increase | Number of parsing faults | counter | count | | | |
| metric_writer_parse_metric_count_increase | Number of parsed metrics | counter | count | | | |
| metric_writer_recv_data_delay_p95 | Receiving latency (P95) | histogram | dtdurationms | | | |
| amp_alert_process_convergence_req_total_increase | Total number of convergence requests | counter | count | | | |
| amp_alert_process_duration_ms_bucket_p95 | Request duration (P95) | histogram | dtdurationms | | | |
| amp_alert_process_process_duration_ms_bucket_p95 | Processing duration (P95) | histogram | dtdurationms | | | |
| amp_alert_process_req_total_increase | Requests | counter | count | | | |
| amp_alert_process_richError_req_total_increase | Number of rich errors | counter | count | | | |
| amp_alert_process_send_success_req_total_increase | Number of successful sends | counter | count | | | |
| amp_subscribe_process_duration_ms_bucket_p95 | Subscription processing duration (P95) | histogram | dtdurationms | | | |
| amp_subscribe_process_req_hit_total_increase | Number of request hits | counter | count | | | |
| amp_subscribe_process_yehe_notify_send_increase | Number of messages sent to the message | counter | count | | | |

| Metric/Event Name | Metric Meaning | Type | Unit | Threshold | Contingency Plan | |
|---|---|---|---|---|---|---|
| | center | | | | | |
| flink_taskmanager_Status_JVM_CPU_Time_increase | CPU usage time | counter | dtdurations | | | |
| flink_taskmanager_Status_JVM_GarbageCollector_ConcurrentMarkSweep_Count_increase | Number of CMS_GC operations | counter | count | | | |
| flink_taskmanager_Status_JVM_GarbageCollector_ConcurrentMarkSweep_Time_increase | CMS_GC duration | counter | dtdurationms | | | |
| flink_taskmanager_Status_JVM_Memory_Heap_Used | Memory usage | gauge | bytes | | | |
| barad_access_message_delay_histogram_p95 | Metric latency (P95) | histogram | dtdurations | | | |

# Alarm Handling
# Status of flink

## Alarm Description

Flink task exception. The database record status is non-4 (5 represents stopping tasks, 3 represents starting, and 4 represents normal operation).

## Alarm Attributes

| Module | Event Name | Event Level | Manual/Automatic Clearing |
| --- | --- | --- | --- |
| Flink job status monitoring | Status of flink | Level-1 alarm | Automatic clearing |

## Impact on the System

It affects the continuity of monitoring data.

## Possible Causes

The supporting Kafka or yarn cluster is abnormal.

## Processing Procedures

Step 1: Check whether the supporting Kafka instances in the environment are running normally.
Step 2: Check whether the supporting yarn cluster in the environment is running normally.
Step 3: Restart the Flink task.

## Reference Information

Check the status of job tasks (5 represents stopping tasks, 3 represents starting, and 4 represents normal operation):
mysql22001

```
select Status from online_galileo.Job;
```

# Emergency Response Plan

1. IDC power outage contingency plan

- Failure impact:
  The IDC may experience a global power outage of servers handling Cloud Monitor service due to power supply, air conditioning, cabinet, or switch failures, resulting in service interruption.
- Recovery steps:
  i. Use batch scripts or commands to confirm that all machines are powered on and network connectivity is recovered.
  ii. Confirm whether the processes running on each machine have been started normally.
  iii. Confirm that the storm, Kafka, and ES services have started, and observe the corresponding UI.
  iv. The data reception service cluster is stateless. After the process is started, it can automatically recover.
  v. If the previous services have not been started, you need to confirm whether the ZK service is normal.
  vi. Use the local ZK-data file to recover ZK, and then monitor the startup status of the above services.
  vii. Monitor whether the Kafka data directory file is updated normally.
  viii. Observe whether the storm-ui data is calculated normally.

2. Kafka service overall cluster troubleshooting contingency plan

- Pre-check:
  Check whether the Kafka status is normal.
  Enter the bin directory under the deployed ZK, and run zkCli.sh and ls/kafka/kafka/okers/ids.
  If the above command returns fewer than 3 items, it indicates a node exception. Execute get /kafka/kafka/brokers/ids/130618222416261120 to obtain the node 2.ip information (the ID number is taken from the above return results). Execute the returned ID in sequence and compare them with the IP addresses of Kafka deployment, and you will obtain the information about the abnormal Kafka node.
- Failure impact:
  The Kafka service, as a cache node for monitoring data, is an important link in stream computing. Normally, it works in cluster mode. After the cluster crashes, data cannot be written to the cache, resulting in interruption of monitoring data and failing to display isd.barad data. Monitoring alarms cannot be sent normally, seriously affecting operations.
- Recovery steps:
  i. Recover and start all Kafka servers as soon as possible, and start the corresponding processes.
  ii. If the machine cannot start, you need to replace it with a new storage-type machine.

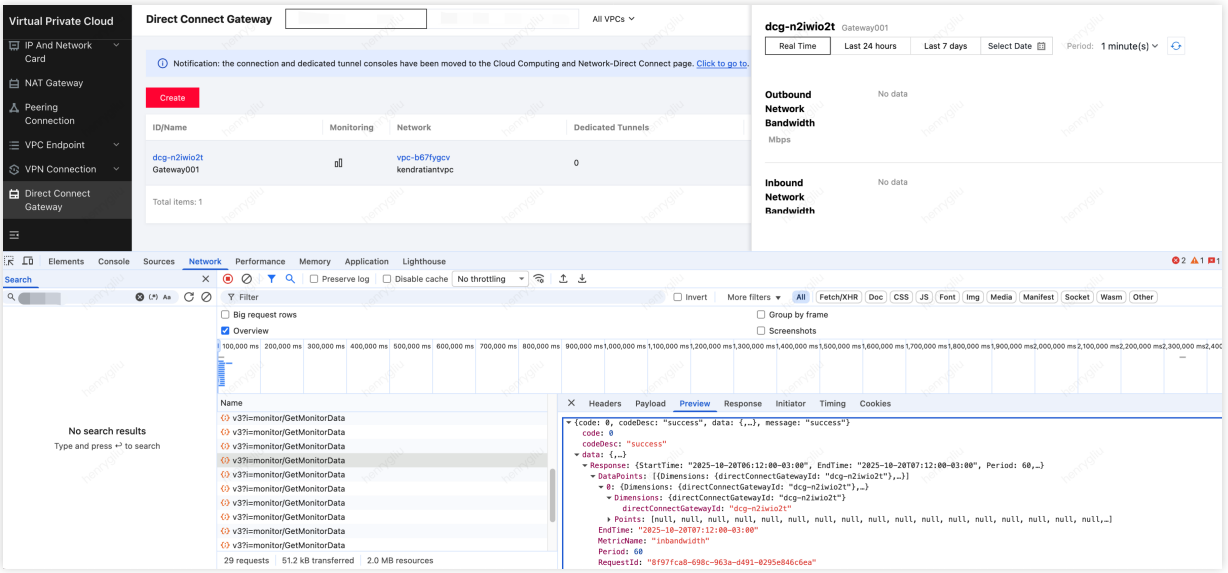
iii. The kafka=id of the new machine should be consistent with that of the old machine. The Kafka ID information can be obtained from ZK.

iv. After the abnormal Kafka process is recovered, it will automatically proceed to Kafka data recovery.

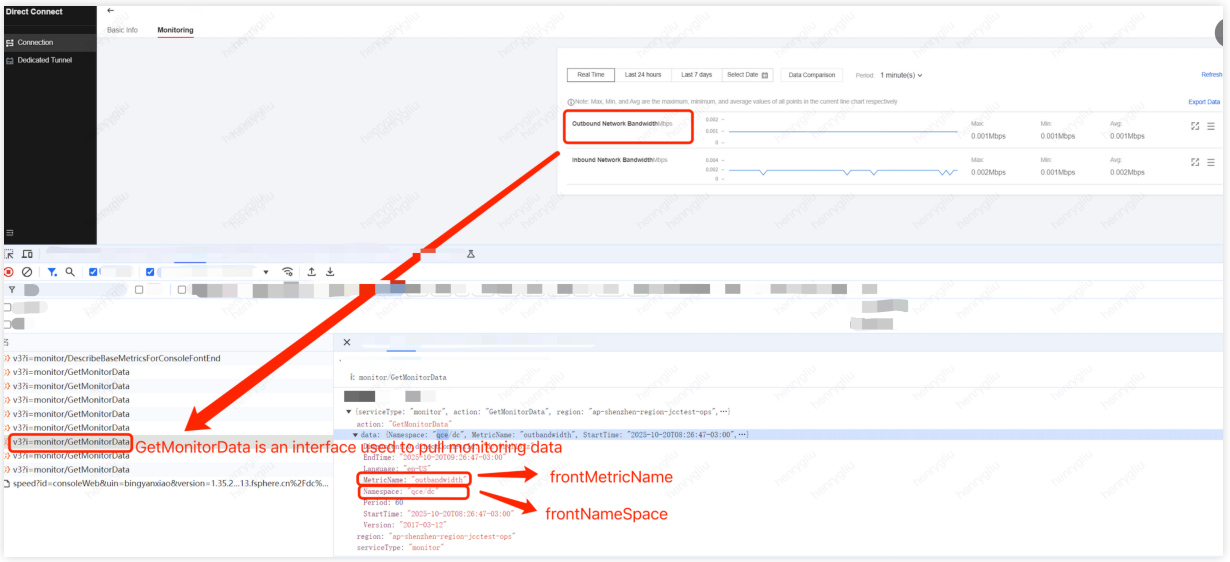
v. Log in to the Kafka server and execute the kafka-topic-describe command to obtain the real-time status.

# Best Practices

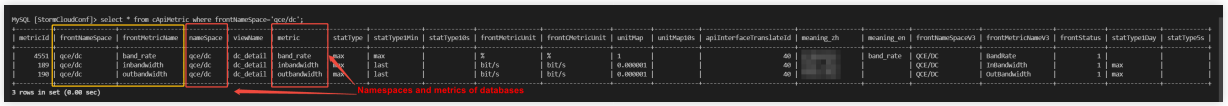
1. No monitoring data of the connection, DC tunnel, and dedicated gateway
   Symptom:




   i. No data is observed on the frontend. Press F12 to obtain the frontNameSpace and frontMetricName of the input parameters.




   ii. Use frontNameSpace and frontMetricName obtained from frontend packet capture in backend packet capture to check whether any data is reported. Manual packet capture: Access the regional database of the Cloud Monitor database. (Generally, mysql22003 supports directly accessing the regional database. An alias is used for the database. If mysql22003 does not support accessing the regional database in the customer's environment, use the actual port of the customer's environment.)
   For example, select *from StormCloudConf.cApiMetric where frontNameSpace = 'qce/dc';




Query the mapped backend namespace and backend metrics, and then perform packet capture in barad-nwspod, for example, tcpdump -i any -s0 -nn -A 'dst port 80'|grep'qce/dc' |grep 'inbandwidth'.

iii. Confirm whether there is a problem with the access configuration, for example, whether the front-end reports a network error. Check whether the database has the namespace access configuration. Compare with the archived .db file, the following methods can basically confirm whether there is a problem with the access configuration. Detect and find whether qce/dc, qce/dcx and qce/dcg are accessed.

- Observe whether the left side of the page of the monitoring configuration center provided by barad-isd lacks the related namespace. Sample address: http://isd.barad.Example.com. (The specific domain can be obtained from the TCS master, kubectl get ingress -A |grepisd.barad.)
- For example, execute select * from cShowProductConfig where namespace = '{$namespace}' in the primary database of StormCloudConf; no record is found.
- select * from cApiMetric where namespace = '{$namespace}'; no data exists or the data does not match. A special case is that no API metrics are received.
- select * from cObservationMetric where namespace = '{$namespace}'; no data exists or the data does not match.

iv. Solution:

Re-import the correct access configuration. Some customers also need to update the barad-isd image packet. Update the image (or qce/dc qce/dcx qce/dcg, the access configuration of the barad-isd image, is good).

Enter the barad-isd pod:

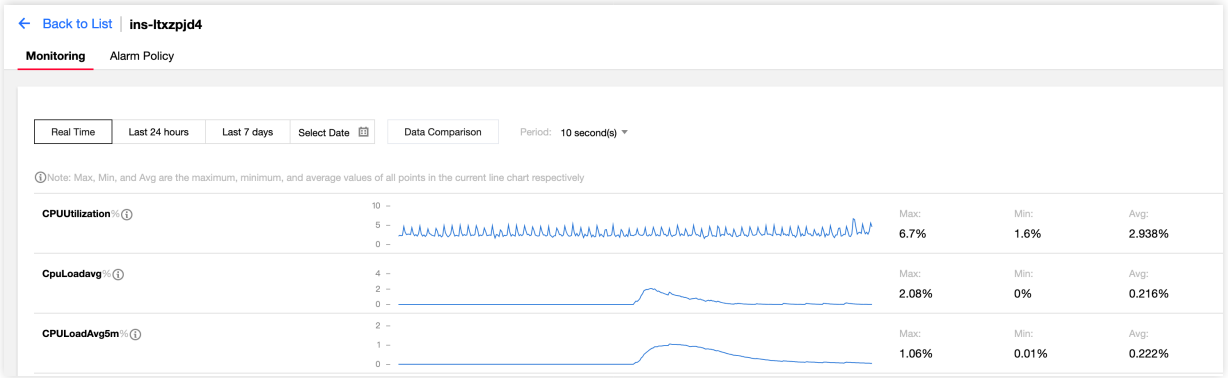
cd /data/release/product.pkg/tools , shsync_product.sh qce/dc qce/dcx qce/dcg

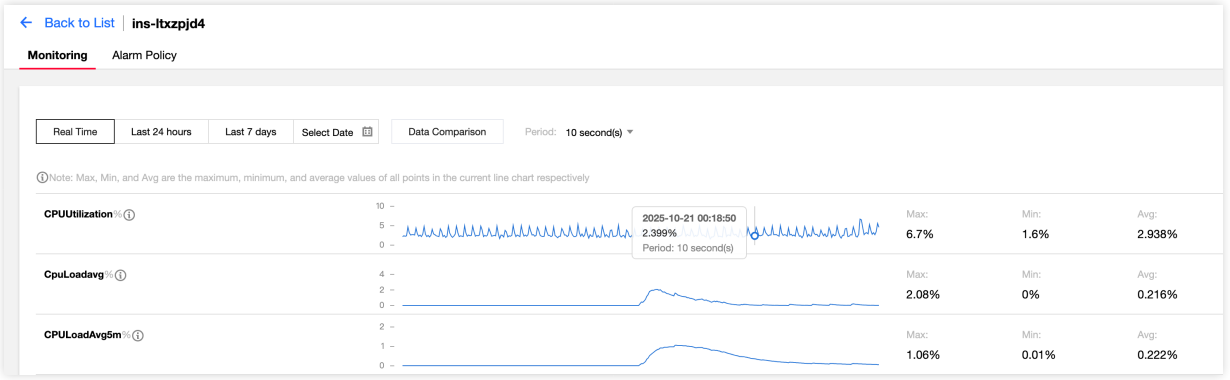
If no error is reported after execution, go to the page to check whether the monitoring data display is recovered.

2. Data generated in real-time monitoring, but not in 7-day monitoring

- Symptoms are as follows:

There is real-time data, but the 7-day data is abnormal.

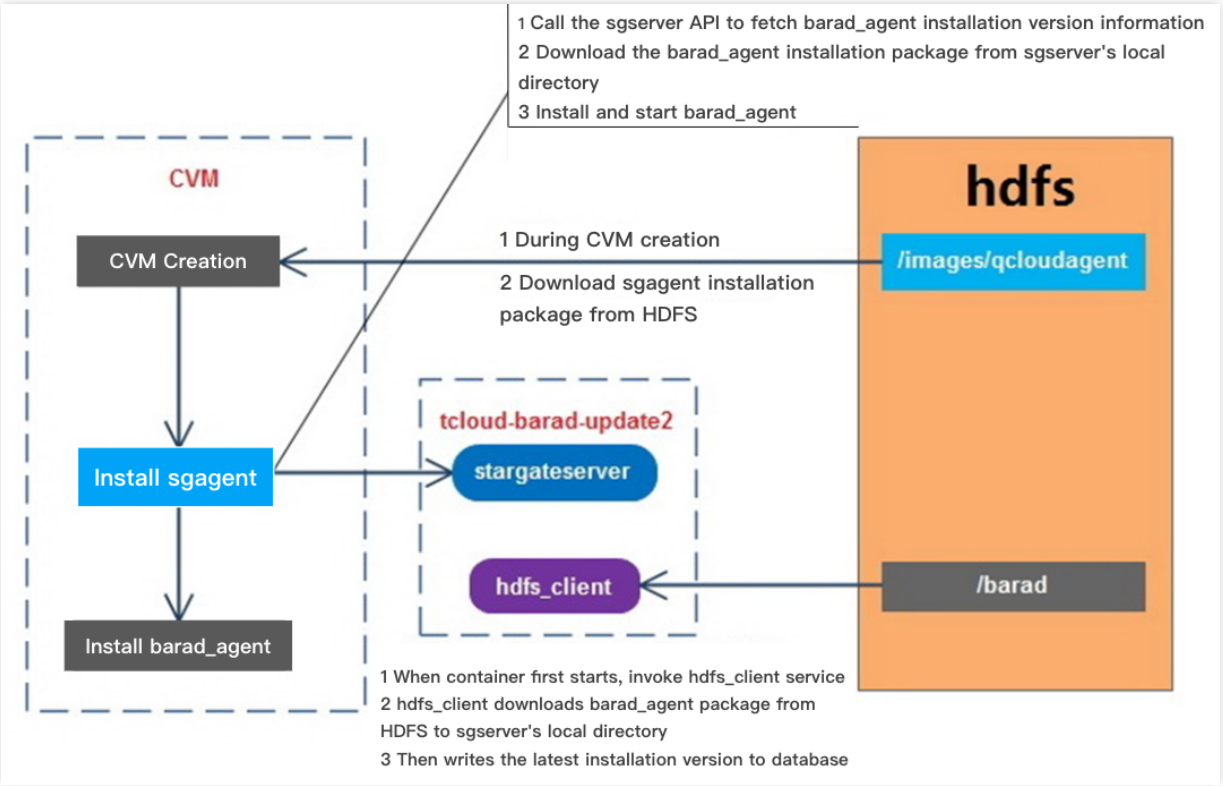





- Reason for abnormal 7-day data:
  Aggregation rule of Cloud Monitor. If an aggregation rule is configured, this rule will be used preferentially to aggregate data. Otherwise, the daily data and hourly data are aggregated based on a certain granularity. The 5-minute granularity is prioritized over the 1-minute granularity.
- Solution:
  Execute the following SQL on the secondary database to delete the daily aggregation method and use the default 5 minutes to calculate the 3600s data. Be sure to back up the database before the operation.

DELETEa from cAssociationMiningRule as a, cDimensionView as b, cDimensionGroup as cwhere a.outputDimGroupId=c.dimGroupId and a.outputPeriod>300 andb.dimensionGroup=c.dimensionGroup and b.namespace=c.namespace and b.namespacein("qcce/nat_gw_bm","qce/bm_lb","qce/cmq","qce/cvm","qce/vpc_region_conn","qce/cmongo","qce/bm_intra_lb","qce/cmqtopic","qce/internal_loadbalance_rs","qce/docker","qce/lb","tnm/vsr_vpnconn","qce/nat_qos_stats","qce/cbs","qce/dcg","qce/cos","qce/pcx","qqce/dc","qce/dcx","qce/lb_total","qce/cdp","qce/cdn","qce/ckafka","qce/bm_pcx","qce/scf","qce/ov_cdn","qce/apigateway","qce/loadbalance","qce/qaap","qce/nat_rs_bm","qce/internal_scf","qce/rlb","qce/cdb","qce/redis","qce/phy_host","qce/bm_rlb","qcce/sqlserver","qce/tdsql","qce/ctsdb","qce/iotmq","qce/csg","qce/vpc_net_detect","qce/ces","qce/emr","pce/bm","pce/cos","pce/netdevice","pce/stgw_lb_ld","pce/lb_ld","pce/cfs_resource","qce/vpc_gateway","qce/vpc_nat","qce/vpc_jnsgw","pce/cdb_privaate","pce/redis","pce/ccb_weblogic");

3. No monitoring data for some CVM metrics
   i. Failure symptoms:
      There is no monitoring data on the CPU utilization, memory usage, and memory utilization of the CVM.
   ii. Installation flowchart of barad_agent




iii. Installation process of barad_agent

- If the user chooses to enable Cloud Monitor when creating a CVM.
- After the CVM is successfully created, it will download the latest version of the Stargate installation package from the /images/qcloudagent directory of HDFS (generally the stargate_linux_install_v1.2.6.tar.gz version).
- Install and start the sgagent service.
- The container (tcloud-barad-update2) where sgserver is located will execute the hdfs_client service command when it is first started.
- hdfd_client will download the latest barad_agent installation package from the /barad directory of HDFS (generally /barad/baradagent_installer of the cloud platform) to the local /data/www/update2.agent.yun.com/update directory and write the version information to the database.
- After sgagent starts, it will call the sgserver API to query the installed version information of barad_agent.
- Download the barad_agent installation package from the /data/www/update2.agent.yun.com/update directory of sgserver.
- Install and start the barad_agent service.
- In the whole process:
  - The barad_agent installation package is provided by the CVM team, which ensures that the latest installation package information is uploaded to the /barad directory of HDFS.
  - The barad_agent installation package and configuration are maintained by the CVM team. The sgagent is merely responsible for download, installation, and startup.

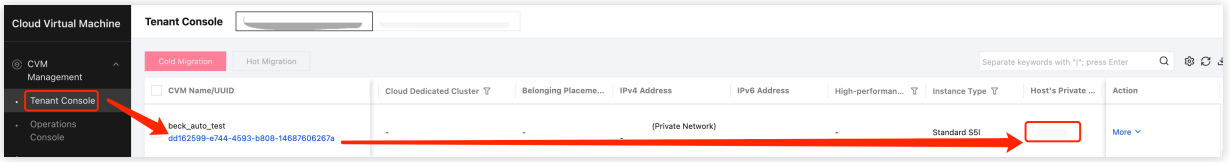
iv. Steps for problem localization in the CVM data reporting process

- First, perform packet capture confirmation on the NWS machine to check whether there is data reporting.
  - Capture packets by using the UUID of the CVM as a keyword.
    tcpdump-i any -nn -s0 -A 'dst port 80' | grep '8d8bf61d-e938-4f0d-95cc-b455d9f778a1'
    Filtering only by the UUID keyword may only capture the reported data from the host, while there is no data reported from the CVM instance.

- So it is recommended to add the metric keyword for packet capture: tcpdump -i any-nn -s0 -A 'dst port 80' | grep '8d8bf61d-e938-4f0d-95cc-b455d9f778a1' | grep'"cpu_usage'
- If no data is captured, log in to the instance to continue confirmation.

◦ Method to log in to the CVM instance

- Query the IP address of the host through the UUID on the operation side.




- Log in to the host.
  Enter the jump server in the environment where it is located, for example, M14.
  Use the root account for login. Contact technical support personnel for the password.
  [m14@cloud64 ~]$ ssh root@10.19.0.21
  Log in to the CVM instance via UUID.
  virsh console 8d8bf61d-e938-4f0d-95cc-b455d9f778a1
  Some CVM instances require an instance account and password to log in. (You can ask the creator of the CVM instance for them.)

VM_0_34_centos login: root
Password:
Last failed login: Sun May 24 12:52:59 CST 2020 on ttyAMA0
There were 2 failed login attempts since the last successful login.
Last login: Thu May 21 14:34:53 on ttyAMA0
Welcome to tlinux 2.4 for aarch64
Version 2.4 20191012
tlinux-2.4-kvm-guest-aarch64-20200115
[root@VM_0_34_centos ~]#

- Confirm whether the sgagent has been properly installed.
  Check if the sgagent process has started normally.
  ps -aux | grep agent

```
[root@VM_0_4_centos ~]# ps -aux | grep agent
root      1038  0.0  0.2 106076  1824 ttyAMA0  S+   12:57   0:00 grep --color=auto agent
[root@VM_0_4_centos ~]#
[root@VM_0_4_centos ~]#
```

- If the sgagent is not installed, you need to confirm two points.
- Whether a user selects to enable Cloud Monitor when creating a CVM.
- Whether there is an installation package of sgagent in the /images/qcloudagent/ directory in HDFS.

- Method of logging in to HDFS
  - Search for the node information on HDFS.
  - Enter the tcloud-barad-update2 container.
  - View TCE service configuration information vim/tce/conf/cm/local.json.
  - Find HDFS-related configurations.




- Log in to HDFS.
  Log in to the HDFS server with the root account. Contact technical support personnel for the password.
  [m14@cloud64 ~]$ ssh root@10.23.0.133
- View HDFS directory information.
  View the sgagent installation package.
  hadoop fs -ls/images/qcloudagent
  View the barad_agent installation package.
  hadoop fs -ls /barad
- Upload the sgagent installation package.
  If you do not see the /images/qcloudagent/stargate_linux_install_v1.2.6.tar.gz installation package in HDFS, it may be caused by the installation fault of the production component product-barad-sgagent.
  - If it is an internal environment, such as yf1 or TencentOS, rerun the QCI process and manually install the agent.
  - If it is a customer environment, reinstallation by the technical personnel is required.

    > Note:
    >
    > The product-barad-sgagent production component installation procedure mainly involves two things.

    Decompress the original stargate_linux_install installation package, replace the related configuration in the TCE environment, and then repackage it.
    Upload the packaged installation package to the /images/qcloudagent/ directory in HDFS.
  - If it is a temporary test, you can consider manually uploading the installation package to HDFS.
    hadoop fs -put /tmp/stargate_linux_install_v1.2.6.tar.gz /images/qcloud
  - After ensuring that the HDFS has the sgagent installation package, when a user creates a new CVM and selects to enable Cloud Monitor, sgagent will be automatically downloaded and installed.

- Confirm whether the barad_agent is normal.
  - Log in to the CVM instance to confirm whether the barad_agent starts normally.
    ps -aux | grep agent

```
[root@VM_0_17_linux logs]# ps -aux | grep agent
root      1025  0.0  0.1  97296  1680 ?        Sl   May19   0:01 /usr/local/qcloud/stargate/sgagent -d
root      1102  0.0  0.7 155216  7576 ?        S    May19   0:11 barad_agent
root      1108  0.0  0.9 181716  9760 ?        S    May19  10:30 barad_agent
root      1109  0.2  1.3 740296 14192 ?        Sl   May19  43:56 barad_agent
root     30738  0.0  0.0 112660   968 ttyS0    R+   12:53   0:00 grep --color=auto agent
[root@VM_0_17_linux logs]#
```

- If the barad_agent process starts normally but there is still no data reporting, the cause may be environmental issues or barad_agent configuration problems. (Contact CVM and TCE personnel to handle it.)
  You can also follow the steps below to check and provide relevant information to CVM and TCE personnel for locating.
  - Confirm whether the plugin.ini configuration is normal.
    cd /usr/local/qcloud/monitor/barad/
    [root@VM_0_17_linux barad]# catetc/plugin.ini
  - View whether dispatcher.log and executor.log output abnormal information.
  - Check whether the metadata is reachable.
    Whether curl metadata_url/uuid can return data normally, such as curl http://product-cvm-metadata.cityA.yf-1.tcepoc.fsphere.cn:80/meta-data/uuid.
- If the barad_agent is not started:
  - Confirm whether the sgagent exception causes the barad_agent installation fault.
    Check whether the URL in /usr/local/qcloud/stargate/etc/base.conf can be accessed normally. Check whether there are exception logs in /usr/local/qcloud/stargate/logs/stargate.log.
  - If the address http://update2.agent.yun.com/interface.php requested by base.conf is not accessible:
    Enter the tcloud-barad-update2 container and confirm whether the sgserver service is running normally.
  - Confirm whether there is the latest barad_agent installation package under /data/www/update2.agent.yun.com/update in the tcloud-barad-update2 container.
  - If there is no latest installation package, the latest package can be downloaded from HDFS through hdfs_client.
    cd /usr/local/services/hdfs_client/ && ./hdfs_client -f/barad/baradagent_installer
  - ehdfs_client update fails.
    First, check whether the hdfs_client configuration is correct and the configuration is abnormal. (The cause is environmental issues. Contact our TCE colleagues to handle it.)
    It is also possible that there is no barad_agent installation package on HDFS. (Contact our CVM colleagues to handle it.)
- After confirming that there is the latest barad_agent installation package on sgserver, log in to the CVM instance and restart the sgagent service, and it can be automatically installed and started.

# Node Restart

1. Product deployment architecture

   Introduce the roles, features, and components of product nodes.

| Module/ Container | Node Type | Component Introduction |
| --- | --- | --- |
| tcloud-barad-update2 | Container | It contains Stargate servers and alarm loading services.<br>1. The Stargate server provides services for the Stargate agent (ps aux \| grep sgagent) on the CVM, mainly used to record and update CVM status information as well as distribute BARAD agents.<br>1. The alarm loading service is a unified query service designed to reduce database queries of the storm supervisor. It caches a copy of the alarm configuration data in the database to the /data/www/alarmdb.barad.yun.com/alarm_file directory in the container, and exposes an HTTP service through Nginx for the storm supervisor to pull. |
| tcloud-barad-event | Container | Contains event center CGI and handler services.<br>1. Event center CGI provides HTTP services for other vertical products to report events. After simple verification of the reported events, it writes them to TDMQ for Kafka. The topic configuration is cm_event.<br>1. The handler service consumes data from TDMQ for Kafka and converges repeatedly reported events (the basis for judgment of repeated reports is: whether the alarm object is the same, determined by the StormCloudConf.cEventProductConfig.astrictDimension field; the convergence duration, determined by the StormCloudConf.cEventProductConfig.astricPeriod), performs verification, detects whether an alarm policy is configured, and stores the data in ES. If an alarm policy is configured, it sends an HTTP request to ams to trigger an alarm.<br>1. For multi-region deployment, the tcloud-barad-event image should be deployed in each region. |
| tcloud-barad-isd | Container | Contains multiple backend management services and scripts, the service for timeout detection of Overlay instance agent reporting (to be removed soon), and the service for pulling configurations from the Cloud Monitor configuration center. Generally, multiple instances are not required.<br>Other containers generally have the ConfClient component to cache configurations in the Cloud Monitor configuration center. |
| tcloud-barad-dispatch-tsm | Container | Contains only one service that imports the data of some Cloud Monitor namespace into Argus. |
| tcloud-barad-script | Container | Contains 5 bypass services.<br>1. barad_sync_server automatically synchronizes CVM information from CCDB to the translation table for NWS to perform dimension translation.<br>1. cm_php_script is used to synchronize alarm policy information of the primary database to the regional database.<br>1. dict_access, a server for reporting dimension translation information for other products, caches data into TDMQ for Kafka. The Topic list is the table of tDictionary_* |

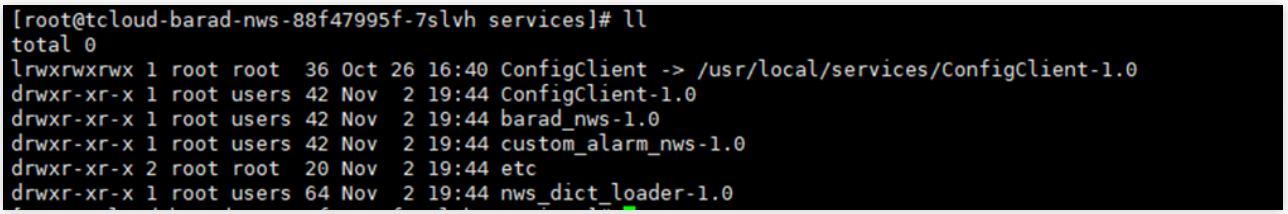
| Module/ Container | Node Type | Component Introduction |
| --- | --- | --- |
| | | in the StormDictionary library. It is recommended to create only two partitions for each topic. Otherwise, the container will continuously restart because of too many processes.<br>1. dict_sync consumes dimension translation information from TDMQ for Kafka, and then stores it in the database (to form a complete service together with dict_access).<br>1. The role of getAppTraffic is to obtain traffic data of the view under qce/lb in each region, aggregate and calculate the total inbound/outbound traffic of each tenant to qce/lb_total, and display it in the traffic monitoring column on the tenant side. |
| tcloud-barad-dataapi | Container | Contains two services: dataapi and newdataapi. Both are old APIs. They may be removed later because there are few callers. |
| tcloud-barad-nws | Container | Contains three services.<br>The NWS service is used to receive FT report data, perform legality judgments on the data, perform dimension translation, and then store it in TDMQ for Kafka.<br>The feature of NWS to perform dimension translation is guaranteed by dict_loader. This component caches the dimension translation information from the DB to the /data/storm/local_dict/ directory in the container (the information in the DB is written by dict_access and dict_sync of tcloud-barad-script).<br>The container also has a special CGI. The custom alarm receiving feature is provided by customAlarmNws. Users can customize alarms through cagent_tools on the CVM. Since the feature involves complex components, details will not be given here. |
| tcloud-barad-alarm | Container | Contains only the service cloud_alarm and mainly provides two external services. One service receives storms and reverses and transmits the alarm status (normal > abnormal > alarm > normal). The other receives platform events. |
| tcloud-barad-api | Container | Contains customapi, baradapi, ams, customAlarm, and policyapi.<br>customapi integrates with Cloud API Management and provides a frontend API and some open APIs.<br>Cloud Monitor API encapsulates the logic for calling ES, equivalent to an ES client exclusive to Cloud Monitor.<br>ams, located downstream of tcloud-barad-alarm, is used to record alarm information, connect to the message center, and send alarms.<br>customAlarm, a custom alarm related to tcloud-barad-nws, is used to update clientkey into CKV.<br>New policyapi 342 derives from customapi. |
| tcloud-barad-skywalker | Container | Contains Flink stream computation logic components and their management scripts. |
| product-barad-deploy | Container | As a production component used for deployment, it needs to be further improved. Some practical tools, such as the tool fix_es_table_err under TCE/tools/fix that fixes the timestamp type of ES tables to mills, and the tool fix_ft_host that fixes the hard-coded domain name in the ft callback API of the database can be deployed without this component. |

| Module/ Container | Node Type | Component Introduction |
| --- | --- | --- |
| product-barad-sgagent | | As a standardized component of Stargate agent, it executes deploy to directly standardize Stargate agent and transmit it to HDFS for download and installation when the host creates CVM instances. |
| yunapi-barad | | Cloud API Management configuration of BARAD |
| tcloud-barad-api-go | | Cloud Monitor API V3, which is derived mostly by modifying the policy API. The policy API is a V2 API, which should be upgraded to V3 in private cloud environments. Therefore, a proxy is implemented for tcloud-barad-api-go to convert parameters of gateway requests via V3 APIs. The parameters are converted into those for V2 APIs to call the policy API. |

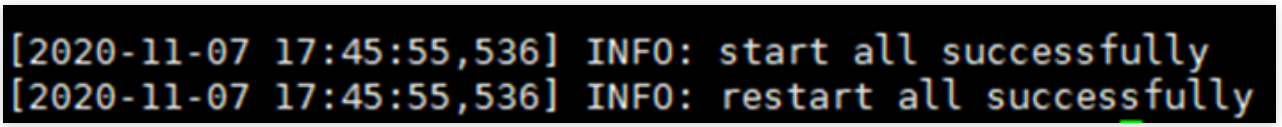
2. Restarting the pod in general
   In general, directly restart the pod where the component is located, and Kubernetes will automatically perform load traffic transfer. After deletion, it will automatically start and switch back to the cluster.
   i. Obtain the Cloud Monitor Pod list: kubectlget pod --all-namespaces -o wide |grep barad
   ii. Delete a Pod: kubectl delete pod -n tce ${pod}
   iii. Wait until the start status is:




3. Restarting by component
   i. Obtain the Cloud Monitor Pod list: kubectl get pod --all-namespaces-o wide |grep barad-nws
   ii. Enter pod: kubectl exec -it -n tce ${pod}bash.
   iii. The directory where the component is located in general: /usr/local/services/.




   iv. sh/usr/local/services/barad_nws-1.0/admin/restart.sh all
   Normal:



# Backup and Restoration

1. Database
   The stored data in the MySQL database mainly include alarm policies, cloud product access configurations, and environment configuration information. There are primary regional databases and regional databases. Primary regional databases are deployed in the primary region, while regional databases have an instance in each region.
   Back up the database. Take StormCloudConf as an example. You can use the following command.
   - Back up the whole database:

```
mysqldump-h${host} -P22001 -umysql_user -p'mysql_passwd' StormCloudConf >StormCloudConf_220
01.sql
```

- Back up the table:

```
mysqldump-h${host} -umysql_user -pmysql_passwd -P22001 StormCloudConf cStormTransmitMapcDi
ctionaryConf > StormCloudConf_22001.sql
```

cDictionaryConf and cStormTransmitMap are tables in the database StormCloudConf.

- Recover the database with the above backup file:

```
mysql-h${host} -P22001 -umysql_user -pmysql_passwd -D StormCloudConf <  StormCloudConf_2200
1.sql
```

2. Monitoring data
   i. ElasticSearch introduces the concepts of repository and snapshot to implement data backup and restoration.
   - Create a repository:

```
curl-xPOST http://${ES_IP}:9200/_snapshot/my_backup_1 -d'{
  "type": "fs",
  "settings": {
    "location":"\/data\/elasticsearch-5.1.1\/data\/back\/my_backup_1",
    "max_snapshot_bytes_per_sec":"20mb",
    "max_restore_bytes_per_sec":"20mb",
    "compress": true
  }
}'
```

- The parameters are explained as follows:
  type: The type of the repository is a shared file system.
  location: Specify the path of the repository. It should be a subdirectory of path.repo.
  max_snapshot_bytes_per_sec: This parameter can control the traffic throttling of the process when snapshot data enters the repository. The default value is 20 MB per second.
  max_restore_bytes_per_sec: This parameter controls the traffic throttling of the process when data is restored from the repository. The default value is 20 MB per second.
  compress: Whether the data is compressed.

  > Note:
  >
  > If the mounted path is a remote directory, you should reasonably configure this value to avoid network traffic overwhelming.

2. Snapshot index:
   Snapshot all open indexes.

curl-XPUT http://${ES_IP}:9200/_snapshot/my_backup_1/snapshot_1?wait_for_completion=true

Snapshot the specified index.

```
curl-XPUT http://${ES_IP}:9200/_snapshot/my_backup_1/snapshot_3 -d'{
  "indices":"my_index_1,my_index_2",
  "ignore_unavailable": true,
 "include_global_state": true
}'
```

3. Recover the snapshot:

curl-XPOST http://${ES_IP}:9200/_snapshot/my_backup/snapshot_1/_restore?wait_for_completion=true

4. Snapshot information:
   Obtain information for a single snapshot.

curl-XGET http://${ES_IP}:9200/_snapshot/my_backup/snapshot_1

- Obtain information for all snapshots under the repository.

curl-XGET http://${ES_IP}:9200/_snapshot/my_backup/_all

# Appendix A: Common Operations References

1. Configuration file list

| Image Name | Module Name | Configuration File Name | Configuration File Path |
|---|---|---|---|
| ocloud-barad-customize | tbarad | config.ini | /usr/local/services/barad/ocloud/conf/ |
| tcloud-barad-alarm | cloud_alarm | local.json | /tce/conf/cm/ |
| tcloud-barad-event | barad.cm.event | arm_event_front_nws.conf<br>event_handle.ini | /tce/conf/config/barad.cm.event/ |
| tcloud-barad-isd | barad.cm.isd | local.json | /tce/conf/cm |
| tcloud-barad-nws | Barad.cm.nws | custom_alarm_nws.conf<br>nws_nws.conf | /tce/conf/config/barad.cm.nws/ |
| tcloud-barad-storm | Storm-nimbus | Barad_Comm_config.ini<br>CloudMIC_config.ini<br>NWS_CAT1_config.ini<br>UpdateConf_config.ini<br>alarm_loader_client.config.ini<br>storm.yaml | /tce/conf/config/storm.cm |
| tcloud-barad-update2 | stargate server<br>Alarm loading service | alarm_loader_config.ini | /tce/conf/config/barad.cm.update2 |
| tcloud-barad-dispatch-tsm | barad_dispatch_argus | barad_dispatch_argus.toml | /tce/conf/config/ |
| tcloud-barad-script | barad_sync_server | sync_config.ini | /tce/conf/config/barad.cm.script/ |

2. Detailed description of the configuration file

   i. tbarad (provides customized API service for the operation-side)

   config.ini configures information on APIs customized by the operation side.

| Configuration Item | Example |
| --- | --- |
| INS_URL | http://ocloud-cfs-ops.cfs:80/cfs_api/tce/get_barad_dimention |
| INS_JSON | {"version":"1.0","caller":"CGW","password":"CGW","callee":"cdn", "eventId":1097532882,"timestamp":1427732027," interface":{"interfaceName":"Qcloud.Cfs.GetBaradDimessions","para":{}}, "postOperation":[]} |
| DB_MODE | mysql |
| HOST | db-2.db.gaia-1.yfm17.city A.yf-m17.tcecqpoc.fsphere.cn |
| PORT | 22003 |
| USER | 43o68ui4qcdl |
| PASS | a97oo3faahcf |
| NAME | cnas |
| NAS_ACCESS_IP_LIST | ["10.28.12.33","10.28.11.37"] |

2. cloud_alarm (reverses the alarm status and receives platform event services)
   local.json cloud_alarm service configuration

| Configuration Item | Example | Description |
| --- | --- | --- |
| Request domain | alarm.barad.yun.com | Alarm request address |
| ip | 192.168.192.224 | Host address |
| url | http://alarm.barad.yun.com:8000/barad.cgi? | URL address |
| port | 8000 | Requester |

3. barad.cm.event (event reporting service)
   arm_event_front_nws.conf configuration event center CGI.

| Configuration Item | Example | Description |
| --- | --- | --- |
| listen_port | *:45008 | Service listening port |
| maximum_connections | 100000 | Maximum Connections |

| Configuration Item | Example | Description |
| --- | --- | --- |
| maxfds | 65535 | Maximum number of files |
| log_level | debug | Log level |
| log_rotate_size | 10 | Log rotation size |
| max_cachesize | 131072 | Maximum cache size |
| working_threads | 4 | Number of work processes |
| kfk_broker_list | kafka-1.kafka.gaia-1.yfm17.city A.yf-m17.tcecqpoc.fsphere.cn:9092 | Kafka connection address |
| kfk_request_timeout | 1000 | Kafka request timeout |
| kfk_message_timeout | 2000 | Kafka message timeout duration |
| kfk_statistics_intervals | 1000 | Statistical interval time |
| datafile_path | /data/backup/eventdata | Data storage path |
| conf_db_host | db-1.db.gaia-1.yfm17.city A.yf-m17.tcecqpoc.fsphere.cn | Configuring the database hostname |
| conf_db_port | 22001 | Database port |
| conf_db_user | wtn34rkf5m1d | Database account |
| conf_db_pass | z4czfz3ro58x | Password |
| conf_db_base | StormCloudConf | Database name |
| log_path | ../log/event_front | Log path |
| topic_name | cm_event | Kafka topic name |

4. event_handle.inihandle service configuration

| Configuration Item | Example | Description |
| --- | --- | --- |
| LOG_PATH | ../log/event_handle | Log path |
| LOG_LEVEL | 3 | Log level |

| Configuration Item | Example | Description |
| --- | --- | --- |
| PARTITION_NUM | 8 | Number of rotations |
| KFK_BROKER | db-1.db.gaia-1.yfm17.city A.yf-m17.tcecqpoc.fsphere.cn | Kafka connection address |
| KFK_TOPIC_NAME | cm_event | Kafka topic name |
| KFK_GROUP_ID | cm_event_grp | Kafka group ID |
| ES_URL | zcac3j77.ces.infra.tce.io:48104 | ES domain name |
| DB_PORT | 22001 | Database port |
| DB_IP | db-1.db.gaia-1.yfm17.city A.yf-m17.tcecqpoc.fsphere.cn | Database address |
| DB_PWD | z4czfz3ro58x | Database password |
| DB_CHARSET | utf8 | Character encoding |
| DB_USER | wtn34rkf5m1d | Database account |
| DB_NAME | StormCloudConf | Database name |
| ALARM_DB_NAME | CloudAlarmV2 | Table Name |
| AMS_URL | ams-producteventalarm.tcloud-barad-api.barad/ams/productEventAlarm.php | AWS request address |
| CKV_HOST_LIST | ckv1.barad:9101,ckv2.barad:9101,ckv3.barad:9101 | CKV host address |
| REDIS_HOST | gm5iaz.credis.infra.tce.io | Redis address |
| REDIS_PORT | 40620 | Redis port |
| REDIS_BID | redis-barad | Redis name |
| REDIS_PWD | redis | Redis password |

5. Alarm loading service

alarm_loader_config.ini alarm loading service configuration

| Configuration Item | Example | Description |
| --- | --- | --- |
| host | 127.0.0.1:9393 | Listening host port |
| port | 22003 | Database port |
| dbname | StormDictionary | Database Name |
| timeout | 2000 | Connection timeout |
| path | ../log/ | Log path |
| level | info | Log level |
| size | 256 | Log size |

6. barad-nws (data access)

custom_alarm_nws.conf custom message configuration file

| Configuration Item | Example | Description |
| --- | --- | --- |
| listen_port | *:8880 | Listening port |
| maximum_connections | 200000 | Maximum Connections |
| maxfds | 65535 | Maximum number of files |
| log_level | error | Log level |
| log_rotate_size | 100 | Log rotation size |
| max_cachesize | 131072 | Cache size |
| working_threads | 4 | Number of work threads |
| servername | NWS | Service name |
| accesslog | ../log | Access log path |
| accesslog_level | normal | Access log level |
| accesslog_rotate_num | 10 | Number of access log rotations |
| accesslog_rotate_time | 300 | Access log rotation time |

| Configuration Item | Example | Description |
| --- | --- | --- |
| accesslog_rotate_size | 1000 | Access log rotation size |
| module | ../lib/libalarmreport.so | Custom alarm, connect to the management center |
| receiver_uri | /sendAlarm.cgi | Specify a CGI name to receive data |
| alarm_center_uri=/ams/customAlarm.php | /ams/customAlarm.php | Connection method of the alarm management center |
| alarm_center_domain | customalarm.tcloud-barad-api.barad | |
| alarm_center_hostip | 192.168.252.73 | |
| alarm_center_timeout | 30 | |
| retry_times | 3 | Three retries after a sending fault |
| limit_per_key_per_minute | 120 | Frequency limit of one appclientkey per minute |
| failed_save | 1 | Whether to enable saving as a file after a retry fault |
| failed_send | 1 | Whether to enable continuing to try to send the content saved as a file |
| failed_report_filepath | /data/alarmlog | Path to save as a file |
| failed_report_file_row | 100 | Number of alarms for a single file |
| failed_report_file_time | 30 | Data timeliness of a single file |

7. nws.nws.conf nws data access configuration

| Configuration Item | Example | Description |
| --- | --- | --- |
| zookeeper_host | 10.28.0.9:2181,10.28.0.11:2181,10.28.0.10:2181 | ZK node |
| zookeeper_timeout | 20000 | Connection timeout |
| zookeeper_logfile | ../log/zookeeper.log | ZK log path |
| zookeeper_log_level | error | Log level |
| zookeeper_update_interval | 60 | Update interval |

| Configuration Item | Example | Description |
| --- | --- | --- |
| namespace_cgi_rootpath | /storm110/cloud/TransConf | Namespace CGI root path |
| kfk_broker_listpath | /kafka/kafka/brokers/ids | Kafka node configuration |
| kfk_max_message_len | 499000 | Maximum message length |
| kfk_message_timeout | 180000 | Message timeout duration |
| send_thread_num | 8 | Number of threads for receiving messages |
| dictConfPath | /storm110/cloud/DictConf | Translation dictionary path |
| dictDBConfPath | /storm110/cloud/ExtendConf/DictAndPlugConf | Translation database |
| localDictStorePath | /data/storm/local_dict | Local dictionary storage path |
| transRuleConfPath | /storm110/cloud/translateConf | Translation rule path |

8. barad_dispatch_argus (import part of Cloud Monitor namespace data to argus)
   barad_dispatch_argus.toml

| Configuration Item | Example | Description |
| --- | --- | --- |
| LogPath | ../log/barad_dispatch_argus.log | Log directory |
| LogLevel | 5 | Log level: Critical-0, Error-1, Warning-2, Notice-3, Info-4, and Debug-5 |
| TsmInterface | http://internal.ocloud-osppro-argus2-interface.osppro/report/ArgusCustomReport | API address |

| Configuration Item | Example | Description |
| --- | --- | --- |
| Dsn | 93tp38qmns44:i28ytm4t93vb@tcp(db-2.db.gaia-1.yfm17.city A.yf-m17.tcecqpoc.fsphere.cn:22003)/StormCloudConf?autocommit=true | Configuring Cloud Monitor DB |
| TickerSecs | 30 | Cache loading interval |

9. barad_sync_server (bypass service configuration)

| Configuration Item | Example | Description |
| --- | --- | --- |
| dictdb | db_name = StormDictionary<br>db_host = db-2.db.gaia-1.yfm17.city A.yf-m17.tcecqpoc.fsphere.cn<br>db_port = 22003<br>db_user = 4qzhoi0bz17y<br>db_passwd = kcq2depb8hnv | Translation dictionary database configuration |
| confdb | db_host = db-2.db.gaia-1.yfm17.city A.yf-m17.tcecqpoc.fsphere.cn<br>db_port = 22003<br>db_user = 93tp38qmns44<br>db_passwd = i28ytm4t93vb<br>db_name = StormCloudConf | Connection configuration of configuration databases |
| log | log_path = ../log/<br>log_level = info<br>#log size (MB)<br>log_size = 100 | Log path, level, and size configuration |
| server | [server]<br>server_host = 1.1.1.1<br>server_port = 9090 | Service IP port configuration |

10. barad-isd (monitoring configuration center)

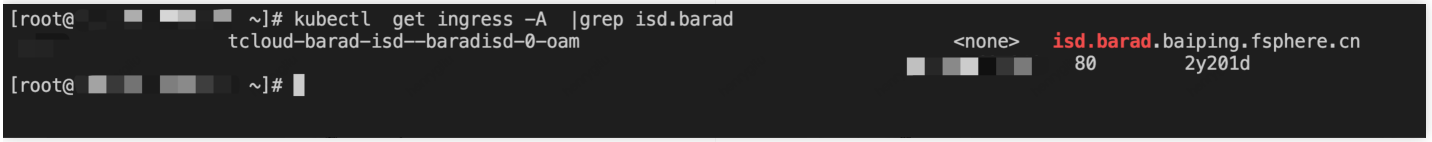
| Configuration Item | Example | Description |
| --- | --- | --- |
| Request domain | conf.barad.yun.com | Configuration center address |
| ip | 192.168.192.222 | Host address |

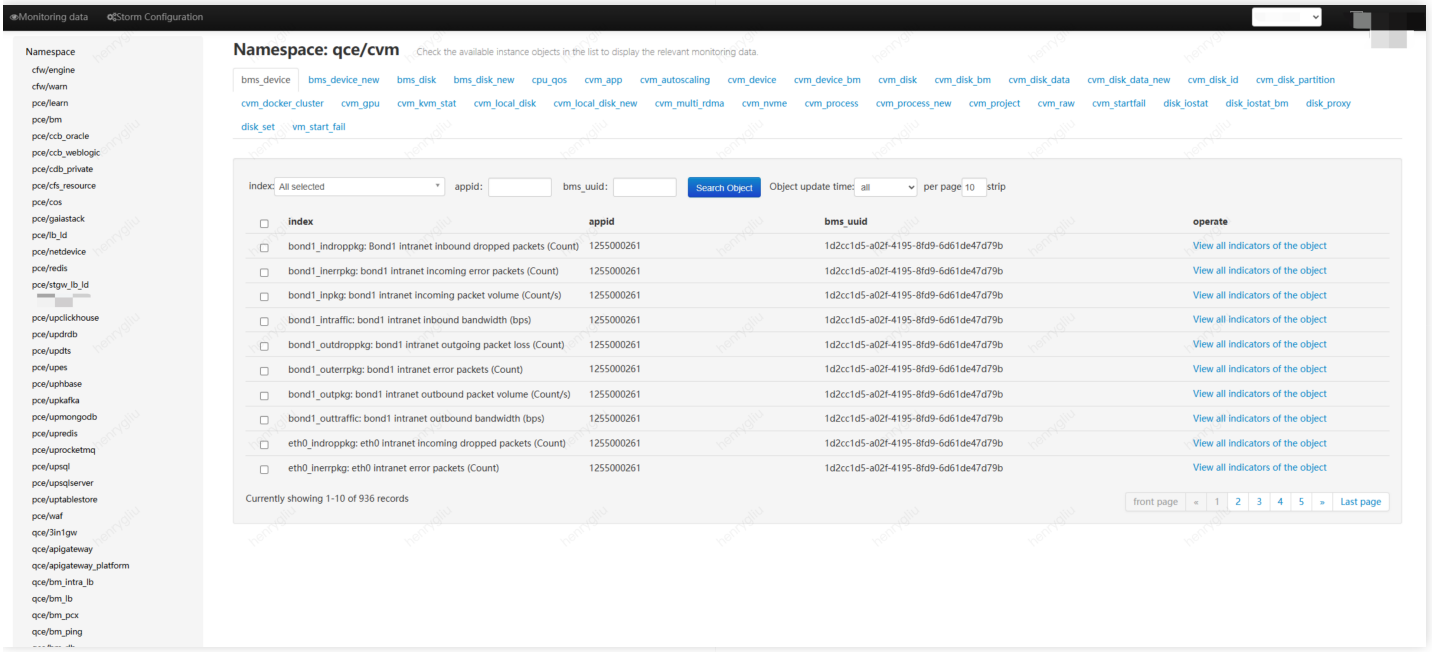
| Configuration Item | Example | Description |
| --- | --- | --- |
| url | http://conf.barad.yun.com | URL address |
| port | 80 | Request port |

# Monitoring Tool Query and Usage

kubectl get ingress -A | grep isd.barad

Access the domain names found on the page to query instance metrics.



# Obtaining Operation Logs

Common query API names and log paths:

| Module | Configuration Name | Log Path |
| --- | --- | --- |
| tce-ocloud-barad-customize | nginx | /data/storage/log/tnginx_1_0_0-1.0/*.log |
| tce-tcloud-barad-alarm-amp | alarm_apm | /data/storage/log/barad-alarm-amp/*.log |
| tce-tcloud-barad-alarm-consumption | consumption | /data/storage/log/consumption/*.log |
| tce-tcloud-barad-alarm-detector | detector | /data/storage/log/barad-alarm-detector/*.log |
| tce-tcloud-barad-alarm-policy | policy | /data/storage/log/policy/*.log |
| tce-tcloud-barad-alarm-synchronizer | synchronizer | /data/storage/log/barad-alarm-synchronizer/*.log |
| tce-tcloud-barad-api | ams_new | /data/storage/log/barad/barad.deploy.api/ams_new/*.log |
| tce-tcloud-barad-api | deploy | /data/storage/log/barad.deploy.api/api/*.log |
| tce-tcloud-barad-api | nginx | /data/storage/log/tnginx_1_0_0-1.0/*.log |
| tce-tcloud-barad-api | yunapi | /data/storage/log/barad/barad.deploy.api/yunapi/*.log |
| tce-tcloud-barad-event | event_front | /data/storage/log/event_front-1.0/*.log |
| tce-tcloud-barad-event | event_handle | /data/storage/log/event_handle-1.0/*.log |
| tce-tcloud-barad-isd | nginx | /data/storage/log/barad/barad.deploy.isd/dockerlog/tnginx_1_0_0-1.0/*.log |
| tce-tcloud-barad-monitor-dashboard-api | api_metric_translator | /data/storage/log/api_metric_translator-1.0/*.log |
| tce-tcloud-barad-nws | custom_alarm_nws | /data/storage/log/custom_alarm_nws-1.0/*.log |
| tce-tcloud-barad-query-api | query | /data/storage/log/query-api/init.log |

| Module | Configuration Name | Log Path |
| --- | --- | --- |
| tce-tcloud-barad-script | barad_sync_server | /data/storage/log/barad_sync_server-1.0/*.log |
| tce-tcloud-barad-script | dict_access | /data/storage/log/dict_access-1.0/*.log |
| tce-tcloud-barad-script | dict_sync_tool | /data/storage/log/dict_sync_tool-1.0/*.log |
| tce-tcloud-barad-script | getAppTraffic | /data/storage/log/dockerlog/getAppTraffic/*.log |
| tce-tcloud-barad-update2 | alarm_loader | /data/storage/log/alarm_loader-1.0/*.log |
| tce-tcloud-barad-update2 | ConfigClient | /data/storage/log/ConfigClient-1.0/*.log |
| tce-tcloud-barad-update2 | php | /data/storage/log/barad-php-5-6-29-1.0/*.log |
| tce-tcloud-barad-update2 | update2 | /data/storage/log/barad/update2/www/*.log |

# Log File

Component Logs

| Module | Log Location | Log Feature |
| --- | --- | --- |
| alarm-policy | /usr/local/services/barad_nws-1.0 | The NWS service is used to receive FT report data, perform legality judgments on the data, perform dimension translation, and then store it in TDMQ for Kafka.<br>The feature of NWS to perform dimension translation is guaranteed by dict_loader. This component caches the dimension translation information from a DB to a container. The log records operations such as reporting. |
| alarm-synchronizer | /usr/local/services/kafka/logs | Records the operation log and error log of Kafka. |
| metric-writer | /usr/local/services/barad-skywalker/log | Records the operation log of Flink. |
| oceanus-ctcw1 | /data1/containers/1529781375000001609/es/logs | Records the operations performed by users on the ManageOne client that impact system security, such as logging in to ManageOne. |
| oceanus-galileo1 | /data/zookeeper/zookeeper.log | Startup information and running information logs of ZooKeeper. |
| oceanus-servant1 | /usr/local/services/barad_sync_server-1.0/log | Log of synchronizing CVM information from CCDB to the translation table. |
| alarm-amp | /usr/local/services/cm_php_script-1.0/log | Log of synchronizing alarm policy information of the primary database to the regional database. |
| alarm-consumption | /usr/local/services/dict_access-1.0/log | Log of the server for reporting dimension translation information for other products. |
| script | /usr/local/services/dict_sync_tool-1.0/log | Log of consuming dimension translation information from TDMQ for Kafka and then storing it in the |

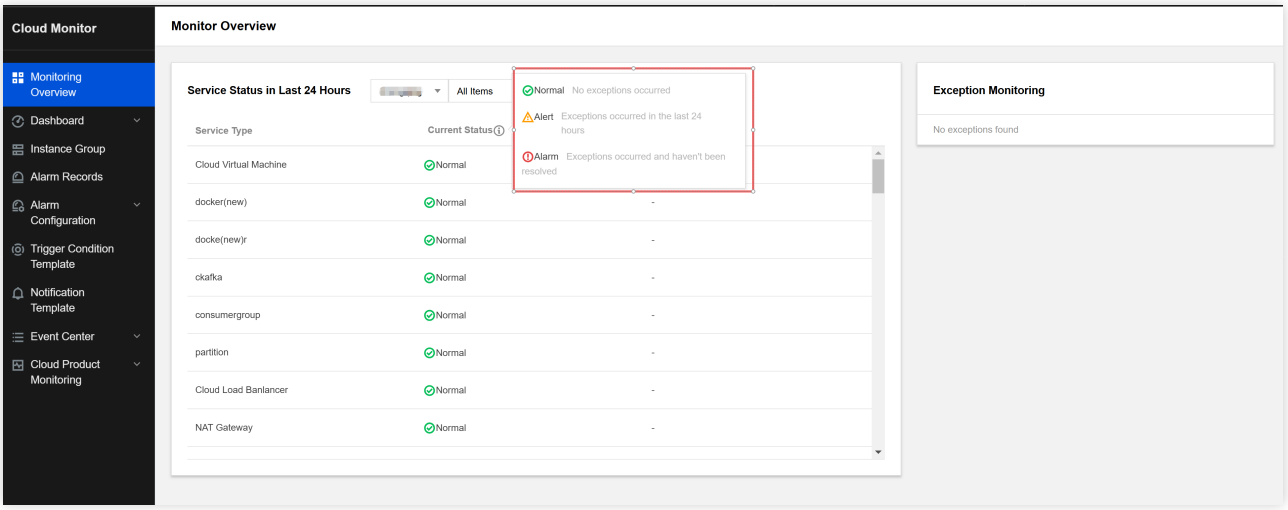
| Module | Log Location | Log Feature |
| --- | --- | --- |
| | | database. |
| script | /usr/local/services/getAppTraffic/log | Log of the table creation status for the scheduled task getAppTraffic. |
| script | /usr/local/services/monitor_cvm_msg_forward-1.0 | Logs of synchronizing instance destruction information of synchronized cvm and deleting the information in the alarm policy bound to the cvm through synchronous invocation API |
| event_handle | /data/log/event_handle-1.0 under the tcloud-barad-event container | Event center log. |
| update2 | /data/www/update2.agent.yun.com/conf/logs | Mainly used to record and update CVM status information as well as distribute BARAD agents. |
| Alarm Management | /usr/local/services/barad-alarm-synch/log under the tcloud-barad-alarm-synchronizer container | Alarm synchronization log. |
| Alarm Management | /usr/local/services/barad-alarm-amp/log under the tcloud-barad-alarm-amp container | Alarm sending log. |
| Alarm Management | /usr/local/services/barad-alarm-detector/log under the tcloud-barad-alarm-detector container | Alarm detection log. |
| api | /data/www/api.barad.yun.com/app/storage/logs under the tcloud-barad-api container | API logs include logs such as request logs and error logs. |
| api-go | /usr/local/services/consoleApi/log | Cloud API logs include logs such as request logs and error logs. |
| query-api | /usr/local/services/queryApi/log | Log of API querying results from ES storage data. |
| nws-tke | /usr/local/services/barad-nws-tke/log | Log of receiving data reported by TKE. |
| monitor-dashboard | /usr/local/services/monitor-dashboard/log | Dashboard query log. |

| Module | Log Location | Log Feature |
| --- | --- | --- |
| manager-api | /usr/local/services/barad-manager-api/log | Log for the middle platform to manage API services. |
| config-center | /usr/local/services/barad-config-center/log | Log that configures services for the monitoring data middle platform. |
| ctsdb-manager | /usr/local/services/ctsdb-manager/log | Log that reads the access configuration in the database and creates metric templates in CTSDB. |
| alarm-amp | /usr/local/services/barad-alarm-amp/log | Log for unifying alarm sending. |
| alarm-consumption | /usr/local/services/consumption/logs | Alarm history query log. |
| alarm-detector | /usr/local/services/barad-alarm-detector/log | Alarm detection log. |
| alarm-policy | /usr/local/services/policy/log | Alarm policy CRUD log. |
| alarm-synchronizer | /usr/local/services/barad-alarm-synchronizer/log | Alarm synchronization component log. |
| metric-writer | /usr/local/services/metric-writer/log | Log that writes Kafka consumption data to ES. |
| oceanus-ctcw1 | /data/logs/tstream_cluster_admin<br>/data/logs/tstream_cluster_master<br>/data/logs/tstream_taskcenter<br>/data/logs/tstream_watchdog | Task execution log. |
| oceanus-galileo1 | /data/logs/tstream_galile<br>/data/logs/tstream_netcenter | External API log. |
| oceanus-servant1 | /data/logs/tstream_servant | Service detection log. |

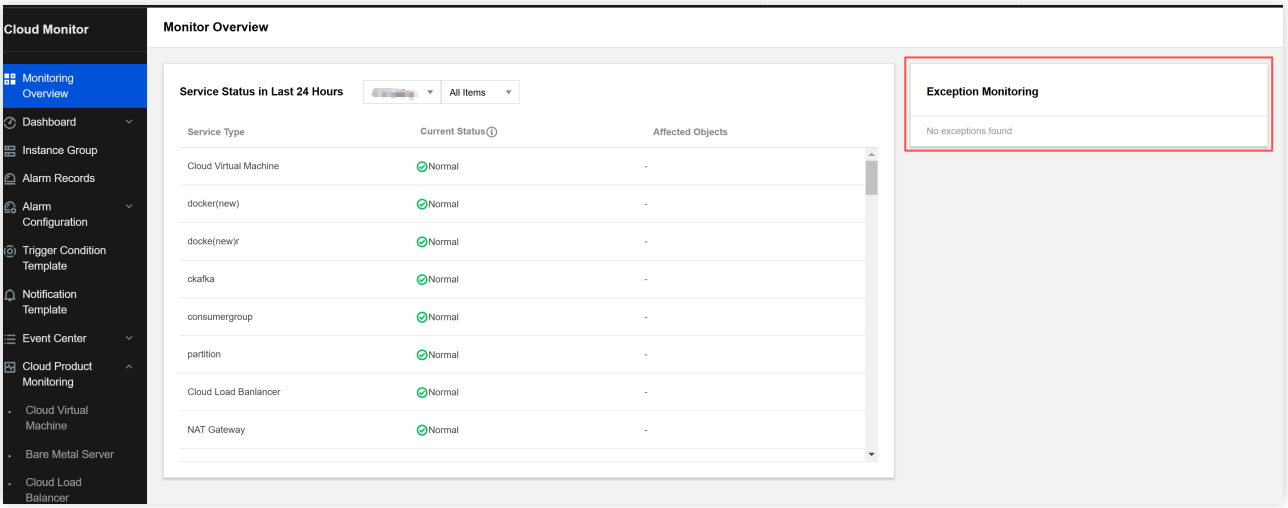
# User Guide
# Getting Started
# Monitoring Overview

The monitoring overview module provides an entry for viewing the overall monitoring status of cloud products. You can view the service health status from the past 24 hours, monitoring timeline from the past 7 days, exception monitoring status, and other information in this module.
Operation steps:

1. Log in to the cloud platform console and click Cloud Monitor to go to the Monitoring Overview page by default.
2. The Monitoring Overview page contains the following four modules.
    i. Service Health Status in the Last 24 Hours
    It displays the current three statuses of cloud resources, including Normal, Reminder, and In Alarm. The number of objects affected by abnormal cloud resources is also displayed.
    - Normal: no resource exceptions.
    - Reminder: Exceptions occurred in resources in the last 24 hours.
    - In Alarm: There are currently resources in an abnormal status that have not been restored.
    You can hover over the abnormal status icon to view the alarm exception records, and select the region where the cloud resource is located in the drop-down list.

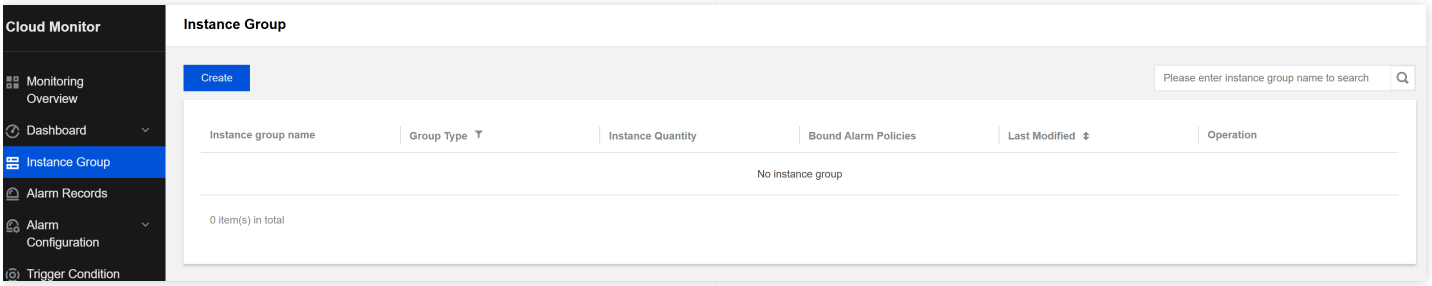



    ii. Exception Monitoring
    It displays the number of policies for which no alarm receiving group is configured under the current account. You can click Configure Reception Group to go to the alarm policy list for configuration.


Cloud Monitor
Monitor Overview
Monitoring Overview
Dashboard
Instance Group
Alarm Records
Alarm Configuration
Trigger Condition Template
Notification Template
Event Center
Cloud Product Monitoring
Cloud Virtual Machine
Bare Metal Server
Cloud Load Balancer
Service Status in Last 24 Hours
All Items
Service Type
Current Status
Affected Objects
Cloud Virtual Machine
Normal
docker(new)
Normal
docke(new)r
Normal
ckafka
Normal
consumergroup
Normal
partition
Normal
Cloud Load Banlancer
Normal
NAT Gateway
Normal
Exception Monitoring
No exceptions found

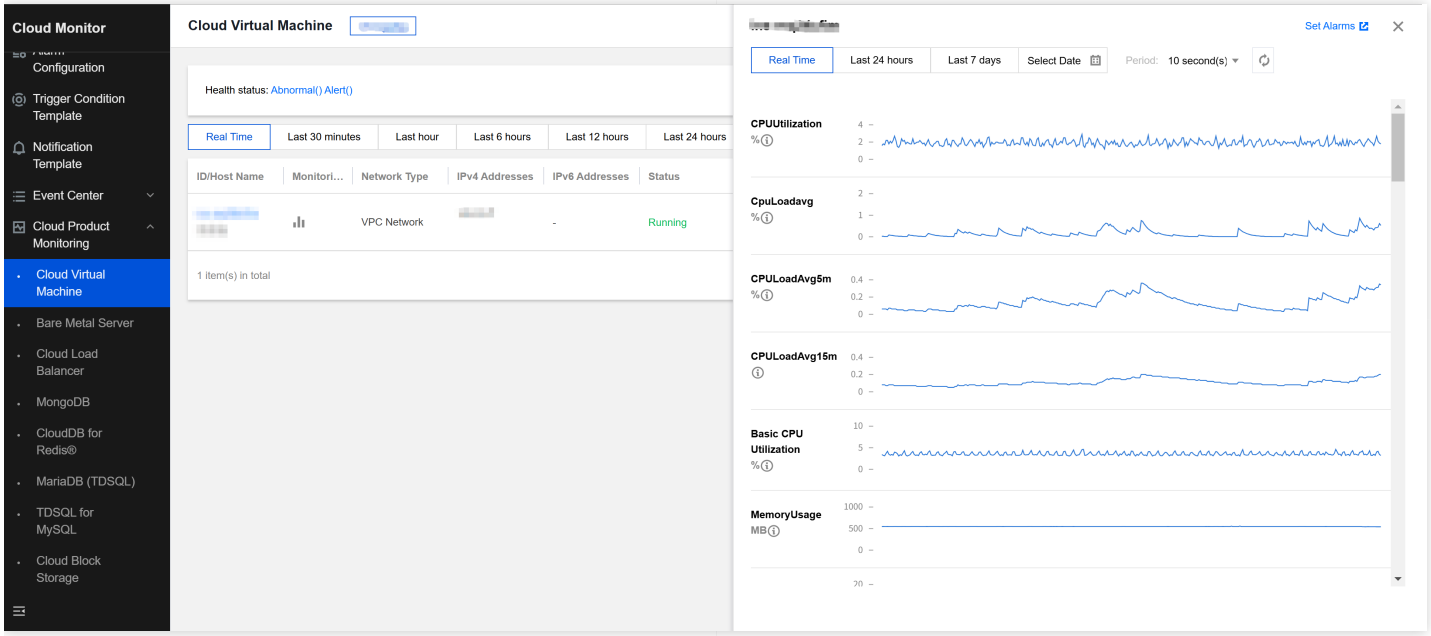
# Instance Grouping

You can group instances within the same cloud service across regions for unified management. When creating a large number of instances, you can create groups as needed, making it easier to manage instances, set alarm policies, and improve operational efficiency.



# Cloud Product Monitoring

Cloud Product Monitoring is a unified platform that collects and displays the most detailed monitoring information in the Cloud Monitor console. In Cloud Product Monitoring, users can view the list of cloud resources under the account, as well as the corresponding monitoring and alarm details. Compared to the monitoring data displayed in the business console, Cloud Product Monitoring integrates the health status and alarm data of instances. It provides users with the capability to quickly locate abnormal instances and supports users in troubleshooting issues by combining specific data and exception information.




Monitoring details are displayed based on the cloud products you have deployed. For cloud products that have not been deployed, monitoring details will not be displayed. Currently, cloud products such as Cloud Virtual Machine (CVM), Cloud Load Balancer (CLB), Cloud MongoDB (MongoDB), CloudDB for Redis® (Redis®), and Cloud Block Storage (CBS) are supported in Cloud Product Monitoring.

# Operation Guide
# Obtaining Monitoring Views
# Metric Data Use Limits

Metrics represent the performance data of user systems and are a core concept of Cloud Monitor. Metric data refers to the specific monitoring data of various cloud services, denoting a time-ordered collection of data points sent to Cloud Monitor. Statistical data about these data points can be retrieved as a time series. Currently, Cloud Monitor offers 1-minute and 5-minute data granularities, and the data storage durations for the two granularities are different.

| Granularity | Storage Duration |
| --- | --- |
| 1min | 30 days |
| 5min | 30 days |

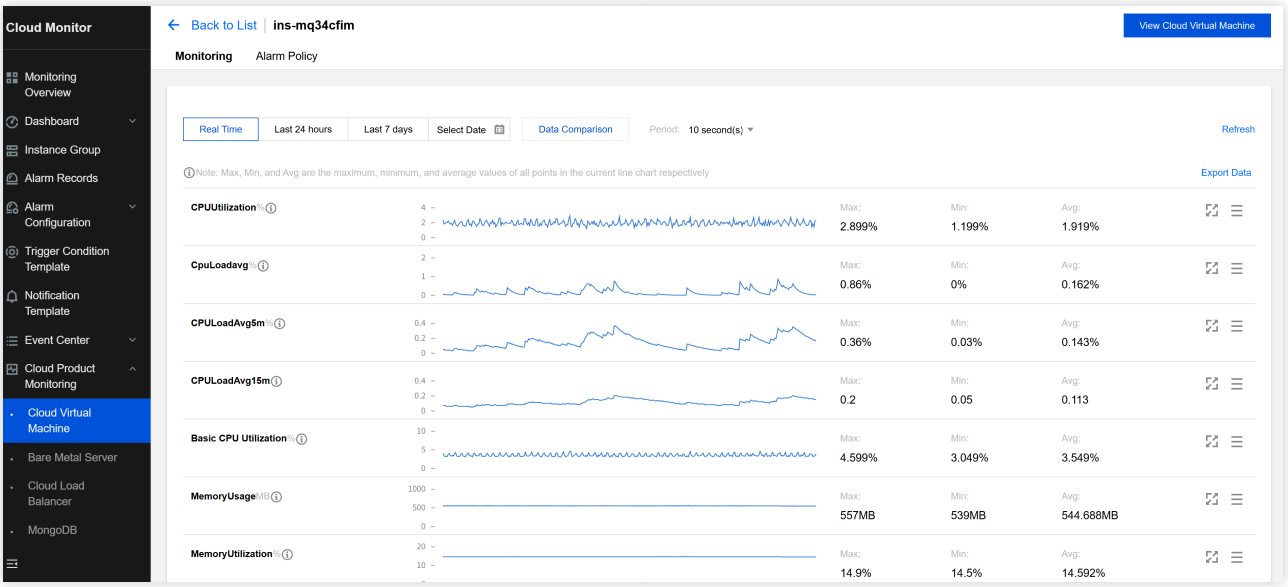
Storage duration refers to the period from when Cloud Monitor collects the monitoring data until the data is terminated after reaching the corresponding period.

> Note:
>
> If a user's cloud resources are reclaimed or terminated, the monitoring data generated by the cloud resources will not be immediately cleared. Cloud Monitor continues to follow a unified storage duration to store and manage the monitoring data of terminated instances.

Users can view or retrieve the monitoring data generated by cloud resources under their accounts through the frontend console or API calls. If the resource object to be viewed has not been reclaimed or terminated, users can view the monitoring data in the frontend console. If the target instance no longer exists, users need to retrieve the corresponding monitoring data through API calls.

# Obtaining Monitoring Views and Reports

1. View monitoring charts.
   Cloud Monitor provides customers with various channels to view monitoring data.
2. Viewing the detailed monitoring data of a single instance.
   i. View through Cloud Monitor > Cloud Product Monitoring.
   Cloud Product Monitoring is a unified platform that collects and displays the most detailed monitoring information in the Cloud Monitor console. In Cloud Product Monitoring, users can view the list of cloud resources under the account, as well as the corresponding monitoring and alarm details.
   Compared to the monitoring data displayed in the business console, Cloud Product Monitoring integrates the health status and alarm data of instances. It provides users with the capability to quickly locate abnormal instances and supports users in troubleshooting issues by combining specific data and exception information.




   ii. View through various cloud product consoles.
   Users can also log in to the console of each cloud service to view the monitoring details of an instance. Using Cloud Virtual Machine (CVM) as an example:
   - Users can quickly browse the performance data of each instance in the CVM list through the monitoring pop-up window.
   - Users can also click a CVM ID in the list to further view the instance's monitoring details.
3. View the monitoring data of multiple cloud services.
   i. View the monitoring data through the dashboard.
   View monitoring data across instances and products via the dashboard. This feature enables users to view data from multiple instances at once.
   - Users can subscribe to their most concerned monitoring metrics through the dashboard and monitor the performance status of core instances through real-time monitoring or large-screen display of dashboard panels.
   - Users can subscribe to the overall performance status of a workload or cluster through the dashboard. This allows them to intuitively understand the overall resource situation without the inefficiency of checking the

monitoring data for each resource individually.

For specific usage of the dashboard, see Creating a Dashboard.

4. Obtain data reports.

   i. Export data from the dashboard.

   Log in to the cloud platform console and choose Cloud Monitor > Dashboard to enter the dashboard list. Click the corresponding dashboard. On the dashboard page, locate the monitoring item for which you want to export data, and click ... above it to export the corresponding instance monitoring details list.

# Obtaining Monitoring Data for Specific Metrics

Cloud Monitor provides the following methods to support users in obtaining the specific metric data of a monitoring object at a specific time.

1. Obtain the specific metric data of a monitoring object at a specific time from the console.
    i. Log in to the cloud platform console and navigate to the corresponding product console where you want to view monitoring data.
    ii. Select the specific object you want to monitor and click the object ID to enter the monitoring details page. Alternatively, click the monitoring icon in the object list to view the data through the monitoring pop-up window.
    iii. Locate the target metric on the monitoring details page or in the pop-up window.
    iv. Use the time selector and granularity selector to adjust the view and display the specific content of the data. This allows you to see the specific metric data of the monitoring object at a specific time.
2. Obtain the specific metric data of a monitoring object at a specific time through an API.
   See Query Monitoring Metrics: GetMonitorData API Call Usage Guide.

# Alarm Service
# Alarm Overview

When the status of certain products changes, you can create alarms to promptly notify you to take action. Alarms monitor specific metrics over a certain period and send alarms at configured intervals based on configured thresholds.
An alarm consists of the following components:

- Alarm trigger condition (under what conditions will the alarm be sent)
- Alarm object (which object will send an alarm)
- Alarm recipient group (who receives the alarm).
- Alarm receiving method (how the alarm is received)
  This document describes how to create alarms for one or more objects, and select the objects to receive alarms.

1. Basic Concepts

| Term | Definition |
| --- | --- |
| Alarm policy type | Policy type identifies policy category and corresponds to specific cloud products, with a distinct layer of granularity under cloud products. For example, when you choose Cloud Virtual Machine > Basic Monitoring policy, you can customize alarm metrics such as CPU utilization and disk utilization; when you choose Cloud Virtual Machine > Storage Monitoring policy, you can customize alarm metrics such as disk I/O wait time and disk read traffic. |
| Alarm rule | Alarm rules correspond to the trigger conditions that specific monitoring metrics should satisfy. For example, when disk utilization exceeds 80% per minute and lasts for 3 minutes, an alarm is triggered once every hour. |
| Relationship between policy types and alarm rules | An alarm type contains multiple alarm rules and is a collection of a series of alarm rules. For example, when you choose Cloud Virtual Machine > Basic Monitoring policy, the policy includes disk utilization alarms, ping unreachable alarms, CPU utilization alarms, and more. |

2. Alarm statuses

| Alarm Status | Description |
| --- | --- |
| Unresolved | Alarms that have not been handled or are being handled, but the resources are still in an abnormal status. |

| Alarm Status | Description |
| --- | --- |
| Resolved | Alarms have been resolved. |
| Insufficient Data | - The alarm policy that triggered an alarm has been deleted.<br>- The CVM has been migrated from one project to another.<br>- No data is reported because the Cloud Monitor control has not been installed or has been uninstalled. |
| Expired | - The alarm policy has changed.<br>- The most recent alarm time has not been updated for more than 24 hours. |

# Alarm Policies
# Creating Alarm Policies

When users need to set alarms for a specific product status, they need to first create an alarm policy. The alarm policy consists of six required components: policy name, monitoring type, policy type, alarm object, trigger condition, and notification template. You can create an alarm policy according to the following instructions:

1. Log in to the Cloud Platform console and choose Cloud Monitor > Alarm Configuration > Alarm Policy.

2. Click Create and configure the alarm according to the following instructions.

| Configuration Type | Configuration Item | Description |
|---|---|---|
| Basic information | Policy name | Enter a custom policy name. |
| | Remarks | Enter custom policy remarks. |
| | Monitoring type | Only cloud service monitoring is supported. |
| | Policy type | Choose the cloud product policy type you need to monitor, for example, Cloud Virtual Machine > Basic Monitoring. |
| Alarm rule configuration | Alarm object | ◦ If you select an instance ID, the alarm policy will be associated with the selected instance.<br>◦ If you select an instance group, the alarm policy will be associated with the selected instance group.<br>◦ If you select all objects, the alarm policy will be associated with all instances under the current account. |
| | Manual configuration (metric alarm) | Alarm trigger condition: It is a semantically meaningful condition composed of metrics, comparison relation, thresholds, statistical granularity, and continuous N monitoring data points. You can set up alarm thresholds based on the trend of metric changes in the chart. For example, when the metric is CPU utilization, the comparison relation is >, the threshold is 80%, the statistical granularity is 5 minutes, and the number of continuous monitoring data points is 2, the condition means to collect CPU utilization data every 5 minutes, and |

| Configuration Type | Configuration Item | Description |
| --- | --- | --- |
| | | if the CPU utilization of a CVM is greater than 80% for two consecutive times, an alarm will be triggered.<br>Alarm frequency: You can set a repeated notification policy for each alarm rule. This way, an alarm notification will be sent repeatedly at a specified frequency when an alarm is triggered.<br>Options: No repetition, 5 minutes, 10 minutes, periodic exponential increasing interval, and other repetition frequencies.<br>An exponentially increasing interval means that alarm information is sent to you when an alarm is triggered the first time, second time, fourth time, eighth time, ... and 2 raised to the power of N times. In other words, the alarm information will be sent less and less frequently as time goes on to reduce the disturbance caused by repeated alarms.<br>Default logic for repeated alarms: An alarm notification will be sent to you at the configured frequency within 24 hours after an alarm is triggered. After 24 hours, the alarm notification will be sent once every day. |
| | Manual configuration (event alarm) | When an exception occurs in cloud product resources or underlying infrastructure services, you can create an event alarm to promptly notify you to take action. For details, see the Event Center document. |
| | Template selection | Click the template selection button, and select the configured template from the dropdown list. For specific configurations, see the Trigger Condition Template document. If the created template is not displayed, you can click Refresh on the right side of the dropdown list to refresh the template selection list. |
| Alarm notification configuration | Alarm notification | Supports choosing a system preset notification template or a user-defined notification template. Each alarm policy can be associated with up to three notification templates. For details, see the Notification Template document. |

3. After configuring the above information, click Complete to create the alarm policy.

# Trigger Condition Templates

When the same alarm trigger conditions are required for different alarm policies, users can configure an alarm trigger condition template, which can be reused with one click when an alarm policy is created, avoiding repeated configurations. For alarm policies using the alarm trigger condition template, users can modify the template trigger conditions, and the changes will automatically apply to the corresponding alarm policies, improving the Ops efficiency.

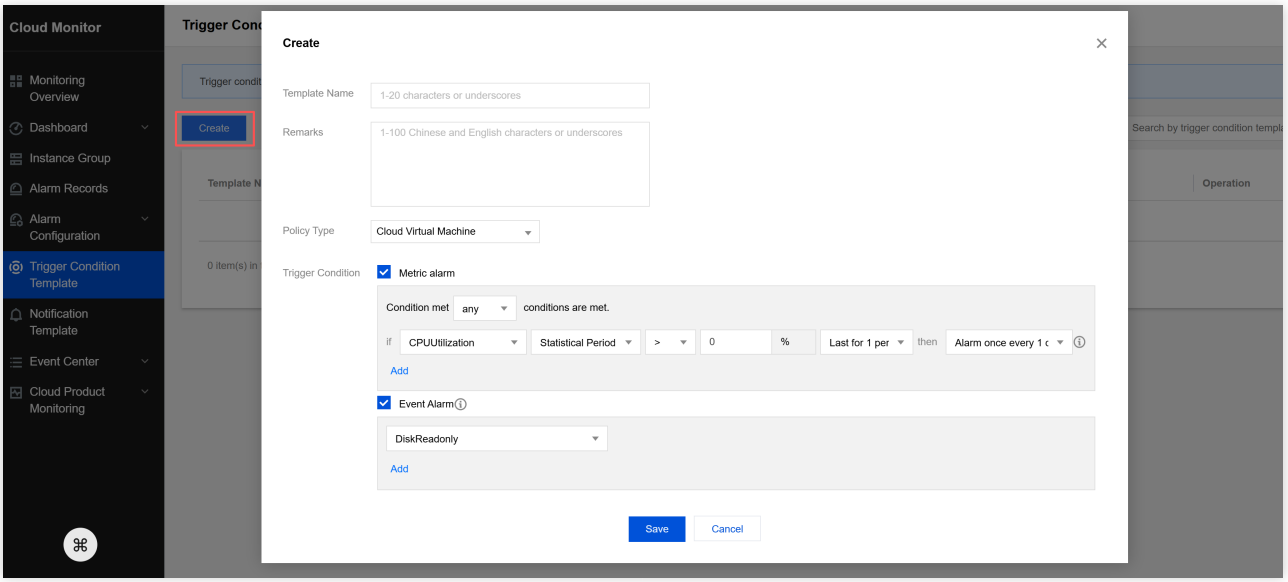
> Note:
>
> One trigger condition template is used to set up trigger condition rules for one cloud product.

1. Create a trigger condition template.

   i. Log in to the cloud platform console and choose Cloud Monitor > Trigger Condition Template.

   ii. Click Create, and configure the trigger conditions in the pop-up window as follows.

   - Template name: Enter a template name.
   - Note: Enter a note for the template.
   - Policy type: Choose the service to be monitored, such as Cloud Virtual Machine > Basic Monitoring.
   - Trigger condition: It is divided into metric alarm and event alarm. Below it, click Add to set up multiple alarm items.




   iii. After completing the configuration, click Save.

2. Edit a trigger condition template.

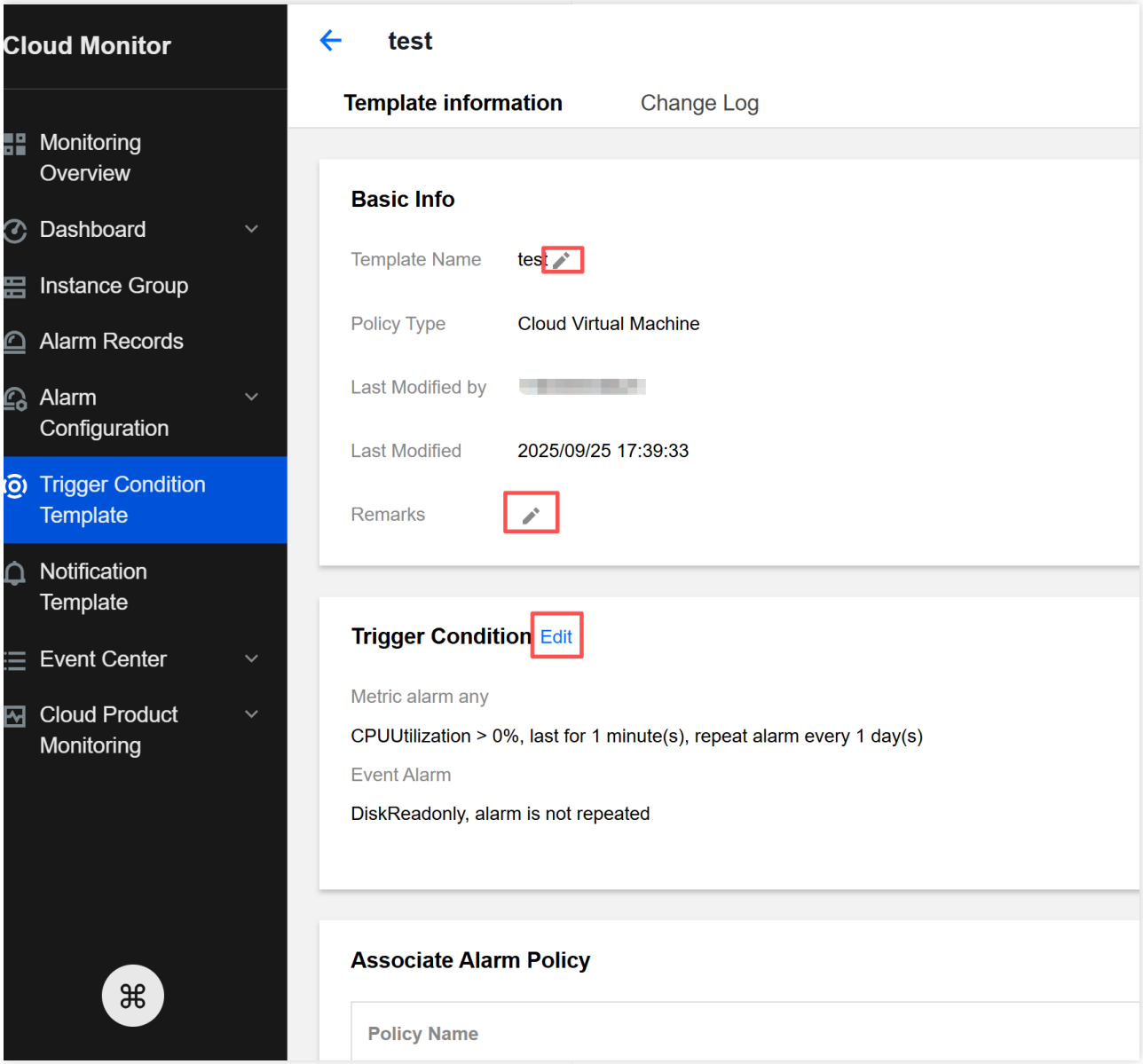
> Note:
>
> When a trigger condition template is already associated with alarm policies, if the trigger condition template is edited, the modification will be applied to all associated alarm policies.

   i. Log in to the cloud platform console and choose Cloud Monitor > Trigger Condition Template.

   ii. In the template list, click the name of the template that needs to be edited to go to the template details

page.

iii. In the Template Information Tab, you can click different buttons to edit and modify different information items.

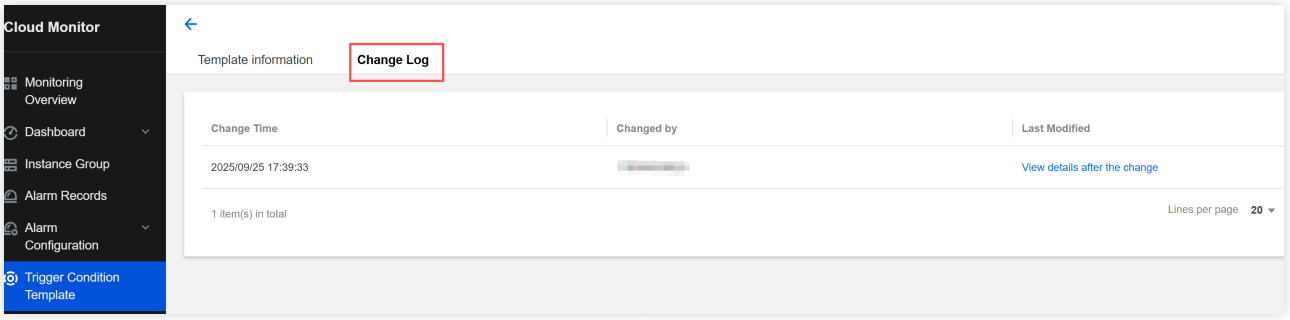



3. View the change records of the trigger condition template.

1. Log in to the cloud platform console and choose Cloud Monitor > Trigger Condition Template.

2. In the template list, click the name of the template for which you need to view the change records to go to the template details page.

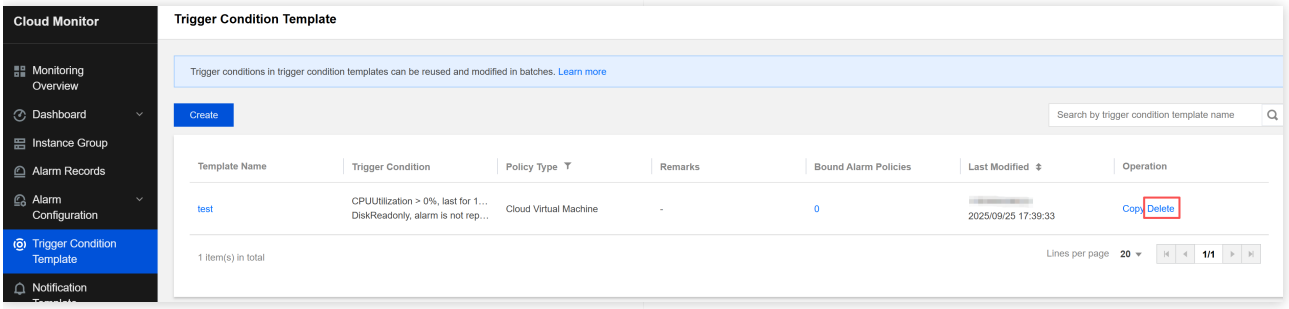
3. In the Change Log tab, you can view the change time, updater, and change result of historical change operations.




4. Delete a trigger condition template.

> Note:
>
> When the trigger condition template is already associated with alarm policies, deleting the trigger condition template will cause the alarm policies associated with the template to become invalid.

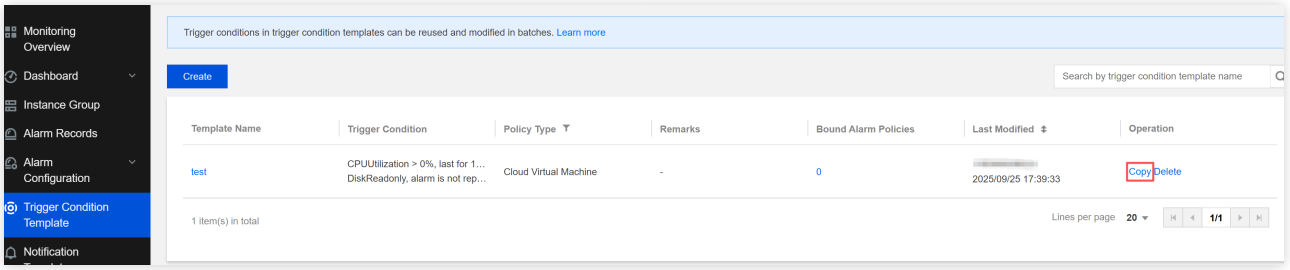
i. Log in to the cloud platform console and choose Cloud Monitor > Trigger Condition Template.
ii. Locate the template you need to modify in the template list, and click Delete in the operation column on the right to delete the template.




5. Copy a trigger condition template.

> Note:
>
> Copying a trigger condition template means that only the trigger condition rules of the template will be copied, while the association between the template and alarm policies will not be copied.

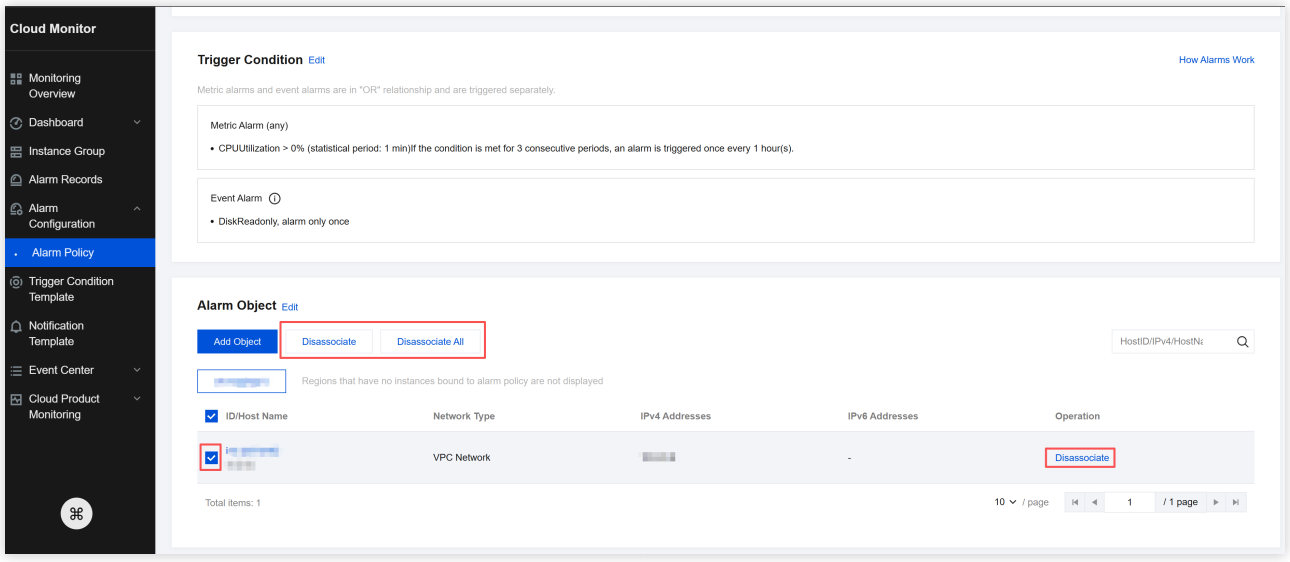
1. Log in to the cloud platform console and choose Cloud Monitor > Trigger Condition Template.
2. On the trigger condition template list page, locate the template you need to copy, and click Copy in the operation column on the right.
3. In the pop-up dialog box, click Confirm Copy to copy the trigger condition template.


Monitoring Overview
Dashboard
Instance Group
Alarm Records
Alarm Configuration
Trigger Condition Template
Notification
Trigger conditions in trigger condition templates can be reused and modified in batches. Learn more
Create
Search by trigger condition template name
Template Name
Trigger Condition
Policy Type
Remarks
Bound Alarm Policies
Last Modified
Operation
test
CPUUtilization > 0%, last for 1…
DiskReadonly, alarm is not rep…
Cloud Virtual Machine
-
0
2025/09/25 17:39:33
Copy Delete
1 item(s) in total
Lines per page 20
1/1

# Associating and Dissociating Objects

After creating an alarm policy, you need to associate some alarm objects with it to indicate which instance objects will trigger an alarm when they meet the alarm trigger conditions.

1. Add an associated object.
    i. Log in to the Cloud Platform console and choose Cloud Monitor > Alarm Configuration > Alarm Policy.
    ii. In the alarm policy list, click on the name of the alarm policy you want to associate objects with to enter the policy details page.
    iii. In the Alarm Object module, click Edit.
    iv. In the pop-up box for alarm object editing, users can select instance ID, instance group, or all object types, and then choose the instance objects or instance groups available for that type accordingly.
    v. Click OK to establish the association between the alarm policy and the alarm object.
2. Remove the association.
    i. Log in to the Cloud Platform console and choose Cloud Monitor > Alarm Configuration > Alarm Policy.
    ii. In the alarm policy list, click on the name of the alarm policy you want to associate objects with to enter the policy details page.
    iii. In the Alarm Object module:
    - If the alarm policy is already associated with an alarm object, click Remove after selecting the instances for which you want to remove the association, and then click Remove in the pop-up confirmation dialog box. If you need to remove the association between the alarm policy and all instances, click Remove All, and then click Remove All in the pop-up confirmation dialog box to remove the association between the selected instances and the alarm policy.

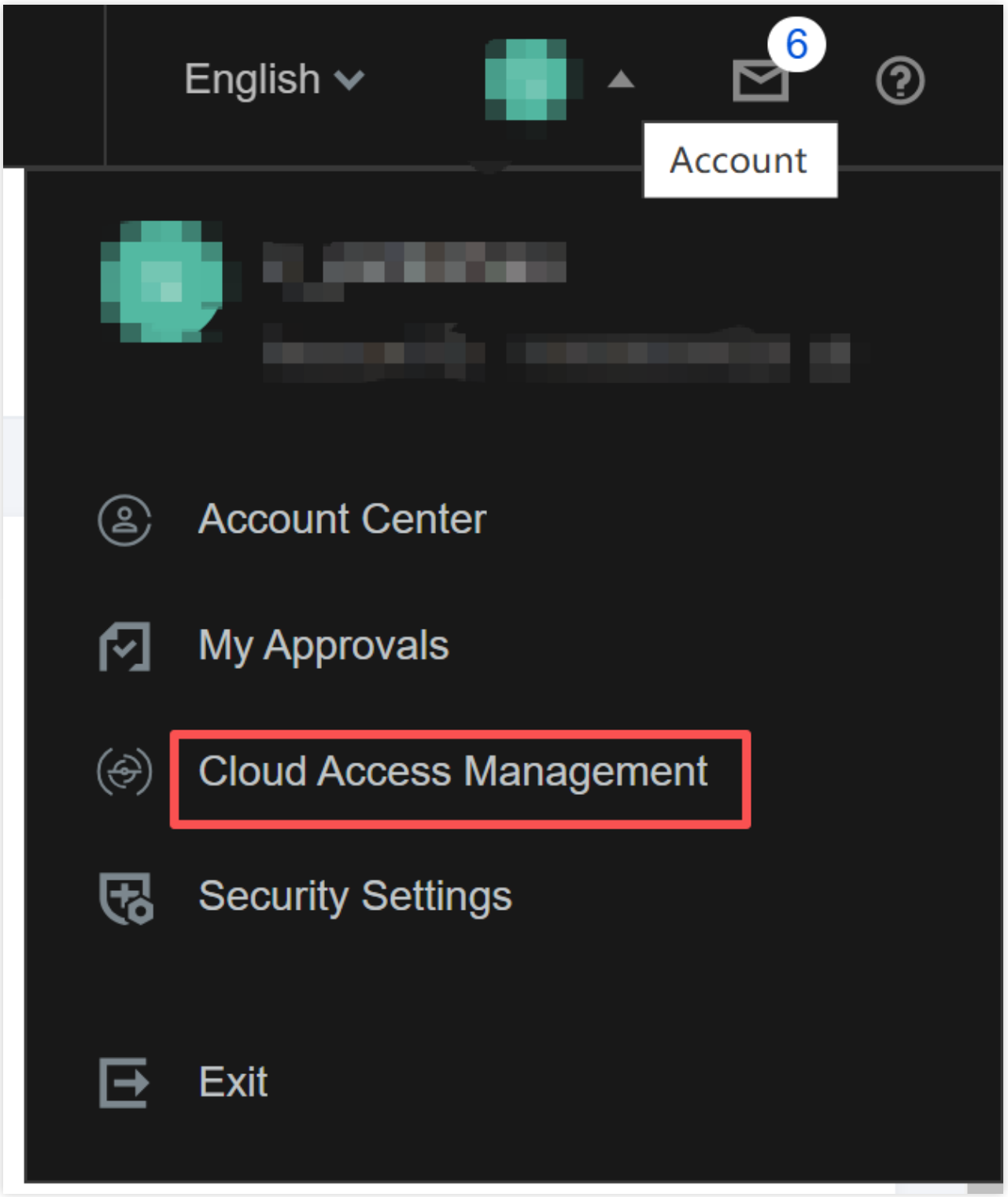



    - If the alarm policy is already associated with an instance group, click Unbind on the right side of the instance group for which you want to remove the association. In the pop-up confirmation dialog box, click Confirm Unbind to remove its association with the alarm policy.

# Alarm Notifications
# Creating and Managing Alarm Recipient Groups

Alarm recipient groups determine who can receive alarm notifications. You can group people concerned with the same alarms into a group, so when an alarm is triggered, everyone in the group receives the corresponding alarm notification.

1. Create an alarm recipient group.
    i. Log in to the cloud platform console, move the mouse pointer over your account avatar, and click Access Management.




ii. Choose User Management > User Group.

iii. Click Create New User Group, fill in the user group name and remarks, and then click Submit to create it successfully.

iv. In the operation column on the right side of the created user group, click Add User. Alternatively, click the user group name to go to the user group details page, choose the Added Users tab, and click Add User.

v. In the pop-up Add User window, select the users you want to add to the user group, and then click

OK.

2. You can also manage and create alarm recipient groups when configuring the alarm notification template.

   i. Log in to the Cloud Platform console and choose Cloud Monitor > Alarm Configuration > Alarm Policy.
   ii. In the alarm policy list, click the name of the alarm policy you want to associate objects with to enter the Details page.
   iii. In the Alarm Notification module, click Create New Template.
   iv. In the pop-up Create Notification Template box, when a user group is selected as the recipient, click Add User Group to redirect to the CAM console to configure user groups.

3. Manage an alarm recipient group.

   i. Log in to the cloud platform console, move the mouse pointer over your account avatar, and click Access Management.
   ii. Choose User Management > User Group.
   iii. If you want to delete a user group, locate it in the user group list, click Delete in the operation column on the right, and then click Confirm to Delete.
   iv. If you want to manage user members in a user group, click the name of the user group you want to manage to go to the user group details page, and choose the Added Users tab.
   v. If you want to add a user, click Add User. If you want to remove a user, select the user you want to remove, and click Remove User or Remove from Group.

# Alarm Callback

The callback API allows your system to directly receive alarm notifications from the cloud platform, providing the capability to push alarm information via an HTTP POST request to a URL accessible over the public network. Users can take further actions based on the alarm information pushed through the callback API.

1. Usage method
    - Callback API: Users need to provide a URL that is capable of receiving HTTP POST requests and is accessible over the public network as the callback address.
    - Callback trigger: The trigger logic is similar to that of alarm SMS and emails. When an alarm policy created by the user is triggered or recovered, alarm messages will be sent through the callback API. The callback API also supports repeated alarms.
    - Association with API callbacks: Users can configure API callbacks during the API Callback step of creating an alarm, or add API callbacks on the alarm policy details page. A single alarm policy group can only be associated with one alarm callback URL.
    - Response content: After sending alarm information to the user's bound URL, we need to receive the following response content to confirm that the user has successfully received the information.

      > Note:
      >
      > sessionId is used to authenticate the callback request.
      >
      > retCode is used to determine whether the request has been successfully sent.

```
{
  sessionId: "xxxxxxxx",
  retCode: 0
}
```

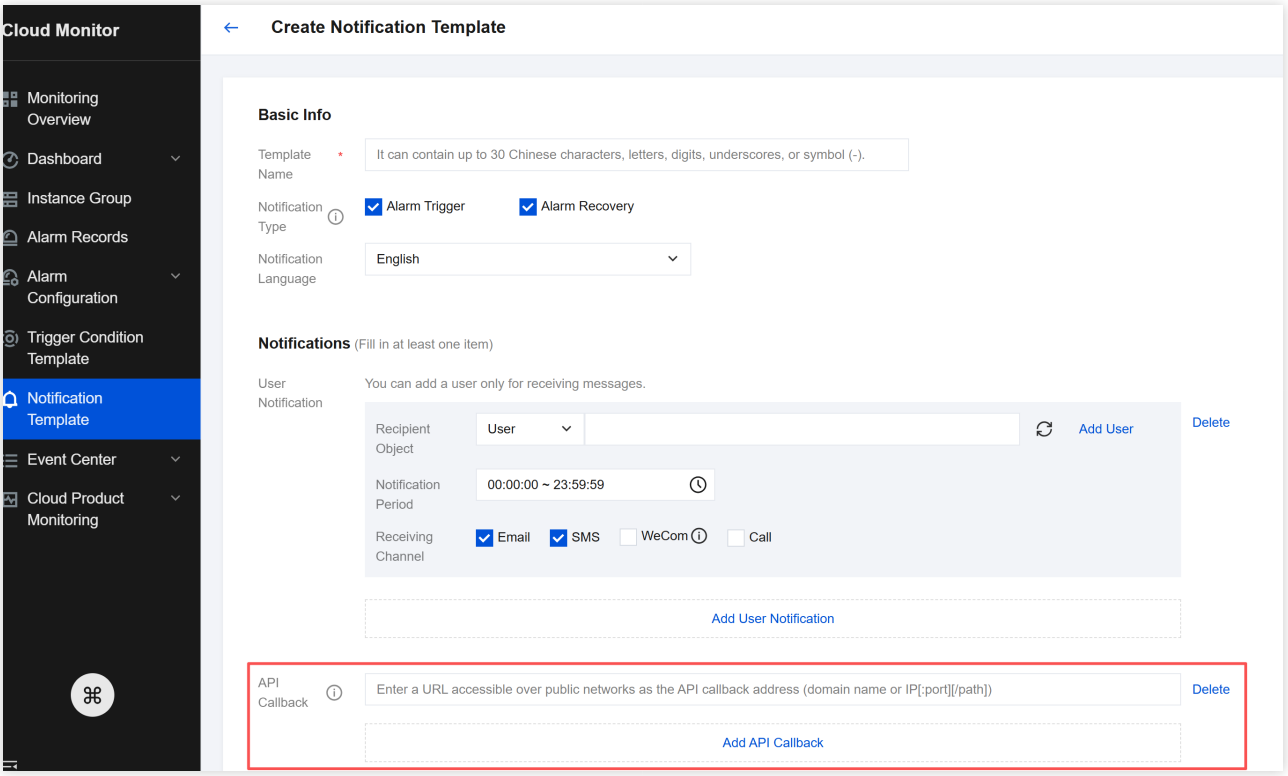
2. Callback parameters
   The callback API sends data in JSON format through HTTP POST requests. The parameters are as follows:

```
{
    "sessionId": "xxxxxxxx",
    "alarmStatus": 1,
    "alarmType": "metric",          // Alarm type (metric or event).
    "alarmObjInfo": {
```

```
        "region": "gz",             // This field will not be returned for products that are not region-specifi
c.
        "namespace": "qce/cvm",     // Product namespace.
        "dimensions": {             // Content in the dimensions field varies by product.
           "unInstanceId": "ins-o9p3rg3m",
           "objId":"xxxxxxxxxxxx",
        }
    }
    "alarmPolicyInfo": {
           "policyId": "policy-n4exeh88",   // ID of the alarm policy group.
           "policyType": "cvm_device",      // Alarm policy type.
           "policyName": "test",            // Name of the alarm policy group.
           "conditions": {
              "metricName": "cpu usage",         // Metric name.
              "metricShowName": "CPU utilization",     // Displayed metric name.
              "calcType": ">>",             // This field will not be returned for metrics without a threshold.
              "calcValue": "90",            // This field will not be returned for metrics without a threshold.
              "currentValue": "100",        // This field will not be returned for metrics without a threshold.
              "unit": "%",                  // This field will not be returned for metrics without a threshold.
              "period": "60",               // This field will not be returned for metrics without a threshold.
              "periodNum": "1",             // This field will not be returned for metrics without a threshold.
              "alarmNotifyType": "continuousAlarm",    // Whether repeated alarms are supported. This f
ield will not be returned for metrics without a threshold.
              "alarmNotifyPeriod": 300                 // Frequency of repeated alarms. This field will not be r
eturned for metrics without a threshold.
           }
           "firstOccurTime": "2017-03-09 07:00:00",     // Time when the alarm is triggered for the first ti
me.
           "durationTime": 500,       // Alarm duration in seconds.
           "recoverTime": "0",        // Time when the alarm is resolved in seconds. If it is not resolved, th
e value is 0.
           "alarmLevel": "",          // Alarm level.
           "appID": 1255000001,
           "uin": 100001100000
    }
}
```

# Receiving Alarm Notifications Through a WeCom Group

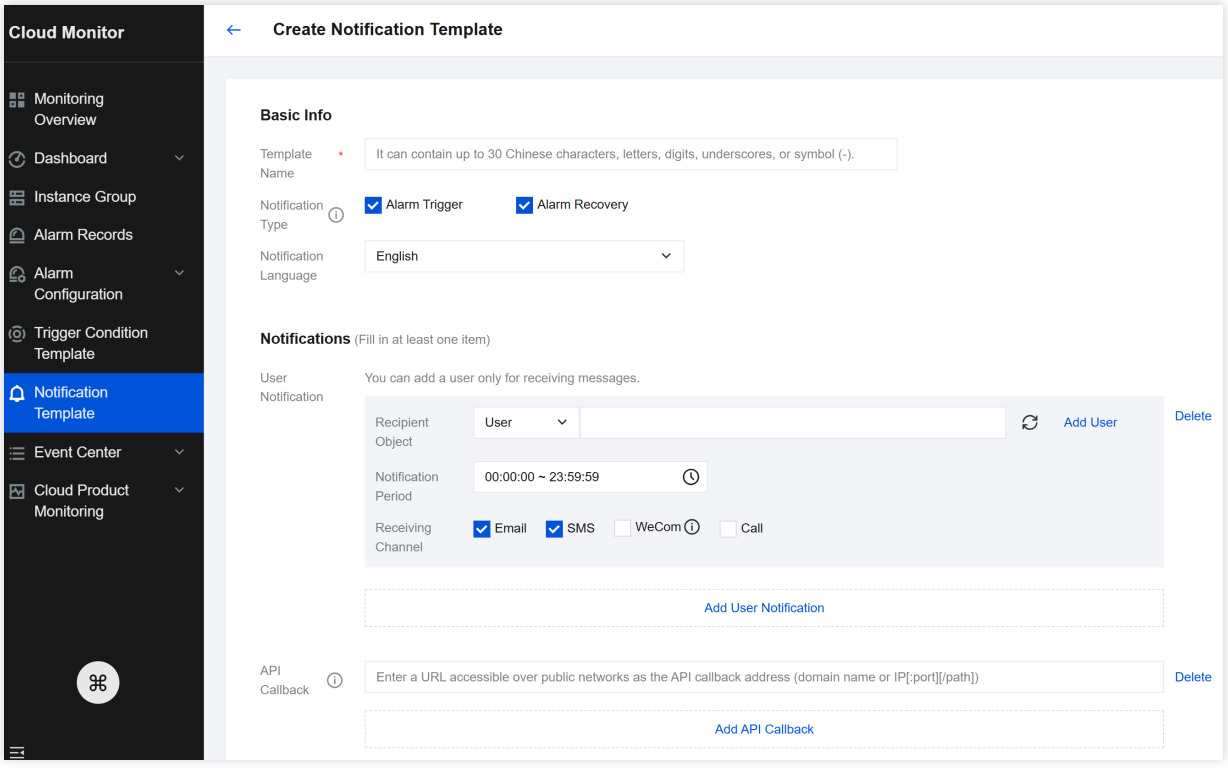
1. Limitations
   Regarding sending WeCom group messages, the number of messages sent by each bot cannot exceed 20 per minute. If you have many alarm policies, it is recommended to create multiple bots and associate alarm policies with different bots. Otherwise, multiple alarm policies may trigger alarms simultaneously, and you may fail to receive some alarm notifications as a result.
2. Step 1: Add a bot on WeCom.
   i. WebCom for PC
      On WeCom for PC, find the target WeCom group for receiving alarm notifications.
      Select and right-click the WeCom group. In the pop-up box, click Add Group Robot.
      In the pop-up box, click Create a New Robot.
      In the pop-up box, enter a custom bot name and click Add Robot.
      Copy the webhook URL and configure the API callback by following Step 2.
   ii. WebCom for Web
      On WeCom for Web, open the target WeCom group for receiving alarm notifications.
      Click the group settings icon in the upper-right corner.
      On the Group Settings page, choose Group Bot > Add Robot.
      On the Add Bot page, enter a custom name for the new bot.
      Click Add, copy the webhook URL, and configure the API callback by following Step 2.
      WeCom alarm messages are in markdown format.
3. Step 2: Configure the alarm API callback.
   i. Log in to the cloud platform console and choose Cloud Monitor > Notification Template.
   ii. Click Create to enter the Create Notification Template page.
   iii. After configuring the basic information, paste the copied webhook URL in the API callback field.




iv. On the Alarm Policy List page, click the name of the alarm policy to be associated with an alarm callback to enter the Manage Alarm Policy page. Associate a notification template on the Alarm Policy page. After successful configuration, when an alarm policy is triggered or recovered, you can receive alarm notifications sent by the group bot in the WeCom group.

# Notification Templates

1. Create a notification template.
   i. Log in to the cloud platform console and choose Cloud Monitor > Notification Template.
   ii. Click Create and enter relevant information on the Create Notification Template page.
   - Template name: Enter a custom template name.
   - Notification type:
     - Alarm trigger: A notification will be sent when an alarm is triggered.
     - Alarm recovery: A notification will be sent when an alarm is resolved.
   - User notification:
     - Recipient: Users or user groups can be selected. To create an alarm recipient group, see Creating and Managing Alarm Recipient Groups.
     - Notification period: Define the time period for receiving alarms.
     - Receiving channel: Three alarm channels are supported: email, SMS, and WeCom. You can modify users' phone numbers, email addresses, and WeCom information in Access Management to determine how users receive alarm notifications.
     - API callback: Enter a publicly accessible URL as the callback API address. You can configure up to 3 alarm callback addresses. Cloud Monitor will promptly push alarm notifications to this address. If an HTTP status of 200 is returned, the verification is successful. For descriptions of alarm callback fields, see Alarm Callback.




   - After you save the callback URL, the system will automatically verify your URL once. The timeout threshold for this verification is 5 seconds. When an alarm policy created by the user is triggered or recovered, alarm messages will be pushed through API callbacks. An alarm message can be pushed up to three times, and the timeout threshold for each request is 5 seconds.
   - When an alarm policy created by the user is triggered or recovered, alarm messages will be pushed through API callbacks. API callbacks also support repeated alarms.

- The outbound IP address for the Cloud Monitor callback API is dynamically and randomly assigned, so specific IP information cannot be provided to you. However, the IP port is fixed at 80. It is recommended that you configure an allow-all policy for port 80 on the security group.

iii. Click Complete to finish creating the notification template.

2. Use the default notification template.

   The system automatically creates a default notification template for you as detailed below:

| Feature | Default Configuration |
| --- | --- |
| Template name | System preset notification template |
| Notification type | Alarm trigger and alarm recovery |
| Alarm recipient | Root account administrator |
| Notification time period | 00:00:00–23:59:59 (all day) |
| Receiving channel | Email and SMS |

# Custom Callback Content Templates

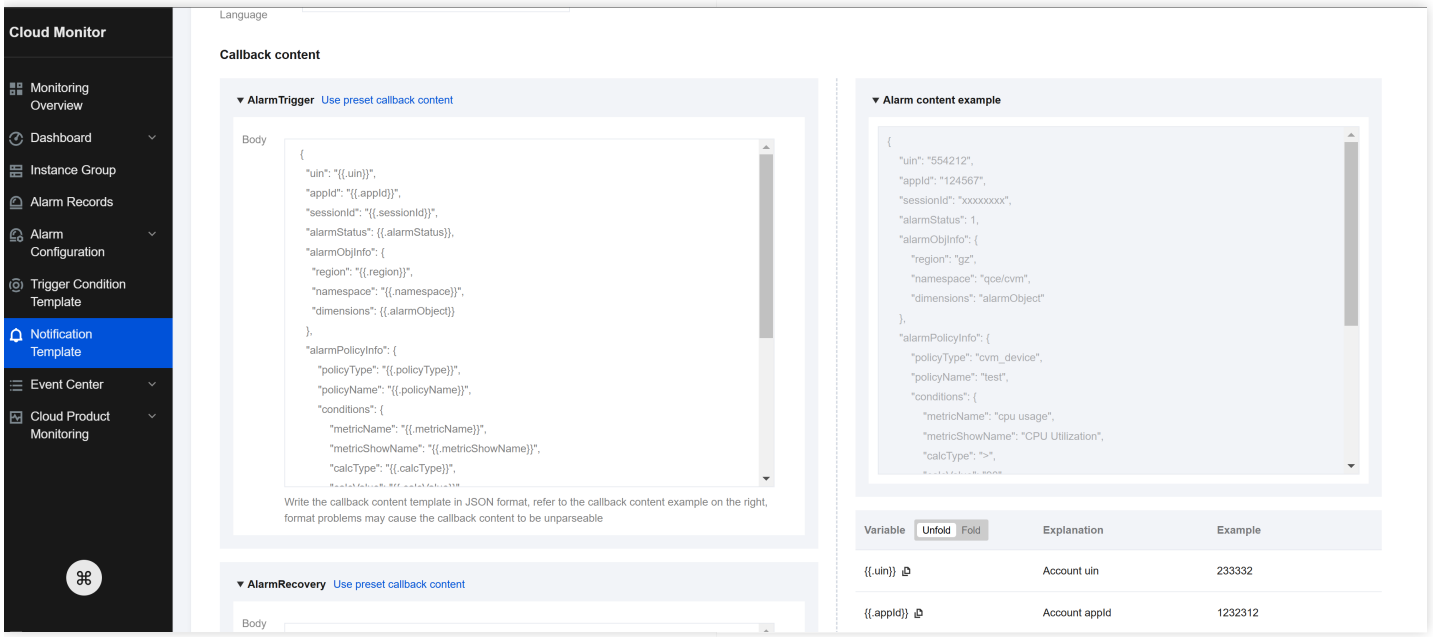
## Scenarios

Through a custom callback content template, you can customize the content and field mapping of the callback message body. This document describes how to create a custom callback content template.

## Operation Steps

1. Log in to the cloud platform console, choose Cloud Monitor > Notification Template, and select the Callback Content Template tab.
2. Click Create Callback Content Template, and configure the content and field mapping of the callback message body on the page.

> Note:
>
> Write the callback content template in JSON format. For details, see the callback content example on the right side of the page. Format issues may cause the callback content to fail to parse.
>
> i. Users can click Use the Preset Callback Content to automatically fill in the preset callback content and perform customization based on the preset callback content.
> ii. The list on the right of the callback content shows the currently supported variable placeholders. Users can directly click to copy and use them.




3. Click OK to complete the creation of the callback content template.
4. When you choose Create Alarm Policy > Configure Alarm Notifications, if the notification template includes a callback method, you can select a custom callback content template. After the configuration is completed, the callback content will be sent based on the template.

> Note:
>
> Users can only select a callback content template with the same language as the notification template.

# Variable Description

Currently, only the following variables can be used as placeholders for custom notification content. If an invalid variable is used, the field of the callback content may have no data in it.

| Variable | Description | Example |
| --- | --- | --- |
| {{.uin}} | Account UIN. | 233332 |
| {{.appId}} | Account AppID. | 1232312 |
| {{.sessionId}} | Alarm sessionId. | LxePI1H4oYQCeg9BN1WZ74JG |
| {{.alarmStatus}} | Alarm status. | 0: alarm recovered; 1: alarm triggered; -: no status notification. |
| {{.region}} | Region. | City A. |
| {{.namespace}} | Namespace. | qce/cvm |

| | | |
|---|---|---|
| {{.alarmObject}} | Cloud product dimension information. | {"unInstanceId": "ins-o9p3rg3m", "objId": "xxxxxxxxxxxx"} |
| {{.policyType}} | Alarm policy type. | cvm_device |
| {{.policyName}} | Alarm policy name. | CVM alarm. |
| {{.metricName}} | Metric name. | cpu usage |
| {{.metricShowName}} | Metric display name. | CPU utilization. |
| {{.calcType}} | Calculation type. | > |
| {{.calcValue}} | Threshold. | 90 |
| {{.currentValue}} | Current value. | 92 |
| {{.unit}} | Unit. | % |
| {{.period}} | Statistical granularity (unit: s). | 5 |
| {{.periodNum}} | Number of lasted data points. | 5 |
| {{.alarmNotifyType}} | Whether repeated alarms are supported. singleAlarm: single alarm; exponentialAlarm: exponential period alarm; continuousAlarm: continuous alarm. For metrics without thresholds, this field is not returned. | singleAlarm |
| {{.alarmNotifyPeriod}} | Alarm notification frequency (unit: s). | 300 |
| {{.firstOccurTime}} | First occurrence time. | 2024-07-09 14:35:00 (UTC+08:00) |
| {{.durationTime}} | Alarm duration (unit: s). | 500 |
| {{.recoverTime}} | Recovery time (0 when not recovered). | 0 |

# Viewing Alarm Records

Cloud Monitor provides you with the alarm records feature, allowing you to trace and view alarm records from the last six months. You can also quickly subscribe to alarm policies on the Alarm Records page.

1. View alarm records.

   i. Log in to the cloud platform console, and choose Cloud Monitor > Alarm Records.

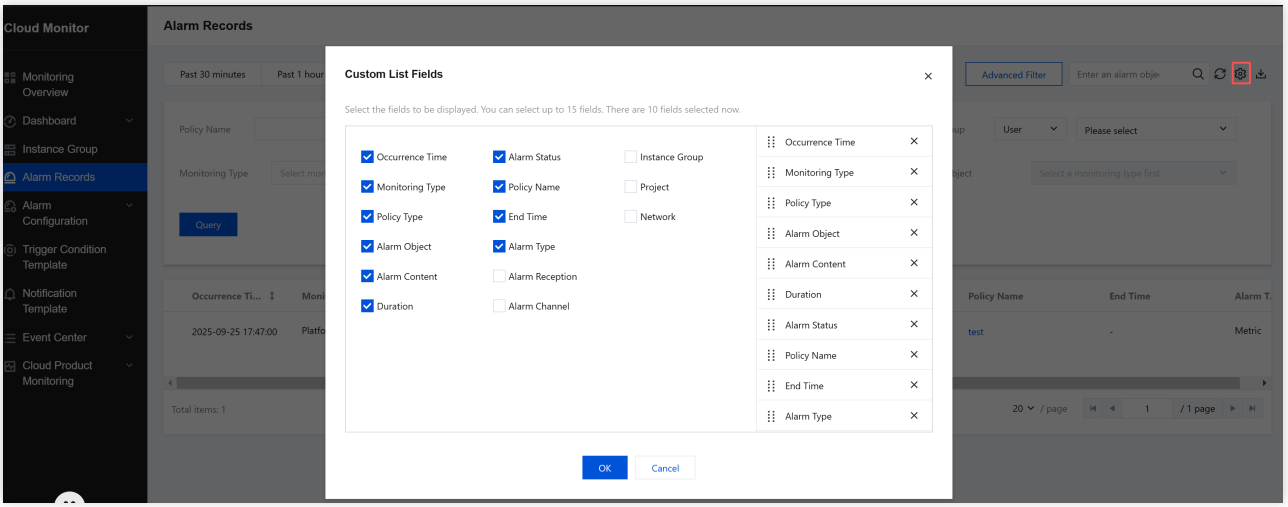
   ii. (Optional) To view alarm records for a certain time period, click the time filter button in the upper-left corner. You can filter alarms generated today, yesterday, and in the last 7 days or 30 days, and you can also select a custom time period. You can view the alarm records in the last six months at most.

   iii. (Optional) You can enter the information of an alarm object (such as instance name, public IP address, and private IP address) in the Alarm Object search box to search for corresponding records.

   iv. (Optional) You can click Advanced Filter to search for matching alarm records based on policy name, alarm content, user information, monitoring type, policy type, and alarm object.

   v. (Optional) You can click the alarm policy name in the list of alarm records to redirect to the detailed alarm policy information.

2. Clear filter conditions.
   After successfully filtering alarm records, click Clear Filter Conditions in the list.
3. Customize list fields.
   i. Log in to the cloud platform console, and choose Cloud Monitor > Alarm Records.
   ii. Click the settings icon in the upper-right corner. You can select the fields to be displayed in the left section of the pop-up box and drag field names in the right section to reorder them, as shown below.




4. Download the list of alarm records.
   i. Log in to the cloud platform console, and choose Cloud Monitor > Alarm Records.
   ii. Click the download icon in the upper-right corner to save the alarm records as a local .csv file.

# Dashboard
# Dashboard Overview

Dashboards are intelligent tools provided by Cloud Monitor for monitoring data of cloud product metrics, featuring visualization and analysis capabilities.
You can create a dashboard for cloud service metrics, and the dashboard will automatically present the monitoring data in a visually appealing chart format on the monitoring panel. This makes the data more intuitive and assists you in analyzing metrics through trends and exception values.

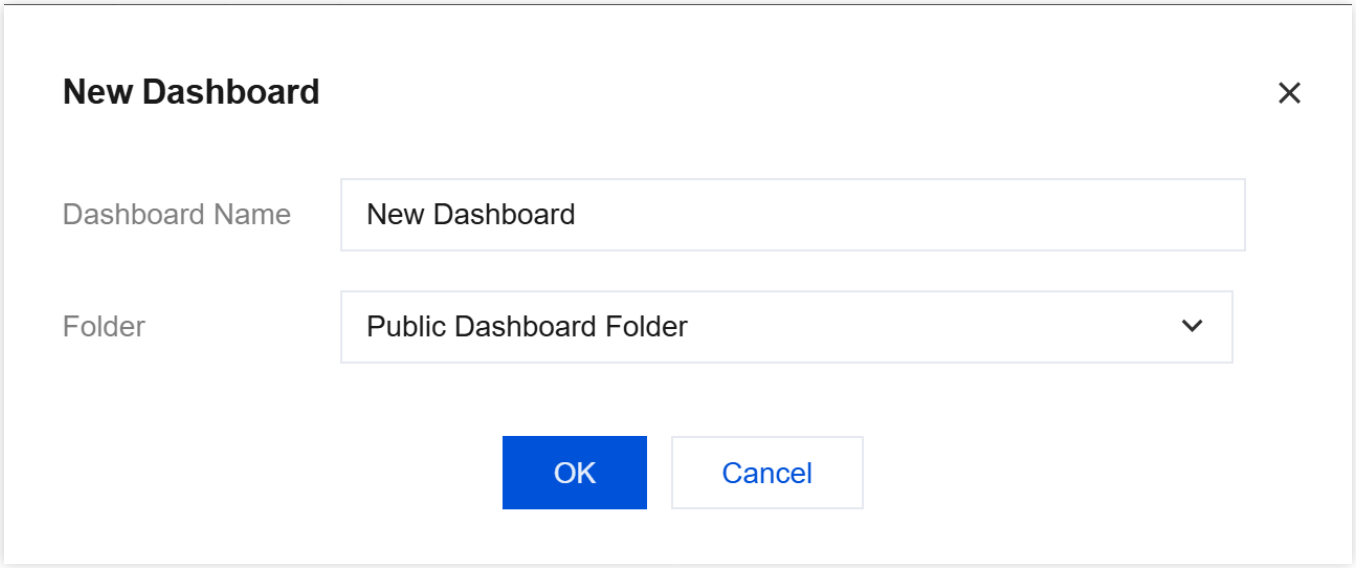
1. Use Cases
    - When receiving a metric alarm, you can use the dashboard to analyze the cause of the alarm.
    - After a new feature is released, you can use the dashboard to check for resource exceptions.
    - Through real-time metric display, you can optimize performance during business peaks.
    - View the loads through the dashboard to determine whether resource scaling is needed.
2. Limitations

| Category | Upper Limitation |
| --- | --- |
| Custom dashboard | 30 |
| Number of charts per dashboard | 20 |
| Number of instances that can be associated with each chart | 50 |

# Creating Dashboards

Before creating monitoring charts, you need to create a dashboard first. You can refer to the following steps to create a dashboard.

1. Log in to the cloud platform console and choose Cloud Monitor > Dashboard to enter the dashboard list.
2. Click Create Dashboard in the upper-left corner of the dashboard list to go to the dashboard creation page.
3. Click Create New Chart in the empty chart to go to the chart editing page. For details, see Creating a Chart.
4. Configure chart parameters. You can click <- in the upper-left corner to return to the dashboard creation page to add a chart.
5. After configuring the chart, click Save in the upper-right corner, and then fill in the dashboard name and choose the folder it belongs to in the pop-up box.

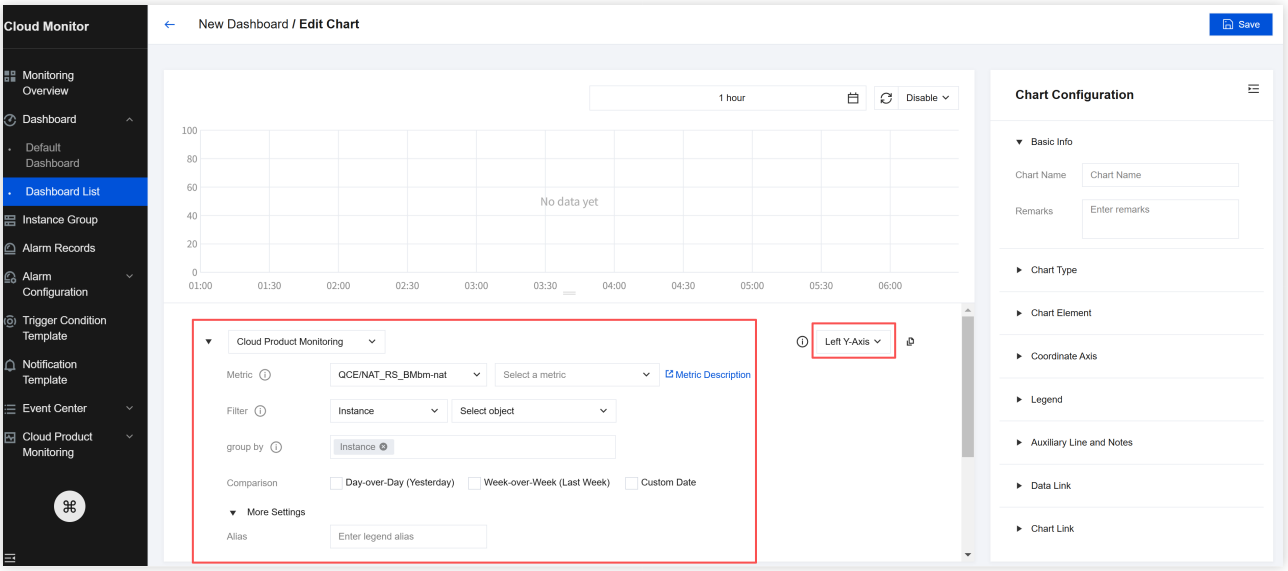



6. Click OK to create the dashboard.

# Creating Charts

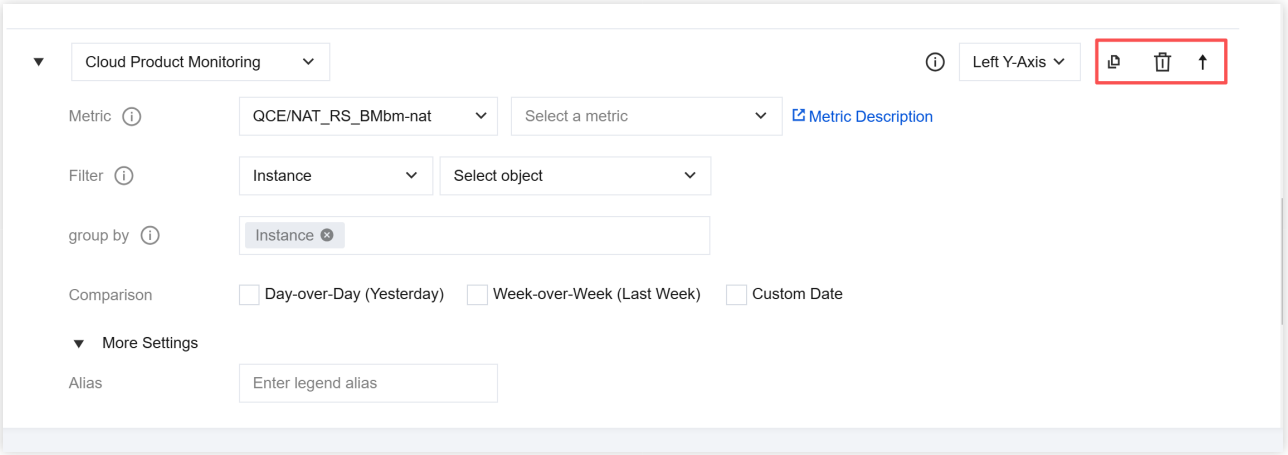
You can refer to the following steps to create a chart.

1. Add metrics.
    i. Log in to the cloud platform console and choose Cloud Monitor > Dashboard to enter the dashboard list.
    ii. Select the corresponding dashboard, and click Create > Create Chart.
    iii. Configure the metric information as follows:
    - Metric: Select the product type and metrics.
    - Filtering: Select the filter conditions to filter the data that meets the conditions and display the data in the chart.
      Instance: The chart will display the monitoring data of the selected instances.
      Template Variables: The chart will display instances from the template variable filter. For configuration of template variables, refer to the template variable documentation.
    - Group by: Similar to the group by feature in SQL, it groups data based on specified tags and then aggregates them according to aggregation algorithms. When no tag is selected, you can customize the statistical method for metrics within the statistical granularity. Statistical methods such as average (avg), maximum (max), minimum (min), and sum (sum) are supported.
    - Comparison: Day-on-day comparison (same period yesterday), week-on-week comparison (same period last week), and custom time comparison are supported. When both options are selected, the chart will display the monitoring curves for the selected instance during the same time period of yesterday and last week, making it easy for you to compare the data.
    - Left Y-axis, Right Y-axis: The Y-axis can be adjusted to the left or right.
    - More configurations:
      Alias: Aliases can be quickly specified for all instances. If you need different aliases for different instances, you can create multiple metrics and enter an alias for each one.
      Enable Sorting Feature: The instances associated with the chart will be sorted according to the sorting rules and display quantity. This feature is useful for monitoring a large number of servers and identifying high and low loads.
      Sorting Rules: There are various ways to sort metrics. Instances can be filtered based on the sorting results.
      Display quantity: It displays the number of instances. For example, if you set the sorting rule as "Maximum Value; Descending Order" and the display quantity as 10, the chart will display the top 10 instances with the maximum values in descending order.
    iv. Configure chart information as follows:
    - Basic Information: includes the chart name and chart remarks.
    - Chart Type: Choose from line chart, bar chart, dashboard, pie chart, numerical statistical chart, heat map, and table. Customize the chart display method based on each chart type.




v. After completing the configuration, click <- in the upper-left corner to return to the dashboard page and add chart configurations.

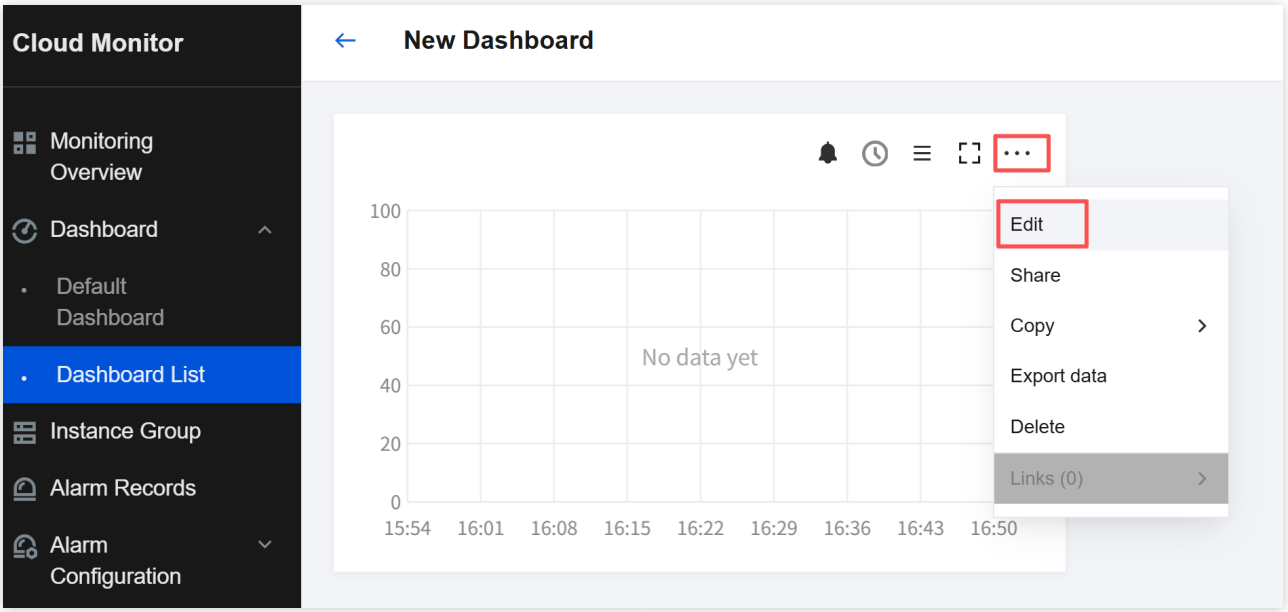
2. Manage multiple metrics.
   - You can click Add Metric or the copy icon to display multiple metrics in the same chart for cross-instance metric data comparison.
   - You can click the up and down arrows to adjust the sorting of metrics.
   - You can click deletion to delete the metric.




3. Edit a chart.

   i. Click … in the upper-right corner of the chart you want to edit.

   ii. In the pop-up box, click Edit to go to the chart editing page.


Cloud Monitor
Monitoring Overview
Dashboard
Default Dashboard
Dashboard List
Instance Group
Alarm Records
Alarm Configuration
New Dashboard
100
80
60
40
20
0
No data yet
15:54
16:01
16:08
16:15
16:22
16:29
16:36
16:43
16:50
Edit
Share
Copy
Export data
Delete
Links (0)

# Viewing Dashboards

You can refer to the following steps to view a dashboard.

1. Operation Steps
    i. Log in to the cloud platform console and choose Cloud Monitor > Dashboard to enter the dashboard list.
    ii. By clicking the corresponding dashboard, you can view the dashboard overview. The system allows you to view a dashboard in full-screen mode, customize the time span, and bookmark panels.
2. View a chart in full-screen mode.
   Click the full screen icon in the upper-right corner of a chart to view it in full-screen mode.
   To exit full-screen mode, press ESC or click the X icon in the upper-right corner.
3. Zoom and move a chart.
    - Chart zooming: You can scale a chart by hovering your mouse pointer over the lower-right corner of the chart until a straight-angle icon appears, as shown below.
    - Chart moving: You can move a chart by hovering your mouse pointer over the name of the chart until a movement icon appears, as shown below.
4. View monitoring data at a specific time.
   You can hover your mouse pointer over the monitoring chart to view the monitoring data at a specific time.
5. Adjust the time span of charts to view monitoring data.
   By default, dashboards display the data of the last 12 hours.
   You can adjust the data interval and granularity for all charts in the dashboard by using the time selection control in the upper-right corner of the dashboard. This feature allows users to review historical monitoring data and troubleshoot issues.
   Comparison table of time periods and granularities in charts

| Time Range | Default Statistical Granularity |
| --- | --- |
| <=1 hour | 1 minute |
| (1 hour, 12 hours] | 1 minute |
| (12 hours, 3 days] | 5 minutes |
| (3 days, 30 days] | 1 hour |
| (30 days, 186 days] | 1 day |

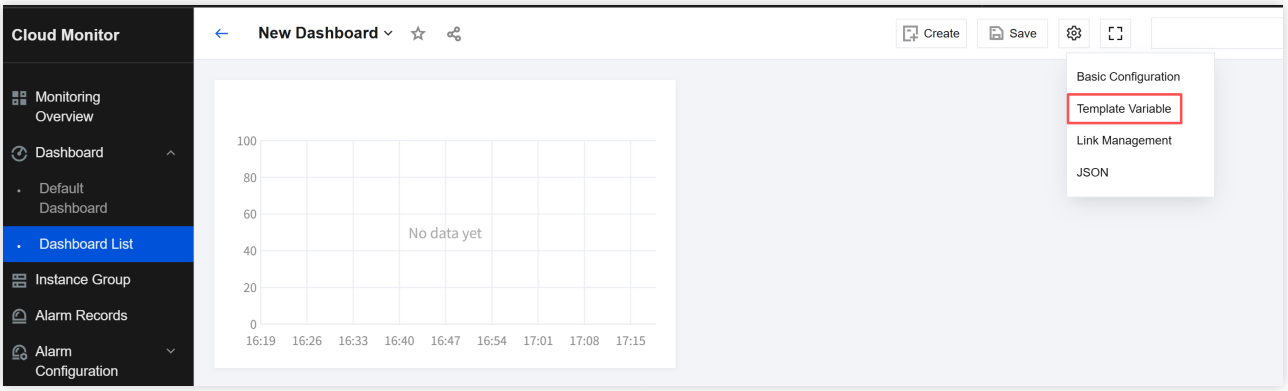
6. Switch dashboard.

Click the inverted triangle button on the right of the current dashboard name to switch between panels and view by current folder, recently visited, favorites, or all.

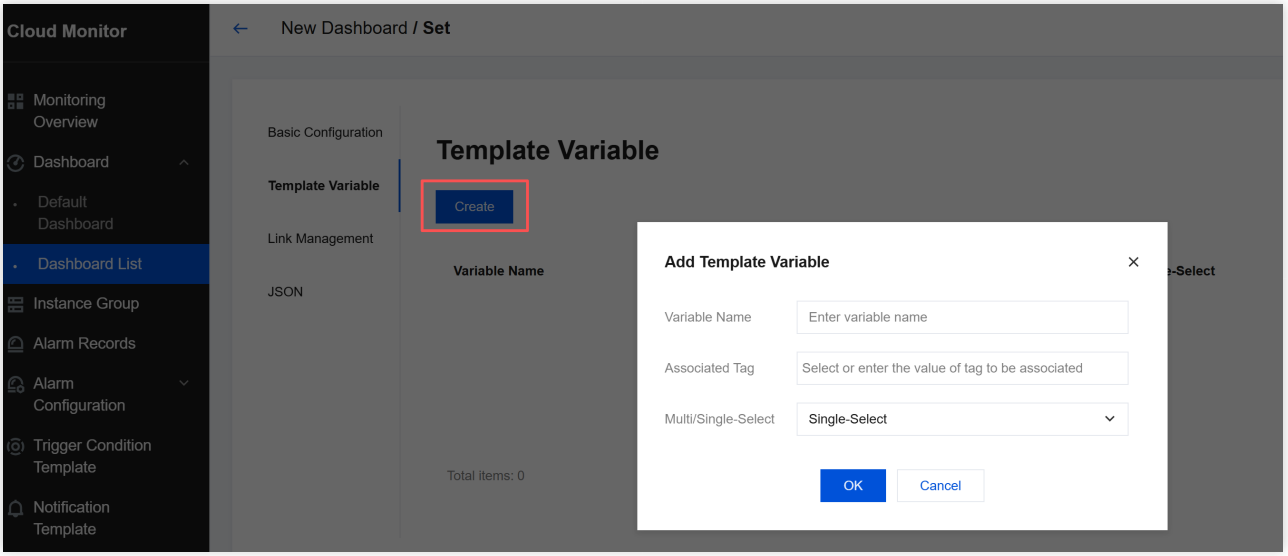
- Current folder: It displays the dashboards in the folder where the current dashboard is located.
- Recently visited: It displays the dashboards you have recently visited.
- Favorites: It displays your favorite dashboards.
- All: It displays all the dashboards you have customized.

# Template Variables

1. Configure template variables.
   i. Log in to the console and choose Cloud Monitor > Dashboard to go to the dashboard list.
   ii. Choose the corresponding dashboard, click the settings icon, and select Template Variables on the Dashboard Management page.



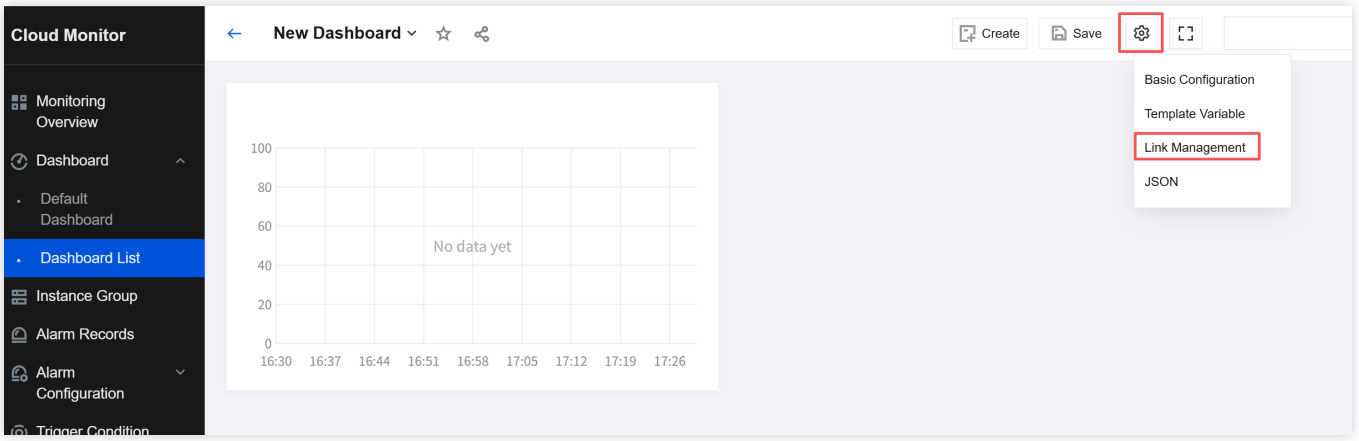

   iii. Template variables are used for customizing dashboard filter conditions. On the Dashboard Management page, you can use template variables to quickly filter instances.
   iv. Click Create on the template management page, and then click OK to create a template variable.



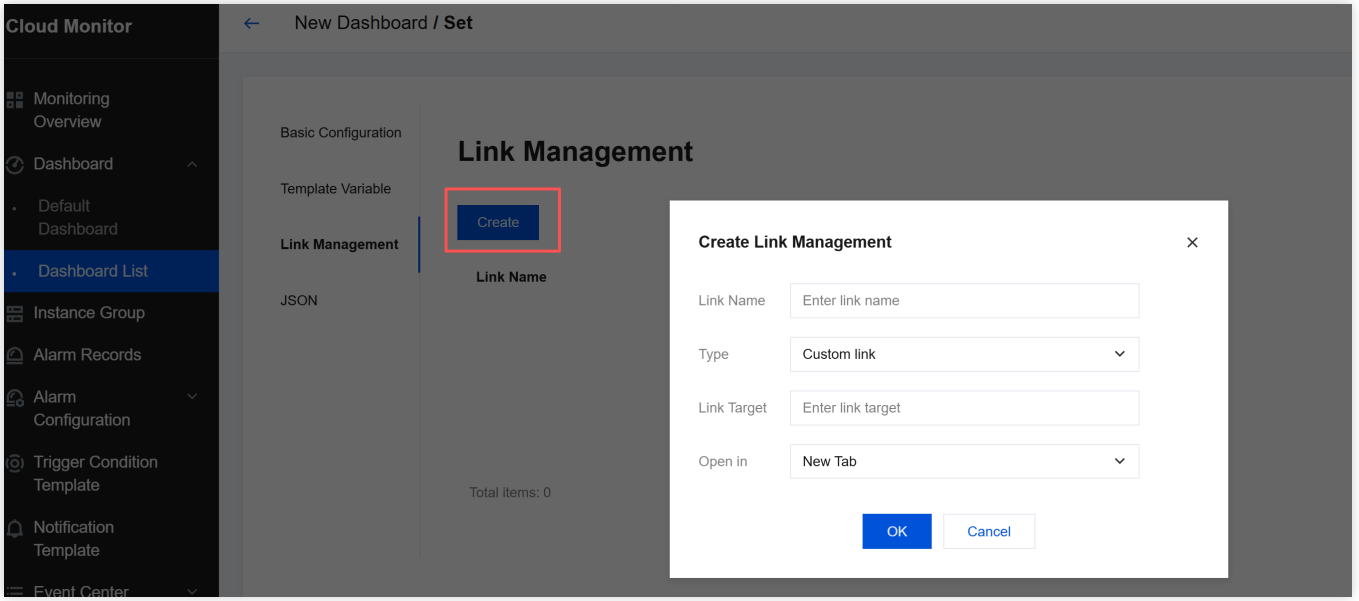

2. Use template variables.
   After template variables are created, they can be used as quick selectors for dashboards and monitoring charts.
   - Use in dashboards: Template variables can be used on the Dashboard Management page to filter the dashboard data.
   - Use in monitoring charts: On the Chart Editing page, quickly select tags to group and aggregate the instance data to be displayed.

# Link Management

1. Log in to the cloud platform console and choose Cloud Monitor > Dashboard to enter the dashboard list.
2. Choose the corresponding dashboard, click the settings icon, and select Link Management on the dashboard management page.




3. Link management is used for customizing quick links to dashboards. To create a link, specify the link name, type, link target, and parameter,




as described below:

- Link Name: custom link name.

  Type
- Custom Link: All links are supported.
- Other Panels: links to other panels.
- Link Target: link address or link panel.

  Parameter
- Time: whether the destination page synchronizes the panel's time during link redirection.

Open Mode

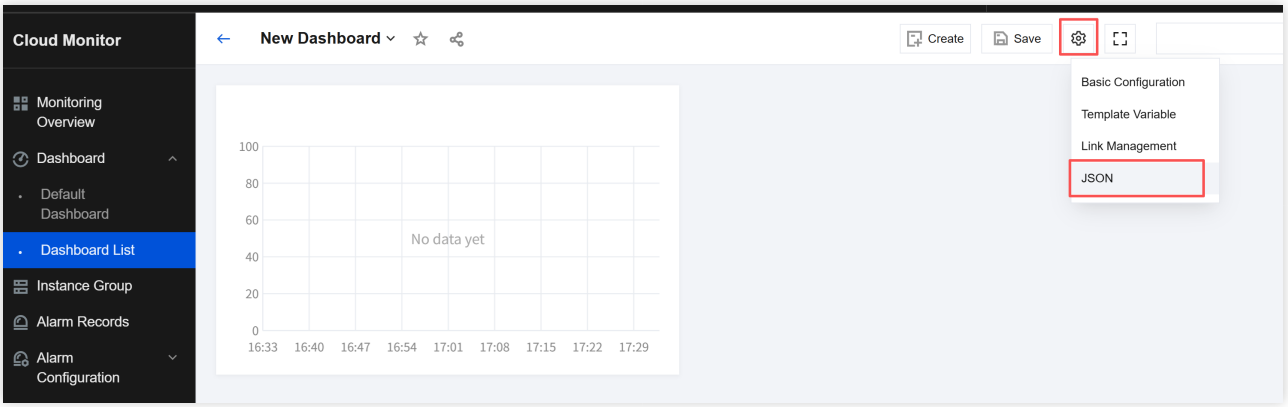
- New Tab: opens the link in a new tab.
- Current Page: opens the link in the current tab.

4. Use links.

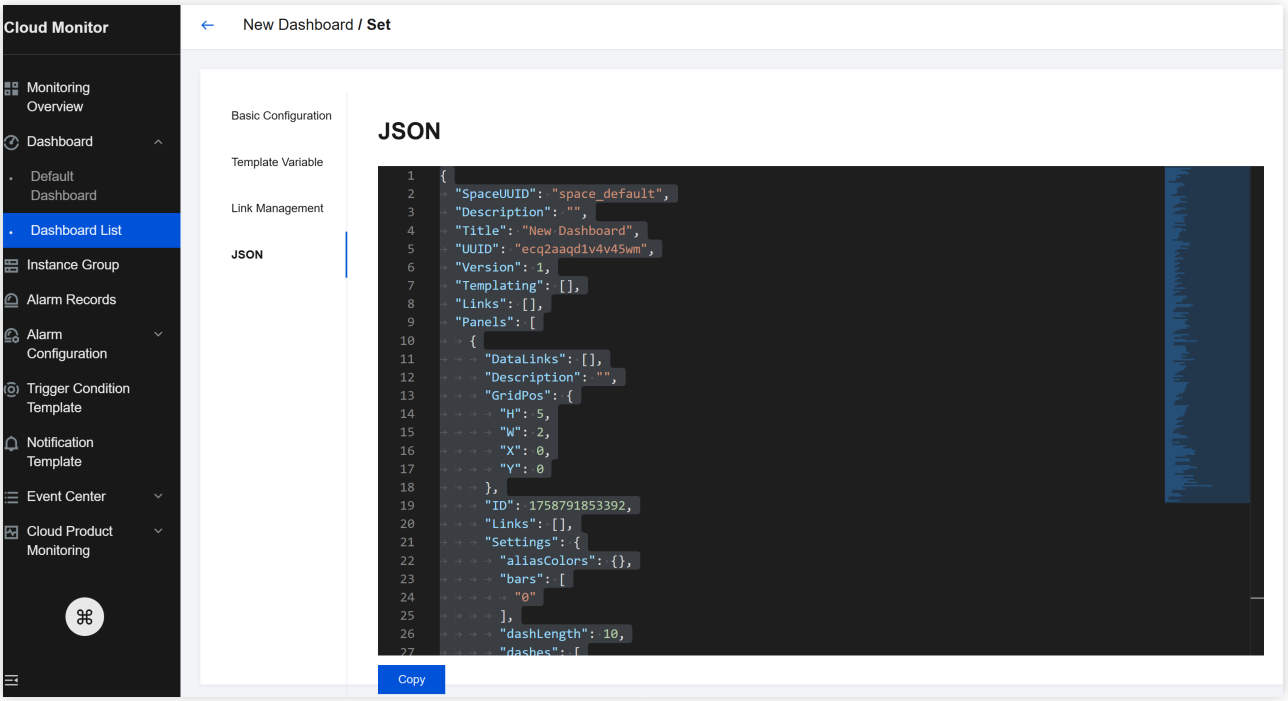
   On the dashboard management page, you can quickly navigate to your custom links for troubleshooting or monitoring data comparison.

# JSON

1. View and copy JSON data.
   i. Log in to the cloud platform console and choose Cloud Monitor > Dashboard to enter the dashboard list.
   ii. Choose the corresponding dashboard, click the settings icon, and select JSON on the dashboard management page.




   iii. You can copy the JSON content to deploy the dashboard to your self-built system.




   The JSON template includes the dashboard's attributes, template variables, and panel query fields. For the specific field description, see the following text.
2. JSON field description.

```
{
    "Description": "",
    "Refresh": "close",
    "Title": "JSONTEST",
```

```
    "UUID": "jdq4joy56is4w60q",
    "Version": 1,
    "Templating": [],
    "Links": [],
    "Panels": [],
    "Time": {
            "From": "now-12h",
            "To": "now"
     },
}
```

| Name | Description |
| --- | --- |
| Description | Current dashboard remarks. |
| Refresh | Auto-refresh time interval. |
| Title | Current dashboard name. |
| UUID | Unique ID of the dashboard. |
| Version | Dashboard version, which increases with each save. |
| Templating | Dashboard template variables. For more information, see Templating. |
| Link | Dashboard link. For more information, see Links. |
| Panels | Chart configuration. See Panels. |
| Time | Time range of the dashboard. |

1. Templating

```
"Templating": [ // Template variables.
      {
         "Label": "CVM instance name", // Template variable alias.
         "Multi": true, // Whether multiple selection is allowed.
         "Name": "cvm", // Tag.
         "Selected": [], // Template variable value.
         "Type": "monitor", // Template variable type.
         "TemplatingType": "basics", // Tag type: basic monitoring or custom monitoring.
    "TemplatingType": '123', // Variable ID.
      }
  ]
```

2. Link

```
"Links": [ // Dashboard link.
      {
         "IncludeVars": true, // Whether to include template variables in the link parameters.
         "KeepTime": true, // Whether to include time variables in the link parameters.
```

```
        "TargetBlank": true, // Whether to open the link in a new tab.
        "Title": "xxx", // Link name.
        "Type": "other", // Link type: custom links or links to other dashboards.
        "Url": "/monitor/dashboard2/dashboards/d/130618223492100096hh64oj49rru3ctk/dashboard2-0yan-shi-an-li-xia-zuan-lian-jie",  // Link address.
        "ID": 0 // ID
      }
  ]
```

3. Panels

```
"Panels": [ // Panel configuration.
    {
      "Collapsed": false,  // Whether the chart group is collapsed.
      "Datasource": null, // Data source.
      "GridPos": { // Chart position.
        "H": 1,
        "W": 6,
        "X": 0,
        "Y": 0
      },
      "ID": 1595472129140, // Chart ID.
      "Panels": [], // Panel is a field for chart groups, containing sub-panels.
      "Title": "Default Chart Group", // Title.
      "Type": "row" // Chart type.
    },
    {
      "DataLinks": [], // Data links configured for the chart.
      "Description": "", // Remarks for the chart.
      "GridPos": {
        "H": 5,
        "W": 6,
        "X": 0,
        "Y": 1
      },
      "ID": 1595471392817,
      "Links": [], // Chart links.
      "Settings": { // Visualization configuration for the chart.
        "aliasColors": {},
        "bars": false,
        "dashLength": 10,
        "dashes": false,
        "datasource": null,
        "decimals": 2, // Legend precision.
        "fieldConfig": {
          "defaults": {
            "custom": {}
          },
          "overrides": []
        },
        "fill": "0.8", // Chart fill transparency.
```

```
"fillGradient": 0,
"gridPos": {
  "h": 8,
  "w": 12,
  "x": 0,
  "y": 0
},
"hiddenSeries": false,
"id": "1595471392817", // ID of the chart panel.
"legend": {// Legend settings.
  "alignAsTable": [ // Whether the legend is displayed in table form.
    "1"
  ],
  "avg": [// Whether to display the average value. ['1'] for true; ['0'] for false.
    "1"
  ],
  "current": [// Whether to display the current value. ['1'] for true; ['0'] for false.
    "1"
  ],
  "max": [// Whether to display the maximum value. ['1'] for true; ['0'] for false.
    "1"
  ],
  "min": [// Whether to display the minimum value. ['1'] for true; ['0'] for false.
    "1"
  ],
  "rightSide": [// Whether to place the legend on the right. ['1'] for true; ['0'] for false.
    "1"
  ],
  "show": [// Whether to display the legend. ['1'] for true; ['0'] for false.
    "1"
  ],
  "total": [// Whether to display the cumulative value. ['1'] for true; ['0'] for false.
    "1"
  ],
  "values": false
},
"lines": [// Whether to display the curve. ['1'] for true; ['0'] for false.
  "1"
],
"linesType": true, // Whether to display a smooth curve. ['1'] for true; ['0'] for false.
"linewidth": "2", // Curve width.
"markline": { // Mark settings.
  "marklineMax": {
    "max": [// Whether to display the maximum value. ['1'] for true; ['0'] for false.
      "1"
    ]
  }
},
"nullPointMode": "1", // Display method for null values in the data. 0: connects null data; 1: does not fill i
n the gaps caused by null values; 2: automatically replaces null values with zeros.
"options": {
  "dataLinks": [] // Array of data links.
```

```
},
"percentage": false,
"pointradius": 2,
"points": false,
"renderer": "flot",
"seriesOverrides": [],
"spaceLength": 10,
"stack": [// Whether to display data in a stacked format. ['1'] for true; ['0'] for false.
    "1"
],
"steppedLine": false,
"targets": [
    {
        "refId": "A",
        "scenarioId": "random_walk"
    }
],
"thresholds": [],
"timeFrom": null,
"timeRegions": [],
"timeShift": null,
"title": "New Chart",
"tooltip": {
    "shared": true,
    "sort": 0,
    "value_type": "individual"
},
"type": "graph",
"xaxis": {
    "buckets": null,
    "mode": "time",
    "name": null,
    "show": true,
    "values": []
},
"yaxes": [
    {
        "decimals": 2, // Left y-axis precision.
        "format": "%", // Left y-axis label unit.
        "label": null,
        "logBase": 1,
        "max": 2, // Maximum value of the left y-axis coordinates.
        "min": 0, // Minimum value of the left y-axis coordinates.
        "show": [// Whether to display the left y-axis. ['1'] for true; ['0'] for false.
            "1"
        ]
    },
    {
        "decimals": 2,
        "format": "",
        "label": null,
        "logBase": 1,
```

```
                "max": null,
                "min": null,
                "show": [
                    "1"
                ]
            }
        ],
        "yaxis": {
            "align": false,
            "alignLevel": null
        }
    },
    "Targets": [ // Metric configuration.
        {
            "Aggregate": "", // Statistical method.
            "CompareLastWeek": false, // Week-on-week comparison.
            "CompareYesterday": false, // Day-on-day comparison.
            "Conditions": [ // Filter conditions.
                {
                    "Dimension": [
                        "{\"InstanceId\":\"ins-19827u5b\"}",
                        "{\"InstanceId\":\"ins-xxooxxoo\"}",
                        "{\"InstanceId\":\"ins-19719mfp\"}"
                    ],
                    "Region": "ap-city1",
                    "Type": "normal"
                }
            ],
            "ConfigId": "cvm",
            "Datasource": "DS_QCEMetric", // Product type.
            "DimensionKey": [
                "InstanceId"
            ],
            "GroupBy": [ // groupby
                "InstanceId"
            ],
            "MetricNames": [ // Metric names.
                "BaseCpuUsage"
            ],
            "Namespace": "QCE/CVM", // Namespace.
            "Period": 60 // Granularity.
        },
    ],
    "Title": "Single Metric - Default Configuration", // Chart name.
    "Type": "graph" // Chart type.
  },
]
```

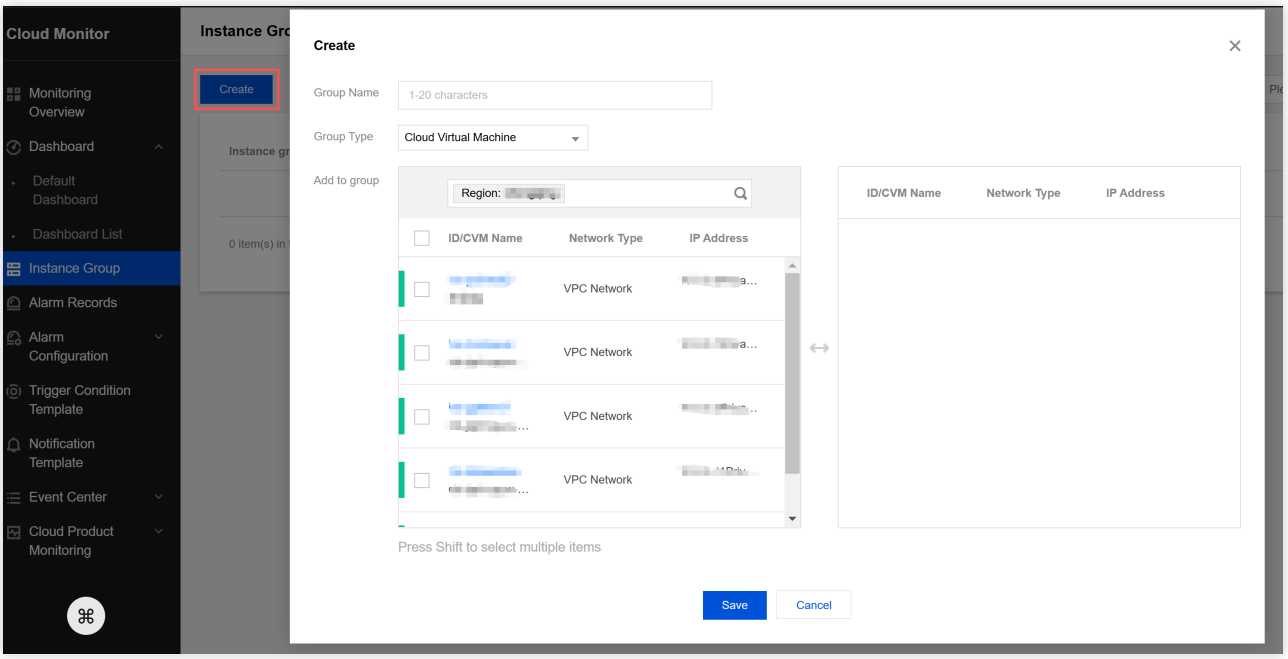
# Grouping Instances

1. Creating an instance group

   i. Log in to the cloud platform console and choose Cloud Monitor > Instance Group.

   ii. Click Create and configure the following options.

   - Group Name: Enter an instance group name.
   - Group Type: Select an option as needed.
   - Add to Group: Select the instances to be added to the group.




   iii. Click Save to create the instance group.

2. Viewing an instance group

   i. Log in to the cloud platform console and choose Cloud Monitor > Instance Group.

   ii. On the instance list page, you can view information such as the instance group name, number of instances, and number of associated alarm policies. You can also click the number of alarm policies to view the instances associated with the group.

3. Editing an instance group

> Note:
>
> - If an instance group is already bound to an alarm policy, instances added to this instance group will automatically be bound to this alarm policy.
> - If an instance group is already bound to an alarm policy, instances removed from this instance group will no longer be bound to this alarm policy.

   i. Log in to the cloud platform console and choose Cloud Monitor > Instance Group.

   ii. Click the name of the instance group that needs to be edited to go to the Instance Management page. On

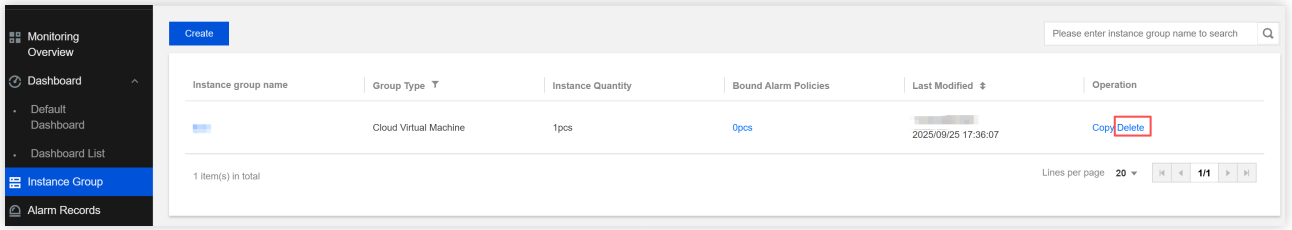
this page, you can edit (add or delete) the instances of this instance group.

- Adding an instance
  - Click Create.
  - In the pop-up configuration box, select the instance that needs to be added to this instance group.
  - Click OK to add the instance to this instance group.
- Deleting an instance
  - Select the instance you want to remove and click Remove.
  - In the pop-up confirmation dialog box, click Confirm to Remove Selected Instance to remove the instance from the instance group.

4. Deleting an instance group

> Note:
>
> If an instance group is already bound to an alarm policy, deleting the instance group will cause the bound alarm policy to become invalid.

i. Log in to the cloud platform console and choose Cloud Monitor > Instance Group.
ii. Find the instance group you want to delete, and click Delete in its right-side operation column.

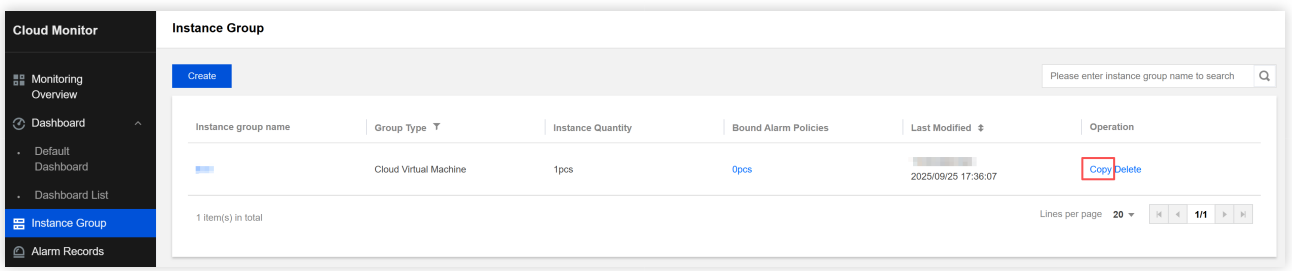



iii. In the pop-up confirmation dialog box, click Confirm Deletion to delete the instance group.

5. Copying an instance group

> Note:
>
> When an instance group is copied, only the instances contained within it will be copied, and the binding relationship between this instance group and the alarm policy will not be copied.

i. Log in to the cloud platform console and choose Cloud Monitor > Instance Group.
ii. Find the instance group you want to delete, and click Copy in its right-side operation column.



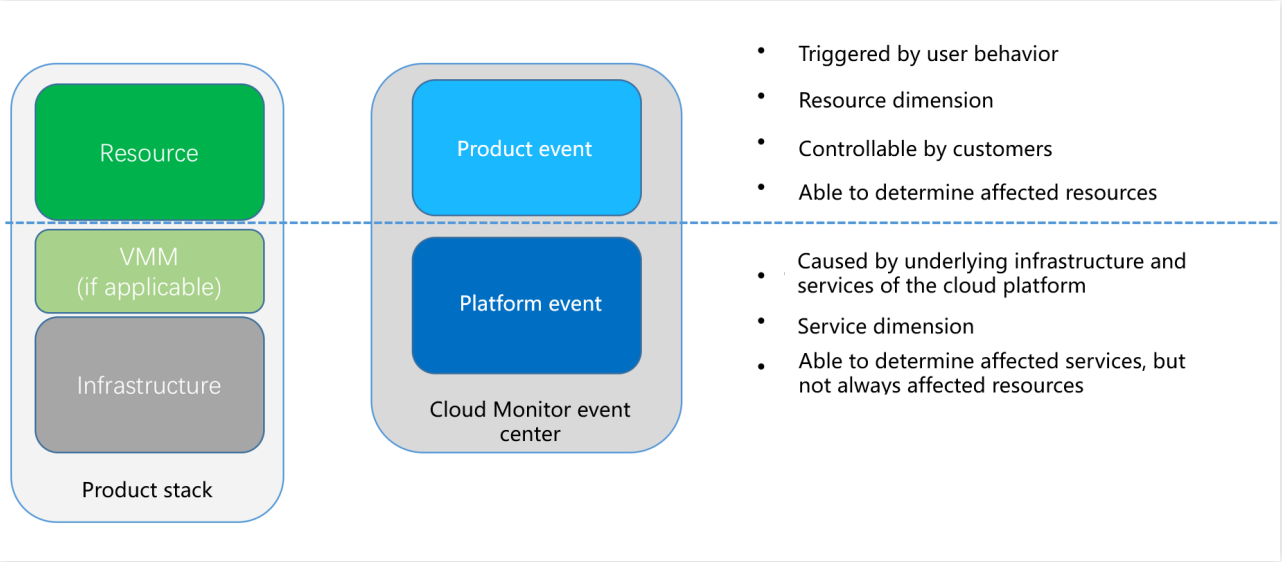

iii. In the pop-up confirmation dialog box, click Confirm to Copy to successfully copy the instance group.

# Event Center
# Event Center Overview

1. Event overview
   An event is a medium that carries information, describing a non-persistent change of a specific object at a particular moment, associated with a unique moment and a unique object. For example, a computer changing from running to shutdown status, the start and end of a program run, or a power outage in an office building. An event is a record of the change at a moment between two different states of an object.
   The purpose of monitoring services is to assist in the discovery, location, and resolution of issues, ensuring the overall stability and performance of systems and services. Information and data are the core drivers of Ops and monitoring.
   Metrics, as the primary source of monitoring information, describe the numerical values and trends of specific variables of particular objects over time in a threshold-based, periodic, persistent form. They reflect the usage and health status of resources and environments. For example, the CPU utilization of a CVM and the number of slow queries in a database instance.
   Metrics can cover the vast majority of scenarios where monitoring identifies issues. However, using metrics alone as the monitoring information source has the following drawbacks:
   - Information transmission latency: Changes in metric thresholds may only become apparent (or not at all) some time after a specific event has occurred.
   - Low accuracy and distinguishability of transmitted information: It becomes more challenging to determine and locate real events or the causes of issues based only on changes in thresholds.
   - Limited dimensions of information: Metrics record information only in threshold form, lacking the capability to provide additional supplementary and ancillary information through textual means.
     Events are introduced as a carrier of monitoring information to more accurately and directly describe the operational status of resources and underlying infrastructure services, facilitating the discovery, location, and resolution of issues in a more efficient manner.
     The advantages of introducing events include improving information accuracy in a result-oriented manner, reducing latency, conveying more information, and enhancing the dimensionality of monitoring information.
2. Event Center introduction
   As the event information center of the cloud platform, Cloud Monitor's Event Center manages the important event information during the lifecycle and running of all product resources on the cloud platform. It builds a complete event consumption channel and process to support users' monitoring and Ops on the cloud.
   The event information provided by Event Center is obtained from various product modules and underlying infrastructure services on the cloud platform. It is then aggregated, determined, condensed, and finally displayed. The information sources originate from the underlying system logs and monitoring items of each module, ensuring the accuracy and value of the information for users.
3. Event Type
   Based on the information source, cause, characteristics, and form of events, the events covered by Event Center are divided into two main categories: product events and platform events, each supported by two separate product pages and consumption processes.
   - Product events: They originate from the products and resource instances (such as a CVM instance) used by

users on the cloud. They are directly or indirectly triggered by users' operational behaviors during use, and belong to the dimension specific to particular resource instances. Users have control over the events and the ability to handle them themselves. The impact and associated resource instances of the events can be identified.

- Platform events: They are generated in the underlying infrastructure and services that support the cloud platform (such as the virtualization layer VMM, underlying physical servers, networks, and storage modules that support CVM). They are generated or caused by the underlying infrastructure and services that support the cloud platform. Since the root causes of events are not user behaviors and events belong to the service dimension, users can not control these events, and only the cloud platform has the handling capability. Services and product modules affected by and associated with events can be identified, but the affected and associated resource instances may not be identified.



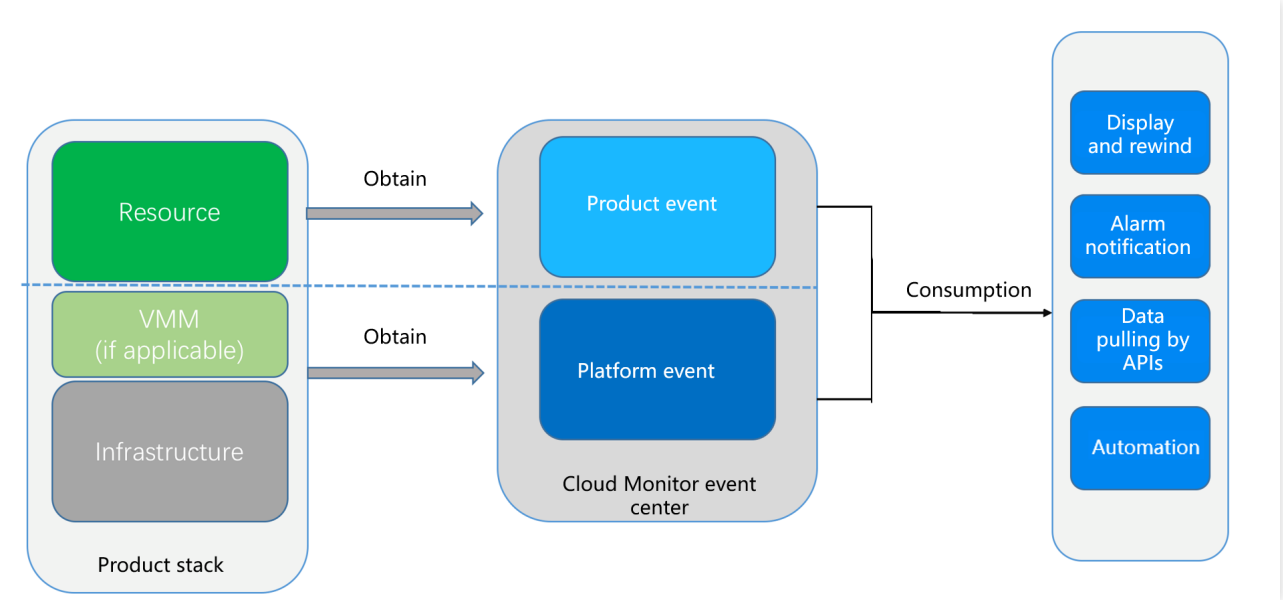

- Event comparison

| Comparison Item | Product Event | Platform Event |
| --- | --- | --- |
| Event source | Products and resource instances used by users on the cloud. | Underlying infrastructure services of the cloud platform. |
| Event cause | Directly or indirectly triggered by users' operational behaviors. | Generated or caused by the underlying cloud infrastructure and services. |
| Event dependency dimension | Resource instances. | Underlying services. |
| Event handling capability | Users have control over the events and can handle them themselves. | Users have no control over the events and need the cloud platform to handle them. |
| Event-related object granularity | The impacted and associated resource instances can be | The impacted and associated underlying services or product modules can be confirmed, while the |

| | | |
|---|---|---|
| | clearly confirmed. | impacted resource instances upstream may not be confirmed. |

4. Event Center use cases

   Based on event information, Event Center plans and provides comprehensive event consumption channels and capabilities.




   - Presentation and tracing: Records all event information related to the account, with the console providing the ability to view, globally filter, search, and trace event information.
   - Alarm notifications: Enables users to receive alarms or subscribe to notifications for monitored events and resources, ensuring timely communication with the owner.
   - Retrieval with APIs: Provides APIs to retrieve event information for integration with self-built monitoring systems.
   - Automation: Subsequent updates will support configuring linked invocation operations with events as triggers, enabling users to achieve automated Ops.

5. Benefits
   - It enhances the monitoring information dimensions of cloud resources, providing more comprehensive data support for monitoring and Ops.
   - It provides event information consumption channels, aiding in the transformation of monitoring information into value.
   - It converges determination logic to more efficiently and directly pinpoint impacted resources and causes.
   - It allows traceable review of event information, ensuring awareness of significant changes in resource and platform lifecycle events.
   - It can trigger automatic responses to specific resource and environment changes.

# Product Event List

The Event Center of Cloud Monitor currently provides the following monitoring information for product events:The Event Center of Cloud Monitor currently provides the following monitoring information for product events:

1. CVMCVM

| Event English Name | Event Type | Event Dimension | Recovery Concept Involved | Event Description | Handling Method and Suggestion |
|---|---|---|---|---|---|
| GuestCoreError | Exception | CVM instance | No | An operating system kernel bug or driver issue causes a critical error in the operating system kernel. | 1.Check whether any kernel driver modules are loaded into the system other than those provided by the kernel. Try not to load these modules and observe the operating status of the system. 2. Monitor bug reports of the kernel and operating system distributions, and consider upgrading the kernel to resolve the bugs. 3. By default, kdump is enabled for CVM instances. When a panic occurs, system memory dump information will be generated in the /var/crash directory. Use |

| Event English Name | Event Type | Event Dimension | Recovery Concept Involved | Event Description | Handling Method and Suggestion |
|---|---|---|---|---|---|
| | | | | | the crash tool to analyze the information. |
| GuestOom | Exception | CVM instance | No | System memory usage is overloaded. | 1.Check whether the memory capacity configured in the current system meets business requirements. If additional capacity is required, it is recommended to upgrade the CVM memory configuration. 2. View processes that are terminated during OOM based on system logs (such as dmesg and /var/log/messages) to check whether the memory used by the processes meets the expectations. Use tools such as valgrind to analyze whether memory leakage occurs. |
| DiskReadonly | Exception | CVM instance | Yes | Data cannot be written into the disk. | 1.Check whether the disk storage is full. 2. For Linux operating |

| Event English Name | Event Type | Event Dimension | Recovery Concept Involved | Event Description | Handling Method and Suggestion |
|---|---|---|---|---|---|
| | | | | | systems, check whether inodes are exhausted (df -i). 3. Check whether the file system is corrupted. |
| GuestReboot | Status change | CVM instance | Yes | The CVM instance is restarted. | This event is triggered when the CVM instance is restarted. Check whether the status change meets the expectations based on the actual situation. |
| PacketDroppedByQosWanOutBandwidth | Exception | CVM instance | Yes | The outbound public network bandwidth of the CVM instance exceeds the upper limit, causing packet loss. | Increase the upper limit of the public network bandwidth. If the maximum supported limit has been reached, reduce the bandwidth usage of the instance through means such as load balancing. |
| PacketDroppedByQosConnectionSession | Exception | CVM instance | Yes | An excessive number of connections to the CVM instance leads to packet loss. Taking the | Contact after-sales service personnel. |

| Event English Name | Event Type | Event Dimension | Recovery Concept Involved | Event Description | Handling Method and Suggestion |
|---|---|---|---|---|---|
| | | | | 8255C CPU model as an example, it is generally recognized that if the number of connections exceeds 500,000, packet loss may occur. | |

2. VPN GatewayVPN Gateway

| Event English Name | Event Type | Event Dimension | Recovery Concept Involved | Event Description | Handling Method and Suggestion |
|---|---|---|---|---|---|
| PacketDroppedByQosWanOutBandwidth | Exception | VPN gateway instance | Yes | The outbound public network bandwidth of the VPN gateway instance exceeds the upper limit, causing packet loss. | Increase the upper limit of the public network bandwidth. |
| PacketDroppedByQosConnectionSession | Exception | VPN gateway instance | Yes | The number of connections to the VPN gateway instance exceeds the limit, causing packet loss. | Contact after-sales service personnel. |

# Cloud Product Monitoring
# Cloud Product Monitoring Metrics Overview

This section describes the monitoring metrics for various products in Cloud Monitor.

## Computing

- For CVM, see CVM.
- For Kubernetes Engine (TKE), see Monitoring Metrics for TKE.
- For Bare Metal Server (BMS), see Monitoring Metrics for BMS.

## Network

- For Cloud Load Balancer (CLB), see Monitoring Metrics for CLB.
- For Virtual Private Cloud (VPC), see Monitoring Metrics for VPC.
- For Direct Connect (DC), see Monitoring Metrics for DC.

## Storage

- For Cloud Block Storage (CBS), see Monitoring Metrics for CBS.
- For Cloud Object Storage (COS), see Monitoring Metrics for COS.
- For Cloud Storage Private (CSP), see Monitoring Metrics for CSP.
- For Cloud File Storage (CFS), see Monitoring Metrics for CFS.

## Database

- For CloudDB for Redis® (Redis®), see Monitoring Metrics for Redis®.
- For TDSQL for MySQL, see Monitoring Metrics for TDSQL for MySQL.
- For Cloud MongoDB (MongoDB), see Monitoring Metrics for MongoDB.

## Middleware

- For TDMQ for CMQ (CMQ), see Monitoring Metrics for CMQ.
- For TDMQ for CKafka (Ckafka), see Monitoring Metrics for Ckafka.
- For Service Framework (TSF), see Monitoring Metrics for TSF.
- For TDMQ for Apache Pulsar (TDMQ Pulsar), see Monitoring Metrics for TDMQ Pulsar.

## TI-ONE

- For TI-ONE Platform, see Monitoring Metrics for TI-ONE Platform.

# CVM
# CVM Monitoring Metrics

CVM monitoring metrics need to be collected through the CVM monitoring component. Users can automatically install the monitoring component by selecting Cloud Monitor when creating machines, or manually install it based on the Monitoring Component Installation Guide.
Cloud Monitor provides the following monitoring metrics for CVM instances:

| Metric/Event ID | Metric/Event Name | Unit | Metric Description |
|---|---|---|---|
| CPUUsage | CPU utilization | % | Real-time CPU utilization during machine operation. |
| BaseCpuUsage | Basic CPU utilization | % | Basic CPU utilization is collected and reported by the host, and data can be viewed without the need to install the monitoring component. Data can still be collected and reported even when the instance is under high load. |
| MemUsed | Memory usage | MB | Amount of memory actually used by the user, excluding the memory occupied by buffers and system cache. |
| MemUsage | Memory utilization | % | Amount of memory actually used by the user, excluding the memory occupied by buffers and system cache. |
| CvmDiskUsage | Disk utilization | % | Disk utilization monitoring. |
| DiskIOAwait | Disk I/O wait | ms | Average wait time per I/O operation for disk partitions. |
| DiskReadtraffic | Disk read traffic | KB/s | Average data read volume from disk to memory per second. |
| DiskWritetraffic | Disk write traffic | KB/s | Average data write volume from memory to disk per second. |
| LanIntraffic | Inbound private network bandwidth | MB/s | Average inbound traffic per second on private NIC. |

| Metric/Event ID | Metric/Event Name | Unit | Metric Description |
|---|---|---|---|
| LanOuttraffic | Outbound private network bandwidth | MB/s | Average outbound traffic per second on private NIC. |
| LanInpkg | Number of inbound packets on the private network | Counts/ second | Average inbound packet volume per second for private NIC. |
| LanOutpkg | Number of outbound packets on the private network | Counts/ second | Average outbound packet volume per second for private NIC. |
| WanIntraffic | Inbound public network bandwidth | MB/s | Average inbound traffic per second on public NIC. |
| WanOuttraffic | Outbound public network bandwidth | MB/s | Average outbound traffic per second on public NIC. |
| WanInpkg | Number of inbound packets on the public network | Counts/ second | Average inbound packet volume per second for public network NIC. |
| WanOutpkg | Number of outbound packets on the public network | Counts/ second | Average outbound packet volume per second for public network NIC. |
| TcpCurrEstab | Number of TCP connections | Count | Monitoring of TCP connections. |

# Installing the CVM Monitoring Component

If users need to use Cloud Monitor to view the CVM metric data and trigger alarms, they need to correctly install the monitoring component on the CVM, as CVM metric data collection depends on the monitoring component.
Note: To ensure the normal reporting of monitoring data, the user's CVM needs to allow tcp dport 80.

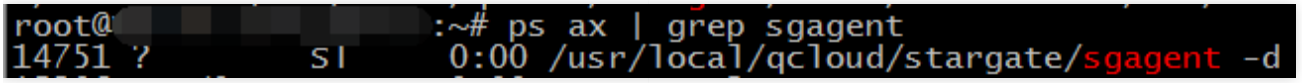
## Linux Installation Guide

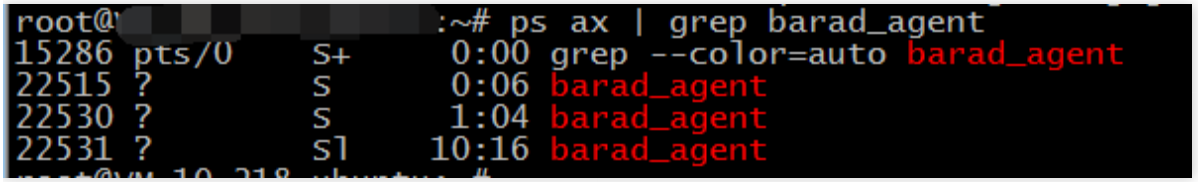
After logging in to the Linux instance, users can execute the following commands for installation:

```
wget http://update2.agent.yun.com/update/linux_stargate_installer
chmod +x linux_stargate_installer
./linux_stargate_installer
```

If the installation is successful, the following content will be displayed:




## Windows Installation Guide

1. After logging in to the Windows instance, users can access http://update2.agent.yun.com/update/windows-stargate-installer.exe via the private network to download the installation package windows-stargate-installer.exe .
2. Run this program to perform automated installation.

# Common Causes and Solutions for Monitoring Components Going Offline

Offline criteria for monitoring components:

> When the monitoring component of a CVM fails to report data for 5 consecutive minutes, the platform considers the monitoring component of this CVM to be offline.

The following are the top causes for monitoring components going offline and the corresponding troubleshooting methods.

## 1. User Operations on the CVM Through the Console or Command Line

When a CVM is shut down through operations, it enters a shutdown status, causing the monitoring component to go offline and no data available.

When users operate through the CVM console or log into the CVM to perform common CVM Ops such as restarting, upgrading the CVM, reinstalling, or creating images, it will result in a timeout in CVM monitoring data reporting, causing the monitoring component to go offline.

Troubleshooting method: You can check whether there are relevant Ops on the CVM based on the specific time point. You can also view operation logs by clicking Operation Log on the CVM Details page.

## 2. High Load on the CVM

High CPU load, full memory, and full bandwidth on CVM can all lead to abnormal data reporting by the monitoring component.

Troubleshooting method: You can log in to the CVM or view the monitoring views to check whether CPU, memory, and bandwidth utilization reach 100% . If so , you can scale out the CVM based on the actual situation.

## 3. Configuration Error of CVM's Private DNS

The configuration error of CVM's private DNS will prevent the monitoring component from reporting data.

Troubleshooting method: For private DNS configuration of the Cloud Platform, see Private DNS Access and Configuration.

# Monitoring Metrics for TKE

Cloud Monitor provides monitoring metrics for TKE in the following dimensions:

## Monitoring Metrics for Clusters

| Metric ID | Metric Name | Unit |
| --- | --- | --- |
| K8sClusterNodeTotal | Number of nodes | Count |
| K8sClusterPodsUsedTotal | Number of Pods | Count |
| K8sClusterAllocatablePodsTotal | Number of allocatable Pods | Count |
| K8sClusterCpuCoreTotal | Total CPU configuration | Cores |
| K8sClusterRateCpuCoreRequest | CPU allocation percentage | % |
| K8sClusterCpuCoreUsed | CPU usage | Cores |
| K8sClusterEksCpuCoreUsed | CPU usage (elastic containers) | Cores |
| K8sClusterRateCpuCoreUsed | CPU utilization | % |
| K8sClusterMemoryTotal | Total memory | GB |
| K8sClusterRateMemRequestBytes | Memory allocation rate | % |
| K8sClusterMemUsageBytes | Memory usage | MB |
| K8sClusterRateMemUsageBytes | Memory utilization | % |
| K8sClusterMemNoCacheBytes | Memory usage (excluding cache) | MB |
| K8sClusterRateMemNoCacheBytes | Memory utilization (excluding cache) | % |
| K8sClusterEksMemUsageBytes | Memory usage (elastic containers) | MB |
| K8sClusterEksMemNoCacheBytes | Memory usage (elastic containers, excluding cache) | MB |
| K8sClusterNetworkReceiveBytes | Inbound network traffic | MB |
| K8sClusterNetworkTransmitBytes | Outbound network traffic | MB |
| K8sClusterNetworkReceiveBytesBw | Inbound network bandwidth | MB/s |

| Metric ID | Metric Name | Unit |
| --- | --- | --- |
| K8sClusterNetworkTransmitBytesBw | Outbound network bandwidth | MB/s |
| K8sClusterNetworkReceivePackets | Number of inbound network packets | Counts/ second |
| K8sClusterNetworkTransmitPackets | Number of outbound network packets | Counts/ second |
| K8sClusterFsReadTimes | Number of reads from the block device | Times |
| K8sClusterFsWriteTimes | Number of writes on the block device | Times |
| K8sClusterFsReadBytes | Size of data read from the block device | MB |
| K8sClusterFsWriteBytes | Size of data written on the block device | MB |
| K8sClusterEtcdDbTotalSizeBytes | etcd storage capacity | MB |
| K8sClusterGpuTotal | Total GPU | Cards |
| K8sClusterRateGpuRequest | GPU allocation rate | % |
| K8sClusterGpuUsed | GPU usage | Cards |
| K8sClusterRateGpuUsed | GPU utilization | % |
| K8sClusterGpuMemoryTotalBytes | Total GPU memory | GB |
| K8sClusterRateGpuMemoryRequest | GPU memory allocation rate | % |
| K8sClusterGpuMemoryUsedBytes | GPU memory usage | MB |
| K8sClusterRateGpuMemoryUsed | GPU memory utilization | % |

# Monitoring Metrics for Nodes

| Metric ID | Metric Name | Unit |
| --- | --- | --- |
| K8sNodeStatus | Node status | - |
| K8sNodePodRestartTotal | Number of Pod restarts | Times |
| K8sNodeCpuCoreRequest | CPU allocation | Cores |
| K8sNodeCpuUsage | CPU utilization | % |

| Metric ID | Metric Name | Unit |
| --- | --- | --- |
| K8sNodeMemRequestBytes | Memory allocation | MB |
| K8sNodeMemUsage | Memory utilization | % |
| K8sNodeLanIntraffic | Inbound private network bandwidth | MB/s |
| K8sNodeLanOuttraffic | Outbound private network bandwidth | MB/s |
| K8sNodeWanIntraffic | Inbound public network bandwidth | MB/s |
| K8sNodeWanOuttraffic | Outbound public network bandwidth | MB/s |
| K8sNodeTcpCurrEstab | Number of TCP connections | Count |
| K8sNodeGpuUsed | GPU usage | Cards |
| K8sNodeRateGpuUsed | GPU utilization | % |
| K8sNodeGpuMemoryUsedBytes | GPU memory usage | MB |
| K8sNodeRateGpuMemoryUsed | GPU memory utilization | % |
| K8sNodeEniipRequestNode | eniip allocation on the node | Count |
| K8sNodeDirecteniRequestNode | direct_eni allocation on the node | Count |

# Monitoring Metrics for Workloads

| Metric ID | Metric Name | Unit |
| --- | --- | --- |
| K8sWorkloadAnomaly | Workload exception | - |
| K8sWorkloadPodsUsedTotal | Number of Pods | Count |
| K8sWorkloadPodRestartTotal | Number of Pod restarts | Times |
| K8sWorkloadCpuCoreUsed | CPU usage | Cores |
| K8sWorkloadRateCpuCoreUsed | CPU utilization | % |
| K8sWorkloadRateEksCpuCoreUsedPod | CPU utilization (elastic containers, as % of Pod specification) | % |
| K8sWorkloadMemUsageBytes | Memory usage | MB |
| K8sWorkloadRateMemUsed | Memory utilization | % |

| Metric ID | Metric Name | Unit |
| --- | --- | --- |
| K8sWorkloadMemNoCacheUsedBytes | Memory usage (excluding cache) | MB |
| K8sWorkloadRateMemNoCacheUsed | Memory utilization (excluding cache) | % |
| K8sWorkloadRateEksMemUsedPod | Memory utilization (elastic containers, as % of Pod specification) | % |
| K8sWorkloadRateEksMemNoCacheUsedPod | Memory utilization (elastic containers, as % of Pod specification, excluding cache) | % |
| K8sWorkloadNetworkReceiveBytesBw | Inbound network bandwidth | MB |
| K8sWorkloadNetworkTransmitBytesBw | Outbound network bandwidth | MB |
| K8sWorkloadNetworkReceiveBytes | Inbound network traffic | MB |
| K8sWorkloadNetworkTransmitBytes | Outbound network traffic | MB |
| K8sWorkloadNetworkReceivePackets | Number of inbound network packets | Counts/ second |
| K8sWorkloadNetworkTransmitPackets | Number of outbound network packets | Counts/ second |
| K8sWorkloadGpuUsed | GPU usage | Cards |
| K8sWorkloadRateGpuUsed | GPU utilization | % |
| K8sWorkloadRateEksGpuUsed | GPU utilization (elastic containers) | % |
| K8sWorkloadGpuMemoryUsedBytes | GPU memory usage | MB |
| K8sWorkloadRateGpuMemoryUsed | GPU memory utilization | % |
| K8sWorkloadEksGpuVideoMemoryUsedBytes | GPU memory usage (elastic containers) | MB |
| K8sWorkloadRateEksGpuVideoMemoryUsed | GPU memory utilization (elastic containers) | % |
| K8sWorkloadFsReadTimes | Number of reads from the block device | Times |
| K8sWorkloadFsWriteTimes | Number of writes on the block device | Times |
| K8sWorkloadFsReadBytes | Size of data read from the block device | MB |
| K8sWorkloadFsWriteBytes | Size of data written on the block device | MB |

# Monitoring Metrics for Pods

| Metric ID | Metric Name | Unit |
|---|---|---|
| K8sPodRestartTotal | Number of Pod restarts | Times |
| K8sPodReadyStatus | PodReady status | - |
| K8sPodCpuCoreUsed | CPU usage | Cores |
| K8sPodRateCpuCoreUsedNode | CPU utilization (% of node) | % |
| K8sPodRateCpuCoreUsedRequest | CPU utilization (% of request) | % |
| K8sPodRateCpuCoreUsedLimit | CPU utilization (% of limit) | % |
| K8sPodEksCpuCoreUsedPod | CPU utilization (elastic containers, as % of Pod specification) | % |
| K8sPodMemUsageBytes | Memory usage | MB |
| K8sPodMemNoCacheBytes | Memory usage (excluding cache) | MB |
| K8sPodRateMemUsageNode | Memory utilization (% of node) | % |
| K8sPodRateMemUsageRequest | Memory utilization (% of request) | % |
| K8sPodRateMemUsageLimit | Memory utilization (% of limit) | % |
| K8sPodRateMemNoCacheNode | Memory utilization (% of node, excluding cache) | % |
| K8sPodRateMemNoCacheRequest | Memory utilization (% of request, excluding cache) | % |
| K8sPodRateMemNoCacheLimit | Memory utilization (% of limit, excluding cache) | % |
| K8sPodRateEksMemUsagePod | Memory utilization (elastic containers, as % of Pod specification) | % |
| K8sPodRateEksMemNoCachePod | Memory utilization (elastic containers, as % of Pod specification, excluding cache) | % |
| K8sPodNetworkReceiveBytesBw | Inbound network bandwidth | MB |
| K8sPodNetworkTransmitBytesBw | Outbound network bandwidth | MB/s |
| K8sPodNetworkReceiveBytes | Inbound network traffic | MB |

| Metric ID | Metric Name | Unit |
| --- | --- | --- |
| K8sPodNetworkTransmitBytes | Outbound network traffic | MB |
| K8sPodNetworkReceivePackets | Number of inbound network packets | Counts/second |
| K8sPodNetworkTransmitPackets | Number of outbound network packets | Counts/second |
| K8sPodGpuRequest | GPU application volume | Cards |
| K8sPodGpuUsed | GPU usage | MB |
| K8sPodRateGpuUsedNode | GPU utilization (% of node) | % |
| K8sPodRateGpuUsedRequest | GPU utilization (% of request) | % |
| K8sPodRateEksGpuUsed | GPU utilization (elastic container) | % |
| K8sPodGpuMemoryRequest | GPU memory application volume | MB |
| K8sPodGpuMemoryUsed | GPU memory usage | MB |
| K8sPodRateGpuMemoryUsedNode | GPU memory utilization (% of node) | % |
| K8sPodRateGpuMemoryUsedRequest | GPU memory utilization (% of request) | % |
| K8sPodEksGpuVideoMemoryUsed | GPU memory usage (elastic containers) | MB |
| K8sPodRateEksGpuVideoMemoryUsed | GPU memory utilization (elastic containers) | % |
| K8sPodFsReadBytes | Block device read bandwidth | MB/s |
| K8sPodFsWriteBytes | Block device write bandwidth | MB/s |
| K8sPodFsReadTimes | Number of reads from the block device | Times |
| K8sPodFsWriteTimes | Number of writes on the block device | Times |
| K8sPodRootfsSize | rootfs space size | MiB |
| K8sPodRootfsUsed | rootfs space usage | GB |

# Monitoring Metrics for Containers

| Metric ID | Metric Name | Unit |
| --- | --- | --- |

| Metric ID | Metric Name | Unit |
| --- | --- | --- |
| K8sContainerCpuCoreUsed | CPU usage | Cores |
| K8sContainerRateCpuCoreUsedNode | CPU utilization (% of node) | % |
| K8sContainerRateCpuCoreUsedRequest | CPU utilization (% of request) | % |
| K8sContainerRateCpuCoreUsedLimit | CPU utilization (% of limit) | % |
| K8sContainerMemUsageBytes | Memory usage | MB |
| K8sContainerRateMemUsageNode | Memory utilization (% of node) | % |
| K8sContainerRateMemUsageRequest | Memory utilization (% of request) | % |
| K8sContainerRateMemUsageLimit | Memory utilization (% of limit) | % |
| K8sContainerMemNoCacheBytes | Memory usage (excluding cache) | MB |
| K8sContainerRateMemNoCacheNode | Memory utilization (% of node, excluding cache) | % |
| K8sContainerRateMemNoCacheRequest | Memory utilization (% of request, excluding cache) | % |
| K8sContainerRateMemNoCacheLimit | Memory utilization (% of limit, excluding cache) | % |
| K8sContainerFsReadBytes | Block device read bandwidth | MB/s |
| K8sContainerFsWriteBytes | Block device write bandwidth | MB/s |
| K8sContainerFsReadTimes | Number of writes on the block device | Times |
| K8sContainerFsWriteTimes | Number of reads from the block device | Times |
| K8sContainerGpuUsed | GPU usage | MB |
| K8sContainerRateGpuUsedNode | GPU utilization (% of node) | % |
| K8sContainerRateGpuUsedRequest | GPU utilization (% of request) | % |
| K8sContainerGpuMemoryUsedBytes | GPU memory usage | MB |
| K8sContainerRateGpuMemoryUsedNode | GPU memory utilization (% of node) | % |
| K8sContainerRateGpuMemoryUsedRequest | GPU memory utilization (% of request) | % |
| K8sContainerNetworkReceiveBytes | Inbound network traffic | MB |

| Metric ID | Metric Name | Unit |
| --- | --- | --- |
| K8sContainerNetworkTransmitBytes | Outbound network traffic | MB |
| K8sContainerNetworkReceiveBytesBw | Inbound network bandwidth | MB/s |
| K8sContainerNetworkTransmitBytesBw | Outbound network bandwidth | MB/s |
| K8sContainerNetworkReceivePackets | Number of inbound network packets | Counts/ second |
| K8sContainerNetworkTransmitPackets | Number of outbound network packets | Counts/ second |
| K8sContainerRootfsUsed | rootfs space usage | GB |

# Monitoring Metrics for Cluster Components

| Metric Name | Unit |
| --- | --- |
| Whether kube-apiserver is normal | - |
| Whether kube-controller-manager is normal | - |
| Whether etcd is normal | - |
| Whether etcd is normal | - |

# Monitoring Metrics for BMS

Cloud Monitor provides monitoring metrics for BMS in the following dimensions:

## Metrics for Basic Monitoring

| Metric ID | Metric/Event Name | Unit | Metric Description |
| --- | --- | --- | --- |
| cpu_usage | CPU utilization | % | CPU utilization |
| cpu_loadavg | 1-minute average load of CPU | - | Average CPU load within 1 minute, calculated using the data of the first column in /proc/loadavg (this metric is not available on Windows operating systems), dependent on the monitoring component installation for collection |
| MemUsed | Memory usage | MB | Memory usage |
| MemUsage | Memory utilization | % | Memory utilization |
| TcpCurrEstab | Number of TCP connections | Count | Number of TCP connections |

## Metrics for Network Monitoring

| Metric ID | Metric/Event Name | Unit | Metric Description |
| --- | --- | --- | --- |
| WanIntraffic | Inbound public network bandwidth | Mb/s | Average inbound traffic per second on public NIC |
| WanOuttraffic | Outbound public network bandwidth | Mb/s | Average outbound traffic per second on public NIC |
| WanInpkg | Number of inbound packets on the public network | Counts/ second | Average inbound packets per second on public NIC |
| WanOutpkg | Number of outbound packets on the public network | Counts/ second | Average outbound packets per second on public NIC |

# Metrics for Disk Monitoring

| Metric ID | Metric/Event Name | Unit | Metric Description |
|---|---|---|---|
| CvmDiskUsage | Disk utilization | % | Disk utilization monitoring. |
| DiskIOAwait | Disk I/O wait | - | Disk I/O wait time |
| DiskReadtraffic | Disk read traffic | Kb/s | Average disk read traffic per second |
| DiskWritetraffic | Disk write traffic | Kb/s | Average disk write traffic per second |
| DiskReadIops | Disk read IOPS | Count | Average disk write traffic per second |
| DiskWriteIops | Disk write IOPS | Count | Average disk read I/O operations per second |

# Metrics for Private Network Monitoring

| Metric/Event Name | Unit |
|---|---|
| bond1 inbound packet loss on the private network | Count |
| bond1 inbound error packets on the private network | Count |
| bond1 inbound packets on the private network | Count/s |
| bond1 inbound bandwidth on the private network | Byte/s |
| bond1 outbound packet loss on the private network | Count |
| bond1 outbound error packets on the private network | Count |
| bond1 outbound packets on the private network | Count/s |
| bond1 outbound bandwidth on the private network | Byte/s |
| eth0 inbound packet loss on the private network | Count |
| eth0 inbound error packets on the private network | Count |
| eth0 inbound packets on the private network | Count/s |
| eth0 inbound bandwidth on the private network | Byte/s |
| eth0 outbound packet loss on the private network | Count |
| eth0 outbound error packets on the private network | Count |

| Metric/Event Name | Unit |
| --- | --- |
| eth0 outbound packets on the private network | Count/s |
| eth0 outbound bandwidth on the private network | Byte/s |
| eth1 inbound packet loss on the private network | Count |
| eth1 inbound error packets on the private network | Count |
| eth1 inbound packets on the private network | Count/s |
| eth1 inbound bandwidth on the private network | Byte/s |
| eth1 outbound packet loss on the private network | Count |
| eth1 outbound error packets on the private network | Count |
| eth1 outbound packets on the private network | Count/s |
| eth1 outbound bandwidth on the private network | Byte/s |

# Monitoring Metrics for CLB

## Monitoring Metrics for Listeners on Private Network

| Metric/Event ID | Metric Name | Unit |
| --- | --- | --- |
| intraffic | Inbound bandwidth | Mbps |
| outtraffic | Outbound bandwidth | Mbps |
| inpkg | Number of inbound packets | Counts/second |
| outpkg | Number of outbound packets | Counts/second |

Cloud Monitor provides monitoring metrics for CLB in the following dimensions:

## Monitoring Metrics for Listeners on Public Network

| Metric/Event ID | Metric Name | Unit |
| --- | --- | --- |
| intraffic | Inbound bandwidth | Mbps |
| outtraffic | Outbound bandwidth | Mbps |
| inpkg | Number of inbound packets | Counts/second |
| outpkg | Number of outbound packets | Counts/second |

## Monitoring Metrics for Server Ports

| Metric/Event ID | Metric Name | Unit |
| --- | --- | --- |
| re_port_status | Abnormal server port status | - |
| health_status | Number of normal RS for health check | Count |

# Monitoring Metrics for VPC

Cloud Monitor provides monitoring metrics for VPC in the following dimensions:

## Monitoring Metrics for the NAT Gateway

| Metric/Event ID | Metric Name | Unit | Metric Description |
|---|---|---|---|
| WanOutBandwidth | Outbound public network bandwidth | MB/s | Average outbound traffic per second on the NAT gateway |
| WanInBandwidth | Inbound public network bandwidth | MB/s | Average inbound traffic per second on the NAT gateway |
| Outpkg | Number of outbound packets on the public network | Counts/ second | Average outbound packets per second on the NAT gateway |
| Inpkg | Number of inbound packets on the public network | Counts/ second | Average inbound packets per second on the NAT gateway |
| WanConnCount | Public network connections | Count | Real-time concurrent connections on the NAT gateway |
| Droppkg | Number of dropped packets | Count | Number of lost data packets |

## Monitoring Metrics for the DC Gateway

| Metric/Event ID | Metric Name | Unit | Metric Description |
|---|---|---|---|
| OutBandwidth | Outbound network bandwidth | MB/s | Average outbound traffic per second on the DC gateway |
| InBandwidth | Inbound network bandwidth | MB/s | Average inbound traffic per second on the DC gateway |
| Outpkg | Number of outbound packets | Counts/ second | Average outbound packets per second on the DC gateway |
| Inpkg | Number of inbound packets | Counts/ second | Average inbound packets per second on the DC gateway |

# Monitoring Metrics for the Network Probe

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| TimeOut | Latency | ms | Network latency |
| RateDroppkg | Packet loss rate | % | Ratio of lost network data packets to total data packets |

# Monitoring Metrics for PC

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| OutBandwidth | Outbound network bandwidth | MB/s | Average outbound traffic per second on peering connection (PC) |
| InBandwidth | Inbound network bandwidth | MB/s | Average inbound traffic per second on PC |
| Outpkg | Number of outbound packets | Counts/ second | Average outbound packets per second on PC |
| Inpkg | Number of inbound packets | Counts/ second | Average inbound packets per second on PC |
| RateDroppkg | Packet loss rate | % | Ratio of packets dropped due to bandwidth limit to total packets on PC |

# Monitoring Metrics for the VPN Gateway

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| OutBandwidth | Outbound network bandwidth | MB/s | Average outbound traffic per second on the VPN gateway |
| InBandwidth | Inbound network bandwidth | MB/s | Average inbound traffic per second on the VPN gateway |
| Outpkg | Number of outbound packets | Counts/ second | Average outbound packets per second on the VPN gateway |

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| Inpkg | Number of inbound packets | Counts/ second | Average inbound packets per second on the VPN gateway |

# Monitoring Metrics for EIP

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| VipOutBandwidth | Outbound bandwidth | MB/s | Outbound bandwidth of EIP |
| VipInBandwidth | Inbound bandwidth | MB/s | Inbound bandwidth of EIP |
| VipOutpkg | Number of outbound packets | Counts/second | Outbound packets of EIP |
| VipInpkg | Number of inbound packets | Counts/second | Inbound packets of EIP |

# Monitoring Metrics for DC

Cloud Monitor provides the following monitoring metrics for DC:

## Connection Monitoring Metrics

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| InBandwidth | Inbound network bandwidth | MB/s | Inbound bandwidth data measured from the access point AR to the VPC, collected every 1 or 5 minutes |
| OutBandwidth | Outbound network bandwidth | MB/s | Outbound bandwidth data measured from the VPC to the access point AR, collected every 1 or 5 minutes |

## DC Tunnel Monitoring Metrics

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| InBandwidth | Inbound network bandwidth | MB/s | Average inbound traffic per second on the physical dedicated line |
| OutBandwidth | Outbound network bandwidth | MB/s | Average outbound traffic per second on the physical dedicated line |

# Monitoring Metrics for CBS

Cloud Monitor provides the following monitoring metrics for CBS:

## Monitoring Metrics for CBS

| Metric/Event ID | Metric Name | Unit | Metric Description |
|---|---|---|---|
| disk_read_iops | Disk read IOPS | Times | Average disk read operations per second |
| disk_write_iops | Disk write IOPS | Times | Average disk write operations per second |
| disk_read_traffic | Disk read traffic | KB/s | Average amount of data read from disk to memory per second |
| disk_write_traffic | Disk write traffic | KB/s | Average amount of data written from memory to disk per second |
| disk_await | Disk I/O wait time | ms | Average wait time per disk I/O operation |
| disk_svctm | Disk I/O service time | ms | Average time taken per disk I/O operation |
| disk_util | Busy time ratio of disk I/O | % | The percentage of time taken by I/O operations on the disk out of the total time |

## Monitoring Metrics for CBS Snapshots

| Metric/Event ID | Metric Name | Unit |
|---|---|---|
| snap_capacity_usage | Capacity used by CBS snapshots | GB |
| snap_count_usage | Number of used CBS snapshots | Count |

# Monitoring Metrics for COS

Cloud Monitor provides the following types of monitoring metrics for COS:

## Monitoring Metrics for Status Codes

| Metric/Event ID | Metric Name | Unit | Metric Description |
|---|---|---|---|
| 2xxResponse | 2xx status code | Count | Number of requests returned with 2xx status code |
| 2xxResponseRate | Percentage of 2xx status code | % | Percentage of requests returned with 2xx status code out of total requests |
| 3xxResponse | 3xx status code | Count | Number of requests returned with 3xx status code |
| 3xxResponseRate | Percentage of 3xx status code | % | Percentage of requests returned with 3xx status code out of total requests |
| 4xxResponse | 4xx status code | Count | Number of requests returned with 4xx status code |
| 4xxResponseRate | Percentage of 4xx status code | % | Percentage of requests returned with 4xx status code out of total requests |
| 400Response | 400 status code | Count | Number of requests returned with 400 status code |
| 400ResponseRate | Percentage of 400 status code | % | Percentage of requests returned with 400 status code out of total requests |
| 403Response | 403 status code | Count | Number of requests returned with 403 status code |
| 403ResponseRate | Percentage of 403 status code | % | Percentage of requests returned with 403 status code out of total requests |
| 404Response | 404 status code | Count | Number of requests returned with 404 status code |
| 404ResponseRate | Percentage of 404 status code | % | Percentage of requests returned with 404 status code out of total requests |
| 5xxResponse | 5xx status code | Count | Number of requests returned with 5xx status code |

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| 5xxResponseRate | Percentage of 5xx status code | % | Percentage of requests returned with 5xx status code out of total requests |
| 500Response | 500 status code | Count | Number of requests returned with 500 status code |
| 500ResponseRate | Percentage of 500 status code | % | Percentage of requests returned with 500 status code out of total requests |
| 501Response | 501 status code | Count | Number of requests returned with 501 status code |
| 501ResponseRate | Percentage of 501 status code | % | Percentage of requests returned with 501 status code out of total requests |
| 502Response | 502 status code | Count | Number of requests returned with 502 status code |
| 502ResponseRate | Percentage of 502 status code | % | Percentage of requests returned with 502 status code out of total requests |
| 503Response | 503 status code | Count | Number of requests returned with 503 status code |
| 503ResponseRate | Percentage of 503 status code | % | Percentage of requests returned with 503 status code out of total requests |
| 504Response | 504 status code | Count | Number of requests returned with 504 status code |

# Monitoring Metrics for Requests

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| StdReadRequests | Read requests for standard storage | Times | Number of read requests for the standard storage type, calculated based on the number of request commands sent |
| StdWriteRequests | Write requests for standard storage | Times | Number of write requests for the standard storage type, calculated based on the number of request commands sent |

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| LFReadRequests | Infrequent storage read requests | Times | Number of read requests for infrequent storage type, calculated based on the number of request commands sent |
| LFWriteRequests | Infrequent storage write requests | Times | Number of write requests for infrequent storage type, calculated based on the number of request commands sent |

# Monitoring Metrics for Traffic

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| UploadFlow | Upload traffic | Byte | Traffic from local to remote |
| DownloadFlow | Download traffic | Byte | Traffic from remote to local |

# Monitoring Metrics for Capacity

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| BucketCapacity | Bucket capacity | MB | Bucket capacity |
| BucketsUsedRate | Percentage of bucket quota used by the tenant | % | Percentage of used buckets out of total buckets |
| CapacityUsedRate | Percentage of capacity quota used by the tenant | % | Percentage of used capacity out of total capacity |
| ObjectsUsedRate | Percentage of object quota used by the tenant | % | Percentage of used objects out of total objects |
| ObjCapacityUsedRate | Percentage of used bucket capacity quota | % | Percentage of used bucket capacity out of total bucket capacity |
| ObjNumsUsedRate | Percentage of used bucket object quota | % | Percentage of used bucket objects out of total bucket objects |

# Monitoring Metrics for CSP

Cloud Monitor provides the following monitoring metrics for CSP:

| Metric/Event ID | Metric Name | Unit | Metric Description |
|---|---|---|---|
| 2xx_response | 2xx status code | Count | Number of requests returned with 2xx status code |
| 2xx_response_pct | Percentage of 2xx status code | % | - |
| 3xx_response | 3xx status code | Count | Number of requests returned with 3xx status code |
| 3xx_response_pct | Percentage of 3xx status code | % | - |
| 400_response | 400 status code | Count | Number of requests returned with 400 status code |
| 400_response_pct | Percentage of 400 status code | % | - |
| 403_response | 403 status code | Count | Number of requests returned with 403 status code |
| 403_response_pct | Percentage of 403 status code | % | - |
| 404_response | 404 status code | Count | Number of requests returned with 404 status code |
| 404_response_pct | Percentage of 404 status code | % | - |
| 4xx_response | 4xx status code | Count | Number of requests returned with 4xx status code |
| 4xx_response_pct | Percentage of 4xx status code | % | - |
| 500_response | 500 status code | Count | Number of requests returned with 500 status code |
| 500_response_pct | Percentage of 500 status code | % | - |

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| 501_response | 501 status code | Count | Number of requests returned with 501 status code |
| 501_response_pct | Percentage of 501 status code | % | - |
| 502_response | 502 status code | Count | Number of requests returned with 502 status code |
| 502_response_pct | Percentage of 502 status code | % | - |
| 503_response | 503 status code | Count | Number of requests returned with 503 status code |
| 503_response_pct | Percentage of 503 status code | % | - |
| 5xx_response | 5xx status code | Count | Number of requests returned with 5xx status code |
| 5xx_response_pct | Percentage of 5xx status code | % | - |
| bucket_storage | Bucket capacity | Bytes | Capacity of the bucket |
| bucket_obj | Number of bucket objects | Count | Number of objects in the bucket |
| inbound_traffic | Upload traffic | Bytes | - |
| outbound_traffic | Download traffic | Bytes | - |
| std_read_requests | Read requests for standard storage | Count | Number of read requests for the standard storage type, calculated based on the number of request commands sent |
| std_write_requests | Write requests for standard storage | Count | Number of write requests for the standard storage type, calculated based on the number of request commands sent |

# Monitoring Metrics for CFS

Cloud Monitor provides the following types of monitoring metrics for CFS:

## Monitoring Metrics for Bandwidth

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| DataReadIoBytes | Read bandwidth | KB/s | Average amount of data read from a file system per second |
| DataWriteIoBytes | Write bandwidth | KB/s | Average amount of data written to a file system per second |

## Monitoring Metrics for Read/Write Counts

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| DataReadIoCount | Read IOPS | Times/second | Read operations per second on the file system |
| DataWriteIoCount | Write IOPS | Times/second | Write operations per second on the file system |

## Monitoring Metrics for Storage

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| Storage | Storage capacity | GB | File system storage capacity at the current time |

## Monitoring Metrics for Latencies

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| DataReadIoLatency | Read latency | ms | Average read latency on the file system |
| DataWriteIoLatency | Write latency | ms | Average write latency on the file system |

# Other Monitoring Metrics

| Metric/Event ID | Metric Name | Unit | Metric Description |
|---|---|---|---|
| ConnClientsNums | Number of connected clients | Count | Number of connected clients |
| HealthCheckSuccessRate | Success rate of client health monitoring | % | Percentage of successful client health check attempts |

# Monitoring Metrics for Redis®

Cloud Monitor provides monitoring metrics for Redis® in the following dimensions:

## Monitoring Metrics for Community Edition

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| InFlow | Inbound traffic | MB | - |
| InFlowUtil | Inbound traffic utilization | % | Ratio of actual inbound traffic usage to maximum traffic |
| OutFlow | Outbound traffic | MB | - |
| OutFlowUtil | Outbound traffic utilization | % | Ratio of actual outbound traffic usage to maximum traffic |
| CPUUsedUtilMax | Maximum CPU utilization of cache shards | % | Peak CPU utilization of shards within the time window |
| MemUsedUtilMax | Maximum memory utilization of shards | % | Peak memory utilization of shards within the time window |
| MemUsed | Memory usage | MB | Memory capacity actually used, including data and cache |
| Connections | Number of connections | Count | Number of TCP connections to the instance |
| ConnectionsUtil | Connection utilization | % | Ratio of actual TCP connections to maximum connections |
| Commands | Total number of requests | Times | Total number of commands executed |
| CmdRead | Number of read requests | Times/ second | Number of read command executions per second |
| CmdWrite | Number of write requests | Times/ second | Number of write command executions per second |
| CmdValue | Number of large-value requests | Times/ second | Number of large-value command executions per second |

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| CmdOther | Number of other requests | Times/ second | Number of executions of commands other than read and write commands per second |
| CmdSLow | Number of slow queries | Count | Number of commands with the execution latency greater than the value of slowlog-log-slower-than |
| KeyNum | Total number of keys | Count | Total number of keys stored in an instance (first-level keys) |
| KeyExpired | Number of expired keys | Count | Number of keys evicted within the time window, corresponding to the evicted_keys metric generated by the info command |
| KeyEvicted | Number of evicted keys | Count | Number of keys evicted within the time window, corresponding to the expired_keys metric generated by the info command |
| CmdMiss | Read request misses | Count | Number of read requests where keys do not exist, corresponding to the keyspace_misses metric generated by the info command |
| CmdErr | Number of execution errors | Count | Number of command execution errors, such as nonexistent commands and parameter errors |

# Monitoring Metrics for Memory Edition (5-Second Granularity)

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| InFlow | Inbound traffic | MB/s | - |
| InFlowUtil | Inbound traffic utilization | % | Ratio of actual inbound traffic usage to maximum traffic |
| InFlowLimit | Throttling-triggered inbound traffic | Count | Number of times when inbound traffic triggers throttling |
| OutFlow | Outbound traffic | MB/s | - |

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| OutFlowUtil | Outbound traffic utilization | % | Ratio of actual outbound traffic usage to maximum traffic |
| OutFlowLimit | Throttling-triggered outbound traffic | Count | Number of times when outbound traffic triggers throttling |
| MemUsed | Memory usage | MB | Memory capacity actually used, including data and cache |
| MemUsedUtil | Memory utilization | % | Ratio of actual memory usage to total memory requested |
| CpuUsedUtil | CPU utilization | % | Percentage of CPU in non-idle status, calculated from /proc/stat data |
| MemUsedUtilMax | Maximum memory utilization of nodes | % | Peak memory utilization within the time window |
| CpuUsedUtilMax | Maximum node CPU utilization | % | Peak CPU utilization within the time window |
| Commands | Total number of requests | Times | Total number of commands executed |
| CmdRead | Number of read requests | Times/ second | Number of read command executions per second |
| CmdWrite | Number of write requests | Times/ second | Number of write command executions per second |
| CmdKey | Number of key requests | Times/ second | Number of key requests executed per second |
| CmdMget | Number of MGET requests | Times/ second | Number of MGET requests executed per second |
| CmdValue | Number of large-value requests | Times/ second | Number of large-value command executions per second |
| CmdOther | Number of other requests | Times/ second | Number of executions of commands other than read and write commands per second |
| CmdHits | Read request hits | Times | Number of read requests where keys exist, corresponding to the keyspace_hits metric generated by the info command |

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| CmdMiss | Read request misses | Count | Number of read requests where keys do not exist, corresponding to the keyspace_misses metric generated by the info command |
| CmdHitsRatio | Read request hit rate | % | Key hits/(Key hits + Key misses), which can reflect the cache miss status |
| CmdSLow | Number of slow queries | Count | Number of commands with the execution latency greater than the value of slowlog-log-slower-than |
| CmdErr | Number of execution errors | Count | Number of command execution errors, such as nonexistent commands and parameter errors |
| Connections | Number of connections | Count | Number of TCP connections to the instance |
| ConnectionsUtil | Connection utilization | % | Ratio of the actual number of TCP connections to the maximum number of connections |
| KeyNum | Total number of keys | Count | Total number of keys stored in an instance (first-level keys) |
| KeyEvicted | Number of evicted keys | Count | Number of keys evicted within the time window, corresponding to the expired_keys metric generated by the info command |
| KeyExpired | Number of expired keys | Count | Number of keys evicted within the time window, corresponding to the evicted_keys metric generated by the info command |
| LatencyRead | Average read latency | ms | Average execution latency of read commands from Proxy to the Redis® server |
| latency_write | Average write latency | ms | Average execution latency of write commands from Proxy to the Redis® server |
| LatencyAvgProxy | Average execution latency | ms | Average execution latency of requests from Proxy to the Redis® server |
| LatencyMax | Maximum execution latency | ms | Maximum execution latency of requests from Proxy to the Redis® server |
| LatencyOther | Average latency of other commands | ms | Average execution latency of commands other than read and write commands from |

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| | | | Proxy to the Redis® server |

# Monitoring Metrics for TDSQL for MySQL

Cloud Monitor provides the following monitoring metrics for TDSQL for MySQL cloud database instances:

## Monitoring Metrics for Nodes

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| DBNosurviveNode | Number of inactive DB nodes | - | Number of inactive DB nodes |
| ProxyNosurviveNode | Number of inactive proxy nodes | - | Number of inactive proxy nodes |

## Monitoring Metrics for Instances

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| ActiveThreadCount | Number of active threads | Count | A monitoring metric for instances, calculated by summing active threads on the primary and standby nodes of all shards |
| BinlogMinDiskAvailable | Minimum remaining Binlog disk space | GB | A monitoring metric for instances, calculated by summing the minimum values of the BinlogDiskAvailableShard metric across all shards |
| BinlogUsedDisk | Binlog disk space | GB | A monitoring metric for instances, calculated by summing the used Binlog disk space on the primary nodes of all shards |
| ConnUsageRate | Maximum DB connection utilization | % | A monitoring metric for instances, calculated as the maximum value of the DB connection utilization on the primary and standby nodes of all shards in the instance |

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| CpuUsageRate | Maximum CPU utilization | % | A monitoring metric for instances, calculated as the maximum value of the CPU utilization on the primary nodes of all shards in the instance |
| DataDiskAvailable | Available data disk space | GB | A monitoring metric for instances, calculated by summing the available data disk space on the primary nodes of all shards |
| DataDiskUsedRate | Maximum data disk space utilization | % | A monitoring metric for instances, calculated as the maximum data disk space utilization on the primary nodes of all shards in the instance |
| DeleteTotal | Number of DELETE requests | Times/ second | A monitoring metric for instances, calculated by summing the number of DELETE requests on the primary nodes of all shards in the instance |
| IOUsageRate | Maximum I/O utilization | % | A monitoring metric for instances, calculated as the maximum I/O utilization on the primary nodes of all shards in the instance |
| InnodbBufferPoolReads | Number of InnoDB disk page reads | Times | A monitoring metric for instances, calculated by summing the read page counts in the InnoDB disk on all primary and standby nodes of all shards in the instance |
| InnodbBufferPoolReadAhead | Number of buffer pool page reads ahead in InnoDB | Times | A monitoring metric for instances, calculated by summing the number of page reads ahead in the InnoDB buffer pool on all primary and standby nodes of all shards in the instance |

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| InnodbBufferPoolReadRequests | Number of page reads in the InnoDB buffer pool | Times | A monitoring metric for instances, calculated by summing the number of page reads in the InnoDB buffer pool on all primary and standby nodes of all shards in the instance |
| InnodbRowsDeleted | Number of deleted rows in InnoDB | Rows | A monitoring metric for instances, calculated by summing the number of deleted rows in InnoDB on the primary nodes of all shards in the instance |
| InnodbRowsInserted | Number of rows inserted in InnoDB | Rows | A monitoring metric for instances, calculated by summing the number of inserted rows in InnoDB on the primary nodes of all shards in the instance |
| InnodbRowsRead | Number of rows read in InnoDB | Rows | A monitoring metric for instances, calculated by summing the number of rows read in InnoDB on all primary and standby nodes of all shards in the instance |
| InnodbRowsUpdated | Number of rows updated in InnoDB | Rows | A monitoring metric for instances, calculated by summing the number of updated rows in InnoDB on the primary nodes of all shards in the instance |
| InsertTotal | Number of INSERT requests | Times/ second | A monitoring metric for instances, calculated by summing the number of INSERT requests on the primary nodes of all shards in the instance |
| LongQueryCount | Number of slow queries | Times | A monitoring metric for instances, calculated by summing the number of slow queries on the primary nodes of |

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| | | | all shards in the instance |
| MemAvailable | Available cache space | GB | A monitoring metric for instances, calculated by summing the available cache space on the primary nodes of all shards in the instance |
| MemHitRate | Cache hit rate | % | A monitoring metric for instances, calculated as the minimum value of the cache hit rate on the primary nodes of all shards in the instance. |
| ReplaceSelectTotal | Number of REPLACE_SELECT requests | Times/ second | A monitoring metric for instances, calculated by summing the number of REPLACE_SELECT requests on the primary nodes of all shards in the instance. |
| ReplaceTotal | Number of REPLACE requests | Times/ second | A monitoring metric for instances, calculated by summing the number of REPLACE requests on the primary nodes of all shards in the instance. |
| RequestTotal | Total number of requests | Times/ second | A monitoring metric for instances, calculated by summing the number of requests on all primary and standby nodes in the instance. |
| SelectTotal | Number of SELECT requests | Times/ second | A monitoring metric for instances, calculated by summing the number of SELECT requests on all primary and standby nodes of all shards in the instance. |
| SlaveDelay | Secondary latency | Seconds | A monitoring metric for instances. The standby delay of an instance is the maximum standby delay of all shards in the instance, with the standby delay |

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| | | | of each shard calculated first. The standby delay of a shard is the minimum standby delay of all the standby nodes of the shard. |
| UpdateTotal | Number of UPDATE requests | Times/ second | A monitoring metric for instances, calculated by summing the number of UPDATE requests on the primary nodes of all shards in the instance. |
| ThreadsConnected | Number of open connections | Times | A monitoring metric for instances, calculated by summing the number of current open connections on all primary and standby nodes of all shards in the instance. |
| ConnMax | Maximum number of connections | Count | A monitoring metric for instances, calculated by summing the maximum connections on all primary and standby nodes of all shards in the instance. |
| ClientConnTotal | Total number of client connections | Count | A monitoring metric for instances, calculated by summing the number of connections on all proxies in the instance. This metric accurately indicates how many of your clients are connected to the database instance. |
| SQLTotal | Total number of SQL queries | Count | A monitoring metric for instances, indicating the number of SQL statements sent to the database instance. |
| ErrorSQLTotal | Number of SQL errors | Count | A monitoring metric for instances, indicating the number of SQL statements executed with errors. |

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| SuccessSQLTotal | Number of successful SQL queries | Count | A monitoring metric for instances, indicating the number of successfully executed SQL statements. |
| TimeRange0 | Number of requests with execution time less than 5 ms | Times/ second | A monitoring metric for instances, indicating the number of requests with execution time less than 5ms. |
| TimeRange1 | Number of requests with execution time between 5 ms and 20 ms | Times/ second | A monitoring metric for instances, indicating the number of requests with execution time between 5 ms and 20 ms |
| TimeRange2 | Number of requests with execution time between 20 ms and 30 ms | Times/ second | A monitoring metric for instances, indicating the number of requests with execution time between 20 ms and 30 ms |
| TimeRange3 | Number of requests with execution time more than 30 ms | Times/ second | A monitoring metric for instances, indicating the number of requests with execution time more than 30 ms |
| MasterSwitchedTotal | Number of primary-secondary switches | Times | A monitoring metric for instances, indicating the number of primary-secondary switchovers that occurred for the instance |

# Monitoring Metrics for MongoDB

Cloud Monitor provides the following dimensions of monitoring metrics for MongoDB:

## Monitoring Metrics for the Replica Sets

| Metric Name | Unit | Metric Description |
| --- | --- | --- |
| Cache dirty data percentage | % | Percentage of dirty data volume to total data volume in the cache |
| Cache utilization | % | Percentage of used cache space to total cache space |
| Disk utilization | % | - |
| Cache hit rate | % | Percentage of cache hits to total access times |
| Oplog retention time | Hour | - |
| Average primary/secondary node latency per unit of time | s | Average latency of primary/secondary nodes per unit of time |

## Monitoring Metrics for Nodes

| Metric Name | Unit | Metric Description |
| --- | --- | --- |
| CPU utilization | % | - |
| Memory utilization | % | - |
| Number of read requests in the waiting queue | Count | - |
| Number of write requests in the waiting queue | Count | - |
| Inbound network traffic | MB/s | Average inbound traffic per second |
| Outbound network traffic | MB/s | Average outbound traffic per second |
| Number of connections | Count | - |
| Node disk usage | MB | Disk space size used by a single node |

| Metric Name | Unit | Metric Description |
| --- | --- | --- |
| WT engine's ActiveRead | Count | Number of active read operations in the WT engine |
| WT engine's ActiveWrite | Count | Number of active write operations in the WT engine |
| Number of data entries deleted after TTL is triggered | Count | - |
| Number of TTL running rounds | Count | Number of rounds of data cleanup operations triggered based on the TTL index |

# Monitoring Metrics Instances

| Metric Name | Unit | Metric Description |
| --- | --- | --- |
| Number of write requests | Times | Number of write requests initiated |
| Number of read requests | Times | Number of read requests initiated |
| Number of update requests | Times | Number of update requests initiated |
| Number of deletion requests | Times | Number of deletion requests initiated |
| Number of count requests | Times | Number of count requests initiated |
| Number of aggregation requests | Times | Number of aggregation requests initiated |
| Number of command requests | Times | Number of command requests initiated |
| Number of successful requests | Times | Number of requests that succeeded and received the expected response |
| Disk utilization | % | - |
| Number of requests with unit time latency (10 ms to 50 ms) | Times | Number of requests with latency between 10 ms and 50 ms |
| Number of requests with unit time latency (50 ms to 100 ms) | Times | Number of requests with latency between 50 ms and 100 ms |
| Number of requests with unit time latency (over 100 ms) | Times | Number of requests with latency over 100 ms |
| Connection utilization | % | - |
| Number of connections | Count | - |

# Monitoring Metrics for Instances (Replica Set Instance for Versions Above MongoDB 4.0)

| Metric Name | Unit | Metric Description |
| --- | --- | --- |
| Number of write requests | Times | - |
| Number of read requests | Times | - |
| Number of update requests | Times | - |
| Number of deletion requests | Times | - |
| Number of successful requests | Times | - |
| Disk utilization | % | - |
| Number of requests with unit time latency (10 ms to 50 ms) | Times | Number of requests with latency between 10 ms and 50 ms |
| Number of requests with unit time latency (50 ms to 100 ms) | Times | Number of requests with latency between 50 ms and 100 ms |
| Number of requests with unit time latency (over 100 ms) | Times | Number of requests with latency over 100 ms |
| Number of connections | Count | - |
| Connection percentage | % | - |

# Monitoring Metrics for CMQ

Cloud Monitor provides the following monitoring metrics for CMQ:

| Metric/Event ID | Metric Name | Unit |
| --- | --- | --- |
| msg_heap_num | Number of heaped messages | Count |
| msg_product_rate | Message production rate | Count/s |
| msg_product_flow | Message production traffic | Byte/s |
| msg_consume_rate | Message consumption rate | Count/s |
| msg_consume_flow | Message consumption traffic | Byte/s |
| invisible_msg_Limit_count | Upper limit of invisible messages | Count |
| batch_delete_message_rate | Batch deletion request rate | Count/s |
| batch_product_message_rate | Batch production rate | Count/s |
| batch_consume_message_rate | Batch consumption request rate | Count/s |
| delete_message_rate | Deletion request rate | Count/s |
| product_message_rate | Production request rate | Count/s |
| consume_message_rate | Consumption request rate | Count/s |

# Monitoring Metrics for Ckafka

Cloud Monitor provides the following monitoring metrics for Ckafka:

## Monitoring Metrics for Topics

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| CtopicProFlow | Topic produced traffic | MB | Topic production traffic (excluding traffic generated by replicas), summed statistically according to the selected time granularity |
| CtopicProFlowMax | Maximum traffic of topic production | MB | Peak traffic of topic production |
| CtopicProCount | Number of messages produced for the topic | Count | The actual number of messages produced for the topic, summed statistically according to the selected time granularity |
| CtopicConFlow | Topic consumption traffic | MB | Topic consumption traffic (excluding traffic generated by replicas), summed according to the selected time granularity |
| CtopicConFlowMax | Maximum traffic of topic consumption | MB | Peak traffic of topic consumption |
| CtopicConCount | Number of messages consumed for the topic | Count | The number of messages consumed for the topic, summed statistically according to the selected time granularity |
| CtopicMsg | Total disk usage of messages for the topic | MB | Total disk usage of messages for the topic |
| CtopicMsgCount | Total number of messages written to the disk for the topic | Count | Total number of messages written to the disk for the topic (excluding replicas), calculated as the latest value according to the selected time granularity |

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| CtopicMsgRequestProCount | Number of requests produced for the topic | Count | The actual number of requests produced for the topic |
| CtopicMsgRequestConCount | Number of requests consumed for the topic | Count | The actual number of requests consumed for the topic |

# Monitoring Metrics for Instances

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| InstanceProCount | Number of messages produced for the instance | Count | The actual number of messages produced for the instance, summed statistically according to the selected time granularity |
| InstanceConCount | Number of messages consumed for the instance | Count | The number of messages consumed for the instance, summed statistically according to the selected time granularity |
| InstanceProTrafficMax | Peak bandwidth of messages produced for the instance | MB/s | - |
| RequestProCount | Number of requests produced for the instance | Count | The actual number of requests produced for the instance |
| RequestConCount | Number of requests consumed for the instance | Count | The actual number of requests consumed for the instance |
| InstanceMsgCount | Total number of messages written to disk for the instance | Count | The number of messages written to the disk for the instance (excluding replicas), calculated as the latest value according to the selected time granularity |
| InstanceProFlow | Instance production traffic | MB/ min | Instance production traffic (excluding traffic generated by replicas), summed statistically according to the selected time |

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| | | | granularity |
| InstanceMaxProFlow | Maximum production traffic | MB/s | The peak traffic of messages produced for the instance (excluding bandwidth generated by replica production) |
| InstanceProTrafficRate | Instance production bandwidth percentage | % | - |
| InstanceProLimitNums | Instance production throttling frequency | Count | - |
| InstanceConFlow | Instance consumption traffic | MB/ min | Instance consumption traffic (excluding traffic generated by replicas), summed statistically according to the selected time granularity |
| InstanceMaxConFlow | Maximum consumption traffic | MB/s | The peak traffic of messages consumed for the instance (no concept of replicas during consumption) |
| InstanceConTrafficRate | Instance consumption bandwidth percentage | % | - |
| InstanceConLimitNums | Instance consumption throttling frequency | Count | The throttling frequency of messages consumed for the instance |
| InstanceConGroupNums | Number of consumer groups for the instance | Count | Number of consumer groups |
| InstanceConGroupNumsRate | Consumer group percentage | % | - |
| InstanceDiskUsage | Disk usage percentage | % | The percentage of current disk usage to the total disk capacity of the instance specification |
| InstanceMsgHeap | Disk usage of the instance | MB | Disk usage of the instance (including replicas), calculated as |

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| | | | the latest value according to the selected time granularity |
| InstanceConnectionCount | Number of instance connections | Count | The current total number of connections of the instance |
| InstanceConnectionRate | Instance connection percentage | % | The ratio of the current total number of connections to the total number of instances |
| PartitionNums | Number of instance partitions | Count | - |
| PartitionNums | Instance partition percentage | % | - |
| TopicNums | Number of instance topics | Count | - |
| TopicNums | Instance topic percentage | % | - |

# Monitoring Metrics for Consumer Groups

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| CgroupTopicMaxOffset | Maximum offset of the consumer group under the current topic | Count | Maximum offset of the consumer group under the current topic |
| CgroupTopicMsgOffset | Current consumption offset of the consumer group | Count | Current offset of the consumer group |
| CgroupTopicUnconsumeMsgCount | Number of unconsumed messages under the consumer group | Count | Number of unconsumed messages under the consumer group |
| CgroupTopicMsgRate | Topic consumption rate | Count/min | Message consumption speed under the consumer group |
| CgroupTopicUnconsumeMsgSize | Size of unconsumed messages under the | MB | Size of unconsumed messages under the |

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| | consumer group | | consumer group |
| CgroupPartMaxOffset | Maximum offset of the consumer group | Count | Maximum offset under the consumer group |
| CgroupPartOffset | Partition offset | Count | Offset under the partition |
| CgroupPartUnconsumeMsgCount | Number of unconsumed messages in the partition | Count | Number of unconsumed messages under the partition |
| CgroupPartMsgRate | Partition consumption speed | Count/ min | Message consumption speed under the partition |
| CgroupPartUnconsumeMsgSize | Backlog of unconsumed messages in the partition | MB | Backlog of unconsumed messages under the partition |

# Monitoring Metrics for Partitions

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| CpartitionProFlow | Partition production traffic | MB | Partition production traffic (excluding traffic generated by replicas), summed statistically according to the selected time granularity |
| CpartitionConFlow | Partition consumption traffic | MB | Partition consumption traffic (excluding traffic generated by replicas), summed statistically according to the selected time granularity |

# Monitoring Metrics for TSF

Cloud Monitor provides the following monitoring metrics for TSF:

## Monitoring Metrics for Services

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| RequestAverageTime | Average time consumed for received requests | ms | The average processing time consumed by the requested service |
| RequestErrorPercentage | Failure rate of received requests | % | Percentage of abnormal results from the requested service |
| RequestCount | Volume of received requests | Times | Number of times the service was requested |
| Response4xx | HTTP 4xx response code | Count | Number of requests with 4xx response code |
| Response5xx | HTTP 5xx response code | Count | Number of requests with 5xx response code |

## Monitoring Metrics for Deployment Groups

| Metric/Event ID | Metric Name | Unit | Metric Description |
| --- | --- | --- | --- |
| HealthyInstancePercentage | Health rate of deployment group nodes | % | Percentage of healthy nodes in a TSF component |

## Monitoring Metrics for APIs

| Metric/Event ID | Metric Name | Unit |
| --- | --- | --- |
| ReqCount | Number of requests | Count |
| TimeSpentAvg | Average duration | ms |
| FailureRate | Failure rate | % |

# Monitoring Metrics for Logs

| Metric/Event ID | Metric Name | Unit |
| --- | --- | --- |
| KeyNums | Keyword occurrence count | Count |

# Monitoring Metrics for TCT

| Metric/Event ID | Metric Name | Unit |
| --- | --- | --- |
| FailureTimes | Number of execution failures | Count |

# Monitoring Metrics for Instances

| Metric/Event ID | Metric Name | Unit |
| --- | --- | --- |
| full_gcEventNums | FULL_GC event statistics metric | Count |

# Monitoring Metrics for TDMQ Pulsar

Cloud Monitor provides the following dimensions of monitoring metrics for TDMQ Pulsar:

## Monitoring Metrics for the Subscription

| Metric Name | Unit | Metric Description |
| --- | --- | --- |
| Number of heaped messages | count | Number of heaped messages |
| Percentage of quota used by unacknowledged messages | % | Ratio of unacknowledged messages to the total quota |
| Message consumption rate | count/s | - |
| Amount of consumed traffic | Byte/s | - |
| Number of unacknowledged messages | Count | Number of unacknowledged messages |
| Number of delayed messages | Count | Number of delayed messages |
| Number of consumers | Count | - |

## Monitoring Metrics for Topics

| Metric Name | Unit | Metric Description |
| --- | --- | --- |
| Message production rate | count/s | - |
| Message production traffic | Byte/s | - |
| Message consumption rate | count/s | - |
| Message consumption traffic | Byte/s | - |
| Message storage size | Count | - |
| Number of consumers | Count | - |
| Number of producers | Count | - |

| Metric Name | Unit | Metric Description |
| --- | --- | --- |
| Quota usage percentage of message production traffic | % | The ratio of message production traffic to the total quota |
| Quota usage percentage of message consumption traffic | % | The ratio of message consumption traffic to the total quota |
| Quota usage percentage of message production count | % | The ratio of produced message count to the total quota |
| Quota usage percentage of message consumption count | % | The ratio of consumed message count to the total quota |
| Number of heaped messages | Count | - |
| Heaped message size | Byte | - |
| Producer count throttling times | Count | - |
| Quota usage percentage of heaped messages | % | The ratio of heaped messages to the total quota |

# Monitoring Metrics for Clusters

| Metric Name | Unit | Metric Description |
| --- | --- | --- |
| Tenant-level message production rate | count/s | - |
| Tenant-level message consumption rate | count/s | - |
| Peak cluster production bandwidth | count/s | - |
| Peak cluster consumption bandwidth | Byte/s | - |
| Tenant-level heaped message size | Byte | - |
| Transactions per second (TPS) of the Pulsar cluster | Count/s | The number of messages processed per second by the cluster |

# Monitoring Metrics for TI-ONE Platform
# TI-ONE Task-Based Modeling

| Metric English Name | Metric Description | Unit | Dimension | Statistical Granularity |
|---|---|---|---|---|
| CfsClientDataReadBandwidth | Instance-dimension TuroCFS single-node server read bandwidth | KBytes/s | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| CfsClientDataReadBandwidth | Instance-dimension TuroCFS single-node server read bandwidth | KBytes/s | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| CfsClientDataWriteBandwidth | Instance-dimension TuroCFS single-node server write bandwidth | KBytes/s | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| CfsDataReadIoBytes | Instance-dimension CFS server read bandwidth | KBytes/s | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| CfsDataReadIoLatency | Instance-dimension CFS read latency | ms | AppId<br><br>InstanceId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg |

| | | | | |
|---|---|---|---|---|
| | | | SubUin | ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| CfsDataWriteIoBytes | Instance-dimension CFS server write bandwidth | KBytes/s | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| CfsDataWriteIoLatency | Instance-dimension CFS write latency | ms | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| CfsStrageUsageGb | Instance-dimension CFS storage data capacity | GBytes | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Cpuutil | Task-dimension CPU utilization | % | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DcgmFiDevFbUsed | GPU memory usage in the task instance-GPU card dimension | MBytes | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ] |

| | | | | |
|---|---|---|---|---|
| | | | | [ 86400s, avg ] |
| DcgmFiDevGpuUtil | GPU utilization in the task instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DcgmFiDevMemCopyUtil | GPU memory utilization in the task instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DiskIoUtil | Instance-dimension disk I/O utilization | % | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DiskIoWait | Instance-dimension disk I/O wait time | % | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DiskReadByte | Instance-dimension disk read bandwidth | MBytes/s | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| DiskReadIops | Instance-dimension disk read IOPS | Count | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DiskUsageRadio | Instance-dimension system disk partition utilization | % | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DiskWriteByte | Instance-dimension disk write bandwidth | MBytes/s | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DiskWriteIops | Instance-dimension disk write IOPS | Count | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Fp16EngineActivity | Instance-dimension FP16 active time ratio | % | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Fp32EngineActivity | Instance-dimension FP32 | % | AppId<br>InstanceId | [ 10s, avg ]<br>[ 60s, avg ] |

| | | | | |
|---|---|---|---|---|
| | active time ratio | | SubUin | [ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Fp64EngineActivity | Instance-dimension FP64 active time ratio | % | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuFp16EngineActivity | FP16 active time ratio in the task instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuFp32EngineActivity | FP32 active time ratio in the task instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuFp64EngineActivity | FP64 active time ratio in the task instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Gpumemutil | Task-dimension GPU memory utilization | % | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, |

| | | | | |
|---|---|---|---|---|
| | | | | avg ]<br>[ 86400s, avg ] |
| Gpumemvalue | Task-dimension GPU memory usage | MBytes | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuNvlinkBandwidth | NVLink transmission rate in the task instance-GPU card dimension | Bytes/s | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuPcieBandwidth | PCIe bus transmission rate in the task instance-GPU card dimension | Bytes/s | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuSmActivity | SM active time ratio in the task instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| GpuTensorActivity | Tensor active time ratio in the task instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Gpuutil | Task-dimension GPU utilization | % | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Instancecpuutil | Instance-dimension CPU utilization | % | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Instancegpumemutil | Instance-dimension GPU memory utilization | % | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Instancegpumemvalue | Instance-dimension GPU memory usage | MBytes | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Instancegpuutil | GPU utilization in the task | % | AppId<br>InstanceId | [ 10s, avg ]<br>[ 60s, avg ] |

| | | | | |
|---|---|---|---|---|
| | instance dimension | | SubUin | [ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Instancememutil | Instance-dimension memory utilization | % | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Instancememvalue | Instance-dimension memory usage | MBytes | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Memutil | Task-dimension memory utilization | % | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Memvalue | Task-dimension memory usage | MBytes | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| NvlinkBandwidth | Instance-dimension NVLink transmission rate | Bytes/s | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| PcieBandwidth | Instance-dimension PCIe bus transmission rate | Bytes/s | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| RdmaInpkt | Instance-dimension RDMA NIC inbound packet count | pps | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| RdmaIntraffic | Instance-dimension RDMA NIC receive bandwidth | Mbps | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| RdmaOutpkt | Instance-dimension RDMA NIC outbound packet count | pps | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| RdmaOuttraffic | Instance-dimension | Mbps | AppId<br>InstanceId | [ 10s, avg ]<br>[ 60s, avg ] |

| | | | | |
|---|---|---|---|---|
| | RDMA NIC transmit bandwidth | | SubUin | [ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| SmActivity | Instance-dimension SM active time ratio | % | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskCfsClientDataReadBandwidth | Task-dimension TuroCFS single-node server read bandwidth | KBytes/s | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskCfsClientDataWriteBandwidth | Task-dimension TuroCFS single-node server write bandwidth | KBytes/s | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskCfsDataReadIoBytes | Task-dimension CFS server read bandwidth | KBytes/s | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskCfsDataReadIoLatency | Task-dimension CFS read latency | ms | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, |

| | | | | |
|---|---|---|---|---|
| | | | | avg ]<br>[ 86400s, avg ] |
| TaskCfsDataWriteIoBytes | Task-dimension CFS server write bandwidth | KBytes/s | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskCfsDataWriteIoLatency | Task-dimension CFS write latency | ms | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskCfsStrageUsageGb | Task-dimension CFS storage data capacity | GBytes | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskDiskIoUtil | Task-dimension disk I/O utilization | % | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| TaskDiskIoWait | Task-dimension disk I/O wait time | % | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskDiskReadByte | Task-dimension disk read bandwidth | MBytes/s | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskDiskReadIops | Task-dimension disk read IOPS | Count | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskDiskUsageRadio | Task-dimension system disk partition utilization | % | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskDiskWriteByte | Task-dimension disk write bandwidth | MBytes/s | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskDiskWriteIops | Task-dimension disk write IOPS | Count | AppId<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ] |

| | | | | |
|---|---|---|---|---|
| | | | SubUin | [ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskFp16EngineActivity | Task-dimension FP16 active time ratio | % | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskFp32EngineActivity | Task-dimension FP32 active time ratio | % | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskFp64EngineActivity | Task-dimension FP64 active time ratio | % | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskNvlinkBandwidth | Task-dimension NVLink transmission rate | Bytes/s | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskPcieBandwidth | Task-dimension PCIe bus transmission rate | Bytes/s | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, |

| | | | | |
|---|---|---|---|---|
| | | | | avg ]<br>[ 86400s, avg ] |
| TaskRdmaInpkt | Task-dimension RDMA NIC inbound packet count | pps | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskRdmaIntraffic | Task-dimension RDMA NIC receive bandwidth | Mbps | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskRdmaOutpkt | Task-dimension RDMA NIC outbound packet count | pps | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskRdmaOuttraffic | Task-dimension RDMA NIC transmit bandwidth | Mbps | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| TaskSmActivity | Task-dimension SM active time ratio | % | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskTensorActivity | Task-dimension Tensor active time ratio | % | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TensorActivity | Instance-dimension Tensor active time ratio | % | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuDecUtil | Instance-dimension GPU decoder utilization | % | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuEncUtil | Instance-dimension GPU encoder utilization | % | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuMemoryClock | Instance-dimension GPU | s | AppId<br>InstanceId | [ 10s, avg ]<br>[ 60s, avg ] |

| | | | | |
|---|---|---|---|---|
| | memory frequency | | SubUin | [ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuMemoryFree | Instance-dimension available GPU memory | MBytes | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuMemoryUtil | Instance-dimension GPU memory utilization | % | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuNvlinkRxMb | Instance-dimension NVLink received data volume | Mbps | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuNvlinkTxMb | Instance-dimension NVLink transmitted data volume | Mbps | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuPcieRxMb | Instance-dimension PCIe received data volume | Mbps | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, |

| | | | | |
|---|---|---|---|---|
| | | | | avg ]<br>[ 86400s, avg ] |
| GpuPcieTxMb | Instance-dimension PCIe transmitted data volume | Mbps | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuSmClock | Instance-dimension SM clock frequency | s | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskGpuDecUtil | Task-dimension GPU decoder utilization | % | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskGpuEncUtil | Task-dimension GPU encoder utilization | % | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| TaskGpuMemoryClock | Task-dimension GPU memory frequency | s | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskGpuMemoryFree | Task-dimension available GPU memory | MBytes | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskGpuMemoryUtil | Task-dimension GPU memory bandwidth utilization | % | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskGpuNvlinkRxMb | Task-dimension NVLink received data volume | Mbps | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskGpuNvlinkTxMb | Task-dimension NVLink transmitted data volume | Mbps | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskGpuPcieRxMb | Task-dimension PCIe received | Mbps | AppId<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ] |

| | | | | |
|---|---|---|---|---|
| | data volume | | SubUin | [ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskGpuPcieTxMb | Task-dimension PCIe transmitted data volume | Mbps | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskGpuSmClock | Task-dimension SM clock frequency | s | AppId<br>TaskId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskGpuDecUtilGpu | GPU decoder utilization in the task instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskGpuEncUtilGpu | GPU encoder utilization in the task instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| TaskGpuMemoryClockGpu | GPU memory frequency in the task instance-GPU card dimension | s | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskGpuMemoryFreeGpu | Available GPU memory in the task instance-GPU card dimension | MBytes | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskGpuMemoryUtilGpu | GPU memory bandwidth utilization in the task instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskGpuNvlinkRxMbGpu | NVLink received data volume in the task instance-GPU card dimension | Mbps | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskGpuNvlinkTxMbGpu | NVLink transmitted data volume in the task instance-GPU card dimension | Mbps | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskGpuPcieRxMbGpu | PCIe received data volume in | Mbps | AppId<br>InstanceGpuNum | [ 10s, avg ]<br>[ 60s, avg ] |

| | | | | |
|---|---|---|---|---|
| | the task instance-GPU card dimension | | SubUin | [ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskGpuPcieTxMbGpu | PCIe transmitted data volume in the task instance-GPU card dimension | Mbps | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TaskGpuSmClockGpu | SM clock frequency in the task instance-GPU card dimension | s | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

# TI-ONE Development Machine

| Metric English Name | Metric Description | Unit | Dimension | Statistical Granularity |
|---|---|---|---|---|
| CfsClientDataReadBandwidth | Instance-dimension TuroCFS single-node server read bandwidth | KBytes/s | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| CfsClientDataWriteBandwidth | Instance-dimension TuroCFS single-node server write bandwidth | KBytes/s | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| CfsDataReadIoBytes | Instance-dimension CFS server read bandwidth | KBytes/s | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| CfsDataReadIoLatency | Instance-dimension CFS read latency | ms | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| CfsDataWriteIoBytes | Instance-dimension CFS server write bandwidth | KBytes/s | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| CfsDataWriteIoLatency | Instance-dimension CFS write latency | ms | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| CfsStrageUsageGb | Instance-dimension CFS storage data capacity | GBytes | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DiskIoUtil | Instance-dimension disk I/O utilization | % | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DiskIoWait | Instance-dimension disk I/O wait time | % | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DiskReadByte | Instance-dimension disk read bandwidth | MBytes/s | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DiskReadIops | Instance-dimension disk read IOPS | Count | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| DiskUsageRadio | Instance-dimension system disk partition utilization | % | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DiskWriteByte | Instance-dimension disk write bandwidth | MBytes/s | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DiskWriteIops | Instance-dimension disk write IOPS | Count | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Instancecpuutil | Instance-dimension CPU utilization | % | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Instancegpumemutil | Instance-dimension GPU memory utilization | % | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Instancegpumemvalue | Instance-dimension GPU memory usage | MBytes | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| Instancegpuutil | Instance-dimension GPU utilization | % | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Instancememutil | Instance-dimension memory utilization | % | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Instancememvalue | Instance-dimension memory usage | MBytes | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuFp16EngineActivity | FP16 active time ratio in the instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuFp32EngineActivity | FP32 active time ratio in the instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuFp64EngineActivity | FP64 active time ratio in the instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| NvlinkBandwidth | NVLink transmission rate in the instance-GPU card dimension | Bytes/s | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| PcieBandwidth | PCIe bus transmission rate in the instance-GPU card dimension | Bytes/s | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuSmActivity | SM active time ratio in the instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TensorActivity | Tensor active time ratio in the instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Dcgmfidevfbused | GPU memory usage in the instance-GPU card dimension | MBytes | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DcgmFiDevGpuUtil | GPU utilization in the instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| DcgmFiDevMemCopyUtil | GPU memory utilization in the instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuMemoryClockGpu | GPU memory frequency in the instance-GPU card dimension | s | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuMemoryFreeGpuv | Available GPU memory in the instance-GPU card dimension | MBytes | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuNvlinkRxMb | NVLink received data volume in the instance-GPU card dimension | Mbps | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuNvlinkTxMb | NVLink transmitted data volume in the instance-GPU card dimension | Mbps | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuPcieRxMb | PCIe received data volume in the instance-GPU card dimension | Mbps | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| GpuPcieTxMb | PCIe transmitted data volume in the instance-GPU card dimension | Mbps | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuSmClock | SM clock frequency in the instance-GPU card dimension | s | AppId<br>InstanceGpuNum<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| PodDiskLimit | Instance-dimension total disk size | GBytes | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| PodDiskValue | Instance-dimension disk usage | GBytes | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| NodeDiskLimit | Instance-dimension total node disk size | GBytes | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| NodeDiskValue | Instance-dimension node disk usage | GBytes | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| RdmaInpkt | Instance-dimension RDMA NIC inbound packet count | pps | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| RdmaOutpkt | Instance-dimension RDMA NIC outbound packet count | pps | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| RdmaIntraffic | Instance-dimension RDMA NIC receive bandwidth | Mbps | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| RdmaOuttraffic | Instance-dimension RDMA NIC transmit bandwidth | Mbps | AppId<br>InstanceId<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

# TI-ONE Online Services

| Metric English Name | Metric Description | Unit | Dimension | Statistical Granularity |
|---|---|---|---|---|
| Apicallerrortotal | Number of failed API calls | Count | ServiceGroupId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Apicalllimittotal | Total number of API calls throttled | Count | ServiceGroupId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Apicallsuccesstotal | Total number of successful API calls | Count | ServiceGroupId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Apicalltotal | Total number of API calls | Count | ServiceGroupId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Apiresponsetime | Average response time | ms | ServiceGroupId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ] |

| | | | | |
|---|---|---|---|---|
| | | | | [ 86400s, avg ] |
| CfsClientDataReadBandwidth | TuroCFS single-node server read bandwidth in the task instance dimension | KBytes/s | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| CfsClientDataWriteBandwidth | TuroCFS single-node server write bandwidth in the task instance dimension | KBytes/s | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| CfsDataReadIoBytes | CFS server read bandwidth in the task instance dimension | KBytes/s | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| CfsDataReadIoLatency | CFS read latency in the task instance dimension | ms | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| CfsDataWriteIoBytes | CFS server write bandwidth in the task instance dimension | KBytes/s | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| CfsDataWriteIoLatency | CFS write latency in the task instance dimension | ms | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| CfsStrageUsageGb | CFS storage data capacity in the task instance dimension | GBytes | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Cpuutil | Task-dimension CPU utilization | % | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DiskIoUtil | Disk I/O utilization in the task instance dimension | % | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DiskIoWait | Disk I/O wait time in the task instance dimension | % | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DiskReadByte | Disk read bandwidth in the task | MBytes/s | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg |

| | | | | |
|---|---|---|---|---|
| | instance dimension | | | ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DiskReadIops | Disk read IOPS in the task instance dimension | Count | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DiskUsageRadio | System disk partition utilization in the task instance dimension | % | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DiskWriteByte | Disk write bandwidth in the task instance dimension | MBytes/s | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DiskWriteIops | Disk write IOPS in the task instance dimension | Count | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Gpumemutil | Task-dimension GPU memory utilization | % | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, |

| | | | | |
|---|---|---|---|---|
| | | | | avg ] |
| Gpuutil | Task-dimension GPU utilization | % | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Instancecpuutil | CPU utilization in the task instance dimension | % | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Instancegpumemutil | GPU memory utilization in the task instance dimension | % | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Instancegpuutil | GPU utilization in the task instance dimension | % | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Instancehttpqps | HTTP requests per second in the task instance dimension | Count/s | InstanceId<br>Source<br>SubUin | [ 10s, max ]<br>[ 60s, max ]<br>[ 300s, max ]<br>[ 3600s, max ]<br>[ 86400s, max ] |

| | | | | |
|---|---|---|---|---|
| Instancehttpqpslimit | HTTP requests per second limited in the task instance dimension | Count/s | InstanceId<br>Source<br>SubUin | [ 10s, max ]<br>[ 60s, max ]<br>[ 300s, max ]<br>[ 3600s, max ]<br>[ 86400s, max ] |
| Instancememutil | Memory utilization in the task instance dimension | % | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Instancememvalue | Memory usage in the task instance dimension | MBytes | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Instancenetworkibytes | Inbound network traffic in the task instance dimension | MBytes | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Instanceready | Number of running instances in the task dimension | Count | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, last ]<br>[ 60s, last ]<br>[ 300s, last ]<br>[ 3600s, last ]<br>[ 86400s, last ] |

| | | | | |
|---|---|---|---|---|
| InstanceTiemsCurrentRequests | Number of concurrent requests in the task instance dimension | Count | InstanceId<br>Source<br>SubUin | [ 10s, max ]<br>[ 60s, max ]<br>[ 300s, max ]<br>[ 3600s, max ]<br>[ 86400s, max ] |
| Instancetotal | Number of instances in the task dimension | Count | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, last ]<br>[ 60s, last ]<br>[ 300s, last ]<br>[ 3600s, last ]<br>[ 86400s, last ] |
| Memutil | Task-dimension memory utilization | % | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Memvalue | Task-dimension memory usage | MBytes | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Networkreceivebytes | Inbound network traffic in the task dimension | MBytes | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| ServiceCfsClientDataReadBandwidth | Task-dimension TuroCFS single-node server read bandwidth | KBytes/s | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| ServiceCfsClientDataWriteBandwidth | Task-dimension TuroCFS single-node server write bandwidth | KBytes/s | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| ServiceCfsDataReadIoBytes | Task-dimension CFS server read bandwidth | KBytes/s | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| ServiceCfsDataReadIoLatency | Task-dimension CFS read latency | ms | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| ServiceCfsDataWriteIoBytes | Task-dimension CFS server write bandwidth | KBytes/s | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| ServiceCfsDataWriteIoLatency | Task-dimension CFS write | ms | AppId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg |

| | | | | |
|---|---|---|---|---|
| | latency | | TaskId | ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| ServiceCfsStrageUsageGb | Task-dimension CFS storage data capacity | GBytes | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| ServiceDiskIoUtil | Task-dimension disk I/O utilization | % | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| ServiceDiskIoWait | Task-dimension disk I/O wait time | % | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| ServiceDiskReadByte | Task-dimension disk read bandwidth | MBytes/s | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| ServiceDiskReadIops | Task-dimension disk read IOPS | Count | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, |

| | | | | |
|---|---|---|---|---|
| | | | | avg ] |
| ServiceDiskUsageRadio | Task-dimension system disk partition utilization | % | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| ServiceDiskWriteByte | Task-dimension disk write bandwidth | MBytes/s | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| ServiceDiskWriteIops | Task-dimension disk write IOPS | Count | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Servicehttpqps | HTTP requests per second in the task dimension for the service | Count/s | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, max ]<br>[ 60s, max ]<br>[ 300s, max ]<br>[ 3600s, max ]<br>[ 86400s, max ] |
| Servicehttpqpslimit | Rate-limited HTTP requests per second in the task dimension for the service | Count/s | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, max ]<br>[ 60s, max ]<br>[ 300s, max ]<br>[ 3600s, max ]<br>[ 86400s, |

| | | | | |
|---|---|---|---|---|
| | | | | max ] |
| ServiceTiemsCurrentRequests | Number of concurrent requests in the task dimension | Count | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, max ]<br>[ 60s, max ]<br>[ 300s, max ]<br>[ 3600s, max ]<br>[ 86400s, max ] |
| ServiceGpuMemValue | Task-dimension GPU memory usage | MBytes | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| ServiceEmsTokenThroughput | Number of tokens processed per minute in the task dimension | Count | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| ServiceEmsTokenThroughputInput | Number of input tokens processed per minute in the task dimension | Count | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| ServiceEmsTokenThroughputOutput | Number of output tokens processed per minute in the task dimension | Count | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| ServiceEmsFirstTokenLatency | Task-dimension first token latency | s | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| ServiceEmsNonFirstTokenLatency | Task-dimension non-first token latency | s | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| ServiceEmsProcessingRequestCount | Number of requests in processing in the task dimension | Count | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, sum ]<br>[ 60s, sum ]<br>[ 300s, sum ]<br>[ 3600s, sum ]<br>[ 86400s, sum ] |
| ServiceEmsQueuingRequestCount | Number of queued requests in the task dimension | Count | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, sum ]<br>[ 60s, sum ]<br>[ 300s, sum ]<br>[ 3600s, sum ]<br>[ 86400s, sum ] |
| ServiceEmsTotalProcessedTokens | Total number of tokens processed in the task dimension | Count | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, sum ]<br>[ 60s, sum ]<br>[ 300s, sum ]<br>[ 3600s, sum ] |

| | | | | |
|---|---|---|---|---|
| | | | | [ 86400s, sum ] |
| ServiceEmsTotalProcessedTokensInput | Total number of output tokens in the task dimension | Count | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, sum ]<br>[ 60s, sum ]<br>[ 300s, sum ]<br>[ 3600s, sum ]<br>[ 86400s, sum ] |
| ServiceEmsTotalProcessedTokensOutput | Total number of output tokens in the task dimension | Count | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, sum ]<br>[ 60s, sum ]<br>[ 300s, sum ]<br>[ 3600s, sum ]<br>[ 86400s, sum ] |
| ServiceEmsAverageLengthInput | Average input length in tokens in the task dimension | Count | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| ServiceEmsAverageLengthOutput | Average output length in tokens in the task dimension | Count | AppId<br>Source<br>SubUin<br>TaskId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuMemValue | GPU memory usage in the task instance dimension | MBytes | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ] |

| | | | | |
|---|---|---|---|---|
| | | | | [ 86400s, avg ] |
| EmsTokenThroughput | Number of tokens processed per minute in the task instance dimension | Count | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| EmsTokenThroughputInput | Total number of input tokens processed per minute in the task instance dimension | Count | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| EmsTokenThroughputOutput | Total number of output tokens processed per minute in the task instance dimension | Count | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| EmsFirstTokenLatency | First token latency in the task instance dimension | s | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| EmsNonFirstTokenLatency | Non-first token latency in the task instance dimension | s | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| EmsProcessingRequestCount | Number of requests in processing in the task instance dimension | Count | InstanceId<br>Source<br>SubUin | [ 10s, sum ]<br>[ 60s, sum ]<br>[ 300s, sum ]<br>[ 3600s, sum ]<br>[ 86400s, sum ] |
| EmsQueuingRequestCount | Number of queued requests in the task instance dimension | Count | InstanceId<br>Source<br>SubUin | [ 10s, sum ]<br>[ 60s, sum ]<br>[ 300s, sum ]<br>[ 3600s, sum ]<br>[ 86400s, sum ] |
| EmsTotalProcessedTokens | Total number of tokens processed in the task instance dimension | Count | InstanceId<br>Source<br>SubUin | [ 10s, sum ]<br>[ 60s, sum ]<br>[ 300s, sum ]<br>[ 3600s, sum ]<br>[ 86400s, sum ] |
| EmsTotalProcessedTokensInput | Total number of input tokens processed in the task instance dimension | Count | InstanceId<br>Source<br>SubUin | [ 10s, sum ]<br>[ 60s, sum ]<br>[ 300s, sum ]<br>[ 3600s, sum ]<br>[ 86400s, sum ] |
| EmsTotalProcessedTokensOutput | Total number of output tokens | Count | InstanceId<br>Source<br>SubUin | [ 10s, sum ]<br>[ 60s, sum ] |

| | | | | |
|---|---|---|---|---|
| | processed in the task instance dimension | | | [ 300s, sum ]<br>[ 3600s, sum ]<br>[ 86400s, sum ] |
| EmsAverageLengthInput | Average input length (tokens) in the task instance dimension | Count | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| EmsAverageLengthOutput | Average output length (tokens) in the task instance dimension | Count | InstanceId<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Fp16EngineActivity | FP16 active time ratio in the service instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| Fp32EngineActivity | FP32 active time ratio in the service instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| Fp64EngineActivity | FP64 active time ratio in the service instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| NvlinkBandwidth | NVLink transmission rate in the service instance-GPU card dimension | Bytes/s | AppId<br>InstanceGpuNum<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| PcieBandwidth | PCIe bus transmission rate in the service instance-GPU card dimension | Bytes/s | AppId<br>InstanceGpuNum<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| SmActivity | SM active time ratio in the service instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| TensorActivity | Tensor active time ratio in the service instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DcgmFiDevFbUsed | GPU memory usage in the | MBytes | AppId<br>InstanceGpuNum<br>Source | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg |

| | | | | |
|---|---|---|---|---|
| | service instance-GPU card dimension | | SubUin | ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DcgmFiDevGpuUtil | GPU utilization in the service instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| DcgmFiDevMemCopyUtil | GPU memory utilization in the service instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuDecUtil | GPU decoder utilization in the service instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuEncUtil | GPU encoder utilization in the service instance-GPU card dimension | % | AppId<br>InstanceGpuNum<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| GpuMemoryClock | GPU memory frequency in the service instance-GPU card dimension | S | AppId<br>InstanceGpuNum<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuNvlinkRxMb | NVLink received data volume in the service instance-GPU card dimension | Mbps | AppId<br>InstanceGpuNum<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuNvlinkTxMb | NVLink transmitted data volume in the service instance-GPU card dimension | Mbps | AppId<br>InstanceGpuNum<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuPcieRxMb | PCIe received data volume in the service instance-GPU card dimension | Mbps | AppId<br>InstanceGpuNum<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuPcieTxMb | PCIe transmitted data volume in the service instance-GPU card dimension | Mbps | AppId<br>InstanceGpuNum<br>Source<br>SubUin | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GpuSmClock | SM clock frequency in the service | S | AppId<br>InstanceGpuNum<br>Source | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg |

| | | | | |
|---|---|---|---|---|
| | instance-GPU card dimension | | SubUin | ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

# TI-ONE Resource Management

| Metric English Name | Metric Description | Unit | Dimension | Statistical Granularity |
|---|---|---|---|---|
| GroupCpuUsage | CPU utilization of the resource group | % | ResourceGroupId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GroupGpuUtil | GPU utilization of the resource group | % | ResourceGroupId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GroupLanInTraffic | Inbound private network bandwidth of the resource group | Mbps | ResourceGroupId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GroupLanOutTraffic | Outbound private network bandwidth of the resource group | Mbps | ResourceGroupId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GroupMemUsage | Memory utilization of the resource group | % | ResourceGroupId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GroupWanInTraffic | Inbound public network bandwidth of the resource group | Mbps | ResourceGroupId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| GroupWanOutTraffic | Outbound public network bandwidth of the resource group | Mbps | ResourceGroupId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| InsCpuUsage | CPU utilization of the resource (node) | % | ResourceId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| InsGpuUtil | GPU utilization of the resource (node) | % | ResourceId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| InsMemUsage | Memory utilization of the resource (node) | % | ResourceId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| InsLanInTraffic | Inbound private network bandwidth of the resource (node) | Mbps | ResourceId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| InsLanOutTraffic | Outbound private network bandwidth of the resource (node) | Mbps | ResourceId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| InsWanInTraffic | Inbound public network bandwidth of the resource (node) | Mbps | ResourceId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| InsWanOutTraffic | Outbound public network bandwidth of the resource (node) | Mbps | ResourceId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GroupWanOutratio | Public network bandwidth utilization of the resource group | % | ResourceGroupId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GroupFp16EngineActivity | FP16 active time ratio of the resource group | % | ResourceGroupId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GroupFp32EngineActivity | FP32 active time ratio of the resource group | % | ResourceGroupId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GroupFp64EngineActivity | FP64 active time ratio of the resource group | % | ResourceGroupId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| GroupGpuDecUtil | GPU decoder utilization of the resource group | % | ResourceGroupId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GroupGpuEccPersistent | Persistent ECC errors of the resource group | Count | ResourceGroupId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GroupGpuEccVolatile | Volatile ECC errors of the resource group | Count | ResourceGroupId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GroupGpuEncUtil | GPU encoder utilization of the resource group | % | ResourceGroupId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GroupGpuPowerUsage | GPU power consumption of the resource group | W | ResourceGroupId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GroupGpuRetiredPages | Disabled memory pages of the resource group | Count | ResourceGroupId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| GroupGpuTemperature | GPU temperature of the resource group | °C | ResourceGroupId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| GroupGpuXidErrors | GPU XID count of the resource group | Count | ResourceGroupId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| InsWanOutratio | Public network bandwidth utilization of the resource (node) | % | ResourceId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| InsGpuDecUtil | GPU decoder utilization of the resource (node) | % | ResourceId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| InsGpuEccPersistent | Persistent ECC errors of the resource (node) | Count | ResourceId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| InsGpuEccVolatile | Volatile ECC errors of the resource (node) | Count | ResourceId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| InsGpuEncUtil | GPU encoder utilization of the resource (node) | % | ResourceId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| InsGpuPowerUsage | GPU power consumption of the resource (node) | W | ResourceId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| InsGpuRetiredPages | Disabled memory pages of the resource (node) | Count | ResourceId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| InsGpuTemperature | GPU temperature of the resource (node) | °C | ResourceId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| InsGpuXidErrors | GPU XID count of the resource (node) | Count | ResourceId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| InsFp16EngineActivity | FP16 active time ratio of the resource (node) | % | ResourceId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

| | | | | |
|---|---|---|---|---|
| InsFp32EngineActivity | FP32 active time ratio of the resource (node) | % | ResourceId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |
| InsFp64EngineActivity | FP64 active time ratio of the resource (node) | % | ResourceId<br>AppId | [ 10s, avg ]<br>[ 60s, avg ]<br>[ 300s, avg ]<br>[ 3600s, avg ]<br>[ 86400s, avg ] |

# Querying Monitoring Metrics: Guide for Calling the GetMonitorData API

## API Call Guide

1. API description

| API Name | Version | API Feature |
| --- | --- | --- |
| GetMonitorData | 2018-07-24 | Pulls the monitoring data of cloud product metrics. |

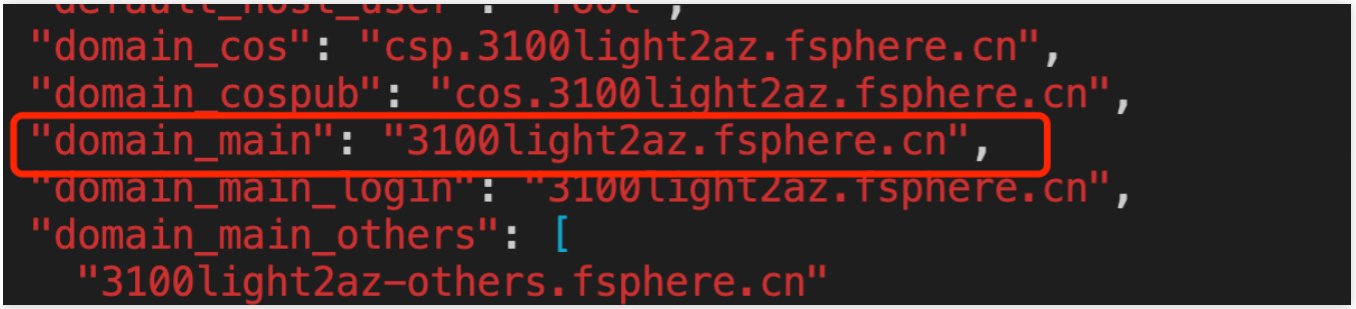
2. Input parameters

| Parameter Name | Required | Type | Description |
| --- | --- | --- | --- |
| Namespace | Yes | String | Namespace, such as qce/cvm.<br>Example value: qce/cvm. |
| MetricName | Yes | String | Metric name, such as cpu_usage. Only single-metric pulling is supported.<br>Example value: cpu_usage |
| Dimensions | Yes | Array | Dimensions of instance objects, in the form of a collection of key-value pairs. This field varies with the instance types. For example, this field is [{"unInstanceId": "ins-miltleuc"}] for a CVM instance. Each request supports batch retrieval of monitoring data for up to 10 instances.<br>Example value: [{"unInstanceId": "ins-miltleuc"}, {"unInstanceId": "ins-4xny0cbt"}] |
| Period | Yes | Integer | Monitoring statistical period, such as 60, in seconds. The statistical period supported by each metric may not be the same. For the statistical periods supported by each cloud product, see the appendix of this document. The maximum number of data points of a single request is 1,440.<br>Example value: 300. |
| StartTime | Yes | Time | Start time, for example, 2018-09-22T19:51:23+08:00.<br>Example value: 2019-03-24T10:51:23+08:00. |
| EndTime | Yes | Time | End time, such as 2018-09-22T20:51:23+08:00. The current time is used by default. EndTime cannot be earlier than StartTime.<br>Example value: 2019-03-24T20:51:23+08:00. |

3. Request domain name

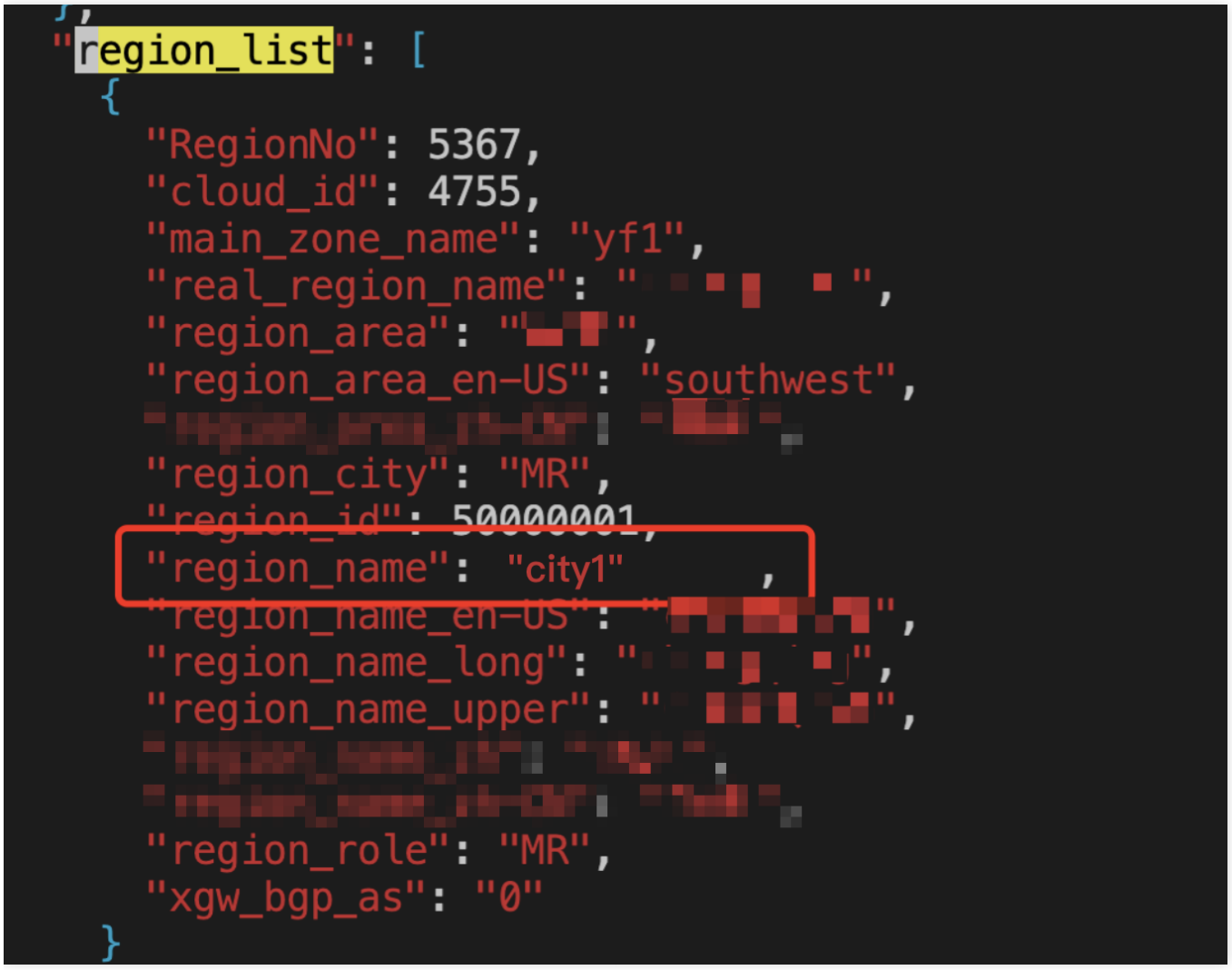
API request domain name: monitor.api3.${domain_main}, such as monitor.api3.3100light2az.fsphere.cn.

> Note:
>
> ${domain_main} can be obtained by logging into any Pod and searching for the domain_main keyword in /tce/conf/cm/local.json.




4. Request region

Since monitoring data is reported and stored by region, it is necessary to explicitly specify the region where the instance to be pulled is located when an API is called. The method to obtain region information is as follows: Log in to any Pod, open /tce/conf/cm/local.json, search for the region_list keyword, and the region_name indicates available regions.




5. Calling frequency limit

Default API request frequency limit: 20 times/second.

6. Output parameters

| Parameter Name | Type | Description |
| --- | --- | --- |
| MetricName | String | Metric name.<br>Example value: cpu_usage. |
| Period | Integer | Monitoring statistical period.<br>Example value: 60. |
| DataPoints[].Dimensions | Json | Instance object dimension group.<br>Example value: {"unInstanceId": "ins-miltleuc"}. |
| DataPoints[].Points | Array | Data point array. |
| StartTime | Time | Start time.<br>Example value: 2019-03-24T10:50:00+08:00. |

| Parameter Name | Type | Description |
| --- | --- | --- |
| EndTime | Time | End time.<br>Example value: 2019-03-24T20:50:00+08:00. |
| RequestId | String | Unique request ID. It is generated by the server and is returned for each request. (No RequestId is returned if the request is not received by the server due to certain reasons.) RequestId is required for locating an issue. |

# API Call Example

Scenario: Retrieve CPU utilization monitoring data for a CVM instance over a specified time range, with a 5-minute statistical period.

1. Input example

```
POST HTTP/1.1
Host: monitor.api3.3100light2az.fsphere.cn
Content-Type: application/json
Action: GetMonitorData
{
    "Namespace": "qce/cvm",
    "MetricName": "CPUUsage",
    "Period": 300,
    "Dimensions": [
        {
            "unInstanceId": "ins-miltleuc"
        }
    ],
    "StartTime": "2024-03-24T10:51:23+08:00",
    "EndTime": "2024-03-24T20:51:23+08:00"
}
```

2. Output example

```
{
    "Response": {
        "StartTime": "2019-03-24 10:50:00",
        "EndTime": "2019-03-24 20:50:00",
        "Period": 300,
        "MetricName": "cpu_usage",
        "DataPoints": [
            {
                "Dimensions": {
                        "unInstanceId": "ins-miltleuc"},
```

```
            "Points": [
                2.566,
                2.283,
                6.316,
                2.816,
                2.7,
                2.35
            ]
        }
    ],
    "RequestId": "d96ec542-6547-4af2-91ac-fee85c1b8b85"
  }
}
```


# SDK Code Call

Pull CPU utilization monitoring data with a 5-minute statistical period for a CVM instance over a period.

1. Golang example

```
package main

import (
    "fmt"

    "github.com/tcecloud/tcecloud-sdk-go/tcecloud/common"
    "github.com/tcecloud/tcecloud-sdk-go/tcecloud/common/errors"
    "github.com/tcecloud/tcecloud-sdk-go/tcecloud/common/profile"
    monitor "github.com/tcecloud/tcecloud-sdk-go/tcecloud/monitor/v20170312"
)

func main() {
    secretId := "AKIDxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx"        // Calling credentials. Refer to the appendix.
    secretKey := "eyzi69sGK3SWFeiblb8NKwD9l2yZplNf"          // Calling credentials. Refer to the appendix.

    credential := common.NewCredential(secretId, secretKey)
    cpf := profile.NewClientProfile()
    cpf.HttpProfile.Scheme = "HTTP"
    cpf.HttpProfile.ReqTimeout = 10
    cpf.HttpProfile.Endpoint = "monitor.api3.3100light2az.fsphere.cn"      // Calling domain name. Refer to 3. R
equest Domain Name in this chapter.
    client, _ := monitor.NewClient(credential, "city1", cpf)               // Region where the monitoring data of clou
d product instances is located. Refer to 4. Request Region in this chapter.

    request := monitor.NewGetMonitorDataRequest()
```

```
    request.Namespace = common.StringPtr("qce/cvm")
    request.MetricName = common.StringPtr("cpu_usage")
    request.Period = common.Uint64Ptr(60)
    request.StartTime = common.StringPtr("2024-10-17T15:10:42+08:00")
    request.EndTime = common.StringPtr("2024-10-17T16:10:42+08:00")

    // Build the Dimensions field.
    dimensions := []string{
        "{\"unInstanceId\": \"ins-miltleuc\"}",
    }
    request.Dimensions = common.StringPtrs(dimensions)
    response, err := client.GetMonitorData(request)

    if _, ok := err.(*errors.CloudSDKError); ok {
        fmt.Printf("An API error has returned: %s", err)
        return
    }
    // Non-SDK exception, direct failure. Other processing can be added to the actual code.
    if err != nil {
        panic(err)
    }
    // Print the returned json string
    fmt.Printf("%s", response.ToJsonString())
}
```

2. Python example

```
# -*- coding: utf8 -*-
#
import json
from tcecloud.common import credential
from tcecloud.common.exception.cloud_sdk_exception import CloudSDKException
from tcecloud.monitor.v20170312 import monitor_client, models
from tcecloud.common.profile.client_profile import ClientProfile
from tcecloud.common.profile.http_profile import HttpProfile
import ssl

ssl._create_default_https_context = ssl._create_unverified_context

try:
    secret_id = "AKIDxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx"          # Calling credentials. Refer to the appendix.
    secret_key = "1qVuzkBH3WgWhX3F7LJyJJyIJ0NP77rM"             # Calling credentials. Refer to the appendi
x.
    cred = credential.Credential(secret_id, secret_key)

    httpProfile = HttpProfile()
    httpProfile.endpoint = "monitor.api3.3100light2az.fsphere.cn"          # Calling domain name. Refer to 3. Re
quest Domain Name in this chapter.
```

```
    clientProfile = ClientProfile()
    clientProfile.httpProfile = httpProfile
    client = monitor_client.MonitorClient(cred, "city1", clientProfile)     # Region where the monitoring data of cloud product instances is located. Refer to 4. Request Region in this chapter.

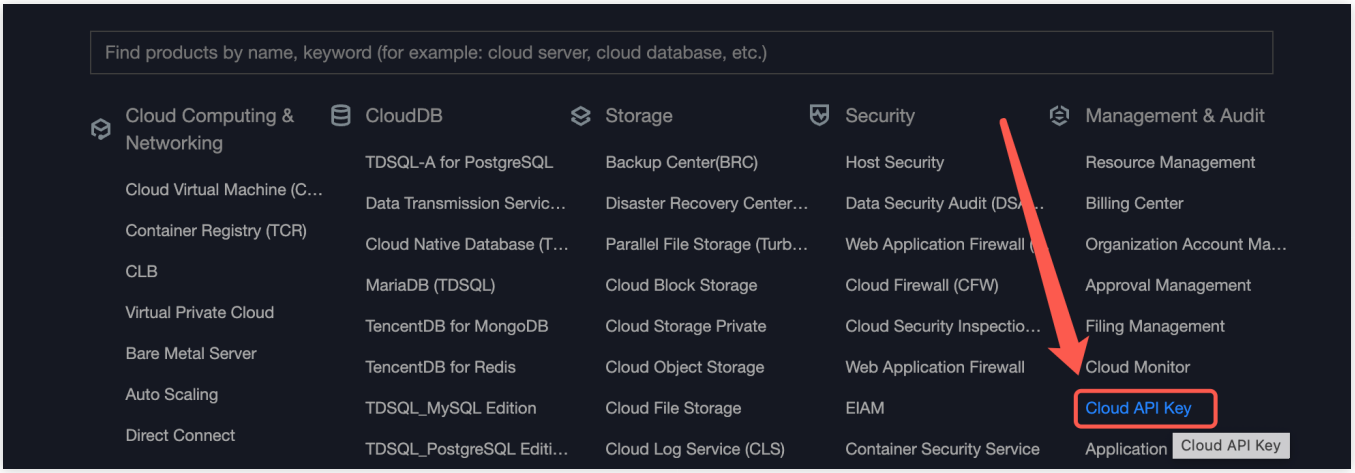
    req = models.GetMonitorDataRequest()
    params = {
        "Namespace": "qce/cvm",
        "MetricName": "cpu_usage",
        "Period": 60,
        "StartTime": "2024-10-17T15:10:42+08:00",
        "EndTime": "2024-10-17T16:10:42+08:00",
        "Dimensions": ['''{"unInstanceId": "ins-miltleuc"}''']
    }
    req.from_json_string(json.dumps(params))
    resp = client.GetMonitorData(req)

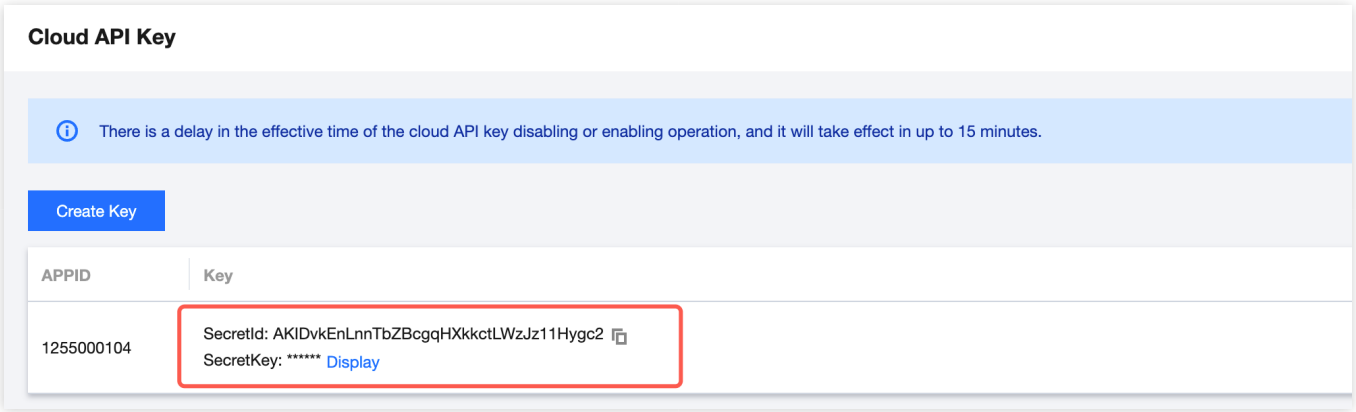
    # Output json format string back package
    print(resp.to_json_string(indent=2))

except CloudSDKException as err:
    print(err)
```

# Appendix: Calling Credentials Creation Process

1. Calling credentials: used for calling authentication, which means that the called AppID and UIN can be identified based on secretId and secretKey.
2. Creation process:




Cloud API Key
There is a delay in the effective time of the cloud API key disabling or enabling operation, and it will take effect in up to 15 minutes.
Create Key
APPID
Key
1255000104
SecretId: AKIDvkEnLnnTbZBcgqHXkkctLWzJz11Hygc2
SecretKey: ****** Display

# Shipping Monitoring Metrics to Kafka

## I. Feature Description

Cloud Monitor data can be forwarded to a specified Kafka, supporting:

- Full data forwarding, such as forwarding all metric data of CVMs.
- Partial forwarding. For example, metric data of a CVM instance or metric data of all CVM instances under a specific AppID is forwarded, or specified metric data of specified instances is forwarded.
  The above features are configured through forwarding rules.

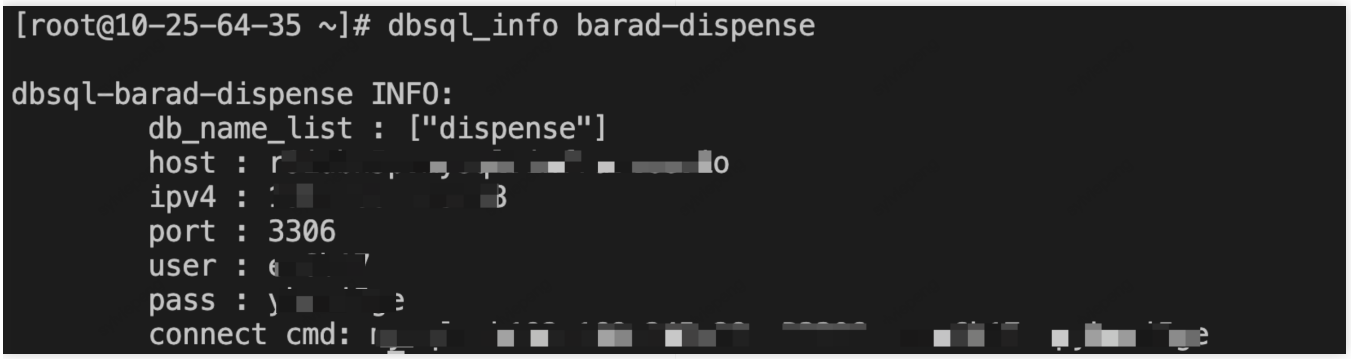
> Note:
>
> The tcloud-barad-dispense application must be deployed in the environment.

## II. Forwarding Configuration

Currently, forwarding rules need to be configured via CLI-based operations, that is, by inserting forwarding rules into the database. The steps are as follows:

1. Connect to the forwarding database: Set the database name to dispense and the corresponding service_id to dbsql-barad-dispense , and run the following command on the master node:

```
dbsql_info barad-dispense        // This is barad-dispense, not dbsql-barad-dispense.
```




> Note:
>
> The connect cmd is the command to connect to the dispense database.

2. Configure forwarding rules.

```
-- Create a rule.
insert into t_rule (RuleName, Status) values ('forwarding rule name', 1);
// Note:
// RuleName: rule name.
// Status: whether the rule is enabled. 1: enabled, 0: disabled.

-- Configure the metrics that need to be forwarded.
insert into t_metric (RuleId, Metric, Namespace, Measurement, Periods) values (10, 'cpu_usage', 'qce/cvm', 'cv
m_device', '60');
// Note:
// RuleId: ID of the new rule specified by t_rule.
// Metric: metric name.
// Namespace: Namespace to which the metric belongs.
// Measurement: view of Cloud Monitor. To obtain the view, refer to Appendix 1.
// Periods: time granularity of the metric. Multiple granularities can be specified, such as (60, 300), or the aste
risk (*) can be used to indicate all supported granularities.
// If you need to know the namespace to which the metrics belong, the measurement, as well as the support
ed time granularities, refer to the appendix.

-- Configure filter conditions (Optional. Configure this if you only want to forward data of a specific instance.
There are 10 tables in total. You can obtain the information about which table ${n} resides on by performing
a ruleId % 10 calculation.)
insert into t_condition_${n} (RuleId, Filters) values (10, '[{"Key":"vm_uuid","Values":["c7fc0b4c-0059-4318-bb0
c-dc4a31a071da"],"Expression":"="},{"Key":"__appid__","Values":["251005942", "1"],"Expression":"in"}]');
// Note:
// RuleId: ID of the new rule specified by t_rule.
// Filters: forwarding rule filter, which is of a JSON array with an and relationship between rules.

-- Configure the forwarding target Kafka.
insert into t_target (RuleId, Target, ProtoType) values (10, '{"brokers":"11.150.215.105:9092","topic":"Nowrit
e","user":"","password":""}', 1);
// Note:
// RuleId: ID of the new rule specified by t_rule.
// Target: target Kafka address.
// ProtoType: message types forwarded to Kafka. 1: JSON, 2: Protobuf.
```

# III. Message Protocol

The message protocol forwarded to Kafka is as follows:

```
// json
{
    "Namespace": "qce/cvm",
    "MetricName": "cpu_usage",
    "Statistic": "MAX",
```

```
    "Period": 60,
    "Dimensions": [{
        "Name": "appid",
        "Value": "1"
    }, {
        "Name": "projectid",
        "Value": "0"
    }, {
        "Name": "vm_uuid",
        "Value": "4f7639f1-f156-41b4-9072-f7e27af97d3b"
    }],
    "Timestamp": 1737107700000,
    "Value": 22.916666666666668
}
```

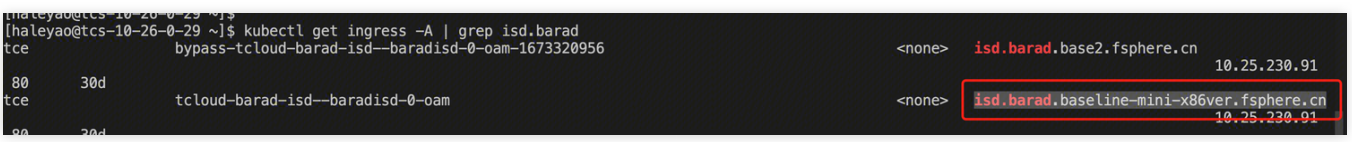

# IV. Forwarding Table Structure Description

| Table Name | Meaning | Description |
| --- | --- | --- |
| t_rule | Forwarding a rule table | Includes rule metadata such as rule ID, rule name, and configuration user.<br>`Id: 38230`<br>`RuleName: 1118-Phase 2-cvm-cpu // Forwarding rule name.`<br>`AppId: 1300260700 // User AppID for configuring rules, can be empty.`<br>`SubUin: 0 // User UIN for configuring rules, can be empty.`<br>`Status: 1 // Rule enabling status, 1: enabled; 0: disabled.`<br>`CreateTime: 2024-11-20 11:14:23`<br>`UpdateTime: 2024-11-20 11:14:23`<br>`IsAllMetric: 0 // Can be empty.` |
| t_metric | Forwarding a metric table | Information about the metrics to be forwarded.<br>Note:<br>If forwarding is performed within a private network, `ExtNamespace` and `ExtMetric` are empty.<br><br>` Id: 3732457`<br>` RuleId: 38230`<br>` ExtMetric: CpuUtil // External metric name, can be empty.`<br>`ExtNamespace: QCE/TI_MODEL // External namespace, can be empty.`<br>` Measurement: model_svc // View name.`<br>` Periods: * // * indicates all granularities. If you only want to forward granularities 60 and 300, configure it as 60,300.`<br>` Metric: cpu_util // Internal metric name.`<br>` Namespace: qce/ti_model // Internal namespace.`<br>` ENMeaning: cpu_util // Metric English name, which can be empty.`<br>` CName: `<br>` Unit: % // Metric unit, which can be empty.` |

| | | |
|---|---|---|
| | | ` CreateTime: 2024-12-12 18:30:00`<br>` UpdateTime: 2024-12-12 18:30:00` |
| t_condition_${n} | Forwarding a condition table | • There are 10 tables (numbered 0 to 9). The specific table for forwarding conditions can be determined by ruleId % 10.<br>• When the current table is empty, it indicates that unconditional forwarding is configured in the private network.<br>• The Expression in Filters can only be set to `=` or `in`. When it is set to `=`, only a single value is contained in Values. When it is set to `in`, multiple values may be contained in Values.<br><br>`Id: 12`<br>`RuleId: 38230`<br>`Filters: [{"Key":"vm_uuid","Values":["c7fc0b4c-0059-4318-bb0c-dc4a31a071da"],"Expression":"="},{"Key":"__appid__","Values":["251005942", "1"],"Expression":"in"}]`<br>`CreateTime: 2024-11-19 17:49:19`<br>`UpdateTime: 2024-11-19 17:49:19` |
| t_target | Forwarding a target table | • TargetType must be set to `Kafka`.<br>• Valid values of ProtoType:<br>• 1: `JSON`<br>• 2: `Protobuf`<br><br>` Id: 34539`<br>` RuleId: 38230`<br>`TargetType: Kafka`<br>` Target: {"brokers":"11.150.215.105:9092","topic":"Nowrite","user":"","password":""}`<br>` ProtoType: 1`<br>`CreateTime: 2024-11-19 16:48:31`<br>`UpdateTime: 2024-11-19 16:48:31` |

# 5. Appendix: Viewing Metrics, Namespace, Measurement, and Period

1. Run kubectl get ingress -A | grep isd.barad on the master node.



2. Access the domain name through a browser.

# Best Practices
# Application of Instance Grouping in Alarms

This document introduces how to implement unified settings of alarm policies in batch for cross-instance and cross-region projects of the same product according to workload classification.

1. Overview
   As your business grows, the number of cloud resources increases, making efficient daily monitoring a challenge for Ops. With the instance grouping feature in Cloud Monitor, you can classify a large batch of cloud resources by workload. This allows you to apply unified alarm policies to all instances within a group, helping you stay informed about their usage and operational status, and significantly improving Ops efficiency.

- Use cases
  Suppose you have five CVMs, three of which (CVM-1, CVM-2, and CVM-3) are used for Game A, while the remaining two are assigned to other workloads.
  - CVM-1 is located in Region A and assigned to Project A.
  - CVM-2 is located in Region B and assigned to Project B.
  - CVM-3 is located in Region C and assigned to Project C.
    To promptly learn about the usage and operation status of the three CVMs for Game A, the following configurations are required.

  i. Create an instance group. Add CVM-1, CVM-2, and CVM-3 to the same instance group.
  ii. Create an alarm policy, which includes metrics such as CPU utilization, memory utilization, disk utilization, and outbound bandwidth of the public network (for monitoring traffic usage). Alarm notifications are sent automatically when metrics of any instance reach a certain threshold.

2. Grouping instances
   i. Log in to the Cloud Platform console and choose Cloud Monitor > Instance Group.
   ii. Click Create and configure the instance group in the pop-up window as follows.
   - Group name: The name is "Game A".
   - Group type: Choose Cloud Virtual Machine > Basic Monitoring.
   - Add to the group: Select regions City A, City B, and City C respectively, and add CVM-1, CVM-2, and CVM-3 from each region to the instance group.

   iii. After the configuration is completed, click Save.
3. Creating alarms
   i. Log in to the Cloud Platform console and choose Cloud Monitor > Alarm Configuration >

Alarm Policy.

ii. Click Create, and perform configurations according to the prompts on the page.

- Policy name: Cutomized by you.
- Policy type: Choose Cloud Virtual Machine > Basic Monitoring.
- Alarm object: Choose Instance Group > Game Business A.
- Alarm trigger conditions.

  CPU utilization, statistical period 1 minute, >, 85%, lasts for 1 data point, and alarm once per day.

  Memory utilization, statistical period 1 minute, >, 85%, lasts for 1 data point, and alarm once per day.

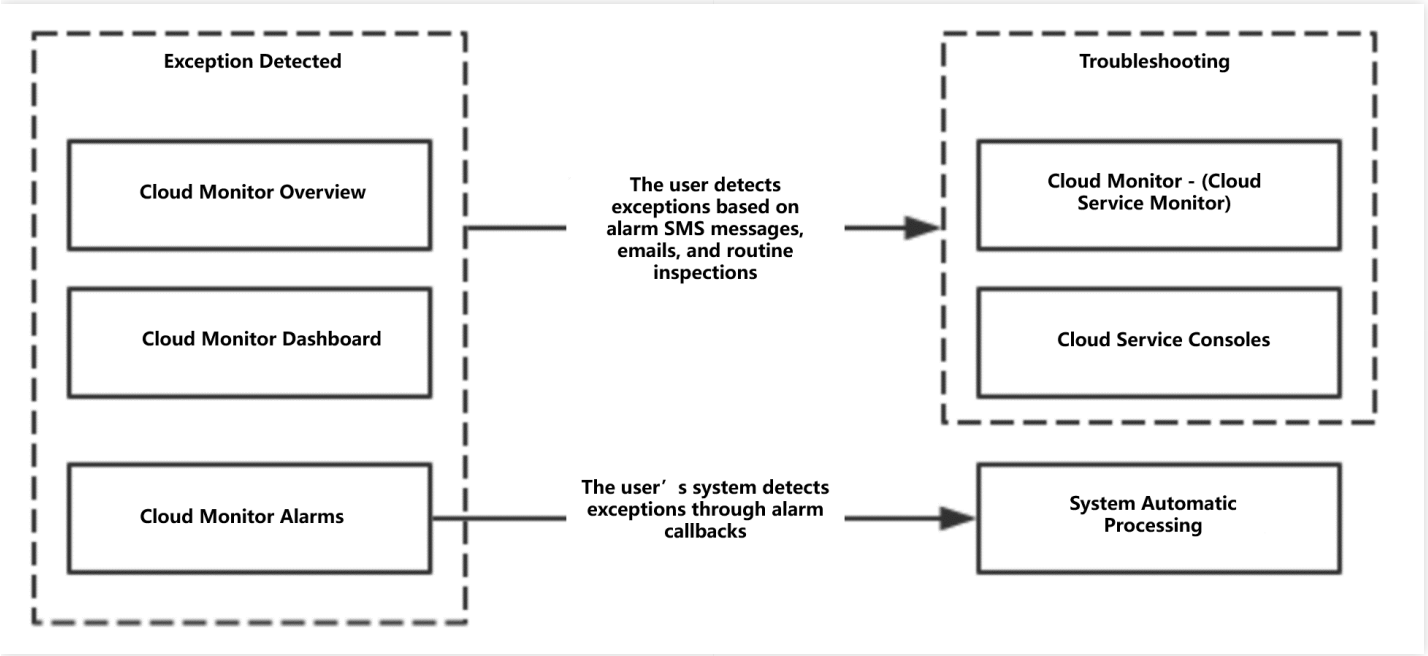
  Disk utilization, statistical period 1 minute, >, 85%, lasts for 1 data point, and alarm once per day.

  Outbound bandwidth, statistical period 1 minute, >, 100Mbps, lasts for 1 data point, and alarm once per day.
- Notification template: The system default is set to preset notification template. You can also click Select Template to choose an existing one you've created, or click Create New Template to create one based on your needs.

iii. After the configuration is completed, click Complete.

# Best Practices for Monitoring Scenarios

Cloud Monitor provides multiple methods to help users identify resource exceptions and send exception information to users through various channels as quickly as possible.




1. Locating exceptions
    i. Detecting exceptions via monitoring and alarms
    Monitoring and alarms enable the cloud platform to detect exceptions promptly and send proactive notifications, allowing users to identify exceptions passively. This ensures timely awareness of exceptions under all circumstances. Users can log in to the Cloud Platform console and go to the Cloud Monitor console to configure appropriate alarm policies for the resources they care about. For details, see Creating an Alarm Policy.
    Key performance metrics and events configured as alarm rules will trigger multiple notification methods to promptly reach users and their systems in the event of an exception.
    Alarm policies configured with designated recipients or recipient groups will notify users promptly via SMS, email, and other channels. Features such as repeated alarms and alarm convergence are supported to help ensure critical alarms are not missed while minimizing unnecessary disturbances.
    Users can also configure the callback API feature in alarm notifications to deliver exception alarm information to their systems for further aggregation and processing.
    ii. Detecting exceptions via the monitoring dashboard
    Locating exceptions via the monitoring dashboard is a user-driven approach that relies on analyzing average performance trends and historical data. It requires users to proactively identify exceptions. For exceptions that are not covered by alarm policies or are difficult to

detect through existing alarm policies, they can still be identified during daily inspections. Compared with alarms, the monitoring dashboard enables users to assess the overall impact scope of resource exceptions. By subscribing key resources to the dashboard and configuring charts appropriately, users can highlight exception information under various scenarios. For details, see Create New Dashboard.

For individual instances, users can subscribe to detailed instance views to easily compare performance trends on the dashboard panel.

For resource clusters, users can subscribe to aggregated data within the same cluster to conveniently view the overall monitoring dashboard on the dashboard panel and compare it with the trend of individual instances within the cluster. For details, see Monitoring Scenarios for Large Volumes.

Exceptions identified through the monitoring dashboard can be further analyzed using the dashboard's sorting and ranking features, enabling users to locate the affected resources and assess the scope of impact for in-depth troubleshooting.

2. Exception troubleshooting

   Locating exception objects via the monitoring overview page

   During daily inspections or upon receiving alarm notifications, users can log in to the Cloud Platform console and choose Cloud Monitor > Monitoring Overview.

   i. Check the Service Status in Last 24 Hours module to view exception statuses of resources across different regions.

      Use the Exception Information Overview feature to get a preliminary view of recent exceptions.

   ii. Click the number of exception objects to be redirected to the specific exception product menu page under Cloud Product Monitoring.

      The specific product list page under Cloud Product Monitoring will automatically filter and display the exception resource objects for the user.

   iii. Click the ID of a specific object to be redirected to its monitoring details page, which provides historical status rollback to assist in detailed exception troubleshooting.

   - The exception timeline provides users with access to both current and historical information about the exception object. By reviewing historical alarms and status changes, users can be assisted in troubleshooting the current exception.
   - The monitoring data for resource performance provides users with the most comprehensive resource performance data. It allows for week-over-week and day-over-day comparisons of current and historical data for the same metric and facilitates troubleshooting by comparing data changes in different metrics over the same time period.

3. Locating exception objects via the dashboard

   Log in to the Cloud Platform console and choose Cloud Monitor > Dashboard > Dashboard List to go to the corresponding dashboard page.

   i. When an abnormal trend appears in the monitoring chart, click Expand the Instance List, and then a sorted list of corresponding instances will be displayed below the curved line graph.

Through the list, the specific objects that are causing the exceptions can be located.

ii. By clicking the object name in the sorted list, users can be redirected to the monitoring details page of the object, where historical status information is provided to assist users in locating exceptions.

- The exception timeline provides users with access to both current and historical information of the exception object. By reviewing past alarms and status changes, users can be assisted in troubleshooting the current exception.
- The monitoring data for resource performance provides users with the most comprehensive resource performance data. It allows for week-over-week and day-over-day comparisons of current and historical data for the same metric and facilitates troubleshooting by comparing data changes in different metrics over the same time period.

# Configuring CVM Metric Alarms

Scenarios

The following example illustrates how to configure an alarm: Suppose you want to receive an SMS alarm at the phone number 12345678888 when the CPU utilization of CVM instance ins-12345678 (located in Region A) exceeds 80% for two consecutive five-minute periods.

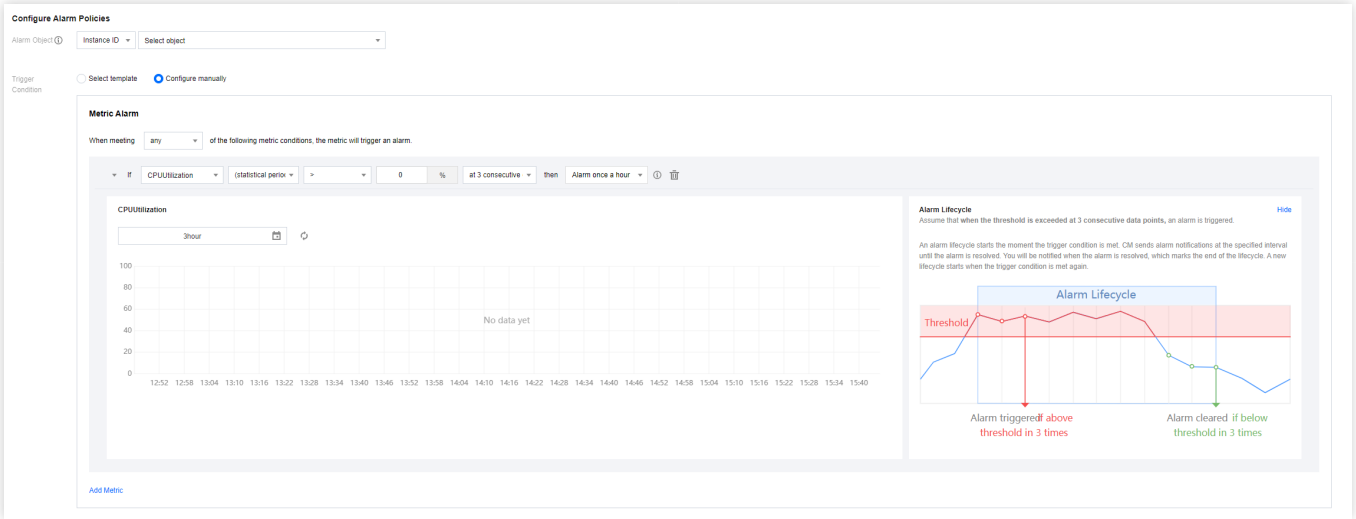
Operation Steps

1. Log in to the Cloud Platform console and choose Cloud Monitor > Alarm Configuration > Alarm Policy.
2. Click Create and configure the following options.
3. Configure the basic information.

- Policy name: CPU alarm.
- Policy type: Choose Cloud Virtual Machine > Basic Monitoring.

4. Configure the alarm object. Under the Alarm Object module, select instance ID, and then choose the CVM instance from the drop-down list on the right.
5. Configure trigger conditions. Under the Trigger Conditions module, configure the following conditions.

- Select Manual Configuration.
- In the Metric Alarm module, select CPU utilization, statistical granularity of 5 minutes, >, 80%, for 2 consecutive data points, and trigger an alarm every 15 minutes.




6. Configure the alarm notification template. Click Create New Template, enter a template name, select the recipient with the phone number 12345678888, and choose SMS as the notification channel. After completion, click Confirm.

7. Click Complete to complete the alarm configuration.

8. At this point, when the instance's CPU utilization exceeds 80% for two consecutive 5-minute periods, an SMS alarm will be sent by Cloud Monitor to the phone number 12345678888.

# Product White Paper
# Product Overview

Cloud Monitor is a platform that can perform real-time monitoring and alarming for cloud product resources. It provides users with the capability to uniformly monitor resources of cloud products, such as Cloud Virtual Machine (CVM) and cloud databases. Users can use Cloud Monitor to comprehensively understand the resource utilization, application performance, and running status of cloud products and perform monitoring, collecting, and analysis on various services and resources on the cloud platform in real time, ensuring the stability, availability, and security of the cloud environment.
Cloud Monitor leverages its powerful visualization and alarm capabilities to provide users with a one-stop monitoring solution based on the monitoring data of metrics, linkages, and events. It improves Ops and troubleshooting efficiency and guarantees the health and stability of workloads.

Introduction to Submodules of Cloud Monitor

| Submodule | Introduction | Common Use Case |
| --- | --- | --- |
| Dashboard | It provides intelligent dashboards with visualization and analysis capabilities for monitoring data of cloud product metrics. | Analyze the causes of metric exceptions through dashboards. |
| Alarm management | It provides the alarm feature for monitoring metrics related to cloud product monitoring, cloud automated testing, application performance monitoring, and frontend performance monitoring. | Users can set alarm rules for monitoring metrics that affect system performance to enable the alarm system to automatically check monitoring data and send alarm notifications when an exception occurs. |
| Cloud product monitoring | It supports viewing and pulling the monitoring data of cloud product metrics. | - |

# Product Strengths

- Instance-Level Cloud Product Monitoring
  Cloud product resource monitoring is implemented at the instance level to accurately display the resource running status.
- Prompt and Personalized Alarms
  - Resource alarm policies for different scenarios can be set in different aspects, such as metric and event.
  - Alarm levels can be used to distinguish resource exceptions of different severity levels.
  - Alarm notifications can be sent through different channels, such as email, SMS message, and WeCom account.
- Multi-Dimensional Monitoring Chart Display
  Rich chart styles and metrics can be configured for data display, allowing users to view resource status from dimensions, such as cloud product and cloud resource instance.

# Use Cases

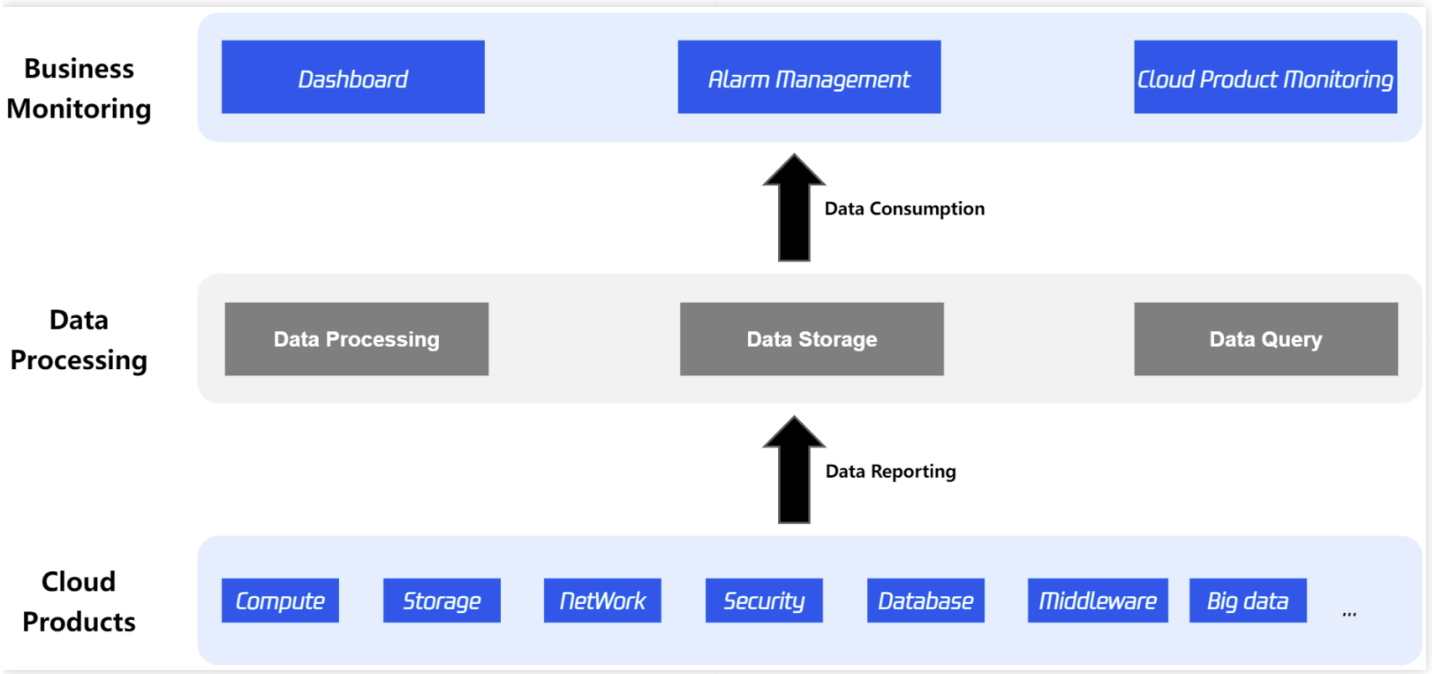
## Troubleshooting and Locating Exceptions

- Detect exceptions through monitoring and alarming.
  Monitoring and alarming is a method that supports the cloud platform in timely detecting and proactively notifying exceptions, as well as passively detecting exceptions by users. It ensures users can promptly detect exceptions under any circumstances. By configuring alarm rules for critical performance metrics and events, users or user-side systems can receive alarm notifications through multiple channels when exceptions related to these metrics and events occur.
- Detect exceptions through monitoring views.
  Locating exceptions through monitoring views is a method for users to proactively locate exceptions based on the average trends of performance metrics and historical data. For exceptions with no corresponding alarm configuration or that are hard to be detected by alarm rules, users can detect them through monitoring views during daily inspections. Compared with monitoring and alarming, this method helps users determine the scope of impact on resources from a global perspective.

## Monitoring Large-Scale Resources

- Performance view for large-scale resource monitoring.
- Aggregated view for large-scale resources monitoring.
- Detailed view for large-scale resources monitoring.
- Alarm policy configuration for large-scale resources.

# Product Architecture




Various cloud services uniformly report monitoring data to Cloud Monitor. After relevant data processing operations, Cloud Monitor can implement product capabilities, including dashboard, alarm management, and cloud product monitoring.

# Features

- Monitoring Cloud Product Resources
  Cloud Monitor provides users with cloud resource monitoring services.
  - Cloud Monitor provides the features of collecting basic monitoring metrics automatically, accurately, and almost instantaneously, as well as storing monitoring metric data.
  - The cloud product monitoring and dashboard modules support creating diverse charts to display the metric data of cloud products and configuring alarms. In addition, the metric data can be pulled via APIs for further application and analysis.
  - The monitoring metrics of most cloud services can be collected automatically for monitoring data display. Users do not need to perform any monitoring-related configuration.
  - For certain cloud services, such as CVM and Cloud Bare Metal (CBM), users need to install and run the corresponding monitoring agent to collect monitoring data. Users can choose to automatically install the agent for these cloud Services when purchasing resources.
- Providing Alarm Services
  Cloud Monitor provides users with prompt and personalized alarm services.
  - Based on existing monitoring metric data, users can apply default alarm settings or customize the settings for cloud resources and custom metrics.
  - Users can configure different alarm policies, such as event alarms, failure alarms, and threshold alarms, for resources in different Ops scenarios.
  - Users can configure alarm frequency and alarm logics to distinguish alarms at different levels.
  - Users can receive alarm notifications promptly through different channels by configuring mobile numbers, email addresses, WeCom accounts, and callback URLs.
- Displaying Data with Multi-Dimensional Charts
  Cloud Monitor allows users to view existing monitoring data in diversified custom charts.
  - Users can conveniently view the monitoring data of various cloud resources in the monitoring pop-up window or on the monitoring details page in the console of each cloud product. By default, the latest data is displayed in a curved line graph.
  - Users can view the overall monitoring status of cloud services on the Monitoring Overview page and comprehensively monitor the cloud service status from three aspects: health status, timeline, and exception.
  - In Dashboard, users can subscribe to key metrics, customize the display forms and calculation rules of different metrics, and sort and compare metric data in lists and charts to assist in troubleshooting and analyzing issues.
  - Users can view the monitoring data and corresponding alarm information on all cloud services under their account in Cloud Product Monitoring.
- Grouping Instances
  Users can group instances of the same cloud product across regions for unified management. When

they create a large number of instances, they can create instance groups as needed. By managing instance groups, they can manage instances more conveniently and set alarm policies to improve Ops efficiency.

# API Documentation
# Cloud Monitor(monitor)
# version(2018-07-24)
# API overview

## API version

V3

## Alarm APIs

| Interface name | Interface function |
| --- | --- |
| CreateAlarmPolicy | CreateAlarmPolicy |
| DescribeAlarmEvents | DescribeAlarmEvents |
| DescribeAlarmMetrics | DescribeAlarmMetrics |
| DescribeAlarmPolicy | DescribeAlarmPolicy |
| DescribeMiniDashboardAlarmInfo | DescribeMiniDashboardAlarmInfo |
| ModifyAlarmPolicyInfo | ModifyAlarmPolicyInfo |
| ModifyAlarmPolicyStatus | ModifyAlarmPolicyStatus |

## Cloud Monitor API

| Interface name | Interface function |
| --- | --- |
| BindingPolicyObject | BindingPolicyObject |
| CreateAlarmNotice | CreateAlarmNotice |
| CreateNoticeContentTmpl | CreateNoticeContentTmpl |
| CreatePolicyGroup | CreatePolicyGroup |

| Interface name | Interface function |
| --- | --- |
| DeleteAlarmNotices | DeleteAlarmNotices |
| DeleteNoticeContentTmpls | DeleteNoticeContentTmpls |
| DeletePolicyGroup | DeletePolicyGroup |
| DescribeAbnormalObjects | DescribeAbnormalObjects |
| DescribeAccidentEventList | DescribeAccidentEventList |
| DescribeAgentStatusHistory | DescribeAgentStatusHistory |
| DescribeAlarmHistories | DescribeAlarmHistories |
| DescribeAlarmNotice | DescribeAlarmNotice |
| DescribeAlarmNoticeCallbacks | DescribeAlarmNoticeCallbacks |
| DescribeAlarmNotices | DescribeAlarmNotices |
| DescribeAlarmPolicies | DescribeAlarmPolicies |
| DescribeAllNamespaces | DescribeAllNamespaces |
| DescribeBaseMetrics | DescribeBaseMetrics |
| DescribeBasicAlarmList | DescribeBasicAlarmList |
| DescribeBindingPolicyObjectList | DescribeBindingPolicyObjectList |
| DescribeMonitorProducts | DescribeMonitorProducts |
| DescribeNoticeContentTmpl | DescribeNoticeContentTmpl |
| DescribePolicyConditionList | DescribePolicyConditionList |
| DescribePolicyGroupInfo | DescribePolicyGroupInfo |
| DescribePolicyGroupList | DescribePolicyGroupList |
| DescribeProductEventList | DescribeProductEventList |
| DescribeProductHealthStatusList | DescribeProductHealthStatusList |
| DescribeProductList | DescribeProductList |
| DescribeStatisticData | DescribeStatisticData |
| DescribeUnifyDashboards | DescribeUnifyDashboards |

| Interface name | Interface function |
| --- | --- |
| GetMonitorData | GetMonitorData |
| ModifyAlarmNotice | ModifyAlarmNotice |
| ModifyAlarmPolicyCondition | ModifyAlarmPolicyCondition |
| ModifyAlarmPolicyNotice | ModifyAlarmPolicyNotice |
| ModifyAlarmPolicyTasks | ModifyAlarmPolicyTasks |
| ModifyAlarmReceivers | ModifyAlarmReceivers |
| ModifyNoticeContentTmpl | ModifyNoticeContentTmpl |
| PutMonitorData | PutMonitorData |
| SetDefaultAlarmPolicy | SetDefaultAlarmPolicy |
| UnBindingAllPolicyObject | UnBindingAllPolicyObject |
| UnBindingPolicyObject | UnBindingPolicyObject |

# Other APIs

| Interface name | Interface function |
| --- | --- |
| DescribeLogTopics | DescribeLogTopics |

# CallingMethod signaturev1

Converge Cloud authenticates each access request, i.e. each request needs to include authentication information (Signature) in the common parameters to verify the identity of the requester.
The Signature is generated by the security credentials which include SecretId and SecretKey. If you don't have the security credentials yet, go to the API Key page to apply for them; otherwise, you cannot invoke the API.

## 1. Applying for Security Credentials

Before using the API for the first time, go to the API Key page to apply for security credentials.
Security credentials consist of SecretId and SecretKey:

- SecretId is used to identify the API requester.
- SecretKey is used to encrypt the signature string and verify it on the server.
- You must keep your security credentials private and avoid disclosure.

You can apply for the security credentials through the following steps:

1. Log in to the Converge Cloud Console.
2. Go to the Cloud API Key page.
3. On the API Key Management page, click Create Key to create a SecretId/SecretKey pair.

Note: Each account can have up to two pairs of SecretId/SecretKey.

## 2. Generating a Signature

With the SecretId and SecretKey, a signature can be generated. The following describes how to generate a signature:

Assume that the SecretId and SecretKey are:

- SecretId: AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPLE
- SecretKey: Gu5t9xGARNpq86cd98joQYCN3EXAMPLE

Note: This is just an example. For actual operations, please use your own SecretId and SecretKey.

Take the Cloud Virtual Machine's request to view the instance list (DescribeInstances) as an example. When

you invoke this API, the request parameters may be as follows:

| Parameter name | Description | Parameter value |
| --- | --- | --- |
| Action | Method name | DescribeInstances |
| SecretId | Key ID | AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPLE |
| Timestamp | Current timestamp | 1465185768 |
| Nonce | Random positive integer | 11886 |
| Region | Region where the instance is located | luzon |
| InstanceIds.0 | ID of the instance to query | ins-09dx96dg |
| Offset | Offset | 0 |
| Limit | Allowed maximum output | 20 |
| Version | API version number | 2017-03-12 |

## 2.1. Sorting Parameters

First, sort all the request parameters in an ascending lexicographical order (ASCII code) by their names. Notes: (1) Parameters are sorted by their names instead of their values; (2) The parameters are sorted based on ASCII code, not in an alphabetical order or by values. For example, InstanceIds.2 should be arranged after InstanceIds.12. You can complete the sorting process using a sorting function in a programming language, such as the ksort function in PHP. The parameters in the example are sorted as follows:

```
{
    'Action' : 'DescribeInstances',
    'InstanceIds.0' : 'ins-09dx96dg',
    'Limit' : 20,
    'Nonce' : 11886,
    'Offset' : 0,
    'Region' : 'luzon',
    'SecretId' : 'AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPLE',
    'Timestamp' : 1465185768,
    'Version': '2017-03-12',
}
```

When developing in another programming language, you can sort these sample parameters and it will work as long as you obtain the same results.

## 2.2. Concatenating a Request String

This step generates a request string.
Format the request parameters sorted in the previous step into the form of "parameter name"="parameter value". For example, for the Action parameter, its parameter name is "Action" and its parameter value is "DescribeInstances", so it will become Action=DescribeInstances after formatted.
Note: The "parameter value" is the original value but not the value after URL encoding.

Then, concatenate the formatted parameters with "&". The resulting request string is as follows:

```
Action=DescribeInstances&InstanceIds.0=ins-09dx96dg&Limit=20&Nonce=11886&Offset=0&Region=luzon&SecretId=AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPLE&Timestamp=1465185768&Version=2017-03-12
```

## 2.3. Concatenating the Signature Original String

This step generates a signature original string.
The signature original string consists of the following parameters:

1. HTTP method: POST and GET modes are supported, and GET is used here for the request. Please note that the method name should be in all capital letters.
2. Request host: the domain name of the request to view the list of instances (DescribeInstances) is cvm.convergecloud.com. The actual request domain name varies by the module to which the API belongs. For more information, see the instructions of the specific API.
3. Request path: The request path in the current version of Cloud API is fixed to / .
4. Request string: the request string generated in the previous step.

The concatenation rule of the signature original string is: Request method + request host + request path + ? + request string

The concatenation result of the example is:

```
GETcvm.convergecloud.com/?Action=DescribeInstances&InstanceIds.0=ins-09dx96dg&Limit=20&Nonce=11886&Offset=0&Region=luzon&SecretId=AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPLE&Timestamp=1465185768&Version=2017-03-12
```

## 2.4. Generating a Signature String

This step generates a signature string.
First, use the HMAC-SHA1 algorithm to sign thesignature original stringobtained in the previous step, and then encode the generated signature using Base64 to obtain the final signature.

The specific code is as follows with the PHP language being used as an example:

```
$secretKey = 'Gu5t9xGARNpq86cd98joQYCN3EXAMPLE';
$srcStr = 'GETcvm.convergecloud.com/?Action=DescribeInstances&InstanceIds.0=ins-09dx96dg&Limit
=20&Nonce=11886&Offset=0&Region=luzon&SecretId=AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPL
E&Timestamp=1465185768&Version=2017-03-12';
$signStr = base64_encode(hash_hmac('sha1', $srcStr, $secretKey, true));
echo $signStr;
```

The final signature is:

```
EliP9YW3pW28FpsEdkXt/+WcGeI=
```

When developing in another programming language, you can sign and verify the original in the example above and it works as long as you get the same results.

# 3. Encoding a Signature String

The generated signature string cannot be directly used as a request parameter and must be URL encoded.

For example, if the signature string generated in the previous step is EliP9YW3pW28FpsEdkXt/+WcGeI= , the final signature string request parameter (Signature) is EliP9YW3pW28FpsEdkXt%2f%2bWcGeI%3d, which will be used to generate the final request URL.

**Note: If your request method is GET, or the request method is POST and the Content-Type is application/x-www-form-urlencoded, then all the request parameter values need to be URL encoded (except the parameter key and the symbol of =) when sending the request. Non-ASCII characters need to be encoded with UTF-8 before URL encoding. **

Note: The network libraries of some programming languages automatically URL encode all parameters, in which case there is no need to URL encode the signature string; otherwise, two rounds of URL encoding will cause the signature to fail.

Note: Other parameter values also need to be encoded using RFC 3986. Use %XY in percent-encoding for special characters such as Chinese characters, where "X" and "Y" are hexadecimal characters (0-9 and uppercase A-F), and using lowercase will cause an error.

# 4. Signature Failure

The following situational error codes for signature failure may occur. Please resolve the errors accordingly.

| Error code | Error description |
| --- | --- |
| AuthFailure.SignatureExpire | The signature is expired |

| Error code | Error description |
| --- | --- |
| AuthFailure.SecretIdNotFound | The key does not exist |
| AuthFailure.SignatureFailure | Signature error |
| AuthFailure.TokenFailure | Token error |
| AuthFailure.InvalidSecretId | Invalid key (not a Cloud API key type) |

# 5. Signature Demo

When calling API 3.0, you are recommended to use the corresponding Converge Cloud SDK 3.0, Swhich encapsulates the signature process, enabling you to focus on only the specific APIs provided by the product when developing. See SDK Center for more information. Currently, the following programming languages are supported:

- Python
- Java
- PHP
- Go
- Node

To further explain the signing process, we will use a programming language to implement the process described above. The request domain name, API and parameter values in the sample are used here. This goal of this example is only to provide additional clarification for the signature process, please see the SDK for actual usage.

The final output URL might be:  https://cvm.convergecloud.com/?Action=DescribeInstances&InstanceIds.0=ins-09dx96dg&Limit=20&Nonce=11886&Offset=0&Region=luzon&SecretId=AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPLE&Signature=EliP9YW3pW28FpsEdkXt%2F%2BWcGeI%3D&Timestamp=1465185768&Version=2017-03-12

Note: The key in the example is fictitious, and the timestamp is not the current time of the system, so if this URL is opened in the browser or called using commands such as curl, an authentication error will be returned: Signature expired. In order to get a URL that can work properly, you need to replace the SecretId and SecretKey in the example with your real credentials and use the current time of the system as the Timestamp.

Note: In the example below, even if you use the same programming language, the order of the parameters in the URL may be different for each execution. However, the order does not matter, as long as all the parameters are included in the URL and the signature is calculated correctly.

Note: The following code is only applicable to API 3.0. It cannot be directly used in other signature processes. Even with an older API, signature calculation errors may occur due to the differences in details. Please refer to the corresponding documentation.

## Java

```
import java.io.UnsupportedEncodingException;
import java.net.URLEncoder;
import java.util.Random;
import java.util.TreeMap;
import javax.crypto.Mac;
import javax.crypto.spec.SecretKeySpec;
import javax.xml.bind.DatatypeConverter;

public class CloudAPIDemo {
    private final static String CHARSET = "UTF-8";

    public static String sign(String s, String key, String method) throws Exception {
        Mac mac = Mac.getInstance(method);
        SecretKeySpec secretKeySpec = new SecretKeySpec(key.getBytes(CHARSET), mac.getAlgorithm());
        mac.init(secretKeySpec);
        byte[] hash = mac.doFinal(s.getBytes(CHARSET));
        return DatatypeConverter.printBase64Binary(hash);
    }

    public static String getStringToSign(TreeMap<String, Object> params) {
        StringBuilder s2s = new StringBuilder("GETcvm.convergecloud.com/?");
        // When signing, the parameters need to be sorted in lexicographical order. TreeMap is used here
to guarantee the correct order.
        for (String k : params.keySet()) {
            s2s.append(k).append("=").append(params.get(k).toString()).append("&");
        }
        return s2s.toString().substring(0, s2s.length() - 1);
    }

    public static String getUrl(TreeMap<String, Object> params) throws UnsupportedEncodingException
{
        StringBuilder url = new StringBuilder("https://cvm.convergecloud.com/?");
        // There is no requirement for the order of the parameters in the actual request URL.
        for (String k : params.keySet()) {
            // The request string needs to be URL encoded. As the Key is all in English letters, only the valu
e is URL encoded here.
            url.append(k).append("=").append(URLEncoder.encode(params.get(k).toString(), CHARSET)).app
end("&");
        }
        return url.toString().substring(0, url.length() - 1);
```

```
    }

    public static void main(String[] args) throws Exception {
        TreeMap<String, Object> params = new TreeMap<String, Object>(); // TreeMap enables automati
c sorting
        // A random number should be used when actually calling, for example: params.put("Nonce", new
Random().nextInt(java.lang.Integer.MAX_VALUE));
        params.put("Nonce", 11886); // Common parameter
        // The current time of the system should be used when actually calling, for example:   params.put
("Timestamp", System.currentTimeMillis() / 1000);
        params.put("Timestamp", 1465185768); // Common parameter
        params.put("SecretId", "AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPLE"); // Common parameter
        params.put("Action", "DescribeInstances"); // Common parameter
        params.put("Version", "2017-03-12"); // Common parameter
        params.put("Region", "luzon"); // Common parameter
        params.put("Limit", 20); // Business parameter
        params.put("Offset", 0); // Business parameter
        params.put("InstanceIds.0", "ins-09dx96dg"); // Business parameter
        params.put("Signature", sign(getStringToSign(params), "Gu5t9xGARNpq86cd98joQYCN3EXAMPLE
", "HmacSHA1")); // Common parameter
        System.out.println(getUrl(params));
    }
}
```

## Python

Note: If running in a Python 2 environment, the following requests dependency package must be installed first:  pip install requests .

```
# -*- coding: utf8 -*-
import base64
import hashlib
import hmac
import time

import requests

secret_id = "AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPLE"
secret_key = "Gu5t9xGARNpq86cd98joQYCN3EXAMPLE"

def get_string_to_sign(method, endpoint, params):
    s = method + endpoint + "/?"
    query_str = "&".join("%s=%s" % (k, params[k]) for k in sorted(params))
    return s + query_str

def sign_str(key, s, method):
```

```
    hmac_str = hmac.new(key.encode("utf8"), s.encode("utf8"), method).digest()
    return base64.b64encode(hmac_str)

if __name__ == '__main__':
    endpoint = "cvm.convergecloud.com"
    data = {
        'Action' : 'DescribeInstances',
        'InstanceIds.0' : 'ins-09dx96dg',
        'Limit' : 20,
        'Nonce' : 11886,
        'Offset' : 0,
        'Region' : 'luzon',
        'SecretId' : secret_id,
        'Timestamp' : 1465185768, # int(time.time())
        'Version': '2017-03-12'
    }
    s = get_string_to_sign("GET", endpoint, data)
    data["Signature"] = sign_str(secret_key, s, hashlib.sha1)
    print(data["Signature"])
    # n actual invocation would occur here, which may incur fees after success
    # resp = requests.get("https://" + endpoint, params=data)
    # print(resp.url)
```

# signaturev3

Converge Cloud The API will authenticate each access request, meaning that each request must include signature information (Signature) in the public request parameters to verify the requester's identity.
The signature information is generated by the security credentials, which include SecretId and SecretKey . If the user does not have security credentials, please apply for them on the Cloud API Key page, otherwise they cannot call the Cloud API interfaces.

## 1. Apply for security credentials.

Before using the Cloud API for the first time, please go to the Cloud API Key page to apply for a security credential.
Security credentials include SecretId and SecretKey:

- SecretId The security credentials are used to identify the identity of the API caller.
- SecretKey The secret key is used to encrypt and sign the string for server-side verification of the signature.
- Users must strictly protect their security credentials and avoid disclosure them.

The specific steps to apply for security credentials are as follows:

1. Log in to Converge Cloud Administration Center Console
2. Go to the console page for cloud API keys
3. On the Cloud API Key page, click New to create a pair of SecretId/SecretKey

Note: Developer accounts can have up to two pairs SecretId / SecretKey.

## 2. TC3-HMAC-SHA256 signature method

Note: For GET method, only Content-Type: application/x-www-form-urlencoded protocol format is supported. For POST method, Content-Type: application/json and Content-Type: multipart/form-data are currently supported. By default, all business interfaces are supported in json format. Only specific business interfaces are supported in multipart format. In this case, the interface cannot be called in json format. Refer to the specific business interface document description.

The following takes Cloud Virtual Machine querying Region-1 instance list as an example to introduce the signature calculation process step by step. We only use two parameters for the query instance list: Limit and Offset, called using the GET method.

Assume that the SecretId and SecretKey of the user are AKIDz 8krbsJ 5yKBZQpn 74WFkmLPx 3EXAMPLE and Gu 5t9xGARNpq 86cd 98joQYCN3EXAMPLE, respectively.

## 2.1. splice specification request string

Concatenate CanonicalRequest strings in the following format:

```
CanonicalRequest =
  HTTPRequestMethod + '\n' +
  CanonicalURI + '\n' +
  CanonicalQueryString + '\n' +
  CanonicalHeaders + '\n' +
  SignedHeaders + '\n' +
  HashedRequestPayload
```

- HTTPRequestMethod: HTTP request methods (GET, POST), GET in this example;
- CanonicalURI: URI parameter, API 3.0 fixed to forward slash (/);
- CanonicalQueryString: Query string in the URL that initiates the HTTP request. For POST requests, it is fixed as an empty string. For GET requests, it is the question mark (?) in the URL. The following string content, this example value is: Limit=10&Offset=0. Note: CanonicalQueryString needs to be URL encoded.
- CanonicalHeaders: The header information of participating signature, at least including host and content-type, can also be added with customized header to improve the uniqueness and security of its own request. Splicing rules: 1) head key and value are uniformly converted to lowercase, and the first and last spaces are removed, and spliced according to the format of key:value\n;2) multiple heads are spliced according to the dictionary sorting of head key (lowercase). In this example: content-type: application/x-www-form-urlencoded\nhost: cvm.convergecloud.com\n
- SignedHeaders: The header information involved in the signature indicates which headers participate in the signature of this request, and the header content contained in CanonicalHeaders is one-to-one corresponding. content-type and host are required headers. constructed as: 1) The head key is uniformly converted to lowercase; 2) Multiple header keys (lowercase) are concatenated according to dictionary sorting and separated by semicolons (;). In this case: content-type;host
- HashedRequestPayload: The hash value of the request body, calculated as Lowercase(HexEncode(Hash.SHA256(RequestPayload))), SHA256 hash the entire payload of the HTTP request body, then hexadecimal encoding, and finally converting the encoding string to lowercase letters. Note: For GET requests, RequestPayload is fixed as an empty string. For POST requests, RequestPayload is the payload of the HTTP request body.

According to the above rules, the resulting canonical request string in the example is as follows (for clarity, \nnewlines are replaced by a new line printed separately):

```
GET
```

```
/
Limit=10&Offset=0
content-type:application/x-www-form-urlencoded
host:cvm.convergecloud.com

content-type;host
e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855
```

## 2.2. Concatenate the string to be signed

Concatenate the string to be signed in the following format:

```
StringToSign =
  Algorithm + \n +
  RequestTimestamp + \n +
  CredentialScope + \n +
  HashedCanonicalRequest
```

- Algorithm: Signature algorithm, currently fixed as TC3-HMAC-SHA256;
- RequestTimestamp: Request timestamp, i.e. the X-TC-Timestamp value of the request header. The request in the example above is 1539084154;
- CredentialScope: Credential range in the format Date/service/tc3_request, containing date, requested service and termination string (tc3_request). **Date is the date of UTC standard time, the value needs to be consistent with UTC standard time converted by common parameter X-TC-Timestamp **;service is the product name, which must be consistent with the product domain name called, such as cvm. The above example request, the value is 2018-10-09/cvm/tc3_request;
- HashedCanonicalRequest: The hash value of the canonical request string obtained by concatenating the above steps is calculated by Lowercase(HexEncode(Hash.SHA256(CanonicalRequest))).

note:

> 1. Date must be calculated from timestamp X-TC-Timestamp and time zone UTC+0. Adding local time zone information to the system, such as East Zone 8, will cause calls to succeed during the day and night, but calls to fail in the early hours of the morning. Assuming the timestamp is 1551113065, the time in the East Eight District is 2019-02-26 00:44:25, but the calculated Date with UTC+0 should be 2019-02-25 instead of 2019-02-26.
> 2. Timestamp must be the current system time and ensure that the system time and standard time are synchronized. If the difference exceeds five minutes, it will fail. If it is not synchronized with the standard time for a long time, it may cause the request to fail after running for a period of time (return signature expiration error).

According to the above rules, the string to be signed in the example is as follows (for clarity,\nnewlines are

replaced by new lines printed):

```
TC3-HMAC-SHA256
1539084154
2018-10-09/cvm/tc3_request
91c9c192c14460df6c1ffc69e34e6c5e90708de2a6d282cccf957dbf1aa7f3a7
```

## 2.3. calculate the signature

1. Compute the derived signature key, pseudocode as follows

```
SecretKey = "Gu5t9xGARNpq86cd98joQYCN3EXAMPLE"
SecretDate = HMAC_SHA256("TC3" + SecretKey, Date)
SecretService = HMAC_SHA256(SecretDate, Service)
SecretSigning = HMAC_SHA256(SecretService, "tc3_request")
```

- SecretKey: Original SecretKey;
- Date: That is, the Date field information in Credential, as shown in the example above, is 2018-10-09.;
- Service: That is, the Service field information in Credential, as shown in the example above, is cvm;

2. Calculate the signature, pseudocode as follows

```
Signature = HexEncode(HMAC_SHA256(SecretSigning, StringToSign))
```

- SecretSigning: That is, the derivative signature key calculated above;
- StringToSign: That is, the character string to be signed calculated in step 2;

## 2.4. Splicing Authorization

Splice as follows Authorization:

```
Authorization =
    Algorithm + ' ' +
    'Credential=' + SecretId + '/' + CredentialScope + ', ' +
    'SignedHeaders=' + SignedHeaders + ', '
    'Signature=' + Signature
```

- Algorithm: Signature method, fixed as TC3-HMAC-SHA256;
- SecretId: SecretId in key pair;
- CredentialScope: See above, Scope of Voucher;
- SignedHeaders: See above, header information for participating signatures;
- Signature: signature value

According to the above rules, the values obtained in the example are:

```
TC3-HMAC-SHA256 Credential=AKIDEXAMPLE/Date/service/tc3_request, SignedHeaders=content-type;host, Signature=5da7a33f6993f0614b047e5df4582db9e9bf4672ba50567dba16c6ccf174c474
```

The final complete invocation information is as follows:

```
https://cvm.convergecloud.com/?Limit=10&Offset=0

Authorization: TC3-HMAC-SHA256 Credential=AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPLE/2018-10-09/cvm/tc3_request, SignedHeaders=content-type;host, Signature=5da7a33f6993f0614b047e5df4582db9e9bf4672ba50567dba16c6ccf174c474
Content-Type: application/x-www-form-urlencoded
Host: cvm.convergecloud.com
X-TC-Action: DescribeInstances
X-TC-Version: 2017-03-12
X-TC-Timestamp: 1539084154
X-TC-Region: luzon
```

# 3. signing failed

According to the actual situation, there are the following signature failure error codes, please handle according to the actual situation

| error code | error description |
| --- | --- |
| AuthFailure.SignatureExpire | signature expiration |
| AuthFailure.SecretIdNotFound | key is not present |
| AuthFailure.SignatureFailure | signature error |
| AuthFailure.TokenFailure | token error |
| AuthFailure.InvalidSecretId | Illegal key (not cloud API key type) |

# 4. Signature Demo

Java

```
import java.io.BufferedReader;
import java.io.InputStream;
```

```
import java.io.InputStreamReader;
import java.net.URL;
import java.text.SimpleDateFormat;
import java.util.Date;
import java.util.Map;
import java.util.TimeZone;
import java.util.TreeMap;
import javax.crypto.Mac;
import javax.crypto.spec.SecretKeySpec;
import javax.net.ssl.HttpsURLConnection;
import javax.xml.bind.DatatypeConverter;

import org.apache.commons.codec.digest.DigestUtils;

public class CloudAPITC3Demo {
    private final static String CHARSET = "UTF-8";
    private final static String ENDPOINT = "cvm.convergecloud.com";
    private final static String PATH = "/";
    private final static String SECRET_ID = "AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPLE";
    private final static String SECRET_KEY = "Gu5t9xGARNpq86cd98joQYCN3EXAMPLE";
    private final static String CT_X_WWW_FORM_URLENCODED = "application/x-www-form-urlencode
d";
    private final static String CT_JSON = "application/json";
    private final static String CT_FORM_DATA = "multipart/form-data";

    public static byte[] sign256(byte[] key, String msg) throws Exception {
        Mac mac = Mac.getInstance("HmacSHA256");
        SecretKeySpec secretKeySpec = new SecretKeySpec(key, mac.getAlgorithm());
        mac.init(secretKeySpec);
        return mac.doFinal(msg.getBytes(CHARSET));
    }

    public static void main(String[] args) throws Exception {
        String service = "cvm";
        String host = "cvm.convergecloud.com";
        String region = "luzon";
        String action = "DescribeInstances";
        String version = "2017-03-12";
        String algorithm = "TC3-HMAC-SHA256";
        String timestamp = "1539084154";
        //String timestamp = String.valueOf(System.currentTimeMillis() / 1000);
        SimpleDateFormat sdf = new SimpleDateFormat("yyyy-MM-dd");
        // Pay attention to the time zone, otherwise it is easy to make mistakes.
        sdf.setTimeZone(TimeZone.getTimeZone("UTC"));
        String date = sdf.format(new Date(Long.valueOf(timestamp + "000")));

        // ************* Step 1: Assemble the specification request string *************
```

```
        String httpRequestMethod = "GET";
        String canonicalUri = "/";
        String canonicalQueryString = "Limit=10&Offset=0";
        String canonicalHeaders = "content-type:application/x-www-form-urlencoded\n" + "host:" + host
+ "\n";
        String signedHeaders = "content-type;host";
        String hashedRequestPayload = DigestUtils.sha256Hex("");
        String canonicalRequest = httpRequestMethod + "\n" + canonicalUri + "\n" + canonicalQueryStri
ng + "\n"
                + canonicalHeaders + "\n" + signedHeaders + "\n" + hashedRequestPayload;
        System.out.println(canonicalRequest);

        // ************* Step 2: Stitching the string to be signed *************
        String credentialScope = date + "/" + service + "/" + "tc3_request";
        String hashedCanonicalRequest = DigestUtils.sha256Hex(canonicalRequest.getBytes(CHARSET));
        String stringToSign = algorithm + "\n" + timestamp + "\n" + credentialScope + "\n" + hashedCan
onicalRequest;
        System.out.println(stringToSign);

        // ************* Step 3: Calculate the signature *************
        byte[] secretDate = sign256(("TC3" + SECRET_KEY).getBytes(CHARSET), date);
        byte[] secretService = sign256(secretDate, service);
        byte[] secretSigning = sign256(secretService, "tc3_request");
        String signature = DatatypeConverter.printHexBinary(sign256(secretSigning, stringToSign)).toLow
erCase();
        System.out.println(signature);

        // ************* Step 4: Splicing Authorization *************
        String authorization = algorithm + " " + "Credential=" + SECRET_ID + "/" + credentialScope + ", "
                + "SignedHeaders=" + signedHeaders + ", " + "Signature=" + signature;
        System.out.println(authorization);

        TreeMap<String, String> headers = new TreeMap<String, String>();
        headers.put("Authorization", authorization);
        headers.put("Host", host);
        headers.put("Content-Type", CT_X_WWW_FORM_URLENCODED);
        headers.put("X-TC-Action", action);
        headers.put("X-TC-Timestamp", timestamp);
        headers.put("X-TC-Version", version);
        headers.put("X-TC-Region", region);
    }
}
```

## Python

```
# -*- coding: utf-8 -*-
```

```
import hashlib, hmac, json, os, sys, time
from datetime import datetime

# key parameters
secret_id = "AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPLE"
secret_key = "Gu5t9xGARNpq86cd98joQYCN3EXAMPLE"

service = "cvm"
host = "cvm.convergecloud.com"
endpoint = "https://" + host
region = "luzon"
action = "DescribeInstances"
version = "2017-03-12"
algorithm = "TC3-HMAC-SHA256"
timestamp = 1539084154
date = datetime.utcfromtimestamp(timestamp).strftime("%Y-%m-%d")
params = {"Limit": 10, "Offset": 0}

# ************* Step 1: Assemble the specification request string *************
http_request_method = "GET"
canonical_uri = "/"
canonical_querystring = "Limit=10&Offset=0"
ct = "x-www-form-urlencoded"
payload = ""
if http_request_method == "POST":
    canonical_querystring = ""
    ct = "json"
    payload = json.dumps(params)
canonical_headers = "content-type:application/%s\nhost:%s\n" % (ct, host)
signed_headers = "content-type;host"
hashed_request_payload = hashlib.sha256(payload.encode("utf-8")).hexdigest()
canonical_request = (http_request_method + "\n" +
                     canonical_uri + "\n" +
                     canonical_querystring + "\n" +
                     canonical_headers + "\n" +
                     signed_headers + "\n" +
                     hashed_request_payload)
print(canonical_request)

# ************* Step 2: Stitching the string to be signed *************
credential_scope = date + "/" + service + "/" + "tc3_request"
hashed_canonical_request = hashlib.sha256(canonical_request.encode("utf-8")).hexdigest()
string_to_sign = (algorithm + "\n" +
                  str(timestamp) + "\n" +
                  credential_scope + "\n" +
                  hashed_canonical_request)
print(string_to_sign)
```

```
# ************* Step 3: Calculate the signature *************
# Compute signature digest function
def sign(key, msg):
    return hmac.new(key, msg.encode("utf-8"), hashlib.sha256).digest()
secret_date = sign(("TC3" + secret_key).encode("utf-8"), date)
secret_service = sign(secret_date, service)
secret_signing = sign(secret_service, "tc3_request")
signature = hmac.new(secret_signing, string_to_sign.encode("utf-8"), hashlib.sha256).hexdigest()
print(signature)

# ************* Step 4: Splicing Authorization *************
authorization = (algorithm + " " +
                 "Credential=" + secret_id + "/" + credential_scope + ", " +
                 "SignedHeaders=" + signed_headers + ", " +
                 "Signature=" + signature)
print(authorization)

# Common parameters added to request header
headers = {
    "Authorization": authorization,
    "Host": host,
    "Content-Type": "application/%s" % ct,
    "X-TC-Action": action,
    "X-TC-Timestamp": str(timestamp),
    "X-TC-Version": version,
    "X-TC-Region": region,
}
```

# request

## 1. Service Address

Region refers to the geographical area of a physical data center. Converge Cloud ensures complete isolation between different regions, providing maximum stability and fault tolerance. To reduce access latency and improve download speed, it is recommended to choose the region closest to your customers.

You can view the complete region list by using the API interface "DescribeRegions".

## 2. Communications Protocol

All the Converge Cloud APIs communicate via HTTPS, providing highly secure communication tunnels.

## 3. Request Methods

Supported HTTP request methods:

- POST (recommended)
- GET

PThe Content-Type types supported by POST requests:

- application/json (recommended). The TC3-HMAC-SHA256 signature algorithm must be used.
- application/x-www-form-urlencoded. The HmacSHA1 or HmacSHA256 signature algorithm must be used.
- multipart/form-data (only supported by certain APIs). You must use TC3-HMAC-SHA256 to calculate the signature.

The size of a GET request packet is up to 32 KB. The size of a POST request is up to 1 MB when the HmacSHA1 or HmacSHA256 signature algorithm is used, and up to 10 MB when TC3-HMAC-SHA256 is used.

## 4. Character Encoding

Only UTF-8 encoding is used.

# result

## Response for Successful Requests

For example, when calling CAM API (version: 2017-03-12) to view the status of instances (DescribeInstancesStatus), if the request has succeeded, you may see the response as shown below:


```
{
    "Response": {
        "TotalCount": 0,
        "InstanceStatusSet": [],
        "RequestId": "b5b41468-520d-4192-b42f-595cc34b6c1c"
    }
}
```


- The API will return Response, which contains RequestId, as long as it processes the request. It does not matter if the request is successful or not.
- RequestId is the unique ID of an API request. Contact us with this ID when an exception occurs.
- Except for the fixed fields, all fields are action-specified. For the definitions of action-specified fields, see the corresponding API documentation. In this example, TotalCount and InstanceStatusSet are the fields specified by the API DescribeInstancesStatus. 0 TotalCount means that the requester owns 0 CVM instance so the InstanceStatusSet is empty.

## Response for Failed Requests

If the request has failed, you may see the response as shown below:


```
{
    "Response": {
        "Error": {
            "Code": "AuthFailure.SignatureFailure",
            "Message": "The provided credentials could not be validated. Please check your signature is cor
rect."
        },
        "RequestId": "ed93f3cb-f35e-473f-b9f3-0d451b8b79c6"
    }
}
```


- The presence of the Error field indicates that the request has failed. A response for a failed request will include Error, Code and Message fields.

- Code is the code of the error that helps you identify the cause and solution. There are two types of error codes so you may find the code in either common error codes or API-specified error codes.
- Message explains the cause of the error. Note that the returned messages are subject to service updates. The information the messages provide may not be up-to-date and should not be the only source of reference.
- RequestId is the unique ID of an API request. Contact us with this ID when an exception occurs.

# Common Error Codes

If there is an Error field in the response, it means that the API call failed. The Code field in Error indicates the error code. The following table lists the common error codes that all actions can return.

| Error Code | Description |
| --- | --- |
| AuthFailure.InvalidSecretId | Invalid key (not a API key type). |
| AuthFailure.MFAFailure | MFA failed. |
| AuthFailure.SecretIdNotFound | The key does not exist. |
| AuthFailure.SignatureExpire | Signature expired. |
| AuthFailure.SignatureFailure | Signature error. |
| AuthFailure.TokenFailure | Token error. |
| AuthFailure.UnauthorizedOperation | The request does not have CAM authorization. |
| DryRunOperation | DryRun Operation. It means that the request would have succeeded, but the DryRun parameter was used. |
| FailedOperation | Operation failed. |
| InternalError | Internal error. |
| InvalidAction | The API does not exist. |
| InvalidParameter | Incorrect parameter. |
| InvalidParameterValue | Invalid parameter value. |
| LimitExceeded | Quota limit exceeded. |
| MissingParameter | A parameter is missing. |
| NoSuchVersion | The API version does not exist. |

| Error Code | Description |
| --- | --- |
| RequestLimitExceeded | The number of requests exceeds the frequency limit. |
| ResourceInUse | Resource is in use. |
| ResourceInsufficient | Insufficient resource. |
| ResourceNotFound | The resource does not exist. |
| ResourceUnavailable | Resource is unavailable. |
| UnauthorizedOperation | Unauthorized operation. |
| UnknownParameter | Unknown parameter. |
| UnsupportedOperation | Unsupported operation. |
| UnsupportedProtocol | hHTTPS request method error. Only GET and POST requests are supported. |
| UnsupportedRegion | API does not support the requested region. |

# PublicParameter

Common parameters are parameters used to identify users and interface authentication purposes. If not necessary, these parameters are not described in the separate interface document of each interface, but these parameters need to be carried in each request before the request can be normally initiated.

## signature method v3

When using TC3-HMAC-SHA256 signature method, the common parameters need to be uniformly placed in the HTTP Header request header, as follows:

| parameter name | type | required | described |
|---|---|---|---|
| X-TC-Action | String | Yes | The name of the operation interface. Refer to the description of the public parameter Action in the input parameters of the interface documentation for the value. For example, the value for the query instance list interface of the cloud server is DescribeInstances. |
| X-TC-Region | String | Yes | region parameters, Data used to identify which region you wish to operate on. The region value accepted by the interface refers to the description of the input parameter common parameter Region in the interface document. Note: Some interfaces do not need to pass this parameter, which is specifically stated in the interface documentation. In this case, even passing this parameter will not take effect. |
| X-TC-Timestamp | Integer | Yes | The current UNIX timestamp, which records the time when the API request was initiated. for example 1529223702. Note: If the time difference from the server exceeds 5 minutes, it will cause a signature expiration error. |
| X-TC-Version | String | Yes | Version of API for operation. Refer to the description of the public parameter Version in the interface document for the value. For example, the version of the cloud server. 2017-03-12. |
| Authorization | String | Yes | HTTP The standard authentication header field is "Authorization". For example: TC3-HMAC-SHA256 Credential=AKIDEXAMPLE/Date/service/tc3_request, SignedHeaders=content-type;host, Signature=fe5f80f77d5fa3beca038a248ff027d0445342fe2855ddc963176630326f1024 Here,<br>- TC3-HMAC-SHA256: Signature method, currently fixed as this value,<br>- Credential: Credential: Signature credential; AKIDEXAMPLE is the SecretId; Date is a date in UTC time, and this value must match the value of X-TC-Timestamp (a common parameter) in UTC time format; service is the name of the product/service, and is generally a domain name prefix. For example,cvm;<br>- SignedHeaders: The headers that contains the authentication information; content-type and host are the required headers;<br>- Signature: Signature digest. |
| X-TC-Token | String | No | The token used for temporary certificates needs to be used in conjunction with a temporary key. Temporary keys and tokens need to be obtained by calling the access management service interface. Long-term keys do not require tokens. |

## signature method v1

When using HmacSHA1 and HmacSHA256 signature methods, common parameters need to be placed uniformly in the request string, as follows

| parameter name | type | required | described |
| --- | --- | --- | --- |
| Action | String | Yes | The name of the interface to be operated. The value refers to the explanation of the public parameter Action in the interface documentation. For example, for the cloud server's query instance list interface, the value is DescribeInstances . |
| Region | String | Yes | The region parameter is used to identify which region's data you want to operate on. The region value accepted by the interface is specified in the input parameter of the public parameter Region in the interface documentation. Note: Some interfaces do not require this parameter to be passed, which will be specifically mentioned in the interface documentation. Even if this parameter is passed, it will not take effect in this case. |
| Timestamp | Integer | Yes | The current UNIX timestamp, which can record the time when the API request is initiated. For example, 1529223702. If the difference between the current time is too large, it will cause a signature expiration error. |
| Nonce | Integer | Yes | A random positive integer combined with Timestamp is used to prevent replay attacks. |
| SecretId | String | Yes | The SecretId is an identifier applied for the cloud API key to identify the user's identity. A SecretId corresponds to a unique SecretKey , and the SecretKey is used to generate the request signature Signature . |
| Signature | String | Yes | Request signature is used to verify the legitimacy of this request. Users need to calculate it based on the actual input parameters. The specific calculation method can be found in the interface authentication documentation. |
| Version | String | Yes | The version of the API being operated on. The value refers to the description of the public parameter Version in the interface documentation. For example, the version of the cloud server. 2017-03-12. |
| SignatureMethod | String | No | Signature method, currently supports HmacSHA256 and HmacSHA1. HmacSHA256 algorithm is used to verify signatures only if this parameter is specified as HmacSHA 256, otherwise HmacSHA1 is used to verify signatures. |
| Token | String | No | Token used for temporary certificate needs to be used together with temporary key. The temporary key and Token need to be obtained from the Access Management service invocation interface. Long-term keys do not require tokens. |

# zone list

A region (Region) refers to the geographical area of a physical data center. Converge Cloud Translation: To deliver verification that completely isolates different regions, ensuring maximum stability and fault tolerance between regions. In order to reduce access latency and improve download speed, it is recommended that you choose the region closest to your customers.

You can view the complete region list by using the API interface Query Region List.

# Alarm APIs
# CreateAlarmPolicy

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

API for creating alarm policies

Default interface request rate limit: 20 times per second.

Interface update time: 2025-07-22 18:11:39.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: CreateAlarmPolicy |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| Module | Yes | No | String | The value is fixed to "monitor".<br>Example: monitor |
| PolicyName | Yes | No | String | Policy name, which can contain up to 20 characters.<br>Example: CVM alarm policy |
| Remark | No | No | String | Remarks, which can contain up to100 letters, digits, underscores, and hyphens.<br>Example: Policy demo |
| MonitorType | Yes | No | String | Monitor type. Valid values: MT_QCE (Cloud Platform service monitoring)<br>Example: MT_QCE |
| Enable | No | No | Int64 | Whether to enable. Valid values: 0 (no), 1 (yes). Default value: 1. This parameter can be left empty.<br>Example: 1 |
| ProjectId | No | No | Int64 | Project ID. Valid values: -1 (no project), 0 (default project). Default value: -1. This parameter can be left empty.<br>Example: 0 |
| Namespace | Yes | No | String | Alarm policy type such as cvm_device, which is obtained through the DescribeAllNamespaces API.<br>Example: cvm_device |
| PolicyExpression | No | No | String | Bare write alarm rules including trigger rules and filter conditions.<br>Example: 29936048 AND 2993604 |
| ConditionTemplateId | No | No | Int64 | ID of the trigger condition template. This parameter can be left empty.<br>Example: 123456 |
| Condition | No | No | AlarmPolicyCondition | Metric trigger condition.<br>Example:View |
| EventCondition | No | No | Array of AlarmPolicyEventCondition | Event trigger condition.<br>Example:View |
| NoticeIds | No | No | Array of String | List of notification rule IDs, which is obtained through the DescribeAlarmNotices API.<br>Example: ["notice-bv9b4ghqbg","notice-gj2n9wnt29"] |
| TriggerTasks | No | No | Array of AlarmPolicyTriggerTask | List of trigger tasks.<br>Example:View |
| AlarmLevel | No | No | String | Alarm Level<br>Example: Remind |

## 3. Output Parameters

| Parameter name | Type | Description |
|---|---|---|
| PolicyId | String | Alarm policy ID.<br>Example: policy-abc123 |
| OriginId | String | Policy ID for APIs for binding and unbinding instances or instance groups (BindingPolicyObject, UnBindingAllPolicyObject, UnBindingPolicyObject).<br>Example: 1234556 |

| Parameter name | Type | Description |
| --- | --- | --- |
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

## 4. Error Code

The following lists only the error codes related to the interface business logic, For other error codes, see public ErrorCode.

| Error Code | Description |
| --- | --- |
| InternalError | Internal error. |
| InvalidParameter | Invalid parameter. |
| ResourceNotFound | The resource does not exist. |
| FailedOperation | Operation failed. |
| UnsupportedOperation | Unsupported operation. |
| LimitExceeded | The quota limit has been reached. |
| InvalidParameterValue | Invalid parameter value. |

# DescribeAlarmEvents

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

This API is used to query the list of alarm events.

Default interface request rate limit: 20 times per second.

Interface update time: 2021-11-22 16:46:49.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: DescribeAlarmEvents |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| Module | Yes | No | String | Module name. The value is fixed to "monitor".<br>Example: monitor |
| Namespace | Yes | No | String | Alarm policy type such as cvm_device, which is obtained through the DescribeAllNamespaces API.<br>Example: cvm_device |
| MonitorType | No | No | String | Monitor type, such as "MT_QCE". If not specified, the default value is "MT_QCE".<br>Example: MT_QCE |

## 3. Output Parameters

| Parameter name | Type | Description |
|---|---|---|
| Events | Array of AlarmEvent | Alarm event list.<br>Example:View |
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

## 4. Error Code

The following lists only the error codes related to the interface business logic, For other error codes, see public ErrorCode.

| Error Code | Description |
|---|---|
| ResourceNotFound | The resource does not exist. |
| UnsupportedOperation | Unsupported operation. |

# DescribeAlarmMetrics

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

This API is used to query the list of alarm metrics.

Default interface request rate limit: 20 times per second.

Interface update time: 2022-11-16 10:17:56.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: DescribeAlarmMetrics |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| Module | Yes | No | String | The value is fixed to "monitor".<br>Example: monitor |
| MonitorType | Yes | No | String | Filters by monitor type. "MT_QCE" (Cloud Platform service monitoring).<br>Example: MT_QCE |
| Namespace | Yes | No | String | Alarm policy type such as cvm_device, which is obtained through the DescribeAllNamespaces API.<br>Example: cvm_device |

## 3. Output Parameters

| Parameter name | Type | Description |
|---|---|---|
| Metrics | Array of Metric | Alarm metric list.<br>Example:View |
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

## 4. Error Code

The following lists only the error codes related to the interface business logic, For other error codes, see public ErrorCode.

| Error Code | Description |
|---|---|
| ResourceNotFound | The resource does not exist. |
| UnsupportedOperation | Unsupported operation. |

# DescribeAlarmPolicy

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

This API is used to get the details of a single alarm policy.

Default interface request rate limit: 20 times per second.

Interface update time: 2022-11-22 16:25:07.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: DescribeAlarmPolicy |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| Module | Yes | No | String | The value is fixed to "monitor".<br>Example: monitor |
| PolicyId | Yes | No | String | The ID of the alarm policy.<br>Example: policy-t1pvz47f |

## 3. Output Parameters

| Parameter name | Type | Description |
|---|---|---|

| Parameter name | Type | Description |
| --- | --- | --- |
| Policy | AlarmPolicy | Policy details.<br>Example:View |
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

# 4. Error Code

The following lists only the error codes related to the interface business logic, For other error codes, see public ErrorCode.

| Error Code | Description |
| --- | --- |
| InternalError | Internal error. |
| ResourceNotFound | The resource does not exist. |
| FailedOperation | Operation failed. |
| UnsupportedOperation | Unsupported operation. |

# DescribeMiniDashboardAlarmInfo

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

MiniDashboard: pulls alarm-related information of an instance.

Default interface request rate limit: 20 times per second.

Interface update time: 2023-05-23 19:20:49.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: DescribeMiniDashboardAlarmInfo |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| StartTime | Yes | No | Int64 | The start time.<br>Example: 1611218128 |
| EndTime | Yes | No | Int64 | End time.<br>Example: 1611218200 |
| MonitorType | Yes | No | String | Monitor type. Valid values: MT_QCE (Cloud Platform service monitoring)<br>Example: MT_QCE |
| Namespace | Yes | No | String | Alarm namespace: Is used to distinguish between cloud products<br>Example: cvm_device |
| Dimensions | Yes | No | String | Instance Information of Cloud Product, JSON Format<br>Example: {"name":"instanceId","value":"1"} |
| Module | Yes | No | String | The value is fixed to "monitor".<br>Example: monitor |
| SevenDaysStartTime | No | No | Int64 | Start Time of Total Quantity of Alarms Seven Days Ago<br>Example: 1611218200 |

## 3. Output Parameters

| Parameter name | Type | Description |
|---|---|---|

| Parameter name | Type | Description |
| --- | --- | --- |
| AlarmCountInSevenDays | Int64 | Quantity of Alarms within Seven Days<br>Example: 3 |
| AlarmHistory | Array of AlarmHistory | Alarm History in Specified Period<br>Example:View |
| AlarmMetrics | Array of AlarmHistoryMetric | Alarm Metric Corresponding to Alarm History<br>Example:View |
| PoliciesInfo | Array of MiniDashboardAlarmInfo | Whether Instance Has Policy Related Information<br>Example:View |
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

# 4. Error Code

The following lists only the error codes related to the interface business logic, For other error codes, see public ErrorCode.

| Error Code | Description |
| --- | --- |
| ResourceNotFound | The resource does not exist. |
| UnsupportedOperation | Unsupported operation. |
| FailedOperation | Operation failed. |

# ModifyAlarmPolicyInfo

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

This API is used to edit the basic information of a v2.0 alarm policy, including policy name and remarks.

Default interface request rate limit: 20 times per second.

Interface update time: 2021-03-21 19:35:31.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: ModifyAlarmPolicyInfo |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| Module | Yes | No | String | Module name. Enter "monitor" here.<br>Example: monitor |
| PolicyId | Yes | No | String | Alarm policy ID.<br>Example: policy-298gn92 |
| Key | Yes | No | String | Field to be modified. NAME (policy name), REMARK (policy remarks).<br>Example: NAME |
| Value | Yes | No | String | Value after modification.<br>Example: New CVM alarm policy |

## 3. Output Parameters

| Parameter name | Type | Description |
|---|---|---|
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

# 4. Error Code

The following lists only the error codes related to the interface business logic, For other error codes, see public ErrorCode.

| Error Code | Description |
| --- | --- |
| InternalError | Internal error. |
| InvalidParameter | Invalid parameter. |
| ResourceNotFound | The resource does not exist. |
| FailedOperation | Operation failed. |
| UnsupportedOperation | Unsupported operation. |
| InvalidParameterValue | Invalid parameter value. |

# ModifyAlarmPolicyStatus

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

This API is used to enable/disable an alarm policy.

Default interface request rate limit: 20 times per second.

Interface update time: 2021-03-21 19:41:55.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: ModifyAlarmPolicyStatus |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| Module | Yes | No | String | Module name. The value is fixed to "monitor". Example: monitor |
| PolicyId | Yes | No | String | Alarm policy ID. Example: policy-2g22g82 |
| Enable | Yes | No | Int64 | Enabled or disabled status. 1: enabled. 0: disabled. Example: 0 |

## 3. Output Parameters

| Parameter name | Type | Description |
| --- | --- | --- |
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

# 4. Error Code

The following lists only the error codes related to the interface business logic, For other error codes, see public ErrorCode.

| Error Code | Description |
| --- | --- |
| InternalError | Internal error. |
| InvalidParameter | Invalid parameter. |
| ResourceNotFound | The resource does not exist. |
| FailedOperation | Operation failed. |
| InvalidParameterValue | Invalid parameter value. |

# Cloud Monitor API
# BindingPolicyObject

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

This API is used to bind an alarm policy to a specific object.

Default interface request rate limit: 20 times per second.

Interface update time: 2021-03-12 12:00:48.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: BindingPolicyObject |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| GroupId | Yes | No | Int64 | Policy group ID.<br>Example: 111111 |
| InstanceGroupId | No | No | Int64 | Instance group ID.<br>Example: 0 |
| Dimensions | No | No | Array of BindingPolicyObjectDimension | Dimensions of an object to be bound.<br>Example:View |
| Module | Yes | No | String | Required. The value is fixed to "monitor".<br>Example: monitor |
| PolicyId | No | No | String | Alarm policy ID. When this field is used, any value can be passed in for GroupId.<br>Example: policy-abc123 |

## 3. Output Parameters

| Parameter name | Type | Description |
|---|---|---|
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

## 4. Error Code

The following lists only the error codes related to the interface business logic, For other error codes, see public ErrorCode.

| Error Code | Description |
|---|---|
| InvalidParameter.InvalidParameterParam | Invalid parameter. |
| InvalidParameter | Invalid parameter. |
| AuthFailure.UnauthorizedOperation | The request is not authorized. Please refer to the CAM documentation for the description of authentication. |
| InternalError | Internal error. |

| Error Code | Description |
| --- | --- |
| FailedOperation | Operation failed. |

# CreateAlarmNotice

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

This API is used to create a notification template.

Default interface request rate limit: 20 times per second.

Interface update time: 2022-07-25 20:37:38.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: CreateAlarmNotice |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| Module | Yes | No | String | Module name. Enter "monitor" here.<br>Example: monitor |
| Name | Yes | No | String | Notification template name, which can contain up to 60 characters.<br>Example: Default notification template |
| NoticeType | Yes | No | String | Notification type. Valid values: ALARM (for unresolved alarms), OK (for resolved alarms), ALL (for all alarms)<br>Example: ALL |
| NoticeLanguage | Yes | No | String | Notification language.<br>Example: zh-CN |
| UserNotices | No | No | Array of UserNotice | Number of user notifications (up to 5).<br>Example:View |
| URLNotices | No | No | Array of URLNotice | Number of callback notifications (up to 3).<br>Example:View |
| ExtraChannels | No | No | Array of String | List of other non-public notification channels.<br>Example: ["TSS"] |
| BigCustomerNotices | No | No | Array of BigCustomerNotice | Number of major client notifications (up to 1).<br>Example:View |

## 3. Output Parameters

| Parameter name | Type | Description |
|---|---|---|
| NoticeId | String | Alarm notification template ID.<br>Example: notice-abcdef |
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

## 4. Error Code

The following lists only the error codes related to the interface business logic, For other error codes, see public ErrorCode.

| Error Code | Description |
|---|---|
| InvalidParameter | Invalid parameter. |
| InvalidParameterValue | Invalid parameter value. |
| AuthFailure.UnauthorizedOperation | The request is not authorized. Please refer to the CAM documentation for the description of authentication. |
| InternalError | Internal error. |
| ResourceNotFound | The resource does not exist. |

| Error Code | Description |
| --- | --- |
| FailedOperation | Operation failed. |

# CreateNoticeContentTmpl

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

This API is used to create a custom notification content template.

Default interface request rate limit: 20 times per second.

Interface update time: 2025-02-24 15:24:30.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: CreateNoticeContentTmpl |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| TmplName | Yes | No | String | Template name<br>Example: tempName |
| MonitorType | Yes | No | String | Monitoring type<br>Example: MT_QCE |
| TmplContents | Yes | No | NoticeContentTmplItem | Template Content<br>Example:View |
| TmplLanguage | Yes | No | String | Template language. Valid values: en and zh.<br>Example: zh |
| Module | Yes | No | String | The value is fixed to "monitor".<br>Example: monitor |

# 3. Output Parameters

| Parameter name | Type | Description |
| --- | --- | --- |
| TmplID | String | ID of the custom content template.<br>Example: tmpl-1 |
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

# 4. Error Code

There is no error code related to business logic for this interface, For other error codes, see public ErrorCode.

# CreatePolicyGroup

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

This API is used to add a policy group.

Default interface request rate limit: 20 times per second.

Interface update time: 2020-12-01 11:34:23.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: CreatePolicyGroup |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| GroupName | Yes | No | String | Group policy name.<br>Example: test_group |
| ViewName | No | No | String | Name of the view to which the policy group belongs. If the policy group is created based on a template, this parameter can be left empty.<br>Example: cvm_device |
| ProjectId | No | No | Int64 | ID of the project to which the policy group belongs, which will be used for authentication.<br>Example: 0 |
| ConditionTempGroupId | No | No | Int64 | ID of a template-based policy group. This parameter is required only when the policy group is created based on a template.<br>Example: 123 |
| IsShielded | No | No | Int64 | Whether the policy group is shielded. The value 0 indicates that the policy group is not shielded. The value 1 indicates that the policy group is shielded. The default value is 0.<br>Example: 1 |
| Remark | No | No | String | Remarks of the policy group.<br>Example: backup_group |
| InsertTime | No | No | Int64 | Insertion time in the format of Unix timestamp. If this parameter is not configured, the backend processing time is used.<br>Example: 106648246 |
| Conditions | No | No | Array of CreatePolicyGroupCondition | Alarm threshold rules in the policy group.<br>Example:View |
| EventConditions | No | No | Array of CreatePolicyGroupEventCondition | Event alarm rules in the policy group.<br>Example:View |
| Module | Yes | No | String | The value is fixed to "monitor".<br>Example: monitor |
| BackEndCall | No | No | Int64 | Whether it is a backend call. If the value is 1, rules from the policy template will be used to fill in the Conditions and EventConditions fields.<br>Example: 0 |
| IsUnionRule | No | No | Int64 | The 'AND' and 'OR' rules for alarm metrics. The value 0 indicates 'OR', which means that an alarm will be triggered when any rule is met. The value 1 indicates 'AND', which means that an alarm will be triggered only when all rules are met.<br>Example: 1 |

## 3. Output Parameters

| Parameter name | Type | Description |
|---|---|---|
| GroupId | Int64 | Created policy ID.<br>Example: 1288099 |
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

## 4. Error Code

The following lists only the error codes related to the interface business logic, For other error codes, see public ErrorCode.

| Error Code | Description |
|---|---|
| InternalError | Internal error. |
| InvalidParameter | Invalid parameter. |
| FailedOperation | Operation failed. |
| MissingParameter | Missing parameter. |
| UnknownParameter | Unknown parameter error. |
| UnauthorizedOperation | Unauthorized operation. |
| UnsupportedOperation | Unsupported operation. |
| ResourceNotFound | The resource does not exist. |
| LimitExceeded | The quota limit has been reached. |
| ResourceUnavailable | The resource is unavailable. |
| ResourceInsufficient | Insufficient resource. |
| FailedOperation | Operation failed. |
| ResourceInUse | The resource is in use. |
| ResourcesSoldOut | The resources have been sold out. |

# DeleteAlarmNotices

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

This API is used to delete alarm notification templates.

Default interface request rate limit: 20 times per second.

Interface update time: 2021-03-31 11:10:40.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: DeleteAlarmNotices |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| Module | Yes | No | String | Module name. Enter "monitor" here.<br>Example: monitor |
| NoticeIds | Yes | No | Array of String | List of alarm notification template IDs.<br>Example: ["notice-235h325y","notice-h25hj534h"] |

## 3. Output Parameters

| Parameter | Type | Description |
|---|---|---|

| | | |
|---|---|---|
| name | | |
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

# 4. Error Code

The following lists only the error codes related to the interface business logic, For other error codes, see public ErrorCode.

| Error Code | Description |
|---|---|
| InternalError | Internal error. |
| InvalidParameter | Invalid parameter. |
| ResourceNotFound | The resource does not exist. |
| FailedOperation | Operation failed. |
| AuthFailure.UnauthorizedOperation | The request is not authorized. Please refer to the CAM documentation for the description of authentication. |

# DeleteNoticeContentTmpls

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

This API is used to delete a custom notification content template.

Default interface request rate limit: 20 times per second.

Interface update time: 2025-02-24 15:24:57.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: DeleteNoticeContentTmpls |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| TmplIDs | Yes | No | Array of String | Template ID<br>Example: temp-xxx |
| Module | Yes | No | String | The value is fixed to "monitor".<br>Example: monitor |

## 3. Output Parameters

| Parameter | Type | Description |
|---|---|---|

| | | |
|---|---|---|
| name | | |
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

# 4. Error Code

There is no error code related to business logic for this interface, For other error codes, see public ErrorCode.

# DeletePolicyGroup

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

This API is used to delete an alarm policy group.

Default interface request rate limit: 20 times per second.

Interface update time: 2020-11-30 19:34:04.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: DeletePolicyGroup |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| Module | Yes | No | String | The value is fixed to "monitor".<br>Example: monitor |
| GroupId | Yes | No | Array of Int64 | Policy group ID.<br>Example: 12322 |

## 3. Output Parameters

| Parameter | Type | Description |
|---|---|---|

| | | |
|---|---|---|
| name | | |
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

# 4. Error Code

The following lists only the error codes related to the interface business logic, For other error codes, see public ErrorCode.

| Error Code | Description |
|---|---|
| InvalidParameterValue | Invalid parameter value. |
| InternalError | Internal error. |
| InvalidParameter.InvalidParameterParam | Invalid parameter. |
| InvalidParameter.InvalidParameter | Invalid parameter. |
| InvalidParameter | Invalid parameter. |
| InternalError.ExeTimeout | Execution timed out. |
| FailedOperation | Operation failed. |
| MissingParameter | Missing parameter. |
| UnknownParameter | Unknown parameter error. |
| UnauthorizedOperation | Unauthorized operation. |
| UnsupportedOperation | Unsupported operation. |
| ResourceNotFound | The resource does not exist. |
| LimitExceeded | The quota limit has been reached. |
| ResourceUnavailable | The resource is unavailable. |
| ResourceInsufficient | Insufficient resource. |
| FailedOperation | Operation failed. |
| ResourceInUse | The resource is in use. |

| Error Code | Description |
| --- | --- |
| DryRunOperation | DryRun operation, which means the DryRun parameter is passed in yet the request will still be successful. |
| ResourcesSoldOut | The resources have been sold out. |
| LimitExceeded.MetricQuotaExceeded | Quota limit on metrics is reached. Requests containing unregistered metrics are prohibited. |
| FailedOperation.ServiceNotEnabled | The service is not enabled and can be used only after being enabled. |
| FailedOperation.AlertPolicyCreateFailed | Failed to create the alarm policy. |
| FailedOperation.DbRecordCreateFailed | Failed to create the database record. |
| FailedOperation.DbTransactionBeginFailed | Failed to start the database transaction. |
| FailedOperation.DbRecordDeleteFailed | Failed to delete the database record. |
| FailedOperation.DbQueryFailed | Failed to query the database. |
| FailedOperation.DbTransactionCommitFailed | Failed to submit the database transaction. |
| FailedOperation.DbRecordUpdateFailed | Failed to update the database record. |
| AuthFailure.UnauthorizedOperation | The request is not authorized. Please refer to the CAM documentation for the description of authentication. |
| FailedOperation.DuplicateName | The name already exists. |
| FailedOperation.DruidQueryFailed | Failed to query analysis data. |
| FailedOperation.DimQueryRequestFailed | Failed to query the service in the request dimension. |
| FailedOperation.AlertPolicyDescribeFailed | Failed to query the alarm policy. |
| FailedOperation.AlertPolicyDeleteFailed | Failed to delete the alarm policy. |
| FailedOperation.AlertPolicyModifyFailed | Failed to modify the alarm policy. |
| FailedOperation.AlertTriggerRuleDeleteFailed | Failed to delete the trigger condition. |
| FailedOperation.AlertFilterRuleDeleteFailed | Failed to delete the filter rule. |

# DescribeAbnormalObjects

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

API for pulling business instances with exceptions in the past 24 hours (alarm dimension)

Default interface request rate limit: 20 times per second.

Interface update time: 2022-10-24 15:08:03.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: DescribeAbnormalObjects |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| Module | Yes | No | String | The value is fixed to "monitor".<br>Example: monitor |
| ViewName | Yes | No | String | Name of the alarm type.<br>Example: cvm_device |
| StartTime | No | No | String | Start time. Format: "2019-01-01".<br>Example: 1734950808 |
| ProjectIds | No | No | Array of Int64 | List of project IDs.<br>Example: 10 |
| AbnormalStatus | No | No | String | Alarm status, "all" indicates all instances, "abnormal" indicates abnormal instances (alarms have not been recovered), "warning" indicates warning instances (alarms have been recovered).<br>Example: abnormal |
| GroupBy | No | No | String | Whether to aggregate or not. If this parameter is left empty, it means no aggregation. "instance" means aggregation by instance.<br>Example: instance |
| Dimensions | No | No | String | JSON strings filtered by dimension.<br>Example: {"unInstanceId":"ins-11111111"} |

## 3. Output Parameters

| Parameter name | Type | Description |
|---|---|---|
| ThresholdObjects | Array of DescribeAbnormalObjectsThresholdObject | List of exceptional objects in the metric alarm<br>Example:View |
| EventObjects | Array of DescribeAbnormalObjectsEventObject | List of exceptional objects in the product event alarm<br>Example:View |
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

## 4. Error Code

The following lists only the error codes related to the interface business logic, For other error codes, see public ErrorCode.

| Error Code | Description |
|---|---|
| InvalidParameterValue | Invalid parameter value. |
| InternalError | Internal error. |
| InvalidParameter.InvalidParameterParam | Invalid parameter. |
| InvalidParameter.InvalidParameter | Invalid parameter. |
| InvalidParameter | Invalid parameter. |
| InternalError.ExeTimeout | Execution timed out. |
| FailedOperation | Operation failed. |
| MissingParameter | Missing parameter. |
| UnknownParameter | Unknown parameter error. |
| UnauthorizedOperation | Unauthorized operation. |
| UnsupportedOperation | Unsupported operation. |
| ResourceNotFound | The resource does not exist. |
| LimitExceeded | The quota limit has been reached. |
| ResourceUnavailable | The resource is unavailable. |
| ResourceInsufficient | Insufficient resource. |
| FailedOperation | Operation failed. |
| ResourceInUse | The resource is in use. |
| DryRunOperation | DryRun operation, which means the DryRun parameter is passed in yet the request will still be successful. |
| ResourcesSoldOut | The resources have been sold out. |
| LimitExceeded.MetricQuotaExceeded | Quota limit on metrics is reached. Requests containing unregistered metrics are prohibited. |

| Error Code | Description |
| --- | --- |
| FailedOperation.ServiceNotEnabled | The service is not enabled and can be used only after being enabled. |
| FailedOperation.AlertPolicyCreateFailed | Failed to create the alarm policy. |
| FailedOperation.DbRecordCreateFailed | Failed to create the database record. |
| FailedOperation.DbTransactionBeginFailed | Failed to start the database transaction. |
| FailedOperation.DbRecordDeleteFailed | Failed to delete the database record. |
| FailedOperation.DbQueryFailed | Failed to query the database. |
| FailedOperation.DbTransactionCommitFailed | Failed to submit the database transaction. |
| FailedOperation.DbRecordUpdateFailed | Failed to update the database record. |
| AuthFailure.UnauthorizedOperation | The request is not authorized. Please refer to the CAM documentation for the description of authentication. |
| FailedOperation.DuplicateName | The name already exists. |
| FailedOperation.DruidQueryFailed | Failed to query analysis data. |
| FailedOperation.DimQueryRequestFailed | Failed to query the service in the request dimension. |
| FailedOperation.AlertPolicyDescribeFailed | Failed to query the alarm policy. |
| FailedOperation.AlertPolicyDeleteFailed | Failed to delete the alarm policy. |
| FailedOperation.AlertPolicyModifyFailed | Failed to modify the alarm policy. |
| FailedOperation.AlertTriggerRuleDeleteFailed | Failed to delete the trigger condition. |
| FailedOperation.AlertFilterRuleDeleteFailed | Failed to delete the filter rule. |
| FailedOperation.DruidTableNotFound | The druid table does not exist. |
| FailedOperation.DivisionByZero | The dividend is zero. |

# DescribeAccidentEventList

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

This API is used to get the platform event list.

Default interface request rate limit: 20 times per second.

Interface update time: 2020-03-01 17:38:44.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: DescribeAccidentEventList |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| Module | Yes | No | String | API module name. The value of the current API is "monitor".<br>Example: monitor |
| StartTime | No | No | Int64 | Start time. The default value is the timestamp that is 1 day ago from this day.<br>Example: 1734900808 |
| EndTime | No | No | Int64 | End time. The default value is the current timestamp.<br>Example: 1734950808 |
| Limit | No | No | Int64 | Pagination parameter. Number of returned results per page. Value range: 1-100. Default: 20.<br>Example: 10 |
| Offset | No | No | Int64 | Pagination parameter. Page offset. It starts from 0 and the default value is 0.<br>Example: 0 |
| UpdateTimeOrder | No | No | String | Sorting rule by UpdateTime. Valid values: asc or desc.<br>Example: desc |
| OccurTimeOrder | No | No | String | Sorting rule by OccurTime. Valid values: asc or desc. Sorting by UpdateTimeOrder takes priority.<br>Example: desc |
| AccidentType | No | No | Array of Int64 | Filters by event type. The value 1 indicates service issues. The value 2 indicates other subscriptions.<br>Example: 1 |
| AccidentEvent | No | No | Array of Int64 | Filters by event. The value 1 indicates CVM storage issues. The value 2 indicates CVM network connection issues. The value 3 indicates that the CVM has an exception. The value 202 indicates that an ISP network jitter occurs.<br>Example: 1 |
| AccidentStatus | No | No | Array of Int64 | Filters by event status. The value 0 indicates that the event has been recovered. The value 1 indicates that the event has not been recovered.<br>Example: 0 |
| AccidentRegion | No | No | Array of String | Filters by event region.<br>Example: gz |
| AffectResource | No | No | String | Filters by affected resource, such as ins-19a06bka.<br>Example: ins-19a06bka |

## 3. Output Parameters

| Parameter name | Type | Description |
|---|---|---|
| Alarms | Array of DescribeAccidentEventListAlarms | List of platform events.<br>Example:View |
| Total | Int64 | Total number of platform events.<br>Example: 10 |
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

## 4. Error Code

The following lists only the error codes related to the interface business logic, For other error codes, see public ErrorCode.

| Error Code | Description |
|---|---|
| InvalidParameterValue | Invalid parameter value. |
| InternalError | Internal error. |
| InvalidParameter | Invalid parameter. |
| LimitExceeded | The quota limit has been reached. |

# DescribeAgentStatusHistory

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

API for obtaining the status history of the CVM

Default interface request rate limit: 20 times per second.

Interface update time: 2022-08-27 21:47:02.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: DescribeAgentStatusHistory |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| Module | Yes | No | String | The value is fixed to "monitor".<br>Example: monitor |
| StartTime | Yes | No | Datetime_iso | Start time of the time range to query.<br>Example: 2023-04-02T16:04:00+08:00 |
| UnInstanceIDs | Yes | No | Array of String | The IDs of the instances you want to restart.<br>Example: lhins-eqecz |
| EndTime | No | No | Datetime_iso | End time of the query.<br>Example: 2023-04-02T15:04:00+08:00 |

# 3. Output Parameters

| Parameter name | Type | Description |
| --- | --- | --- |
| Data | Array of DescribeAgentStatusHistoryData | CVM status information list.<br>Example:View |
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

# 4. Error Code

The following lists only the error codes related to the interface business logic, For other error codes, see public ErrorCode.

| Error Code | Description |
| --- | --- |
| InternalError | Internal error. |
| InvalidParameter | Invalid parameter. |
| FailedOperation | Operation failed. |
| UnauthorizedOperation | Unauthorized operation. |
| UnsupportedOperation | Unsupported operation. |
| FailedOperation | Operation failed. |

# DescribeAlarmHistories

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

V2.0 Alarm history list API

Default interface request rate limit: 20 times per second.

Interface update time: 2025-07-22 18:10:52.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: DescribeAlarmHistories |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| PageNumber | No | No | Int64 | Number of pages. It starts from 1 and the default value is 1.<br>Example: 1 |
| PageSize | No | No | Int64 | Number of entries per page. Value range: 1-100. Default: 20.<br>Example: 20 |
| Order | No | No | String | Sorts by the first occurrence time in descending order by default. Valid values: ASC (ascending), DESC (descending).<br>Example: DESC |
| StartTime | No | No | Int64 | Start time. The default value is the timestamp that is 1 day ago from this day.<br>Example: 1611600000 |
| EndTime | No | No | Int64 | End time. The default value is the current timestamp.<br>Example: 1611650000 |
| MonitorTypes | No | No | Array of String | Filters by monitor type. Valid values: "MT_QCE" (Cloud Platform service monitoring), "MT_CUSTOM" (custom monitoring), "MT_PROME"=prometheus monitoring. If this parameter is left empty, all types will be queried by default.<br>Example: MT_QCE |
| AlarmObject | No | No | String | Filter by alarm object. Fuzzy search with string is supported.<br>Example: tdsqlshard-h2phzy69 |
| AlarmStatus | No | No | Array of String | Filters by alarm status. Valid values: ALARM (not resolved), OK (resolved), NO_CONF (expired), NO_DATA (insufficient data). If this parameter is left empty, all will be queried by default.<br>Example: ALARM |
| ProjectIds | No | No | Array of Int64 | Filters by project ID. Valid values: -1 (no project), 0 (default project).<br>Example: 10 |
| InstanceGroupIds | No | No | Array of Int64 | Filters by instance group ID<br>Example: 10 |
| Namespaces | No | No | MonitorTypeNamespace | Filters by policy type.<br>Example:View |
| PolicyName | No | No | String | Fuzzy search by policy name.<br>Example: Testing |
| Content | No | No | String | Fuzzy search by alarm content.<br>Example: abc |
| ReceiverUids | No | No | Array of Int64 | Searches by recipient<br>Example: 1234 |
| ReceiverGroups | No | No | Array of Int64 | Searches by recipient group.<br>Example: 1234 |
| Module | Yes | No | String | The value is fixed to "monitor".<br>Example: monitor |
| PolicyIds | No | No | Array of String | Searches by alarm policy ID list.<br>Example: policy-abcedf |
| MetricNames | No | No | Array of String | Filters by metric name.<br>Example: CpuUsage |
| UniqueIds | No | No | Array of String | Searches by unique ID of the alarm instance.<br>Example: dadeqe13czc |
| Dimensions | No | No | String | Dimension information of an instance.<br>Example: [[{"name":"unInstanceId","value":"ins-abcdef"}]] |
| AlarmLevel | No | No | Array of String | Alarm Level<br>Example: Remind |

## 3. Output Parameters

| Parameter name | Type | Description |
|---|---|---|
| TotalCount | Int64 | Total number.<br>Example: 100 |
| Histories | Array of AlarmHistory | Alarm history list.<br>Example:View |
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

## 4. Error Code

The following lists only the error codes related to the interface business logic, For other error codes, see public ErrorCode.

| Error Code | Description |
|---|---|
| InternalError | Internal error. |
| InvalidParameter | Invalid parameter. |
| LimitExceeded | The quota limit has been reached. |
| InvalidParameterValue | Invalid parameter value. |
| MissingParameter | Missing parameter. |

# DescribeAlarmNotice

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

This API is used to query the details of a single notification template.

Default interface request rate limit: 20 times per second.

Interface update time: 2021-03-16 18:31:33.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: DescribeAlarmNotice |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| Module | Yes | No | String | Module name. Enter "monitor" here.<br>Example: monitor |
| NoticeId | Yes | No | String | Alarm notification template ID.<br>Example: notice-2g892hg8 |

## 3. Output Parameters

| Parameter name | Type | Description |
|---|---|---|

| Parameter name | Type | Description |
| --- | --- | --- |
| Notice | AlarmNotice | Alarm notification template details.<br>Example:View |
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

# 4. Error Code

The following lists only the error codes related to the interface business logic, For other error codes, see public ErrorCode.

| Error Code | Description |
| --- | --- |
| InternalError | Internal error. |
| InvalidParameter | Invalid parameter. |
| ResourceNotFound | The resource does not exist. |
| FailedOperation | Operation failed. |
| UnsupportedOperation | Unsupported operation. |
| AuthFailure.UnauthorizedOperation | The request is not authorized. Please refer to the CAM documentation for the description of authentication. |
| InvalidParameterValue | Invalid parameter value. |

# DescribeAlarmNoticeCallbacks

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

This API is used to get all the callback URLs of an alarm notification template.

Default interface request rate limit: 20 times per second.

Interface update time: 2021-03-01 14:40:46.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: DescribeAlarmNoticeCallbacks |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| Module | Yes | No | String | Module name. Enter "monitor" here.<br>Example: monitor |

## 3. Output Parameters

| Parameter name | Type | Description |
|---|---|---|
| URLNotices | Array of URLNotice | Alarm callback notification.<br>Example:View |

| Parameter name | Type | Description |
| --- | --- | --- |
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

# 4. Error Code

The following lists only the error codes related to the interface business logic, For other error codes, see public ErrorCode.

| Error Code | Description |
| --- | --- |
| InvalidParameter | Invalid parameter. |
| InvalidParameterValue | Invalid parameter value. |
| UnsupportedOperation | Unsupported operation. |
| AuthFailure.UnauthorizedOperation | The request is not authorized. Please refer to the CAM documentation for the description of authentication. |
| InternalError | Internal error. |
| ResourceNotFound | The resource does not exist. |
| FailedOperation | Operation failed. |
| MissingParameter | Missing parameter. |

# DescribeAlarmNotices

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

This API is used to query the list of notification templates.

Default interface request rate limit: 20 times per second.

Interface update time: 2022-11-22 16:04:12.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: DescribeAlarmNotices |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| Module | Yes | No | String | Module name. Enter "monitor" here.<br>Example: monitor |
| OwnerUid | No | No | Int64 | Root account UID, which is used to create preset notifications.<br>Example: 1234567 |
| Name | No | No | String | Alarm notification template name, which is used for fuzzy search.<br>Example: template |
| ReceiverType | No | No | String | Filter by recipient. The type of notified users should be selected for the alarm notification template. Valid values: USER (user), GROUP (user group). If this parameter is left empty, no filter by recipient will be performed.<br>Example: USER |
| UserIds | No | No | Array of Int64 | Recipient object list.<br>Example: [1234567] |
| GroupIds | No | No | Array of Int64 | Recipient group list.<br>Example: [1234567] |
| PageNumber | Yes | No | Int64 | Page number. Minimum value: 1.<br>Example: 1 |
| PageSize | Yes | No | Int64 | Number of entries per page. Value range: 1–200.<br>Example: 20 |
| Order | Yes | No | String | Sorts by update time. Valid values: ASC (ascending), DESC (descending).<br>Example: DESC |
| AMPConsumerIds | No | No | Array of String | Filters by the background amp consumer ID list. If this parameter is left empty, the filter operation will not be performed.<br>Example: ["Consumer-9gfn2g92hg92"] |
| NoticeIds | No | No | Array of String | Filters by notification template ID. If an empty array is passed in or if this parameter is left empty, the filter operation will not be performed.<br>Example: ["notice-ah28cv9d"] |

## 3. Output Parameters

| Parameter name | Type | Description |
|---|---|---|
| TotalCount | Int64 | Total number of alarm notification templates.<br>Example: 1 |
| Notices | Array of AlarmNotice | List of alarm notification templates.<br>Example:View |
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

## 4. Error Code

The following lists only the error codes related to the interface business logic, For other error codes, see public ErrorCode.

| Error Code | Description |
|---|---|
| InternalError | Internal error. |
| InvalidParameter | Invalid parameter. |
| ResourceNotFound | The resource does not exist. |
| FailedOperation | Operation failed. |
| UnsupportedOperation | Unsupported operation. |
| AuthFailure.UnauthorizedOperation | The request is not authorized. Please refer to the CAM documentation for the description of authentication. |
| InvalidParameterValue | Invalid parameter value. |

# DescribeAlarmPolicies

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

This API is used to query the list of alarm policies.

Default interface request rate limit: 20 times per second.

Interface update time: 2025-07-22 18:09:06.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: DescribeAlarmPolicies |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| Module | Yes | No | String | The value is fixed to "monitor".<br>Example: monitor |
| PageNumber | No | No | Int64 | Number of pages. It starts from 1 and the default value is 1.<br>Example: 1 |
| PageSize | No | No | Int64 | Number of entries per page. Value range: 1-100. Default: 20.<br>Example: 20 |
| PolicyName | No | No | String | Fuzzy search by policy name.<br>Example: policy-24sxe890 |
| MonitorTypes | No | No | Array of String | Filters by monitor type. MT_QCE (Cloud Platform service monitoring). If this parameter is left empty, all will be queried by default.<br>Example: ["MT_QCE"] |
| Namespaces | No | No | Array of String | Filters by namespace.<br>Example: ["cvm_device"] |
| Measurements | No | No | Array of String | Filters by policy type (viewName) (for Cloud Platform service monitoring).<br>Example: [cvm] |
| Dimensions | No | No | String | Alarm object list.<br>Example: [[{"name":"unInstanceId","value":"ins-qr888845g"}]] |
| ReceiverUids | No | No | Array of Int64 | Searches by recipient<br>Example: [9989] |
| ReceiverGroups | No | No | Array of Int64 | Searches by recipient group.<br>Example: [1112] |
| PolicyType | No | No | Array of String | Filters by default policy. Valid values: DEFAULT (display default policy), NOT_DEFAULT (display non-default policies). If this parameter is left empty, all policies will be displayed.<br>Example: ["DEFAULT"] |
| Field | No | No | String | Sort field.<br>Example: PolicyId |
| Order | No | No | String | Sort order. Valid values: ASC (ascending), DESC (descending).<br>Example: DESC |
| ProjectIds | No | No | Array of Int64 | Array of project IDs.<br>Example: [1] |
| AlarmStatus | No | No | Array of Int64 | Obsolete field. Enable or disable. 1: enable. 0: disable.<br>Example: [1] |
| NoticeIds | No | No | Array of String | Alarm notification ID list.<br>Example: [1,2] |
| RuleTypes | No | No | Array of String | Filters by trigger condition. Valid values: STATIC (display policies with static threshold), DYNAMIC (display policies with dynamic threshold). If this parameter is left empty, all policies will be displayed.<br>Example: [STATIC] |
| Enable | No | No | Array of Int64 | Enable or disable. 1: enable. 0: disable.<br>Example: 1 |
| NotBindingNoticeRule | No | No | Int64 | Whether the notification rule is configured, 1: not configured; 0: configured.<br>Example: 1 |
| InstanceGroupId | No | No | Int64 | Instance group ID.<br>Example: 1 |
| ConditionTemplateId | No | No | Int64 | Condition template ID.<br>Example: 1 |
| Like | No | No | String | Fuzzy match.<br>Example: Abandoned |
| Limit | No | No | String | limit.<br>Example: 10 |
| Offset | No | No | Int64 | Offset.<br>Example: 0 |
| AlarmLevels | No | No | Array of String | Alarm Level<br>Example: Remind |

## 3. Output Parameters

| Parameter name | Type | Description |
|---|---|---|
| TotalCount | Int64 | Total number of policies.<br>Example: 100 |

| Parameter name | Type | Description |
| --- | --- | --- |
| Policies | Array of AlarmPolicy | Policy array.<br>Example:View |
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

## 4. Error Code

The following lists only the error codes related to the interface business logic, For other error codes, see public ErrorCode.

| Error Code | Description |
| --- | --- |
| UnsupportedOperation | Unsupported operation. |
| InternalError | Internal error. |
| FailedOperation | Operation failed. |

# DescribeAllNamespaces

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

API for pulling all namespaces.

Default interface request rate limit: 20 times per second.

Interface update time: 2022-10-27 19:42:25.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: DescribeAllNamespaces |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| SceneType | Yes | No | String | Filter by monitoring types, "ST_DASHBOARD"=Cloud product monitoring, and "ST_ALARM"=Cloud monitoring alarm.<br>Example: ST_ALARM |
| MonitorTypes | No | No | Array of String | Filters by namespace ID. If this parameter is left empty, all will be queried.<br>Example: ["MT_QCE"] |
| Ids | No | No | Array of String | Filters by use case. Valid values: "ST_DASHBOARD" (dashboard type), "ST_ALARM"(alarm type).<br>Example: cvm_device |
| Module | Yes | No | String | The value is fixed to "monitor".<br>Example: monitor |

## 3. Output Parameters

| Parameter name | Type | Description |
|---|---|---|
| QceNamespaces | Array of CommonNamespace | Cloud product namespace.<br>Example:View |
| CustomNamespaces | Array of CommonNamespace | Custom monitoring namespace.<br>Example:View |
| CommonNamespaces | Array of String | CommonNamespaces<br>Example: {"Id":"performance_metric","MonitorType":"MT_TAW","Name":"Performance metric","Dimensions":[]} |
| CustomNamespacesNew | Array of String | CustomNamespacesNew<br>Example:<br>{"Id":"=wenlong_sidecar","Name":"=wenlong_sidecar","ProductName":"=wenlong_sidecar","Value":"=wenlong_sidecar","Config":"","AvailableRegions":[],"DashboardId":"=wenlong_sidecar","SortId":0} |
| QceNamespacesNew | Array of DashboardNamespace | QceNamespacesNew<br>Example:View |
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

## 4. Error Code

There is no error code related to business logic for this interface, For other error codes, see public ErrorCode.

# DescribeBaseMetrics

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

API for obtaining the details of basic metrics.

Default interface request rate limit: 20 times per second.

Interface update time: 2021-03-31 19:56:55.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: DescribeBaseMetrics |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| Namespace | Yes | No | String | Service namespace.<br>Example: QCE/CVM |
| MetricName | No | No | String | Metric name.<br>Example: AccOuttraffic |

## 3. Output Parameters

| Parameter name | Type | Description |
|---|---|---|

| Parameter name | Type | Description |
| --- | --- | --- |
| MetricSet | Array of MetricSet | List of queried metric descriptions.<br>Example:View |
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

# 4. Error Code

There is no error code related to business logic for this interface, For other error codes, see public ErrorCode.

# DescribeBasicAlarmList

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

This API is used to get the basic alarm list.

Default interface request rate limit: 20 times per second.

Interface update time: 2020-09-25 15:17:37.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: DescribeBasicAlarmList |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| Module | Yes | No | String | API module name. The current value is "monitor".<br>Example: monitor |
| StartTime | No | No | Int64 | Start time. The default value is the timestamp that is 1 day ago from this day.<br>Example: 1573660800 |
| EndTime | No | No | Int64 | End time. The default value is the current timestamp.<br>Example: 1573919999 |
| Limit | No | No | Int64 | Pagination parameter. Number of returned results per page. Value range: 1-100. Default: 20.<br>Example: 20 |
| Offset | No | No | Int64 | Pagination parameter. Page offset. It starts from 0 and the default value is 0.<br>Example: 0 |
| OccurTimeOrder | No | No | String | Sorts by the occurrence time. Valid values: ASC, DESC.<br>Example: DESC |
| ProjectIds | No | No | Array of Int64 | Filters by project ID.<br>Example: ["0"] |
| ViewNames | No | No | Array of String | Filters by policy type.<br>Example: ["cvm_device"] |
| AlarmStatus | No | No | Array of Int64 | Filters by alarm status.<br>Example: ["1"] |
| ObjLike | No | No | String | Filters by alarm object.<br>Example: 123 |
| InstanceGroupIds | No | No | Array of Int64 | Filters by instance group ID<br>Example: [11233] |

## 3. Output Parameters

| Parameter name | Type | Description |
| --- | --- | --- |
| Alarms | Array of DescribeBasicAlarmListAlarms | List of alarms.<br>Example:View |
| Total | Int64 | Total number.<br>Example: 2 |
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

## 4. Error Code

The following lists only the error codes related to the interface business logic, For other error codes, see public ErrorCode.

| Error Code | Description |
| --- | --- |
| InvalidParameterValue | Invalid parameter value. |
| InternalError | Internal error. |
| InvalidParameter | Invalid parameter. |
| MissingParameter | Missing parameter. |
| LimitExceeded | The quota limit has been reached. |

# DescribeBindingPolicyObjectList

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

This API is used to get the bound object list.

Default interface request rate limit: 20 times per second.

Interface update time: 2022-11-22 16:13:15.

The interface only verifies the signature but does not authenticate.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: DescribeBindingPolicyObjectList |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| Module | Yes | No | String | The value is fixed to "monitor".<br>Example: monitor |
| GroupId | Yes | No | Int64 | Policy group ID.<br>Example: 11111 |
| Limit | No | No | Int64 | Pagination parameter. Number of returned results per page. Value range: 1-100. Default: 20.<br>Example: 20 |
| Offset | No | No | Int64 | Pagination parameter. Page offset. It starts from 0 and the default value is 0.<br>Example: 0 |
| Dimensions | No | No | Array of DescribeBindingPolicyObjectListDimension | Dimension information of the filtered object.<br>Example:View |

## 3. Output Parameters

| Parameter name | Type | Description |
|---|---|---|
| List | Array of DescribeBindingPolicyObjectListInstance | List of bound object instances.<br>Example:View |
| Total | Int64 | Total number of bound object instances.<br>Example: 3 |
| NoShieldedSum | Int64 | Number of unshielded object instances.<br>Example: 3 |
| InstanceGroup | DescribeBindingPolicyObjectListInstanceGroup | Information about bound instance groups. Leave it empty if no instance group is bound.<br>Example:View |
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

## 4. Error Code

The following lists only the error codes related to the interface business logic, For other error codes, see public ErrorCode.

| Error Code | Description |
|---|---|
| InvalidParameterValue | Invalid parameter value. |
| InternalError | Internal error. |
| InvalidParameter.InvalidParameterParam | Invalid parameter. |

| Error Code | Description |
| --- | --- |
| InvalidParameter.InvalidParameter | Invalid parameter. |
| InvalidParameter | Invalid parameter. |
| InternalError.ExeTimeout | Execution timed out. |
| FailedOperation | Operation failed. |
| MissingParameter | Missing parameter. |
| UnknownParameter | Unknown parameter error. |
| UnauthorizedOperation | Unauthorized operation. |
| UnsupportedOperation | Unsupported operation. |
| ResourceNotFound | The resource does not exist. |
| LimitExceeded | The quota limit has been reached. |
| ResourceUnavailable | The resource is unavailable. |
| ResourceInsufficient | Insufficient resource. |
| FailedOperation | Operation failed. |
| ResourceInUse | The resource is in use. |
| DryRunOperation | DryRun operation, which means the DryRun parameter is passed in yet the request will still be successful. |
| ResourcesSoldOut | The resources have been sold out. |
| LimitExceeded.MetricQuotaExceeded | Quota limit on metrics is reached. Requests containing unregistered metrics are prohibited. |
| FailedOperation.ServiceNotEnabled | The service is not enabled and can be used only after being enabled. |
| FailedOperation.AlertPolicyCreateFailed | Failed to create the alarm policy. |
| FailedOperation.DbRecordCreateFailed | Failed to create the database record. |
| FailedOperation.DbTransactionBeginFailed | Failed to start the database transaction. |
| FailedOperation.DbRecordDeleteFailed | Failed to delete the database record. |
| FailedOperation.DbQueryFailed | Failed to query the database. |
| FailedOperation.DbTransactionCommitFailed | Failed to submit the database transaction. |
| FailedOperation.DbRecordUpdateFailed | Failed to update the database record. |
| AuthFailure.UnauthorizedOperation | The request is not authorized. Please refer to the CAM documentation for the description of authentication. |
| FailedOperation.DuplicateName | The name already exists. |
| FailedOperation.DruidQueryFailed | Failed to query analysis data. |
| FailedOperation.DimQueryRequestFailed | Failed to query the service in the request dimension. |
| FailedOperation.AlertPolicyDescribeFailed | Failed to query the alarm policy. |
| FailedOperation.AlertPolicyDeleteFailed | Failed to delete the alarm policy. |
| FailedOperation.AlertPolicyModifyFailed | Failed to modify the alarm policy. |
| FailedOperation.AlertTriggerRuleDeleteFailed | Failed to delete the trigger condition. |
| FailedOperation.AlertFilterRuleDeleteFailed | Failed to delete the filter rule. |
| FailedOperation.DruidTableNotFound | The druid table does not exist. |
| FailedOperation.DivisionByZero | The dividend is zero. |

# DescribeMonitorProducts

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

Queries the monitoring product list.

Default interface request rate limit: 20 times per second.

Interface update time: 2022-08-25 14:20:38.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: DescribeMonitorProducts |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| Module | Yes | No | String | The value is fixed to "monitor".<br>Example: monitor |

## 3. Output Parameters

| Parameter name | Type | Description |
|---|---|---|
| ProductList | String | Cloud product list. JSON string.<br>Example: {"avaliableRegions":[],"id":"dcchannel","isShowInDashboard":true,"meta":{"alarms":[{"dimensions":<br>{"directconnectconnid":"directConnectTunnelId"},"policyViewName":"dcchannel"}],"columns":[{"enName":"Channel ID/Name","isShowInMonitorList":true,"keys":<br>["directConnectTunnelId","directConnectTunnelName"],"name":"\u901a\u9053ID/\u540d\u79f0","required":true},<br>{"enName":"Status","isShowInMonitorList":true,"isShowInSelector":false,"name":"\u72b6\u6001","renderConditions":{"key":"status","switch":[[0,<br>{"enShow":"Connected","show":"\u5df2\u8fde\u63a5"},"success"],[1,{"enShow":"Applying","show":"\u7533\u8bf7\u4e2d"}],[2,{"enShow":"Configuring","show":"\u914d\u7f6e\u4e2d"}],[20,<br>{"enShow":"Waiting for connection","show":"\u7b49\u5f85\u8fde\u63a5"}],[21,{"enShow":"Reject","show":"\u62d2\u7edd"},"danger"]]}},<br>{"enName":"Physical Dedicated Line","isShowInMonitorList":false,"key":"directConnectId","name":"\u7269\u7406\u4e13\u7ebf"},<br>{"enName":"Associated dedicated line gateway","isShowInMonitorList":false,"isShowInSelector":false,"key":"directConnectGatewayId","name":"\u5173\u8054\u4e13\u7ebf\u7f51\u5173"},<br>{"enName":"Virtual Private Cloud","isShowInMonitorList":true,"keys":["vpcId","vpcName"],"name":"\u79c1\u6709\u7f51\u7edc"}],"enName":"VPC -<br>Direct Connect tunnel","enProductName":"VPC - Direct Connect tunnel","enSearchPlaceHolder":"Direct Connect Channel ID","id":"dcchannel","idKey":<br>["directConnectTunnelId","directConnectConnId"],"instanceLoader":{"cmd":"DescribeDirectConnectTunnels","fieldsMapping":<br>{"directConnectConnId":"directConnectTunnelId"},"fieldsToInstances":{"isArray":false,"reqParams":{"directConnectTunnelId":"<%=obj.directconnectconnid |
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

## 4. Error Code

The following lists only the error codes related to the interface business logic, For other error codes, see public ErrorCode.

| Error Code | Description |
|---|---|
| InvalidParameterValue | Invalid parameter value. |
| InternalError | Internal error. |
| InvalidParameter.InvalidParameterParam | Invalid parameter. |
| InvalidParameter.InvalidParameter | Invalid parameter. |
| InvalidParameter | Invalid parameter. |
| InternalError.ExeTimeout | Execution timed out. |
| FailedOperation | Operation failed. |
| MissingParameter | Missing parameter. |
| UnknownParameter | Unknown parameter error. |
| UnauthorizedOperation | Unauthorized operation. |
| UnsupportedOperation | Unsupported operation. |
| ResourceNotFound | The resource does not exist. |
| LimitExceeded | The quota limit has been reached. |
| ResourceUnavailable | The resource is unavailable. |
| ResourceInsufficient | Insufficient resource. |
| FailedOperation | Operation failed. |
| ResourceInUse | The resource is in use. |
| DryRunOperation | DryRun operation, which means the DryRun parameter is passed in yet the request will still be successful. |

| Error Code | Description |
| --- | --- |
| ResourcesSoldOut | The resources have been sold out. |
| LimitExceeded.MetricQuotaExceeded | Quota limit on metrics is reached. Requests containing unregistered metrics are prohibited. |
| FailedOperation.ServiceNotEnabled | The service is not enabled and can be used only after being enabled. |
| FailedOperation.AlertPolicyCreateFailed | Failed to create the alarm policy. |
| FailedOperation.DbRecordCreateFailed | Failed to create the database record. |
| FailedOperation.DbTransactionBeginFailed | Failed to start the database transaction. |
| FailedOperation.DbRecordDeleteFailed | Failed to delete the database record. |
| FailedOperation.DbQueryFailed | Failed to query the database. |
| FailedOperation.DbTransactionCommitFailed | Failed to submit the database transaction. |
| FailedOperation.DbRecordUpdateFailed | Failed to update the database record. |
| AuthFailure.UnauthorizedOperation | The request is not authorized. Please refer to the CAM documentation for the description of authentication. |
| FailedOperation.DuplicateName | The name already exists. |
| FailedOperation.DruidQueryFailed | Failed to query analysis data. |
| FailedOperation.DimQueryRequestFailed | Failed to query the service in the request dimension. |
| FailedOperation.AlertPolicyDescribeFailed | Failed to query the alarm policy. |
| FailedOperation.AlertPolicyDeleteFailed | Failed to delete the alarm policy. |
| FailedOperation.AlertPolicyModifyFailed | Failed to modify the alarm policy. |
| FailedOperation.AlertTriggerRuleDeleteFailed | Failed to delete the trigger condition. |
| FailedOperation.AlertFilterRuleDeleteFailed | Failed to delete the filter rule. |

# DescribeNoticeContentTmpl

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

This API is used to query custom notification content templates.

Default interface request rate limit: 20 times per second.

Interface update time: 2025-02-24 15:23:52.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: DescribeNoticeContentTmpl |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| TmplIDs | No | No | Array of String | Template ID<br>Example: temp-xxx |
| TmplName | No | No | String | Template name<br>Example: tmplName |
| TmplLanguage | No | No | String | Template language. Valid values: en and zh. If this parameter is left blank, filtering is not performed.<br>Example: zh |
| PageNumber | Yes | No | Uint64 | Page number.<br>Example: 1 |
| PageSize | Yes | No | Uint64 | Page Size<br>Example: 10 |
| Module | Yes | No | String | The value is fixed to "monitor".<br>Example: monitor |

## 3. Output Parameters

| Parameter name | Type | Description |
|---|---|---|
| NoticeContentTmpls | Array of NoticeContentTmpl | Custom notification content template.<br>Example:View |
| NoticeContentTmplBindPolicyCounts | Array of NoticeContentTmplBindPolicyCount | Number of alarm policies bound to the notification content template.<br>Example:View |
| PageNumber | Uint64 | Page number.<br>Example: 0 |
| PageSize | Uint64 | Page Size<br>Example: 0 |

| Parameter name | Type | Description |
| --- | --- | --- |
| TotalCount | Uint64 | Number of returned results.<br>Example: 0 |
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

# 4. Error Code

There is no error code related to business logic for this interface, For other error codes, see public ErrorCode.

# DescribePolicyConditionList

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

This API is used to get basic alarm policy conditions.

Default interface request rate limit: 20 times per second.

Interface update time: 2022-10-27 16:28:47.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: DescribePolicyConditionList |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| Module | Yes | No | String | The value is fixed to "monitor".<br>Example: monitor |

## 3. Output Parameters

| Parameter name | Type | Description |
|---|---|---|
| Conditions | Array of DescribePolicyConditionListCondition | Alarm policy condition list.<br>Example:View |

| Parameter name | Type | Description |
| --- | --- | --- |
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

# 4. Error Code

The following lists only the error codes related to the interface business logic, For other error codes, see public ErrorCode.

| Error Code | Description |
| --- | --- |
| InvalidParameterValue | Invalid parameter value. |
| InternalError | Internal error. |
| InvalidParameter.InvalidParameterParam | Invalid parameter. |
| InvalidParameter.InvalidParameter | Invalid parameter. |
| InvalidParameter | Invalid parameter. |
| InternalError.ExeTimeout | Execution timed out. |
| FailedOperation | Operation failed. |
| MissingParameter | Missing parameter. |
| UnknownParameter | Unknown parameter error. |
| UnauthorizedOperation | Unauthorized operation. |
| UnsupportedOperation | Unsupported operation. |
| ResourceNotFound | The resource does not exist. |
| LimitExceeded | The quota limit has been reached. |
| ResourceUnavailable | The resource is unavailable. |
| ResourceInsufficient | Insufficient resource. |
| FailedOperation | Operation failed. |
| ResourceInUse | The resource is in use. |

| Error Code | Description |
| --- | --- |
| DryRunOperation | DryRun operation, which means the DryRun parameter is passed in yet the request will still be successful. |
| ResourcesSoldOut | The resources have been sold out. |
| LimitExceeded.MetricQuotaExceeded | Quota limit on metrics is reached. Requests containing unregistered metrics are prohibited. |
| FailedOperation.ServiceNotEnabled | The service is not enabled and can be used only after being enabled. |
| FailedOperation.AlertPolicyCreateFailed | Failed to create the alarm policy. |
| FailedOperation.DbRecordCreateFailed | Failed to create the database record. |
| FailedOperation.DbTransactionBeginFailed | Failed to start the database transaction. |
| FailedOperation.DbRecordDeleteFailed | Failed to delete the database record. |
| FailedOperation.DbQueryFailed | Failed to query the database. |
| FailedOperation.DbTransactionCommitFailed | Failed to submit the database transaction. |
| FailedOperation.DbRecordUpdateFailed | Failed to update the database record. |
| AuthFailure.UnauthorizedOperation | The request is not authorized. Please refer to the CAM documentation for the description of authentication. |
| FailedOperation.DuplicateName | The name already exists. |
| FailedOperation.DruidQueryFailed | Failed to query analysis data. |
| FailedOperation.DimQueryRequestFailed | Failed to query the service in the request dimension. |
| FailedOperation.AlertPolicyDescribeFailed | Failed to query the alarm policy. |
| FailedOperation.AlertPolicyDeleteFailed | Failed to delete the alarm policy. |
| FailedOperation.AlertPolicyModifyFailed | Failed to modify the alarm policy. |
| FailedOperation.AlertTriggerRuleDeleteFailed | Failed to delete the trigger condition. |
| FailedOperation.AlertFilterRuleDeleteFailed | Failed to delete the filter rule. |

# DescribePolicyGroupInfo

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

This API is used to get details of a basic policy group.

Default interface request rate limit: 20 times per second.

Interface update time: 2020-10-12 11:25:49.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: DescribePolicyGroupInfo |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| Module | Yes | No | String | The value is fixed to "monitor".<br>Example: monitor |
| GroupId | Yes | No | Int64 | Policy group ID.<br>Example: 123456 |

## 3. Output Parameters

| Parameter name | Type | Description |
|---|---|---|
| GroupName | String | Policy group name.<br>Example: Policies |
| ProjectId | Int64 | Project ID to which the policy group belongs.<br>Example: 0 |

| Parameter name | Type | Description |
| --- | --- | --- |
| IsDefault | Int64 | Whether it is a default policy. 0: non-default policy. 1: default policy.<br>Example: 1 |
| ViewName | String | Policy type<br>Example: cvm_device |
| Remark | String | Policy description.<br>Example: Testing |
| ShowName | String | Policy type name.<br>Example: Basic CVM monitoring |
| LastEditUin | String | UIN of the last editor.<br>Example: 123 |
| UpdateTime | String | Last modification time.<br>Example: 1718159371 |
| Region | Array of String | Regions supported by this policy.<br>Example: bj |
| DimensionGroup | Array of String | List of dimensions for policy types.<br>Example: uuid |
| ConditionsConfig | Array of DescribePolicyGroupInfoCondition | Threshold rules list.<br>Example:View |
| EventConfig | Array of DescribePolicyGroupInfoEventCondition | List of product event rules.<br>Example:View |
| ReceiverInfos | Array of DescribePolicyGroupInfoReceiverInfo | User recipient list.<br>Example:View |
| Callback | DescribePolicyGroupInfoCallback | User callback information.<br>Example:View |
| ConditionsTemp | DescribePolicyGroupInfoConditionTpl | Template-based policy group.<br>Example:View |
| CanSetDefault | Bool | Whether it can be set as the default policy.<br>Example: false |
| IsUnionRule | Int64 | Whether it is an AND rule.<br>Example: 1 |
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

# 4. Error Code

The following lists only the error codes related to the interface business logic, For other error codes, see public ErrorCode.

| Error Code | Description |
| --- | --- |
| InvalidParameterValue | Invalid parameter value. |
| InternalError | Internal error. |
| InvalidParameter.InvalidParameterParam | Invalid parameter. |
| InvalidParameter.InvalidParameter | Invalid parameter. |
| InvalidParameter | Invalid parameter. |
| InternalError.ExeTimeout | Execution timed out. |
| FailedOperation | Operation failed. |
| MissingParameter | Missing parameter. |
| UnknownParameter | Unknown parameter error. |
| UnauthorizedOperation | Unauthorized operation. |
| UnsupportedOperation | Unsupported operation. |
| ResourceNotFound | The resource does not exist. |
| LimitExceeded | The quota limit has been reached. |
| ResourceUnavailable | The resource is unavailable. |
| ResourceInsufficient | Insufficient resource. |
| FailedOperation | Operation failed. |
| ResourceInUse | The resource is in use. |
| DryRunOperation | DryRun operation, which means the DryRun parameter is passed in yet the request will still be successful. |
| ResourcesSoldOut | The resources have been sold out. |
| LimitExceeded.MetricQuotaExceeded | Quota limit on metrics is reached. Requests containing unregistered metrics are prohibited. |
| FailedOperation.ServiceNotEnabled | The service is not enabled and can be used only after being enabled. |
| FailedOperation.AlertPolicyCreateFailed | Failed to create the alarm policy. |

| Error Code | Description |
| --- | --- |
| FailedOperation.DbRecordCreateFailed | Failed to create the database record. |
| FailedOperation.DbTransactionBeginFailed | Failed to start the database transaction. |
| FailedOperation.DbRecordDeleteFailed | Failed to delete the database record. |
| FailedOperation.DbQueryFailed | Failed to query the database. |
| FailedOperation.DbTransactionCommitFailed | Failed to submit the database transaction. |
| FailedOperation.DbRecordUpdateFailed | Failed to update the database record. |
| AuthFailure.UnauthorizedOperation | The request is not authorized. Please refer to the CAM documentation for the description of authentication. |
| FailedOperation.DuplicateName | The name already exists. |
| FailedOperation.DruidQueryFailed | Failed to query analysis data. |
| FailedOperation.DimQueryRequestFailed | Failed to query the service in the request dimension. |
| FailedOperation.AlertPolicyDescribeFailed | Failed to query the alarm policy. |
| FailedOperation.AlertPolicyDeleteFailed | Failed to delete the alarm policy. |
| FailedOperation.AlertPolicyModifyFailed | Failed to modify the alarm policy. |
| FailedOperation.AlertTriggerRuleDeleteFailed | Failed to delete the trigger condition. |
| FailedOperation.AlertFilterRuleDeleteFailed | Failed to delete the filter rule. |

# DescribePolicyGroupList

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

This API is used to get the list of basic policy alarm groups.

Default interface request rate limit: 20 times per second.

Interface update time: 2022-10-24 10:18:55.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: DescribePolicyGroupList |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| Module | Yes | No | String | The value is fixed to "monitor".<br>Example: monitor |
| Limit | Yes | No | Int64 | Pagination parameter, the number of entries per page. Value range: 1-1000.<br>Example: 10 |
| Offset | Yes | No | Int64 | Pagination parameter. Page offset. It starts from 0.<br>Example: 0 |
| Like | No | No | String | Searches by policy name.<br>Example: test |
| InstanceGroupId | No | No | Int64 | Instance group ID.<br>Example: 12345 |
| UpdateTimeOrder | No | No | String | Sorts by update time. Valid values: asc, desc.<br>Example: asc |
| ProjectIds | No | No | Array of Int64 | List of project IDs.<br>Example: [1] |
| ViewNames | No | No | Array of String | List of alarm policy types.<br>Example: [cvm_device] |
| FilterUnuseReceiver | No | No | Int64 | Whether to filter policy groups that have no recipients. 1: yes. 0: no.<br>Example: 0 |
| Receivers | No | No | Array of String | Filters by recipient group list.<br>Example: [123] |
| ReceiverUserList | No | No | Array of String | Filters by recipient list.<br>Example: [12345] |
| Dimensions | No | No | String | Dimension set field (JSON string), for example, [[{"name":"unInstanceId","value":"ins-6e4b2aaa"}]].<br>Example: {} |
| ConditionTempGroupId | No | No | String | Template-based policy group ID. Multiple IDs are separated by commas.<br>Example: 123,345 |
| ReceiverType | No | No | String | Filters by recipient or recipient group. The value 'user' indicates by recipient. The value 'group' indicates by recipient group.<br>Example: user |

## 3. Output Parameters

| Parameter name | Type | Description |
|---|---|---|
| GroupList | Array of DescribePolicyGroupListGroup | Policy group list.<br>Example:View |

| Parameter name | Type | Description |
| --- | --- | --- |
| Total | Int64 | Total number of policy groups.<br>Example: 142 |
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

## 4. Error Code

The following lists only the error codes related to the interface business logic, For other error codes, see public ErrorCode.

| Error Code | Description |
| --- | --- |
| InvalidParameterValue | Invalid parameter value. |
| InternalError | Internal error. |
| InvalidParameter.InvalidParameterParam | Invalid parameter. |
| InvalidParameter.InvalidParameter | Invalid parameter. |
| InvalidParameter | Invalid parameter. |
| InternalError.ExeTimeout | Execution timed out. |
| FailedOperation | Operation failed. |
| MissingParameter | Missing parameter. |
| UnknownParameter | Unknown parameter error. |
| UnauthorizedOperation | Unauthorized operation. |
| UnsupportedOperation | Unsupported operation. |
| ResourceNotFound | The resource does not exist. |
| LimitExceeded | The quota limit has been reached. |
| ResourceUnavailable | The resource is unavailable. |
| ResourceInsufficient | Insufficient resource. |
| FailedOperation | Operation failed. |
| ResourceInUse | The resource is in use. |
| DryRunOperation | DryRun operation, which means the DryRun parameter is passed in yet the request will still be successful. |
| ResourcesSoldOut | The resources have been sold out. |
| LimitExceeded.MetricQuotaExceeded | Quota limit on metrics is reached. Requests containing unregistered metrics are prohibited. |
| FailedOperation.ServiceNotEnabled | The service is not enabled and can be used only after being enabled. |
| FailedOperation.AlertPolicyCreateFailed | Failed to create the alarm policy. |
| FailedOperation.DbRecordCreateFailed | Failed to create the database record. |
| FailedOperation.DbTransactionBeginFailed | Failed to start the database transaction. |
| FailedOperation.DbRecordDeleteFailed | Failed to delete the database record. |
| FailedOperation.DbQueryFailed | Failed to query the database. |
| FailedOperation.DbTransactionCommitFailed | Failed to submit the database transaction. |
| FailedOperation.DbRecordUpdateFailed | Failed to update the database record. |
| AuthFailure.UnauthorizedOperation | The request is not authorized. Please refer to the CAM documentation for the description of authentication. |
| FailedOperation.DuplicateName | The name already exists. |
| FailedOperation.DruidQueryFailed | Failed to query analysis data. |
| FailedOperation.DimQueryRequestFailed | Failed to query the service in the request dimension. |
| FailedOperation.AlertPolicyDescribeFailed | Failed to query the alarm policy. |
| FailedOperation.AlertPolicyDeleteFailed | Failed to delete the alarm policy. |
| FailedOperation.AlertPolicyModifyFailed | Failed to modify the alarm policy. |
| FailedOperation.AlertTriggerRuleDeleteFailed | Failed to delete the trigger condition. |
| FailedOperation.AlertFilterRuleDeleteFailed | Failed to delete the filter rule. |

# DescribeProductEventList

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

This API is used to get the list of product events by page.

Default interface request rate limit: 20 times per second.

Interface update time: 2020-01-02 10:12:28.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: DescribeProductEventList |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| Module | Yes | No | String | API module name. The value is set to monitor.<br>Example: monitor |
| ProductName | No | No | Array of String | Product type as a filter condition. For example, the value cvm filters Cloud Virtual Machine (CVM).<br>Example: cvm |
| EventName | No | No | Array of String | Event name as a filter condition. For example, the value guest_reboot filters a machine reboot.<br>Example: guest_reboot |
| InstanceId | No | No | Array of String | Affected object. Sample value: ins-19708ino.<br>Example: ins-19708ino |
| Dimensions | No | No | Array of DescribeProductEventListDimensions | Dimension as a filter condition. Sample value: 10.0.0.1. The sample value is a public IP address.<br>Example:View |
| RegionList | No | No | Array of String | Region as a filter condition. Sample value: gz.<br>Example: gz |
| Type | No | No | Array of String | Event type as a filter condition. Valid values: status_change and abnormal. The value status_change filters state change events. The value abnormal filters exceptional event.<br>Example: abnormal |
| Status | No | No | Array of String | Event state as a filter condition. Valid values: recover, alarm, and -. recover: filters recovered events. alarm: filters unrecovered events. -: filters events that have no states.<br>Example: recover |
| Project | No | No | Array of String | Project ID as a filter condition.<br>Example: 0 |
| IsAlarmConfig | No | No | Int64 | Specifies whether alarm states are configured as a filter condition. 1: configured. 0: not configured.<br>Example: 1 |
| TimeOrder | No | No | String | Time-based sorting order. ASC: ascending order. DESC: descending order. Default value: DESC.<br>Example: DESC |
| StartTime | No | No | Int64 | Start time. The default value is the timestamp that is 1 day ago from this day.<br>Example: 1572624000 |
| EndTime | No | No | Int64 | End time. The default value is the current timestamp.<br>Example: 1573228799 |
| Offset | No | No | Int64 | Page offset. Default value: 0.<br>Example: 0 |
| Limit | No | No | Int64 | Number of entries to return per page. Default value: 20.<br>Example: 20 |

## 3. Output Parameters

| Parameter name | Type | Description |
|---|---|---|
| Events | Array of DescribeProductEventListEvents | Event list.<br>Example:View |
| OverView | DescribeProductEventListOverView | Event statistics.<br>Example:View |
| Total | Int64 | Number of events.<br>Example: 6 |
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

## 4. Error Code

The following lists only the error codes related to the interface business logic, For other error codes, see public ErrorCode.

| Error Code | Description |
|---|---|
| InvalidParameterValue | Invalid parameter value. |
| InternalError | Internal error. |
| InvalidParameter.InvalidParameterParam | Invalid parameter. |
| InvalidParameter.InvalidParameter | Invalid parameter. |
| InvalidParameter | Invalid parameter. |
| MissingParameter | Missing parameter. |
| LimitExceeded | The quota limit has been reached. |

# DescribeProductHealthStatusList

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

API for pulling the health statistics of each service in the past 24 hours

Default interface request rate limit: 20 times per second.

Interface update time: 2022-08-31 15:04:45.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: DescribeProductHealthStatusList |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| Module | Yes | No | String | The value is fixed to "monitor".<br>Example: monitor |
| ViewName | No | No | String | Filters by view name, such as "cvm_device".<br>Example: cvm_device |
| AllRegions | No | No | Int64 | 1 means to pull all regions, and 0 is the opposite.<br>Example: 1 |
| ProjectIds | No | No | Array of Int64 | Filters by project ID.<br>Example: [1234] |

# 3. Output Parameters

| Parameter name | Type | Description |
| --- | --- | --- |
| List | Array of ProductHealthStatus | List of health status of each product.<br>Example:View |
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

# 4. Error Code

The following lists only the error codes related to the interface business logic, For other error codes, see public ErrorCode.

| Error Code | Description |
| --- | --- |
| InvalidParameterValue | Invalid parameter value. |
| InternalError | Internal error. |
| InvalidParameter.InvalidParameterParam | Invalid parameter. |
| InvalidParameter.InvalidParameter | Invalid parameter. |
| InvalidParameter | Invalid parameter. |
| InternalError.ExeTimeout | Execution timed out. |
| FailedOperation | Operation failed. |
| MissingParameter | Missing parameter. |
| UnknownParameter | Unknown parameter error. |
| UnsupportedOperation | Unsupported operation. |
| LimitExceeded | The quota limit has been reached. |
| FailedOperation | Operation failed. |

# DescribeProductList

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

API for querying the Cloud Monitor product list

Default interface request rate limit: 20 times per second.

Interface update time: 2021-01-25 19:29:02.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: DescribeProductList |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| Order | No | No | String | Sorting method: DESC/<br>ASC (case sensitive). The default value is DESC.<br>Example: DESC |
| Offset | No | No | Uint64 | Offset of the pagination query. The default value is 0.<br>Example: 0 |
| Limit | No | No | Uint64 | Number of data entries per page for pagination query. Default: 20.<br>Example: 5 |
| Module | Yes | No | String | The value is fixed to "monitor".<br>Example: monitor |

## 3. Output Parameters

| Parameter name | Type | Description |
|---|---|---|

| Parameter name | Type | Description |
| --- | --- | --- |
| ProductList | Array of ProductSimple | List of product information.<br>Example:View |
| TotalCount | Uint64 | Total number of products.<br>Example: 294 |
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

# 4. Error Code

The following lists only the error codes related to the interface business logic, For other error codes, see public ErrorCode.

| Error Code | Description |
| --- | --- |
| InvalidParameter | Invalid parameter. |
| InvalidParameterValue | Invalid parameter value. |
| UnsupportedOperation | Unsupported operation. |
| FailedOperation.AlertPolicyCreateFailed | Failed to create the alarm policy. |
| InternalError | Internal error. |
| InvalidParameter.InvalidParameter | Invalid parameter. |
| InvalidParameter.InvalidParameterParam | Invalid parameter. |
| ResourceNotFound | The resource does not exist. |
| FailedOperation | Operation failed. |
| MissingParameter | Missing parameter. |
| InternalError.ExeTimeout | Execution timed out. |

# DescribeStatisticData

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

This API is used to query monitoring data by dimension conditions.

Default interface request rate limit: 20 times per second.

Interface update time: 2021-04-21 15:13:52.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: DescribeStatisticData |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| Module | Yes | No | String | Module. The value is fixed to "monitor".<br>Example: monitor |
| Namespace | Yes | No | String | Namespace. Supported value: QCE/TKE.<br>Example: QCE/TKE2 |
| MetricNames | Yes | No | Array of String | List of metric names.<br>Example: ["K8sPodCpuCoreUsed"] |
| Conditions | No | No | Array of MidQueryCondition | Dimension condition. The = and in operators are supported.<br>Example:View |
| Period | No | No | Uint64 | Statistical period in seconds. Default value: 300.<br>Example: 300 |
| StartTime | No | No | String | Start time, which is the current time by default, such as 2020-12-08T19:51:23+08:00.<br>Example: 2020-11-24T15:15:50+08:00 |
| EndTime | No | No | String | End time, which is the current time by default, such as 2020-12-08T19:51:23+08:00.<br>Example: 2020-11-24T15:25:50+08:00 |
| GroupBys | No | No | Array of String | groupBy by the specified dimension.<br>Example: ["K8sPodCpuCoreUsed"] |

## 3. Output Parameters

| Parameter name | Type | Description |
|---|---|---|
| Period | Uint64 | Statistical period.<br>Example: 300 |
| StartTime | String | Start time.<br>Example: 2020-11-24 15:15:00 |

| Parameter name | Type | Description |
| --- | --- | --- |
| EndTime | String | End time.<br>Example: 2020-11-24 15:25:00 |
| Data | Array of MetricData | Monitoring data<br>Example:View |
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

# 4. Error Code

The following lists only the error codes related to the interface business logic, For other error codes, see public ErrorCode.

| Error Code | Description |
| --- | --- |
| AuthFailure.UnauthorizedOperation | The request is not authorized. Please refer to the CAM documentation for the description of authentication. |
| InvalidParameterValue.InvalDashboardNameExists | The dashboard name already exists. |
| FailedOperation | Operation failed. |
| FailedOperation.DbRecordUpdateFailed | Failed to update the database record. |
| InvalidParameter.InvalidParameter | Invalid parameter. |
| MissingParameter | Missing parameter. |
| FailedOperation.DataQueryFailed | Failed to query the data. |
| InvalidParameter.InvalidParameterParam | Invalid parameter. |
| UnauthorizedOperation | Unauthorized operation. |
| FailedOperation.DbTransactionCommitFailed | Failed to submit the database transaction. |
| InvalidParameter | Invalid parameter. |
| FailedOperation.DataTableNotFound | The data table doesn't exist. |
| FailedOperation.DbRecordDeleteFailed | Failed to delete the database record. |
| ResourceInsufficient | Insufficient resource. |
| FailedOperation.DbQueryFailed | Failed to query the database. |
| FailedOperation.DimQueryRequestFailed | Failed to query the service in the request dimension. |
| ResourceInUse | The resource is in use. |
| ResourceNotFound | The resource does not exist. |
| FailedOperation.DbRecordCreateFailed | Failed to create the database record. |
| InternalError | Internal error. |
| LimitExceeded | The quota limit has been reached. |
| FailedOperation.DataColumnNotFound | The data table field doesn't exist. |
| FailedOperation.DbTransactionBeginFailed | Failed to start the database transaction. |
| InvalidParameterValue | Invalid parameter value. |
| InvalidParameterValue.VersionMismatch | Inconsistent versions. |
| UnknownParameter | Unknown parameter error. |

# DescribeUnifyDashboards

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

API for obtaining a dashboard list

Default interface request rate limit: 20 times per second.

Interface update time: 2022-08-27 21:53:11.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: DescribeUnifyDashboards |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| IsStarred | No | No | String | Whether to add to "Favorite".<br>Example: false |
| Keyword | No | No | String | The keyword for search.<br>Example: dadceqe231cx |
| IsFolder | No | No | String | Whether it is a folder.<br>Example: false |
| Caller | No | No | String | Mini program: applet.<br>Example: applet |

# 3. Output Parameters

| Parameter name | Type | Description |
| --- | --- | --- |
| Dashboards | Array of UnifyDashboardMeta | Dashboard list.<br>Example:View |
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

# 4. Error Code

The following lists only the error codes related to the interface business logic, For other error codes, see public ErrorCode.

| Error Code | Description |
| --- | --- |
| UnsupportedOperation | Unsupported operation. |
| InternalError | Internal error. |
| InvalidParameter.InvalidParameter | Invalid parameter. |
| LimitExceeded | The quota limit has been reached. |
| ResourceUnavailable | The resource is unavailable. |
| UnknownParameter | Unknown parameter error. |
| LimitExceeded.MetricQuotaExceeded | Quota limit on metrics is reached. Requests containing unregistered metrics are prohibited. |
| ResourceInUse | The resource is in use. |
| MissingParameter | Missing parameter. |
| UnauthorizedOperation | Unauthorized operation. |
| InvalidParameter | Invalid parameter. |
| FailedOperation | Operation failed. |
| ResourceInsufficient | Insufficient resource. |

| Error Code | Description |
| --- | --- |
| ResourceNotFound | The resource does not exist. |

# GetMonitorData

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

Obtains the monitoring data of a cloud product. After you specify your product namespace, object dimension description, and monitoring metrics, you can obtain the corresponding monitoring data.
Highest frequency of calls that can be made per second/minute: 20/1200.
If you call excessive metrics and objects in a short period of time, the system may fail to pull data because frequency is limited. We recommend that you balance your requests by time.

Default interface request rate limit: 20 times per second.

Interface update time: 2022-11-30 19:22:03.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: GetMonitorData |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| Namespace | Yes | No | String | Namespace.<br>Example: QCE/CVM |
| MetricName | Yes | No | String | Metric name.<br>Example: CpuUsage |
| Period | No | No | Uint64 | cycle,<br>Example: 60 |
| StartTime | No | No | Datetime_iso | Start time. Sample value: 2018-09-22T19:51:23+08:00.<br>Example: 2019-03-24T10:51:23+08:00 |
| EndTime | No | No | Datetime_iso | End time. Sample value: 2018-09-22T20:51:23+08:00. By default, the system uses the current time as the parameter value. The time corresponding to the value of EndTime cannot be earlier than that of StartTime.<br>Example: 2019-03-24T20:51:23+08:00 |
| Instances | Yes | No | Array of Instance | Combination of instance object dimensions, a group of key-value pairs.<br>Example:View |

## 3. Output Parameters

| Parameter name | Type | Description |
|---|---|---|
| Period | Uint64 | Statistical period.<br>Example: 300 |
| MetricName | String | Metric name.<br>Example: cpu_usage |
| DataPoints | Array of DataPoint | Data point array.<br>Example:View |
| StartTime | Datetime | Start time.<br>Example: 2019-03-24T10:50:00+08:00 |
| EndTime | Datetime | End time.<br>Example: 2019-03-24T20:50:00+08:00 |
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

## 4. Error Code

The following lists only the error codes related to the interface business logic, For other error codes, see public ErrorCode.

| Error Code | Description |
|---|---|
| InvalidParameterValue | Invalid parameter value. |
| InternalError | Internal error. |

# ModifyAlarmNotice

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

This API is used to edit an alarm notification template.

Default interface request rate limit: 20 times per second.

Interface update time: 2021-02-24 20:11:26.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: ModifyAlarmNotice |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| Module | Yes | No | String | Module name. Enter "monitor" here.<br>Example: monitor |
| Name | Yes | No | String | Name of the alarm notification rule, which can contain up to 60 characters.<br>Example: Default notification template |
| NoticeType | Yes | No | String | Notification type. Valid values: ALARM (for unresolved alarms), OK (for resolved alarms), ALL (for all alarms)<br>Example: ALL |
| NoticeLanguage | Yes | No | String | Notification language.<br>Example: zh-CN |
| UserNotices | No | No | Array of UserNotice | Number of user notifications (up to 5).<br>Example:View |
| URLNotices | No | No | Array of URLNotice | Number of callback notifications (up to 3).<br>Example:View |
| NoticeId | Yes | No | String | ID of the alarm notification template.<br>Example: notice-qwer |
| ExtraChannels | No | No | Array of String | List of other non-public notification channels.<br>Example: ["TSS"] |

## 3. Output Parameters

| Parameter name | Type | Description |
|---|---|---|
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

## 4. Error Code

The following lists only the error codes related to the interface business logic, For other error codes, see public ErrorCode.

| Error Code | Description |
|---|---|
| InvalidParameter | Invalid parameter. |
| InvalidParameterValue | Invalid parameter value. |

| Error Code | Description |
| --- | --- |
| AuthFailure.UnauthorizedOperation | The request is not authorized. Please refer to the CAM documentation for the description of authentication. |
| InternalError | Internal error. |
| ResourceNotFound | The resource does not exist. |
| FailedOperation | Operation failed. |

# ModifyAlarmPolicyCondition

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

API for editing trigger conditions for an alarm policy

Default interface request rate limit: 20 times per second.

Interface update time: 2021-03-21 19:47:17.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: ModifyAlarmPolicyCondition |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| Module | Yes | No | String | Module name. The value is fixed to "monitor".<br>Example: monitor |
| PolicyId | Yes | No | String | Alarm policy ID.<br>Example: policy-hi498gw3h2 |
| PolicyExpression | No | No | String | Bare write alarm rules including trigger rules and filter conditions.<br>Example: expression |
| ConditionTemplateId | No | No | Int64 | Trigger condition template ID, to be transformed into a string.<br>Example: 123 |
| Condition | No | No | AlarmPolicyCondition | Metric trigger condition.<br>Example:View |
| EventCondition | No | No | AlarmPolicyEventCondition | Event trigger condition.<br>Example:View |

## 3. Output Parameters

| Parameter name | Type | Description |
|---|---|---|
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

## 4. Error Code

The following lists only the error codes related to the interface business logic, For other error codes, see public ErrorCode.

| Error Code | Description |
| --- | --- |
| InternalError | Internal error. |
| InvalidParameter | Invalid parameter. |
| ResourceNotFound | The resource does not exist. |
| FailedOperation | Operation failed. |
| UnsupportedOperation | Unsupported operation. |
| InvalidParameterValue | Invalid parameter value. |

# ModifyAlarmPolicyNotice

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

This API is used to modify the alarm notification template bound to an alarm policy.

Default interface request rate limit: 20 times per second.

Interface update time: 2025-01-21 14:21:06.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: ModifyAlarmPolicyNotice |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| Module | Yes | No | String | Module name. Enter "monitor" here.<br>Example: monitor |
| PolicyId | Yes | No | String | Alarm policy ID.<br>Example: policy-asgh3gb2 |
| NoticeIds | No | No | Array of String | List of alarm notification template IDs.<br>Example: ["notice-bv9b4ghqbg","notice-gj2n9wnt29"] |
| NoticeContentTmplBindInfos | No | No | Array of NoticeContentTmplBindInfo | Notification template ID and content template ID.<br>Example:View |

## 3. Output Parameters

| Parameter name | Type | Description |
|---|---|---|
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

## 4. Error Code

The following lists only the error codes related to the interface business logic, For other error codes, see public ErrorCode.

| Error Code | Description |
| --- | --- |
| InvalidParameterValue | Invalid parameter value. |
| InternalError | Internal error. |
| ResourceNotFound | The resource does not exist. |
| FailedOperation | Operation failed. |

# ModifyAlarmPolicyTasks

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

This API is used to modify the task triggered by an alarm policy. The TriggerTasks field contains the list of triggered tasks. If an empty array is passed in for TriggerTasks , it indicates to unbind all the triggered tasks from this policy.

Default interface request rate limit: 20 times per second.

Interface update time: 2021-03-01 11:43:16.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: ModifyAlarmPolicyTasks |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| Module | Yes | No | String | Module name. Enter "monitor" here.<br>Example: monitor |
| PolicyId | Yes | No | String | Alarm policy ID.<br>Example: policy-29ng82fs |
| TriggerTasks | No | No | Array of AlarmPolicyTriggerTask | List of tasks triggered by an alarm policy. If an empty array is passed in, it indicates to unbind all the triggered tasks from this policy.<br>Example:View |

## 3. Output Parameters

| Parameter name | Type | Description |
|---|---|---|
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

## 4. Error Code

The following lists only the error codes related to the interface business logic, For other error codes, see public ErrorCode.

| Error Code | Description |
|---|---|
| InvalidParameter | Invalid parameter. |
| InvalidParameterValue | Invalid parameter value. |
| InternalError | Internal error. |
| ResourceNotFound | The resource does not exist. |
| FailedOperation | Operation failed. |

# ModifyAlarmReceivers

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

This API is used to modify alarm recipients.

Default interface request rate limit: 20 times per second.

Interface update time: 2020-12-01 11:15:09.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: ModifyAlarmReceivers |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| GroupId | Yes | No | Int64 | ID of a policy group whose recipient needs to be modified.<br>Example: 1234 |
| ReceiverInfos | No | No | Array of ReceiverInfo | New recipient information. If this parameter is not configured, all recipients will be deleted.<br>Example:View |
| Module | Yes | No | String | Required. The value is fixed to "monitor".<br>Example: monitor |

## 3. Output Parameters

| Parameter name | Type | Description |
|---|---|---|
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

## 4. Error Code

There is no error code related to business logic for this interface, For other error codes, see public ErrorCode.

# ModifyNoticeContentTmpl

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

This API is used to modify a custom notification content template.

Default interface request rate limit: 20 times per second.

Interface update time: 2025-02-24 15:24:12.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: ModifyNoticeContentTmpl |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| TmplName | No | No | String | Template name<br>Example: tempName |
| TmplID | Yes | No | String | Template ID<br>Example: temp-xxx |
| TmplContents | No | No | NoticeContentTmplItem | Template Content<br>Example:View |
| Module | Yes | No | String | The value is fixed to "monitor".<br>Example: monitor |

# 3. Output Parameters

| Parameter name | Type | Description |
|---|---|---|
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

# 4. Error Code

There is no error code related to business logic for this interface, For other error codes, see public ErrorCode.

# PutMonitorData

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

Default frequency limit on API requests: 50 requests/second.
Default metric limit for a single tenant: 100 metrics.
A maximum of 30 metric/value pairs are reported at a time. When the request returns an error, all metrics/values in the request will not be saved.

The reported timestamp is the timestamp expected to be saved. It is recommended to create a timestamp with the number of minutes expressed in integers.
The time range of the timestamp must be between the current time and 300 seconds ago.
Data of the same IP metric pair should be reported in chronological order based on minutes.

Default interface request rate limit: 50 times per second.

Interface update time: 2019-12-13 19:17:02.

The interface only verifies the signature but does not authenticate.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: PutMonitorData |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| AnnounceIp | No | No | String | IP specified by yourself when reporting data.<br>Example: 10.0.0.1 |
| AnnounceTimestamp | No | No | Uint64 | Timestamp specified by yourself when reporting data.<br>Example: 1569379988 |
| Metrics | Yes | No | Array of MetricDatum | A set of metrics and data.<br>Example:View |
| AnnounceInstance | No | No | String | IP or product instance ID specified by yourself when reporting data .<br>Example: 10.0.0.1 |

## 3. Output Parameters

| Parameter name | Type | Description |
|---|---|---|
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when |

| Parameter name | Type | Description |
| --- | --- | --- |
| | | locating the problem. |

# 4. Error Code

The following lists only the error codes related to the interface business logic, For other error codes, see public ErrorCode.

| Error Code | Description |
| --- | --- |
| InvalidParameterValue | Invalid parameter value. |
| InternalError | Internal error. |
| InvalidParameter.InvalidParameterParam | Invalid parameter. |
| InvalidParameter.InvalidParameter | Invalid parameter. |
| InvalidParameter | Invalid parameter. |
| InternalError.ExeTimeout | Execution timed out. |
| FailedOperation | Operation failed. |
| MissingParameter | Missing parameter. |
| UnknownParameter | Unknown parameter error. |
| UnauthorizedOperation | Unauthorized operation. |
| UnsupportedOperation | Unsupported operation. |
| ResourceNotFound | The resource does not exist. |
| LimitExceeded | The quota limit has been reached. |
| ResourceUnavailable | The resource is unavailable. |
| FailedOperation | Operation failed. |
| DryRunOperation | DryRun operation, which means the DryRun parameter is passed in yet the request will still be successful. |
| LimitExceeded.MetricQuotaExceeded | Quota limit on metrics is reached. Requests containing unregistered metrics are prohibited. |
| FailedOperation.ServiceNotEnabled | The service is not enabled and can be used only after being enabled. |

# SetDefaultAlarmPolicy

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

Sets an alarm policy as the default policy in the current policy type under the current project. Alarm policies in the same type under the project will be set as non-default.

Default interface request rate limit: 20 times per second.

Interface update time: 2021-03-01 11:52:23.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: SetDefaultAlarmPolicy |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| Module | Yes | No | String | Module name. The value is fixed to "monitor". Example: monitor |
| PolicyId | Yes | No | String | Alarm policy ID. Example: policy-jfg9hg2g2 |

## 3. Output Parameters

| Parameter | Type | Description |
|---|---|---|

| | | |
|---|---|---|
| name | | |
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

# 4. Error Code

The following lists only the error codes related to the interface business logic, For other error codes, see public ErrorCode.

| Error Code | Description |
|---|---|
| InvalidParameter | Invalid parameter. |
| InvalidParameterValue | Invalid parameter value. |
| UnsupportedOperation | Unsupported operation. |
| InternalError | Internal error. |
| ResourceNotFound | The resource does not exist. |
| FailedOperation | Operation failed. |

# UnBindingAllPolicyObject

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

This API is used to delete all bound objects.

Default interface request rate limit: 20 times per second.

Interface update time: 2020-01-20 09:27:50.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: UnBindingAllPolicyObject |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| Module | Yes | No | String | The value is fixed to "monitor".<br>Example: monitor |
| GroupId | Yes | No | Int64 | Policy group ID.<br>Example: 11111 |

## 3. Output Parameters

| Parameter name | Type | Description |
|---|---|---|

| Parameter name | Type | Description |
| --- | --- | --- |
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

# 4. Error Code

The following lists only the error codes related to the interface business logic, For other error codes, see public ErrorCode.

| Error Code | Description |
| --- | --- |
| InvalidParameterValue | Invalid parameter value. |
| InternalError | Internal error. |
| InvalidParameter.InvalidParameterParam | Invalid parameter. |
| InvalidParameter.InvalidParameter | Invalid parameter. |
| InvalidParameter | Invalid parameter. |
| InternalError.ExeTimeout | Execution timed out. |
| FailedOperation | Operation failed. |
| MissingParameter | Missing parameter. |
| UnknownParameter | Unknown parameter error. |
| UnauthorizedOperation | Unauthorized operation. |
| UnsupportedOperation | Unsupported operation. |
| ResourceNotFound | The resource does not exist. |
| LimitExceeded | The quota limit has been reached. |
| ResourceUnavailable | The resource is unavailable. |
| ResourceInsufficient | Insufficient resource. |
| FailedOperation | Operation failed. |
| ResourceInUse | The resource is in use. |

| Error Code | Description |
| --- | --- |
| DryRunOperation | DryRun operation, which means the DryRun parameter is passed in yet the request will still be successful. |
| ResourcesSoldOut | The resources have been sold out. |
| LimitExceeded.MetricQuotaExceeded | Quota limit on metrics is reached. Requests containing unregistered metrics are prohibited. |
| FailedOperation.ServiceNotEnabled | The service is not enabled and can be used only after being enabled. |
| FailedOperation.AlertPolicyCreateFailed | Failed to create the alarm policy. |
| FailedOperation.DbRecordCreateFailed | Failed to create the database record. |
| FailedOperation.DbTransactionBeginFailed | Failed to start the database transaction. |
| FailedOperation.DbRecordDeleteFailed | Failed to delete the database record. |
| FailedOperation.DbQueryFailed | Failed to query the database. |
| FailedOperation.DbTransactionCommitFailed | Failed to submit the database transaction. |
| FailedOperation.DbRecordUpdateFailed | Failed to update the database record. |
| AuthFailure.UnauthorizedOperation | The request is not authorized. Please refer to the CAM documentation for the description of authentication. |
| FailedOperation.DuplicateName | The name already exists. |
| FailedOperation.DruidQueryFailed | Failed to query analysis data. |
| FailedOperation.DimQueryRequestFailed | Failed to query the service in the request dimension. |
| FailedOperation.AlertPolicyDescribeFailed | Failed to query the alarm policy. |
| FailedOperation.AlertPolicyDeleteFailed | Failed to delete the alarm policy. |
| FailedOperation.AlertPolicyModifyFailed | Failed to modify the alarm policy. |
| FailedOperation.AlertTriggerRuleDeleteFailed | Failed to delete the trigger condition. |
| FailedOperation.AlertFilterRuleDeleteFailed | Failed to delete the filter rule. |

# UnBindingPolicyObject

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

This API is used to delete an object that is bound to a policy.

Default interface request rate limit: 20 times per second.

Interface update time: 2020-03-02 00:28:36.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: UnBindingPolicyObject |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| Module | Yes | No | String | The value is fixed to "monitor".<br>Example: monitor |
| GroupId | Yes | No | Int64 | Policy group ID.<br>Example: 11111 |
| UniqueId | Yes | No | Array of String | List of unique IDs of the object instances to be deleted.<br>Example: ["3dd5113208fd467b2e5d0c1111111111\n"] |
| InstanceGroupId | No | No | Int64 | Instance group ID. The UniqueId parameter is invalid if object instances are deleted by instance group.<br>Example: 0 |

## 3. Output Parameters

| Parameter name | Type | Description |
|---|---|---|
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

## 4. Error Code

The following lists only the error codes related to the interface business logic, For other error codes, see public ErrorCode.

| Error Code | Description |
|---|---|
| InvalidParameterValue | Invalid parameter value. |
| InternalError | Internal error. |
| InvalidParameter.InvalidParameterParam | Invalid parameter. |
| InvalidParameter.InvalidParameter | Invalid parameter. |
| InvalidParameter | Invalid parameter. |
| InternalError.ExeTimeout | Execution timed out. |

| Error Code | Description |
| --- | --- |
| FailedOperation | Operation failed. |
| MissingParameter | Missing parameter. |
| UnknownParameter | Unknown parameter error. |
| UnauthorizedOperation | Unauthorized operation. |
| UnsupportedOperation | Unsupported operation. |
| ResourceNotFound | The resource does not exist. |
| LimitExceeded | The quota limit has been reached. |
| ResourceUnavailable | The resource is unavailable. |
| ResourceInsufficient | Insufficient resource. |
| FailedOperation | Operation failed. |
| ResourceInUse | The resource is in use. |
| DryRunOperation | DryRun operation, which means the DryRun parameter is passed in yet the request will still be successful. |
| ResourcesSoldOut | The resources have been sold out. |
| LimitExceeded.MetricQuotaExceeded | Quota limit on metrics is reached. Requests containing unregistered metrics are prohibited. |
| FailedOperation.ServiceNotEnabled | The service is not enabled and can be used only after being enabled. |
| FailedOperation.AlertPolicyCreateFailed | Failed to create the alarm policy. |
| FailedOperation.DbRecordCreateFailed | Failed to create the database record. |
| FailedOperation.DbTransactionBeginFailed | Failed to start the database transaction. |
| FailedOperation.DbRecordDeleteFailed | Failed to delete the database record. |
| FailedOperation.DbQueryFailed | Failed to query the database. |
| FailedOperation.DbTransactionCommitFailed | Failed to submit the database transaction. |
| FailedOperation.DbRecordUpdateFailed | Failed to update the database record. |
| AuthFailure.UnauthorizedOperation | The request is not authorized. Please refer to the CAM documentation for the description of authentication. |
| FailedOperation.DuplicateName | The name already exists. |
| FailedOperation.DruidQueryFailed | Failed to query analysis data. |
| FailedOperation.DimQueryRequestFailed | Failed to query the service in the request dimension. |
| FailedOperation.AlertPolicyDescribeFailed | Failed to query the alarm policy. |
| FailedOperation.AlertPolicyDeleteFailed | Failed to delete the alarm policy. |
| FailedOperation.AlertPolicyModifyFailed | Failed to modify the alarm policy. |
| FailedOperation.AlertTriggerRuleDeleteFailed | Failed to delete the trigger condition. |
| FailedOperation.AlertFilterRuleDeleteFailed | Failed to delete the filter rule. |

# Other APIs
# DescribeLogTopics

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

Pulls CLS log topics.

Default interface request rate limit: 20 times per second.

Interface update time: 2022-08-27 22:31:17.

The interface only verifies the signature but does not authenticate.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: DescribeLogTopics |
| Version | Yes | No | String | Public parameter, The value of this interface: 2018-07-24 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| LogSetId | Yes | No | String | Log set ID.<br>Example: d5f10b84-def8-4e58-bab0-16ef567b1f22 |

## 3. Output Parameters

| Parameter name | Type | Description |
|---|---|---|
| Data | String | Result data in JSON.<br>Example: {"topics":[{"collection":true,"create_time":"2019-10-22 13:55:09","extract_rule":{"beginning_regex":"(\\S+)\\s\\S+\\s(\\S+)\\s\\[([^\\]]+)\\]\\s\\"([^\\"]+)\\s(\\S+)\\s([^\\"]+)\\"\\s(\\d+)\\s(\\d+)\\s\\"([^\\"]+)\\"\\s\\"([^\\"]+)\\"\\s\\"([^\\"]+)\\"$","filter_keys":[],"filter_regex":[],"keys":["remote_addr","remote_user","time_local","http_method","http_uri","http_protocol","http_status","body_bytes_sent","http_referer","http_user_agent","http_x_forwarded_for"],"log_regex":"(\\S+)\\s\\S+\\s(\\S+)\\s\\[([^\\]]+)\\]\\s\\"([^\\"]+)\\s(\\S+)\\s([^\\"]+)\\"\\s(\\d+)\\s(\\d+)\\s\\"([^\\"]+)\\"\\s\\"([^\\"]+)\\"\\s\\"([^\\"]+)\\"\$"},"index":true,"log_type":"fullregex_log","logset_id":"d5f10b84-def8-4e58-bab0-16ef567b1f33","partition_count":1,"path":"","shipper":false,"sql_flag":false,"topic_id":"d4253323-7dfc-4b1b-8889-d82cef77375b","topic_name":"cms-nginx-regex","wild_path":"/data/log/nginx/**/default.log"}]} |
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

## 4. Error Code

The following lists only the error codes related to the interface business logic, For other error codes, see public ErrorCode.

| Error Code | Description |
|---|---|
| InternalError | Internal error. |
| InvalidParameter | Invalid parameter. |
| FailedOperation | Operation failed. |
| MissingParameter | Missing parameter. |
| UnauthorizedOperation | Unauthorized operation. |
| FailedOperation | Operation failed. |
| UnsupportedOperation | Unsupported operation. |

## Data structure

### DescribeEventPolicyConfig

DescribeEventPolicyConfig

Referenced by the following interfaces: DescribeEventPolicyConfig

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| EventID | yes | no | String | Event ID.<br>Example: 1234 |
| ProductName | yes | no | String | Product name.<br>Example: cvm |
| ProductEName | yes | no | String | The product name.<br>Example: cvm |
| ProductCName | yes | no | String | The name of the product.<br>Example: CVM instance |
| EventName | yes | no | String | Event name.<br>Example: ReadOnly |
| EventEName | yes | no | String | English name of an event.<br>Example: ReadOnly |
| EventCName | yes | no | String | The name of an event.<br>Example: Disk read-only |
| Dimensions | yes | yes | Array of DescribeEventPolicyConfigDimension | Dimension.<br>Example:View |

### Instance

Array of instance dimension combinations

Referenced by the following interfaces: DescribePolicyGroupCount, GetMonitorData

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Dimensions | yes | no | Array of Dimension | Dimension combination of an instance.<br>Example:View |

### GroupTemplateList

GetRecommendedTemplateResponse.GroupTemplateList

Referenced by the following interfaces: DescribeRecommendedTemplate

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| GroupID | yes | no | Int64 | Template policy group ID.<br>Example: 1234 |
| GroupName | yes | no | String | Template-based policy group name.<br>Example: Testing |
| IsDefault | yes | no | Int64 | Whether it is a default policy.<br>Example: 1 |
| Enable | yes | no | Bool | Enablement status of the alarm policy.<br>Example: true |
| LastEditUin | yes | no | Int64 | UIN of the last user who modified it.<br>Example: 123456 |
| NoShieldedInstanceCount | yes | no | Int64 | Number of unshielded instances.<br>Example: 10 |
| TotalInstanceCount | yes | no | Int64 | Total number of bound instances.<br>Example: 10 |
| ParentGroupID | yes | no | Int64 | Parent policy group ID.<br>Example: 1234 |
| Remark | yes | no | String | Remarks.<br>Example: Marker |
| ViewName | yes | no | String | View.<br>Example: cvm_device |
| Conditions | yes | no | Array of Condition | Metric alarm trigger condition.<br>Example:View |
| EventConditions | yes | no | Array of EventCondition | Event alarm trigger condition.<br>Example:View |
| CanSetDefault | yes | no | Bool | Whether the policy can be set as the default policy.<br>Example: true |
| InsertTime | yes | no | Int64 | Creation time.<br>Example: 1735129241 |
| UpdateTime | yes | no | Int64 | Modification time.<br>Example: 1735129241 |

### DescribeBindingPolicyObjectListDimension

Dimensions of the DescribeBindingPolicyObjectList API

Referenced by the following interfaces: DescribeBindingPolicyObjectList

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| RegionId | yes | no | Int64 | Region ID.<br>Example: 1 |
| Region | yes | no | String | Region name abbreviation.<br>Example: gz |
| Dimensions | yes | no | String | Combined JSON string of dimensions.<br>Example: {"unInstanceId":"ins-xxx"} |
| EventDimensions | yes | no | String | Combined JSON string of event dimensions.<br>Example: {"unInstanceId":"ins-xxx"} |

### MetricSet

Description of the unit and supported statistical period of the business metric

Referenced by the following interfaces: DescribeBaseMetrics

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Namespace | yes | no | String | Namespace. Each cloud product has a namespace.<br>Example: QCE/CVM |
| MetricName | yes | no | String | Metric name.<br>Example: AccOuttraffic |
| Unit | yes | no | String | Unit used by the metric.<br>Example: MB |
| UnitCname | yes | no | String | Unit used by the metric.<br>Example: "" |
| Period | yes | no | Array of Int64 | Statistical period in seconds supported by the metric, such as 60 and 300.<br>Example: [10,60] |
| Periods | yes | no | Array of PeriodsSt | Metric in a statistical period.<br>Example:View |
| Meaning | yes | no | MetricObjectMeaning | The meaning of the statistical metric.<br>Example:View |
| Dimensions | yes | no | Array of DimensionsDesc | The description of the dimension.<br>Example:View |

### AttributeServerInfoOutput

AttributeServerInfoOutput

Referenced by the following interfaces: DescribeAttributeAllServer, DescribeIdcServer

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| ServerId | yes | yes | Uint64 | Server ID.<br>Example: 1 |
| ServerName | yes | yes | String | Server name.<br>Example: test |
| Ip | yes | yes | String | Server IP address.<br>Example: 127.0.0.1 |
| IdcId | yes | yes | Int64 | Region ID.<br>Example: 1 |
| IdcName | yes | yes | String | The region name.<br>Example: test |

### DescribeAccidentConfigAccident

Platform event of the DescribeAccidentConfig API.

Referenced by the following interfaces: DescribeAccidentConfig

| name | required | NULL allowed | type | description |
|---|---|---|---|---|

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| BusinessId | yes | no | Int64 | Business ID.<br>Example: 10 |
| BusinessName | yes | no | String | Business name.<br>Example: qwer |
| BusinessEnName | yes | no | String | Business name.<br>Example: qwer |
| PolicyShowName | yes | no | String | Displayed strategy name.<br>Example: qwer |
| PolicyShowEnName | yes | no | String | Displayed strategy name in English.<br>Example: qwer |
| PolicyViewName | yes | no | String | Policy type name.<br>Example: cvm_device |
| AccidentTypeInfo | yes | no | Array of DescribeAccidentConfigAccidentTypeInfo | List of platform event types.<br>Example:View |

## PCLMMetricSet

Log monitoring metric set.

Referenced by the following interfaces: CLMDescribeMetricSets

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| LogsetId | yes | yes | String | CLS logset ID.<br>Example: d5f10b84-def8-4e58-bab0-16ef567b1f22 |
| LogsetName | yes | yes | String | CLS log set name.<br>Example: Testing |
| LogtopicId | yes | yes | String | CLS log topic ID.<br>Example: d4253323-7dfc-4b1b-8889-d82cef77375a |
| LogtopicName | yes | yes | String | CLS log topic name.<br>Example: Testing |
| Name | yes | yes | String | Metric set name.<br>Example: Testing |
| StatisticCycle | yes | yes | Int64 | Statistical period. Valid values: 60, 300.<br>Example: 1 |
| TimeField | yes | yes | String | Time field.<br>Example: 1314141 |
| LogFilterRules | yes | yes | Array of PCLMLogFilterRule | Log filtering rule.<br>Example:View |
| MetricBasicItems | yes | yes | Array of PCLMMetricBasicItem | Basic metric rule.<br>Example:View |
| MetricCustomItems | yes | yes | Array of PCLMMetricCustomItem | Compound metric rule.<br>Example:View |
| Dimensions | yes | yes | Array of PCLMDimension | Dimension rule.<br>Example:View |
| CID | no | yes | String | CID (only for output parameter; leave it empty for input parameter).<br>Example: cid-xxx |
| CreatedAt | no | yes | String | Creation time (only for output parameter; leave it empty for input parameter)<br>Example: qwer |
| UpdatedAt | no | yes | String | Update time (only for output parameter; leave it empty for input parameter)<br>Example: qwer |
| AppId | no | yes | Int64 | App ID (only for output parameter; leave it empty for input parameter).<br>Example: 1234 |
| Uin | no | yes | String | UIN (only for output parameter; leave it empty for input parameter).<br>Example: 123456 |
| Region | no | yes | String | Region (only for output parameter; leave it empty for input parameter).<br>Example: gz |
| LogProfileItems | yes | yes | Array of PCLMLogProfileItem | Log data structure description.<br>Example:View |
| SubAccountUin | no | yes | String | SubAccountUin (only for output parameter; leave it empty for input parameter).<br>Example: 123456 |

## PCLMMetricAnalysisPoint

CLM metric analysis data point.

Referenced by the following interfaces: DescribeMetricAnalysisData

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Timestamp | yes | no | Int64 | Timestamp (seconds).<br>Example: 1735129241 |
| Value | yes | yes | Float | Value.<br>Example: 10 |
| Name | yes | no | String | Metric name.<br>Example: Testing |

## PCLMMetricBasicItem

Basic metric configurations for the log monitoring metric set.

Referenced by the following interfaces: CLMDescribeMetricSets

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Field | yes | yes | String | Field name.<br>Example: timeField |
| Operator | yes | yes | String | Metric statistics method: sum, count, min, max.<br>Example: = |
| Name | yes | yes | String | Metric name.<br>Example: Testing |
| Desc | yes | yes | String | Metric description name.<br>Example: Testing |
| StatisticFilterRules | no | yes | Array of PCLMStatisticFilterRule | Metric filtering rule.<br>Example:View |
| MetricSetCID | no | yes | String | Metric set CID (only for output parameter; leave it empty for input parameter).<br>Example: cid-xxx |
| CreatedAt | no | yes | String | Creation time (only for output parameter; leave it empty for input parameter)<br>Example: qwer |
| UpdatedAt | no | yes | String | Update time (only for output parameter; leave it empty for input parameter)<br>Example: qwer |

## DescribePolicyConditionListConfigManualContinueTime

DescribePolicyConditionList.ConfigManual.ContinueTime

Referenced by the following interfaces: DescribePolicyConditionList

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Default | yes | yes | Int64 | Default duration in seconds.<br>Example: 10 |
| Keys | yes | yes | Array of Int64 | Custom duration in seconds.<br>Example: 10 |
| Need | yes | no | Bool | Whether required.<br>Example: true |

## DimensionsDesc

Dimension information

Referenced by the following interfaces: DescribeBaseMetrics

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Dimensions | yes | no | Array of String | Array of dimension names.<br>Example: ["vm_uuid"] |

## Metric

Metric, which can be used to set alarms and query data

Referenced by the following interfaces: DescribeAlarmMetrics

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Namespace | yes | no | String | Alarm policy type.<br>Example: cvm_device |
| QceNamespace | yes | no | String | It is only valid for cloud product monitoring and is used by certain APIs to query data.<br>Example: QCE/CVM |
| MetricName | yes | no | String | Metric name.<br>Example: CpuUsage |

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Description | yes | no | String | Displayed metric name.<br>Example: CPU Utilization |
| Aggregations | yes | no | Array of MetricAggregation | The aggregation method.<br>Example:View |
| Min | yes | no | Float | Minimum value.<br>Example: 0 |
| Max | yes | no | Float | Maximum value.<br>Example: 100 |
| Dimensions | yes | no | Array of String | The dimensions.<br>Example: vm_uuid |
| Unit | yes | no | String | Unit.<br>Example: % |
| Tooltip | yes | yes | MetricTooltip | Rich text prompt.<br>Example:View |
| MetricConfig | yes | yes | MetricConfig | Metric configuration<br>Example:View |
| Operators | no | no | Array of String | Operators<br>Example: {"Id":"eq","Name":"="} |
| Periods | no | no | Array of Int64 | Periods<br>Example: 5 |

## CCMInstanceAttributeDataOutput

CCMInstanceAttributeDataOutput

Referenced by the following interfaces: DescribeCCMInstanceDatas

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| AttributeId | yes | no | Uint64 | Metric ID.<br>Example: 1 |
| ServerId | yes | no | Uint64 | Server ID.<br>Example: 1 |
| Values | yes | no | Array of AttributeTimestampValueOutput | Metric report data.<br>Example:View |
| InstanceName | yes | no | String | Instance name.<br>Example: ins-qwer |

## CgrpInstanceNode

Group instance details.

Referenced by the following interfaces: CreateInstances, DeleteInstances, DescribeInstance

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Id | no | yes | String | Auto increment ID.<br>Example: 1234 |
| InstanceId | yes | no | String | Instance ID.<br>Example: ins-00jvv9mo |
| InstanceName | yes | no | String | Instance name.<br>Example: test |
| TypeId | yes | no | String | The instance type.<br>Example: cvm_device |
| Region | yes | no | String | Region.<br>Example: gz |
| ModuleId | yes | no | Int64 | Leaf group ID.<br>Example: 1 |
| ModuleName | yes | no | String | Leaf group name.<br>Example: qwer |
| L1Id | no | yes | Int64 | Level-1 group ID.<br>Example: 1 |
| L2Id | no | yes | Int64 | Level-2 group ID.<br>Example: 1 |
| L3Id | no | yes | Int64 | Level-3 group ID.<br>Example: 1 |
| AppId | no | yes | Int64 | appId<br>Example: 1234 |
| Uin | no | yes | String | Uin<br>Example: 123456 |
| SubUin | no | yes | String | SubUin<br>Example: 123456 |
| Group | no | yes | Array of CgrpInstanceGroupNode | Group array.<br>Example:View |

## DescribeDashboardsList

DescribeDashboardsList

Referenced by the following interfaces: DescribeDashboards

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| DashboardID | yes | no | String | DashboardID<br>Example: eipv6 |
| DescName | yes | no | String | Description name.<br>Example: test |
| Meta | yes | no | String | Metadata.<br>Example: {"aggregateType":"detail","aggregations":["Avg","Max","Min"],"chartTypes":["column"],"configId":"cvm","layout":{"h":5,"h2":0,"w":4,"x":0,"y":0},"timeAggregate":"last"} |

## PolicyGroupReceiverInfo

Information on recipients in the policy template list (API v2018)

Referenced by the following interfaces: DescribeConditionsTemplateList

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| EndTime | yes | no | Int64 | End time of a valid time period.<br>Example: 1734957911 |
| NeedSendNotice | yes | no | Int64 | Whether it is required to send notifications.<br>Example: 1 |
| NotifyWay | yes | yes | Array of String | The channel for sending alarms.<br>Example: sms |
| PersonInterval | yes | no | Int64 | Alarm call intervals for individuals in seconds.<br>Example: 1 |
| ReceiverGroupList | yes | yes | Array of Int64 | Message recipient group list.<br>Example: 10 |
| ReceiverType | yes | no | String | Recipient type.<br>Example: sms |
| ReceiverUserList | yes | yes | Array of Int64 | Recipient list. The list of recipient IDs that is queried by API.<br>Example: [1234] |
| RecoverNotify | yes | yes | Array of String | Notification method when an alarm is recovered.<br>Example: sms |
| RoundInterval | yes | no | Int64 | Intervals of alarm call rounds in seconds.<br>Example: 1 |
| RoundNumber | yes | no | Int64 | Number of alarm call rounds.<br>Example: 1 |
| SendFor | yes | yes | Array of String | Alarm call notification time. Valid values: OCCUR (indicating that a notice is sent when the alarm is triggered) and RECOVER (indicating that a notice is sent when the alarm is recovered).<br>Example: qwer |
| StartTime | yes | no | Int64 | Start time of a valid time period.<br>Example: 1734953808 |
| UIDList | yes | yes | Array of Int64 | UID of the alarm call recipient.<br>Example: 1234 |

## AlarmPolicy

Alarm policy details

Referenced by the following interfaces: DescribeAlarmPolicies, DescribeAlarmPolicy

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| PolicyId | yes | yes | String | Alarm policy ID.<br>Example: policy-abc123 |
| PolicyName | yes | yes | String | The name of the alarm policy.<br>Example: Testing |
| Remark | yes | yes | String | Remarks.<br>Example: Testing |
| MonitorType | yes | yes | String | Monitor type. Valid values: MT_QCE (Cloud Platform service monitoring)<br>Example: MT_QCE |

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Enable | yes | yes | Int64 | Enabled or disabled status. 1: enabled. 0: disabled.<br>Example: 1 |
| UseSum | yes | yes | Int64 | Number of instances bound to the policy group.<br>Example: 10 |
| ProjectId | yes | yes | Int64 | Project ID. -1: no project. 0: default project.<br>Example: 1 |
| ProjectName | yes | yes | String | Project name.<br>Example: Testing |
| Namespace | yes | yes | String | Alarm policy type.<br>Example: cvm_device |
| Measurement | yes | yes | String | That is, viewName / View. Cloud Platform service monitoring: cvm_device, etc. Custom monitoring: Measurement under Namespace.<br>Example: test |
| PolicyExpression | yes | yes | String | Writes alarm rules including trigger rules, filter conditions, etc. It is suitable for prometheus monitoring.<br>Example: Testing |
| ConditionTemplateId | yes | yes | String | Trigger condition template ID.<br>Example: 1234 |
| Condition | yes | yes | AlarmPolicyCondition | Metric trigger condition.<br>Example:View |
| EventCondition | yes | yes | AlarmPolicyEventCondition | Event trigger condition.<br>Example:View |
| NoticeIds | yes | yes | Array of String | List of notification rule IDs.<br>Example: ["1234"] |
| Notices | yes | yes | Array of AlarmNotice | Notification rule list.<br>Example:View |
| TriggerTasks | yes | yes | Array of AlarmPolicyTriggerTask | List of trigger tasks.<br>Example:View |
| OriginID | no | yes | String | Obsolete field. Policy ID for APIs for binding and unbinding instances or instance groups (BindingPolicyObject, UnBindingAllPolicyObject, UnBindingPolicyObject).<br>Example: 1234 |
| ConditionsTemp | yes | yes | ConditionsTemp | Template-based policy group.<br>Note: This field may return null, indicating that no valid value was found.<br>Example:View |
| LastEditUin | yes | yes | String | UIN of the last user who modified it.<br>Example: 5678 |
| UpdateTime | yes | yes | Int64 | Update time.<br>Note: This field may return null, indicating that no valid value was found.<br>Example: 1735022702 |
| InsertTime | yes | yes | Int64 | Creation time.<br>Note: This field may return null, indicating that no valid value was found.<br>Example: 1735002702 |
| Region | yes | yes | Array of String | Region.<br>Example: gz |
| NamespaceShowName | yes | no | String | Namespace display name.<br>Example: cvm_device |
| IsDefault | yes | no | Int64 | Whether it is the default policy. Valid values: 1 (yes), 0 (no).<br>Example: 1 |
| CanSetDefault | yes | no | Int64 | Whether the default policy can be set. Valid values: 1 (yes), 0 (no).<br>Example: 1 |
| InstanceGroupId | yes | no | Int64 | Instance group ID.<br>Example: 1234 |
| InstanceSum | yes | no | Int64 | Total number of instances in instance group.<br>Example: 10 |
| InstanceGroupName | yes | no | String | Instance group name.<br>Example: test |
| JsConfig | yes | no | String | JsConfig foreground configuration.<br>Example: {"cluster_name":"-es-dnxssss"} |
| RuleType | yes | no | String | Trigger condition type. Valid values: STATIC (static threshold), DYNAMIC (dynamic).<br>Example: STATIC |
| OriginId | yes | no | String | Policy ID for APIs for binding and unbinding instances or instance groups (BindingPolicyObject, UnBindingAllPolicyObject, UnBindingPolicyObject).<br>Example: 1234 |
| TagInstances | yes | yes | Array of TagInstance | Tag.<br>Example:View |
| Filter | no | yes | String | Filter<br>Example: {"Dimensions":"","Expression":"","Type":""} |
| GroupBy | no | yes | String | GroupBy<br>Example: {"Id":"","Name":""} |
| NoticeContentTmplBindInfos | no | no | NoticeContentTmplBindInfo | Information on the notification template and content template.<br>Example:View |
| AlarmLevel | no | no | String | Alarm Level<br>Example: Remind |

## CreatePolicyGroupEventCondition

Event alarm condition passed in when a policy is created.

Referenced by the following interfaces: CreatePolicyGroup

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| RuleId | no | no | Int64 | If a metric is created based on a template, the RuleId of the metric in the template must be passed in.<br>Example: 1 |
| EventId | yes | no | Int64 | Alarm event ID.<br>Example: 1 |
| AlarmNotifyType | yes | no | Int64 | Alarm sending and converging type. The value 0 indicates that alarms are sent consecutively. The value 1 indicates that alarms are sent exponentially.<br>Example: 1 |
| AlarmNotifyPeriod | yes | no | Int64 | Alarm sending period in seconds. The value u003c0 indicates that no alarm will be triggered. The value 0 indicates that an alarm is triggered only once. The value >0 indicates that an alarm is triggered at the interval of triggerTime.<br>Example: 5 |

## DashboardMetricEntry

Dashboard2.0 metric information.

Referenced by the following interfaces: DescribeDashboardMetrics

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Namespace | yes | no | String | Namespace value.<br>Example: QCE/CVM |
| MetricName | yes | no | String | Metric name.<br>Example: WanIntraffic |
| MetricType | yes | no | String | The metric type.<br>Example: QCEMetric |
| Aggregations | yes | no | Array of String | Aggregation method of the metric.<br>Example: Max |
| Description | yes | no | String | Displayed metric name.<br>Example: Public Network Inbound Bandwidth |
| Unit | yes | no | String | Metric data unit.<br>Example: Mbps |
| Min | yes | no | Float | Minimum value.<br>Example: 0 |
| Max | yes | no | Float | Maximum value.<br>Example: 0 |
| Dimensions | yes | no | Array of String | The dimensions.<br>Example: {"unInstanceId":"ins-xxx"} |

## DescribePolicyConditionListConfigManualStatType

DescribePolicyConditionList.ConfigManual.StatType

Referenced by the following interfaces: DescribePolicyConditionList

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| P5 | yes | yes | String | Data aggregation method in a period of 5 seconds.<br>Example: 5 |
| P10 | yes | yes | String | Data aggregation method in a period of 10 seconds.<br>Example: 10 |
| P60 | yes | yes | String | Data aggregation method in a period of 1 minute.<br>Example: 60 |
| P300 | yes | yes | String | Data aggregation method in a period of 5 minute.<br>Example: 300 |
| P600 | yes | yes | String | Data aggregation method in a period of 10 minute.<br>Example: 600 |
| P1800 | yes | yes | String | Data aggregation method in a period of 30 minute.<br>Example: 1800 |
| P3600 | yes | yes | String | Data aggregation method in a period of 1 hour.<br>Example: 3600 |
| P86400 | yes | yes | String | Data aggregation method in a period of 1 day.<br>Example: 86400 |

## DescribePolicyQuotaV3ResponseDataQuota

DescribePolicyQuotaV3ResponseDataQuota

Referenced by the following interfaces: DescribePolicyQuota

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Used | yes | no | Int64 | User.<br>Example: 10 |
| Total | yes | no | Int64 | Total number.<br>Example: 10 |

## SubscribeInfoReceiver

Output parameter of DescribeSubscribeInfo.

Referenced by the following interfaces: DescribeSubscribeInfo

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Username | yes | yes | String | User name.<br>Example: test |
| Uin | yes | yes | String | User UIN.<br>Example: 123456 |
| Uid | yes | yes | String | User UID.<br>Example: 1234 |

## AttributeServerInfoOutputData

AttributeServerInfoOutputData

Referenced by the following interfaces: DescribeAttributeAllServer, DescribeIdcServer

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Data | yes | no | Array of AttributeServerInfoOutput | Returns a list of servers reporting this attribute.<br>Example:View |
| TotalCount | yes | no | Int64 | Returns the number of server lists reporting this attribute.<br>Example: 10 |

## CCMChartData

CCMChartData

Referenced by the following interfaces: DescribeCCMCharts

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Data | yes | no | Array of CCMChartEntry | Returns the data list.<br>Example:View |
| TotalCount | yes | no | Int64 | Queries the total amount of data.<br>Example: 10 |

## MetricAggregation

Metric aggregation method.

Referenced by the following interfaces: DescribeAlarmMetrics

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| NameCn | yes | no | String | Name.<br>Example: test |
| NameEn | yes | no | String | The name.<br>Example: Testing |
| Id | no | yes | String | The ID.<br>Example: 1 |
| Name | no | yes | String | Name<br>Example: Testing |

## CustomCallbackTmplMatcher

Matcher for custom notification content templates.

Referenced by the following interfaces: CreateNoticeContentTmpl, DescribeNoticeContentTmpl, ModifyNoticeContentTmpl

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| MatchingStatus | yes | no | Array of String | Matching status. Valid values: Invalid, Trigger, and Recovery.<br>Example: Trigger |
| Template | yes | no | CustomCallbackNoticeTmpl | Template configuration<br>Example:View |

## CgrpModuleNode

Group leaf node data structure.

Referenced by the following interfaces: CreateModule, DeleteModule, DescribeModule, ModifyModule

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| AppId | no | yes | Int64 | AppId<br>Example: 1234 |
| High | no | yes | Int64 | Grouping tree height.<br>Example: 1 |
| Id | yes | no | Int64 | Leaf group ID.<br>Example: 1 |
| L1Id | no | yes | Int64 | Level-1 group ID.<br>Example: 1 |
| L1Name | no | yes | String | Level-1 group name.<br>Example: Testing |
| L2Id | no | yes | Int64 | Level-2 group ID.<br>Example: 1 |
| L2Name | no | yes | String | Level-2 group name.<br>Example: Testing |
| L3Id | no | yes | Int64 | Level-3 group ID.<br>Example: 1 |
| L3Name | no | yes | String | Level-3 group name.<br>Example: Testing |
| Name | no | yes | String | Leaf group name.<br>Example: Testing |
| OwnerBack | no | yes | String | Backup owner.<br>Example: qwer123 |
| OwnerMain | no | yes | String | Major owner.<br>Example: qwer123 |
| ParentId | yes | no | Int64 | Parent group ID.<br>Example: 1 |
| SubUin | no | yes | String | SubUin<br>Example: 123456 |
| TypeId | yes | no | String | Type of the instance that is allowed to be added to the leaf group.<br>Example: cvm_device |
| Uin | no | yes | String | Uin<br>Example: 123456 |

## AlarmPolicyRule

Alarm policy trigger condition (shared by metrics and events and example values are separated by /).

Referenced by the following interfaces: CreateAlarmPolicy, DescribeAlarmPolicies, DescribeAlarmPolicy, ModifyAlarmPolicyCondition

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| MetricName | no | no | String | Metric name.<br>Example: CpuUsage/disk_readonly |
| AggregationDimensions | no | yes | Array of String | Aggregation dimension list (not passed for events).<br>Example: ["sum"] |
| Period | no | no | Int64 | Seconds. Statistical period (not passed for events).<br>Example: 60 |
| AggregateType | no | no | String | Data aggregation type. Multiple metrics may be involved due to compound conditions. Metrics are indicated in JSON format, such as {"report_metric_gauge":"last"} (not passed for events).<br>Example: last |
| Operator | no | no | String | English operators<br>intelligent = No intelligent detection for the threshold<br>eq = Equal to<br>ge = Greater than or equal to<br>gt = Greater than<br>le = Less than or equal to<br>lt = Less than<br>ne = Not equal to<br>day_increase = Day-on-day growth<br>day_decrease = Day-on-day decline<br>day_wave = Day-on-day fluctuation |

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| | | | | week_increase = Week-on-week growth<br>week_decrease = Week-on-week decline<br>week_wave = Week-on-week fluctuation<br>cycle_increase = Month-on-month growth<br>cycle_decrease = Month-on-month decline<br>cycle_wave = Month-on-month fluctuation<br>re = Regular expression matching (not passed for events)<br>Example: gt |
| Value | no | no | String | Threshold (not passed for events).<br>Example: 100 |
| ContinuePeriod | no | no | Int64 | Number of cycles. Continuous notification cycle. 1 = last for 1 cycle; 2 = last for 2 cycles... (not passed for events).<br>Example: 1 |
| NoticeFrequency | no | no | Int64 | Seconds. Alarm interval: 0 = non-repeating; 300 = every 5 minutes; 600 = every 10 minutes; 900 = every 15 minutes; 1800 = every 30 minutes; 3600 = every 1 hour; 7200 = every 2 hours; 10800 = every 3 hours; 21600 = every 6 hours; 43200 = every 12 hours; 86400 = every day (not passed for events).<br>Example: 300 |
| IsPowerNotice | no | no | Int64 | Whether the alarm frequency increases exponentially. 0 = no; 1 = yes (not passed for events).<br>Example: 0 |
| Filter | no | yes | AlarmPolicyFilter | Filter condition for a single trigger rule (not passed for events).<br>Example:View |
| Description | no | no | String | Metric display name, which is used in the output parameter.<br>Example: CPU utilization/read-only disks |
| Unit | no | no | String | Unit, which is used for output parameters (not passed for events).<br>Example: % |
| RuleType | no | no | String | Trigger condition type. STATIC = static threshold; DYNAMIC = dynamic threshold (not passed for events).<br>Example: STATIC |
| ValueMax | no | yes | String | ValueMax (not passed for events).<br>Example: 0 |
| ValueMin | no | yes | String | ValueMin (not passed for events).<br>Example: 0 |

## ClmStatisticFilterRule

Metric filter condition in log monitoring metric set configurations.

Referenced by the following interfaces: CreateMetricSet, DescribeMetricSet, DescribeMetricSets, ModifyMetricSet

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Id | no | no | Int64 | Item ID (only for output parameter; leave it empty for input parameter).<br>Example: 1 |
| MetricItemId | no | no | Int64 | Associated metric item ID (only for output parameter; leave it empty for input parameter).<br>Example: 1 |
| CreatedAt | no | no | String | Creation time (only for output parameter; leave it empty for input parameter).<br>Example: eqeadfrqc |
| UpdatedAt | no | no | String | Update time (only for output parameter; leave it empty for input parameter).<br>Example: eqeadfrqc |
| Relation | yes | no | Int64 | Relationship. 1: AND. 2: OR.<br>Example: 12 |
| Field | yes | no | String | Field name.<br>Example: timeField |
| Operator | yes | no | String | Filter operator: Numeric fields supoort: gt lt ge le eq ne in. Character fields support: eq ne in contains.<br>Example: = |
| Value | no | no | String | Filter operation value.<br>Example: 50 |

## DescribePolicyConditionListMetric

Metric alarm configuration.

Referenced by the following interfaces: DescribePolicyConditionList

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| ConfigManual | yes | yes | DescribePolicyConditionListConfigManual | Metric configuration<br>Example:View |
| MetricId | yes | no | Int64 | Metric ID.<br>Example: 10 |
| MetricShowName | yes | no | String | Metric name.<br>Example: cvm_device |
| MetricUnit | yes | no | String | Metric unit.<br>Example: % |

## ClmLogFilterRule

Metric set filtering rule for log monitoring.

Referenced by the following interfaces: CreateMetricSet, DescribeMetricSet, DescribeMetricSets, ModifyMetricSet

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Relation | yes | no | Int64 | Relationship. 1: AND. 2: OR.<br>Example: 1 |
| Field | yes | no | String | Field name.<br>Example: timeField |
| Operator | yes | no | String | Filter operator: Numeric fields supoort: gt lt ge le eq ne in. Character fields support: eq ne in contains.<br>Example: = |
| Value | yes | no | String | Filter operation value.<br>Example: 50 |
| Id | no | no | Int64 | Item ID (only for output parameter; leave it empty for input parameter)<br>Example: 1 |
| MetricSetId | no | no | Int64 | Metric set ID (only for output parameter; leave it empty for input parameter)<br>Example: 12 |
| CreatedAt | no | no | String | Creation time (only for output parameter; leave it empty for input parameter)<br>Example: eqeadfrqc |
| UpdatedAt | no | no | String | Update time (only for output parameter; leave it empty for input parameter)<br>Example: eqeadfrqc |

## DescribeAgentStatusHistory

DescribeAgentStatusHistory

Referenced by the following interfaces: DescribeAgentStatusHistory

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Timestamp | yes | no | Datetime_iso | Timestamp.<br>Example: 2019-11-29T12:30:13+08:00 |
| Status | yes | no | Int64 | State.<br>Example: 1 |

## CustomCallbackNoticeTmpl

Elements in the custom notification content template.

Referenced by the following interfaces: CreateNoticeContentTmpl, DescribeNoticeContentTmpl, ModifyNoticeContentTmpl

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| ContentTmpl | yes | no | String | Alarm notification content template.<br>Example: Alarm notification template |
| TitleTmpl | no | no | String | Alarm notification topic template.<br>Example: Title |

## DescribeBindingPolicyObjectListInstance

Object instance information returned by the DescribeBindingPolicyObjectListInstance API.

Referenced by the following interfaces: DescribeBindingPolicyObjectList

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| UniqueId | yes | no | String | Unique ID of the object.<br>Example: eqeq-czda |
| Dimensions | yes | no | String | Dimension set of an object instance, which is a jsonObj string.<br>Example: {"unInstanceId":"ins-xxx"} |
| IsShielded | yes | no | Int64 | Whether the object is shielded. The value 0 indicates that the object is not shielded. The value 1 indicates that the object is shielded.<br>Example: 1 |
| Region | yes | no | String | Region where the object resides.<br>Example: gz |

## DescribePolicyInfoByInstanceConditionsTemp

DescribePolicyInfoByInstanceConditionsTemp

Referenced by the following interfaces: DescribePolicyInfoByInstance

| name | required | NULL allowed | type | description |
|---|---|---|---|---|

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| GroupID | yes | yes | Int64 | Group ID.<br>Example: 1234 |
| GroupName | yes | no | String | Group name.<br>Example: Testing |
| ViewName | yes | no | String | View name.<br>Example: cvm_device |
| Remark | yes | no | String | Description.<br>Example: Testing |
| LastEditUin | yes | yes | Int64 | UIN of the last editor.<br>Example: 123456 |
| IsUnionRule | yes | yes | Int64 | Whether the AND rule is used.<br>Example: 1 |
| InsertTime | yes | yes | Int64 | Creation time.<br>Example: 1735095122 |
| UpdateTime | yes | yes | Int64 | Update time.<br>Example: 1735095122 |

## PCLMLogFilterRule

Metric set filtering rule for log monitoring.

Referenced by the following interfaces: CLMDescribeMetricSets

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Relation | yes | no | Int64 | Relationship. 1: AND. 2: OR.<br>Example: 10 |
| Field | yes | no | String | Field name.<br>Example: timeField |
| Operator | yes | no | String | Filter operator: Numeric fields supoort: gt lt ge le eq ne in. Character fields support: eq ne in contains.<br>Example: = |
| Value | yes | no | String | Filter operation value.<br>Example: 10 |
| MetricSetCID | no | no | String | Metric set CID (only for output parameter; leave it empty for input parameter).<br>Example: cid-xxx |
| CreatedAt | no | no | String | Creation time (only for output parameter; leave it empty for input parameter)<br>Example: qwer |
| UpdatedAt | no | no | String | Update time (only for output parameter; leave it empty for input parameter)<br>Example: qwer |

## ModifyRecoverNotifyBatchNotifyInfo

ModifyRecoverNotifyBatch recovery method information.

Referenced by the following interfaces: ModifyRecoverNotifyBatch

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| GroupId | yes | no | Int64 | Policy group ID.<br>Example: 1234 |
| RecoverNotify | yes | no | Array of String | List of alarm recovery notification methods.<br>Example: sms |

## AlarmBindingInstance

AlarmBindingInstance

Referenced by the following interfaces: DescribeAlarmBindingInstanceList

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| UUID | yes | no | String | Instance UUID.<br>Example: u78haeoyaq8w8wbr |
| Information | yes | no | String | Information about instance dimension combinations.<br>Example: {"appid":1251000011,"projectid":10,"vm_uuid":"9566dd15-fb0a-4515-9b87-09ad16d87884"} |

## ClmDescribeMetricSetsData

DescribeMetricSets return data.

Referenced by the following interfaces: DescribeMetricSets

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| MetricSets | yes | yes | Array of ClmMetricSet | Metric set list data.<br>Example:View |
| TotalCount | yes | no | Int64 | Total number.<br>Example: 10 |

## DescribePolicyConditionListConfigManualCalcValue

DescribePolicyConditionList.ConfigManual.CalcValue

Referenced by the following interfaces: DescribePolicyConditionList

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Default | yes | yes | String | Default value.<br>Example: test |
| Fixed | yes | yes | String | Fixed value.<br>Example: qwer |
| Max | yes | yes | String | Maximum value.<br>Example: 100 |
| Min | yes | yes | String | Minimum value.<br>Example: 10 |
| Need | yes | no | Bool | Whether required.<br>Example: true |

## AlarmPolicyFilter

Filter condition of alarm policy

Referenced by the following interfaces: CreateAlarmPolicy, DescribeAlarmPolicies, DescribeAlarmPolicy, ModifyAlarmPolicyCondition

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Type | yes | no | String | Filter condition type. Valid values: DIMENSION (uses dimensions for filtering).<br>Example: = |
| Expression | no | no | String | Bare write filter expression.<br>Example: test |
| Dimensions | no | no | String | JSON string generated by serializing the AlarmPolicyDimension two-dimensional array. The one-dimensional arrays are in OR relationship, and the elements in a one-dimensional array are in AND relationship.<br>Example: test |

## AttributeIdDeleteOutput

AttributeIdDeleteOutput

Referenced by the following interfaces: DeleteAttribute, DeleteAttributes

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| DeleteCount | yes | no | Int64 | Number of deletions.<br>Example: 1 |
| AttributeId | yes | no | Array of Uint64 | Attribute ID.<br>Example: 10 |

## PolicyGroupItem

PolicyGroups returned by the DescribeInstanceGroup API.

Referenced by the following interfaces: DescribeInstanceGroup

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| GroupId | yes | yes | Int64 | Policy ID.<br>Example: 1234 |
| GroupName | yes | yes | String | Policy name.<br>Example: Testing |

## BigCustomerNotice

Major client notification.

Referenced by the following interfaces: CreateAlarmNotice, DescribeAlarmNotice, DescribeAlarmNotices, DescribeAlarmPolicies, DescribeAlarmPolicy

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| NeedPushChatGroup | yes | no | Int64 | Whether to push to the after-sales group of major clients. 0: no. 1: yes.<br>Example: 1 |

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| StartTime | yes | no | Int64 | Number of seconds since notification start time 00:00:00 (Value range: 0-86399).<br>Example: 0 |
| EndTime | yes | no | String | Number of seconds since notification end time 00:00:00 (Value range: 0-86399).<br>Example: 3600 |

### DescribeProductEventListEventsDimensions

Dimensions of events returned by the DescribeProductEventList API

Referenced by the following interfaces: DescribeProductEventList

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| Key | yes | yes | String | Dimension name (English).<br>Example: readOnly |
| Name | yes | yes | String | Dimension name.<br>Example: Disk read-only |
| Value | yes | yes | String | The value of the dimension.<br>Example: qwer |

### AlarmPolicyTriggerTask

Task triggered by alarm policy

Referenced by the following interfaces: CreateAlarmPolicy, DescribeAlarmPolicies, DescribeAlarmPolicy, ModifyAlarmPolicyTasks

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| Type | yes | no | String | Triggered task type. Valid value: AS (auto scaling).<br>Example: AS |
| TaskConfig | yes | no | String | Configuration information in JSON format, such as {"Key1":"Value1","Key2":"Value2").<br>Example: {"Key1":"Value1","Key2":"Value2"} |

### MetricConfig

Metric configuration

Referenced by the following interfaces: DescribeAlarmMetrics

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| Operator | yes | no | Array of String | Allowed operator.<br>Example: = |
| Period | yes | no | Array of Int64 | Allowed data cycle in seconds.<br>Example: 5 |
| ContinuePeriod | yes | no | Array of Int64 | Allowed number of consecutive periods.<br>Example: 5 |

### MonitorTypeNamespace

Policy type

Referenced by the following interfaces: DescribeAlarmHistories

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| MonitorType | yes | no | String | Monitoring type.<br>Example: MT_QCE |
| Namespace | yes | no | String | Policy type value.<br>Example: cvm_device |

### DescribePolicyConditionListCondition

Policy conditions returned by the DescribePolicyConditionList API

Referenced by the following interfaces: DescribePolicyConditionList

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| PolicyViewName | yes | no | String | Policy view name.<br>Example: cvm_device |
| EventMetrics | yes | yes | Array of DescribePolicyConditionListEventMetric | Event alarm conditions<br>Example:View |
| IsSupportMultiRegion | yes | no | Bool | Whether multiple regions are supported.<br>Example: true |
| Metrics | yes | yes | Array of DescribePolicyConditionListMetric | Metric alarm conditions.<br>Example:View |
| Name | yes | no | String | Policy type name.<br>Example: test |
| SortId | yes | no | Int64 | Sorting ID.<br>Example: 1 |
| SupportDefault | yes | no | Bool | Whether the default policy is supported.<br>Example: true |
| SupportRegions | yes | yes | Array of String | List of regions that support this policy type.<br>Example: ["gz"] |
| DeprecatingInfo | no | yes | String | Expiration Information<br>Example: qwer |

### DescribePolicyConditionListConfigManualPeriodNum

DescribePolicyConditionList.ConfigManual.PeriodNum

Referenced by the following interfaces: DescribePolicyConditionList

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| Default | yes | yes | Int64 | Number of default periods.<br>Example: 10 |
| Keys | yes | yes | Array of Int64 | Number of custom periods.<br>Example: 10 |
| Need | yes | no | Bool | Whether required.<br>Example: true |

### DescribeProductEventListEventsGroupInfo

GroupInfo in Events returned by the DescribeProductEventList API

Referenced by the following interfaces: DescribeProductEventList

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| GroupId | yes | yes | Int64 | Policy ID.<br>Example: 1234 |
| GroupName | yes | yes | String | Policy name.<br>Example: Testing |

### ClmAlertPolicy

Expired.

Referenced by the following interfaces: CreateAlertPolicy, DescribeAlertPolicies, DescribeAlertPolicy, ModifyAlertPolicy

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| Name | yes | yes | String | Policy name.<br>Example: test |
| MetricSetId | yes | yes | Int64 | Metric set ID (invalid parameter).<br>Example: 1 |
| Status | yes | yes | Int64 | Status. 1: enabled. 2: not enabled.<br>Example: 1 |
| AlertGroupId | yes | yes | String | Alarm recipient group ID. Multiple IDs are separated by commas.<br>Example: 1234 |
| AlertChannel | yes | yes | String | Channels for receiving alarms: "SMS", "EMAIL", "WECHAT", "CALL". Multiple values are separated by commas.<br>Example: sms |
| NoticePeriodBegin | yes | yes | Int64 | Start time of the time range to send notifications (number of seconds since 00:00:00).<br>Example: 100 |
| NoticePeriodEnd | yes | yes | Int64 | End time of the time range to send notifications (number of seconds since 00:00:00).<br>Example: 200 |
| UrlScheme | no | yes | String | Scheme of the callback URL.<br>Example: http |
| CallbackUrl | no | yes | String | Callback URL, excluding the scheme part.<br>Example: http://test.com |
| AlertFilterRules | no | yes | Array of ClmAlertFilterRule | List of alarm filter conditions.<br>Example:View |
| AlertTriggerRules | yes | yes | Array of ClmAlertTriggerRule | Alarm trigger condition list.<br>Example:View |

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Id | no | yes | Int64 | Alarm policy ID (output parameter).<br>Example: 1 |
| CreatedAt | no | yes | String | Creation time (output parameter).<br>Example: eqeadfrqc |
| UpdatedAt | no | yes | String | Update time (output parameter).<br>Example: eqeadfrqc |
| AlertUserId | no | yes | String | Alarm recipient ID. Multiple IDs are separated by commas.<br>Example: 1234 |
| LatestAlertTime | no | yes | String | Last alarm time (output parameter).<br>Example: 1735041205 |
| FilterRelation | no | yes | Int64 | Relationship between filter conditions. 1: AND. 2: OR.<br>Example: 1 |
| TriggerRelation | yes | yes | Int64 | Relationship between trigger conditions. 1: AND. 2: OR.<br>Example: 1 |
| MetricSetCID | no | no | Float | Metric set CID.<br>Example: 123455 |

## AttributeTimestampValueOutput

AttributeTimestampValueOutput

Referenced by the following interfaces: DescribeAttributeAggregateData, DescribeCCMInstanceDatas, DescribeServerData, DescribeServerDatas, DescribeViewData

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Timestamp | yes | no | Int64 | Timestamp.<br>Example: 1735041205 |
| Value | yes | yes | Int64 | Attribute value.<br>Example: 10 |

## ClmMetricCustomItem

Compound metric configurations for the log monitoring metric set.

Referenced by the following interfaces: CreateMetricSet, DescribeMetricSet, DescribeMetricSets, ModifyMetricSet

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Formula | yes | no | String | Expression.<br>Example: {"cluster_name":"es-dnxssss"} |
| Name | yes | no | String | Metric name.<br>Example: Testing |
| Desc | yes | no | String | Metric description name.<br>Example: Testing |
| Id | no | no | Int64 | Item ID (only for output parameter; leave it empty for input parameter)<br>Example: 1 |
| MetricSetId | no | no | Int64 | Metric set ID (only for output parameter; leave it empty for input parameter)<br>Example: 12 |
| CreatedAt | no | no | String | Creation time (only for output parameter; leave it empty for input parameter)<br>Example: eqeadfrqc |
| UpdatedAt | no | no | String | Update time (only for output parameter; leave it empty for input parameter)<br>Example: eqeadfrqc |

## ServiceInfo

ServiceInfo

Referenced by the following interfaces: CreateService, DescribeService

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| ServiceId | yes | yes | Int64 | Service ID.<br>Example: 1 |
| IsOpen | yes | yes | Int64 | Whether to enable the service.<br>Example: 1 |
| ViewId | yes | yes | Int64 | Default view ID.<br>Example: 10 |

## DescribeInstanceGroupListInstanceGroup

Instance group information returned by the DescribeInstanceGroupList API.

Referenced by the following interfaces: DescribeInstanceGroupList

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| InstanceGroupId | yes | no | Int64 | Instance group ID.<br>Example: 1234 |
| ViewName | yes | no | String | Instance group policy type name.<br>Example: cvm_device |
| LastEditUin | yes | no | String | UIN of the latest update.<br>Example: 123456 |
| GroupName | yes | no | String | Instance group name.<br>Example: Testing |
| InstanceSum | yes | no | Int64 | Number of instances in the group.<br>Example: 10 |
| UpdateTime | yes | no | Int64 | Update time in the format of Unix timestamp.<br>Example: 1735095122 |
| InsertTime | yes | no | Int64 | Creation time in the format of Unix timestamp.<br>Example: 1735095122 |
| PolicyGroups | yes | yes | Array of DescribeInstanceGroupListPolicyGroup | List of bound alarm policy groups.<br>Example:View |

## DescribePolicyGroupListGroup

DescribePolicyGroupList.Group

Referenced by the following interfaces: DescribePolicyGroupList

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| GroupId | yes | no | Int64 | Policy group ID.<br>Example: 1234 |
| GroupName | yes | no | String | Policy group name.<br>Example: test |
| IsOpen | yes | no | Bool | Indicates whether self-healing notifications are enabled.<br>Example: true |
| ViewName | yes | no | String | Policy view name.<br>Example: cvm_device |
| LastEditUin | yes | no | String | UIN of the last editor.<br>Example: 123456 |
| UpdateTime | yes | no | Int64 | Last modification time.<br>Example: 12314155251 |
| InsertTime | yes | no | Int64 | Creation time.<br>Example: 12314155251 |
| UseSum | yes | no | Int64 | Number of instances bound to the policy group.<br>Example: 10 |
| NoShieldedSum | yes | no | Int64 | Number of unshielded instances that are bound to the policy group.<br>Example: 10 |
| IsDefault | yes | no | Int64 | Whether it is a default policy. 0: non-default policy. 1: default policy.<br>Example: 1 |
| CanSetDefault | yes | no | Bool | Whether it can be set as the default policy.<br>Example: true |
| ParentGroupId | yes | no | Int64 | Parent policy group ID.<br>Example: 1234 |
| Remark | yes | no | String | Remarks of the policy group.<br>Example: test |
| ProjectId | yes | no | Int64 | ID of the project to which the policy group belongs.<br>Example: 0 |
| Conditions | yes | yes | Array of DescribePolicyGroupInfoCondition | Threshold rules list.<br>Example:View |
| EventConditions | yes | yes | Array of DescribePolicyGroupInfoEventCondition | List of product event rules.<br>Example:View |
| ReceiverInfos | yes | yes | Array of DescribePolicyGroupInfoReceiverInfo | User recipient list.<br>Example:View |
| ConditionsTemp | yes | yes | DescribePolicyGroupInfoConditionTpl | Template-based policy group.<br>Example:View |
| InstanceGroup | yes | yes | DescribePolicyGroupListGroupInstanceGroup | Instance group that is bound to the policy group.<br>Example:View |
| IsUnionRule | yes | yes | Int64 | The 'AND' or 'OR' rule. The value 0 indicates the 'OR' rule (indicating that an alarm will be triggered if any rule meets the threshold condition). The value 1 indicates the 'AND' rule (indicating that an alarm will be triggered when all rules meet the threshold conditions).<br>Example: 1 |

## CreatePolicyGroupCondition

Alarm threshold condition passed in when a policy is created.

Referenced by the following interfaces: CreatePolicyGroup

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| MetricId | yes | no | Int64 | Metric ID.<br>Example: 1234 |
| CalcType | no | no | Int64 | Comparative type. The value 1 indicates greater than. The value 2 indicates greater than or equal to. The value 3 indicates smaller than. The value 4 indicates smaller than or equal to. The value 5 indicates equal to. The value 6 indicates not equal to. This parameter is optional if a default comparative type is configured for the metric.<br>Example: 1 |
| CalcValue | no | no | Float | Comparative value. This parameter is optional if the metric has no requirement.<br>Example: 1.13 |
| CalcPeriod | no | no | Int64 | Data aggregation period in seconds. This parameter is optional if the metric has a default value.<br>Example: 1 |
| ContinuePeriod | no | no | Int64 | Number of consecutive periods after which an alarm will be triggered.<br>Example: 5 |
| AlarmNotifyType | yes | no | Int64 | Alarm sending and converging type. The value 0 indicates that alarms are sent consecutively. The value 1 indicates that alarms are sent exponentially.<br>Example: 0 |
| AlarmNotifyPeriod | yes | no | Int64 | Alarm sending period in seconds. The value u003c0 indicates that no alarm will be triggered. The value 0 indicates that an alarm is triggered only once. The value >0 indicates that an alarm is triggered at the interval of triggerTime.<br>Example: 0 |
| RuleId | no | no | Int64 | If a metric is created based on a template, the RuleId of the metric in the template must be passed in.<br>Example: 1 |

## DashboardMetricQuery

Dashboard 2.0 metric query input parameter structure.

Referenced by the following interfaces: DescribeDashboardMetricData

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| StartTime | yes | no | Datetime_iso | Start time of the query period in ISO format, such as: 2018-04-10T10:15:58.858Z.<br>Example: 2019-11-29T12:30:13+08:00 |
| EndTime | yes | no | Datetime_iso | End time of the query period in ISO format, such as: 2018-04-10T10:15:58.858Z.<br>Example: 2019-03-24T20:50:00+08:00 |
| Datasource | yes | no | String | Data source.<br>Example: DS_QCEMetric |
| Namespace | yes | no | String | The name of the namespace.<br>Example: QCE/CVM |
| MetricName | yes | no | String | Name of the query metric.<br>Example: LanOuttraffic |
| Period | no | no | Int64 | Statistical period in seconds. The default value is 60 seconds.<br>Example: 60 |
| GroupBy | yes | no | Array of String | Group. Fill in the metric dimension.<br>Example: LanOuttraffic |
| Aggregate | no | no | String | Aggregation method. Fill in the aggregation method according to the metric type.<br>Example: min |
| Conditions | yes | no | Array of DashboardMetricCondition | Filter condition.<br>Example:View |
| SeriesId | no | no | String | Query flag.<br>Example: 1a6deff1-7991-4d82-a8c1-44b66e1c33f6 |
| DataSourceRegion | no | no | String | Name of the data source region.<br>Example: gz |
| ConfigId | no | no | String | Product configuration ID.<br>Example: cvm |
| QueryVersion | no | no | String | Queries the version number. Leave it empty for the old version. For the new version, pass in "2020-10-21".<br>Example: 2020-10-21 |
| OrderBy | no | no | String | Calculation method of sort value. Valid values: avg, sum, min, max.<br>Example: desc |
| AscDesc | no | no | String | Sort order. Valid values: asc, desc.<br>Example: desc |
| Limit | no | no | Uint64 | Number of items to be sorted.<br>Example: 10 |

## DescribePolicyGroupInfoCallback

User callback information output by the policy query

Referenced by the following interfaces: DescribePolicyGroupInfo

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| CallbackUrl | yes | no | String | URL of the user callback API.<br>Example: http://test.com |
| ValidFlag | yes | no | Int64 | Status of the user callback API. The value 0 indicates that the API is not verified. The value 1 indicates that the API is verified. The value 2 indicates that a URL exists but the API fails to be verified.<br>Example: 1 |
| VerifyCode | yes | no | String | Verification code of the user callback API.<br>Example: 13eq |

## ReceiverInfo

Recipient information.

Referenced by the following interfaces: ModifyAlarmReceivers

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Id | no | no | Int64 | The ID.<br>Example: 1234 |
| StartTime | yes | no | Int64 | Alarm period start time. Range: [0,86400). Remove the date after converting the time from UNIX timestamp to UTC+8 time. For example, 7200 means "10:0:0".<br>Example: 1734953008 |
| EndTime | yes | no | Int64 | End time of the alarm period. The meaning is the same as that of StartTime.<br>Example: 1734953808 |
| NotifyWay | yes | no | Array of String | Alarm notification method. Valid values: "SMS", "SITE", "EMAIL", "CALL", and "WECHAT".<br>Example: sms |
| SendFor | no | no | Array of String | Alarm call notification time. Valid values: OCCUR (indicating that a notice is sent when the alarm is triggered) and RECOVER (indicating that a notice is sent when the alarm is recovered).<br>Example: qwer |
| UidList | no | no | Array of Int64 | UID of the alarm call recipient.<br>Example: 1234 |
| RoundNumber | no | no | Int64 | Number of alarm call rounds.<br>Example: 1 |
| PersonInterval | no | no | Int64 | Alarm call intervals for individuals in seconds.<br>Example: 1 |
| RoundInterval | no | no | Int64 | Intervals of alarm call rounds in seconds.<br>Example: 1 |
| RecoverNotify | no | no | Array of String | Notification method when an alarm is recovered. Valid value: SMS.<br>Example: sms |
| NeedSendNotice | no | no | Int64 | Whether to send an alarm call delivery notice. The value 0 indicates that no notice needs to be sent. The value 1 indicates that a notice needs to be sent.<br>Example: 1 |
| ReceiverType | yes | no | Array of String | Recipient type. Valid values: user, group.<br>Example: group |
| ReceiverGroupList | no | no | Array of Int64 | Recipient group list. The list of recipient group IDs that is queried by API.<br>Example: [1234] |
| ReceiverUserList | no | no | Array of Int64 | Recipient list. The list of recipient IDs that is queried by API.<br>Example: [1234] |

## DescribePolicyConditionListEventMetric

DescribePolicyConditionList.EventMetric

Referenced by the following interfaces: DescribePolicyConditionList

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| EventId | yes | no | Int64 | Event ID.<br>Example: 10 |
| EventShowName | yes | no | String | Event name.<br>Example: ReadOnly |
| NeedRecovered | yes | no | Bool | Whether to recover.<br>Example: true |
| Type | yes | no | Int64 | Event type, which is a reserved field. Currently, it is fixed to 2.<br>Example: 1 |

## StrategyStatesInfo

StrategyStatesInfo

Referenced by the following interfaces: ModifyStrategyStates

| name | required | NULL allowed | type | description |
|---|---|---|---|---|

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| StrategyId | yes | no | Int64 | The ID of the alarm policy.<br>Example: 1 |
| IsStart | yes | no | Int64 | Enablement status. Default value: 1. 0: stopping. 1: starting.<br>Example: 1 |

## DescribeInstanceGroupListPolicyGroup

Policy group information returned by the DescribeInstanceGroupList API.

Referenced by the following interfaces: DescribeInstanceGroupList

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| GroupId | yes | no | Int64 | Policy group ID.<br>Example: 1234 |
| GroupName | yes | no | String | Policy group name.<br>Example: Testing |

## DescribeBasicAlarmListAlarms

Alarms returned by DescribeBasicAlarmList

Referenced by the following interfaces: DescribeBasicAlarmList

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Id | yes | no | Uint64 | ID of the alarm.<br>Example: 1 |
| ProjectId | yes | yes | Int64 | Project ID.<br>Example: 1 |
| ProjectName | yes | yes | String | Project Name<br>Example: Testing |
| Status | yes | yes | Int64 | Alarm status ID.<br>Example: 1 |
| AlarmStatus | yes | yes | String | Alarm status.<br>Example: ALARM |
| GroupId | yes | yes | Int64 | Policy group ID.<br>Example: 1234 |
| GroupName | yes | yes | String | Policy group name.<br>Example: Testing |
| FirstOccurTime | yes | yes | String | Event time.<br>Example: 1735095122 |
| Duration | yes | yes | Int64 | Duration in seconds.<br>Example: 10 |
| LastOccurTime | yes | yes | String | End time.<br>Example: 1735095122 |
| Content | yes | yes | String | The alarm content.<br>Example: qwer |
| ObjName | yes | yes | String | Alarm object.<br>Example: 172.16.0.15#75904 |
| ObjId | yes | yes | String | Alarm object ID.<br>Example: 0a523bfb-dd09-47e2-a85c-0b23455b0949 |
| ViewName | yes | yes | String | Policy type<br>Example: cvm_device |
| Vpc | yes | yes | String | VPC, which is unique to CVM.<br>Example: qwer |
| MetricId | yes | yes | Int64 | Metric ID.<br>Example: 10 |
| MetricName | yes | yes | String | Metric name.<br>Example: CpuUsage |
| AlarmType | yes | yes | Int64 | Alarm type. The value 0 indicates metric alarms. The value 2 indicates product event alarms. The value 3 indicates platform event alarms.<br>Example: 1 |
| Region | yes | yes | String | Region.<br>Example: gz |
| Dimensions | yes | yes | String | Dimension information of an alarm object.<br>Example: {"unInstanceId":"ins-xxx"} |
| NotifyWay | yes | yes | Array of String | Notification method.<br>Example: sms |
| InstanceGroup | yes | yes | Array of InstanceGroup | Instance group information.<br>Example:View |

## PCLMDimension

Metric set dimension object for log monitoring.

Referenced by the following interfaces: CLMDescribeMetricSets

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Field | yes | no | String | Field name.<br>Example: timeField |
| CreatedAt | no | no | String | Creation time (only for output parameter; leave it empty for input parameter)<br>Example: qwer |
| UpdatedAt | no | no | String | Update time (only for output parameter; leave it empty for input parameter)<br>Example: qwer |
| MetricSetCID | no | no | String | Metric set CID (only for output parameter; leave it empty for input parameter).<br>Example: cid-xxx |

## DescribeBaseMetricsDataMeaning

DescribeBaseMetricsDataMeaning

Referenced by the following interfaces: DescribeBaseMetricsForConsoleFontEnd

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Language | yes | no | String | Language.<br>Example: zh |
| Meaning | yes | no | String | Meaning.<br>Example: qwer |

## StrategyData

StrategyData

Referenced by the following interfaces: DescribeCCMGroupViewStrategy, DescribeStrategys

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Data | yes | no | Array of StrategyEntry | Return data.<br>Example:View |
| TotalCount | yes | no | Int64 | Queries the total number of data lists.<br>Example: 10 |

## PCLMMetricAnalysisData

CLM metric analysis data structure.

Referenced by the following interfaces: DescribeMetricAnalysisData

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Compare | yes | yes | Array of PCLMMetricAnalysisPoint | Comparison data.<br>Example:View |
| Current | yes | yes | Array of PCLMMetricAnalysisPoint | Current data.<br>Example:View |
| Granularity | yes | yes | Int64 | Time granularity (seconds).<br>Example: 1 |

## AlarmHistory

Alarm record data

Referenced by the following interfaces: DescribeAlarmHistories, DescribeMiniDashboardAlarmInfo

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| AlarmId | yes | no | String | Alarm history ID.<br>Example: 01e1f33f-b4ec-43db-8fd0-6f7e163e6ca5 |
| MonitorType | yes | no | String | Monitoring type.<br>Example: MT_QCE |
| Namespace | yes | no | String | Policy type<br>Example: cvm_device |
| AlarmObject | yes | no | String | Alarm object.<br>Example: ins-abcdef |

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Content | yes | no | String | The alarm content.<br>Example: CPU utilization > 80% |
| FirstOccurTime | yes | no | Int64 | Timestamp of the first occurrence time.<br>Example: 1611700000 |
| LastOccurTime | yes | no | Int64 | Timestamp of the final occurrence time.<br>Example: 1611734567 |
| AlarmStatus | yes | no | String | Alarm status. ALARM=not recovered. OK=recovered. NO_CONF=expired. NO_DATA=insufficient data.<br>Example: ALARM |
| PolicyId | yes | no | String | Alarm policy ID.<br>Example: policy-abcdef |
| PolicyName | yes | no | String | Policy name.<br>Example: CVM alarm policy |
| VPC | yes | no | String | VPC of the alarm object for the basic product alarm.<br>Example: 0 |
| ProjectId | yes | no | Int64 | Project ID.<br>Example: 0 |
| ProjectName | yes | no | String | Project name.<br>Example: Default project |
| InstanceGroup | yes | yes | Array of InstanceGroups | Instance group of alarm object<br>Example:View |
| ReceiverUids | yes | no | Array of Int64 | Recipient list.<br>Example: [1234] |
| ReceiverGroups | yes | no | Array of Int64 | Recipient group list.<br>Example: [1234] |
| NoticeWays | yes | no | Array of String | Alarm channel list. Valid values: SMS (SMS), EMAIL (email), CALL (phone), WECHAT (WeChat).<br>Example: ["SMS"] |
| OriginId | yes | no | String | Compatibility alarm v1.0 policy group ID.<br>Example: 12345 |
| AlarmType | yes | no | String | Alarm type.<br>Example: METRIC |
| EventId | yes | no | Int64 | Event ID.<br>Example: 0 |
| Region | yes | no | String | Region.<br>Example: region1 |
| PolicyExists | yes | no | Int64 | Whether the policy exists. 0: no. 1: yes.<br>Example: 1 |
| ExtraTags | yes | yes | String | Extra information. JSON string.<br>Example: {"cluster_name":"es-dnxssssss"} |
| MetricsInfo | yes | yes | Array of AlarmHistoryMetric | Metric information.<br>Example:View |
| Dimensions | no | yes | String | Dimensions<br>Example: {"cluster_name":"es-dnxssssss"} |
| AlarmLevel | no | no | String | Alarm Level<br>Example: Remind |

## DescribeBaseMetricsForConsoleFontEndData

DescribeBaseMetricsForConsoleFontEndData

Referenced by the following interfaces: DescribeBaseMetricsForConsoleFontEnd

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| NameSpace | yes | no | String | Namespace.<br>Example: QCE/CVM |
| MetricName | yes | no | String | Metric name.<br>Example: CpuUsage |
| MetricCNName | yes | no | String | Metric name.<br>Example: CPU Usage |
| MetricENName | yes | no | String | English name of metrics.<br>Example: CpuUsage |
| Dimensions | yes | no | String | Dimension.<br>Example: {"unInstanceId":"ins-xxx"} |
| Unit | yes | no | String | Unit.<br>Example: % |
| UnitCNName | yes | no | String | Unit name.<br>Example: Percentage |
| PeriodAndStatTypes | yes | yes | Array of DescribeBaseMetricsDataPeriod | Period and statistical method.<br>Example:View |
| Meaning | yes | yes | Array of DescribeBaseMetricsDataMeaning | Metric description.<br>Example:View |

## DescribePolicyInfoByInstanceInstanceGroup

DescribePolicyInfoByInstanceInstanceGroup

Referenced by the following interfaces: DescribePolicyInfoByInstance

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| InstanceGroupID | yes | yes | Int64 | Instance group ID.<br>Example: 1234 |
| ViewName | yes | no | String | View.<br>Example: cvm_device |
| LastEditUin | yes | yes | Int64 | UIN of the last editor.<br>Example: 123456 |
| GroupName | yes | no | String | Instance group name.<br>Example: Testing |
| InstanceSum | yes | yes | Int64 | Number of instances.<br>Example: 10 |
| UpdateTime | yes | yes | Int64 | Update time.<br>Example: 1735095122 |
| InsertTime | yes | yes | Int64 | Creation time.<br>Example: 1735095122 |

## DescribeProductEventListOverView

OverView object returned by the DescribeProductEventList API

Referenced by the following interfaces: DescribeProductEventList

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| StatusChangeAmount | yes | yes | Int64 | Number of state change events.<br>Example: 1 |
| UnConfigAlarmAmount | yes | yes | Int64 | Number of events corresponding to which alarm states are not configured.<br>Example: 1 |
| UnNormalEventAmount | yes | yes | Int64 | Number of abnormal events.<br>Example: 1 |
| UnRecoverAmount | yes | yes | Int64 | Number of unrecovered events.<br>Example: 1 |

## MidQueryCondition

DescribeMidDimensionValueList query conditions

Referenced by the following interfaces: DescribeMidDimensionValueList, DescribeStatisticData

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Key | yes | no | String | Dimension.<br>Example: qwer |
| Operator | yes | no | String | Operator. Valid values: eq (equal to), ne (not equal to), in.<br>Example: = |
| Value | yes | no | Array of String | Dimension value. If Operator is eq or ne, only the first element will be used.<br>Example: 10 |

## UnifyDashboardMeta

Dashboard list element information.

Referenced by the following interfaces: DescribeUnifyDashboards

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| UUID | yes | no | String | dashboard uuid<br>Example: 1234 |
| IsStarred | yes | no | Bool | Whether to add to "Favorite".<br>Example: false |
| Title | yes | no | String | Dashboard name.<br>Example: 1735179378 |

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| IsDefault | yes | no | Bool | Default<br>Example: false |
| Type | yes | no | String | Dashboard type.<br>Example: qwer |
| LastVisitTime | yes | no | String | Last access time.<br>Example: 1735179378 |
| CreatedAt | yes | no | String | Creation time.<br>Example: qwer |
| UpdatedAt | yes | no | String | Update time.<br>Example: qwer |
| FolderUUID | no | no | String | The folder UUID.<br>Example: f-ikxn2fzkagqrdjjo |
| IsFolder | no | no | Bool | Whether it is a folder.<br>Example: false |
| Order | no | no | String | Order.<br>Example: desc |

## NoticeContentTmplBindInfo

Information on binding of the notification template and content template.

Referenced by the following interfaces: DescribeAlarmPolicies, DescribeAlarmPolicy, ModifyAlarmPolicyNotice

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| ContentTmplID | no | no | String | Content template ID.<br>Example: tmpl-xxx |
| NoticeID | no | no | String | Notification template ID.<br>Example: notice-xxx |

## MetricData

DescribeMetricData output parameters

Referenced by the following interfaces: DescribeDataForMiniProgram, DescribeStatisticData

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| MetricName | yes | no | String | None.<br>Example: CpuUsage |
| Points | yes | no | String | None.<br>Example: qwer |

## ConditionsTemp

Alarm condition template

Referenced by the following interfaces: DescribeAlarmPolicies, DescribeAlarmPolicy

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| ConditionId | yes | yes | Int64 | Alarm condition ID.<br>Example: 1234 |
| TemplateName | yes | yes | String | Template name.<br>Example: test |
| Condition | yes | yes | AlarmPolicyCondition | Metric trigger condition.<br>Example:View |
| EventCondition | yes | yes | AlarmPolicyEventCondition | Event trigger condition.<br>Example:View |

## ProductSimple

Brief information on products supported by Cloud Monitor.

Referenced by the following interfaces: DescribeProductList

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Namespace | yes | no | String | Namespace.<br>Example: QCE/CVM |
| ProductName | yes | no | String | The name of the product.<br>Example: cvm_device |
| ProductEnName | yes | no | String | English name of the product.<br>Example: test |

## CustomAlarmList

CustomAlarmList

Referenced by the following interfaces: DescribeCustomAlarmList

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| LocalIP | yes | no | String | IP<br>Example: 127.0.0.1 |
| Msg | yes | no | String | Message.<br>Example: ok |
| OccurTime | yes | no | String | Event time.<br>Example: 1735041205 |
| Caller | yes | no | String | Caller.<br>Example: remote |

## DashboardMetricData

Reported data returned by Dashboard2.0 query metrics.

Referenced by the following interfaces: DescribeDashboardMetricData

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| MetricName | yes | no | String | Metric name.<br>Example: CpuUsage |
| Aggregate | yes | no | String | The aggregation method.<br>Example: avg |
| GroupBy | yes | no | Array of Dimension | Query condition array.<br>Example:View |
| Timestamps | yes | no | Array of Uint64 | Returns the Timestamp array.<br>Example: [1604303550] |
| Value | yes | no | String | Metric report data. JSON serialized strings are returned in array form.<br>Example: [1,0.8] |
| StartTime | yes | no | Datetime_iso | Returns the start time of the query period in ISO format, such as: 2018-04-10T10:15:58.858Z.<br>Example: 2019-11-29T12:30:13+08:00 |
| EndTime | yes | no | Datetime_iso | Returns the start time of the query period in ISO format, such as: 2018-04-10T10:15:58.858Z.<br>Example: 2019-03-24T20:50:00+08:00 |
| Period | yes | no | Int64 | Statistical period in seconds.<br>Example: 60 |
| Conditions | yes | no | Array of DashboardMetricCondition | Filter condition.<br>Example:View |
| Namespace | yes | no | String | Namespace.<br>Example: QCE/CVM |
| Error | yes | no | String | Error message.<br>Example: error |
| Code | yes | no | String | Error code.<br>Example: 11001 |
| SeriesId | yes | no | String | Query flag.<br>Example: 1a6deff1-7991-4d82-a8c1-44b66e1c33f6 |
| ConfigId | yes | yes | String | Product configuration ID.<br>Example: qwer |
| Dimensions | yes | no | Array of Dimension | Dimension combination of an instance.<br>Example:View |
| OrderTag | yes | yes | Float | Field for sorting.<br>Example: 1234 |
| Type | no | yes | String | Type<br>Example: qwer |

## UserNotice

Cloud Monitor alarm notification template - user notification details

Referenced by the following interfaces: CreateAlarmNotice, DescribeAlarmNotice, DescribeAlarmNotices, DescribeAlarmPolicies, DescribeAlarmPolicy, ModifyAlarmNotice

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| ReceiverType | yes | no | String | Recipient type. Valid values: USER (user), GROUP (user group).<br>Example: USER |

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| UserIds | no | yes | Array of Int64 | User uid list.<br>Example: [100001,100002] |
| GroupIds | no | yes | Array of Int64 | User group ID list.<br>Example: [10001,10002] |
| StartTime | yes | no | Int64 | Number of seconds since notification start time 00:00:00 (Value range: 0-86399).<br>Example: 0 |
| EndTime | yes | no | Int64 | Number of seconds since notification end time 00:00:00 (Value range: 0-86399).<br>Example: 86399 |
| NoticeWay | yes | no | Array of String | Notification channel list. Valid values: EMAIL (email), SMS (SMS), CALL (phone), WECHAT (WeChat).<br>Example: ["SMS", "CALL"] |
| PhoneOrder | no | yes | Array of Int64 | Phone polling list.<br>Example: [100001,100002] |
| PhoneCircleTimes | no | no | Int64 | Number of phone pollings. Value range: 1–5.<br>Example: 1 |
| PhoneInnerInterval | no | no | Int64 | Call interval in seconds within one polling. Value range: 60–900.<br>Example: 60 |
| PhoneCircleInterval | no | no | Int64 | Polling interval in seconds. Value range: 60–900.<br>Example: 60 |
| NeedPhoneArriveNotice | no | no | Int64 | Whether receipt notification is required. Valid values: 0 (no), 1 (yes).<br>Example: 1 |

## DashboardDimensionSource

Dashboard2.0 dimension list.

Referenced by the following interfaces: DescribeDashboardDimensionKeys

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| DataSource | yes | no | String | Data source.<br>Example: DS_QCEMetric |
| Namespace | yes | no | String | Namespace.<br>Example: cvm_device |
| DimensionName | yes | no | String | Dimension name.<br>Example: Testing |
| MetricName | yes | no | String | Metric name.<br>Example: cpu_usage |

## ServerInfoData

ServerInfoData

Referenced by the following interfaces: DescribeServers

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| Data | yes | no | Array of ServerInfo | Return data.<br>Example:View |
| TotalCount | yes | no | Int64 | Total number.<br>Example: 10 |

## AlarmPolicyBasic

Basic information of a v2.0 alarm policy.

Referenced by the following interfaces: DescribeAlarmPoliciesBasic

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| MonitorType | yes | yes | String | Monitoring type.<br>Example: MT_QCE |
| OriginId | yes | yes | String | V1.0 alarm policy ID.<br>Example: 1234 |
| PolicyId | no | yes | String | V2.0 alarm policy ID.<br>Example: policy-abc123 |
| PolicyName | no | yes | String | The name of the alarm policy.<br>Example: Testing |

## HistoryAlarmInfo

Return parameters of the DescribeAlarmHistoryByAlarmId API.

Referenced by the following interfaces: DescribeAlarmHistoryByAlarmId

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| Id | yes | no | Int64 | Alarm ID.<br>Example: 1 |
| GroupId | yes | no | Int64 | Policy group ID.<br>Example: 1234 |
| AlarmStatus | yes | yes | String | Alarm status.<br>Example: 1 |
| AlarmType | yes | yes | Int64 | Alarm type. 0 (metric alarm), 2 (product event alarm), 3 (platform event alarm).<br>Example: 1 |
| Content | yes | no | String | The alarm content.<br>Example: qwer |
| CurrentValue | yes | yes | String | Current value.<br>Example: 10 |
| FirstOccurTime | yes | yes | String | Initial triggered time.<br>Example: 1735129241 |
| LastOccurTime | yes | yes | String | Final triggered time.<br>Example: 1735129241 |
| Duration | yes | yes | Int64 | Duration.<br>Example: 10 |
| ObjId | yes | yes | String | Object ID.<br>Example: 0a523bfb-dd09-47e2-a85c-0b23455b0949 |
| ObjName | yes | yes | String | Object name.<br>Example: 172.16.0.15#75904 |
| ProjectId | yes | yes | Int64 | Project ID.<br>Example: 10 |
| ProjectName | yes | yes | String | Project name.<br>Example: Default |
| Region | yes | yes | String | Region.<br>Example: gz |
| Status | yes | yes | Int64 | Status. Valid values: 0, 1, 2, 3, 4, 5. 0: not recovered. 1: recovered. 4: expired. 2/3/5: insufficient data.<br>Example: 1 |
| ViewName | yes | yes | String | View name.<br>Example: cvm_device |

## DescribeAccidentConfigAccidentTypeInfo

Event type of the DescribeAccidentConfig API.

Referenced by the following interfaces: DescribeAccidentConfig

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| AccidentId | yes | no | Int64 | Platform event ID.<br>Example: 1 |
| AccidentName | yes | no | String | Platform event name.<br>Example: qwer |
| AccidentEnName | yes | no | String | Platform event name in English.<br>Example: qwer |

## MiniDashboardAlarmInfo

MiniDashboard AlarmInfo

Referenced by the following interfaces: DescribeMiniDashboardAlarmInfo

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| Dimension | no | no | String | Dimension information<br>Example: {"name":"instanceId","value":"1"} |
| NoPolicy | no | no | Bool | No Alarm Policy<br>Example: true |
| DefaultPolicyWithoutNotice | no | no | String | Default Policy without Notification Rule<br>Example: policy-abcdef |

## DashboardMetricCondition

Query condition for the Dashboard2.0 metric query.

Referenced by the following interfaces: DescribeDashboardMetricData

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| Dimension | yes | no | Array of String | Array elements are JSON serialized strings.<br>Example: {"unInstanceId":"ins-00jvv9mo"} |
| Region | yes | no | String | Domain name<br>Example: gz |

### PCLMDescribeMetricSetsData

CLMDescribeMetricSets return data.

Referenced by the following interfaces: CLMDescribeMetricSets

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| MetricSets | yes | yes | Array of PCLMMetricSet | Metric set list data.<br>Example:View |
| TotalCount | yes | yes | Int64 | Total number.<br>Example: 10 |

### DescribePolicyGroupInfoCondition

Alarm threshold conditions output by the policy query.

Referenced by the following interfaces: DescribePolicyGroupInfo, DescribePolicyGroupList

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| MetricShowName | yes | no | String | Metric name.<br>Example: CpuUsage |
| Period | yes | no | Int64 | Data aggregation period in seconds.<br>Example: 60 |
| MetricId | yes | no | Int64 | Metric ID.<br>Example: 10 |
| RuleId | yes | no | Int64 | Threshold rule ID.<br>Example: 10 |
| Unit | yes | no | String | Metric unit.<br>Example: % |
| AlarmNotifyType | yes | no | Int64 | Alarm sending and converging type. The value 0 indicates that alarms are sent consecutively. The value 1 indicates that alarms are sent exponentially.<br>Example: 10 |
| AlarmNotifyPeriod | yes | no | Int64 | Alarm sending period in seconds. The value u003c0 indicates that no alarm will be triggered. The value 0 indicates that an alarm is triggered only once. The value >0 indicates that an alarm is triggered at the interval of triggerTime.<br>Example: 10 |
| CalcType | yes | no | Int64 | Comparative type. The value 1 indicates greater than. The value 2 indicates greater than or equal to. The value 3 indicates smaller than. The value 4 indicates smaller than or equal to. The value 5 indicates equal to. The value 6 indicates not equal to. The value 7 indicates day-on-day increase. The value 8 indicates day-on-day decrease. The value 9 indicates week-on-week increase. The value 10 indicates week-on-week decrease. The value 11 indicates periodical increase. The value 12 indicates periodical decrease.<br>Example: 10 |
| CalcValue | yes | no | String | Detection threshold.<br>Example: 100 |
| ContinueTime | yes | no | Int64 | Duration at which an alarm will be triggered in seconds.<br>Example: 50 |

### MonitorDimensionSource

Dimensio Key Sources.

Referenced by the following interfaces: DescribeDashboardDimensionKeys

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| Namespace | yes | no | String | Namespace.<br>Example: QCE/CVM |
| Metric | yes | no | String | List of metrics.<br>Example: CpuUsage |
| SearchValue | yes | no | String | The values.<br>Example: qwer |

### TemplateTag

Output parameter of DescribeTemplateTags: list of tags for template variables.

Referenced by the following interfaces: DescribeTemplateTags

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| Tag | yes | no | String | Tag name.<br>Example: qwer |
| TagType | yes | no | String | Tag type.<br>Example: qwer |
| ProductId | yes | no | String | Associated product ID (basic monitoring).<br>Example: qwer |

### IdcInfoData

IdcInfoData

Referenced by the following interfaces: DescribeIdcServerCount

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| Data | yes | no | Array of IdcInfo | Return data.<br>Example:View |
| TotalCount | yes | no | Int64 | Total number.<br>Example: 10 |

### MonitorSelectedDimension

Dimension selected.

Referenced by the following interfaces: DescribeDashboardDimensionValues

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| DimensionName | no | no | String | Dimension name.<br>Example: Testing |
| DimensionValue | no | no | String | The value of the dimension.<br>Example: 100 |

### MetricDatum

Metric names and values

Referenced by the following interfaces: PutMonitorData

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| MetricName | yes | no | String | Metric name.<br>Example: CpuUsage |
| Value | yes | no | Uint64 | Metric value.<br>Example: 10 |

### MetricObjectMeaning

Meaning of metric data

Referenced by the following interfaces: DescribeBaseMetrics

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| En | yes | no | String | Meaning of metric data (English).<br>Example: qwer |
| Zh | yes | no | String | Meaning of metric data.<br>Example: qwer |

### MetricTooltip

Rich text prompt for the metric.

Referenced by the following interfaces: DescribeAlarmEvents, DescribeAlarmHistories, DescribeAlarmMetrics, DescribeMiniDashboardAlarmInfo

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| Message | yes | no | String | Text command.<br>Example: This metric indicates the CPU utilization of this instance. |
| URL | yes | no | String | Details link.<br>Example: https://convergecloud.com/ |
| UrlName | yes | no | String | Link name.<br>Example: View Details |

### PCLMStatisticFilterRule

Metric filter condition in log monitoring metric set configurations.

Referenced by the following interfaces: CLMDescribeMetricSets

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| MetricItemId | no | yes | Int64 | Associated metric item ID (only for output parameter; leave it empty for input parameter).<br>Example: 1 |
| CreatedAt | no | yes | String | Creation time (only for output parameter; leave it empty for input parameter).<br>Example: qwer |
| UpdatedAt | no | yes | String | Update time (only for output parameter; leave it empty for input parameter).<br>Example: qwer |
| Relation | yes | no | Int64 | Relationship. 1: AND. 2: OR.<br>Example: 1 |
| Field | yes | no | String | Field name.<br>Example: timeField |
| Operator | yes | no | String | Filter operator: Numeric fields supoort: gt lt ge le eq ne in. Character fields support: eq ne in contains.<br>Example: = |
| Value | yes | no | String | Filter operation value.<br>Example: 10 |

## AttributeValueInfoOutputData

AttributeValueInfoOutputData

Referenced by the following interfaces: DescribeServerData, DescribeViewData

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Data | yes | no | Array of AttributeValueInfoOutput | Attribute report data list.<br>Example:View |
| TotalCount | yes | no | Int64 | Number of attribute data lists.<br>Example: 10 |

## ServerAttributeDataOutput

ServerAttributeDataOutput

Referenced by the following interfaces: DescribeServerDatas

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| AttributeId | yes | no | Uint64 | Metric ID.<br>Example: 1 |
| ServerId | yes | no | Uint64 | Server ID.<br>Example: 1 |
| Values | yes | no | Array of AttributeTimestampValueOutput | Reported value of the metric.<br>Example:View |

## DashboardDataSource

Dashboard2.0 data source information.

Referenced by the following interfaces: DescribeDashboardDataSources

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| UUID | yes | no | String | DataSource unique identifier.<br>Example: u78haeoyaq8w8wbr |
| Name | yes | no | String | Data source name.<br>Example: Testing |
| Type | yes | no | String | Data source type, such as MySQL, Prometheus.<br>Example: MySQL |
| TypeLogoUrl | yes | yes | String | Data source icon.<br>Example: http://test.com |
| Access | yes | yes | String | Access mode.<br>Example: mode |
| Url | yes | yes | String | Data source access address<br>Example: http://test.com |
| Password | yes | no | String | Password. It can be an empty string.<br>Example: 123456 |
| User | yes | no | String | Username. It can be an empty string.<br>Example: qwer |
| Database | yes | no | String | Database.<br>Example: mysql |
| BasicAuth | yes | no | Bool | It is "false" by default.<br>Example: false |
| IsDefault | yes | no | Bool | Whether it is the default data source.<br>Example: false |
| JsonData | yes | yes | String | Additional information of the storage data source, such as ES link information "{ "esVersion": 5, "logLevelField": "","logMessageField": " ", "maxConcurrentShardRequests": 256, "timeField": "@timestamp"}".<br>Example: { "esVersion": 5, "logLevelField": "","logMessageField": "", "maxConcurrentShardRequests": 256, "timeField": "@timestamp"} |
| ReadOnly | yes | no | Bool | Whether the data source data is read-only.<br>Example: false |

## BindingPolicyObjectDimension

Dimensions of instances bound to a policy

Referenced by the following interfaces: BindingPolicyObject, CreateInstanceGroup, DescribeInstanceGroup, ModifyInstanceGroup, ShieldPolicyAlarm

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| RegionId | no | no | Int64 | Region ID.<br>Example: 1 |
| Region | yes | no | String | Region name.<br>Example: gz |
| Dimensions | no | no | String | Dimension information<br>Example: {"unInstanceId":"ins-00jxv9mo"} |
| EventDimensions | no | no | String | Event dimension information.<br>Example: {"uuid":"c734cd0d-d050-44bd-8d17-f6381039b4b6"} |

## DescribePolicyGroupInfoConditionTpl

Template-based policy group information output by the policy query

Referenced by the following interfaces: DescribePolicyGroupInfo, DescribePolicyGroupList

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| GroupId | yes | no | Int64 | Policy group ID.<br>Example: 1234 |
| GroupName | yes | no | String | Policy group name.<br>Example: Testing |
| ViewName | yes | no | String | Policy type<br>Example: cvm_device |
| Remark | yes | no | String | Policy group remarks.<br>Example: test |
| LastEditUin | yes | no | String | UIN of the last user who modified it.<br>Example: 123456 |
| UpdateTime | yes | yes | Int64 | Update time.<br>Example: 1735095122 |
| InsertTime | yes | yes | Int64 | Creation time.<br>Example: 1735095122 |
| IsUnionRule | yes | yes | Int64 | Whether it is an AND rule.<br>Example: 1 |

## DescribePolicyGroupListGroupInstanceGroup

Instance group that is bound to a policy group of the DescribePolicyGroupList API

Referenced by the following interfaces: DescribePolicyGroupList

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| InstanceGroupId | yes | no | Int64 | Instance group name ID.<br>Example: 1234 |
| ViewName | yes | no | String | Policy type view name.<br>Example: cvm_device |
| LastEditUin | yes | no | String | UIN of the last editor.<br>Example: 123456 |
| GroupName | yes | no | String | Instance group name.<br>Example: Testing |
| InstanceSum | yes | no | Int64 | Number of instances.<br>Example: 10 |
| UpdateTime | yes | no | Int64 | Update time.<br>Example: 1735095122 |
| InsertTime | yes | no | Int64 | Creation time.<br>Example: 1735095122 |

## PolicyTag

Policy tag.

Referenced by the following interfaces: BindingPolicyTag, DescribeGroupIDByTags, UnbindingPolicyTag

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Key | yes | no | String | Tag key.<br>Example: qwer |
| Value | yes | no | String | Tag value.<br>Example: qwer |

## DescribePolicyConditionListConfigManualCalcType

DescribePolicyConditionList.ConfigManual.CalcType

Referenced by the following interfaces: DescribePolicyConditionList

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Keys | yes | yes | Array of Int64 | Value of CalcType.<br>Example: 1 |
| Need | yes | no | Bool | Whether required.<br>Example: true |

## ClmMetricAnalysisBasicMetric

Basic metric parameters in CLM metric analysis.

Referenced by the following interfaces: DescribeMetricAnalysisData

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Operator | yes | no | String | Operator.<br>Example: = |
| Name | yes | no | String | Metric name.<br>Example: Testing |

## Condition

Alarm condition

Referenced by the following interfaces: DescribeConditionsTemplateList, DescribePolicyInfoByInstance, DescribeRecommendedTemplate

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| AlarmNotifyPeriod | yes | no | Int64 | Alarm notification frequency.<br>Example: 7200 |
| AlarmNotifyType | yes | no | Int64 | Predefined repeated notification policy. 0: One-time alarm; 1: exponential alarm; 2: consecutive alarm.<br>Example: 0 |
| CalcType | yes | yes | String | Test method.<br>Example: "1" |
| CalcValue | yes | yes | String | Test value.<br>Example: "95" |
| ContinueTime | yes | yes | String | Duration.<br>Example: "300" |
| MetricID | yes | no | Int64 | Metric ID.<br>Example: 33 |
| MetricDisplayName | yes | no | String | Displayed metric name.<br>Example: "CPU utilization" |
| Period | yes | no | Int64 | Interval.<br>Example: 60 |
| RuleID | yes | no | Int64 | Rule ID.<br>Example: 623123 |
| Unit | yes | no | String | Metric unit.<br>Example: "%" |

## TransLogItem

Changelog returned by DescribeTransLog.

Referenced by the following interfaces: DescribeTransLog

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Id | yes | yes | Int64 | Changelog ID.<br>Example: 1 |
| ModuleId | yes | yes | Int64 | Module ID.<br>Example: 1 |
| LogData | yes | yes | String | Log information. JSON string.<br>Example: qwer |
| CreateTime | yes | yes | Int64 | Creation time.<br>Example: 1735179378 |
| LastEditUin | yes | yes | String | UIN of the last user who modified it.<br>Example: 123456 |

## IdcData

IdcData

Referenced by the following interfaces: DescribeIdcTree

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Id | yes | no | Int64 | idc id, which represents zoneId on the cloud platform.<br>Example: 1 |
| Name | yes | no | String | idc id, which represents zone name on the cloud platform.<br>Example: Testing |

## TemplateGroup

Template list

Referenced by the following interfaces: DescribeConditionsTemplateList

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Conditions | yes | yes | Array of Condition | Metric alarm rule.<br>Example:View |
| EventConditions | yes | yes | Array of EventCondition | Event alarm rule.<br>Example:View |
| PolicyGroups | yes | yes | Array of PolicyGroup | Associated alarm policy group.<br>Example:View |
| GroupID | yes | no | Int64 | Template-based policy group ID.<br>Example: 1 |
| GroupName | yes | no | String | Template-based policy group name.<br>Example: test |
| InsertTime | yes | no | Int64 | Creation time.<br>Example: 11 |
| LastEditUin | yes | no | Int64 | UIN of the last user who modified it.<br>Example: 1 |
| Remark | yes | no | String | Description.<br>Example: test |
| UpdateTime | yes | no | Int64 | Update time.<br>Example: 1 |
| ViewName | yes | no | String | View.<br>Example: cvm_device |
| IsUnionRule | yes | no | Int64 | Whether the AND rule is used.<br>Example: 1 |

## AttributeIdOutput

AttributeIdOutput

Referenced by the following interfaces: ModifyAttribute

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| AttributeId | yes | yes | Uint64 | Attribute ID.<br>Example: 10 |

## DescribePolicyQuotaV3ResponseDataQuotaList

DescribePolicyQuotaV3ResponseDataQuotaList

Referenced by the following interfaces: DescribePolicyQuota

| name | required | NULL allowed | type | description |
|---|---|---|---|---|

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| Used | yes | no | Int64 | User.<br>Example: 10 |
| Total | yes | no | Int64 | Total number.<br>Example: 10 |
| ProjectID | yes | no | Int64 | Project ID.<br>Example: 10 |

## DescribePolicyQuotaV3ResponseData

DescribePolicyQuotaV3ResponseData

Referenced by the following interfaces: DescribePolicyQuota

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| ViewName | yes | no | String | View.<br>Example: cvm_device |
| IsQuotaByProject | yes | no | Bool | Whether it is a project quota.<br>Example: true |
| QuotaList | yes | yes | Array of DescribePolicyQuotaV3ResponseDataQuotaList | List of quotas.<br>Example:View |
| Quota | yes | yes | DescribePolicyQuotaV3ResponseDataQuota | The quota.<br>Example:View |

## DescribeBaseMetricsDataPeriod

DescribeBaseMetricsDataPeriod

Referenced by the following interfaces: DescribeBaseMetricsForConsoleFontEnd

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| Period | yes | no | Uint64 | Interval.<br>Example: 60 |
| StatType | yes | yes | Array of String | Statistical method.<br>Example: ["avg"] |

## AlarmHistoryMetric

Metric information of alarm records

Referenced by the following interfaces: DescribeAlarmHistories, DescribeMiniDashboardAlarmInfo

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| QceNamespace | yes | no | String | Queries the namespace used by the data.<br>Example: qce/cvm |
| MetricName | yes | no | String | Metric name.<br>Example: CpuUsage |
| Period | yes | no | Int64 | Statistical period.<br>Example: 60 |
| Value | yes | no | String | Value that triggers the alarm.<br>Example: 10 |
| Tooltip | yes | yes | MetricTooltip | Explanation of the metric.<br>Example:View |
| Description | yes | no | String | Display name of the metric.<br>Example: CPU Utilization |

## DescribeAbnormalObjectsEventObject

Product event alarm exception object returned by the DescribeAbnormalObjects API.

Referenced by the following interfaces: DescribeAbnormalObjects

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| Status | yes | no | Int64 | Alarm status.<br>Example: 1 |
| Content | yes | no | String | The alarm content.<br>Example: qewrrq |
| InstanceId | yes | no | String | Instance ID.<br>Example: ins-00jvv9mo |
| Dimensions | yes | no | String | Dimension combination. JSON string.<br>Example: {"unInstanceId":"ins-xxx"} |
| AbnormalStatus | yes | no | String | Alarm status: "abnormal" indicates that the alarm has not been recovered, and "warning" indicates that the alarm has been recovered.<br>Example: abnormal |
| FirstOccurTime | yes | no | String | Time of first occurrence.<br>Example: 1735041205 |
| LastOccurTime | yes | no | String | Time of last occurrence.<br>Example: 1735041205 |
| UniqueId | yes | no | String | Dimension combination md5.<br>Example: 1314edae41314 |
| GroupId | yes | yes | Int64 | Policy group ID.<br>Example: 1234 |
| GroupName | yes | yes | String | Policy group name.<br>Example: Testing |
| ProjectId | yes | yes | Int64 | Project ID.<br>Example: 1 |
| ProductCName | yes | no | String | The name of the product.<br>Example: Cloud Virtual Machine |
| ProductEName | yes | no | String | English name of the product.<br>Example: cvm |
| EventEName | yes | no | String | Event name.<br>Example: ReadOnly |
| EventCName | yes | no | String | The name of the event.<br>Example: Disk read-only |
| EventId | yes | no | Int64 | Product event ID.<br>Example: 1 |
| Region | yes | no | String | Region.<br>Example: gz |
| EventUniqueId | yes | no | String | Product dimension combination md5.<br>Example: eqeq-czda |

## MessagePolicy

MessagePolicy

Referenced by the following interfaces: DescribeMsgPolicyList

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| PolicyID | yes | no | String | Policy ID.<br>Example: policy-abc123 |
| PolicyName | yes | no | String | Policy name.<br>Example: Testing |
| ReceiverGroupIds | yes | yes | Array of Int64 | Recipient group list.<br>Example: [1234] |
| NotifyWay | yes | yes | Array of String | Notification method.<br>Example: sms |
| VoiceConfig | yes | yes | DescribeMsgPolicyListVoiceConfig | Voice configurations.<br>Example:View |
| IsDefault | yes | no | Bool | Whether it is a default policy.<br>Example: true |
| AlarmCount | yes | no | Int64 | Number of alarms.<br>Example: 1 |

## AlarmPolicyEventCondition

Event trigger condition of alarm policy

Referenced by the following interfaces: CreateAlarmPolicy, DescribeAlarmPolicies, DescribeAlarmPolicy, ModifyAlarmPolicyCondition

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| Rules | yes | yes | Array of AlarmPolicyRule | Alarm trigger condition list.<br>Example:View |

## DescribeGraphDataPartition

DescribeGraphDataPartition

Referenced by the following interfaces: DescribeGraphData

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Name | yes | no | Array of String | Partition name.<br>Example: Testing |
| Usage | yes | no | Float | Usage.<br>Example: 1234 |
| Total | yes | no | Float | Total.<br>Example: 10 |

### CCMChartEntry

CCMChartEntry

Referenced by the following interfaces: DescribeCCMCharts

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| ChartId | yes | no | Int64 | Monitoring chart ID.<br>Example: 1 |
| ViewId | yes | no | Int64 | Group view ID.<br>Example: 1 |
| ChartName | yes | no | String | Name of the monitoring chart.<br>Example: test |
| ChartType | yes | no | Int64 | Type of the monitoring chart. 1 Detail view, 2 Aggregate view.<br>Example: 1 |
| ProductType | yes | no | Int64 | Product type. 1 Basic monitoring, 2 Custom monitoring.<br>Example: cvm |
| AttributeId | yes | no | Int64 | Metric ID.<br>Example: 1 |
| Aggregation | yes | no | Int64 | Aggregation method. 0 SUM, 1 AVG, 2 MAX, 3 MIN<br>Example: 1 |
| Description | yes | no | String | Description.<br>Example: test |
| CreatorId | yes | no | Uint64 | Creator ID.<br>Example: 1 |
| CreateTime | yes | no | String | Creation time.<br>Example: 1735041205 |
| UpdaterId | yes | no | Uint64 | Updater ID.<br>Example: 1 |
| UpdateTime | yes | no | String | Update time.<br>Example: 1735041205 |
| AttributeName | yes | yes | String | Metric name.<br>Example: Testing |
| AttributeUniqueName | yes | yes | String | Unique English name of the metric.<br>Example: qwer_123 |
| UnitName | yes | yes | String | Metric unit name.<br>Example: test |

### DescribePolicyConditionListConfigManualPeriod

DescribePolicyConditionList.ConfigManual.Period

Referenced by the following interfaces: DescribePolicyConditionList

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Default | yes | yes | Int64 | Default period in seconds.<br>Example: 10 |
| Keys | yes | yes | Array of Int64 | Custom period in seconds.<br>Example: 10 |
| Need | yes | no | Bool | Whether required.<br>Example: true |

### DescribeEventPolicyConfigDimension

DescribeEventPolicyConfigDimension

Referenced by the following interfaces: DescribeEventPolicyConfig

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Name | yes | no | String | The name of the dimension.<br>Example: Testing |
| Key | yes | no | String | Dimension key value.<br>Example: qwer |

### GroupIdTag

1.0 policy ID, tag.

Referenced by the following interfaces: DescribeGroupIDByTags

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| GroupIDList | yes | no | Array of String | 1.0 policy ID.<br>Example: ["123456"] |
| Key | yes | no | String | Tag key.<br>Example: qwer |
| Value | yes | no | String | Tag value.<br>Example: qwer |

### ClmDescribeAlertPoliciesData

CLM alarm policy list return structure.

Referenced by the following interfaces: DescribeAlertPolicies

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| TotalCount | yes | yes | Int64 | Total number.<br>Example: 10 |
| AlertPolicies | yes | yes | Array of ClmAlertPolicy | List element.<br>Example:View |

### DescribePolicyUseList

Object returned by the DescribePolicyUseList API.

Referenced by the following interfaces: DescribePolicyUseList

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| UniqueId | yes | no | String | Unique ID of the alarm object.<br>Example: eqeq-czda |
| Dimensions | yes | no | String | Dimension string.<br>Example: {"unInstanceId":"ins-xxx"} |

### DescribeAlarmCallbackHistory

Historical callback information.

Referenced by the following interfaces: DescribeAlarmCallbackHistory

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Url | yes | no | String | Callback address<br>Example: http://test.com |
| ValidFlag | yes | no | Int64 | Whether the callback is validated.<br>Example: 1 |
| VerifyCode | yes | no | String | Callback address verification code.<br>Example: 1131ad |

### StrategyIdInfo

StrategyIdInfo

Referenced by the following interfaces: CreateStrategy

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| StrategyId | yes | no | Int64 | The ID of the alarm policy.<br>Example: 1 |

### StrategyChannelInfo

StrategyChannelInfo

Referenced by the following interfaces: ModifyStrategyChannels

| name | required | NULL allowed | type | description |
|---|---|---|---|---|

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| StrategyId | yes | no | Uint64 | The ID of the alarm policy.<br>Example: 10 |
| AlarmChannel | no | no | Array of String | Alarm channel list.<br>Example: sms |

### ClmAnalysisFilter

Filter condition in CLM multidimensional analysis.

Referenced by the following interfaces: DescribeDimensionAnalysisData, DescribeMetricAnalysisData

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| Field | yes | no | String | Field name.<br>Example: timeField |
| Operator | yes | no | String | Operator. Only =<br>is supported (when the value array contains multiple values, it is an "in" relationship).<br>Example: = |
| Value | yes | no | Array of String | Value.<br>Example: 50 |

### NoticeContentTmplItem

Elements in the notification content template.

Referenced by the following interfaces: CreateNoticeContentTmpl, DescribeNoticeContentTmpl, ModifyNoticeContentTmpl

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| CustomCallback | no | no | Array of CustomCallbackTmplMatcher | Custom notification channel configuration.<br>Example:View |

### AttributeUnitInfoOutputData

AttributeUnitInfoOutputData

Referenced by the following interfaces: DescribeAttributeUnits

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| Data | yes | no | Array of AttributeUnitInfoOutput | Attribute unit data list.<br>Example:View |
| TotalCount | yes | no | Int64 | Number of attribute unit data lists.<br>Example: 10 |

### IdcInfo

IdcInfo

Referenced by the following interfaces: DescribeIdcServerCount

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| IdcId | yes | no | Int64 | Region ID.<br>Example: 12345 |
| ServerCount | yes | no | Int64 | Number of instances.<br>Example: 10 |
| IdcName | yes | no | String | The region name.<br>Example: Testing |

### PCLMMetricCustomItem

Compound metric configurations for the log monitoring metric set.

Referenced by the following interfaces: CLMDescribeMetricSets

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| Formula | yes | no | String | Expression.<br>Example: {"cluster_name":"es-dnxsssss"} |
| Name | yes | no | String | Metric name.<br>Example: Testing |
| Desc | yes | no | String | Metric description name.<br>Example: Testing |
| MetricSetCID | no | no | String | Metric set CID (only for output parameter; leave it empty for input parameter).<br>Example: cid-xxx |
| CreatedAt | no | no | String | Creation time (only for output parameter; leave it empty for input parameter)<br>Example: qwer |
| UpdatedAt | no | no | String | Update time (only for output parameter; leave it empty for input parameter)<br>Example: qwer |

### PolicyGroup

Policy group information

Referenced by the following interfaces: DescribeConditionsTemplateList

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| CanSetDefault | yes | no | Bool | Whether the policy can be set as the default policy.<br>Example: false |
| GroupID | yes | no | Int64 | Alarm policy group ID.<br>Example: 1234 |
| GroupName | yes | no | String | Alarm policy group name.<br>Example: Testing |
| InsertTime | yes | no | Int64 | Creation time.<br>Example: 1735179378 |
| IsDefault | yes | no | Int64 | Whether it is a default alarm policy.<br>Example: 1 |
| Enable | yes | no | Bool | Enablement status of the alarm policy.<br>Example: false |
| LastEditUin | yes | no | Int64 | UIN of the last user who modified it.<br>Example: 123456 |
| NoShieldedInstanceCount | yes | no | Int64 | Number of unshielded instances.<br>Example: 10 |
| ParentGroupID | yes | no | Int64 | Parent policy group ID.<br>Example: 1234 |
| ProjectID | yes | no | Int64 | Project ID.<br>Example: 1 |
| ReceiverInfos | yes | yes | Array of PolicyGroupReceiverInfo | Information about the alarm recipient.<br>Example:View |
| Remark | yes | no | String | Remarks.<br>Example: Testing |
| UpdateTime | yes | no | Int64 | Modification time.<br>Example: 1735179378 |
| TotalInstanceCount | yes | no | Int64 | Total number of bound instances.<br>Example: 10 |
| ViewName | yes | no | String | View.<br>Example: cvm_device |
| IsUnionRule | yes | no | Int64 | Whether the AND rule is used.<br>Example: 1 |

### TransLogInstance

List of instances returned by DescribeTransLog.

Referenced by the following interfaces: DescribeInstanceGroup

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| Dimensions | yes | yes | String | Dimension of the instance.<br>Example: qwer |
| UniqueId | yes | yes | String | Unique ID of the instance.<br>Example: eqeq-czda |
| IsShielded | yes | yes | Int64 | Shielding status. 1: shielding enabled. 0: no shielding.<br>Example: 1 |
| Region | yes | yes | String | Region.<br>Example: gz |

### AttributeUnitInfoOutput

AttributeUnitInfoOutput

Referenced by the following interfaces: DescribeAttributeUnits

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| UnitId | yes | no | Uint64 | Metric unit ID.<br>Example: 1 |
| UnitName | yes | no | String | Metric unit name.<br>Example: test |

## Contact

Recipient information returned by the DescribeContactList API.

Referenced by the following interfaces: DescribeContactList

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| Uid | yes | no | Int64 | uid<br>Example: 1 |
| Uin | yes | no | String | uin<br>Example: 123456 |
| Name | yes | no | String | Recipient name.<br>Example: Testing |
| CanLogin | yes | no | Int64 | Whether you can log in. 0: you can't log in. 1: you can log in.<br>Example: 1 |
| PhoneFlag | yes | no | Int64 | Whether the mobile phone number is verified.<br>Example: 1 |
| PhoneNum | yes | no | String | The mobile phone.<br>Example: 10 |
| EmailFlag | yes | no | Int64 | Whether the email is verified.<br>Example: 1 |
| Email | yes | no | String | Email.<br>Example: 123@com |
| IsReceiverOwner | yes | no | Int64 | Whether it is the person in charge of receiving.<br>Example: 123456 |
| OwnerUid | yes | no | Int64 | ownerUid<br>Example: 123456 |

## DescribeAbnormalObjectsThresholdObject

Metric alarm exception object returned by the DescribeAbnormalObjects API.

Referenced by the following interfaces: DescribeAbnormalObjects

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| Id | yes | no | Int64 | Alarm ID.<br>Example: 1 |
| RuleId | yes | no | Int64 | Alarm rule ID.<br>Example: 10 |
| GroupId | yes | no | Int64 | Policy group ID.<br>Example: 1234 |
| GroupName | yes | no | String | Policy group name.<br>Example: Testing |
| ProjectId | yes | yes | Int64 | Project ID.<br>Example: 1 |
| Status | yes | no | Int64 | Alarm status.<br>Example: 1 |
| Content | yes | no | String | The alarm content.<br>Example: qwer |
| FirstOccurTime | yes | no | String | Time of first occurrence.<br>Example: 1735041205 |
| LastOccurTime | yes | no | String | Time of last occurrence.<br>Example: 1735041205 |
| Dimensions | yes | no | String | Dimension combination. JSON string.<br>Example: {"unInstanceId":"ins-xxx"} |
| UniqueId | yes | no | String | Dimension combination md5.<br>Example: eqeq-czda |
| AbnormalStatus | yes | no | String | Alarm status: "abnormal" indicates that the alarm has not been recovered, and "warning" indicates that the alarm has been recovered.<br>Example: abnormal |
| InstanceId | yes | no | String | Object ID.<br>Example: ins-00jvv9mo |

## DescribeStorageDurationData

DescribeStorageDurationData

Referenced by the following interfaces: DescribeStorageDuration

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| Period | yes | no | Int64 | Interval.<br>Example: 5 |
| Duration | yes | no | Int64 | Duration.<br>Example: 10 |

## ClmDimension

Metric set dimension object for log monitoring.

Referenced by the following interfaces: CreateMetricSet, DescribeMetricSet, DescribeMetricSets, ModifyMetricSet

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| Field | yes | no | String | Field name.<br>Example: timeField |
| Id | no | no | Int64 | Item ID (only for output parameter; leave it empty for input parameter)<br>Example: 1 |
| CreatedAt | no | no | String | Creation time (only for output parameter; leave it empty for input parameter)<br>Example: eqeadfrqc |
| UpdatedAt | no | no | String | Update time (only for output parameter; leave it empty for input parameter)<br>Example: eqeadfrqc |
| MetricSetId | no | no | Int64 | Metric set ID (only for output parameter; leave it empty for input parameter)<br>Example: 1 |

## URLNotice

Cloud Monitor alarm notification template - callback notification details

Referenced by the following interfaces: CreateAlarmNotice, DescribeAlarmNotice, DescribeAlarmNoticeCallbacks, DescribeAlarmNotices, DescribeAlarmPolicies, DescribeAlarmPolicy, ModifyAlarmNotice

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| URL | yes | yes | String | Callback URL, which can contain up to 256 characters.<br>Example: http://www.123.com |
| IsValid | no | yes | Int64 | Whether verification is passed. Valid values: 0 (no), 1 (yes).<br>Example: 1 |
| ValidationCode | no | yes | String | Verification code.<br>Example: 82hfBt |

## AlarmPolicyCondition

Metric trigger condition of alarm policy

Referenced by the following interfaces: CreateAlarmPolicy, DescribeAlarmPolicies, DescribeAlarmPolicy, ModifyAlarmPolicyCondition

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| IsUnionRule | yes | no | String | The 'AND' or 'OR' rule of the metric. The value 0 indicates the 'OR' rule. The value 1 indicates the 'AND' rule.<br>Example: 1 |
| Rules | yes | no | Array of AlarmPolicyRule | Alarm trigger condition list.<br>Example:View |

## RegionPolicyObjectCount

Region statistics returned by the DescribePolicyObjectCount API.

Referenced by the following interfaces: DescribePolicyObjectCount

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| Region | yes | no | String | Region.<br>Example: gz |
| Count | yes | no | Int64 | Number of instances bound.<br>Example: 10 |

## DescribeBindingPolicyObjectListInstanceGroup

Instance group information returned by the DescribeBindingPolicyObjectList API

Referenced by the following interfaces: DescribeBindingPolicyObjectList

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| InstanceGroupId | yes | no | Int64 | Instance group ID.<br>Example: 1234 |
| ViewName | yes | no | String | Name of the alarm policy type.<br>Example: cvm_device |
| LastEditUin | yes | no | String | Uin that was last edited.<br>Example: 123456 |
| GroupName | yes | no | String | Instance group name.<br>Example: Testing |
| InstanceSum | yes | no | Int64 | Number of instances.<br>Example: 10 |
| UpdateTime | yes | no | Int64 | Update time.<br>Example: 1735095122 |
| InsertTime | yes | no | Int64 | Creation time.<br>Example: 1735095122 |
| Regions | yes | yes | Array of String | Regions where the instances reside.<br>Example: ["gz"] |

## DescribePolicySituationAlarmSituation

DescribePolicySituationAlarmSituation

Referenced by the following interfaces: DescribePolicySituation

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| AlarmRuleCount | yes | no | Int64 | Number of alarm rules.<br>Example: 10 |
| AlarmTouchCount | yes | no | Int64 | Number of alarm deliveries.<br>Example: 10 |

## PeriodsSt

Statistical method during the period

Referenced by the following interfaces: DescribeBaseMetrics

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Period | yes | no | String | Interval.<br>Example: 60 |
| StatType | yes | no | Array of String | Statistical method.<br>Example: avg |

## DescribeDashboardViewList

DescribeDashboardViewList

Referenced by the following interfaces: DescribeDashboardViews

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| ViewID | yes | no | Int64 | View ID.<br>Example: 1 |
| DescName | yes | no | String | Description name.<br>Example: Testing |
| Namespace | yes | no | String | Namespace.<br>Example: QCE/CVM |
| MetricName | yes | yes | Array of String | List of metric names.<br>Example: CpuUsage |
| Meta | yes | no | String | Metadata JSON.<br>Example: {"aggregateType":"detail","aggregations":["Avg","Max","Min"],"chartTypes":["column"],"configId":"cvm","layout":{"h":"5","h2":"0","w":"4","x":"0","y":"0"},"timeAggregate":"last"} |
| Instances | yes | yes | Array of String | Instance list.<br>Example: ["ins-xxx"] |

## WebStorageInfos

Dashboard pop-up information management.

Referenced by the following interfaces: DescribeWebStorage

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| CreatedAt | yes | no | String | Creation time.<br>Example: qwer |
| SubUin | yes | no | String | Sub-user.<br>Example: 123456 |
| Uin | yes | no | String | Primary user.<br>Example: 123456 |
| UpdatedAt | yes | no | String | Update time.<br>Example: qwer |
| WebStorageInfo | yes | no | String | Pop-up information.<br>Example: qwer |

## ClmMetricBasicItem

Basic metric configurations for the log monitoring metric set.

Referenced by the following interfaces: CreateMetricSet, DescribeMetricSet, DescribeMetricSets, ModifyMetricSet

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Field | yes | yes | String | Field name.<br>Example: timeField |
| Operator | yes | yes | String | Metric statistics method: sum, count, min, max.<br>Example: = |
| Name | yes | yes | String | Metric name.<br>Example: Testing |
| Desc | yes | yes | String | Metric description name.<br>Example: Testing |
| StatisticFilterRules | no | yes | Array of ClmStatisticFilterRule | Metric filtering rule.<br>Example:View |
| Id | no | yes | Int64 | Item ID (only for output parameter; leave it empty for input parameter)<br>Example: 1 |
| MetricSetId | no | yes | Int64 | Metric set ID (only for output parameter; leave it empty for input parameter)<br>Example: 1 |
| CreatedAt | no | yes | String | Creation time (only for output parameter; leave it empty for input parameter)<br>Example: eqeadfrqc |
| UpdatedAt | no | yes | String | Update time (only for output parameter; leave it empty for input parameter)<br>Example: eqeadfrqc |

## MoveFailReason

Reasons why the dashboard failed to be moved.

Referenced by the following interfaces: MoveUnifyDashboards

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| UUID | yes | no | String | UUID of the dashboard.<br>Example: daaeqe |
| Msg | yes | no | String | Reasons why the movement failed.<br>Example: ok |

## ModifyPolicyGroupEventCondition

Modification of the event alarm condition passed in by the alarm policy group.

Referenced by the following interfaces: ModifyPolicyGroup

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| RuleId | no | no | Int64 | Rule ID. No filling means new addition while filling in ruleId means to modify existing rules.<br>Example: 12 |
| EventId | yes | no | Int64 | Event ID.<br>Example: 12 |
| AlarmNotifyType | yes | no | Int64 | Alarm sending and converging type. The value 0 indicates that alarms are sent consecutively. The value 1 indicates that alarms are sent exponentially.<br>Example: 1 |
| AlarmNotifyPeriod | yes | no | Int64 | Alarm sending period in seconds. The value u003c0 indicates that no alarm will be triggered. The value 0 indicates that an alarm is triggered only once. The value >0 indicates that an alarm is triggered at the interval of triggerTime.<br>Example: 5 |

## DescribePolicyGroupInfoReceiverInfo

Alarm recipient information output by the policy query

Referenced by the following interfaces: DescribePolicyGroupInfo, DescribePolicyGroupList

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| ReceiverGroupList | yes | no | Array of Int64 | List of alarm recipient group IDs.<br>Example: [1234] |
| ReceiverUserList | yes | no | Array of Int64 | List of alarm recipient IDs.<br>Example: [1234] |
| StartTime | yes | no | Int64 | Alarm period start time. Range: [0,86400). Remove the date after converting the time from UNIX timestamp to UTC+8 time. For example, 7200 means "10:0:0".<br>Example: 1734953008 |
| EndTime | yes | no | Int64 | End time of the alarm period. The meaning is the same as that of StartTime.<br>Example: 1734953808 |
| ReceiverType | yes | no | String | Recipient type. Valid values: group (recipient group) and user (recipient).<br>Example: group |
| NotifyWay | yes | no | Array of String | Alarm notification method. Valid values: "SMS", "SITE", "EMAIL", "CALL", and "WECHAT".<br>Example: sms |
| UidList | yes | yes | Array of Int64 | UID of the alarm call recipient.<br>Example: [1234] |
| RoundNumber | yes | no | Int64 | Number of alarm call rounds.<br>Example: [1234] |
| RoundInterval | yes | no | Int64 | Intervals of alarm call rounds in seconds.<br>Example: 60 |
| PersonInterval | yes | no | Int64 | Alarm call intervals for individuals in seconds.<br>Example: 60 |
| NeedSendNotice | yes | no | Int64 | Whether to send an alarm call delivery notice. The value 0 indicates that no notice needs to be sent. The value 1 indicates that a notice needs to be sent.<br>Example: 0 |
| SendFor | yes | no | Array of String | Alarm call notification time. Valid values: OCCUR (indicating that a notice is sent when the alarm is triggered) and RECOVER (indicating that a notice is sent when the alarm is recovered).<br>Example: OCCUR |
| RecoverNotify | yes | no | Array of String | Notification method when an alarm is recovered. Valid value: SMS.<br>Example: SMS |

## ProductHealthStatus

HealthStatus returned by the DescribeProductHealthStatusList API.

Referenced by the following interfaces: DescribeProductHealthStatusList

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| ViewName | yes | yes | String | View name.<br>Example: cvm_device |
| HealthStatus | yes | yes | Int64 | Health status. 1: not recovered.<br>Example: 1 |
| WarningCount | yes | yes | Int64 | Number of reminders.<br>Example: 1 |
| AbnormalCount | yes | yes | Int64 | Number of exceptions.<br>Example: 1 |
| AlarmRegions | yes | yes | Array of String | List of regions that receive alarms.<br>Example: ["gz"] |

## DescribeProductEventListEvents

Events returned by the DescribeProductEventList API

Referenced by the following interfaces: DescribeProductEventList

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| EventId | yes | yes | Int64 | Event ID.<br>Example: 111 |
| EventCName | yes | yes | String | The name of an event.<br>Example: Unreachable by ping |
| EventEName | yes | yes | String | English name of an event.<br>Example: ping_unreachable |
| EventName | yes | yes | String | Abbreviated name of an event.<br>Example: ping_unreachable |
| ProductCName | yes | yes | String | The name of the product.<br>Example: cvm |
| ProductEName | yes | yes | String | The product name.<br>Example: cvm |
| ProductName | yes | yes | String | Product Name Abbreviation<br>Example: cvm |
| InstanceId | yes | yes | String | Instance ID.<br>Example: ins-3xdfd90 |
| InstanceName | yes | yes | String | Instance name.<br>Example: ins-3xdfd90 |
| ProjectId | yes | yes | String | Project ID.<br>Example: 123 |
| Region | yes | yes | String | Region.<br>Example: gz |
| Status | yes | yes | String | State.<br>Example: 1 |
| SupportAlarm | yes | yes | Int64 | Specifies whether to support alarm notifications.<br>Example: 1 |
| Type | yes | yes | String | Event type.<br>Example: 21 |
| StartTime | yes | yes | Int64 | Start time.<br>Example: 1611560738 |
| UpdateTime | yes | yes | Int64 | Update time.<br>Example: 1611560790 |
| Dimensions | yes | yes | Array of DescribeProductEventListEventsDimensions | Information about an instance object.<br>Example:View |
| AdditionMsg | yes | yes | Array of DescribeProductEventListEventsDimensions | Additional information about an instance object.<br>Example:View |
| IsAlarmConfig | yes | yes | Int64 | Specifies whether to configure alarms.<br>Example: 1 |
| GroupInfo | yes | yes | Array of DescribeProductEventListEventsGroupInfo | Policy information<br>Example:View |

## PCLMDimensionAnalysisResultData

Dimension analysis results data.

Referenced by the following interfaces: DescribeDimensionAnalysisData

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Current | yes | no | String | Dimension data.<br>Example: qwer |
| Granularity | yes | no | Int64 | Granularity.<br>Example: 1 |

## PCLMLogProfileItem

Log monitoring metric set - log data structure description.

Referenced by the following interfaces: CLMDescribeMetricSets

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| MetricSetCID | no | no | String | Associated metric set CID (only for output parameter; leave it empty for input parameter).<br>Example: cid-xxx |
| Field | yes | no | String | Field name.<br>Example: timeField |
| Type | yes | no | String | Field type. Valid values: string, long, double.<br>Example: string |
| CreatedAt | no | no | String | Creation time (only for output parameter; leave it empty for input parameter).<br>Example: qwer |
| UpdatedAt | no | no | String | Update time (only for output parameter; leave it empty for input parameter).<br>Example: qwer |

## DataPoint

Monitoring data point

Referenced by the following interfaces: GetMonitorData

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Dimensions | yes | no | Array of Dimension | Combination of instance object dimensions.<br>Example:View |
| Timestamps | yes | no | Array of Float | Timestamp array that indicates the points in time that have data. A missing timestamp has no data points.<br>Example: 31415513 |

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| Values | yes | no | Array of Float | Metric value array, which corresponds to the values returned for Timestamps.<br>Example: 31314 |

## DescribePolicyGroupInfoEventCondition

Event alarm conditions output by the policy query

Referenced by the following interfaces: DescribePolicyGroupInfo, DescribePolicyGroupList

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| EventId | yes | no | Int64 | Event ID.<br>Example: 10 |
| RuleId | yes | no | Int64 | Event alarm rule ID.<br>Example: 10 |
| EventShowName | yes | no | String | Event name.<br>Example: ReadOnly |
| AlarmNotifyPeriod | yes | no | Int64 | Alarm sending period in seconds. The value u003c0 indicates that no alarm will be triggered. The value 0 indicates that an alarm is triggered only once. The value >0 indicates that an alarm is triggered at the interval of triggerTime.<br>Example: 5 |
| AlarmNotifyType | yes | no | Int64 | Alarm sending and converging type. The value 0 indicates that alarms are sent consecutively. The value 1 indicates that alarms are sent exponentially.<br>Example: 1 |

## AttributeInfoOutputData

Attribute data list.

Referenced by the following interfaces: DescribeAttributes

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| Data | yes | no | Array of AttributeInfoOutput | Attribute data list.<br>Example:View |
| TotalCount | yes | no | Int64 | Number of attribute data lists.<br>Example: 10 |

## CgrpGroupNode

CgrpGroupNode

Referenced by the following interfaces: CreateGroup, DeleteGroup, DescribeGroup, InitGroup, ModifyGroup

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| AppId | no | yes | Int64 | Group ID.<br>Example: 1234 |
| High | no | no | Int64 | Group depth.<br>Example: 1 |
| Id | yes | no | Int64 | Group ID.<br>Example: 1 |
| IsLeaf | no | no | Bool | Whether it is a leaf node.<br>Example: true |
| Level | no | no | Int64 | Current level in the tree.<br>Example: 1 |
| Name | no | no | String | Tree node.<br>Example: Testing |
| ParentId | no | no | Int64 | Parent node ID.<br>Example: 1 |
| SubUin | no | yes | String | SubUin<br>Example: 1234 |
| TypeId | no | no | String | Type of the instance that is allowed to be stored in the leaf node.<br>Example: cvm_device |
| Uin | no | yes | String | Uin<br>Example: 123456 |

## DescribeProductEventListDimensions

Input parameter Dimensions of the DescribeProductEventList API

Referenced by the following interfaces: DescribeProductEventList

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| Name | yes | no | String | Dimension name.<br>Example: Testing |
| Value | yes | no | String | The value of the dimension.<br>Example: XXXX |

## Dimension

Combination of instance object dimensions

Referenced by the following interfaces: DescribeDashboardMetricData, DescribePolicyGroupCount, GetMonitorData, GetTkeData

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| Name | yes | no | String | Instance dimension name.<br>Example: InstanceId |
| Value | yes | no | String | Instance dimension value.<br>Example: ins-c282wnl0 |

## DescribePolicyInfoByInstanceGroupList

DescribePolicyInfoByInstanceGroupList

Referenced by the following interfaces: DescribePolicyInfoByInstance

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| GroupID | yes | no | Int64 | Policy group ID.<br>Example: 1234 |
| GroupName | yes | no | String | Policy name.<br>Example: Testing |
| IsOpen | yes | no | Bool | Indicates whether self-healing notifications are enabled.<br>Example: true |
| ViewName | yes | no | String | View name.<br>Example: cvm_device |
| LastEditUin | yes | no | Int64 | UIN of the last editor.<br>Example: 123456 |
| IsDefault | yes | no | Int64 | Whether it is a default policy.<br>Example: 1 |
| CanSetDefault | yes | no | Bool | Whether it can be set as the default policy.<br>Example: true |
| ParentGroupID | yes | no | Int64 | Parent policy group ID.<br>Example: 1234 |
| Remark | yes | no | String | Description.<br>Example: Testing |
| IsUnionRule | yes | no | Int64 | Whether it is an AND relationship policy.<br>Example: 1 |
| ProjectID | yes | no | Int64 | Project ID.<br>Example: 1 |
| Conditions | yes | yes | Array of Condition | Alarm rule.<br>Example:View |
| ConditionsTemp | yes | yes | DescribePolicyInfoByInstanceConditionsTemp | Rule template information.<br>Example:View |
| InstanceGroup | yes | yes | DescribePolicyInfoByInstanceInstanceGroup | Instance group.<br>Example:View |
| NoShieldedSum | yes | no | Int64 | Number of unshielded instances.<br>Example: 10 |
| UseSum | yes | no | Int64 | The total number of instances.<br>Example: 10 |
| UpdateTime | yes | no | Int64 | Update time.<br>Example: 1735095122 |
| InsertTime | yes | no | Int64 | Creation time.<br>Example: 1735095122 |

## AlarmPolicyQuota

Specific v2.0 alarm policy quota.

Referenced by the following interfaces: DescribeAlarmObjectQuota, DescribeAlarmPolicyQuota

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| AppId | yes | no | Int64 | The account ID.<br>Example: 1234 |

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| Namespace | yes | no | String | Policy type. "": no policy type.<br>Example: cvm_device |
| ProjectId | yes | no | Int64 | Project ID. -1: no project.<br>Example: 1 |
| Total | yes | no | Int64 | Total amount of quotas.<br>Example: 10 |
| Used | yes | no | Int64 | Number of used resources.<br>Example: 10 |

## DescribeAccidentEventListAlarms

Output parameter type of the DescribeAccidentEventList API

Referenced by the following interfaces: DescribeAccidentEventList

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| BusinessTypeDesc | yes | yes | String | Event category.<br>Example: qwer |
| AccidentTypeDesc | yes | yes | String | Event type.<br>Example: qwer |
| BusinessID | yes | yes | Int64 | ID of the event category. The value 1 indicates service issues. The value 2 indicates other subscriptions.<br>Example: 1 |
| EventStatus | yes | yes | Int64 | Event status ID. The value 0 indicates that the event has been recovered. The value 1 indicates that the event has not been recovered.<br>Example: 1 |
| AffectResource | yes | yes | String | Affected object.<br>Example: qwer |
| Region | yes | yes | String | Region where the event occurs.<br>Example: gz |
| OccurTime | yes | yes | String | Time when the event occurs.<br>Example: 1735041205 |
| UpdateTime | yes | yes | String | Update time.<br>Example: 1735041205 |

## InstanceGroups

Instance group of alarm object

Referenced by the following interfaces: DescribeAlarmHistories, DescribeMiniDashboardAlarmInfo

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| Id | yes | no | Int64 | Instance group ID.<br>Example: 1 |
| Name | yes | no | String | Instance group name.<br>Example: Testing |

## ClmMetricSet

Log monitoring metric set.

Referenced by the following interfaces: CreateMetricSet, DescribeMetricSet, DescribeMetricSets, ModifyMetricSet

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| LogsetId | yes | no | String | CLS logset ID.<br>Example: d5f10b84-def8-4e58-bab0-16ef567b1f22 |
| LogsetName | yes | no | String | CLS log set name.<br>Example: Testing |
| LogtopicId | yes | no | String | CLS log topic ID.<br>Example: d4253323-7dfc-4b1b-8889-d82cef77375a |
| LogtopicName | yes | no | String | CLS log topic name.<br>Example: Testing |
| Name | yes | no | String | Metric set name.<br>Example: Testing |
| StatisticCycle | yes | no | Int64 | Statistical period. Valid values: 60, 300.<br>Example: 1 |
| TimeField | yes | no | String | Time field.<br>Example: 3131431 |
| LogFilterRules | yes | yes | Array of ClmLogFilterRule | Log filtering rule.<br>Example:View |
| MetricBasicItems | yes | yes | Array of ClmMetricBasicItem | Basic metric rule.<br>Example:View |
| MetricCustomItems | yes | yes | Array of ClmMetricCustomItem | Compound metric rule.<br>Example:View |
| Dimensions | yes | yes | Array of ClmDimension | Dimension rule.<br>Example:View |
| Id | no | no | Int64 | Item ID (only for output parameter; leave it empty for input parameter)<br>Example: 1 |
| CreatedAt | no | no | String | Creation time (only for output parameter; leave it empty for input parameter)<br>Example: eqeadfrqc |
| UpdatedAt | no | no | String | Update time (only for output parameter; leave it empty for input parameter)<br>Example: eqeadfrqc |
| AppId | no | no | Int64 | App ID (only for output parameter; leave it empty for input parameter).<br>Example: 1234 |
| Uin | no | no | String | UIN (only for output parameter; leave it empty for input parameter).<br>Example: 123456 |
| Region | no | no | String | Region (only for output parameter; leave it empty for input parameter).<br>Example: gz |
| LogProfileItems | yes | yes | Array of ClmLogProfileItem | Log data structure description.<br>Example:View |

## DescribeAlarmSmsQuotaQuota

Quota information of the DescribeAlarmSmsQuota API.

Referenced by the following interfaces: DescribeAlarmSmsQuota

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| Type | yes | no | String | The quota type.<br>Example: cloud_test |
| Name | yes | no | String | Quota name.<br>Example: Cloud Automated Testing alarm |
| FreeLeft | yes | no | Int64 | Remaining free quota.<br>Example: 100 |
| PurchaseLeft | yes | no | Int64 | Remaining paid quota.<br>Example: 200 |
| Used | yes | no | Int64 | Number of used resources.<br>Example: 100 |

## ModifyNotifyBatchNotifyInfo

Alarm notification method of the ModifyNotifyBatch API.

Referenced by the following interfaces: ModifyNotifyBatch

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| GroupId | yes | no | Int64 | Policy group ID.<br>Example: 1234 |
| NotifyWay | yes | no | Array of String | List of alarm methods.<br>Example: sms |

## SubscribeInfo

Output parameter of DescribeSubscribeInfo.

Referenced by the following interfaces: DescribeSubscribeInfo

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| BusinessId | yes | yes | Int64 | Event type ID.<br>Example: 1 |
| BusinessName | yes | yes | String | Event name.<br>Example: qwer |
| AccidentId | yes | yes | Int64 | Question type ID.<br>Example: 1 |
| AccidentName | yes | yes | String | Question name.<br>Example: XXXX |
| NotifyWay | yes | yes | Array of Int64 | Notification method.<br>Example: 1 |

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| Receivers | yes | yes | Array of SubscribeInfoReceiver | Recipient list.<br>Example:View |
| Tips | yes | yes | String | Prompt.<br>Example: qwer |
| UserConfig | yes | yes | String | User configuration. JSON string.<br>Example: {"qwer":"1234"} |

## DescribeAppFlowConverterResponseDataPoint

DescribeAppFlowConverterResponseDataPoint

Referenced by the following interfaces: DescribeAppFlow

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| Value | yes | yes | Float | Value.<br>Example: 100 |

## InstanceGroupInfo

InstanceGroupInfo API returned by the DescribeInstanceGroup API.

Referenced by the following interfaces: DescribeInstanceGroup

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| GroupName | yes | yes | String | Instance group name.<br>Example: Testing |
| InstanceGroupId | yes | yes | Int64 | Instance group ID.<br>Example: 1234 |
| InstanceSum | yes | yes | Int64 | Number of instances.<br>Example: 10 |
| ViewName | yes | yes | String | View name/strategy type.<br>Example: cvm_device |
| InsertTime | yes | yes | Int64 | Creation time.<br>Example: 1735129241 |
| UpdateTime | yes | yes | Int64 | Update time.<br>Example: 1735129241 |
| LastEditUin | yes | yes | String | UIN of the last user who updated it.<br>Example: 123456 |

## DashboardPanel

Dashboard panel.

Referenced by the following interfaces: AddDashboardPanels, TopDashboardPanel

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| Type | no | no | String | copy |
| DashboardUUID | no | no | String | UUID of the dashboard.<br>Example: u78haeoyaq8w8wbr |
| PanelID | no | no | Int64 | Panel ID.<br>Example: 1234 |
| Panel | no | no | String | Panel data.<br>Example: qwer |

## ModifyPolicyGroupCondition

Modification of the metric threshold condition passed in by the alarm policy group.

Referenced by the following interfaces: ModifyPolicyGroup

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| RuleId | no | no | Int64 | Rule ID. No filling means new addition while filling in ruleId means to modify existing rules.<br>Example: 1 |
| MetricId | yes | no | Int64 | Metric ID.<br>Example: 10 |
| CalcType | yes | no | Int64 | Comparative type. The value 1 indicates greater than. The value 2 indicates greater than or equal to. The value 3 indicates smaller than. The value 4 indicates smaller than or equal to. The value 5 indicates equal to. The value 6 indicates not equal to.<br>Example: 10 |
| CalcValue | yes | no | String | Detection threshold.<br>Example: 10 |
| CalcPeriod | yes | no | Int64 | Data period of the detected metric.<br>Example: 5 |
| ContinuePeriod | yes | no | Int64 | Number of consecutive periods.<br>Example: 5 |
| AlarmNotifyType | yes | no | Int64 | Alarm sending and converging type. The value 0 indicates that alarms are sent consecutively. The value 1 indicates that alarms are sent exponentially.<br>Example: 1 |
| AlarmNotifyPeriod | yes | no | Int64 | Alarm sending period in seconds. The value u003c0 indicates that no alarm will be triggered. The value 0 indicates that an alarm is triggered only once. The value >0 indicates that an alarm is triggered at the interval of triggerTime.<br>Example: 5 |

## EventCondition

Event alarm conditions

Referenced by the following interfaces: DescribeConditionsTemplateList, DescribeRecommendedTemplate

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| AlarmNotifyPeriod | yes | no | String | Alarm notification frequency.<br>Example: 5 |
| AlarmNotifyType | yes | no | String | Predefined repeated notification policy. 0: One-time alarm; 1: exponential alarm; 2: consecutive alarm.<br>Example: 0 |
| EventID | yes | no | String | Event ID.<br>Example: qwer |
| EventDisplayName | yes | no | String | Displayed event name.<br>Example: test |
| RuleID | yes | no | String | Rule ID.<br>Example: qwer |

## NoticeContentTmpl

Information on the queried custom content templates.

Referenced by the following interfaces: DescribeNoticeContentTmpl

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| TmplID | no | no | String | Template ID<br>Example: tmpl-xxx |
| TmplName | no | no | String | Template name.<br>Example: tmplName |
| MonitorType | no | no | String | Monitoring type.<br>Example: MT_QCE |
| TmplContents | no | no | NoticeContentTmplItem | Content template details.<br>Example:View |
| TmplLanguage | no | no | String | Template language. Valid values: en and zh.<br>Example: zh |
| CreateTime | no | no | Int64 | Creation time. It is a Unix timestamp in second.<br>Example: 1729655205 |
| UpdateTime | no | no | Int64 | Modification time. It is a Unix timestamp in second.<br>Example: 1729655205 |
| LastModifier | no | no | String | Last Modified By<br>Example: tester-b |
| Creator | no | no | String | Creator.<br>Example: tester-a |
| IsPreset | no | no | Int64 | Whether the template content is the default content. 0: no; 1: yes.<br>Example: 0 |

## CgrpInstanceGroupNode

Group array in which the instance is located.

Referenced by the following interfaces: CreateInstances, DeleteInstances, DescribeInstance

| name | required | NULL allowed | type | description |
| --- | --- | --- | --- | --- |
| Id | no | yes | Int64 | The group ID.<br>Example: 1 |
| Name | no | yes | String | Group name.<br>Example: test |

## ModifyMsgPolicyVoiceConfig

Custom message voice alarm configuration.

Referenced by the following interfaces: CreateMsgPolicy, ModifyMsgPolicy

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| UidList | no | no | Array of Int64 | List of UIDs.<br>Example: [1234] |
| RoundNumber | no | no | Int64 | Number of alarm rounds. Valid values: 1-5.<br>Example: 10 |
| RoundInterval | no | no | Int64 | Intervals of alarm rounds in seconds.<br>Example: 10 |
| PersonInterval | no | no | Int64 | Alarm intervals for individuals in seconds.<br>Example: 10 |
| NeedSendNotice | no | no | Int64 | Whether notification is required. 1: yes. 0: no.<br>Example: 10 |

## ProjectInfo

Output parameter of the DescribeProjectsList API.

Referenced by the following interfaces: DescribeProjectsList

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| AppId | yes | no | Int64 | AppId<br>Example: 1234 |
| CreatorUin | yes | no | Int64 | The UIN of the creator.<br>Example: 123456 |
| Name | yes | no | String | Project name.<br>Example: Testing |
| Info | yes | no | String | Project information.<br>Example: qwer |
| IsDefault | yes | no | Int64 | Indicates whether to use your default value.<br>Example: 1 |
| ProjectId | yes | no | Int64 | Project ID.<br>Example: 1 |
| OwnerUin | yes | no | Int64 | The UIN of the root account.<br>Example: 123456 |
| SrcAppId | yes | no | Int64 | Source App ID.<br>Example: 1234 |
| SrcPlat | yes | no | String | Source platform.<br>Example: qwer |
| Status | yes | no | Int64 | State.<br>Example: 1 |
| CreateTime | yes | no | String | Creation time.<br>Example: 1735179378 |

## TagInstance

Instance tag information of the alarm policy

Referenced by the following interfaces: DescribeAlarmPolicies, DescribeAlarmPolicy

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Key | yes | yes | String | Tag key.<br>Example: tag |
| Value | yes | yes | String | Tag value.<br>Example: value |
| InstanceSum | yes | yes | Int64 | Number of instances.<br>Example: 1 |
| ServiceType | yes | yes | String | Product type. For example, cvm.<br>Example: cvm |
| RegionId | yes | yes | String | Region ID.<br>Example: 1 |
| BindingStatus | yes | yes | Int64 | Binding status. 2: bound; 1: binding.<br>Example: 2 |
| TagStatus | yes | yes | Int64 | Tag status. 2: existent; 1: nonexistent.<br>Example: 2 |

## MonitorProductInfo

Product information returned by the DescribeMonitorProductByIds API.

Referenced by the following interfaces: DescribeMonitorProductByIds

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Id | yes | yes | String | The product code.<br>Example: 10 |
| ProductName | yes | yes | String | Product name.<br>Example: cvm_device |
| PolicyViewName | yes | yes | String | Product view name.<br>Example: cvm_device |
| AvaliableRegions | yes | yes | Array of String | Region supported.<br>Example: ["gz"] |
| IsShowInDashboard | yes | yes | Bool | Whether the Dashboard is visible.<br>Example: false |
| Meta | yes | yes | String | Product details. JSON string.<br>Example: qwer |
| Namespace | yes | yes | String | Namespace of the product.<br>Example: QCE/CVM |

## CgrpInstance

Creates a new instance ID.

Referenced by the following interfaces: CreateInstances, DeleteInstances

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| InstanceId | yes | no | String | Instance ID.<br>Example: ins-00jvv9mo |
| InstanceName | no | no | String | Instance name.<br>Example: test |
| TypeId | no | no | String | Type<br>Example: cvm_device |
| Region | no | no | String | Region.<br>Example: gz |
| ModuleId | yes | no | Int64 | Leaf group node.<br>Example: 1 |

## DescribeSortObjectListDimension

Dimension of the DescribeSortObjectList API. It refers to a dimension, including the dimension name and dimension value.

Referenced by the following interfaces: DescribeSortObjectList

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Name | yes | no | String | Dimension name.<br>Example: appid |
| Value | yes | no | String | The value of the dimension.<br>Example: 1 |

## ServerInfo

ServerInfo

Referenced by the following interfaces: DescribeServers

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| ServerId | yes | no | Uint64 | Server ID.<br>Example: 1 |
| ServerName | yes | no | String | Server name.<br>Example: qwer |
| Ip | yes | no | String | Server private IP address.<br>Example: 127.0.0.1 |
| IdcId | yes | no | Int64 | Region ID.<br>Example: 10 |
| IdcName | yes | no | String | The region name.<br>Example: qwer |

## ClmAlertFilterRule

Expired.

Referenced by the following interfaces: CreateAlertPolicy, DescribeAlertPolicies, DescribeAlertPolicy, ModifyAlertPolicy

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| AlertPolicyId | yes | yes | Int64 | Associated alarm policy ID.<br>Example: 1234 |
| Key | yes | yes | String | Filtered keywords.<br>Example: qwer |
| Operating | yes | yes | String | Operator.<br>Example: = |
| Value | yes | yes | String | Filter value (multiple values are separated by commas).<br>Example: 1234 |
| Id | no | yes | Int64 | Filter condition ID.<br>Example: 1 |
| CreatedAt | no | yes | String | Creation time.<br>Example: eqeadfrqc |
| UpdatedAt | no | yes | String | Update time.<br>Example: eqeadfrqc |

## InstanceGroup

InstanceGroup in Alarms returned by the DescribeBasicAlarmList API

Referenced by the following interfaces: DescribeBasicAlarmList

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| InstanceGroupId | yes | yes | Int64 | Instance group ID.<br>Example: 1234 |
| InstanceGroupName | yes | yes | String | Instance group name.<br>Example: Testing |

## CvmAgentStatus

Agent status returned by the DescribeCvmAgentStatus API.

Referenced by the following interfaces: DescribeCvmAgentStatus

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| InstanceId | yes | no | String | Instance ID.<br>Example: ins-00jvv9mo |
| AgentInstalled | yes | no | Bool | Whether the agent is installed.<br>Example: false |

## ClmMetricAnalysisCustomMetric

Compound metric parameters in CLM metric analysis.

Referenced by the following interfaces: DescribeMetricAnalysisData

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Name | no | no | String | Compound metric name.<br>Example: Testing |
| Formula | yes | no | String | Operation expression.<br>Example: {"cluster_name":"es-dnxssssss"} |

## CommonNamespace

Unified namespace information

Referenced by the following interfaces: DescribeAllNamespaces

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Id | yes | no | String | Namespace identifier.<br>Example: performance_metric |
| Name | yes | no | String | The name of the namespace.<br>Example: Performance metrics |
| Value | yes | no | String | Namespace value.<br>Example: qwer |
| ProductName | yes | no | String | Product name.<br>Example: cvm_device |
| Config | yes | no | String | Configuration information.<br>Example: qwer |
| AvailableRegions | yes | no | Array of String | List of supported regions.<br>Example: ["bj","bjjr"] |
| SortId | yes | no | Int64 | Sort ID.<br>Example: 251 |
| DashboardId | yes | no | String | Unique identifier in Dashboard.<br>Example: eipv6 |

## AlarmNotice

Alarm notification template details

Referenced by the following interfaces: DescribeAlarmNotice, DescribeAlarmNotices, DescribeAlarmPolicies, DescribeAlarmPolicy

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Id | yes | yes | String | ID of the alarm notification template.<br>Example: 12345 |
| Name | yes | yes | String | Name of the alarm notification template.<br>Example: Default notification |
| UpdatedAt | yes | yes | String | Last modification time.<br>Example: 2020-11-16 13:56:56 |
| UpdatedBy | yes | yes | String | Last editor.<br>Example: 1500000688 |
| NoticeType | yes | yes | String | Alarm notification type. ALARM=Notification not recovered. OK=Notification recovered. ALL=All notifications.<br>Example: ALL |
| UserNotices | yes | yes | Array of UserNotice | User notification list.<br>Example:View |
| URLNotices | yes | yes | Array of URLNotice | Callback notification list.<br>Example:View |
| IsPreset | yes | yes | Int64 | Whether it is the system default notification template. 0: no. 1: Yes.<br>Example: 1 |
| NoticeLanguage | yes | yes | String | Notification language<br>Example: zh-CN |
| PolicyIds | no | yes | Array of String | List of alarm policy IDs bound to the alarm notification template.<br>Example: ["policy-abc123"] |
| AMPConsumerId | no | yes | String | Background amp consumer ID.<br>Example: Consumer-asfafasf |
| ExtraChannels | yes | yes | Array of String | Other non-public notification channels.<br>Example: ["TSS"] |
| BigCustomerNotices | no | yes | Array of BigCustomerNotice | BigCustomerNotices<br>Example:View |

## DashboardNamespace

Namespace information under Dashboard2.0 data source.

Referenced by the following interfaces: DescribeAllNamespaces, DescribeDashboardNamespaces

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Id | yes | no | String | Namespace identifier.<br>Example: dcchannel |
| Name | yes | no | String | The name of the namespace.<br>Example: Direct Connect—Dedicated tunnel |
| Value | yes | no | String | Namespace value.<br>Example: QCE/DCX |
| ProductName | yes | yes | String | Product name.<br>Example: Direct Connect—Dedicated tunnel |
| Config | yes | yes | String | Configuration information.<br>Example: {} |
| AvailableRegions | yes | yes | Array of String | List of supported regions.<br>Example: ["gz"] |
| SortId | yes | yes | Int64 | Sort ID.<br>Example: 301 |
| Type | no | yes | String | Namespace type.<br>Example: BUSINESS |
| DashboardId | no | no | String | DashboardId<br>Example: eipv6 |

## CCMGroupViewEntry

CCMGroupViewEntry

Referenced by the following interfaces: DescribeCCMGroupView

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| ViewId | yes | yes | Int64 | Group view ID.<br>Example: 1 |
| ViewName | yes | yes | String | Group view name.<br>Example: cvm_device |
| IsAuto | yes | yes | Int64 | Whether the group view automatically binds the metric. 0: no. 1: yes.<br>Example: 1 |

## AttributeAggrValueInfoOutputData

AttributeAggrValueInfoOutputData

Referenced by the following interfaces: DescribeAttributeAggregateData

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Data | yes | no | Array of AttributeTimestampValueOutput | Returns a list of aggregation data.<br>Example:View |
| TotalCount | yes | no | Int64 | Returns the number of aggregation data lists.<br>Example: 10 |

## AttributeValueInfoOutput

AttributeValueInfoOutput

Referenced by the following interfaces: DescribeServerData, DescribeViewData

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Values | yes | no | Array of AttributeTimestampValueOutput | Attribute value list.<br>Example:View |
| StartTime | yes | no | String | Start time.<br>Example: 1734950008 |
| EndTime | yes | no | String | End time.<br>Example: 1734953808 |
| StartPosition | yes | no | Int64 | Start position.<br>Example: 1 |
| EndPosition | yes | no | Int64 | End position.<br>Example: 1 |

## DescribeMonitorDataByAlarmIDDataPoint

DescribeMonitorDataByAlarmIDDataPoint

Referenced by the following interfaces: DescribeMonitorDataByAlarmID

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| IsNull | yes | no | Bool | Whether it is empty.<br>Example: true |
| Value | yes | no | Float | Value.<br>Example: 10 |

## ClmAlertTriggerRule

Deprecated parameter.

Referenced by the following interfaces: CreateAlertPolicy, DescribeAlertPolicies, DescribeAlertPolicy, ModifyAlertPolicy

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Id | no | yes | Int64 | Trigger condition ID (output parameter).<br>Example: 1 |
| CreatedAt | no | yes | String | Creation time (output parameter).<br>Example: eqeadfrqc |
| UpdatedAt | no | yes | String | Update time (output parameter).<br>Example: eqeadfrqc |
| AlertPolicyId | yes | yes | Int64 | Primary key ID of the associated alarm policy.<br>Example: 1234 |
| MetricName | yes | yes | String | Metric name (English).<br>Example: cpu_usage |
| MetricType | yes | yes | Int64 | Metric type. 1: common metric. 2: compound metric.<br>Example: 1 |
| Operating | yes | yes | String | Operator.<br>Example: = |
| Value | yes | yes | Float | Metric threshold.<br>Example: 50 |
| ContinuousCycleCount | yes | yes | Int64 | Number of consecutive periods.<br>Example: 10 |
| NoticeFrequencySec | yes | yes | Int64 | Notification frequency (interval between notifications in seconds).<br>Example: 10 |
| MetricFormula | no | yes | String | Compound index expression (leave it empty for basic indexes).<br>Example: {"cluster_name":"-es-dnsssss"} |

## NoticeContentTmplBindPolicyCount

Number of alarm policies bound to the notification content template.

Referenced by the following interfaces: DescribeNoticeContentTmpl

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| NoticeContentTmplID | no | no | String | Notification template ID<br>Example: tmpl-xxx |
| BindCount | no | no | Uint64 | Number of bound alarm policies.<br>Example: 1 |

## StrategyEntry

StrategyEntry

Referenced by the following interfaces: DescribeCCMGroupViewStrategy, DescribeStrategys

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| StrategyId | yes | no | Int64 | The ID of the alarm policy.<br>Example: 10 |
| StrategyName | yes | no | String | The name of the alarm policy.<br>Example: qwer |
| ReceiverType | yes | no | Int64 | Alarm reception type.<br>Example: 1 |
| ReceiverId | yes | no | Array of Uint64 | List of IDs of alarm recipient users or user groups.<br>Example: 1234 |
| EffectiveStartTime | yes | no | String | The start time of the validity period.<br>Example: 1734953008 |
| EffectiveEndTime | yes | no | String | The end time of the validity period.<br>Example: 1734953808 |
| AlarmChannel | yes | no | Array of String | Alarm channel list. Valid values: "SMS", "SITE", "EMAIL", "CALL", and "WECHAT".<br>Example: sms |
| CallbackUrl | yes | no | String | Alarm callback address.<br>Example: http://test.com |
| Description | yes | no | String | Description.<br>Example: Testing |
| CreatorId | yes | no | Uint64 | Creator ID.<br>Example: 1 |
| CreateTime | yes | no | String | Creation time.<br>Example: 1735179378 |
| UpdaterId | yes | no | Uint64 | ID of the user who updated it.<br>Example: 10 |
| UpdateTime | yes | no | String | Update time.<br>Example: 1735179378 |
| IsStart | yes | no | Int64 | Whether to start it. 0: stopping. 1: starting.<br>Example: 1 |
| MixType | yes | no | Uint64 | Type of alarm objects, 1: server. 2: view.<br>Example: 1 |
| MixSubType | yes | no | Int64 | Alarm object subtype. When MixType is 2, then MixId is a list of view IDs. When MixId is an empty array [], then the background will automatically query the default view ID. When MixType is 1 (server), if MixSubType is 1, MixId can be left empty. If MixSubType is 2, then MixId is a list of server Ids, which can contain more than one server ID.<br>Example: 10 |
| Alarm | yes | no | Array of StrategyEntryAlarm | Alarm rule list.<br>Example:View |

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| MixId | yes | no | Array of Int64 | List of alarm object type IDs.<br>Example: 1 |
| Ip | yes | no | Array of String | If MixType is the server type and the subtype is the specified server ID, then the Ip list exists.<br>Example: 127.0.0.1 |

## StrategyEntryAlarm

StrategyEntryAlarm

Referenced by the following interfaces: CreateStrategy, DescribeCCMGroupViewStrategy, DescribeStrategys, ModifyStrategy

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| AlarmType | yes | no | Uint64 | Alarm rule type for the policy. 100: Maximum value (i.e. >), 101: Maximum value (i.e. >=), 102: Minimum value alarm (i.e. <), 103: Minimum value alarm (i.e. <=), 104: Fluctuation value alarm.<br>Example: 1 |
| AttributeId | yes | no | Uint64 | Alarm rule attribute ID of the policy.<br>Example: 1 |
| Threshold | yes | no | Int64 | Alarm rule threshold of the policy.<br>Example: 1 |
| DetectCount | yes | no | Int64 | Policy's alarm rule: the number of alarms in DetectPeriod minutes.<br>Example: 1 |
| DetectPeriod | yes | no | Int64 | Alarm period of the policy.<br>Example: 1 |
| TriggerPeriod | yes | no | Int64 | Policy's alarm rule: alarm interval period.<br>Example: 1 |
| Description | yes | no | String | Description.<br>Example: Testing |
| AttributeName | no | yes | String | Metric description.<br>Example: Testing |
| AttributeUniqueName | no | yes | String | Metric description in English. Unique name.<br>Example: qwer_123 |

## ClmLogProfileItem

Log monitoring metric set - log data structure description.

Referenced by the following interfaces: CreateMetricSet, DescribeMetricSet, DescribeMetricSets, ModifyMetricSet

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Id | no | no | Int64 | ID (only for output parameter; leave it empty for input parameter)<br>Example: 1 |
| MetricSetId | no | no | Int64 | Associated metric set ID (only for output parameter; leave it empty for input parameter)<br>Example: 12 |
| Field | yes | no | String | Field name.<br>Example: timeField |
| Type | yes | no | String | Field type. Valid values: string, long, double.<br>Example: string |
| CreatedAt | no | no | String | Creation time (only for output parameter; leave it empty for input parameter).<br>Example: eqeadfrqc |
| UpdatedAt | no | no | String | Update time (only for output parameter; leave it empty for input parameter).<br>Example: eqeadfrqc |

## MonitorTypeInfo

Detailed Information of Monitoring Type

Referenced by the following interfaces: DescribeMonitorTypes

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Id | yes | no | String | Monitoring Type ID<br>Example: 19023 |
| Name | yes | no | String | Monitoring type.<br>Example: MT_QCE |
| SortId | yes | no | Int64 | Arrangement Order<br>Example: 1 |

## ModifyConditionsTemplateRequestCondition

ModifyConditionsTemplateRequestCondition

Referenced by the following interfaces: CreateConditionsTemplate, ModifyConditionsTemplate

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| AlarmNotifyPeriod | no | no | String | Alarm notification cycle.<br>Example: 86400 |
| AlarmNotifyType | no | no | Int64 | Alarm notification method.<br>Example: 0 |
| CalcPeriod | yes | no | String | Statistical period.<br>Example: 60 |
| CalcType | yes | no | String | Statistical method.<br>Example: 1 |
| CalcValue | yes | no | String | Statistics value.<br>Example: 0 |
| ContinuePeriod | yes | no | String | Consecutive period.<br>Example: 1 |
| MetricId | yes | no | Int64 | Metric ID.<br>Example: 10 |
| RuleID | no | no | Int64 | Rule ID.<br>Example: 1 |

## AttributeInfoOutput

Attribute information.

Referenced by the following interfaces: DescribeAttributes, DescribeCCMGroupViewAttribute

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| AttributeId | yes | no | Uint64 | Attribute ID.<br>Example: 1 |
| AttributeName | yes | no | String | Attribute name.<br>Example: Testing |
| AttributeLevel | yes | no | Int64 | Attribute level.<br>Example: 1 |
| Description | yes | no | String | Description.<br>Example: test |
| AttributeType | yes | yes | String | The type.<br>Example: cpu_usage |
| DataType | yes | no | Int64 | Attribute data type.<br>Example: 1 |
| CreateTime | yes | no | String | Creation time.<br>Example: 1735041205 |
| OwnerName | yes | no | String | Attribute owner.<br>Example: test |
| UnitId | yes | no | Int64 | Unit ID.<br>Example: 1 |
| UnitName | yes | no | String | The unit.<br>Example: test |
| StatisticalPeriod | yes | no | Int64 | Statistical period.<br>Example: 60 |
| AttributeUniqueName | yes | no | String | Unique string of the attribute. It consists of letters, digits, hyphens or underscores.<br>Example: qwer_123 |

## AlarmEvent

Alarm event

Referenced by the following interfaces: DescribeAlarmEvents

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| EventName | yes | no | String | Event name.<br>Example: Testing |
| Description | yes | no | String | Displayed event name.<br>Example: Testing |
| Namespace | yes | no | String | Alarm policy type.<br>Example: cvm_device |
| Tooltip | yes | yes | MetricTooltip | Prompt information that can be displayed.<br>Example:View |

## DescribeMsgPolicyListVoiceConfig

DescribeMsgPolicyListVoiceConfig

Referenced by the following interfaces: DescribeMsgPolicyInfo, DescribeMsgPolicyList

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| RoundNumber | yes | no | Int64 | Number of calls.<br>Example: 10 |
| UIDList | yes | yes | Array of Int64 | UIN list.<br>Example: 10 |
| RoundInterval | yes | no | Int64 | Interval between each round.<br>Example: 10 |
| PersonInterval | yes | no | Int64 | Interval.<br>Example: 10 |
| NeedSendNotice | yes | no | Int64 | Whether to send a reminder.<br>Example: 10 |

## AttributeInfoInput

Attribute information.

Referenced by the following interfaces: CreateAttributes

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| AttributeName | yes | no | String | Attribute name.<br>Example: Testing |
| AttributeTypeId | no | no | Uint64 | The type.<br>Example: 1 |
| DataType | yes | no | Int64 | Attribute data type.<br>Example: 1 |
| AttributeLevel | no | no | Int64 | Attribute level.<br>Example: 1 |
| Unit | no | no | Int64 | Attribute unit.<br>Example: 1 |
| StatisticalPeriod | no | no | Int64 | Statistical period.<br>Example: 60 |
| OwnerId | no | no | Array of Uint64 | Owner list ID. The sub_uin of the logged in user is used by default.<br>Example: 1234 |
| Description | no | no | String | Description.<br>Example: Testing |
| AttributeUniqueName | yes | no | String | Unique name of the attribute. It consists of letters, digits, hyphens or underscores.<br>Example: qwer_123 |

## DescribeAgentStatusHistoryData

DescribeAgentStatusHistoryData

Referenced by the following interfaces: DescribeAgentStatusHistory

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| UnInstanceID | yes | no | String | Instance ID.<br>Example: ins-00jvv9mo |
| Status | yes | no | Int64 | CVM status.<br>Example: 1 |
| Histories | yes | no | Array of DescribeAgentStatusHistory | Historical data.<br>Example:View |

## DescribePolicyConditionListConfigManual

DescribePolicyConditionList.ConfigManual

Referenced by the following interfaces: DescribePolicyConditionList

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| CalcType | yes | yes | DescribePolicyConditionListConfigManualCalcType | Test method.<br>Example:View |
| CalcValue | yes | yes | DescribePolicyConditionListConfigManualCalcValue | Detection threshold.<br>Example:View |
| ContinueTime | yes | yes | DescribePolicyConditionListConfigManualContinueTime | Duration.<br>Example:View |
| Period | yes | yes | DescribePolicyConditionListConfigManualPeriod | Data cycle.<br>Example:View |
| PeriodNum | yes | yes | DescribePolicyConditionListConfigManualPeriodNum | Number of consecutive periods.<br>Example:View |
| StatType | yes | yes | DescribePolicyConditionListConfigManualStatType | The aggregation method.<br>Example:View |

## ModifyConditionsTemplateRequestEventCondition

ModifyConditionsTemplateRequestEventCondition

Referenced by the following interfaces: CreateConditionsTemplate, ModifyConditionsTemplate

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| AlarmNotifyPeriod | yes | no | String | Alarm notification cycle.<br>Example: 5 |
| AlarmNotifyType | yes | no | String | Alarm notification method.<br>Example: sms |
| EventID | yes | no | String | Event ID.<br>Example: 10 |
| RuleID | no | no | Int64 | Rule ID.<br>Example: 10 |

# ErrorCode

## Feature Description

If there is an Error field in the response, it means that the API call failed. For example:

```
{
    "Response": {
        "Error": {
            "Code": "AuthFailure.SignatureFailure",
            "Message": "The provided credentials could not be validated. Please check your signature is cor
rect."
        },
        "RequestId": "ed93f3cb-f35e-473f-b9f3-0d451b8b79c6"
    }
}
```

Code in Error indicates the error code, and Message indicates the specific information of the error.

## Error Code List

### Common Error Codes

| Error codes | Description |
| --- | --- |
| AuthFailure.InvalidSecretId | Invalid key (not a Cloud API key type). |
| AuthFailure.MFAFailure | MFA failed. |
| AuthFailure.SecretIdNotFound | Key does not exist. Check if the key has been deleted or disabled in the console, and if not, check if the key is correctly entered. Note that whitespaces should not exist before or after the key. |
| AuthFailure.SignatureExpire | Signature expired. Timestamp and server time cannot differ by more than five minutes. Please ensure your current local time matches the standard time. |
| AuthFailure.SignatureFailure | Invalid signature. Signature calculation error. Please ensure you've followed the signature calculation process described in the Signature API documentation. |
| AuthFailure.TokenFailure | Token error. |

| Error codes | Description |
| --- | --- |
| AuthFailure.UnauthorizedOperation | The request does not have CAM authorization. |
| DryRunOperation | DryRun Operation. It means that the request would have succeeded, but the DryRun parameter was used. |
| FailedOperation | Operation failed. |
| InternalError | Internal error. |
| InvalidAction | The API does not exist. |
| InvalidParameter | Incorrect parameter. |
| InvalidParameterValue | Invalid parameter value. |
| LimitExceeded | Quota limit exceeded. |
| MissingParameter | A parameter is missing. |
| NoSuchVersion | The API version does not exist. |
| RequestLimitExceeded | The number of requests exceeds the frequency limit. |
| ResourceInUse | Resource is in use. |
| ResourceInsufficient | Insufficient resource. |
| ResourceNotFound | The resource does not exist. |
| ResourceUnavailable | Resource is unavailable. |
| UnauthorizedOperation | Unauthorized operation. |
| UnknownParameter | Unknown parameter. |
| UnsupportedOperation | Unsupported operation. |
| UnsupportedProtocol | HTTPS request method error. Only GET and POST requests are supported. |
| UnsupportedRegion | API does not support the requested region. |

## Service Error Codes

| Error codes | Description |
| --- | --- |
| DryRunOperation | DryRun operation, which means the DryRun parameter is passed in yet the request will still be |

| Error codes | Description |
| --- | --- |
| | successful. |
| InvalidParameter.InvalidParameterParam | Invalid parameter. |
| FailedOperation.DbQueryFailed | Failed to query the database. |
| FailedOperation.DbRecordCreateFailed | Failed to create the database record. |
| FailedOperation.DruidTableNotFound | The druid table does not exist. |
| UnknownParameter | Unknown parameter error. |
| FailedOperation.DataTableNotFound | The data table doesn't exist. |
| ResourcesSoldOut | The resources have been sold out. |
| FailedOperation.DruidColumnNotFound | The druid table field does not exist. |
| ResourceInsufficient | Insufficient resource. |
| InvalidParameterValue.InvalDashboardNameExists | The dashboard name already exists. |
| ResourceNotFound | The resource does not exist. |
| FailedOperation.DbRecordDeleteFailed | Failed to delete the database record. |
| FailedOperation.DimQueryRequestFailed | Failed to query the service in the request dimension. |
| FailedOperation.AlertPolicyModifyFailed | Failed to modify the alarm policy. |
| FailedOperation.DataQueryFailed | Failed to query the data. |
| InvalidParameterValue.VersionMismatch | Inconsistent versions. |
| UnsupportedOperation | Unsupported operation. |
| FailedOperation.AlertPolicyCreateFailed | Failed to create the alarm policy. |
| FailedOperation.DivisionByZero | The dividend is zero. |
| FailedOperation.AlertPolicyDescribeFailed | Failed to query the alarm policy. |
| UnauthorizedOperation | Unauthorized operation. |
| FailedOperation.DruidQueryFailed | Failed to query analysis data. |
| InternalError.ExeTimeout | Execution timed out. |
| FailedOperation.DbRecordUpdateFailed | Failed to update the database record. |

| Error codes | Description |
| --- | --- |
| FailedOperation.DuplicateName | The name already exists. |
| FailedOperation.DbTransactionBeginFailed | Failed to start the database transaction. |
| FailedOperation.DataColumnNotFound | The data table field doesn't exist. |
| ResourceInUse | The resource is in use. |
| FailedOperation.ServiceNotEnabled | The service is not enabled and can be used only after being enabled. |
| InternalError | Internal error. |
| LimitExceeded | The quota limit has been reached. |
| FailedOperation.AlertTriggerRuleDeleteFailed | Failed to delete the trigger condition. |
| MissingParameter | Missing parameter. |
| FailedOperation.AlertFilterRuleDeleteFailed | Failed to delete the filter rule. |
| ResourceUnavailable | The resource is unavailable. |
| FailedOperation.AlertPolicyDeleteFailed | Failed to delete the alarm policy. |
| LimitExceeded.MetricQuotaExceeded | Quota limit on metrics is reached. Requests containing unregistered metrics are prohibited. |
| InvalidParameterValue | Invalid parameter value. |
| FailedOperation.DbTransactionCommitFailed | Failed to submit the database transaction. |
| InvalidParameter.InvalidParameter | Invalid parameter. |
| FailedOperation | Operation failed. |
| AuthFailure.UnauthorizedOperation | The request is not authorized. Please refer to the CAM documentation for the description of authentication. |
| InvalidParameter | Invalid parameter. |

# version(2017-03-12)
# API overview

## API version

V3

## Cloud Monitor API

| Interface name | Interface function |
| --- | --- |
| DescribeBaseMetrics | DescribeBaseMetrics |
| GetMonitorData | GetMonitorData |

# CallingMethod signaturev1

Converge Cloud authenticates each access request, i.e. each request needs to include authentication information (Signature) in the common parameters to verify the identity of the requester.
The Signature is generated by the security credentials which include SecretId and SecretKey. If you don't have the security credentials yet, go to the API Key page to apply for them; otherwise, you cannot invoke the API.

## 1. Applying for Security Credentials

Before using the API for the first time, go to the API Key page to apply for security credentials.
Security credentials consist of SecretId and SecretKey:

- SecretId is used to identify the API requester.
- SecretKey is used to encrypt the signature string and verify it on the server.
- You must keep your security credentials private and avoid disclosure.

You can apply for the security credentials through the following steps:

1. Log in to the Converge Cloud Console.
2. Go to the Cloud API Key page.
3. On the API Key Management page, click Create Key to create a SecretId/SecretKey pair.

Note: Each account can have up to two pairs of SecretId/SecretKey.

## 2. Generating a Signature

With the SecretId and SecretKey, a signature can be generated. The following describes how to generate a signature:

Assume that the SecretId and SecretKey are:

- SecretId: AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPLE
- SecretKey: Gu5t9xGARNpq86cd98joQYCN3EXAMPLE

Note: This is just an example. For actual operations, please use your own SecretId and SecretKey.

Take the Cloud Virtual Machine's request to view the instance list (DescribeInstances) as an example. When

you invoke this API, the request parameters may be as follows:

| Parameter name | Description | Parameter value |
| --- | --- | --- |
| Action | Method name | DescribeInstances |
| SecretId | Key ID | AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPLE |
| Timestamp | Current timestamp | 1465185768 |
| Nonce | Random positive integer | 11886 |
| Region | Region where the instance is located | luzon |
| InstanceIds.0 | ID of the instance to query | ins-09dx96dg |
| Offset | Offset | 0 |
| Limit | Allowed maximum output | 20 |
| Version | API version number | 2017-03-12 |

## 2.1. Sorting Parameters

First, sort all the request parameters in an ascending lexicographical order (ASCII code) by their names. Notes: (1) Parameters are sorted by their names instead of their values; (2) The parameters are sorted based on ASCII code, not in an alphabetical order or by values. For example, InstanceIds.2 should be arranged after InstanceIds.12. You can complete the sorting process using a sorting function in a programming language, such as the ksort function in PHP. The parameters in the example are sorted as follows:

```
{
    'Action' : 'DescribeInstances',
    'InstanceIds.0' : 'ins-09dx96dg',
    'Limit' : 20,
    'Nonce' : 11886,
    'Offset' : 0,
    'Region' : 'luzon',
    'SecretId' : 'AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPLE',
    'Timestamp' : 1465185768,
    'Version': '2017-03-12',
}
```

When developing in another programming language, you can sort these sample parameters and it will work as long as you obtain the same results.

## 2.2. Concatenating a Request String

This step generates a request string.
Format the request parameters sorted in the previous step into the form of "parameter name"="parameter value". For example, for the Action parameter, its parameter name is "Action" and its parameter value is "DescribeInstances", so it will become Action=DescribeInstances after formatted.
Note: The "parameter value" is the original value but not the value after URL encoding.

Then, concatenate the formatted parameters with "&". The resulting request string is as follows:

```
Action=DescribeInstances&InstanceIds.0=ins-09dx96dg&Limit=20&Nonce=11886&Offset=0&Region=luzon&SecretId=AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPLE&Timestamp=1465185768&Version=2017-03-12
```

## 2.3. Concatenating the Signature Original String

This step generates a signature original string.
The signature original string consists of the following parameters:

1. HTTP method: POST and GET modes are supported, and GET is used here for the request. Please note that the method name should be in all capital letters.
2. Request host: the domain name of the request to view the list of instances (DescribeInstances) is cvm.convergecloud.com. The actual request domain name varies by the module to which the API belongs. For more information, see the instructions of the specific API.
3. Request path: The request path in the current version of Cloud API is fixed to / .
4. Request string: the request string generated in the previous step.

The concatenation rule of the signature original string is: Request method + request host + request path + ? + request string

The concatenation result of the example is:

```
GETcvm.convergecloud.com/?Action=DescribeInstances&InstanceIds.0=ins-09dx96dg&Limit=20&Nonce=11886&Offset=0&Region=luzon&SecretId=AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPLE&Timestamp=1465185768&Version=2017-03-12
```

## 2.4. Generating a Signature String

This step generates a signature string.
First, use the HMAC-SHA1 algorithm to sign thesignature original stringobtained in the previous step, and then encode the generated signature using Base64 to obtain the final signature.

The specific code is as follows with the PHP language being used as an example:

```
$secretKey = 'Gu5t9xGARNpq86cd98joQYCN3EXAMPLE';
$srcStr = 'GETcvm.convergecloud.com/?Action=DescribeInstances&InstanceIds.0=ins-09dx96dg&Limit
=20&Nonce=11886&Offset=0&Region=luzon&SecretId=AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPL
E&Timestamp=1465185768&Version=2017-03-12';
$signStr = base64_encode(hash_hmac('sha1', $srcStr, $secretKey, true));
echo $signStr;
```

The final signature is:

```
EliP9YW3pW28FpsEdkXt/+WcGeI=
```

When developing in another programming language, you can sign and verify the original in the example above and it works as long as you get the same results.

# 3. Encoding a Signature String

The generated signature string cannot be directly used as a request parameter and must be URL encoded.

For example, if the signature string generated in the previous step is EliP9YW3pW28FpsEdkXt/+WcGeI= , the final signature string request parameter (Signature) is EliP9YW3pW28FpsEdkXt%2f%2bWcGeI%3d, which will be used to generate the final request URL.

**Note: If your request method is GET, or the request method is POST and the Content-Type is application/x-www-form-urlencoded, then all the request parameter values need to be URL encoded (except the parameter key and the symbol of =) when sending the request. Non-ASCII characters need to be encoded with UTF-8 before URL encoding. **

Note: The network libraries of some programming languages automatically URL encode all parameters, in which case there is no need to URL encode the signature string; otherwise, two rounds of URL encoding will cause the signature to fail.

Note: Other parameter values also need to be encoded using RFC 3986. Use %XY in percent-encoding for special characters such as Chinese characters, where "X" and "Y" are hexadecimal characters (0-9 and uppercase A-F), and using lowercase will cause an error.

# 4. Signature Failure

The following situational error codes for signature failure may occur. Please resolve the errors accordingly.

| Error code | Error description |
| --- | --- |
| AuthFailure.SignatureExpire | The signature is expired |

| Error code | Error description |
| --- | --- |
| AuthFailure.SecretIdNotFound | The key does not exist |
| AuthFailure.SignatureFailure | Signature error |
| AuthFailure.TokenFailure | Token error |
| AuthFailure.InvalidSecretId | Invalid key (not a Cloud API key type) |

# 5. Signature Demo

When calling API 3.0, you are recommended to use the corresponding Converge Cloud SDK 3.0, Swhich encapsulates the signature process, enabling you to focus on only the specific APIs provided by the product when developing. See SDK Center for more information. Currently, the following programming languages are supported:

- Python
- Java
- PHP
- Go
- Node

To further explain the signing process, we will use a programming language to implement the process described above. The request domain name, API and parameter values in the sample are used here. This goal of this example is only to provide additional clarification for the signature process, please see the SDK for actual usage.

The final output URL might be:  https://cvm.convergecloud.com/?Action=DescribeInstances&InstanceIds.0=ins-09dx96dg&Limit=20&Nonce=11886&Offset=0&Region=luzon&SecretId=AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPLE&Signature=EliP9YW3pW28FpsEdkXt%2F%2BWcGeI%3D&Timestamp=1465185768&Version=2017-03-12

Note: The key in the example is fictitious, and the timestamp is not the current time of the system, so if this URL is opened in the browser or called using commands such as curl, an authentication error will be returned: Signature expired. In order to get a URL that can work properly, you need to replace the SecretId and SecretKey in the example with your real credentials and use the current time of the system as the Timestamp.

Note: In the example below, even if you use the same programming language, the order of the parameters in the URL may be different for each execution. However, the order does not matter, as long as all the parameters are included in the URL and the signature is calculated correctly.

Note: The following code is only applicable to API 3.0. It cannot be directly used in other signature processes. Even with an older API, signature calculation errors may occur due to the differences in details. Please refer to the corresponding documentation.

## Java

```
import java.io.UnsupportedEncodingException;
import java.net.URLEncoder;
import java.util.Random;
import java.util.TreeMap;
import javax.crypto.Mac;
import javax.crypto.spec.SecretKeySpec;
import javax.xml.bind.DatatypeConverter;

public class CloudAPIDemo {
    private final static String CHARSET = "UTF-8";

    public static String sign(String s, String key, String method) throws Exception {
        Mac mac = Mac.getInstance(method);
        SecretKeySpec secretKeySpec = new SecretKeySpec(key.getBytes(CHARSET), mac.getAlgorithm());
        mac.init(secretKeySpec);
        byte[] hash = mac.doFinal(s.getBytes(CHARSET));
        return DatatypeConverter.printBase64Binary(hash);
    }

    public static String getStringToSign(TreeMap<String, Object> params) {
        StringBuilder s2s = new StringBuilder("GETcvm.convergecloud.com/?");
        // When signing, the parameters need to be sorted in lexicographical order. TreeMap is used here
to guarantee the correct order.
        for (String k : params.keySet()) {
            s2s.append(k).append("=").append(params.get(k).toString()).append("&");
        }
        return s2s.toString().substring(0, s2s.length() - 1);
    }

    public static String getUrl(TreeMap<String, Object> params) throws UnsupportedEncodingException
{
        StringBuilder url = new StringBuilder("https://cvm.convergecloud.com/?");
        // There is no requirement for the order of the parameters in the actual request URL.
        for (String k : params.keySet()) {
            // The request string needs to be URL encoded. As the Key is all in English letters, only the valu
e is URL encoded here.
            url.append(k).append("=").append(URLEncoder.encode(params.get(k).toString(), CHARSET)).app
end("&");
        }
        return url.toString().substring(0, url.length() - 1);
```

```
    }

    public static void main(String[] args) throws Exception {
        TreeMap<String, Object> params = new TreeMap<String, Object>(); // TreeMap enables automati
c sorting
        // A random number should be used when actually calling, for example: params.put("Nonce", new
Random().nextInt(java.lang.Integer.MAX_VALUE));
        params.put("Nonce", 11886); // Common parameter
        // The current time of the system should be used when actually calling, for example:   params.put
("Timestamp", System.currentTimeMillis() / 1000);
        params.put("Timestamp", 1465185768); // Common parameter
        params.put("SecretId", "AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPLE"); // Common parameter
        params.put("Action", "DescribeInstances"); // Common parameter
        params.put("Version", "2017-03-12"); // Common parameter
        params.put("Region", "luzon"); // Common parameter
        params.put("Limit", 20); // Business parameter
        params.put("Offset", 0); // Business parameter
        params.put("InstanceIds.0", "ins-09dx96dg"); // Business parameter
        params.put("Signature", sign(getStringToSign(params), "Gu5t9xGARNpq86cd98joQYCN3EXAMPLE
", "HmacSHA1")); // Common parameter
        System.out.println(getUrl(params));
    }
}
```

## Python

Note: If running in a Python 2 environment, the following requests dependency package must be installed first:  pip install requests .

```
# -*- coding: utf8 -*-
import base64
import hashlib
import hmac
import time

import requests

secret_id = "AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPLE"
secret_key = "Gu5t9xGARNpq86cd98joQYCN3EXAMPLE"

def get_string_to_sign(method, endpoint, params):
    s = method + endpoint + "/?"
    query_str = "&".join("%s=%s" % (k, params[k]) for k in sorted(params))
    return s + query_str

def sign_str(key, s, method):
```

```
    hmac_str = hmac.new(key.encode("utf8"), s.encode("utf8"), method).digest()
    return base64.b64encode(hmac_str)

if __name__ == '__main__':
    endpoint = "cvm.convergecloud.com"
    data = {
        'Action' : 'DescribeInstances',
        'InstanceIds.0' : 'ins-09dx96dg',
        'Limit' : 20,
        'Nonce' : 11886,
        'Offset' : 0,
        'Region' : 'luzon',
        'SecretId' : secret_id,
        'Timestamp' : 1465185768, # int(time.time())
        'Version': '2017-03-12'
    }
    s = get_string_to_sign("GET", endpoint, data)
    data["Signature"] = sign_str(secret_key, s, hashlib.sha1)
    print(data["Signature"])
    # n actual invocation would occur here, which may incur fees after success
    # resp = requests.get("https://" + endpoint, params=data)
    # print(resp.url)
```

# signaturev3

Converge Cloud The API will authenticate each access request, meaning that each request must include signature information (Signature) in the public request parameters to verify the requester's identity.
The signature information is generated by the security credentials, which include SecretId and SecretKey . If the user does not have security credentials, please apply for them on the Cloud API Key page, otherwise they cannot call the Cloud API interfaces.

## 1. Apply for security credentials.

Before using the Cloud API for the first time, please go to the Cloud API Key page to apply for a security credential.
Security credentials include SecretId and SecretKey:

- SecretId The security credentials are used to identify the identity of the API caller.
- SecretKey The secret key is used to encrypt and sign the string for server-side verification of the signature.
- Users must strictly protect their security credentials and avoid disclosure them.

The specific steps to apply for security credentials are as follows:

1. Log in to Converge Cloud Administration Center Console
2. Go to the console page for cloud API keys
3. On the Cloud API Key page, click New to create a pair of SecretId/SecretKey

Note: Developer accounts can have up to two pairs SecretId / SecretKey.

## 2. TC3-HMAC-SHA256 signature method

Note: For GET method, only Content-Type: application/x-www-form-urlencoded protocol format is supported. For POST method, Content-Type: application/json and Content-Type: multipart/form-data are currently supported. By default, all business interfaces are supported in json format. Only specific business interfaces are supported in multipart format. In this case, the interface cannot be called in json format. Refer to the specific business interface document description.

The following takes Cloud Virtual Machine querying Region-1 instance list as an example to introduce the signature calculation process step by step. We only use two parameters for the query instance list: Limit and Offset, called using the GET method.

Assume that the SecretId and SecretKey of the user are AKIDz 8krbsJ 5yKBZQpn 74WFkmLPx 3EXAMPLE and Gu 5t9xGARNpq 86cd 98joQYCN3EXAMPLE, respectively.

## 2.1. splice specification request string

Concatenate CanonicalRequest strings in the following format:

```
CanonicalRequest =
  HTTPRequestMethod + '\n' +
  CanonicalURI + '\n' +
  CanonicalQueryString + '\n' +
  CanonicalHeaders + '\n' +
  SignedHeaders + '\n' +
  HashedRequestPayload
```

- HTTPRequestMethod: HTTP request methods (GET, POST), GET in this example;
- CanonicalURI: URI parameter, API 3.0 fixed to forward slash (/);
- CanonicalQueryString: Query string in the URL that initiates the HTTP request. For POST requests, it is fixed as an empty string. For GET requests, it is the question mark (?) in the URL. The following string content, this example value is: Limit=10&Offset=0. Note: CanonicalQueryString needs to be URL encoded.
- CanonicalHeaders: The header information of participating signature, at least including host and content-type, can also be added with customized header to improve the uniqueness and security of its own request. Splicing rules: 1) head key and value are uniformly converted to lowercase, and the first and last spaces are removed, and spliced according to the format of key:value\n;2) multiple heads are spliced according to the dictionary sorting of head key (lowercase). In this example: content-type: application/x-www-form-urlencoded\nhost: cvm.convergecloud.com\n
- SignedHeaders: The header information involved in the signature indicates which headers participate in the signature of this request, and the header content contained in CanonicalHeaders is one-to-one corresponding. content-type and host are required headers. constructed as: 1) The head key is uniformly converted to lowercase; 2) Multiple header keys (lowercase) are concatenated according to dictionary sorting and separated by semicolons (;). In this case: content-type;host
- HashedRequestPayload: The hash value of the request body, calculated as Lowercase(HexEncode(Hash.SHA256(RequestPayload))), SHA256 hash the entire payload of the HTTP request body, then hexadecimal encoding, and finally converting the encoding string to lowercase letters. Note: For GET requests, RequestPayload is fixed as an empty string. For POST requests, RequestPayload is the payload of the HTTP request body.

According to the above rules, the resulting canonical request string in the example is as follows (for clarity, \nnewlines are replaced by a new line printed separately):

```
GET
```

```
/
Limit=10&Offset=0
content-type:application/x-www-form-urlencoded
host:cvm.convergecloud.com

content-type;host
e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855
```

## 2.2. Concatenate the string to be signed

Concatenate the string to be signed in the following format:

```
StringToSign =
  Algorithm + \n +
  RequestTimestamp + \n +
  CredentialScope + \n +
  HashedCanonicalRequest
```

- Algorithm: Signature algorithm, currently fixed as TC3-HMAC-SHA256;
- RequestTimestamp: Request timestamp, i.e. the X-TC-Timestamp value of the request header. The request in the example above is 1539084154;
- CredentialScope: Credential range in the format Date/service/tc3_request, containing date, requested service and termination string (tc3_request). **Date is the date of UTC standard time, the value needs to be consistent with UTC standard time converted by common parameter X-TC-Timestamp **;service is the product name, which must be consistent with the product domain name called, such as cvm. The above example request, the value is 2018-10-09/cvm/tc3_request;
- HashedCanonicalRequest: The hash value of the canonical request string obtained by concatenating the above steps is calculated by Lowercase(HexEncode(Hash.SHA256(CanonicalRequest))).

note:

> 1. Date must be calculated from timestamp X-TC-Timestamp and time zone UTC+0. Adding local time zone information to the system, such as East Zone 8, will cause calls to succeed during the day and night, but calls to fail in the early hours of the morning. Assuming the timestamp is 1551113065, the time in the East Eight District is 2019-02-26 00:44:25, but the calculated Date with UTC+0 should be 2019-02-25 instead of 2019-02-26.
> 2. Timestamp must be the current system time and ensure that the system time and standard time are synchronized. If the difference exceeds five minutes, it will fail. If it is not synchronized with the standard time for a long time, it may cause the request to fail after running for a period of time (return signature expiration error).

According to the above rules, the string to be signed in the example is as follows (for clarity,\nnewlines are

replaced by new lines printed):

```
TC3-HMAC-SHA256
1539084154
2018-10-09/cvm/tc3_request
91c9c192c14460df6c1ffc69e34e6c5e90708de2a6d282cccf957dbf1aa7f3a7
```

## 2.3. calculate the signature

1. Compute the derived signature key, pseudocode as follows

```
SecretKey = "Gu5t9xGARNpq86cd98joQYCN3EXAMPLE"
SecretDate = HMAC_SHA256("TC3" + SecretKey, Date)
SecretService = HMAC_SHA256(SecretDate, Service)
SecretSigning = HMAC_SHA256(SecretService, "tc3_request")
```

- SecretKey: Original SecretKey;
- Date: That is, the Date field information in Credential, as shown in the example above, is 2018-10-09.;
- Service: That is, the Service field information in Credential, as shown in the example above, is cvm;

2. Calculate the signature, pseudocode as follows

```
Signature = HexEncode(HMAC_SHA256(SecretSigning, StringToSign))
```

- SecretSigning: That is, the derivative signature key calculated above;
- StringToSign: That is, the character string to be signed calculated in step 2;

## 2.4. Splicing Authorization

Splice as follows Authorization:

```
Authorization =
    Algorithm + ' ' +
    'Credential=' + SecretId + '/' + CredentialScope + ', ' +
    'SignedHeaders=' + SignedHeaders + ', '
    'Signature=' + Signature
```

- Algorithm: Signature method, fixed as TC3-HMAC-SHA256;
- SecretId: SecretId in key pair;
- CredentialScope: See above, Scope of Voucher;
- SignedHeaders: See above, header information for participating signatures;
- Signature: signature value

According to the above rules, the values obtained in the example are:

```
TC3-HMAC-SHA256 Credential=AKIDEXAMPLE/Date/service/tc3_request, SignedHeaders=content-type;host, Signature=5da7a33f6993f0614b047e5df4582db9e9bf4672ba50567dba16c6ccf174c474
```

The final complete invocation information is as follows:

```
https://cvm.convergecloud.com/?Limit=10&Offset=0

Authorization: TC3-HMAC-SHA256 Credential=AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPLE/2018-10-09/cvm/tc3_request, SignedHeaders=content-type;host, Signature=5da7a33f6993f0614b047e5df4582db9e9bf4672ba50567dba16c6ccf174c474
Content-Type: application/x-www-form-urlencoded
Host: cvm.convergecloud.com
X-TC-Action: DescribeInstances
X-TC-Version: 2017-03-12
X-TC-Timestamp: 1539084154
X-TC-Region: luzon
```

# 3. signing failed

According to the actual situation, there are the following signature failure error codes, please handle according to the actual situation

| error code | error description |
| --- | --- |
| AuthFailure.SignatureExpire | signature expiration |
| AuthFailure.SecretIdNotFound | key is not present |
| AuthFailure.SignatureFailure | signature error |
| AuthFailure.TokenFailure | token error |
| AuthFailure.InvalidSecretId | Illegal key (not cloud API key type) |

# 4. Signature Demo

Java

```
import java.io.BufferedReader;
import java.io.InputStream;
```

```
import java.io.InputStreamReader;
import java.net.URL;
import java.text.SimpleDateFormat;
import java.util.Date;
import java.util.Map;
import java.util.TimeZone;
import java.util.TreeMap;
import javax.crypto.Mac;
import javax.crypto.spec.SecretKeySpec;
import javax.net.ssl.HttpsURLConnection;
import javax.xml.bind.DatatypeConverter;

import org.apache.commons.codec.digest.DigestUtils;

public class CloudAPITC3Demo {
    private final static String CHARSET = "UTF-8";
    private final static String ENDPOINT = "cvm.convergecloud.com";
    private final static String PATH = "/";
    private final static String SECRET_ID = "AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPLE";
    private final static String SECRET_KEY = "Gu5t9xGARNpq86cd98joQYCN3EXAMPLE";
    private final static String CT_X_WWW_FORM_URLENCODED = "application/x-www-form-urlencode
d";
    private final static String CT_JSON = "application/json";
    private final static String CT_FORM_DATA = "multipart/form-data";

    public static byte[] sign256(byte[] key, String msg) throws Exception {
        Mac mac = Mac.getInstance("HmacSHA256");
        SecretKeySpec secretKeySpec = new SecretKeySpec(key, mac.getAlgorithm());
        mac.init(secretKeySpec);
        return mac.doFinal(msg.getBytes(CHARSET));
    }

    public static void main(String[] args) throws Exception {
        String service = "cvm";
        String host = "cvm.convergecloud.com";
        String region = "luzon";
        String action = "DescribeInstances";
        String version = "2017-03-12";
        String algorithm = "TC3-HMAC-SHA256";
        String timestamp = "1539084154";
        //String timestamp = String.valueOf(System.currentTimeMillis() / 1000);
        SimpleDateFormat sdf = new SimpleDateFormat("yyyy-MM-dd");
        // Pay attention to the time zone, otherwise it is easy to make mistakes.
        sdf.setTimeZone(TimeZone.getTimeZone("UTC"));
        String date = sdf.format(new Date(Long.valueOf(timestamp + "000")));

        // ************* Step 1: Assemble the specification request string *************
```

```
        String httpRequestMethod = "GET";
        String canonicalUri = "/";
        String canonicalQueryString = "Limit=10&Offset=0";
        String canonicalHeaders = "content-type:application/x-www-form-urlencoded\n" + "host:" + host
+ "\n";
        String signedHeaders = "content-type;host";
        String hashedRequestPayload = DigestUtils.sha256Hex("");
        String canonicalRequest = httpRequestMethod + "\n" + canonicalUri + "\n" + canonicalQueryStri
ng + "\n"
                + canonicalHeaders + "\n" + signedHeaders + "\n" + hashedRequestPayload;
        System.out.println(canonicalRequest);

        // ************* Step 2: Stitching the string to be signed *************
        String credentialScope = date + "/" + service + "/" + "tc3_request";
        String hashedCanonicalRequest = DigestUtils.sha256Hex(canonicalRequest.getBytes(CHARSET));
        String stringToSign = algorithm + "\n" + timestamp + "\n" + credentialScope + "\n" + hashedCan
onicalRequest;
        System.out.println(stringToSign);

        // ************* Step 3: Calculate the signature *************
        byte[] secretDate = sign256(("TC3" + SECRET_KEY).getBytes(CHARSET), date);
        byte[] secretService = sign256(secretDate, service);
        byte[] secretSigning = sign256(secretService, "tc3_request");
        String signature = DatatypeConverter.printHexBinary(sign256(secretSigning, stringToSign)).toLow
erCase();
        System.out.println(signature);

        // ************* Step 4: Splicing Authorization *************
        String authorization = algorithm + " " + "Credential=" + SECRET_ID + "/" + credentialScope + ", "
                + "SignedHeaders=" + signedHeaders + ", " + "Signature=" + signature;
        System.out.println(authorization);

        TreeMap<String, String> headers = new TreeMap<String, String>();
        headers.put("Authorization", authorization);
        headers.put("Host", host);
        headers.put("Content-Type", CT_X_WWW_FORM_URLENCODED);
        headers.put("X-TC-Action", action);
        headers.put("X-TC-Timestamp", timestamp);
        headers.put("X-TC-Version", version);
        headers.put("X-TC-Region", region);
    }
}
```

## Python

```
# -*- coding: utf-8 -*-
```

```
import hashlib, hmac, json, os, sys, time
from datetime import datetime

# key parameters
secret_id = "AKIDz8krbsJ5yKBZQpn74WFkmLPx3EXAMPLE"
secret_key = "Gu5t9xGARNpq86cd98joQYCN3EXAMPLE"

service = "cvm"
host = "cvm.convergecloud.com"
endpoint = "https://" + host
region = "luzon"
action = "DescribeInstances"
version = "2017-03-12"
algorithm = "TC3-HMAC-SHA256"
timestamp = 1539084154
date = datetime.utcfromtimestamp(timestamp).strftime("%Y-%m-%d")
params = {"Limit": 10, "Offset": 0}

# ************* Step 1: Assemble the specification request string *************
http_request_method = "GET"
canonical_uri = "/"
canonical_querystring = "Limit=10&Offset=0"
ct = "x-www-form-urlencoded"
payload = ""
if http_request_method == "POST":
    canonical_querystring = ""
    ct = "json"
    payload = json.dumps(params)
canonical_headers = "content-type:application/%s\nhost:%s\n" % (ct, host)
signed_headers = "content-type;host"
hashed_request_payload = hashlib.sha256(payload.encode("utf-8")).hexdigest()
canonical_request = (http_request_method + "\n" +
                     canonical_uri + "\n" +
                     canonical_querystring + "\n" +
                     canonical_headers + "\n" +
                     signed_headers + "\n" +
                     hashed_request_payload)
print(canonical_request)

# ************* Step 2: Stitching the string to be signed *************
credential_scope = date + "/" + service + "/" + "tc3_request"
hashed_canonical_request = hashlib.sha256(canonical_request.encode("utf-8")).hexdigest()
string_to_sign = (algorithm + "\n" +
                  str(timestamp) + "\n" +
                  credential_scope + "\n" +
                  hashed_canonical_request)
print(string_to_sign)
```

```
# ************* Step 3: Calculate the signature *************
# Compute signature digest function
def sign(key, msg):
    return hmac.new(key, msg.encode("utf-8"), hashlib.sha256).digest()
secret_date = sign(("TC3" + secret_key).encode("utf-8"), date)
secret_service = sign(secret_date, service)
secret_signing = sign(secret_service, "tc3_request")
signature = hmac.new(secret_signing, string_to_sign.encode("utf-8"), hashlib.sha256).hexdigest()
print(signature)

# ************* Step 4: Splicing Authorization *************
authorization = (algorithm + " " +
                 "Credential=" + secret_id + "/" + credential_scope + ", " +
                 "SignedHeaders=" + signed_headers + ", " +
                 "Signature=" + signature)
print(authorization)

# Common parameters added to request header
headers = {
    "Authorization": authorization,
    "Host": host,
    "Content-Type": "application/%s" % ct,
    "X-TC-Action": action,
    "X-TC-Timestamp": str(timestamp),
    "X-TC-Version": version,
    "X-TC-Region": region,
}
```

# request

## 1. Service Address

Region refers to the geographical area of a physical data center. Converge Cloud ensures complete isolation between different regions, providing maximum stability and fault tolerance. To reduce access latency and improve download speed, it is recommended to choose the region closest to your customers.

You can view the complete region list by using the API interface "DescribeRegions".

## 2. Communications Protocol

All the Converge Cloud APIs communicate via HTTPS, providing highly secure communication tunnels.

## 3. Request Methods

Supported HTTP request methods:

- POST (recommended)
- GET

PThe Content-Type types supported by POST requests:

- application/json (recommended). The TC3-HMAC-SHA256 signature algorithm must be used.
- application/x-www-form-urlencoded. The HmacSHA1 or HmacSHA256 signature algorithm must be used.
- multipart/form-data (only supported by certain APIs). You must use TC3-HMAC-SHA256 to calculate the signature.

The size of a GET request packet is up to 32 KB. The size of a POST request is up to 1 MB when the HmacSHA1 or HmacSHA256 signature algorithm is used, and up to 10 MB when TC3-HMAC-SHA256 is used.

## 4. Character Encoding

Only UTF-8 encoding is used.

# result

## Response for Successful Requests

For example, when calling CAM API (version: 2017-03-12) to view the status of instances (DescribeInstancesStatus), if the request has succeeded, you may see the response as shown below:


```
{
    "Response": {
        "TotalCount": 0,
        "InstanceStatusSet": [],
        "RequestId": "b5b41468-520d-4192-b42f-595cc34b6c1c"
    }
}
```


- The API will return Response, which contains RequestId, as long as it processes the request. It does not matter if the request is successful or not.
- RequestId is the unique ID of an API request. Contact us with this ID when an exception occurs.
- Except for the fixed fields, all fields are action-specified. For the definitions of action-specified fields, see the corresponding API documentation. In this example, TotalCount and InstanceStatusSet are the fields specified by the API DescribeInstancesStatus. 0 TotalCount means that the requester owns 0 CVM instance so the InstanceStatusSet is empty.

## Response for Failed Requests

If the request has failed, you may see the response as shown below:


```
{
    "Response": {
        "Error": {
            "Code": "AuthFailure.SignatureFailure",
            "Message": "The provided credentials could not be validated. Please check your signature is cor
rect."
        },
        "RequestId": "ed93f3cb-f35e-473f-b9f3-0d451b8b79c6"
    }
}
```


- The presence of the Error field indicates that the request has failed. A response for a failed request will include Error, Code and Message fields.

- Code is the code of the error that helps you identify the cause and solution. There are two types of error codes so you may find the code in either common error codes or API-specified error codes.
- Message explains the cause of the error. Note that the returned messages are subject to service updates. The information the messages provide may not be up-to-date and should not be the only source of reference.
- RequestId is the unique ID of an API request. Contact us with this ID when an exception occurs.

# Common Error Codes

If there is an Error field in the response, it means that the API call failed. The Code field in Error indicates the error code. The following table lists the common error codes that all actions can return.

| Error Code | Description |
| --- | --- |
| AuthFailure.InvalidSecretId | Invalid key (not a API key type). |
| AuthFailure.MFAFailure | MFA failed. |
| AuthFailure.SecretIdNotFound | The key does not exist. |
| AuthFailure.SignatureExpire | Signature expired. |
| AuthFailure.SignatureFailure | Signature error. |
| AuthFailure.TokenFailure | Token error. |
| AuthFailure.UnauthorizedOperation | The request does not have CAM authorization. |
| DryRunOperation | DryRun Operation. It means that the request would have succeeded, but the DryRun parameter was used. |
| FailedOperation | Operation failed. |
| InternalError | Internal error. |
| InvalidAction | The API does not exist. |
| InvalidParameter | Incorrect parameter. |
| InvalidParameterValue | Invalid parameter value. |
| LimitExceeded | Quota limit exceeded. |
| MissingParameter | A parameter is missing. |
| NoSuchVersion | The API version does not exist. |

| Error Code | Description |
| --- | --- |
| RequestLimitExceeded | The number of requests exceeds the frequency limit. |
| ResourceInUse | Resource is in use. |
| ResourceInsufficient | Insufficient resource. |
| ResourceNotFound | The resource does not exist. |
| ResourceUnavailable | Resource is unavailable. |
| UnauthorizedOperation | Unauthorized operation. |
| UnknownParameter | Unknown parameter. |
| UnsupportedOperation | Unsupported operation. |
| UnsupportedProtocol | hHTTPS request method error. Only GET and POST requests are supported. |
| UnsupportedRegion | API does not support the requested region. |

# PublicParameter

Common parameters are parameters used to identify users and interface authentication purposes. If not necessary, these parameters are not described in the separate interface document of each interface, but these parameters need to be carried in each request before the request can be normally initiated.

## signature method v3

When using TC3-HMAC-SHA256 signature method, the common parameters need to be uniformly placed in the HTTP Header request header, as follows:

| parameter name | type | required | described |
|---|---|---|---|
| X-TC-Action | String | Yes | The name of the operation interface. Refer to the description of the public parameter Action in the input parameters of the interface documentation for the value. For example, the value for the query instance list interface of the cloud server is DescribeInstances. |
| X-TC-Region | String | Yes | region parameters, Data used to identify which region you wish to operate on. The region value accepted by the interface refers to the description of the input parameter common parameter Region in the interface document. Note: Some interfaces do not need to pass this parameter, which is specifically stated in the interface documentation. In this case, even passing this parameter will not take effect. |
| X-TC-Timestamp | Integer | Yes | The current UNIX timestamp, which records the time when the API request was initiated. for example 1529223702. Note: If the time difference from the server exceeds 5 minutes, it will cause a signature expiration error. |
| X-TC-Version | String | Yes | Version of API for operation. Refer to the description of the public parameter Version in the interface document for the value. For example, the version of the cloud server. 2017-03-12. |
| Authorization | String | Yes | HTTP The standard authentication header field is "Authorization". For example: TC3-HMAC-SHA256 Credential=AKIDEXAMPLE/Date/service/tc3_request, SignedHeaders=content-type;host, Signature=fe5f80f77d5fa3beca038a248ff027d0445342fe2855ddc963176630326f1024 Here,<br>- TC3-HMAC-SHA256: Signature method, currently fixed as this value,<br>- Credential: Credential: Signature credential; AKIDEXAMPLE is the SecretId; Date is a date in UTC time, and this value must match the value of X-TC-Timestamp (a common parameter) in UTC time format; service is the name of the product/service, and is generally a domain name prefix. For example,cvm;<br>- SignedHeaders: The headers that contains the authentication information; content-type and host are the required headers;<br>- Signature: Signature digest. |
| X-TC-Token | String | No | The token used for temporary certificates needs to be used in conjunction with a temporary key. Temporary keys and tokens need to be obtained by calling the access management service interface. Long-term keys do not require tokens. |

## signature method v1

When using HmacSHA1 and HmacSHA256 signature methods, common parameters need to be placed uniformly in the request string, as follows

| parameter name | type | required | described |
| --- | --- | --- | --- |
| Action | String | Yes | The name of the interface to be operated. The value refers to the explanation of the public parameter Action in the interface documentation. For example, for the cloud server's query instance list interface, the value is DescribeInstances . |
| Region | String | Yes | The region parameter is used to identify which region's data you want to operate on. The region value accepted by the interface is specified in the input parameter of the public parameter Region in the interface documentation. Note: Some interfaces do not require this parameter to be passed, which will be specifically mentioned in the interface documentation. Even if this parameter is passed, it will not take effect in this case. |
| Timestamp | Integer | Yes | The current UNIX timestamp, which can record the time when the API request is initiated. For example, 1529223702. If the difference between the current time is too large, it will cause a signature expiration error. |
| Nonce | Integer | Yes | A random positive integer combined with Timestamp is used to prevent replay attacks. |
| SecretId | String | Yes | The SecretId is an identifier applied for the cloud API key to identify the user's identity. A SecretId corresponds to a unique SecretKey , and the SecretKey is used to generate the request signature Signature . |
| Signature | String | Yes | Request signature is used to verify the legitimacy of this request. Users need to calculate it based on the actual input parameters. The specific calculation method can be found in the interface authentication documentation. |
| Version | String | Yes | The version of the API being operated on. The value refers to the description of the public parameter Version in the interface documentation. For example, the version of the cloud server. 2017-03-12. |
| SignatureMethod | String | No | Signature method, currently supports HmacSHA256 and HmacSHA1. HmacSHA256 algorithm is used to verify signatures only if this parameter is specified as HmacSHA 256, otherwise HmacSHA1 is used to verify signatures. |
| Token | String | No | Token used for temporary certificate needs to be used together with temporary key. The temporary key and Token need to be obtained from the Access Management service invocation interface. Long-term keys do not require tokens. |

# zone list

A region (Region) refers to the geographical area of a physical data center. Converge Cloud Translation: To deliver verification that completely isolates different regions, ensuring maximum stability and fault tolerance between regions. In order to reduce access latency and improve download speed, it is recommended that you choose the region closest to your customers.

You can view the complete region list by using the API interface Query Region List.

# Cloud Monitor API DescribeBaseMetrics

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

API for obtaining the details of basic metrics.

Default interface request rate limit: 20 times per second.

Interface update time: 2022-10-24 18:12:59.

The interface only verifies the signature but does not authenticate.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: DescribeBaseMetrics |
| Version | Yes | No | String | Public parameter, The value of this interface: 2017-03-12 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| Namespace | Yes | No | String | Service namespace.<br>Example: QCE/CVM |
| MetricName | No | No | String | Metric name.<br>Example: AccOuttraffic |

## 3. Output Parameters

| Parameter name | Type | Description |
| --- | --- | --- |
| MetricSet | Array of MetricObject | List of queried metric descriptions.<br>Example:View |
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

# 4. Error Code

The following lists only the error codes related to the interface business logic, For other error codes, see public ErrorCode.

| Error Code | Description |
| --- | --- |
| InternalError | Internal error. |

# GetMonitorData

## 1. Interface Description

Interface request domain name: monitor.api3.convergecloud.com.

Obtains the monitoring data of a cloud product. After you specify your product namespace, object dimension description, and monitoring metrics, you can obtain the corresponding monitoring data.
The API call frequency is limited to: 50 times/second, 500 times/minute.
If you call excessive metrics and objects in a short period of time, the system may fail to pull data because frequency is limited. We recommend that you balance your requests by time.

Default interface request rate limit: 300 times per second.

Interface update time: 2022-11-22 16:40:20.

The interface is both signature verification and authentication.

## 2. Input Parameters

The following list of request parameters only lists interface request parameters and some public parameters, For a complete list of public parameters see public request parameters.

| Parameter name | Required | Allow NULL | Type | Description |
|---|---|---|---|---|
| Action | Yes | No | String | Public parameter, The value of this interface: GetMonitorData |
| Version | Yes | No | String | Public parameter, The value of this interface: 2017-03-12 |
| Region | No | No | String | Public parameter,Regional information A list of supported geographies is available through the DescribeRegions interface |
| Namespace | Yes | No | String | Namespace. Each cloud product has a namespace.<br>Example: qce/cvm |
| MetricName | Yes | No | String | Metric name.<br>Example: cpu_usage |
| Period | No | No | Uint64 | Statistical period of monitoring data. The default value is 300. Unit: seconds.<br>Example: 300 |
| StartTime | No | No | Datetime_iso | Start time. Example: 2018-01-01 00:00:00.<br>Example: 2019-03-24T10:51:23+08:00 |
| EndTime | No | No | Datetime_iso | End time. The default value is the current time. endTime cannot be earlier than startTime.<br>Example: 2019-03-24T20:51:23+08:00 |
| Dimensions | Yes | No | Array of String | Combination of instance object dimensions<br>Example: [{"unInstanceId":"ins-c282wnl0"}] |
| Statistics | No | No | String | Statistical method.<br>Example: qwer |

## 3. Output Parameters

| Parameter name | Type | Description |
|---|---|---|
| MetricName | String | Monitoring metrics.<br>Example: cpu_usage |

| Parameter name | Type | Description |
| --- | --- | --- |
| StartTime | Datetime_iso | Start time of data point.<br>Example: 2019-03-24T10:50:00+08:00 |
| EndTime | Datetime_iso | End time of data point.<br>Example: 2019-03-24T20:50:00+08:00 |
| Period | Uint64 | Statistical period.<br>Example: 300 |
| DataPoints | Array of PointsObject | List of monitoring data.<br>Example:View |
| RequestId | String | Unique request id, Each request will return. The RequestId of the request needs to be provided when locating the problem. |

# 4. Error Code

The following lists only the error codes related to the interface business logic, For other error codes, see public ErrorCode.

| Error Code | Description |
| --- | --- |
| InternalError | Internal error. |
| InvalidParameterValue | Invalid parameter value. |

# Data structure

## MetricObject

Description of the unit and supported statistical period of the business metric

Referenced by the following interfaces: DescribeBaseMetrics

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Namespace | yes | no | String | Namespace. Each cloud product has a namespace.<br>Example: QCE/CVM |
| MetricName | yes | no | String | Metric name.<br>Example: CpuUsage |
| Unit | yes | no | String | Unit used by the metric.<br>Example: % |
| Period | yes | no | Array of Int64 | Statistical period in seconds supported by the metric, such as 60 and 300.<br>Example: 5 |

## PointsObject

List of the monitoring data of an instance.

Referenced by the following interfaces: GetMonitorData

| name | required | NULL allowed | type | description |
|---|---|---|---|---|
| Dimensions | yes | no | Array of String | Dimension combination of the monitored instance.<br>Example: qwer |
| Points | yes | no | Array of Float | Array of monitoring data points. The time span of each point is a Period value.<br>Example: 10 |

# ErrorCode

## Feature Description

If there is an Error field in the response, it means that the API call failed. For example:

```
{
    "Response": {
        "Error": {
            "Code": "AuthFailure.SignatureFailure",
            "Message": "The provided credentials could not be validated. Please check your signature is cor
rect."
        },
        "RequestId": "ed93f3cb-f35e-473f-b9f3-0d451b8b79c6"
    }
}
```

Code in Error indicates the error code, and Message indicates the specific information of the error.

## Error Code List

### Common Error Codes

| Error codes | Description |
| --- | --- |
| AuthFailure.InvalidSecretId | Invalid key (not a Cloud API key type). |
| AuthFailure.MFAFailure | MFA failed. |
| AuthFailure.SecretIdNotFound | Key does not exist. Check if the key has been deleted or disabled in the console, and if not, check if the key is correctly entered. Note that whitespaces should not exist before or after the key. |
| AuthFailure.SignatureExpire | Signature expired. Timestamp and server time cannot differ by more than five minutes. Please ensure your current local time matches the standard time. |
| AuthFailure.SignatureFailure | Invalid signature. Signature calculation error. Please ensure you've followed the signature calculation process described in the Signature API documentation. |
| AuthFailure.TokenFailure | Token error. |

| Error codes | Description |
| --- | --- |
| AuthFailure.UnauthorizedOperation | The request does not have CAM authorization. |
| DryRunOperation | DryRun Operation. It means that the request would have succeeded, but the DryRun parameter was used. |
| FailedOperation | Operation failed. |
| InternalError | Internal error. |
| InvalidAction | The API does not exist. |
| InvalidParameter | Incorrect parameter. |
| InvalidParameterValue | Invalid parameter value. |
| LimitExceeded | Quota limit exceeded. |
| MissingParameter | A parameter is missing. |
| NoSuchVersion | The API version does not exist. |
| RequestLimitExceeded | The number of requests exceeds the frequency limit. |
| ResourceInUse | Resource is in use. |
| ResourceInsufficient | Insufficient resource. |
| ResourceNotFound | The resource does not exist. |
| ResourceUnavailable | Resource is unavailable. |
| UnauthorizedOperation | Unauthorized operation. |
| UnknownParameter | Unknown parameter. |
| UnsupportedOperation | Unsupported operation. |
| UnsupportedProtocol | HTTPS request method error. Only GET and POST requests are supported. |
| UnsupportedRegion | API does not support the requested region. |

## Service Error Codes

| Error codes | Description |
| --- | --- |
| UnknownParameter | Unknown parameter error. |

| Error codes | Description |
| --- | --- |
| ResourcesSoldOut | The resources have been sold out. |
| ResourceInsufficient | Insufficient resource. |
| ResourceNotFound | The resource does not exist. |
| UnsupportedOperation | Unsupported operation. |
| UnauthorizedOperation | Unauthorized operation. |
| InvalidParameter | Invalid parameter. |
| ResourceInUse | The resource is in use. |
| InternalError | Internal error. |
| LimitExceeded | The quota limit has been reached. |
| MissingParameter | Missing parameter. |
| ResourceUnavailable | The resource is unavailable. |
| InvalidParameterValue | Invalid parameter value. |
| FailedOperation | Operation failed. |