

Application Performance Management

API Reference

Issue 01
Date 2026-01-06



Copyright © Huawei Cloud Computing Technologies Co., Ltd. 2026. All rights reserved.

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

Trademarks and Permissions



HUAWEI and other Huawei trademarks are the property of Huawei Technologies Co., Ltd.

All other trademarks and trade names mentioned in this document are the property of their respective holders.

Notice

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

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

Contents

1 Before You Start.....	1
1.1 Overview.....	1
1.2 Endpoints.....	1
1.3 Concepts.....	1
2 API Overview.....	3
3 Calling APIs.....	7
3.1 Making an API Request.....	7
3.2 Authentication.....	10
3.3 Response.....	12
4 Examples.....	14
4.1 Example 1: Querying Applications.....	14
5 APIs.....	15
5.1 Common APIs.....	15
5.1.1 Querying the Application List.....	15
5.1.2 Querying the Master Address.....	17
5.1.3 Obtaining the AK/SK.....	19
5.1.4 Searching for Components, Environments, and Agents in a Region.....	21
5.1.5 Saving a Monitoring Item.....	23
5.1.6 Querying the Monitoring Item List.....	26
5.1.7 Querying All Agents of an Application.....	28
5.1.8 Enabling or Disabling Collection for an Instance.....	32
5.1.9 Deleting an Agent.....	34
5.2 Region Query APIs.....	35
5.2.1 Querying the Regions Where APM Is Enabled.....	35
5.2.2 Querying All Supported Regions.....	37
5.3 Resource Management APIs.....	39
5.3.1 Querying Environment Tags.....	39
5.3.2 Obtaining an Application Tree.....	41
5.3.3 Obtaining the Component List.....	44
5.3.4 Obtaining the Environment List of a Component.....	46
5.3.5 Deleting a Component Based on the Component ID.....	48
5.3.6 Querying the Details About an Application.....	50

5.3.7 Querying the Details About a Sub-application.....	52
5.4 View APIs.....	54
5.4.1 Querying Monitoring Item Configurations.....	54
5.4.2 Querying the Trace Topology.....	58
5.4.3 Querying Event Details.....	61
5.4.4 Querying Span Data.....	65
5.4.5 Obtaining All Data of a Trace.....	71
5.4.6 Obtaining the Trend Graph.....	75
5.4.7 Obtaining Summary Table Data.....	78
5.4.8 Obtaining the Raw Data Table.....	83
5.4.9 Obtaining Raw Data Details.....	88
5.4.10 Obtaining the Instance Information.....	90
5.4.11 Obtaining the Monitoring Item Information.....	93
5.4.12 Obtaining the Details About a Monitoring Item.....	95
5.5 AK/SK Management APIs.....	99
5.5.1 Creating an AK/SK.....	99
5.5.2 Querying the AK/SK of a Tenant.....	100
5.5.3 Deleting an AK/SK.....	102
5.6 Alarm Management APIs.....	104
5.6.1 Querying the Alarm List.....	104
5.6.2 Querying the Alarm Message List.....	108
5.7 Topology Management APIs.....	111
5.7.1 Querying the Global Topology of an Application.....	111
5.7.2 Querying the Topology of a Component Environment.....	115
5.8 Transaction Query APIs.....	119
5.8.1 Querying the URL Tracing Configuration List.....	120
5.8.2 Querying a Region's Environments on Which URLs Are Added for Tracing.....	122
5.8.3 Querying the URL Tracing View List.....	124
5.8.4 Querying the Details About a URL Tracing View.....	127
5.9 Performance Profiling APIs.....	130
5.9.1 Querying a Line-Level Flame Graph.....	130
6 Permissions Policies and Supported Actions.....	134
7 Appendix.....	136
7.1 Error Codes.....	136
7.2 Obtaining a Project ID.....	136
7.3 Obtaining an Account ID.....	137
7.4 Obtaining API Parameters.....	137

1 Before You Start

1.1 Overview

Welcome to Application Performance Management (APM). APM monitors and manages the performance of cloud applications in real time, and analyzes the performance of distributed applications, helping O&M personnel quickly locate and resolve faults and performance bottlenecks.

This document describes how to use APM by calling APIs, such as API for querying applications.

Before using these APIs, ensure that you are familiar with APM concepts and functions.

1.2 Endpoints

An endpoint is the **request address** for calling an API. Endpoints vary depending on services and regions. To obtain the regions and endpoints, contact the enterprise administrator.

Select an endpoint based on your service requirements.

1.3 Concepts

- Account

An account is created upon successful registration with the cloud system. The account has full access permissions for all of its cloud services and resources. It can be used to reset user passwords and grant user permissions. The account is a payment entity and should not be used directly to perform routine management. For security purposes, create Identity and Access Management (IAM) users and grant them permissions for routine management.

- User

An IAM user is created using an account to use cloud services. Each IAM user has their own identity credentials (password and access keys).

An IAM user can view the account ID and user ID on the **My Credentials** page of the console. The account name, username, and password will be required for API authentication.

- Region

A region is a geographic area in which cloud resources are deployed. Availability zones (AZs) in the same region can communicate with each other over an intranet, while AZs in different regions are isolated from each other. For low network latency and quick resource access, select the nearest region. The choice of regions may also be subject to legal compliance requirements.

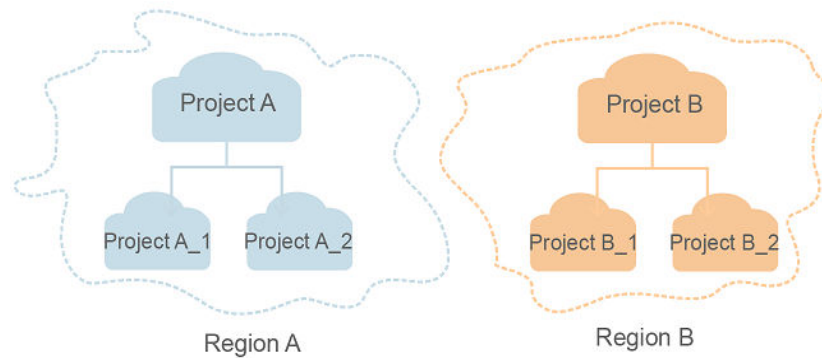
- AZ

An AZ comprises one or multiple physical data centers equipped with independent ventilation, fire, water, and electricity facilities. Computing, network, storage, and other resources in an AZ are logically divided into multiple clusters. AZs within a region are interconnected using high-speed optical fibers to allow you to build cross-AZ high-availability systems.

- Project

A project corresponds to a region. Projects are preset by the system and have physically isolated resources (including compute, storage, and network resources) across regions. Users can be granted permissions in a default project to access all resources in the region associated with the project. If you need more refined access control, create subprojects under a default project and create resources in subprojects. Then you can assign users the permissions required to access only the resources in the specific subprojects.

Figure 1-1 Project isolation model



2 API Overview

APIs provided by APM are extended APIs. By using the APIs provided by APM, you can use all functions of APM. For example, call an API to query applications.

Table 2-1 describes the APIs provided by APM.

Table 2-1 APM API description

API	Description
Querying the application list	The application list can be queried.
Querying the PodLB address of the master service	The podLB address of the master service in a region can be queried.
Obtaining the AK/SK	The created AK/SK can be queried.
Querying the environment list	The components, environments, and Agents in a specified region can be queried.
Saving the monitoring item configuration	The monitoring item configuration can be saved.
Querying the monitoring item list	The monitoring item list can be queried.
Querying application Agents	The Agents of an application can be queried.
Enabling or disabling the collection function for an instance	The collection function of a specified instance can be enabled or disabled.

API	Description
Deleting an Agent	An Agent can be deleted.
Querying the regions where APM is enabled	The regions where APM is enabled by the current user can be queried.
Querying the regions where APM is supported	The regions where APM is supported can be queried.
Querying an application tree	An application tree can be queried.
Querying environment tags	Environment tags can be queried.
Querying the component list	The component list can be queried.
Querying the environment list	The environment list of a component can be queried.
Deleting a component based on the component ID	A specified component can be deleted.
Querying the details of an application	The details of an application can be queried.
Querying the details of a sub-application	The details of a sub-application can be queried.
Querying the configuration of a monitoring item	The configuration of a monitoring item can be queried.
Querying the trace topology	The trace topology can be queried.
Querying the details of an event	The details of an event can be queried.
Querying the span data	The span data can be queried.
Querying trace data	All trace data can be queried by trace ID.

API	Description
Querying a trend graph	A trend graph can be queried.
Querying summary table data	Data in the summary table can be queried.
Querying a raw data table	A raw data table can be queried.
Querying the raw data details	Raw data details can be queried.
Querying the instance information list	The instance information list can be queried.
Querying the monitoring item information	Monitoring item information can be queried.
Creating an AK/SK	An AK/SK can be created.
Querying an AK/SK	The AK/SK of a tenant can be queried.
Deleting an AK/SK	An AK/SK can be deleted.
Querying the alarm list	The list of generated alarms can be queried.
Querying the alarm message list	The list of generated alarm messages can be queried.
Global application topology	The global topology of an application can be queried.
Querying the topology of a component environment	The topology of a component environment can be queried.
Querying the URL tracing configuration list	The URL tracing configuration list can be queried.

API	Description
Querying a region's environments on which URLs are added for tracing	A region's environments on which URLs are added for tracing can be queried.
Querying the URL tracing view list	The list of invoked URL tracing views can be queried.
Querying the details of a URL tracing view	The details of a URL tracing view can be queried.
Obtaining the details of a monitoring item	The details of a monitoring item can be obtained.

3 Calling APIs

3.1 Making an API Request

This section describes the structure of a REST API request, and uses the IAM API for obtaining a user token as an example to describe how to call an API. The obtained token can then be used to authenticate the calling of other APIs.

Request URI

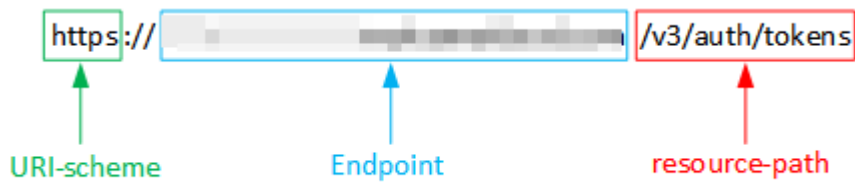
A request URI is in the following format:

{URI-scheme}://{Endpoint}/{resource-path}?{query-string}

Although a request URI is included in the request header, most programming languages or frameworks require the request URI to be transmitted separately.

- **URI-scheme:**
Protocol used to transmit requests. All APIs use **HTTPS**.
- **Endpoint:**
Domain name or IP address of the server bearing the REST service. The endpoint varies between services in different regions. It can be obtained from the enterprise administrator.
- **resource-path:**
Access path of an API for performing a specified operation. Obtain the path from the URI of an API. For example, the **resource-path** of the API used to obtain a user token is **/v3/auth/tokens**.
- **query-string:**
Query parameter, which is optional. Ensure that a question mark (?) is included before each query parameter that is in the format of "*Parameter name=Parameter value*". For example, **?limit=10** indicates that a maximum of 10 data records will be displayed.

Figure 3-1 Example URI



To simplify URI display, each API is provided only with a **resource-path** and a request method. The **URI-scheme** of all APIs is **HTTPS**, and the endpoints of all APIs in the same region are identical.

Request Methods

The HTTP protocol defines the following request methods that can be used to send a request to the server:

- **GET**: requests a server to return specified resources.
- **PUT**: requests a server to update specified resources.
- **POST**: requests a server to add resources or perform special operations.
- **DELETE**: requests a server to delete specified resources, for example, objects.
- **HEAD**: same as GET except that the server must return only the response header.
- **PATCH**: requests a server to update a part of a specified resource. If the resource does not exist, a new resource will be created.

For example, in the case of the API used to obtain a user token, the request method is **POST**. The request is as follows:

```
POST https://Endpoint/v3/auth/tokens
```

Request Header

You can also add additional header fields to a request, such as the fields required by a specified URI or HTTP method. For example, to request for the authentication information, add **Content-Type**, which specifies the request body type.

[Table 3-1](#) lists the common request header fields.

Table 3-1 Common request headers

Header	Description	Mandatory	Example
Host	Server domain name and port number of the resources being requested. The value can be obtained from the URL of the service API. The value is in the format of <i>host name:port number</i> . If no port is specified, the default port will be used. For HTTPS, port 443 is used by default.	No	code.test.com or code.test.com:443
Content-Type	Message body type or format. The default value application/json is recommended. Other values of this field will be provided for specific APIs if any.	Yes	application/json
Content-Length	Length of the request body. The unit is byte.	No	3495
X-Project-Id	Project ID.	No	e9993fc787d94b6c886cb aa340f9c0f4
X-Auth-Token	User token. It is a response to the API for obtaining a user token. This API is the only one that does not require authentication. After the request is processed, the value of X-Subject-Token in the response header is the token value.	No This field is mandatory for token authentication.	The following is part of an example token: MIIPAgYJKoZIhvcNAQc- Co...ggg1BBIINPXsidG9rZ

The API used to obtain a user token does not require authentication. Therefore, only the **Content-Type** field needs to be added to requests for calling the API. An example of such requests is as follows:

```
POST https://endpoint/v3/auth/tokens
Content-Type: application/json
```

Request Body (Optional)

This part is optional. The body of a request is often sent in a structured format as specified in the **Content-Type** header field. The request body transfers content except the request header.

A request body varies between APIs. Some APIs do not require the request body, such as the APIs requested using the GET and DELETE methods.

In the case of the API used to obtain a user token, the request parameters and parameter description can be obtained from the API request. The following provides an example request with a body included. Replace ***username***, ***domainname***, *********** (login password), and ***xxxxxxxxxxxxxxxxxxxx*** (project name), such as with the actual values. You can obtain a project name from the enterprise administrator.

The **scope** parameter specifies where a token takes effect. In the following example, the token takes effect only for the resources in a specified project. For more information about this API, see "Obtaining a User Token".

```
POST https://endpoint/v3/auth/tokens
Content-Type: application/json

{
  "auth": {
    "identity": {
      "methods": [
        "password"
      ],
      "password": {
        "user": {
          "name": "username",
          "password": "*****",
          "domain": {
            "name": "domainname"
          }
        }
      }
    },
    "scope": {
      "project": {
        "name": "xxxxxxxxxxxxxxxxxxxx"
      }
    }
  }
}
```

If all data required for the API request is available, you can send the request to call an API through [curl](#), [Postman](#), or coding. In the response to the API used to obtain a user token, **x-subject-token** is the desired user token. This token can then be used to authenticate the calling of other APIs.

3.2 Authentication

You can use either of the following authentication methods when calling APIs:

- Token authentication: Requests are authenticated using a token.
- AK/SK-based authentication: Requests are authenticated by encrypting the request body using an Access Key ID/Secret Access Key (AK/SK) pair.

Token-based Authentication

The validity period of a token is 24 hours. When using a token for authentication, cache it to prevent frequently calling the IAM API used to obtain a user token.

A token specifies temporary permissions in a computer system. During API authentication using a token, the token is added to requests to get permissions for calling the API.

You can obtain a token by calling the API used to obtain a user token. When you call the API, set **auth.scope** in the request body to **project**.

```
{
  "auth": {
    "identity": {
      "methods": [
        "password"
      ],
      "password": {
        "user": {
          "name": "username",
          "password": "*****",
          "domain": {
            "name": "domainname"
          }
        }
      }
    }
  },
  "scope": {
    "project": {
      "name": "xxxxxxx"
    }
  }
}
```

After a token is obtained, the **X-Auth-Token** header field must be added to requests to specify the token when calling other APIs. For example, if the token is **ABCDEFGH...**, **X-Auth-Token: ABCDEFGH...** can be added to a request as follows:

```
GET https://endpoint/v3/auth/projeAPM
X-Auth-Token: ABCDEFGH...
```

AK/SK-based Authentication

AK/SK-based authentication supports API requests with a body not larger than 12 MB. For API requests with a larger body, token-based authentication is recommended.

In AK/SK-based authentication, AK/SK is used to sign requests and the signature is then added to the requests for authentication.

- AK: access key ID, which is a unique identifier used in conjunction with a secret access key to sign requests cryptographically.
- SK: secret access key used in conjunction with an AK to sign requests cryptographically. It identifies a request sender and prevents the request from being modified.

In AK/SK-based authentication, you can use an AK/SK pair to sign requests based on the signature algorithm or use the signing SDK to sign requests.

The signing SDK is only used for signing requests and is different from the SDKs provided by services.

3.3 Response

Status Code

After sending a request, you will receive a response, including a status code, response header, and response body.

A status code is a group of digits, ranging from 1xx to 5xx. It indicates the status of a request.

For example, if status code **201** is returned for calling the API used to obtain a user token, the request is successful.

Response Header

Similar to a request, a response also has a header, for example, **Content-type**.

Figure 1 shows the response header fields for the API used to obtain a user token. The **x-subject-token** header field is the desired user token. This token can then be used to authenticate the calling of other APIs.

Figure 3-2 Header fields of the response to the request for obtaining a user token

```
connection → keep-alive
content-type → application/json
date → Tue, 12 Feb 2019 06:52:13 GMT
server → Web Server
strict-transport-security → max-age=31536000; includeSubdomains;
transfer-encoding → chunked
via → proxy A
x-content-type-options → nosniff
x-download-options → noopen
x-frame-options → SAMEORIGIN
x-iam-trace-id → 218d45ab-d674-4995-af3a-2d0255ba41b5
x-subject-token
→ MIIVXQVJKoZIhvcNAQcCoIIYJCCEoCAQExDTALBglghkgB8ZQMCAgEwgharBglqhkiG9w0BBwGgghacBIIWmHsidG9rZW4iOansiZXhwaXJlc19hdCI6ijlwMTktMDItMTNUMC
fj3KIs6YgKnpVNRbW2eZ5eb78SZ0kqJACgkIQ1wi4JlGzrpdl8LGXK5tdfdq4lqHCYb8P4NaY0NYejcAgzJVeFYtLWT1GSO0zxKZmlQHQj82HBqHdglZO9fuEbL5dMhdavj+33wEI
xHRCe9I87o+k9-
j+CMZSEB7bUGd5Uj6eRASXI1jipPEGA270g1FruooL6jqglFkNPQuFSOU8+uSsttVwRtNfsC+qTp22Rkd5MCqFGQ8LcuUxC3a+9CMBnOintWW7oeRUVhVpxk8pxiX1wTEboX-
RzT6MUbvpGw-oPNFYxJECKnoH3HRozv0vN--n5d6Nbxg==
x-ss-protection → 1; mode=block;
```

Response Body (Optional)

The body of a response is often returned in structured format as specified in the **Content-type** header field. The response body transfers content except the response header.

The following is part of the response body for the API used to obtain a user token.

```
{
  "token": {
    "expires_at": "2019-02-13T06:52:13.855000Z",
    "methods": [
      "password"
    ],
    "catalog": [
      {
        "endpoints": [
          {
            "region_id": "xx-xx",
            .....

```

If an error occurs during API calling, an error code and a message will be displayed. The following shows an error response body.

```
{
  "error_msg": "The format of message is error",
  "error_code": "AS.0001"
}
```

In the response body, **error_code** is an error code, and **error_msg** provides information about the error.

4 Examples

4.1 Example 1: Querying Applications

Scenario

This section describes how to query the applications of a user.

Related APIs

None

Querying Applications

- URI
URI format: GET /v1/apm2/openapi/cmdb/business/get-business-list
- Example request
GET https://{APM_endpoint}/v1/apm2/openapi/cmdb/business/get-business-list
Obtain *{APM_endpoint}* from the enterprise administrator.
- Example response:

```
{
  "business_nodes" : [ {
    "default" : true,
    "id" : 2,
    "gmt_create" : null,
    "gmt_modify" : null,
    "name" : "apm",
    "display_name" : "apm",
    "is_default" : true,
    "inner_domain_id" : 1,
    "eps_id" : "*****"
  } ]
}
```

5 APIs

5.1 Common APIs

5.1.1 Querying the Application List

Function

This API is used to query the application list.

URI

GET /v1/apm2/openapi/cmdb/business/get-business-list

Request Parameters

Table 5-1 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM.

Response Parameters

Status code: 200

Table 5-2 Response body parameters

Parameter	Type	Description
business_nodes	Array of BusinessNodeModel objects	Data structure used to query an application.

Table 5-3 BusinessNodeModel

Parameter	Type	Description
default	Boolean	Default application.
display_name	String	Display name of an application.
eps_id	String	Enterprise project ID.
gmt_create	String	Creation time.
gmt_modify	String	Update time.
id	Long	Application ID.
inner_domain_id	Integer	Internal tenant ID.
is_default	Boolean	Whether an application is the default application.
name	String	Application name.

Example Requests

Obtain the list of applications on which you have permission. No input parameter is required.

```
/v1/apm2/openapi/cmdb/business/get-business-list
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "business_nodes" : [ {
    "default" : true,
    "id" : 2,
    "gmt_create" : null,
    "gmt_modify" : null,
    "name" : "apm2",
    "display_name" : "apm2",
    "is_default" : true,
    "inner_domain_id" : 1,
    "eps_id" : "*****"
  } ]
}
```

```
}]
}
```

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Semantic or parameter error.
401	Unauthorized: No permissions.
403	Forbidden: Access forbidden.
404	Not Found: The requested resource is not found.

Error Codes

See [Error Codes](#).

5.1.2 Querying the Master Address

Function

This API is used to obtain the PodLB address of the master service based on the region name.

URI

GET /v1/apm2/openapi/systemmng/get-master-address

Table 5-4 Query Parameters

Parameter	Mandatory	Type	Description
region_name	Yes	String	Region name.

Request Parameters

Table 5-5 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM.

Response Parameters

Status code: 200

Table 5-6 Response body parameters

Parameter	Type	Description
region_name	String	Region name.
master_address	String	Address of the APM master service exposed to external systems. It is used for service registration and heartbeat reporting.

Example Requests

Obtain the PodLB address of the master service in . The input parameter is the region ID of .

```
/v1/apm2/openapi/systemmng/get-master-address?region_name=
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "region_name" : "",
  "master_address" : "https://*.*.*.31333"
}
```

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Semantic or parameter error.
401	Unauthorized: No permissions.
403	Forbidden: Access forbidden.
404	Not Found: The requested resource is not found.

Error Codes

See [Error Codes](#).

5.1.3 Obtaining the AK/SK

Function

This API is used to obtain the created AK/SK.

URI

GET /v1/apm2/openapi/systemmng/get-ak-sk-list

Request Parameters

Table 5-7 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM.

Response Parameters

Status code: 200

Table 5-8 Response body parameters

Parameter	Type	Description
access_ak_sk_models	Array of AccessAkskv objects	AK/SK data model.

Table 5-9 AccessAkskVO

Parameter	Type	Description
id	Integer	AK/SK ID.
gmt_create	String	Time when the AK/SK was generated.
gmt_modify	String	Time when the AK/SK was modified.
inner_domain_id	Integer	Internal tenant ID.
ak	String	Generated AK.
sk	String	Generated SK.
status	String	AK/SK status.
descp	String	AK/SK description.

Parameter	Type	Description
gmt_create_timestamp	Integer	Timestamp when the AK/SK was generated.
gmt_modify_timestamp	Integer	Timestamp when the AK/SK was modified.

Example Requests

Obtain the created AK/SK list. No input parameter is required.

```
/v1/apm2/openapi/systemmng/get-ak-sk-list
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "access_ak_sk_models": [ {
    "ak": "vca*****04j",
    "sk": "ktns*****6iq6t9m",
    "status": "enable",
    "descp": "auto create",
    "id": 1,
    "gmt_create": "2024-02-28 09:42:02",
    "gmt_modify": "2024-02-28 09:42:02",
    "inner_domain_id": 1,
    "gmt_create_timestamp": 1709084522000,
    "gmt_modify_timestamp": 1709084522000
  } ]
}
```

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Semantic or parameter error.
401	Unauthorized: No permissions.
403	Forbidden: Access forbidden.
404	Not Found: The requested resource is not found.

Error Codes

See [Error Codes](#).

5.1.4 Searching for Components, Environments, and Agents in a Region

Function

This API is used to search for components, environments, and Agents in a region.

URI

POST /v1/apm2/openapi/apm-service/app-mgr/search

Request Parameters

Table 5-10 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM.
x-business-id	Yes	Long	Application ID.

Table 5-11 Request body parameters

Parameter	Mandatory	Type	Description
business_id	Yes	Long	Application ID.
region	Yes	String	Region name.
page	Yes	Integer	Page number.
page_size	No	Integer	Number of records on each page.
keyword	No	String	Keyword used for search.

Response Parameters

Status code: 200

Table 5-12 Response body parameters

Parameter	Type	Description
app_info_list	Array of AppInfo objects	Component list.

Parameter	Type	Description
app_total_count	Integer	Total number of components.
app_info_map	Map<String,AppInfo>	Mappings between component names and details.

Table 5-13 AppInfo

Parameter	Type	Description
env_name	String	Environment name.
env_id	Long	Environment ID.
app_name	String	Component name.
app_id	Long	Component ID.
online_count	Integer	Number of online Agents.
disable_count	Integer	Number of Agents that are manually stopped.
offline_count	Integer	Number of offline Agents.
data_source	String	Data source.

Example Requests

Search for the Agents whose region ID is and application ID is 6, with the page number set to 1 and page size set to 10.

```
/v1/apm2/openapi/apm-service/app-mgr/search
{
  "business_id" : 6,
  "region" : "",
  "keyword" : "apm",
  "page" : 1,
  "page_size" : 10
}
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "app_info_list" : [ {
    "env_name" : "beta",
    "env_id" : 1,
    "app_name" : "application_name",
    "app_id" : 2,
    "online_count" : 2,
    "disable_count" : 1,
```

```
"offline_count" : 1
}],
"app_total_count" : 4,
"app_info_map" : {
  "additionalProp1" : {
    "env_name" : "string",
    "env_id" : 0,
    "app_name" : "string",
    "app_id" : 0,
    "online_count" : 0,
    "disable_count" : 0,
    "offline_count" : 0
  },
  "additionalProp2" : {
    "env_name" : "string",
    "env_id" : 0,
    "app_name" : "string",
    "app_id" : 0,
    "online_count" : 0,
    "disable_count" : 0,
    "offline_count" : 0
  },
  "additionalProp3" : {
    "env_name" : "string",
    "env_id" : 0,
    "app_name" : "string",
    "app_id" : 0,
    "online_count" : 0,
    "disable_count" : 0,
    "offline_count" : 0
  }
}
}
```

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Semantic or parameter error.
401	Unauthorized: No permissions.
403	Forbidden: Access forbidden.
404	Not Found: The requested resource is not found.

Error Codes

See [Error Codes](#).

5.1.5 Saving a Monitoring Item

Function

This API is used to save a monitoring item.

URI

POST /v1/apm2/openapi/apm-service/monitor-item-mgr/save-monitor-item-config

Request Parameters

Table 5-14 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM.
x-business-id	Yes	Long	Application ID.

Table 5-15 Request body parameters

Parameter	Mandatory	Type	Description
monitor_item_id	Yes	Long	Monitoring item ID.
interval	No	Integer	Collection interval, in seconds.
env_id	Yes	Integer	Environment ID.
config_value_list	No	Array of ConfigItem objects	Configuration item list.

Table 5-16 ConfigItem

Parameter	Mandatory	Type	Description
config_name	No	String	Configuration item name.
config_value	No	String	Configuration item value.
should_override	No	Boolean	Rewrite or not.

Response Parameters

Status code: 200

Table 5-17 Response body parameters

Parameter	Type	Description
flag	String	Whether the monitoring item is successfully modified.

Example Requests

Modify the information about monitoring item 4.

```
/v1/apm2/openapi/apm-service/monitor-item-mgr/save-monitor-item-config
{
  "interval" : 60,
  "config_value_list" : [ {
    "config_name" : "stackThreshold",
    "config_value" : 3,
    "should_override" : true
  } ],
  "monitor_item_id" : 4,
  "env_id" : "1"
}
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "flag" : "ok"
}
```

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Semantic or parameter error.
401	Unauthorized: No permissions.
403	Forbidden: Access forbidden.
404	Not Found: The requested resource is not found.

Error Codes

See [Error Codes](#).

5.1.6 Querying the Monitoring Item List

Function

This API is used to query the monitoring item list.

URI

POST /v1/apm2/openapi/apm-service/monitor-item-mgr/get-env-monitor-item-list

Request Parameters

Table 5-18 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM.
x-business-id	Yes	Long	Application ID.

Table 5-19 Request body parameters

Parameter	Mandatory	Type	Description
env_id	Yes	Long	Environment ID.
page	Yes	Integer	Page number.
page_size	No	Integer	Number of records on each page.

Response Parameters

Status code: 200

Table 5-20 Response body parameters

Parameter	Type	Description
monitor_item_list	Array of MonitorItem objects	Monitoring item list.
total_count	Integer	Total number.
totalPage	Integer	Total number of pages.

Table 5-21 MonitorItem

Parameter	Type	Description
id	Long	Monitoring item ID.
env_id	Long	Environment ID.
collector_id	Integer	Collector ID.
collector_name	String	Collector name.
display_name	String	Display name of the collector.
collect_interval	Integer	Collection interval.
disabled	Boolean	Disable or not.
status_change_user_id	String	ID of the user who last enabled or disabled the monitoring item.
status_change_user_name	String	Name of the user who last enabled or disabled the monitoring item.
status_change_time	String	Date when the monitoring item was enabled or disabled.
config_change_user_id	String	ID of the user who last modified the monitoring item configuration.
config_change_user_name	String	Name of the user who last modified the monitoring item configuration.
config_change_time	String	Time when the monitoring item configuration was modified.

Example Requests

Query the monitoring item list of environment 913.

```
/v1/apm2/openapi/apm-service/monitor-item-mgr/get-env-monitor-item-list
```

```
{
  "env_id": "913",
  "page": 1,
  "page_size": 10
}
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "monitor_item_list": [ {
    "id": 55150,
```

```

"env_id" : 913,
"collector_id" : 61,
"collector_name" : "Thread",
"display_name" : "Thread",
"collect_interval" : 60,
"disabled" : false,
"status_change_user_id" : null,
"status_change_user_name" : null,
"status_change_time" : null,
"config_change_user_id" : null,
"config_change_user_name" : null,
"config_change_time" : "2022-05-28 20:35:13"
}, {
  "id" : 32788,
  "env_id" : 913,
  "collector_id" : 74,
  "collector_name" : "HttpClient",
  "display_name" : "HttpClient monitoring",
  "collect_interval" : 60,
  "disabled" : false,
  "status_change_user_id" : "7df6775cd82b4963a1837368cbbe5a07",
  "status_change_user_name" : "paas_aom_apm",
  "status_change_time" : "2023-07-24 10:07:55",
  "config_change_user_id" : "7df6775cd82b4963a1837368cbbe5a07",
  "config_change_user_name" : "paas_aom_apm",
  "config_change_time" : "2024-05-31 15:04:31"
}],
"total_count" : 2,
"total_page" : 1
}

```

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Semantic or parameter error.
401	Unauthorized: No permissions.
403	Forbidden: Access forbidden.
404	Not Found: The requested resource is not found.

Error Codes

See [Error Codes](#).

5.1.7 Querying All Agents of an Application

Function

This API is used to query all Agents of an application.

URI

POST /v1/apm2/openapi/apm-service/agent-mgr/search

Request Parameters

Table 5-22 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM.
x-business-id	Yes	Long	Application ID.

Table 5-23 Request body parameters

Parameter	Mandatory	Type	Description
business_id	Yes	Long	Application ID.
env_id	No	Long	Environment ID.
status	No	String	Agent status. Enumeration values: <ul style="list-style-type: none"> • online • disable • offline
region	Yes	String	Region name.
order_by_status	No	String	Whether to sort data by collection status. If this parameter is left blank, data is not sorted by status. If this parameter is set to y, data is sorted by status.
page	Yes	Integer	Number of the page to be queried. Minimum: 1.
page_size	No	Integer	Maximum number of records displayed on each page.
keyword	No	String	Keyword.

Response Parameters

Status code: 200

Table 5-24 Response body parameters

Parameter	Type	Description
total_page	Integer	Total number of pages.
total_count	Integer	Total number of Agents.
online_count	Integer	Number of normal Agents.
offline_count	Integer	Number of Agents with abnormal heartbeats.
disable_count	Integer	Number of disabled Agents.
agent_info_list	Array of InstanceInfo objects	Agent address list.

Table 5-25 InstanceInfo

Parameter	Type	Description
instance_id	Long	Instance ID, which can be an input parameter for the API used to enable or disable collection for an instance.
business_name	String	Application name.
business_id	Long	Application ID.
app_name	String	Component name.
host_name	String	Host name.
instance_name	String	Instance name.
ip_address	String	Host IP address.
env_id	Long	Environment ID.
agent_version	String	JavaAgent version.
last_heartbeat	Long	Last heartbeat time, in milliseconds.
register_time	Long	Registration time, in milliseconds.
last_modify_user_id	String	ID of the user who last started or stopped the Agent.
instance_status	Integer	Instance status.
last_modify_user_name	String	Name of the user who last changed the instance status.

Parameter	Type	Description
last_modify_time	Long	Last modification time, in milliseconds.

Example Requests

Query all Agents of application 6 in region .

```
/v1/apm2/openapi/apm-service/agent-mgr/search
```

```
{
  "region": "",
  "page": 1,
  "page_size": 10,
  "business_id": 6
}
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "total_page": 1,
  "total_count": 1,
  "online_count": 1,
  "offline_count": 0,
  "disable_count": 0,
  "agent_info_list": [ {
    "instance_id": 5298,
    "business_name": "LubanApm",
    "business_id": null,
    "app_name": "apm-data-deliver/",
    "host_name": "szv*****1",
    "instance_name": "default",
    "ip_address": ".*.*.*",
    "env_id": 1105,
    "agent_version": "2.1.16",
    "last_heartbeat": 1669684246000,
    "register_time": 1657762671000,
    "last_modify_user_id": "70iWY*****VHVd",
    "instance_status": 0,
    "last_modify_user_name": "yWX",
    "last_modify_time": 1667839085000
  } ]
}
```

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Semantic or parameter error.
401	Unauthorized: No permissions.

Status Code	Description
403	Forbidden: Access forbidden.
404	Not Found: The requested resource is not found.

Error Codes

See [Error Codes](#).

5.1.8 Enabling or Disabling Collection for an Instance

Function

This API is used to enable or disable collection for a specified instance.

URI

POST /v1/apm2/openapi/apm-service/agent-mgr/change-status

Request Parameters

Table 5-26 Request header parameters

Parameter	Mandatory	Type	Description
x-business-id	Yes	Long	Application ID.
X-Auth-Token	Yes	String	User token obtained from IAM.

Table 5-27 Request body parameters

Parameter	Mandatory	Type	Description
instance_list	Yes	Array of integers	List of Agent IDs.
target_status	Yes	Integer	Expected Agent status. Options: 0: enabled. 1: disabled.
region	Yes	String	Region where the Agent is located.
env_id	No	Long	ID of the environment to which the Agent belongs.

Response Parameters

Status code: 200

Table 5-28 Response body parameters

Parameter	Type	Description
flag	String	ok: The operation is successful.

Example Requests

Disable data collection for instance 1097.

```
/v1/apm2/openapi/apm-service/agent-mgr/change-status  
  
{  
  "instance_list": [ 1097 ],  
  "target_status": 1,  
  "region": ""  
}
```

Example Responses

Status code: 200

OK: The request is successful.

```
{  
  "flag": "ok"  
}
```

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Semantic or parameter error.
401	Unauthorized: No permissions.
403	Forbidden: Access forbidden.
404	Not Found: The requested resource is not found.

Error Codes

See [Error Codes](#).

5.1.9 Deleting an Agent

Function

This API is used to delete an Agent.

URI

POST /v1/apm2/openapi/apm-service/agent-mgr/delete-agent

Request Parameters

Table 5-29 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM.
x-business-id	Yes	Long	Application ID.

Table 5-30 Request body parameters

Parameter	Mandatory	Type	Description
instance_list	Yes	Array of integers	Instance list.
region	Yes	String	Region name.
business_id	Yes	Long	Application ID.

Response Parameters

Status code: 200

Table 5-31 Response body parameters

Parameter	Type	Description
delete_status	Integer	Whether the deletion is successful. Options: 0 (successful) and 1 (failed).

Example Requests

Delete the offline Agent 1096 of application 1.

```
/v1/apm2/openapi/apm-service/agent-mgr/delete-agent
```

```
{  
  "instance_list" : [ 1096 ],  
  "region" : "",  
  "business_id" : 1  
}
```

Example Responses

Status code: 200

OK: The request is successful.

```
{  
  "delete_status" : 0  
}
```

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Semantic or parameter error.
401	Unauthorized: No permissions.
403	Forbidden: Access forbidden.
404	Not Found: The requested resource is not found.

Error Codes

See [Error Codes](#).

5.2 Region Query APIs

5.2.1 Querying the Regions Where APM Is Enabled

Function

This API is used to query the regions where APM is enabled.

URI

GET /v1/apm2/openapi/region/get-opened-region

Request Parameters

Table 5-32 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM.

Response Parameters

Status code: 200

Table 5-33 Response body parameters

Parameter	Type	Description
region_list	Array of Region objects	Region list

Table 5-34 Region

Parameter	Type	Description
region_id	String	Region ID.
region_name	String	Region name.
region	String	Region.
project_id	String	Enterprise project ID.
status	String	Region status.

Example Requests

There is no input parameter. Send a request to obtain the regions where APM is enabled.

```
/v1/apm2/openapi/region/get-opened-region
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "region_list" : [ {
    "region_id" : ""
```

```

"region_name" : "",
"region" : "",
"project_id" : "c6*****7b",
"status" : "NORMAL_STATUS"
}]
}

```

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Semantic or parameter error.
401	Unauthorized: No permissions.
403	Forbidden: Access forbidden.
404	Not Found: The requested resource is not found.

Error Codes

See [Error Codes](#).

5.2.2 Querying All Supported Regions

Function

This API is used to query all supported regions.

URI

GET /v1/apm2/openapi/region/get-all-supported-region

Request Parameters

Table 5-35 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM.

Response Parameters

Status code: 200

Table 5-36 Response body parameters

Parameter	Type	Description
region_list	Array of Region objects	Region list

Table 5-37 SupportedRegion

Parameter	Type	Description
region_id	String	Region ID.
region_name	String	Region name.
region	String	Region.

Example Requests

There is no input parameter. Send a request to query supported regions.

```
/v1/apm2/openapi/region/all-supported-region
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "region_list": [ {
    "region_id": "",
    "region_name": "",
    "region": "",
    "project_id": null,
    "status": null
  }, {
    "region_id": "mock-region",
    "region_name": "",
    "region": ""
  }
]
```

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Semantic or parameter error.
401	Unauthorized: No permissions.

Status Code	Description
403	Forbidden: Access forbidden.
404	Not Found: The requested resource is not found.

Error Codes

See [Error Codes](#).

5.3 Resource Management APIs

5.3.1 Querying Environment Tags

Function

This API is used to query environment tags.

URI

POST /v1/apm2/openapi/cmdb/tag/get-env-tag-list

Request Parameters

Table 5-38 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM.
x-business-id	Yes	Long	Application ID.

Table 5-39 Request body parameters

Parameter	Mandatory	Type	Description
business_id	Yes	Long	Application ID.
keyword	No	String	Keyword.
page_enable	No	Boolean	Whether pagination is supported.
page_number	No	Integer	Number of records on each page.

Parameter	Mandatory	Type	Description
page_size	No	Integer	Current page number.

Response Parameters

Status code: 200

Table 5-40 Response body parameters

Parameter	Type	Description
env_tags	Array of CmdbTagEntity objects	Environment tag list.
total_count	Integer	Total number.

Table 5-41 CmdbTagEntity

Parameter	Type	Description
name	String	Environment tag name.
business_id	Long	Application ID.
uuid	String	Unique ID of an environment.
descp	String	Description.
creator_id	Long	Creator ID.
env_id_list	Array of integers	Environment ID list.
id	Long	Environment tag ID.
gmt_create	String	Creation time.
gmt_modify	String	Update time.

Example Requests

Query the environment tag of application 162. The query result is displayed on the same page.

```
/v1/apm2/openapi/cmdb/tag/get-env-tag-list
{
  "business_id" : 162,
  "page_enable" : false
}
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "env_tags" : [ {
    "name" : "",
    "businessId" : 162,
    "uuid" : "e9c8f632*****cbfcfa148c0",
    "descp" : null,
    "creatorId" : null,
    "envIdList" : null,
    "id" : 72,
    "gmt_create" : "2022-10-25T02:18:44.000+00:00",
    "gmt_modify" : "2022-10-25T02:18:44.000+00:00"
  } ],
  "total_count" : 1
}
```

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Semantic or parameter error.
401	Unauthorized: No permissions.
403	Forbidden: Access forbidden.
404	Not Found: The requested resource is not found.

Error Codes

See [Error Codes](#).

5.3.2 Obtaining an Application Tree

Function

This API is used to obtain an application tree.

URI

GET /v1/apm2/openapi/cmdb/topology-trees/get-topology-trees

Table 5-42 Query Parameters

Parameter	Mandatory	Type	Description
region_id	No	String	Region ID.

Parameter	Mandatory	Type	Description
business_id	Yes	Long	Application ID.
env_tag_id	No	Long	Environment tag ID.
env_keyword	No	String	Environment keyword.

Request Parameters

Table 5-43 Request header parameters

Parameter	Mandatory	Type	Description
x-business-id	Yes	Long	Application ID.
X-Auth-Token	Yes	String	User token obtained from IAM.

Response Parameters

Status code: 200

Table 5-44 Response body parameters

Parameter	Type	Description
topology_tree	TopologyTree object	Topology tree information.

Table 5-45 TopologyTree

Parameter	Type	Description
tree	Array of TreeNode objects	Topology tree node.

Table 5-46 TreeNode

Parameter	Type	Description
id	String	ID of a node in the topology tree.
parent	String	Parent node in the topology tree.
real_id	Long	Actual ID of a node in the topology tree.

Parameter	Type	Description
name	String	Name of a node in the topology tree.
display_name	String	Display name of a node in the topology tree.
app_name	String	Component name.
app_id	Long	Component ID.
is_admin	Boolean	Whether the node is a management node.
is_root	Boolean	Whether the node is a root node.
business_id	Long	Application ID.
node_type	String	Node type. Enumeration values: <ul style="list-style-type: none"> ● BUSINESS ● SUB_BUSINESS ● APPLICATION ● ENVIRONMENT
region	String	Region.
is_default	Boolean	Whether the node is a default node.

Example Requests

Obtain the topology tree of application 162.

```
/v1/apm2/openapi/cmdb/topology-trees/get-topology-trees?business_id=162
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "topology_tree" : {
    "tree" : [ {
      "admin" : false,
      "default" : false,
      "id" : "162-55-11",
      "parent" : "162-55",
      "real_id" : 11,
      "name" : "apm-pu-task",
      "display_name" : "apm-pu-task",
      "app_name" : null,
      "app_id" : null,
      "is_admin" : false,
      "is_root" : false,
      "business_id" : 162,
      "node_type" : "APPLICATION",
      "is_default" : false,
      "region" : ""
    } ],
  }
}
```

```

"admin" : false,
"default" : false,
"id" : "162-55-11-11",
"parent" : "162-55-11",
"real_id" : 11,
"name" : "suzhou-roma-2",
"display_name" : "",
"app_name" : "apm-pu-task",
"app_id" : 11,
"is_admin" : false,
"is_root" : false,
"business_id" : 162,
"node_type" : "ENVIRONMENT",
"is_default" : false,
"region" : ""
  ]]
}
}

```

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Semantic or parameter error.
401	Unauthorized: No permissions.
403	Forbidden: Access forbidden.
404	Not Found: The requested resource is not found.

Error Codes

See [Error Codes](#).

5.3.3 Obtaining the Component List

Function

This API is used to obtain the component list.

URI

GET /v1/apm2/openapi/cmdb/apps/get-apps

Table 5-47 Query Parameters

Parameter	Mandatory	Type	Description
business_id	Yes	Long	Application ID.

Request Parameters

Table 5-48 Request header parameters

Parameter	Mandatory	Type	Description
x-business-id	Yes	Long	Application ID.
X-Auth-Token	Yes	String	User token obtained from IAM.

Response Parameters

Status code: 200

Table 5-49 Response body parameters

Parameter	Type	Description
apps	Array of AppNodeModel objects	Component information list.

Table 5-50 CmdbAppNodeModel

Parameter	Type	Description
id	Long	Component ID.
name	String	Component name.
business_id	Long	Application ID.
sub_business_id	Long	Sub-application ID.
data_source	String	Data source.

Example Requests

Query the component list of application 162.

```
/v1/apm2/openapi/cmdb/apps/get-apps?business_id=162
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "apps": [ {
```

```

    "id" : 11,
    "name" : "apm-pu-task",
    "business_id" : 162,
    "sub_business_id" : 55,
    "data_source" : "APM"
  } ]
}

```

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Semantic or parameter error.
401	Unauthorized: No permissions.
403	Forbidden: Access forbidden.
404	Not Found: The requested resource is not found.

Error Codes

See [Error Codes](#).

5.3.4 Obtaining the Environment List of a Component

Function

This API is used to obtain the environment list of a component.

URI

GET /v1/apm2/openapi/cmdb/envs/get-app-envs

Table 5-51 Query Parameters

Parameter	Mandatory	Type	Description
app_id	Yes	Long	Component ID.

Request Parameters

Table 5-52 Request header parameters

Parameter	Mandatory	Type	Description
x-business-id	Yes	Long	Application ID.

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM.

Response Parameters

Status code: 200

Table 5-53 Response body parameters

Parameter	Type	Description
envs	Array of EnvNodeModel objects	Environment information list.

Table 5-54 EnvNodeModel

Parameter	Type	Description
id	Long	Environment ID.
gmt_create	String	Creation time.
gmt_modify	String	Update time.
app_id	Long	Component ID.
business_name	String	Application name.
inner_domain_id	Integer	Tenant ID.
name	String	Environment name.
is_default	Boolean	Whether the environment is a default environment.
app_name	String	Component name.
business_id	Long	Application ID.
region	String	Region.
data_source	String	Data source.

Example Requests

Obtain the environment list of component 11.

/v1/apm2/openapi/cmdb/envs/get-app-envs?app_id=11

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "envs": [
    {
      "default": false,
      "id": 11,
      "gmt_create": null,
      "gmt_modify": null,
      "app_id": 11,
      "business_name": null,
      "inner_domain_id": null,
      "name": "",
      "is_default": false,
      "app_name": null,
      "business_id": 162,
      "region": "",
      "data_source": "APM",
    }
  ]
}
```

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Semantic or parameter error.
401	Unauthorized: No permissions.
403	Forbidden: Access forbidden.
404	Not Found: The requested resource is not found.

Error Codes

See [Error Codes](#).

5.3.5 Deleting a Component Based on the Component ID

Function

This API is used to delete a specified component.

URI

DELETE /v1/apm2/openapi/cmdb/apps/delete-app/{application_id}

Table 5-55 Path Parameters

Parameter	Mandatory	Type	Description
application_id	Yes	Long	Component ID.

Request Parameters

Table 5-56 Request header parameters

Parameter	Mandatory	Type	Description
x-business-id	Yes	Long	Application ID, which is used for authentication.
X-Auth-Token	Yes	String	User token obtained from IAM.

Response Parameters

Status code: 200

Table 5-57 Response body parameters

Parameter	Type	Description
ok	String	ok: The deletion is successful.

Example Requests

Delete component 1.

```
/v1/apm2/openapi/cmdb/apps/delete-app/1
```

Example Responses

Status code: 200

Result returned when you delete an application. ok: The deletion is successful.

```
{'ok':'ok'}
```

Status Codes

Status Code	Description
200	Result returned when you delete an application. ok: The deletion is successful.

Status Code	Description
400	Bad Request: Semantic or parameter error.
401	Unauthorized: No permissions.
403	Forbidden: Access forbidden.
404	Not Found: The requested resource is not found.

Error Codes

See [Error Codes](#).

5.3.6 Querying the Details About an Application

Function

This API is used to query the details about an application.

URI

GET /v1/apm2/openapi/cmdb/business/get-business-detail/{business_id}

Table 5-58 Path Parameters

Parameter	Mandatory	Type	Description
business_id	Yes	Long	Application ID.

Request Parameters

Table 5-59 Request header parameters

Parameter	Mandatory	Type	Description
x-business-id	Yes	Long	Application ID.
X-Auth-Token	Yes	String	User token obtained from IAM.

Response Parameters

Status code: 200

Table 5-60 Response body parameters

Parameter	Type	Description
id	Long	Application ID.
gmt_create	String	Creation time.
gmt_modify	String	Update time.
default	Boolean	Whether an application is the default application.
display_name	String	Application name.
name	String	Display name of an application.
is_default	Boolean	Whether an application is the default application.
inner_domain_id	Integer	Internal tenant ID.
eps_id	String	Enterprise project ID.
creator_id	Long	ID of the creator.
descp	String	Application description.
create_time	String	Creation time.
modify_time	String	Update time.
creator_name	String	Name of the creator.

Example Requests

Query details about application 1.

```
/v1/apm2/openapi/cmdb/business/get-business-detail/1
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "default" : false,
  "id" : 1,
  "gmt_create" : null,
  "gmt_modify" : null,
  "name" : "LubanApm",
  "display_name" : "LubanApm",
  "is_default" : false,
  "inner_domain_id" : 3,
  "eps_id" : "",
  "creator_id" : null,
  "uuid" : "9496a43*****7f4da0c1926",
  "descp" : ""
}
```

```
"create_time" : "2021-07-14 21:40:32",
"create_time_timestamp" : 1626270032000,
"modify_time" : "2021-07-14 21:40:47",
"modify_time_timestamp" : 1626270047000,
"creator_name" : "test"
}
```

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Semantic or parameter error.
401	Unauthorized: No permissions.
403	Forbidden: Access forbidden.
404	Not Found: The requested resource is not found.

Error Codes

See [Error Codes](#).

5.3.7 Querying the Details About a Sub-application

Function

This API is used to query the details about a sub-application.

URI

GET /v1/apm2/openapi/cmdb/sub-business/get-sub-business-detail/
{sub_business_id}

Table 5-61 Path Parameters

Parameter	Mandatory	Type	Description
sub_business_id	Yes	Long	Sub-application ID.

Request Parameters

Table 5-62 Request header parameters

Parameter	Mandatory	Type	Description
x-business-id	Yes	Long	Application ID.

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM.

Response Parameters

Status code: 200

Table 5-63 Response body parameters

Parameter	Type	Description
id	Long	Sub-application ID.
gmt_create	String	Creation time.
gmt_modify	String	Update time.
parent_id	Long	Parent node ID.
name	String	Sub-application name.
display_name	String	Display name of the sub-application.
business_id	Long	ID of the application.
inner_domain_id	Integer	Internal tenant ID.
creator_id	Long	ID of the creator.
uuid	String	UUID of the application.
descp	String	Application description.
create_time	String	Creation time.
modify_time	String	Update time.
creator_name	String	Name of the creator.

Example Requests

Query details about sub-application 8.

```
/v1/apm2/openapi/cmdb/sub-business/get-sub-business-detail/8
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "name" : "App",
```

```

"uuid" : "6e59*****36072",
"descp" : null,
"id" : 8,
"gmt_create" : null,
"gmt_modify" : null,
"display_name" : "App",
"parent_id" : null,
"business_id" : 6,
"inner_domain_id" : 3,
"creator_id" : null,
"create_time" : "2021-07-14 22:33:33",
"modify_time" : "2021-07-14 22:33:33",
"creator_name" : "API"
}
    
```

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Semantic or parameter error.
401	Unauthorized: No permissions.
403	Forbidden: Access forbidden.
404	Not Found: The requested resource is not found.

Error Codes

See [Error Codes](#).

5.4 View APIs

5.4.1 Querying Monitoring Item Configurations

Function

This API is used to query monitoring item configurations.

URI

GET /v1/apm2/openapi/view/config/get-monitor-item-view-config

Table 5-64 Query Parameters

Parameter	Mandatory	Type	Description
env_id	Yes	Long	Environment ID.
collector_id	Yes	Long	Collector ID.

Request Parameters

Table 5-65 Request header parameters

Parameter	Mandatory	Type	Description
x-business-id	Yes	Long	Application ID.
X-Auth-Token	Yes	String	User token obtained from IAM.

Response Parameters

Status code: 200

Table 5-66 Response body parameters

Parameter	Type	Description
title	String	Title.
collector_name	String	Collector name.
view_row_list	Array of ViewRow objects	List of views. Each list represents a list of views.
style	String	Type.

Table 5-67 ViewRow

Parameter	Type	Description
view_list	Array of ViewBase objects	View list, which contains multiple views. Views are adapted based on their actual lengths.
title	String	Title.

Table 5-68 ViewBase

Parameter	Type	Description
collector_name	String	Collector name.
metric_set	String	Name of the metric set corresponding to the view.

Parameter	Type	Description
title	String	Title to be displayed.
table_direction	String	Direction of the table heading. Options: H (default): horizontal. V: vertical. Enumeration values: <ul style="list-style-type: none"> • H • V
group_by	String	Grouping rule.
filter	String	Filter list model.
field_item_list	Array of FieldItem objects	Field list models to be displayed.
span	Boolean	Span information.
span_field	String	Attribute of the span field.
order_by	String	Sorting rule.
latest	String	Whether to display only the latest data.
view_type	String	View type. Enumeration values: <ul style="list-style-type: none"> • trend • sumtable • rawtable

Table 5-69 FieldItem

Parameter	Type	Description
function	String	Expression.
as	String	As.
default_value	String	Default value.
trace	Boolean	Trace or not.
precision	Integer	Percentage.
unit	String	Unit.
visible	Boolean	Visible or not.

Example Requests

Query the configuration of the monitoring item whose environment ID is 11 and collector ID is 50.

```
/v1/apm2/openapi/view/config/get-monitor-item-view-config?env_id=11&collector_id=50
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "title" : "Url",
  "collector_name" : "Url",
  "view_row_list" : [ {
    "view_list" : [ {
      "latest" : null,
      "span" : true,
      "collector_name" : "Url",
      "metric_set" : "url",
      "title" : "URL Summary",
      "table_direction" : null,
      "group_by" : "url,method",
      "filter" : "",
      "field_item_list" : [ {
        "trace" : null,
        "function" : "SUM(invokeCount)",
        "as" : "Number of calls",
        "default_value" : null,
        "precision" : null,
        "unit" : null,
        "visible" : true
      }, {
        "trace" : null,
        "function" : "SUM(range6)",
        "as" : "10s-n",
        "default_value" : null,
        "precision" : null,
        "unit" : null,
        "visible" : false
      }
    ],
    "span_field" : "source",
    "order_by" : null,
    "view_type" : "sumtable"
  } ],
  "title" : ""
} ],
"style" : "tab"
}
```

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Semantic or parameter error.
401	Unauthorized: No permissions.
403	Forbidden: Access forbidden.

Status Code	Description
404	Not Found: The requested resource is not found.

Error Codes

See [Error Codes](#).

5.4.2 Querying the Trace Topology

Function

This API is used to query the trace topology.

URI

GET /v1/apm2/openapi/view/trace/topology

Table 5-70 Query Parameters

Parameter	Mandatory	Type	Description
trace_id	Yes	String	Trace ID.

Request Parameters

Table 5-71 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM.

Response Parameters

Status code: 200

Table 5-72 Response body parameters

Parameter	Type	Description
global_trace_id	String	Global trace ID.

Parameter	Type	Description
line_list	Array of TraceTopologyLine objects	List of lines that indicate invocations between components.
node_list	Array of TraceTopologyNode objects	List of component nodes.

Table 5-73 TraceTopologyLine

Parameter	Type	Description
start_node_id	Long	Start node ID.
end_node_id	Long	End node ID.
span_id	String	Span ID.
client_info	TraceTopologyLineInfo object	Information displayed on the client.
server_info	TraceTopologyLineInfo object	Server information.
id	String	ID.
hint	String	Line prompt.

Table 5-74 TraceTopologyLineInfo

Parameter	Type	Description
start_time	Long	Start time.
time_used	Long	Time required.
argument	String	Parameters, such as the invoked URL.
event_id	String	ID of an event.

Table 5-75 TraceTopologyNode

Parameter	Type	Description
node_id	Long	Node ID.

Parameter	Type	Description
node_name	String	Node name.
hint	String	Node prompt.

Example Requests

Query the topology of trace 16-1666684411910-1326.

```
/v1/apm2/openapi/view/trace/topology?region=&trace_id=16-1666684411910-1326
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "node_list": [ {
    "node_id": 11,
    "node_name": "apm-pu-task:",
    "hint": null
  } ],
  "line_list": [ {
    "start_node_id": null,
    "end_node_id": 11,
    "span_id": "1",
    "client_info": null,
    "server_info": {
      "start_time": 1666684411910,
      "time_used": 1,
      "argument": "(GET)/apm2/health/v1/health-check(200)",
      "event_id": "1"
    },
    "hint": "(1)((GET)/apm2/health/v1/health-check(200))",
    "id": "1+1"
  } ],
  "global_trace_id": null
}
```

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Semantic or parameter error.
401	Unauthorized: No permissions.
403	Forbidden: Access forbidden.
404	Not Found: The requested resource is not found.

Error Codes

See [Error Codes](#).

5.4.3 Querying Event Details

Function

This API is used to query event details.

URI

GET /v1/apm2/openapi/view/trace/get-event-detail

Table 5-76 Query Parameters

Parameter	Mandatory	Type	Description
trace_id	Yes	String	Trace ID.
span_id	Yes	String	Span ID.
event_id	Yes	String	Event ID.
env_id	Yes	Long	Environment ID.

Request Parameters

Table 5-77 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM.

Response Parameters

Status code: 200

Table 5-78 Response body parameters

Parameter	Type	Description
event_info	SpanEventInfo object	Event details.

Table 5-79 SpanEventInfo

Parameter	Type	Description
env_name	String	Environment name.
app_name	String	Component name.
indent	Integer	Indent.
region	String	Region.
host_name	String	Host name.
ip_address	String	IP address.
instance_name	String	Instance name.
event_id	String	ID of an event in a specific span. Example: 1-1-2.
next_spanId	String	Source event ID for the next span invocation.
source_event_id	String	Event ID of the invoker.
method	String	Method.
children_event_count	Integer	Number of sub-events.
discard	Array of DiscardInfo objects	Number of discarded sub-events. Use key to specify the sub-event type.
argument	String	Parameters to be displayed, which vary according to event types.
attachment	Map<String,String>	Attachment in the registration information.
global_trace_id	String	Virtual trace ID.
global_path	String	Path that a virtual trace ID passes through.
trace_id	String	Trace ID.
span_id	String	Span ID.
env_id	Long	Environment ID.
instance_id	Long	Instance ID.
app_id	Long	Component ID.
biz_id	Long	Application ID.
domain_id	Integer	Tenant ID.

Parameter	Type	Description
source	String	This parameter exists only when a root event exists.
real_source	String	URL that is actually invoked. This parameter exists when a root event exists.
start_time	Long	Start time.
time_used	Long	Time required.
code	Integer	Status code, which is valid for HTTP invocation.
class_name	String	Class name.
is_async	Boolean	Whether the invocation is asynchronous.
tags	Map<String,String>	Custom parameters, header or body parameters, httpMethod, bizCode, or parameters that may be added later.
has_error	Boolean	Whether an error occurs. This parameter is used in span scenarios.
error_reasons	String	Error cause.
type	String	Type, such as mysql and kafka.
http_method	String	http_method in tags. Only the URL monitoring item has this parameter.
biz_code	String	Status code collected.
id	String	Span ID.

Table 5-80 DiscardInfo

Parameter	Type	Description
type	String	Type.
count	Integer	Quantity.
totalTime	Long	Total duration.

Example Requests

Obtain the details about the event whose environment ID is 11, trace_id is 16-1666684411910-1326, span_id is 1, and event_id is 11.

```
/v1/apm2/openapi/view/trace/get-event-detail?
env_id=11&event_id=1&span_id=1&trace_id=16-1666684411910-1326
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "event_info" : {
    "global_trace_id" : "16-1666684411910-1326",
    "global_path" : null,
    "trace_id" : "16-1666684411910-1326",
    "span_id" : "1",
    "env_id" : 11,
    "instance_id" : 16,
    "app_id" : 11,
    "biz_id" : 162,
    "domain_id" : 1,
    "source" : "/apm2/health/v1/health-check",
    "real_source" : "/apm2/health/v1/health-check",
    "start_time" : 1666684411910,
    "time_used" : 1,
    "code" : 200,
    "class_name" : "org/apache/catalina/core/StandardHostValve",
    "is_async" : false,
    "tags" : {
      "httpMethod" : "GET",
      "remoteAddr" : "***.***.***.*"
    },
    "has_error" : false,
    "error_reasons" : null,
    "type" : "Tomcat",
    "http_method" : "GET",
    "biz_code" : null,
    "event_id" : "1",
    "next_spanId" : null,
    "source_event_id" : null,
    "method" : "invoke"
    "children_event_count" : 0,
    "discard" : [ ],
    "argument" : "(GET)/apm2/health/v1/health-check(200)",
    "attachment" : {
      "bizName" : "LubanApm",
      "instanceName" : "apm2-apm-pu-task-6b5bbfc84d-gtrrs:***.***:default",
      "envName" : "",
      "appName" : "apm-pu-task",
      "projectId" : "420a7f6d790349c1bbeec18aefe36404",
      "domainId" : "f9b2ea53c6eb48078435d499a3b03419"
    },
    "host_name" : "apm2-apm-pu-task-6b5bbfc84d-gtrrs",
    "ip_address" : "***.***.*",
    "instance_name" : "default",
    "region" : "",
    "env_name" : "",
    "app_name" : "apm-pu-task",
    "indent" : 0,
    "id" : "1+1"
  }
}
```

Status Codes

Status Code	Description
200	OK: The request is successful.

Status Code	Description
400	Bad Request: Semantic or parameter error.
401	Unauthorized: No permissions.
403	Forbidden: Access forbidden.
404	Not Found: The requested resource is not found.

Error Codes

See [Error Codes](#).

5.4.4 Querying Span Data

Function

This API is used to query span data.

URI

POST /v1/apm2/openapi/view/trace/span-search

Request Parameters

Table 5-81 Request header parameters

Parameter	Mandatory	Type	Description
x-business-id	Yes	Long	Application ID.
X-Auth-Token	Yes	String	User token obtained from IAM.

Table 5-82 Request body parameters

Parameter	Mandatory	Type	Description
region	Yes	String	Region name.
order_param	No	OrderParam object	Sorting parameter.
real_source_full_match	No	Boolean	Whether exact search is performed.
source_full_match	No	Boolean	Full match search.

Parameter	Mandatory	Type	Description
tags_content	No	String	Search by the header, body, custom parameter, or other tag keywords.
start_time_string	No	String	Start time.
end_time_string	No	String	End time.
time_used_min	No	Long	Minimum duration.
time_used_max	No	String	Maximum duration.
contain_tags_content	No	Boolean	Whether the search result contains tag details.
page_size	No	Integer	Number of records returned on each page.
page	No	Integer	Number of the page on which data is queried. By default, data on the first page is queried.
parameters	No	String	Parameter.
codes	No	Array of integers	Status codes in string format. Multiple status codes can be queried at a time.
global_trace_id	No	String	Virtual trace ID, which may correspond to multiple actual trace IDs. The virtual trace ID will be transferred from the beginning to the end.
global_path	No	String	Path that a virtual trace ID passes through.
trace_id	No	String	Global ID generated when the root span is invoked, which will be transparently transmitted during invocations.
span_id	No	String	ID of an RPC call. For a root span to be invoked, the value is 1. For the next spans to be invoked, the values are 1-1, 1-2, and so on.
env_id	No	Long	Environment ID.

Parameter	Mandatory	Type	Description
instance_id	No	Long	Instance ID.
app_id	No	Long	Component ID.
biz_id	Yes	Long	Application ID.
domain_id	No	Integer	Tenant ID.
source	No	String	This parameter exists only when a root event exists.
real_source	No	String	URL that is actually invoked. This parameter exists when a root event exists.
start_time	No	Long	Start time.
time_used	No	Long	Time required.
code	No	Integer	Status code, which is valid for HTTP invocation.
class_name	No	String	Class name.
is_async	No	Boolean	Whether the invocation is asynchronous.
tags	No	Map<String,String>	Custom parameters, header or body parameters, httpMethod, bizCode, or parameters that may be added later.
has_error	No	Boolean	Whether an error occurs.
error_reasons	No	String	Error type.
type	No	String	Type.
http_method	No	String	http_method in tags. Only the URL monitoring item has this parameter.
biz_code	No	String	Status code collected.

Table 5-83 OrderParam

Parameter	Mandatory	Type	Description
field	No	String	timeUsed: response time. startTime: generation time.

Parameter	Mandatory	Type	Description
order	No	String	ASC: ascending order. DESC: descending order. Ensure that field is also set. Enumeration values: <ul style="list-style-type: none"> • ASC • DESC

Response Parameters

Status code: 200

Table 5-84 Response body parameters

Parameter	Type	Description
total	Integer	Total number of returned records.
span_info_list	Array of ClientSpanInfo objects	Span information.

Table 5-85 ClientSpanInfo

Parameter	Type	Description
global_trace_id	String	Virtual trace ID.
global_path	String	Path that a virtual trace ID passes through.
trace_id	String	Trace ID.
span_id	String	Span ID.
env_id	Long	Environment ID.
instance_id	Long	Instance ID.
app_id	Long	Component ID.
biz_id	Long	Application ID.
domain_id	Integer	Tenant ID.
source	String	This parameter exists only when a root event exists.
real_source	String	URL that is actually invoked. This parameter exists when a root event exists.

Parameter	Type	Description
start_time	Long	Start time.
time_used	Long	Time required.
code	Integer	Status code, which is valid for HTTP invocation.
class_name	String	Class name.
is_async	Boolean	Whether the invocation is asynchronous.
tags	Map<String,String>	Custom parameters, header or body parameters, httpMethod, bizCode, or parameters that may be added later.
has_error	Boolean	Whether an error is reported.
error_reasons	String	Error cause.
type	String	Type, such as mysql and kafka.
http_method	String	http_method in tags. Only the URL monitoring item has this parameter.
biz_code	String	Status code collected.

Example Requests

Query the span data whose region ID is "" and application ID is 162 in a specified time range.

```
/v1/apm2/openapi/view/trace/span-search
{
  "region": "",
  "biz_id": 162,
  "page": 1,
  "page_size": 15,
  "start_time_string": "2022-10-25 10:03:16",
  "end_time_string": "2022-10-25 10:23:16",
  "codes": []
}
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "total": 2,
  "span_info_list": [ {
    "async": false,
    "global_trace_id": "16-1666684411910-1326",
    "global_path": null,
    "trace_id": "16-1666684411910-1326",
    "span_id": "1",
    "env_id": 11,
    "instance_id": 16,
    "app_id": 11,
  }
]
```

```

    "biz_id" : 162,
    "domain_id" : 1,
    "source" : "/apm2/health/v1/health-check",
    "real_source" : "/apm2/health/v1/health-check",
    "start_time" : 1666684411910,
    "time_used" : 1,
    "code" : 200,
    "class_name" : "org/apache/catalina/core/StandardHostValve",
    "is_async" : false,
    "tags" : { },
    "has_error" : false,
    "error_reasons" : null,
    "type" : "Tomcat",
    "http_method" : "GET",
    "estimated_start_time" : null,
    "estimated_end_time" : null,
    "biz_code" : null,
    "env_name" : "",
    "instance_name" : "apm2-apm-pu-task-6b5bbfc84d-gtrrs(**.*)",
    "app_name" : "apm-pu-task",
    "region" : ""
  }, {
    "async" : false,
    "global_trace_id" : "15-1666684400114-1332",
    "global_path" : null,
    "trace_id" : "15-1666684400114-1332",
    "span_id" : "1",
    "env_id" : 11,
    "instance_id" : 15,
    "app_id" : 11,
    "biz_id" : 162,
    "domain_id" : 1,
    "source" : "/apm2/health/v1/health-check",
    "real_source" : "/apm2/health/v1/health-check",
    "start_time" : 1666684400114,
    "time_used" : 0,
    "code" : 200,
    "class_name" : "org/apache/catalina/core/StandardHostValve",
    "is_async" : false,
    "tags" : { },
    "has_error" : false,
    "error_reasons" : null,
    "type" : "Tomcat",
    "http_method" : "GET",
    "estimated_start_time" : null,
    "estimated_end_time" : null,
    "biz_code" : null,
    "env_name" : "",
    "instance_name" : "apm2-apm-pu-task-6b5bbfc84d-rxlr(**.*)",
    "app_name" : "apm-pu-task",
    "region" : ""
  }
}

```

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Semantic or parameter error.
401	Unauthorized: No permissions.
403	Forbidden: Access forbidden.

Status Code	Description
404	Not Found: The requested resource is not found.

Error Codes

See [Error Codes](#).

5.4.5 Obtaining All Data of a Trace

Function

This API is used to obtain all data of a trace.

URI

GET /v1/apm2/openapi/view/trace/get-trace-events

Table 5-86 Query Parameters

Parameter	Mandatory	Type	Description
trace_id	Yes	String	Trace ID.

Request Parameters

Table 5-87 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM.

Response Parameters

Status code: 200

Table 5-88 Response body parameters

Parameter	Type	Description
span_event_list	Array of SpanEventInfo objects	Span event information.

Table 5-89 SpanEventInfo

Parameter	Type	Description
env_name	String	Environment name.
app_name	String	Component name.
indent	Integer	Indent.
region	String	Region.
host_name	String	Host name.
ip_address	String	IP address.
instance_name	String	Instance name.
event_id	String	ID of an event in a specific span. Example: 1-1-2.
next_spanId	String	Source event ID for the next span invocation.
source_event_id	String	Event ID of the invoker.
method	String	Method.
children_event_count	Integer	Number of sub-events.
discard	Array of DiscardInfo objects	Number of discarded sub-events. Use key to specify the sub-event type.
argument	String	Parameters to be displayed, which vary according to event types.
attachment	Map<String,String>	Attachment in the registration information.
global_trace_id	String	Virtual trace ID.
global_path	String	Path that a virtual trace ID passes through.
trace_id	String	Trace ID.
span_id	String	Span ID.
env_id	Long	Environment ID.
instance_id	Long	Instance ID.
app_id	Long	Component ID.
biz_id	Long	Application ID.
domain_id	Integer	Tenant ID.

Parameter	Type	Description
source	String	This parameter exists only when a root event exists.
real_source	String	URL that is actually invoked. This parameter exists when a root event exists.
start_time	Long	Start time.
time_used	Long	Time required.
code	Integer	Status code, which is valid for HTTP invocation.
class_name	String	Class name.
is_async	Boolean	Whether the invocation is asynchronous.
tags	Map<String,String>	Custom parameters, header or body parameters, httpMethod, bizCode, or parameters that may be added later.
has_error	Boolean	Whether an error occurs. This parameter is used in span scenarios.
error_reasons	String	Error cause.
type	String	Type, such as mysql and kafka.
http_method	String	http_method in tags. Only the URL monitoring item has this parameter.
biz_code	String	Status code collected.
id	String	Span ID.

Table 5-90 DiscardInfo

Parameter	Type	Description
type	String	Type.
count	Integer	Quantity.
totalTime	Long	Total duration.

Example Requests

Obtain all data of trace 16-1666684411910-1326.

```
/v1/apm2/openapi/view/trace/get-trace-events?trace_id=16-1666684411910-1326
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "span_event_list" : [ {
    "global_trace_id" : "16-1666684411910-1326",
    "global_path" : null,
    "trace_id" : "16-1666684411910-1326",
    "span_id" : "1",
    "env_id" : 11,
    "instance_id" : 16,
    "app_id" : 11,
    "biz_id" : 162,
    "domain_id" : 1,
    "source" : "/apm2/health/v1/health-check",
    "real_source" : "/apm2/health/v1/health-check",
    "start_time" : 1666684411910,
    "time_used" : 1,
    "code" : 200,
    "class_name" : "org/apache/catalina/core/StandardHostValve",
    "is_async" : false,
    "tags" : { },
    "has_error" : false,
    "error_reasons" : null,
    "type" : "Tomcat",
    "http_method" : "GET",
    "biz_code" : null,
    "event_id" : "1",
    "next_spanId" : null,
    "source_event_id" : null,
    "method" : "invoke",
    "children_event_count" : 0,
    "discard" : [ ],
    "argument" : "(GET)/apm2/health/v1/health-check(200)",
    "attachment" : null,
    "host_name" : "apm2-apm-pu-task-6b5bbfc84d-gtrrs",
    "ip_address" : " *.*.*",
    "instance_name" : "default",
    "region" : "",
    "env_name" : "",
    "app_name" : "apm-pu-task",
    "indent" : 0,
    "id" : "1+1"
  } ]
}
```

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Semantic or parameter error.
401	Unauthorized: No permissions.
403	Forbidden: Access forbidden.
404	Not Found: The requested resource is not found.

Error Codes

See [Error Codes](#).

5.4.6 Obtaining the Trend Graph

Function

This API is used to obtain the trend graph.

URI

POST /v1/apm2/openapi/view/metric/trend

Request Parameters

Table 5-91 Request header parameters

Parameter	Mandatory	Type	Description
x-business-id	Yes	Long	Application ID.
X-Auth-Token	Yes	String	User token obtained from IAM.

Table 5-92 Request body parameters

Parameter	Mandatory	Type	Description
view_config	No	TrendView object	View configuration information.
instance_id	No	Long	Instance ID.
monitor_item_id	No	Long	Monitoring item ID.
env_id	No	Long	Environment ID.
start_time	No	String	Start time.
end_time	No	String	End time.

Table 5-93 TrendView

Parameter	Mandatory	Type	Description
view_type	No	String	View type. Enumeration values: <ul style="list-style-type: none">• trend• sumtable• rawtable

Parameter	Mandatory	Type	Description
collector_name	No	String	Collector name.
metric_set	No	String	Name of the metric set corresponding to the view.
title	No	String	Title to be displayed.
table_direction	No	String	Direction of the table heading. Options: H (default): horizontal. V: vertical. Enumeration values: <ul style="list-style-type: none">• H• V
group_by	No	String	Group.
filter	No	String	Filter list model.
field_item_list	No	Array of FieldItem objects	Field list models to be displayed.
span	No	Boolean	Span.
span_field	No	String	Attribute of the span field.
order_by	No	String	Sorting order.
latest	No	String	Whether to display only the latest data.

Table 5-94 FieldItem

Parameter	Mandatory	Type	Description
function	No	String	Expression.
as	No	String	As.
default_value	No	String	Default value.
trace	No	Boolean	Trace or not.
precision	No	Integer	Percentage.
unit	No	String	Unit.
visible	No	Boolean	Visible or not.

Response Parameters

Status code: 200

Table 5-95 Response body parameters

Parameter	Type	Description
line_list	Array of FrontLine objects	Trend graph data list.
latest_data_Time	Long	Last date and time.

Table 5-96 FrontLine

Parameter	Type	Description
point_list	Array of FrontPoint objects	Data point set.
title	String	Title.
unit	String	Unit.
precision	Integer	Percentage.
data_type	String	Date.
visible	Boolean	Visible or not.

Table 5-97 FrontPoint

Parameter	Type	Description
time	Long	Time.
value	Object	Value.

Example Requests

Obtain the trend graph. Input parameter **view_type** is set to **trend**.

```
/v1/apm2/openapi/view/metric/trend
{
  "end_time" : "2022-10-25 15:40:00",
  "env_id" : "11",
  "monitor_item_id" : 78,
  "start_time" : "2022-10-25 15:20:00",
  "view_config" : {
```

```
"latest" : null,  
"span" : true,  
"collector_name" : "Url",  
"metric_set" : "url",  
"title" : "Calls",  
"table_direction" : null,  
"group_by" : "",  
"filter" : "url=/apm2/health/v1/health-check^method=GET",  
"field_item_list" : [ {  
  "trace" : null,  
  "function" : "SUM(invokeCount)",  
  "as" : "Calls",  
  "default_value" : null,  
  "precision" : null,  
  "unit" : null,  
  "visible" : true  
} ],  
"span_field" : "source",  
"view_type" : "trend"  
}
```

Example Responses

None

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Semantic or parameter error.
401	Unauthorized: No permissions.
403	Forbidden: Access forbidden.
404	Not Found: The requested resource is not found.

Error Codes

See [Error Codes](#).

5.4.7 Obtaining Summary Table Data

Function

This API is used to obtain summary table data.

URI

POST /v1/apm2/openapi/view/metric/sum-table

Request Parameters

Table 5-98 Request header parameters

Parameter	Mandatory	Type	Description
x-business-id	Yes	Long	Application ID.
X-Auth-Token	Yes	String	User token obtained from IAM.

Table 5-99 Request body parameters

Parameter	Mandatory	Type	Description
request_id	No	String	ID of the last request.
view_config	No	SumTableView object	View configuration information.
strategy	No	String	Policy.
page	No	Integer	Current page number.
page_size	No	Integer	Number of data records on each page.
order_by	No	String	Sorting rule.
search_word	No	String	Search keyword.
instance_id	No	Long	Instance ID.
monitor_item_id	No	Long	Monitoring item ID.
env_id	No	Long	Environment ID.
start_time	No	String	Start time.
end_time	No	String	End time.

Table 5-100 SumTableView

Parameter	Mandatory	Type	Description
view_type	No	String	View type. Enumeration values: <ul style="list-style-type: none"> ● trend ● sumtable ● rawtable

Parameter	Mandatory	Type	Description
collector_name	No	String	Collector name.
metric_set	No	String	Name of the metric set corresponding to the view.
title	No	String	Title to be displayed.
table_direction	No	String	Direction of the table heading. Options: H (default): horizontal. V: vertical. Enumeration values: <ul style="list-style-type: none"> • H • V
group_by	No	String	Grouping rule.
filter	No	String	Filter list model.
field_item_list	No	Array of FieldItem objects	Field list models to be displayed.
span	No	Boolean	Span.
span_field	No	String	Span field.
order_by	No	String	Sorting rule.
latest	No	Boolean	Whether to display only the latest data.

Table 5-101 FieldItem

Parameter	Mandatory	Type	Description
function	No	String	Expression.
as	No	String	As.
default_value	No	String	Default value.
trace	No	Boolean	Trace or not.
precision	No	Integer	Percentage.
unit	No	String	Unit.
visible	No	Boolean	Visible or not.

Response Parameters

Status code: 200

Table 5-102 Response body parameters

Parameter	Type	Description
result_id	String	Result ID, which is contained in the pagination query result.
row_list	Array of FrontRow objects	Data row list.
latest_data_Time	Long	Time when the latest data record was generated.
table_direction	String	Direction of the table heading. Options: H (default): horizontal. V: vertical. Enumeration values: <ul style="list-style-type: none">• H• V
real_start_time	Long	Actual start time.
real_end_time	Long	Actual end time.
notice_msg	String	Prompt message.
total_count	Integer	Total number.

Table 5-103 FrontRow

Parameter	Type	Description
cell_list	Array of FrontCell objects	Data unit set.
filter	String	"Group by" fields concatenated for subsequent query.
header	Boolean	Whether the information is about the header.
tx_id	Long	URL trace ID.

Table 5-104 FrontCell

Parameter	Type	Description
data_type	String	Data type.
function	String	Function.
trace	Boolean	Trace or not.
span	Boolean	Whether the information is related to spans. If yes, you will be redirected to the trace search page.
span_field	String	Span field.
precision	Integer	Number of decimal places.
text	String	Text information.
unit	String	Unit.
visible	Boolean	Visible or not.

Example Requests

Obtain the summary table data. Input parameter **view_type** is set to **sumtable**.

```
/v1/apm2/openapi/view/metric/sum-table
```

```
{
  "end_time" : "2022-10-25 15:17:46",
  "env_id" : "11",
  "monitor_item_id" : 78,
  "page" : 1,
  "page_size" : 10,
  "start_time" : "2022-10-25 14:57:46",
  "view_config" : {
    "latest" : null,
    "span" : true,
    "collector_name" : "Url",
    "metric_set" : "url",
    "title" : "URL Summary",
    "table_direction" : null,
    "group_by" : "url,method",
    "filter" : "",
    "field_item_list" : [ {
      "trace" : null,
      "function" : "SUM(invokeCount)",
      "as" : "Calls",
      "default_value" : null,
      "precision" : null,
      "unit" : null,
      "visible" : true
    }, {
      "trace" : null,
      "function" : "SUM(range6)",
      "as" : "10s-n",
      "default_value" : null,
      "precision" : null,
      "unit" : null,
      "visible" : false
    }
  ],
}
```

```
"span_field" : "source",  
"view_type" : "sumtable"  
}  
}
```

Example Responses

None.

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Semantic or parameter error.
401	Unauthorized: No permissions.
403	Forbidden: Access forbidden.
404	Not Found: The requested resource is not found.

Error Codes

See [Error Codes](#).

5.4.8 Obtaining the Raw Data Table

Function

This API is used to obtain the raw data table.

URI

POST /v1/apm2/openapi/view/metric/raw-table

Request Parameters

Table 5-105 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM.
x-business-id	Yes	Long	Application ID.

Table 5-106 Request body parameters

Parameter	Mandatory	Type	Description
last_row_id	No	String	ID of the data scanned last time.
view_config	Yes	RawTableView object	Raw data table view.
page	Yes	Integer	Current page number.
page_size	Yes	Integer	Number of data records on each page.
order_by	No	String	Sorting order.
search_word	No	String	Search keyword.
instance_id	No	Long	Instance ID.
monitor_item_id	No	Long	Monitoring item ID.
env_id	Yes	Long	Environment ID.
start_time	Yes	String	Start time.
end_time	Yes	String	End time.

Table 5-107 RawTableView

Parameter	Mandatory	Type	Description
view_type	Yes	String	View type. Enumeration values: <ul style="list-style-type: none"> • trend • sumtable • rawtable
collector_name	Yes	String	Collector name.
metric_set	Yes	String	Name of the metric set corresponding to the view.
title	Yes	String	Title to be displayed.
table_direction	No	String	Direction of the table heading. Options: H (default): horizontal. V: vertical. Enumeration values: <ul style="list-style-type: none"> • H • V

Parameter	Mandatory	Type	Description
group_by	No	String	Grouping rule.
filter	Yes	String	Filter list model.
field_item_list	Yes	Array of FieldItem objects	Field list models to be displayed.
span	Yes	Boolean	Span.
span_field	Yes	String	Span field.
order_by	No	String	Sorting rule.
latest	No	Boolean	Whether to display only the latest data.

Table 5-108 FieldItem

Parameter	Mandatory	Type	Description
function	No	String	Expression.
as	No	String	As.
default_value	No	String	Default value.
trace	No	Boolean	Trace or not.
precision	No	Integer	Percentage.
unit	No	String	Unit.
visible	No	Boolean	Visible or not.

Response Parameters

Status code: 200

Table 5-109 Response body parameters

Parameter	Type	Description
row_list	Array of FrontRow objects	Data row.
latest_data_Time	String	Time when the latest data record was generated.

Parameter	Type	Description
table_direction	String	Direction of the table heading. Options: H (default): horizontal. V: vertical. Enumeration values: <ul style="list-style-type: none"> • H • V
result_id	String	ID of the last request.
real_start_time	Long	Actual start time, which is used for the next invocation. It must be transferred during pagination query.
real_end_time	Long	Actual end time.

Table 5-110 FrontRow

Parameter	Type	Description
cell_list	Array of FrontCell objects	Data unit set.
filter	String	"Group by" fields concatenated for subsequent query.
header	Boolean	Whether the information is about the header.
tx_id	Long	URL trace ID.

Table 5-111 FrontCell

Parameter	Type	Description
data_type	String	Data type.
function	String	Function.
trace	Boolean	Trace or not.
span	Boolean	Whether the information is related to spans. If yes, you will be redirected to the trace search page.
span_field	String	Span field.
precision	Integer	Number of decimal places.
text	String	Text information.
unit	String	Unit.

Parameter	Type	Description
visible	Boolean	Visible or not.

Example Requests

The input parameter is `view_type` (whose value is `rawtable`), which can be obtained by calling the API used to query monitoring item configurations.

```
/v1/apm2/openapi/view/metric/raw-table
{
  "end_time" : 1667436600000,
  "env_id" : "913",
  "monitor_item_id" : 10499,
  "page" : 1,
  "page_size" : 10,
  "start_time" : 1667435400000,
  "view_config" : {
    "span" : null,
    "latest" : null,
    "collector_name" : "Exception",
    "metric_set" : "exception",
    "title" : "Error Stack",
    "table_direction" : null,
    "group_by" : "",
    "filter" :
      "className=com.*****.lubanops.apm.alarm.service.impl.AviatorService^exceptionType=java.lang.ClassCastExc
      eption^logType=sl4j_logback_error",
    "field_item_list" : [ {
      "trace" : null,
      "function" : "stackTrace",
      "as" : "Error Stack",
      "default_value" : null,
      "precision" : null,
      "unit" : null,
      "visible" : true
    } ],
    "span_field" : null,
    "view_type" : "rawtable"
  }
}
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "row_list" : [ {
    "header" : true,
    "cell_list" : [ {
      "span" : null,
      "trace" : null,
      "data_type" : "STRING",
      "function" : null,
      "span_field" : null,
      "precision" : null,
      "text" : "time",
      "unit" : null,
      "visible" : true
    } ],
    "span" : null,
```

```

"trace" : null,
"data_type" : null,
"function" : null,
"span_field" : null,
"precision" : null,
"text" : "6-973f4c1c78928bcf67d67dfaccc9d3a09f3cef97b2de70bd11f103a1af449b94",
"unit" : null,
"visible" : true
}],
"filter" : null,
"tx_id" : null
}],
"latest_data_Time" : "2022-11-03 08:41:00",
"table_direction" : "H",
"result_id" : null,
"real_start_time" : 1667435400000,
"real_end_time" : 1667436600000
}

```

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Semantic or parameter error.
401	Unauthorized: No permissions.
403	Forbidden: Access forbidden.
404	Not Found: The requested resource is not found.

Error Codes

See [Error Codes](#).

5.4.9 Obtaining Raw Data Details

Function

This API is used to obtain raw data details.

URI

POST /v1/apm2/openapi/view/metric/get-clob-detail

Request Parameters

Table 5-112 Request header parameters

Parameter	Mandatory	Type	Description
x-business-id	Yes	Long	Application ID.

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM.

Table 5-113 Request body parameters

Parameter	Mandatory	Type	Description
env_id	Yes	Long	Environment ID.
clob_id	Yes	String	CLOB ID.

Response Parameters

Status code: 200

Table 5-114 Response body parameters

Parameter	Type	Description
clob_string	String	CLOB details.

Example Requests

The input parameters are the environment ID and clob_id. Obtain raw data details.

```
/v1/apm2/openapi/view/metric/get-clob-detail
```

```
{
  "clob_id" : "6-4bc170d880f65de1776984774fd7f03d50f3c7de7667d51259f797f44770ed14",
  "env_id" : "913"
}
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "clob_string" : "java.lang.ClassCastException: *****"
}
```

Status Codes

Status Code	Description
200	OK: The request is successful.

Status Code	Description
400	Bad Request: Semantic or parameter error.
401	Unauthorized: No permissions.
403	Forbidden: Access forbidden.
404	Not Found: The requested resource is not found.

Error Codes

See [Error Codes](#).

5.4.10 Obtaining the Instance Information

Function

This API is used to obtain the instance information.

URI

POST /v1/apm2/openapi/view/mainview/get-env-instance-list

Request Parameters

Table 5-115 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM.
x-business-id	Yes	Long	Application ID.

Table 5-116 Request body parameters

Parameter	Mandatory	Type	Description
env_id	Yes	Long	Environment ID.
page	Yes	Integer	Current page number.
page_size	No	Integer	Number of records on each page.
keyword	No	String	Keyword.
status	No	Integer	Instance status.

Parameter	Mandatory	Type	Description
return_count	No	Boolean	Whether to return the counting result.

Response Parameters

Status code: 200

Table 5-117 Response body parameters

Parameter	Type	Description
instance_info_list	Array of InstanceInfo objects	Instance information.
total_count	Integer	Number of instances.
online_count	Integer	Total number of online instances.
offline_count	Integer	Total number of offline instances.
disable_count	Integer	Total number of stopped instances.

Table 5-118 MainViewInstanceInfo

Parameter	Type	Description
instance_id	Long	Instance ID, which can be an input parameter for the API used to enable or disable collection for an instance.
host_name	String	Host name.
instance_name	String	Instance name.
ip_address	String	Host IP address.
agent_version	String	JavaAgent version.
last_heartbeat	Long	Last heartbeat time, in milliseconds.
register_time	Long	Registration time, in milliseconds.
instance_statuses	Integer	Instance status.
last_modify_time	Long	Last modification time, in milliseconds.

Example Requests

Query the instance information list in environment 11.

```
/v1/apm2/openapi/view/mainview/get-env-instance-list  
  
{  
  "env_id": "11",  
  "page": 1,  
  "page_size": 10  
}
```

Example Responses

Status code: 200

OK: The request is successful.

```
{  
  "instance_info_list": [ {  
    "instance_id": 15,  
    "business_name": null,  
    "business_id": null,  
    "app_name": null,  
    "host_name": "apm2-apm-pu-task-6b5bbfc84d-rx1hr",  
    "instance_name": "default",  
    "ip_address": "172.16.3.39",  
    "env_id": null,  
    "agent_version": "2.2.8",  
    "last_heartbeat": 1666682386000,  
    "register_time": 1666664387000,  
    "last_modify_user_id": null,  
    "instance_status": 0,  
    "last_modify_user_name": null,  
    "last_modify_time": 0  
  }, {  
    "instance_id": 16,  
    "business_name": null,  
    "business_id": null,  
    "app_name": null,  
    "host_name": "apm2-apm-pu-task-6b5bbfc84d-gtrrs",  
    "instance_name": "default",  
    "ip_address": "172.16.3.7",  
    "env_id": null,  
    "agent_version": "2.2.8",  
    "last_heartbeat": 1666682377000,  
    "register_time": 1666664436000,  
    "last_modify_user_id": null,  
    "instance_status": 0,  
    "last_modify_user_name": null,  
    "last_modify_time": 0  
  } ],  
  "total_count": 2,  
  "online_count": 0,  
  "offline_count": 0,  
  "disable_count": 0  
}
```

Status Codes

Status Code	Description
200	OK: The request is successful.

Status Code	Description
400	Bad Request: Semantic or parameter error.
401	Unauthorized: No permissions.
403	Forbidden: Access forbidden.
404	Not Found: The requested resource is not found.

Error Codes

See [Error Codes](#).

5.4.11 Obtaining the Monitoring Item Information

Function

This API is used to obtain the monitoring item information.

URI

GET /v1/apm2/openapi/view/mainview/get-env-monitor-item-list

Table 5-119 Query Parameters

Parameter	Mandatory	Type	Description
env_id	Yes	Long	Environment ID.

Request Parameters

Table 5-120 Request header parameters

Parameter	Mandatory	Type	Description
x-business-id	Yes	Long	Application ID.
X-Auth-Token	Yes	String	User token obtained from IAM.

Response Parameters

Status code: 200

Table 5-121 Response body parameters

Parameter	Type	Description
category_info_list	Array of CollectorCategoryInfo objects	Collector type set.
monitor_item_info_list	Array of MonitorItemEntity objects	Monitoring item set.

Table 5-122 CollectorCategoryInfo

Parameter	Type	Description
category_id	Integer	Collector type ID.
category_name	String	Collector type name.
display_name	String	Display name of a collector type.
sequence	Integer	Sequence number.

Table 5-123 MonitorItemEntity

Parameter	Type	Description
category_id	Integer	Collector type ID.
collector_name	String	Collector name.
display_name	String	Display name of a collector type.
show_in_total	Boolean	Whether to display the title.
monitor_item_id	Long	Monitoring item ID.
disabled	Boolean	Disable or not.
collector_id	Integer	Collector ID.
sequence	Integer	Sequence number.
collect_interval	Integer	Default data collection interval.

Example Requests

Obtain the monitoring item information in environment 11.

```
/v1/apm2/openapi/view/mainview/get-env-monitor-item-list?env_id=11
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "category_info_list": [ {
    "category_id": 7,
    "category_name": "Url",
    "display_name": "URL",
    "sequence": 1
  }, {
    "monitor_item_id": 11,
    "disabled": false,
    "collector_id": 16,
    "sequence": 60,
    "collect_interval": 60,
    "category_id": 10,
    "collector_name": "ProbeInfo",
    "display_name": "Agent Monitoring",
    "show_in_total": true
  } ]
}
```

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Semantic or parameter error.
401	Unauthorized: No permissions.
403	Forbidden: Access forbidden.
404	Not Found: The requested resource is not found.

Error Codes

See [Error Codes](#).

5.4.12 Obtaining the Details About a Monitoring Item

Function

This API is used to obtain the details about a monitoring item.

URI

GET /v1/apm2/openapi/apm-service/monitor-item-mgr/get-monitor-item-detail

Table 5-124 Query Parameters

Parameter	Mandatory	Type	Description
monitor_item_id	Yes	Long	Monitored item ID
env_id	Yes	Long	Environment ID

Request Parameters

Table 5-125 Request header parameters

Parameter	Mandatory	Type	Description
x-business-id	Yes	Long	Application ID.
X-Auth-Token	Yes	String	User token obtained from IAM.

Response Parameters

Status code: 200

Table 5-126 Response body parameters

Parameter	Type	Description
interval	String	Collection interval, in seconds.
collector_id	String	Collector ID.
config_item_list	Array of ConfigItemValue objects	Collection parameter configuration list.

Table 5-127 ConfigItemValue

Parameter	Type	Description
config_name	String	Configuration item name.
display_name	String	Display name.
config_data_type	String	Data type.

Parameter	Type	Description
max_length	Integer	Maximum length.
min_value	Double	Minimum value.
max_value	Double	Maximum value.
option_values	Array of OptionValue objects	Optional value.
default_value	String	Default value.
since_version	String	Start version.
dead_version	String	End version.
value	String	Value.
object_array_patterns	Array of object_array_patterns objects	An array of objects.
override_list	Array of ConfigItemOverride objects	Actual value.

Table 5-128 OptionValue

Parameter	Type	Description
value	String	Optional value.
hint	String	Information.

Table 5-129 object_array_patterns

Parameter	Type	Description
key	String	Key.
type	String	Object type.
max_length	Integer	Maximum length.
max_value	Double	Maximum value.
min_value	Double	Minimum value.
nullable	Boolean	Whether this parameter can be null or not.
hint	String	Information.

Table 5-130 ConfigItemOverride

Parameter	Type	Description
env_tag_id	Long	Environment tag ID.
env_tag_name	String	Environment tag name.
key	String	Key.
value	String	Value.

Example Requests

Obtain details about monitoring item 5663 in environment 895.

```
/v1/apm2/openapi/apm-service/monitor-item-mgr/get-monitor-item-detail?
env_id=895&monitor_item_id=5663
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "interval" : 60,
  "collector_id" : 28,
  "config_item_list" : [ {
    "config_name" : "slowRequestThreshold",
    "display_name" : "Threshold for determining that a request is slow",
    "config_data_type" : "integer",
    "max_length" : null,
    "min_value" : 0,
    "max_value" : 20000,
    "option_values" : null,
    "default_value" : "800",
    "object_array_patterns" : null,
    "since_version" : "2.0.0",
    "dead_version" : null,
    "value" : null,
    "override_list" : null
  } ]
}
```

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Semantic or parameter error.
401	Unauthorized: No permissions.
403	Forbidden: Access forbidden.
404	Not Found: The requested resource is not found.

Error Codes

See [Error Codes](#).

5.5 AK/SK Management APIs

5.5.1 Creating an AK/SK

Function

This API is used to create an AK/SK.

URI

POST /v1/apm2/access-keys

Request Parameters

Table 5-131 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM.

Table 5-132 Request body parameters

Parameter	Mandatory	Type	Description
descp	No	String	Description.

Response Parameters

Status code: 200

Table 5-133 Response body parameters

Parameter	Type	Description
ak	String	AK to be created or deleted.
sk	String	SK to be created or deleted.

Example Requests

Create an AK/SK and set the description to "test".

```
/v1/apm2/access-keys
{
  "descp" : "test"
}
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "ak" : "vca*****04j",
  "sk" : "ktns*****6iq6t9m"
}
```

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Semantic or parameter error.
401	Unauthorized: No permissions.
403	Forbidden: Access forbidden.
404	Not Found: The requested resource is not found.

Error Codes

See [Error Codes](#).

5.5.2 Querying the AK/SK of a Tenant

Function

This API is used to query the AK/SK of a tenant.

URI

GET /v1/apm2/access-keys

Request Parameters

Table 5-134 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM.

Response Parameters

Status code: 200

Table 5-135 Response body parameters

Parameter	Type	Description
access_ak_sk_models	Array of AccessAkskV objects	AK/SK data model.

Table 5-136 AccessAkskVO

Parameter	Type	Description
id	Integer	AK/SK ID.
gmt_create	String	Time when the AK/SK was generated.
gmt_modify	String	Time when the AK/SK was modified.
inner_domain_id	Integer	Internal tenant ID.
ak	String	Generated AK.
sk	String	Generated SK.
status	String	AK/SK status.
descp	String	AK/SK description.
gmt_create_timestamp	Integer	Time when the AK/SK was generated.
gmt_modify_timestamp	Integer	Time when the AK/SK was modified.

Example Requests

Query the AK/SK of a tenant. No input parameter is required.

```
/v1/apm2/access-keys
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "access_ak_sk_models": [ {
    "ak": "vca*****04j",
    "sk": "ktns*****6iq6t9m",
```

```

    "status" : "enable",
    "descp" : "auto create",
    "id" : 1,
    "gmt_create" : "2020-11-26 16:51:08",
    "gmt_modify" : "2021-03-27 11:02:50",
    "inner_domain_id" : 1
  } ]
}

```

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Semantic or parameter error.
401	Unauthorized: No permissions.
403	Forbidden: Access forbidden.
404	Not Found: The requested resource is not found.

Error Codes

See [Error Codes](#).

5.5.3 Deleting an AK/SK

Function

This API is used to delete an AK/SK.

URI

DELETE /v1/apm2/access-keys/{ak}

Table 5-137 Path Parameters

Parameter	Mandatory	Type	Description
ak	Yes	String	AK to be deleted.

Request Parameters

Table 5-138 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM.

Response Parameters

Status code: 200

Table 5-139 Response body parameters

Parameter	Type	Description
ak	String	AK to be created or deleted.

Example Requests

Delete AK/SK whose ID is vca*****04j.

```
/v1/apm2/access-keys/vca*****04j
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "ak" : "vca*****04j",
}
```

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Semantic or parameter error.
401	Unauthorized: No permissions.
403	Forbidden: Access forbidden.
404	Not Found: The requested resource is not found.

Error Codes

See [Error Codes](#).

5.6 Alarm Management APIs

5.6.1 Querying the Alarm List

Function

This API is used to query the alarm list.

URI

POST /v1/apm2/openapi/alarm/data/get-alarm-data-list

Request Parameters

Table 5-140 Request header parameters

Parameter	Mandatory	Type	Description
x-business-id	Yes	Long	Application ID.
X-Auth-Token	Yes	String	User token obtained from IAM.

Table 5-141 Request body parameters

Parameter	Mandatory	Type	Description
page	No	Integer	Page number.
page_size	No	Integer	Number of records on each page.
region	Yes	String	Region name.
app_name	No	String	Component environment name.
business_id	Yes	Long	Application ID.
monitor_item_id	No	Long	Monitoring item ID.
status	No	String	Alarm status. Options: RECOVER, ABNORMAL, and ALERT.

Parameter	Mandatory	Type	Description
alarm_level	No	String	Alarm severity. Options: COMMON and CRITICAL.
keyword	No	String	Keyword.
alarm_start_time	No	String	Alarm start time.
alarm_end_time	No	String	Alarm end time.
collector_id	No	Integer	Collector ID.
ip_address	No	String	IP address of the instance.
env_list	No	Array of integers	Environment set.

Response Parameters

Status code: 200

Table 5-142 Response body parameters

Parameter	Type	Description
alarm_data_list	Array of AlarmDataVO objects	Alarm list.
total_count	Integer	Total number of messages.

Table 5-143 AlarmDataVO

Parameter	Type	Description
id	Long	Alarm notification ID.
gmt_create	String	Creation time.
region_alarm_event_id	Long	Event ID.
business_name	String	Application name.
app_name	String	Component name.
version_number	Integer	Version.

Parameter	Type	Description
alarm_rule_type	String	Alarm rule type.
gmt_modify	String	Update time.
process_unit	String	Processing unit.
region	String	Region name.
instance_id	Long	Instance ID.
ip_address	String	IP address of the instance.
instance_name	String	Instance name.
env_id	Long	Environment ID.
business_id	Long	Application ID.
template_id	Long	Template ID.
alarm_rule_id	Long	Alarm rule ID.
monitor_item_id	Long	Monitoring item ID.
collector_id	Integer	Collector ID.
collector_name	String	Collector name.
alarm_rule_name	String	Alarm rule name.
alarm_rule_expression	String	Alarm expression.
alarm_first_time	String	Alarm start time.
alarm_last_time	String	Latest alarm report time.
alarm_level	String	Alarm severity.
restrain_key	String	Unique alarm identifier.
status	String	Alarm status.

Example Requests

Query the list of alarms whose status is "ALERT" under application 1.

```
/v1/apm2/openapi/alarm/data/get-alarm-data-list
```

```
{
  "business_id" : 1,
  "page" : 1,
  "page_size" : 10,
  "status" : "ALERT"
}
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "alarm_data_list" : [ {
    "id" : 42,
    "gmt_create" : null,
    "version_number" : null,
    "alarm_rule_type" : "TEMPLATE",
    "gmt_modify" : null,
    "process_unit" : "-process-unit",
    "region" : "",
    "instance_id" : 8,
    "ip_address" : "192.168.0.87",
    "instance_name" : "ins11_3",
    "env_id" : 7,
    "business_id" : 1,
    "template_id" : 2,
    "alarm_rule_id" : 3,
    "monitor_item_id" : 46,
    "collector_id" : 28,
    "collector_name" : "JVM Monitoring",
    "alarm_rule_name" : "alarm",
    "alarm_rule_expression" : "(cpuRatio>0)",
    "alarm_first_time" : "2022-10-23 18:12:53",
    "alarm_last_time" : "2022-10-25 16:05:55",
    "alarm_level" : "COMMON",
    "restrain_key" : "7:8:3:TEMPLATE",
    "status" : "ALERT",
    "alarm_content" : "The CPU usage is 0.025.",
    "alarm_data_type" : "SINGLE",
    "region_alarm_event_id" : 122,
    "business_name" : "default",
    "app_name" : "app_new3"
  }, {
    "id" : 41,
    "gmt_create" : null,
    "version_number" : null,
    "alarm_rule_type" : "TEMPLATE",
    "gmt_modify" : null,
    "process_unit" : "-process-unit",
    "region" : "",
    "instance_id" : 4,
    "ip_address" : "192.168.0.87",
    "instance_name" : "ins10_5",
    "env_id" : 5,
    "business_id" : 1,
    "template_id" : 2,
    "alarm_rule_id" : 3,
    "monitor_item_id" : 28,
    "collector_id" : 28,
    "collector_name" : "JVM Monitoring",
    "alarm_rule_name" : "alarm",
    "alarm_rule_expression" : "(cpuRatio>0)",
    "alarm_first_time" : "2022-10-23 18:02:44",
    "alarm_last_time" : "2022-10-25 16:05:46",
    "alarm_level" : "COMMON",
    "restrain_key" : "5:4:3:TEMPLATE",
    "status" : "ALERT",
  }
]
```

```

"alarm_content" : "The CPU usage is 0.025.",
"alarm_data_type" : "SINGLE",
"region_alarm_event_id" : 121,
"business_name" : "default",
"app_name" : "app_new5"
}],
"total_count" : 2
}

```

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Semantic or parameter error.
401	Unauthorized: No permissions.
403	Forbidden: Access forbidden.
404	Not Found: The requested resource is not found.

Error Codes

See [Error Codes](#).

5.6.2 Querying the Alarm Message List

Function

Query the trigger details and history of an alarm.

URI

POST /v1/apm2/openapi/alarm/data/get-alarm-notify-list

Request Parameters

Table 5-144 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM.
x-business-id	Yes	Long	Application ID, which is used for authentication.

Table 5-145 Request body parameters

Parameter	Mandatory	Type	Description
page	No	Integer	Page number.
page_size	No	Integer	Number of records on each page.
alarm_data_id	Yes	Integer	Alarm event ID.
region	Yes	String	Region name.

Response Parameters

Status code: 200

Table 5-146 Response body parameters

Parameter	Type	Description
front_alarm_notify_results	Array of FrontAlarmNotifyResult objects	Alarm notification list.
total_count	Integer	Total number of messages.

Table 5-147 FrontAlarmNotifyResult

Parameter	Type	Description
id	Long	Alarm notification ID.
gmt_create	String	Creation time.
notify_type	String	Notification type.
alarm_rule_id	Long	Alarm rule ID.
template_id	Long	Template ID.
alarm_data_event_id	Long	Associated event ID.
notify_status	Boolean	Notification result.
alarm_content	String	Notification content.

Example Requests

Query the trigger details and history of alarm event 42 in region .

```
/v1/apm2/openapi/alarm/data/get-alarm-notify-list
{
  "alarm_data_id" : 42,
  "region" : "",
  "page" : 1,
  "page_size" : 10
}
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "front_alarm_notify_results" : [ {
    "id" : 1741,
    "gmt_create" : "2022-10-09 22:58:27",
    "notify_type" : "RECOVER",
    "alarm_rule_id" : 3,
    "template_id" : 2,
    "alarm_data_event_id" : 42,
    "notify_status" : false,
    "alarm_content" : "The CPU usage is 0.017."
  }, {
    "id" : 639,
    "gmt_create" : "2022-10-09 21:16:04",
    "notify_type" : "ALARM",
    "alarm_rule_id" : 3,
    "template_id" : 2,
    "alarm_data_event_id" : 42,
    "notify_status" : false,
    "alarm_content" : "The CPU usage is 0.021."
  } ],
  "total_count" : 2
}
```

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Semantic or parameter error.
401	Unauthorized: No permissions.
403	Forbidden: Access forbidden.
404	Not Found: The requested resource is not found.

Error Codes

See [Error Codes](#).

5.7 Topology Management APIs

5.7.1 Querying the Global Topology of an Application

Function

This API is used to query the global topology of an application.

URI

POST /v1/apm2/openapi/topology/business-search

Request Parameters

Table 5-148 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM.
x-business-id	Yes	Long	Application ID.

Table 5-149 Request body parameters

Parameter	Mandatory	Type	Description
target_business_id	Yes	Long	Target application ID.
env_tag_list	No	Array of integers	Environment tag list, which can be left empty.
direction	No	String	Direction, which can be left empty.
end_time	Yes	String	End time.
start_time	Yes	String	Start time.
filter_user	No	Boolean	Filter or not.

Response Parameters

Status code: 200

Table 5-150 Response body parameters

Parameter	Type	Description
node_list	Array of TopoNode objects	List of component nodes.
line_list	Array of TopoLine objects	List of lines that indicate invocations between components.
collector_config	Map<String, CollectorConfigModel >	Collector configuration.
real_start_time	Long	Start time.
real_end_time	Long	End time.

Table 5-151 TopoNode

Parameter	Type	Description
node_type	String	Node type.
node_name	String	Node name.
node_id	String	Node ID.
env_id	Long	Environment ID.

Table 5-152 TopoLine

Parameter	Type	Description
from_node	String	Start node.
to_node	String	End node.
direction	String	Invocation direction.
collector	String	Collector name.
target_env_id	Long	Environment ID.
hints	Map<String,String>	Line prompt.
filter_value	String	Filtered value.

Table 5-153 CollectorConfigModel

Parameter	Type	Description
line_view_config	LineViewConfigModel object	Invocation line view configuration.
detail_view_config	DetailViewConfigModel object	Details view configuration.

Table 5-154 LineViewConfigModel

Parameter	Type	Description
metric_set	String	Name of the metric set corresponding to the view.
filter_prefix	String	Filtering parameter.
line_view_item_list	Array of LineViewItem objects	View function set.

Table 5-155 LineViewItem

Parameter	Type	Description
function	String	Expression.
as	String	As.

Table 5-156 DetailViewConfigModel

Parameter	Type	Description
metric_set	String	Name of the metric set corresponding to the view.
filter_prefix	String	Filtering parameter.
detail_view_item_list	Array of DetailViewItem objects	View function set.

Table 5-157 DetailViewItem

Parameter	Type	Description
function	String	Expression.
as	String	As.

Example Requests

Query the global topology of application 6, with the start timestamp set to 1667462525000 and end timestamp set to 1667463725000.

```
/v1/apm2/openapi/topology/business-search
```

```
{
  "direction": "",
  "end_time": 1667463725000,
  "start_time": 1667462525000,
  "filter_user": true,
  "target_business_id": 6,
  "env_tag_list": []
}
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "node_list": [ {
    "node_type": "Env",
    "node_name": "apm-task:",
    "node_id": "55",
    "env_id": 55
  }, {
    "node_type": "Env",
    "node_name": "apm-access:",
    "node_id": "43",
    "env_id": 43
  } ],
  "line_list": [ {
    "from_node": "user",
    "to_node": "51",
    "direction": "in",
    "collector": "Url",
    "target_env_id": 51,
    "hints": {
      "rt": "0.09",
      "count": "272",
      "error": "0"
    }
  }, {
    "filter_value": "user"
  }, {
    "from_node": "47",
    "to_node": "51",
    "direction": "in",
    "collector": "Url",
    "target_env_id": 51,
    "hints": {
      "rt": "10.28",
      "count": "780",
      "error": "0"
    }
  } ]
}
```

```

    },
    "filter_value" : "55"
  }],
  "detail_view_config" : {
    "metric_set" : "connection",
    "group_by" : "db",
    "detail_view_item_list" : [ {
      "function" : "SUM(invokeCount)",
      "as" : "count"
    }, {
      "function" : "SUM(totalTime)/SUM(invokeCount)",
      "as" : "rt"
    }, {
      "function" : "SUM(errorCount)",
      "as" : "errorCount"
    }
  ]
}
},
"real_start_time" : 1667465696000,
"real_end_time" : 1667466896000
}

```

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Semantic or parameter error.
401	Unauthorized: No permissions.
403	Forbidden: Access forbidden.
404	Not Found: The requested resource is not found.

Error Codes

See [Error Codes](#).

5.7.2 Querying the Topology of a Component Environment

Function

This API is used to query the topology of a component environment.

URI

POST /v1/apm2/openapi/topology/env-search

Request Parameters

Table 5-158 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM.
x-business-id	Yes	Long	Application ID.

Table 5-159 Request body parameters

Parameter	Mandatory	Type	Description
target_env_id	Yes	Long	Environment ID.
direction	No	String	Direction, which can be left empty.
end_time	Yes	String	End time.
start_time	Yes	String	Start time.
filter_user	No	Boolean	Filter or not.

Response Parameters

Status code: 200

Table 5-160 Response body parameters

Parameter	Type	Description
node_list	Array of TopoNode objects	List of component nodes.
line_list	Array of TopoLine objects	List of lines that indicate invocations between components.
collector_config	Map<String, CollectorConfigModel >	Collector configuration.
real_start_time	Long	Start time.
real_end_time	Long	End time.

Table 5-161 TopoNode

Parameter	Type	Description
node_type	String	Node type.
node_name	String	Node name.
node_id	String	Node ID.
env_id	Long	Environment ID.

Table 5-162 TopoLine

Parameter	Type	Description
from_node	String	Start node.
to_node	String	End node.
direction	String	Invocation direction.
collector	String	Collector name.
target_env_id	Long	Environment ID.
hints	Map<String,String>	Line prompt.
filter_value	String	Filtered value.

Table 5-163 CollectorConfigModel

Parameter	Type	Description
line_view_config	LineViewConfigModel object	Invocation line view configuration.
detail_view_config	DetailViewConfigModel object	Details view configuration.

Table 5-164 LineViewConfigModel

Parameter	Type	Description
metric_set	String	Name of the metric set corresponding to the view.
filter_prefix	String	Filtering parameter.

Parameter	Type	Description
line_view_item_list	Array of LineViewItem objects	View function set.

Table 5-165 LineViewItem

Parameter	Type	Description
function	String	Expression.
as	String	As.

Table 5-166 DetailViewConfigModel

Parameter	Type	Description
metric_set	String	Name of the metric set corresponding to the view.
filter_prefix	String	Filtering parameter.
detail_view_item_list	Array of DetailViewItem objects	View function set.

Table 5-167 DetailViewItem

Parameter	Type	Description
function	String	Expression.
as	String	As.

Example Requests

Query the global topology of environment 4394, with the start timestamp set to 1667465258000 and end timestamp set to 1667466458000.

```
/v1/apm2/openapi/topology/env-search
{
  "direction": "",
  "end_time": 1667466458000,
  "start_time": 1667465258000,
  "filter_user": false,
  "target_env_id": 4394
}
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "node_list": [ {
    "node_type": "Env",
    "node_name": "apm-archive:",
    "node_id": "4394",
    "env_id": 4394
  }, {
    "node_type": "Env",
    "node_name": "user",
    "node_id": "user",
    "env_id": null
  } ],
  "Url": {
    "line_view_config": {
      "metric_set": "user",
      "filter_prefix": "clusterId=",
      "line_view_item_list": [ {
        "function": "SUM(invokeCount)",
        "as": "count"
      }, {
        "function": "SUM(totalTime)/SUM(invokeCount)",
        "as": "rt"
      }, {
        "function": "SUM(errorCount)",
        "as": "errorCount"
      } ]
    }
  },
  "detail_view_config": null
},
"real_start_time": 1667465258000,
"real_end_time": 1667466458000
}
```

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Semantic or parameter error.
401	Unauthorized: No permissions.
403	Forbidden: Access forbidden.
404	Not Found: The requested resource is not found.

Error Codes

See [Error Codes](#).

5.8 Transaction Query APIs

5.8.1 Querying the URL Tracing Configuration List

Function

This API is used to query the URL tracing configuration list.

URI

POST /v1/apm2/openapi/transaction/transaction-config-search

Request Parameters

Table 5-168 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM.
x-business-id	Yes	Long	Application ID.

Table 5-169 Request body parameters

Parameter	Mandatory	Type	Description
business_id	Yes	Long	Application ID.
env_id	No	Long	Environment ID.
page_no	Yes	Integer	Page number.
page_size	Yes	Integer	Number of records on each page.

Response Parameters

Status code: 200

Table 5-170 Response body parameters

Parameter	Type	Description
transaction_config_item_list	Array of TransactionConfigItem objects	URL tracing configuration list.
total_page	Integer	Total number of pages.
total_count	Integer	Total number of configurations.

Table 5-171 TransactionConfigItem

Parameter	Type	Description
id	Long	Configuration ID.
business_id	Long	Application ID.
env_id	Long	Environment ID.
method	String	Request method.
env_name	String	Environment name.
region	String	Region name.
type	String	Type.
app_name	String	Application name.
url	String	URL.

Example Requests

Query the URL tracing configuration list of application 6.

```
/v1/apm2/openapi/transaction/transaction-config-search  
  
{  
  "business_id" : 6,  
  "page_no" : 1,  
  "page_size" : 10  
}
```

Example Responses

Status code: 200

OK: The request is successful.

```
{  
  "total_page" : 2,  
  "total_count" : 11,  
  "transaction_config_item_list" : [ {  
    "id" : 36,  
    "business_id" : 6,  
    "env_id" : 52,  
    "method" : "GET",  
    "env_name" : "",  
    "region" : "",  
    "type" : "Url",  
    "app_name" : "lubanops-web",  
    "url" : "/apm2/web/cmdb/business/v1/get-business-detail/{business_id}"  
  } ]  
}
```

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Semantic or parameter error.
401	Unauthorized: No permissions.
403	Forbidden: Access forbidden.
404	Not Found: The requested resource is not found.

Error Codes

See [Error Codes](#).

5.8.2 Querying a Region's Environments on Which URLs Are Added for Tracing

Function

This API is used to query a region's environments on which URLs are added for tracing.

URI

POST /v1/apm2/openapi/transaction/business-env

Request Parameters

Table 5-172 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM.
x-business-id	Yes	Long	Application ID.

Table 5-173 Request body parameters

Parameter	Mandatory	Type	Description
business_id	Yes	Long	Application ID.
region	No	String	Region name.

Parameter	Mandatory	Type	Description
start_time	No	String	Start time.
end_time	No	String	End time.

Response Parameters

Status code: 200

Table 5-174 Response body parameters

Parameter	Type	Description
env_entry_list	Array of EnvEntry objects	Environment list.

Table 5-175 EnvEntry

Parameter	Type	Description
env_id	Long	Environment ID.
env_name	String	Environment name.

Example Requests

Query the list of the region's environments on which URLs are added for tracing. The application ID is 6.

```
/v1/apm2/openapi/transaction/business-env
```

```
{
  "region": "",
  "business_id": 6,
  "start_time": "-20m",
  "end_time": "now"
}
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "env_entry_list": [ {
    "env_id": 48,
    "env_name": "apm-api:"
  }, {
    "env_id": 51,
    "env_name": "config-web"
  }
]
```

```
}, {  
  "env_id" : 52,  
  "env_name" : "lubanops-web:"  
}, {  
  "env_id" : 295,  
  "env_name" : "lubanops-api:"  
}]  
}
```

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Semantic or parameter error.
401	Unauthorized: No permissions.
403	Forbidden: Access forbidden.
404	Not Found: The requested resource is not found.

Error Codes

See [Error Codes](#).

5.8.3 Querying the URL Tracing View List

Function

This API is used to query the list of URL tracing views.

URI

POST /v1/apm2/openapi/transaction/search

Request Parameters

Table 5-176 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM.
x-business-id	Yes	Long	Application ID.

Table 5-177 Request body parameters

Parameter	Mandatory	Type	Description
business_id	Yes	Long	Application ID.
region	Yes	String	Region name.
start_time	Yes	String	Start timestamp in milliseconds, for example, 1715916233000.
end_time	Yes	String	End timestamp in milliseconds, for example, 1715916233000.
env_id	No	Long	Environment ID.
request_id	No	String	Request ID. Leave this parameter empty for the first query. For pagination query, set this parameter to the value of result_id returned in last API calling.
page_no	Yes	Integer	Page number.
page_size	No	Integer	Number of records on each page.

Response Parameters

Status code: 200

Table 5-178 Response body parameters

Parameter	Type	Description
tx_item_list	Array of TxItemVo objects	URL tracing view list.
latest_time	Long	Timestamp of the last response.
total_count	Integer	Total number of records.
result_id	String	Request ID.

Table 5-179 TxItemVo

Parameter	Type	Description
app_name	String	Component name.

Parameter	Type	Description
env_name	String	Environment name.
tx_display_name	String	Display name of a transaction. It is in the format of "URL : request mode".
business_id	Long	Application ID.
env_id	Long	Environment ID.
app_id	Long	Component ID.
tx_name	String	Transaction name, which must be unique under an application. Set tx_name to the value of this parameter when you call the API for querying the details about a URL tracing view.
invoke_count	Integer	Number of calls.
total_time	Integer	Total call duration of this API, in milliseconds.
error_count	Integer	Number of errors.

Example Requests

Query the URL tracing view list of application 6 in region

```
/v1/apm2/openapi/transaction/search
{
  "region": "",
  "business_id": 6,
  "start_time": 1667454320000,
  "end_time": 1667455520000,
  "page_no": 1,
  "page_size": 10,
  "request_id": ""
}
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "latest_time": 1667455260000,
  "tx_item_list": [ {
    "business_id": 6,
    "env_id": 295,
    "app_id": 175,
    "tx_name": "3",
    "invoke_count": 369,
    "total_time": 19025,
    "error_count": 0,
    "env_name": "",
    "app_name": "lubanops-api",
    "tx_display_name": "/apm2/api/view/trace/v1/span-search : POST"
  } ],
  "total_count": 1,
```

```
"result_id" : "7776f9f2-0294-4305-a291-fc359802bf19"
}
```

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Semantic or parameter error.
401	Unauthorized: No permissions.
403	Forbidden: Access forbidden.
404	Not Found: The requested resource is not found.

Error Codes

See [Error Codes](#).

5.8.4 Querying the Details About a URL Tracing View

Function

This API is used to query the details about a URL tracing view.

URI

POST /v1/apm2/openapi/transaction/detail

Request Parameters

Table 5-180 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM.
x-business-id	Yes	Long	Application ID.

Table 5-181 Request body parameters

Parameter	Mandatory	Type	Description
tx_name	Yes	String	Transaction name.

Parameter	Mandatory	Type	Description
start_time	Yes	String	Start timestamp, in milliseconds.End timestamp, in milliseconds.
end_time	Yes	String	Start timestamp, in milliseconds.End timestamp, in milliseconds.

Response Parameters

Status code: 200

Table 5-182 Response body parameters

Parameter	Type	Description
tx_node_list	Array of TxNode objects	List of component nodes.
tx_line_list	Array of TxLine objects	List of lines that indicate invocations between components.

Table 5-183 TxNode

Parameter	Type	Description
tx_node_id	String	Node ID.
tx_node_name	String	Node name.
tx_node_type	String	Node type, such as user, HttpClient (network node), env (environment), or Mysql (database node).

Table 5-184 TxLine

Parameter	Type	Description
tx_from_node	String	Start node.
tx_to_node	String	End node.
invoke_count	Long	Number of calls.
rt	Double	Inter-component call duration, in milliseconds.

Parameter	Type	Description
error_count	Long	Number of errors.
type	String	Target node type, such as user, HttpClient (network node), env (environment), or Mysql (database node).
direction	String	Invocation direction.

Example Requests

Query the details about the URL tracing view of transaction 3, with the start timestamp set to 1667456607000 and end timestamp set to 1667457807000.

```
/v1/apm2/openapi/transaction/detail
{
  "start_time" : 1667456607000,
  "end_time" : 1667457807000,
  "tx_name" : "3"
}
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "tx_node_list" : [ {
    "tx_node_id" : "48",
    "tx_node_name" : "apm-api:",
    "tx_node_type" : "env",
    "tx_env_id" : 48
  }, {
    "tx_node_id" : "Elasticsearch:***:9200,***:9200,***:9200",
    "tx_node_name" : "Elasticsearch:***:9200,***:9200,***:9200",
    "tx_node_type" : "EsRestClient",
    "tx_env_id" : null
  } ],
  "tx_line_list" : [ {
    "tx_from_node" : "54",
    "tx_to_node" : "48",
    "invoke_count" : 364,
    "rt" : 28.065934065934066,
    "error_count" : 0,
    "type" : "HttpClient",
    "direction" : "out"
  }, {
    "tx_from_node" : "295",
    "tx_to_node" : "***:3306:apm2_cmdb",
    "invoke_count" : 58,
    "rt" : 0.5862068965517241,
    "error_count" : 0,
    "type" : "Mysql",
    "direction" : "out"
  } ]
}
```

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: semantic or parameter error.
401	Unauthorized: No permissions.
403	Forbidden: Access forbidden.
404	Not Found: The requested resource is not found.

Error Codes

See [Error Codes](#).

5.9 Performance Profiling APIs

5.9.1 Querying a Line-Level Flame Graph

Function

This API is used to query a line-level flame graph.

URI

POST /v1/apm2/openapi/view/profiling/flame-line-tree

Request Parameters

Table 5-185 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM.

Table 5-186 Request body parameters

Parameter	Mandatory	Type	Description
from	Yes	Long	Start time, for example, 1704271204595 .

Parameter	Mandatory	Type	Description
to	Yes	Long	End time, for example, 1704275169491 .
type	Yes	String	Data type. Options: CPU and LATENCY .
instance_id	Yes	Long	Instance ID.
api	Yes	String	API URL, for example, GET_/user/{id} .
region	Yes	String	Region where the instance is located.

Response Parameters

Status code: 200

Table 5-187 Response body parameters

Parameter	Type	Description
data	Array of strings	<p>Data of a flame graph, which is a two-dimensional array.</p> <p>data[0][0]: self time, that is, CPU time (in milliseconds) consumed by a method, excluding the time for calling other methods internally.</p> <p>data[0][1]: total time, that is, CPU time (in milliseconds) consumed by a method, including the time for calling other methods internally.</p> <p>data[0][2]: method index, corresponding to the array subscript of a method.</p> <p>data[0][3]: line number.</p> <p>data[0][4]: a method's child nodes, that is, other methods that are internally called by the current method.</p>

Parameter	Type	Description
methods	Array of strings	Method information on the call stack, which is a two-dimensional array. method[0][0]: unique ID of a method. method[0][1]: package name of a method. method[0][2]: class name of a method. method[0][3]: method name. method[0][4]: method parameter list. method[0][5]: whether the method is a user method. method[0][6]: whether the method is a native method.

Example Requests

POST: /v1/apm2/openapi/view/profiling/flame-line-tree

```
{
  "from": 1704271204595,
  "to": 1704275169491,
  "type": "LATENCY",
  "instance_id": -6870732446083119805,
  "api": "GET_/user/{id}",
  "region":""
}
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "data":[
    100,
    1000,
    null,
    [
      [
        700,
        700,
        0,
        30, // Line number
        []
      ],
      [
        200,
        200,
        1,
        40, // Line number
        []
      ]
    ]
  ],
  "methods":[
    [
      2, // method id
      "com.***.foo",

```

```
"SampleService",
"doService",
"int,long,String",
true, // Whether the method is a user method
false // Whether the method is a native method
],
[
3,
"com.**.bar",
"SampleRemoteCall",
"sampleCallRemoteProcess",
"float,int",
true, // Whether the method is a user method
false // Whether the method is a native method
]
]}
```

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Semantic or parameter error.
401	Unauthorized: No permissions.
403	Forbidden: Access forbidden.
404	Not Found: The requested resource is not found.

Error Codes

See [Error Codes](#).

6 Permissions Policies and Supported Actions

This chapter describes fine-grained permissions management for your APM. If your account does not need individual IAM users, then you may skip over this chapter.

By default, new IAM users do not have any permissions assigned. You need to add a user to one or more groups, and assign permissions policies or roles to these groups. The user then inherits permissions from the groups it is a member of. This process is called authorization. After authorization, the user can perform specified operations on APM.

You can grant users permissions by using roles and policies. Roles are a type of coarse-grained authorization mechanism that defines permissions related to user responsibilities. Policies define API-based permissions for operations on specific resources under certain conditions, allowing for more fine-grained, secure access control of cloud resources.

Policy-based authorization is useful if you want to allow or deny the access to an API.

An account has all the permissions required to call all APIs, and IAM users must be assigned the required permissions. The permissions required for calling an API are determined by the actions supported by the API. Only users who have been granted permissions can call the API successfully. For example, if an IAM user tries to obtain an AK/SK by calling an API, the user must have been granted permissions that allow the **apm:apm2AkSk:read** action.

Supported Actions

APM provides system-defined policies that can be directly used. You can also create custom policies and use them to supplement system-defined policies, implementing more refined access control. Operations supported by policies are specific to APIs. The following are common concepts related to policies:

- Permissions: Defined by actions in a custom policy.
- APIs: REST APIs that can be called by a user who has been granted specific permissions.
- Actions: Specific operations that are allowed or denied.

- Related actions: Actions on which a specific action depends to take effect. When assigning permissions for the action to a user, you also need to assign permissions for the related actions.
- IAM or enterprise projects: Type of projects for which an action will take effect. Policies that contain actions for both IAM and enterprise projects can be used and take effect for both IAM and Enterprise Management. Policies that only contain actions for IAM projects can be used and only take effect for IAM. .

Actions

Permis sion	API	Action	IAM Projec t	Enter prise Projec t
Obtaini ng an AK/SK	GET /v1/apm2/openapi/ systemmng/get-ak-sk- list	apm:apm2AkSk:read	√	√
Queryin g the applicat ion list	GET /v1/apm2/openapi/ cldb/business/get- business-list	-	√	√
Obtaini ng the PodLB address of the master service based on the region name	GET /v1/apm2/openapi/ systemmng/get-master- address	-	√	√

√: supported; x: not supported

7 Appendix

7.1 Error Codes

Status Code	Error Codes	Error Message	Description	Solution
-------------	-------------	---------------	-------------	----------

7.2 Obtaining a Project ID

Obtaining a Project ID from the Console

A project ID is required for some URLs when an API is called. To obtain a project ID from the console, perform the following operations:

- Step 1** Sign up and log in to the management console.
- Step 2** Hover the pointer over the username in the upper right corner and choose **My Credentials** from the drop-down list.
- Step 3** On the displayed **API Credentials** page, view project IDs in the project list.

If a project contains multiple sub-projects, click the plus (+) sign to view sub-project IDs.

----End

Obtaining a Project ID by Calling an API

You can also call the API for querying project information to obtain a project ID.

The API for obtaining a project ID is **GET https://{Endpoint}/v3/projects/**, where *{Endpoint}* indicates the IAM endpoint. For details, see "Regions and Endpoints".

In the following example, **id** indicates a project ID.

```
{
  "projects": [
    {
```

```

    "domain_id": "65382450e8f64ac0870cd180d14e684b",
    "is_domain": false,
    "parent_id": "65382450e8f64ac0870cd180d14e684b",
    "name": "xxxxx",
    "description": "",
    "links": {
      "next": null,
      "previous": null,
      "self": "https://www.example.com/v3/projects/a4a5d4098fb4474fa22cd05f897d6b99"
    },
    "id": "a4a5d4098fb4474fa22cd05f897d6b99",
    "enabled": true
  }
],
"links": {
  "next": null,
  "previous": null,
  "self": "https://www.example.com/v3/projects"
}
}

```

7.3 Obtaining an Account ID

An account ID is required for some URLs when an API is called. To obtain an account ID, perform the following operations:

- Step 1** Sign up and log in to the management console.
- Step 2** Hover the pointer over the username and choose **My Credentials** from the drop-down list.

On the **API Credentials** page, view **Account ID**.

----End

7.4 Obtaining API Parameters

Table 7-1 API parameters

Location	Parameter	Mandatory	Type	Description
Querying the Master Address	region_name	Yes	String	Region name, which can be obtained from the region field in Table 5-34 .
Saving a Monitoring Item	monitor_item_id	Yes	Long	Monitoring item ID, which can be obtained from the monitor_item_id field in Table 5-123 .
Obtaining an Application Tree	region_id	No	String	Region ID, which can be obtained from the region field in Table 5-34 .

Location	Parameter	Mandatory	Type	Description
Obtaining an Application Tree	business_id	Yes	Long	Application ID, which can be obtained from the id field in Table 5-3 .
Obtaining an Application Tree	env_tag_id	No	Long	Environment tag ID, which can be obtained from the id field in Table 5-41 .
Querying Environment Tags	business_id	Yes	Long	Application ID, which can be obtained from the id field in Table 5-3 .
Obtaining the Component List	business_id	Yes	Long	Application ID, which can be obtained from the id field in Table 5-3 .
Obtaining the Environment List of a Component	app_id	Yes	Long	Component ID, which can be obtained from the id field in Table 5-50 .
Querying the Details About an Application	business_id	Yes	Long	Application ID, which can be obtained from the id field in Table 5-3 .
Querying the Details About a Sub-application	sub_business_id	Yes	Long	Sub-application ID, which can be obtained from the real_id of the node whose type is SUB_BUSINESS in Table 5-46 .
Querying Monitoring Item Configurations	env_id	Yes	Long	Environment ID, which can be obtained from real_id of the node whose type is ENVIRONMENT in Table 5-46 .
Querying Monitoring Item Configurations	collector_id	Yes	Long	Collector ID, which can be obtained from the collector_id field in Table 5-123 .
Querying the Trace Topology	trace_id	Yes	String	Trace ID, which can be obtained from the trace_id field in Table 5-85 .

Location	Parameter	Mandatory	Type	Description
Querying Event Details	trace_id	Yes	String	Trace ID, which can be obtained from the trace_id field in Table 5-89 .
Querying Event Details	span_id	Yes	String	Span ID, which can be obtained from the span_id field in Table 5-89 .
Querying Event Details	event_id	Yes	String	Event ID, which can be obtained from the event_id field in Table 5-89 .
Querying Event Details	env_id	Yes	Long	Environment ID, which can be obtained from the env_id field in Table 5-89 .
Querying Span Data	env_id	No	Long	Environment ID, which can be obtained from the id field in Table 5-54 .
Querying Span Data	instance_id	No	Long	Instance ID, which can be obtained from the instance_id field in Table 5-118 .
Querying Span Data	app_id	No	Long	Component ID, which can be obtained from the id field in Table 5-50 .
Querying Span Data	biz_id	Yes	Long	Application ID, which can be obtained from the id field in Table 5-3 .
Obtaining All Data of a Trace	trace_id	Yes	String	Trace ID, which can be obtained from the trace_id field in Table 5-85 .
Obtaining the Trend Graph	view_config	Yes	Table3 TrendView object	View configuration information, which can be obtained from the ViewBase object in Table 5-68 .

Location	Parameter	Mandatory	Type	Description
Obtaining the Trend Graph	instance_id	No	Long	Instance ID, which can be obtained from the instance_id field in Table 5-118 . If this parameter is left blank, the aggregated data of all instances will be obtained.
Obtaining the Trend Graph	monitor_item_id	Yes	Long	Monitoring item ID, which can be obtained from the monitor_item_id field in Table 5-123 .
Obtaining the Trend Graph	env_id	Yes	Long	Environment ID, which can be obtained from real_id of the node whose type is ENVIRONMENT in Table 5-46 .
Obtaining Summary Table Data	view_config	Yes	Table3 SumTableView object	View configuration information, which can be obtained from the ViewBase object in Table 5-68 .
Obtaining Summary Table Data	instance_id	No	Long	Instance ID, which can be obtained from the instance_id field in Table 5-118 . If this parameter is left blank, the aggregated data of all instances will be obtained.
Obtaining Summary Table Data	monitor_item_id	Yes	Long	Monitoring item ID, which can be obtained from the monitor_item_id field in Table 5-123 .
Obtaining Summary Table Data	env_id	Yes	Long	Environment ID, which can be obtained from real_id of the node whose type is ENVIRONMENT in Table 5-46 .

Location	Parameter	Mandatory	Type	Description
Obtaining the Raw Data Table	view_config	Yes	Table 3 RawTableView object	Raw data table view, which can be obtained from the ViewBase object in Table 5-68 .
Obtaining the Raw Data Table	instance_id	Yes	Long	Instance ID, which can be obtained from the instance_id field in Table 5-118 . If this parameter is left blank, the aggregated data of all instances will be obtained.
Obtaining the Raw Data Table	monitor_item_id	Yes	Long	Monitoring item ID, which can be obtained from the monitor_item_id field in Table 5-123 .
Obtaining the Raw Data Table	env_id	Yes	Long	Environment ID, which can be obtained from real_id of the node whose type is ENVIRONMENT in Table 5-46 .
Obtaining the Instance Information	env_id	Yes	Long	Environment ID, which can be obtained from real_id of the node whose type is ENVIRONMENT in Table 5-46 .
Obtaining the Monitoring Item Information	env_id	Yes	Long	Environment ID, which can be obtained from real_id of the node whose type is ENVIRONMENT in Table 5-46 .
Querying the Alarm List	region	No	String	Region name, which can be obtained from the region field in Table 5-34 .
Querying the Alarm List	business_id	Yes	Long	Application ID, which can be obtained from the id field in Table 5-3 .

Location	Parameter	Mandatory	Type	Description
Querying the Alarm List	monitor_item_id	No	Long	Monitoring item ID, which can be obtained from the monitor_item_id field in Table 5-123 .
Querying the Alarm List	collector_id	No	Integer	Collector ID, which can be obtained from the collector_id field in Table 5-123 .
Querying the Alarm Message List	alarm_data_id	Yes	Integer	Alarm/event ID, which can be obtained from the id field in Table 5-143 .
Querying the Alarm Message List	region	Yes	String	Region name, which can be obtained from the region field in Table 5-34 .
Querying the Global Topology of an Application	target_business_id	Yes	Long	Target application ID, which can be obtained from the id field in Table 5-3 .
Table2 Request body parameters	env_tag_list	No	Array of integers	Environment tag list, which can be empty or obtained from the id field in Table 5-41 .
Table 5-159	target_env_id	Yes	Long	Environment ID, which can be obtained from real_id of the node whose type is ENVIRONMENT in Table 5-46 .
Querying the URL Tracing Configuration List	business_id	Yes	Long	Application ID, which can be obtained from the id field in Table 5-3 .
Querying the URL Tracing Configuration List	env_id	No	Long	Environment ID, which can be obtained from the id field in Table 5-41 .

Location	Parameter	Mandatory	Type	Description
Querying a Region's Environments on Which URLs Are Added for Tracing	business_id	Yes	Long	Application ID, which can be obtained from the id field in Table 5-3 .
Querying a Region's Environments on Which URLs Are Added for Tracing	region	Yes	String	Region name, which can be obtained from the region field in Table 5-34 .
Querying the URL Tracing View List	business_id	Yes	Long	Application ID, which can be obtained from the id field in Table 5-3 .
Querying the URL Tracing View List	region	Yes	String	Region name, which can be obtained from the region field in Table 5-34 .
Querying the URL Tracing View List	env_id	No	Long	Environment ID, which can be obtained from the env_id field in Table 5-175 .
Querying the Details About a URL Tracing View	tx_name	Yes	String	Transaction name, which is obtained from the tx_name field in Table 5-179 .