

Distributed Message Service for RocketMQ

API Reference

Issue 01
Date 2023-01-09



Copyright © Huawei Cloud Computing Technologies Co., Ltd. 2023. 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 API Calling	1
1.3 Endpoints	1
1.4 Concepts	2
2 API Overview	3
3 Calling APIs	4
3.1 Making an API Request	4
3.2 Authentication	8
3.3 Returned Values	10
4 Getting Started	12
5 APIs V2 (Recommended)	14
5.1 Lifecycle Management	14
5.1.1 Creating a Pay-per-Use Instance	14
5.1.2 Querying the Instance List	18
5.1.3 Querying an Instance	24
5.1.4 Deleting an Instance	29
5.1.5 Modifying Instance Information	30
5.1.6 Batch Deleting Instances	32
5.1.7 Querying the Broker List	33
5.2 Consumer Group Management	35
5.2.1 Querying the Consumer Group List	35
5.2.2 Creating a Consumer Group or Batch Deleting Consumer Groups	37
5.2.3 Batch Modifying Consumer Groups	40
5.2.4 Deleting a Consumer Group	41
5.2.5 Querying a Consumer Group	42
5.2.6 Modifying a Consumer Group	44
5.2.7 Querying the Consumer Group List or Details	45
5.2.8 Resetting the Consumer Offset	47
5.3 Topic Management	49
5.3.1 Creating a Topic or Deleting Topics in Batches	49
5.3.2 Deleting a Topic	51

5.3.3 Querying a Topic.....	52
5.3.4 Modifying a Topic.....	54
5.3.5 Querying the Topic Consumer Group List.....	55
5.3.6 Querying the Number of Messages in a Topic.....	56
5.4 Message Management.....	58
5.4.1 Querying Messages.....	58
5.4.2 Querying the Message Trace.....	61
5.4.3 Exporting Dead Letter Messages.....	64
5.5 User Management.....	66
5.5.1 Creating a User.....	66
5.5.2 Querying the User List.....	69
5.5.3 Modifying User Parameters.....	72
5.5.4 Querying User Details.....	75
5.5.5 Deleting a User.....	77
5.5.6 Querying Users Granted Permissions for a Topic.....	78
5.5.7 Querying Users Granted Permissions for a Consumer Group.....	80
5.6 Other APIs.....	82
5.6.1 Querying AZ Information.....	82
6 Permissions Policies and Supported Actions.....	85
7 Appendix.....	88
7.1 Status Code.....	88
7.2 Error Codes.....	91
7.3 Instance Statuses.....	111
7.4 Obtaining a Project ID.....	111
7.5 Obtaining the Account Name and Account ID.....	112
A Change History.....	113

1 Before You Start

1.1 Overview

Welcome to *Distributed Message Service for RocketMQ API Reference*. Distributed Message Service (DMS) for RocketMQ is message-oriented middleware that delivers low latency, high flexibility, high throughput, dynamic expansion, easy management, and abundant messaging functions.

This document describes functions, syntax, parameters, and examples of the application programming interfaces (APIs) of DMS for RocketMQ.

NOTICE

DMS for RocketMQ is continuously upgraded with new functions, and the existing APIs are inevitably adjusted. For example, new response parameters may be added.

To reduce the impact of API changes, DMS for RocketMQ is backward compatible with existing APIs. When using DMS, you should accept and ignore unused parameters and parameter values in JSON responses.

1.2 API Calling

DMS for RocketMQ supports Representational State Transfer (REST) APIs, allowing you to call APIs using HTTPS. For details about API calling, see [Calling APIs](#).

1.3 Endpoints

An endpoint is the **request address** for calling an API. Endpoints vary depending on services and regions. For the endpoints of all services, see [Regions and Endpoints](#).

1.4 Concepts

- **Account**
An 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 IAM users and grant them permissions for routine management.
- **IAM user**
An Identity and Access Management (IAM) user is created using an account to use cloud services. Each IAM user has its own identity credentials (password and access keys).
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. Deploying cloud resources in different regions can better suit certain user requirements or comply with local laws or regulations.
- **An AZ** contains one or more physical data centers. Each AZ has independent cooling, fire extinguishing, moisture-proof, and electricity facilities. Within an AZ, compute, network, storage, and other resources are logically divided into multiple clusters. AZs within a region are interconnected using high-speed optical fibers to support cross-AZ high-availability systems.
- **Project**
Projects group and isolate resources (including compute, storage, and network resources) across physical regions. A default project is provided for each region, and subprojects can be created under each default project. Users can be granted permissions to access all resources in a specific project. For more refined access control, create subprojects under a project and purchase resources in the subprojects. Users can then be assigned permissions to access only specific resources in the subprojects.
- **Enterprise project**
Enterprise projects group and manage resources across regions. Resources in enterprise projects are logically isolated from each other. An enterprise project can contain resources of multiple regions, and resources can be added to or removed from enterprise projects.

2 API Overview

Table 2-1 APIs for managing instances

API	Description
Lifecycle management	Create instances, modify instance information, query and delete instances, and query the broker list.
Consumer group management	Query, create, delete, and modify consumer groups, query the consumer list or details, and reset the consumer offset.
Topic management	Create, delete, query, and modify topics, query the consumer group list of a topic, and query the number of messages in a topic.
Message management	Query messages and their traces, and export dead letter messages.
User management	Create, query, modify, and delete users, query the authorized user list of a topic, and query the authorized user list of a consumer group.
Other APIs	Query AZ information.

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 demonstrate 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.

Table 3-1 Parameters in a URI

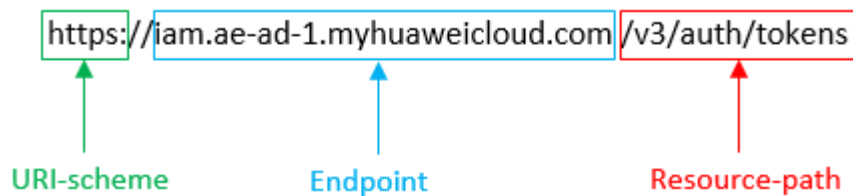
Parameter	Description
URI-scheme	Protocol used to transmit requests. All APIs use HTTPS.
Endpoint	Domain name or IP address of the server bearing the REST service. The endpoint varies between services in different regions. It can be obtained from Regions and Endpoints . For example, the endpoint of IAM in the UAE-Abu Dhabi region is iam.ae-ad-1.myhuaweicloud.com .
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 .

Parameter	Description
query-string	Query parameter, which is optional. Ensure that a question mark (?) is included before each query parameter that is in the format of <i>Parameter name=Parameter value</i> . For example, ?limit=10 indicates that a maximum of 10 data records will be displayed.

For example, to obtain an IAM token in the **ae-ad-1** region, obtain the endpoint of IAM (**iam.ae-ad-1.myhuaweicloud.com**) for this region and the **resource-path** (**/v3/auth/tokens**) in the URI of the API used to **obtain a user token**. Then, construct the URI as follows:

```
https://iam.ae-ad-1.myhuaweicloud.com/v3/auth/tokens
```

Figure 3-1 Example URI



NOTE

To simplify the URI display in this document, 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 can be created using the PATCH method.

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://iam.ae-ad-1.myhuaweicloud.com/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 3-2 lists the common request header fields.

Table 3-2 Common request header fields

Name	Description	Mandatory	Example Value
Host	Request server information, which is obtained from the URL of a service API. The value is in the format of <i>Hostname:Port number</i> . If no port is specified, the default port will be used. For HTTPS, port 443 is used by default.	No This parameter is mandatory for AK/SK authentication.	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. Obtain the project ID by following the instructions in Obtaining a Project ID .	No	e9993fc787d94b6c886cb aa340f9c0f4

Name	Description	Mandatory	Example Value
X-Auth-Token	<p>User token.</p> <p>The user token is a response to the API used to obtain a user token. This API is the only one that does not require authentication.</p> <p>After the request is processed, the value of X-Subject-Token in the response header is the token value.</p>	No Mandatory for token-based authentication.	The following is part of an example token: MIIPAgYJKoZlhvcNAQc-Co...ggg1BBIINPXsidG9rZ

 **NOTE**

In addition to supporting token-based authentication, cloud service APIs also support authentication using the AK/SK. During AK/SK-based authentication, an SDK is used to sign the request, and the **Authorization** (signature information) and **X-Sdk-Date** (time when the request is sent) header fields are automatically added to the request.

For more information, see "AK/SK-based Authentication" in [Authentication](#).

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://iam.ae-ad-1.myhuaweicloud.com/v3/auth/tokens
Content-Type: application/json
```

Request Body (Optional)

The body of a request is often sent in a structured format as specified in the **Content-type** header field, such as JSON or XML. 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) with the actual values (for example,). The project name can be obtained from [Regions and Endpoints](#).

 NOTE

The **scope** parameter specifies where a token takes effect. You can set **scope** to an account or a project under an account. 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://iam.ae-ad-1.myhuaweicloud.com/v3/auth/tokens
Content-Type: application/json
```

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

If all data required by a request is available, you can send the request to call the 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

Requests for calling an API can be authenticated using either of the following methods:

- Token-based authentication: Requests are authenticated using a token.
- AK/SK authentication: Requests are encrypted using AK/SK pairs. AK/SK-based authentication is recommended because it is more secure than token-based authentication.

Token-based Authentication

 NOTE

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 request headers to get permissions for calling the API. You can obtain a token by [calling an API](#).

A cloud service can be deployed as either a project-level service or global service.

- For a project-level service, you need to obtain a project-level token. When you call the API, set **auth.scope** in the request body to **project**.
- For a global service, you need to obtain a global token. When you call the API, set **auth.scope** in the request body to **domain**.

When calling the API used to [obtain a user token](#), you must 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 **ABCDEFJ....**, **X-Auth-Token: ABCDEFJ....** can be added to a request as follows:

```
POST https://iam.ae-ad-1.myhuaweicloud.com/v3/auth/projects
Content-Type: application/json
X-Auth-Token: ABCDEFJ....
```

AK/SK-based Authentication

NOTE

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 request headers 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 to sign requests based on the signature algorithm or use the signing SDK to sign requests. For details about how to sign requests and use the signing SDK, see [API Request Signing Guide](#).

NOTICE

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

3.3 Returned Values

Status Code

After sending a request, you will receive a response, including the 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 response. For more information, see [Status Code](#).

If status code **201** is returned for the calling of the API for [obtaining a user token](#), the request is successful.

Response Header

A response header corresponds to a request header, for example, **Content-type**.

[Figure 3-2](#) shows the response header for the API of [obtaining a user token](#), in which **x-subject-token** is the desired user token. Then, you can use the token to authenticate the calling of other APIs.

Figure 3-2 Header 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
→ MIiYXQYJKoZIhvcNAQcCoIIYJCCGEoCAQExDTALBglghkgBZQMEAgEwgharBgkqhkiG9w0BBwGgghacBIIWmHsidG9rZW4iOnsiZXhwaXJlc19hdCI6IiwMTktMDItMTNUMD.
fj3Kjs6YgKnpVNRbW2eZ5eb78SZOkqjACgklqO1wi4JlGzrpd18LGXK5tdfdq4lqHCYb8P4NaYONyejcAgzJVeFYtLWT1GSO0zxKZmLQHJQ82HBqHdglZO9fuEbl5dMhdavj+33wEI
xHRCe9I87o+k9-
j+CMZSEB7bUGd5Uj6eRASXlIjipPEGA270g1FruooL6jqglFKNPQuFSOU8+uSsttVwRtnfsC+qTp22Rkd5MCqFGQ8LcuUx3a+9CMBnOintWW7oeRUVhVpxk8pxiX1wTEboX-
RzT6MUUbpvGw-oPNFYxJECKnoH3HRozv0vN--n5d6Nbxg==

x-xss-protection → 1; mode=block;
    
```

(Optional) Response Body

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

You can obtain a token by **calling an API**. The following shows part of the response body for the API to obtain a user token.

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

```

If an error occurs during API calling, the system returns an error code and a message to you. The following shows the format of an error response body:

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

In the preceding information, **error_code** is an error code, and **error_msg** describes the error.

4 Getting Started

Scenarios

This section describes how to call an API to create a RocketMQ instance and customize the computing capabilities and storage space of the instance based on service requirements.

For details on how to call APIs, see [Calling APIs](#).

Prerequisites

The endpoints of IAM and RocketMQ have been obtained. For details, see [Regions and Endpoints](#).

Creating a RocketMQ Instance

The following is an example request for creating a RocketMQ instance:

```
{
  "name" : "rocketmq-test",
  "description" : "",
  "engine" : "reliability",
  "engine_version" : "4.8.0",
  "storage_space" : 600,
  "vpc_id" : "3db8490c-*****-047b0de4c5cf",
  "subnet_id" : "0a0f1fcb-*****-301867394d50",
  "security_group_id" : "23c5977f-*****-08d8a0bc4e6c",
  "available_zones" : [ "9f1c58067*****a9b18c77" ],
  "product_id" : "c6.4u8g.cluster",
  "ssl_enable" : false,
  "storage_spec_code" : "dms.physical.storage.ultra.v2",
  "ipv6_enable" : false,
  "enable_publicip" : false,
  "publicip_id" : ""
}
```

- **name**: name of the instance.
- **description**: description of the instance.
- **engine**: message engine. The value is **reliability**.
- **engine_version**: version of the message engine.
- **storage_space**: message storage space in MB. For details about the value range, see [Creating a Pay-per-Use Instance](#).

- **vpc_id**: ID of the Virtual Private Cloud (VPC) where the RocketMQ instance resides. Obtain the value by using the API described in [Creating a Pay-per-Use Instance](#).
- **subnet_id**: ID of the VPC subnet. Obtain the value by using the API described in [Creating a Pay-per-Use Instance](#).
- **security_group_id**: ID of the security group. Obtain the value by using the API described in [Creating a Pay-per-Use Instance](#).
- **available_zones**: ID of the AZ where the instance resides. The value cannot be empty or null. Obtain the value by calling the API described in [Querying AZ Information](#).
- **product_id**: ID of the product. For details about the value range, see [Creating a Pay-per-Use Instance](#).
- **ssl_enable**: Indicates whether to enable SSL-encrypted access.
- **storage_spec_code**: storage I/O specification. For details about the value range, see [Creating a Pay-per-Use Instance](#).
- **ipv6_enable**: Indicates whether to support IPv6.
- **enable_publicip**: Indicates whether to enable public access.
- **publicip_id**: ID of the elastic IP address (EIP) bound to the instance.

5 APIs V2 (Recommended)

5.1 Lifecycle Management

5.1.1 Creating a Pay-per-Use Instance

Function

This API is used to create a pay-per-use instance.

URI

POST /v2/{project_id}/instances

Table 5-1 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.

Request Parameters

Table 5-2 Request body parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Instance name. An instance name starts with a letter, consists of 4 to 64 characters, and can contain only letters, digits, underscores (_), and hyphens (-).

Parameter	Mandatory	Type	Description
description	No	String	Instance description. The description can contain a maximum of 1024 characters. NOTE The backslash (\) and quotation mark (") are special characters for JSON messages. When using these characters in a parameter value, add the escape character (\) before the characters, for example, \ and \".
engine	Yes	String	Message engine. Value: reliability .
engine_version	Yes	String	Version of the message engine. Value: 4.8.0 .
storage_space	Yes	Integer	Storage space
vpc_id	Yes	String	VPC ID. To obtain it, perform the following steps: <ul style="list-style-type: none"> Log in to the VPC console and view the VPC ID in the VPC details.
subnet_id	Yes	String	Subnet information. To obtain it, perform the following steps: <ul style="list-style-type: none"> Log in to the VPC console and click the target subnet on the Subnets page. You can view the network ID on the displayed page.
security_group_id	Yes	String	ID of the security group to which the instance belongs. To obtain it, perform the following steps: <ul style="list-style-type: none"> Log in to the VPC console and view the security group ID on the security group details page.

Parameter	Mandatory	Type	Description
available_zones	Yes	Array of strings	<p>ID of the AZ where instance brokers reside and which has available resources. The parameter value cannot be empty or null. Check whether resources exist in the AZs.</p> <p>A RocketMQ instance can be deployed in 1 AZ or at least 3 AZs. When specifying AZs for brokers, separate multiple AZs with commas (,).</p>
product_id	Yes	String	<p>RocketMQ instance specifications.</p> <ul style="list-style-type: none"> • c6.4u8g.cluster: maximum number of topics on each broker: 4000; maximum number of consumer groups on each broker: 4000 • c6.8u16g.cluster: maximum number of topics on each broker: 8000; maximum number of consumer groups on each broker: 8000 • c6.12u24g.cluster: maximum number of topics on each broker: 12,000; maximum number of consumer groups on each broker: 12,000 • c6.16u32g.cluster: maximum number of topics on each broker: 16,000; maximum number of consumer groups on each broker: 16,000
ssl_enable	No	Boolean	<p>Whether to enable SSL encryption for access.</p> <ul style="list-style-type: none"> • true: enable • false: disable

Parameter	Mandatory	Type	Description
storage_spec_code	Yes	String	Storage I/O specification. <ul style="list-style-type: none"> • dms.physical.storage.high.v2: high I/O disk • dms.physical.storage.ultra.v2: ultra-high I/O disk
enterprise_project_id	No	String	Enterprise project ID. This parameter is mandatory for an enterprise project account.
enable_acl	No	Boolean	Whether to enable access control.
ipv6_enable	No	Boolean	Whether to support IPv6. <ul style="list-style-type: none"> • true: support • false: do not support Default: false
enable_publicip	No	Boolean	Whether to enable public access. By default, public access is disabled. <ul style="list-style-type: none"> • true: enable • false: do not enable Default: false
publicip_id	No	String	ID of the EIP bound to the instance. Use commas (,) to separate multiple EIP IDs. This parameter is mandatory if public access is enabled (that is, enable_publicip is set to true).
broker_num	Yes	Integer	Number of brokers.

Response Parameters

Status code: 200

Table 5-3 Response body parameters

Parameter	Type	Description
instance_id	String	Instance ID.

Example Requests

Creating a pay-per-use instance

```
POST https://{endpoint}/v2/{project_id}/instances

{
  "name" : "reliability-1751840557",
  "description" : "",
  "engine" : "reliability",
  "engine_version" : "4.8.0",
  "storage_space" : 600,
  "vpc_id" : "3db8490c-4d6d-4d8f-8d3f-047b0de4c5cf",
  "subnet_id" : "0a0f1fcb-f019-458d-b9e5-301867394d50",
  "security_group_id" : "23c5977f-ff33-4b95-a73e-08d8a0bc4e6c",
  "available_zones" : [ "9f1c5806706d4c1fb0eb72f0a9b18c77" ],
  "product_id" : "c6.4u8g.cluster",
  "enterprise_project_id" : "0",
  "ssl_enable" : false,
  "storage_spec_code" : "dms.physical.storage.ultra.v2",
  "ipv6_enable" : false,
  "enable_publicip" : false,
  "publicip_id" : "",
  "broker_num" : 1
}
```

Example Responses

Status code: 200

Instance created successfully.

```
{
  "instance_id" : "8959ab1c-7n1a-yyb1-a05t-93dfc361b32d"
}
```

Status Codes

Status Code	Description
200	Instance created successfully.

Error Codes

See [Error Codes](#).

5.1.2 Querying the Instance List

Function

This API is used to query the instances of an account by the specified conditions.

URI

GET /v2/{project_id}/instances

Table 5-4 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.

Table 5-5 Query Parameters

Parameter	Mandatory	Type	Description
engine	No	String	Message engine. Value: reliability .
name	No	String	Instance name.
instance_id	No	String	Instance ID.
status	No	String	Instance status.
include_failure	No	String	Whether to return the number of instances that fail to be created. If the value is true , the number of instances that failed to be created is returned. If the value is not true , the number is not returned.
exact_match_name	No	String	Whether to search for the instance that precisely matches a specified instance name. The default value is <i>false*</i> , indicating that a fuzzy search is performed based on a specified instance name. If the value is true , the instance that precisely matches a specified instance name is queried.
enterprise_project_id	No	String	Enterprise project ID.

Request Parameters

None

Response Parameters

Status code: 200

Table 5-6 Response body parameters

Parameter	Type	Description
instances	Array of ShowInstanceResp objects	Instance list.
instance_num	Integer	Number of instances.

Table 5-7 ShowInstanceResp

Parameter	Type	Description
name	String	Instance name.
engine	String	Message engine.
status	String	Status.
description	String	Message description.
type	String	Instance type. Value: cluster .
specification	String	Instance specification.
engine_version	String	Version.
instance_id	String	Instance ID.
charging_mode	Integer	Billing mode. 1 indicates pay-per-use.
vpc_id	String	VPC ID.
vpc_name	String	VPC name.
created_at	String	Time when the instance is created. The time is a Unix timestamp in millisecond.
product_id	String	Product ID.
security_group_id	String	Security group ID.
security_group_name	String	Security group name.
subnet_id	String	Subnet ID.
subnet_name	String	Subnet name.
subnet_cidr	String	Subnet route.
available_zones	Array of strings	List of AZs where there are available I/O resources.

Parameter	Type	Description
user_id	String	User ID.
user_name	String	Username.
maintain_begin	String	Time at which the maintenance window starts. The format is HH:mm:ss.
maintain_end	String	Time at which the maintenance window ends. The format is HH:mm:ss.
enable_log_collection	Boolean	Whether log collection is enabled.
storage_space	Integer	Storage space. Unit: GB.
used_storage_space	Integer	Used message storage space. Unit: GB.
enable_publicip	Boolean	Whether public access is enabled.
publicip_id	String	ID of the EIP bound to the instance. Use commas (,) to separate multiple EIP IDs. This parameter is mandatory if public access is enabled (that is, enable_publicip is set to true).
publicip_address	String	Public IP address.
ssl_enable	Boolean	Whether SSL is enabled.
cross_vpc_info	String	Cross-VPC access information.
storage_resource_id	String	Storage resource ID.
storage_spec_code	String	Storage specification code.
service_type	String	Service type.
storage_type	String	Storage category.
extend_times	Integer	Extension time.
ipv6_enable	Boolean	Whether IPv6 is enabled.
support_features	String	Features supported by the instance.
disk_encrypted	Boolean	Whether disk encryption is enabled.
ces_version	String	Cloud Eye version.
node_num	Integer	Node quantity.

Parameter	Type	Description
new_spec_billing_enable	Boolean	Whether billing based on new specifications is enabled.
enable_acl	Boolean	Whether access control is enabled.
broker_num	Integer	Broker quantity.
namesrv_address	String	Metadata address.
broker_addresses	String	Service data address.
public_namesrv_address	String	Public network metadata address.
public_broker_address	String	Public network service data address.
tags	Array of TagEntity objects	Tag list.
total_storage_space	Integer	Total storage space.
resource_spec_code	String	Resource specifications.

Table 5-8 TagEntity

Parameter	Type	Description
key	String	Tag key, which can contain a maximum of 36 Unicode characters. The key cannot be left blank or be an empty string. It cannot contain nonprintable ASCII (0-31) characters and the following special characters: =*<>, /
value	String	Tag value, which can contain a maximum of 43 Unicode characters. The value cannot be left blank or be an empty string. It cannot contain nonprintable ASCII (0-31) characters and the following special characters: =*<>, /

Example Requests

Querying all instances

```
GET https://{endpoint}/v2/{project_id}/instances?engine=reliability
```

Example Responses

Status code: 200

Instance list queried successfully.

```
{
  "instances": [ {
    "name": "reliability-ztest",
    "engine": "reliability",
    "status": "RUNNING",
    "description": "",
    "type": "cluster",
    "specification": "c6.4u8g.cluster * 1 broker",
    "engine_version": "4.8.0",
    "instance_id": "68fdc9a8-805e-439d-8dd9-25adc1c58bf6",
    "resource_spec_code": "",
    "charging_mode": 1,
    "vpc_id": "3db8490c-4d6d-4d8f-8d3f-047b0de4c5cf",
    "vpc_name": "vpc-1101840",
    "created_at": "1636699753874",
    "product_id": "c6.4u8g.cluster",
    "security_group_id": "23c5977f-ff33-4b95-a73e-08d8a0bc4e6c",
    "security_group_name": "Sys-default",
    "subnet_id": "0a0f1fcb-f019-458d-b9e5-301867394d50",
    "available_zones": [ "9f1c5806706d4c1fb0eb72f0a9b18c77" ],
    "user_id": "0b01fbb53600d4671fa8c00673c71260",
    "user_name": "",
    "maintain_begin": "02:00:00",
    "maintain_end": "06:00:00",
    "enable_log_collection": false,
    "storage_space": 558,
    "total_storage_space": 600,
    "used_storage_space": 28,
    "enable_publicip": false,
    "ssl_enable": false,
    "cross_vpc_info": "{\n  \"advertised_ip\": \"192.168.1.21\",\n  \"broker_port\": \"-\",\n  \"port_id\": \"fa020857-d899-497c-a5f8-8dd90ed67ff7\",\n  \"namesrv_port\": \"8301\",\n  \"192.168.1.246\": {\n    \"advertised_ip\": \"192.168.1.246\",\n    \"broker_port\": \"10100\",\n    \"port_id\": \"c0f0586f-a4ee-41b5-a7f1-b4e554bbf12d\",\n    \"namesrv_port\": \"-\",\n    \"192.168.1.77\": {\n      \"advertised_ip\": \"192.168.1.77\",\n      \"broker_port\": \"10101\",\n      \"port_id\": \"8eb1d144-4315-402f-8498-37c9a10b630c\",\n      \"namesrv_port\": \"-\",\n      \"192.168.1.159\": {\n        \"advertised_ip\": \"192.168.1.159\",\n        \"broker_port\": \"-\",\n        \"port_id\": \"d35b22af-a202-4329-bda4-26e1bdf2aa8e\",\n        \"namesrv_port\": \"8300\",\n        \"192.168.1.14\": {\n          \"advertised_ip\": \"192.168.1.14\",\n          \"broker_port\": \"10102\",\n          \"port_id\": \"5fcb50fd-3af5-4123-a455-a67f1b926026\",\n          \"namesrv_port\": \"-\"}}},\n    \"storage_resource_id\": \"164bdaef-2e67-4fd5-be8b-a18f91d455a2\",\n    \"storage_spec_code\": \"dms.physical.storage.ultra.v2\",\n    \"service_type\": \"advanced\",\n    \"storage_type\": \"hec\",\n    \"extend_times\": 0,\n    \"ipv6_enable\": false,\n    \"support_features\":\n      \"kafka.crossvpc.domain.enable,feature.physerver.kafka.topic.accesspolicy,rabbitmq.plugin.management,rocketmq.acl,roma_app_enable,auto_topic_switch,feature.physerver.kafka.user.manager,enable.new.authinfo,route,kafka.config.dynamic.modify.enable,kafka.new.pod.port,feature.physerver.kafka.topic.modify,message_trace_enable,features.pod.token.access,kafka.ssl.cert.modify.enable,roma.user.manage.no.support,feature.physerver.kafka.publbc.dynamic,features.log.collection,kafka.config.static.modify.enable\",\n    \"disk_encrypted\": false,\n    \"ces_version\": \"linux,v1,v2\",\n    \"node_num\": 5,\n    \"new_spec_billing_enable\": true,\n    \"broker_num\": 1,\n    \"namesrv_address\": \"****\",
```

```
"broker_address" : "****",  
"public_namesrv_address" : "****",  
"public_broker_address" : "****"  
}],  
"instance_num" : 1  
}
```

Status Codes

Status Code	Description
200	Instance list queried successfully.

Error Codes

See [Error Codes](#).

5.1.3 Querying an Instance

Function

This API is used to query the details about a specified instance.

URI

GET /v2/{project_id}/instances/{instance_id}

Table 5-9 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
instance_id	Yes	String	Instance ID.

Request Parameters

None

Response Parameters

Status code: 200

Table 5-10 Response body parameters

Parameter	Type	Description
name	String	Instance name.
engine	String	Message engine.

Parameter	Type	Description
status	String	Status.
description	String	Message description.
type	String	Instance type. Value: cluster .
specification	String	Instance specification.
engine_version	String	Version.
instance_id	String	Instance ID.
charging_mode	Integer	Billing mode. 1 indicates pay-per-use.
vpc_id	String	VPC ID.
vpc_name	String	VPC name.
created_at	String	Time when the instance is created. The time is a Unix timestamp in millisecond.
product_id	String	Product ID.
security_group_id	String	Security group ID.
security_group_name	String	Security group name.
subnet_id	String	Subnet ID.
subnet_name	String	Subnet name.
subnet_cidr	String	Subnet route.
available_zones	Array of strings	List of AZs where there are available I/O resources.
user_id	String	User ID.
user_name	String	Username.
maintain_begin	String	Time at which the maintenance window starts. The format is HH:mm:ss.
maintain_end	String	Time at which the maintenance window ends. The format is HH:mm:ss.
enable_log_collection	Boolean	Whether log collection is enabled.
storage_space	Integer	Storage space. Unit: GB.
used_storage_space	Integer	Used message storage space. Unit: GB.

Parameter	Type	Description
enable_publicip	Boolean	Whether public access is enabled.
publicip_id	String	ID of the EIP bound to the instance. Use commas (,) to separate multiple EIP IDs. This parameter is mandatory if public access is enabled (that is, enable_publicip is set to true).
publicip_address	String	Public IP address.
ssl_enable	Boolean	Whether SSL is enabled.
cross_vpc_info	String	Cross-VPC access information.
storage_resource_id	String	Storage resource ID.
storage_spec_code	String	Storage specification code.
service_type	String	Service type.
storage_type	String	Storage category.
extend_times	Integer	Extension time.
ipv6_enable	Boolean	Whether IPv6 is enabled.
support_features	String	Features supported by the instance.
disk_encrypted	Boolean	Whether disk encryption is enabled.
ces_version	String	Cloud Eye version.
node_num	Integer	Node quantity.
new_spec_billing_enable	Boolean	Whether billing based on new specifications is enabled.
enable_acl	Boolean	Whether access control is enabled.
broker_num	Integer	Broker quantity.
namesrv_address	String	Metadata address.
broker_addresses	String	Service data address.
public_namesrv_address	String	Public network metadata address.

Parameter	Type	Description
public_broker_address	String	Public network service data address.
tags	Array of TagEntity objects	Tag list.
total_storage_space	Integer	Total storage space.
resource_spec_code	String	Resource specifications.

Table 5-11 TagEntity

Parameter	Type	Description
key	String	Tag key, which can contain a maximum of 36 Unicode characters. The key cannot be left blank or be an empty string. It cannot contain nonprintable ASCII (0-31) characters and the following special characters: =*<>, /
value	String	Tag value, which can contain a maximum of 43 Unicode characters. The value cannot be left blank or be an empty string. It cannot contain nonprintable ASCII (0-31) characters and the following special characters: =*<>, /

Example Requests

Querying an instance

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}
```

Example Responses

Status code: 200

Instance queried successfully.

```
{
  "name" : "reliability-test",
  "engine" : "reliability",
  "status" : "RUNNING",
  "description" : ""
}
```

```
"type" : "cluster",
"specification" : "c6.4u8g.cluster * 1 broker",
"engine_version" : "4.8.0",
"instance_id" : "68fdc9a8-805e-439d-8dd9-25adc1c58bf6",
"resource_spec_code" : "",
"charging_mode" : 1,
"vpc_id" : "3db8490c-4d6d-4d8f-8d3f-047b0de4c5cf",
"vpc_name" : "vpc-1101840",
"created_at" : "1636699753874",
"product_id" : "c6.4u8g.cluster",
"security_group_id" : "23c5977f-ff33-4b95-a73e-08d8a0bc4e6c",
"security_group_name" : "Sys-default",
"subnet_id" : "0a0f1fcb-f019-458d-b9e5-301867394d50",
"subnet_name" : "subnet-boce",
"subnet_cidr" : "192.168.1.0/24",
"available_zones" : [ "9f1c5806706d4c1fb0eb72f0a9b18c77" ],
"user_id" : "0b01fbb53600d4671fa8c00673c71260",
"user_name" : "",
"maintain_begin" : "02:00:00",
"maintain_end" : "06:00:00",
"enable_log_collection" : false,
"storage_space" : 558,
"total_storage_space" : 600,
"used_storage_space" : 28,
"enable_publicip" : true,
"publicip_id" : "7e6b7beb-ef13-4805-878d-285890b17a12,87827295-84e1-4118-9ab3-a90dd8c9dace,91306a18-5781-4529-b739-5aeec10ec4e6,a68e8f0e-a24f-4521-910e-39696a56fcdf,bb1cd147-a7af-46ca-9548-99fec4034ba2",
"publicip_address" : "****",
"ssl_enable" : false,
"cross_vpc_info" : "{\"192.168.1.21\":{\"advertised_ip\":\"192.168.1.21\",\"broker_port\":\"-\",\"port_id\":\"fa020857-d899-497c-a5f8-8dd90ed67ff7\",\"namesrv_port\":\"8301\"},\"192.168.1.246\":{\"advertised_ip\":\"192.168.1.246\",\"broker_port\":\"10100\",\"port_id\":\"c0f0586f-a4ee-41b5-a7f1-b4e554bbf12d\",\"namesrv_port\":\"-\"},\"192.168.1.77\":{\"advertised_ip\":\"192.168.1.77\",\"broker_port\":\"10101\",\"port_id\":\"8eb1d144-4315-402f-8498-37c9a10b630c\",\"namesrv_port\":\"-\"},\"192.168.1.159\":{\"advertised_ip\":\"192.168.1.159\",\"broker_port\":\"-\",\"port_id\":\"d35b22af-a202-4329-bda4-26e1bdf2aa8e\",\"namesrv_port\":\"8300\"},\"192.168.1.14\":{\"advertised_ip\":\"192.168.1.14\",\"broker_port\":\"10102\",\"port_id\":\"5fcb50fd-3af5-4123-a455-a67f1b926026\",\"namesrv_port\":\"-\"}}",
"storage_resource_id" : "164bdaef-2e67-4fd5-be8b-a18f91d455a2",
"storage_spec_code" : "dms.physical.storage.ultra.v2",
"service_type" : "advanced",
"storage_type" : "hec",
"extend_times" : 0,
"ipv6_enable" : false,
"support_features" :
"kafka.crossvpc.domain.enable,feature.physerver.kafka.topic.accesspolicy,rabbitmq.plugin.management,rocketmq.acl,roma_app_enable,auto_topic_switch,feature.physerver.kafka.user.manager,enable.new.authinfo,route,kafka.config.dynamic.modify.enable,kafka.new.pod.port,feature.physerver.kafka.topic.modify,message_trace_enable,features.pod.token.access,kafka.ssl.cert.modify.enable,roma.user.manage.no.support,feature.physerver.kafka.pulbic.dynamic,features.log.collection,kafka.config.static.modify.enable",
"disk_encrypted" : false,
"ces_version" : "linux,v1,v2",
"node_num" : 5,
"new_spec_billing_enable" : true,
"enable_acl" : false,
"broker_num" : 1,
"namesrv_address" : "****",
"broker_address" : "****",
"public_namesrv_address" : "****",
"public_broker_address" : "****"
}
```


Status Codes

Status Code	Description
200	Instance queried successfully.

Error Codes

See [Error Codes](#).

5.1.4 Deleting an Instance

Function

This API is used to delete an instance to release all the resources occupied by it.

URI

DELETE /v2/{project_id}/instances/{instance_id}

Table 5-12 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
instance_id	Yes	String	Instance ID.

Request Parameters

None

Response Parameters

None

Example Requests

Deleting an instance

```
DELETE https://{endpoint}/v2/{project_id}/instances/{instance_id}
```

Example Responses

None

Status Codes

Status Code	Description
204	Specified instance deleted successfully.

Error Codes

See [Error Codes](#).

5.1.5 Modifying Instance Information

Function

This API is used to modify the name and description of an instance.

URI

PUT /v2/{project_id}/instances/{instance_id}

Table 5-13 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
instance_id	Yes	String	Instance ID.

Request Parameters

Table 5-14 Request body parameters

Parameter	Mandatory	Type	Description
name	No	String	Instance name. An instance name consists of 4 to 64 characters including letters, digits, and hyphens (-) and must start with a letter.

Parameter	Mandatory	Type	Description
description	No	String	Instance description. The description can contain a maximum of 1024 characters. NOTE The backslash (\) and quotation mark (") are special characters for JSON messages. When using these characters in a parameter value, add the escape character (\) before the characters, for example, \ and \".
security_group_id	No	String	Security group ID.
enable_acl	No	Boolean	ACL.

Response Parameters

None

Example Requests

Modifying the name and description of an instance

```
PUT https://{endpoint}/v2/{project_id}/instances/{instance_id}
{
  "name" : "rocketmq001",
  "description" : "RocketMQ description"
}
```

Example Responses

None

Status Codes

Status Code	Description
204	Instance information modified successfully.

Error Codes

See [Error Codes](#).

5.1.6 Batch Deleting Instances

Function

This API is used to delete instances in batches. **Data in the instances will be deleted without any backup. Exercise caution when performing this operation.**

URI

POST /v2/{project_id}/instances/action

Table 5-15 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.

Request Parameters

Table 5-16 Request body parameters

Parameter	Mandatory	Type	Description
instances	No	Array of strings	List of instance IDs.
action	Yes	String	Operation to be performed on instances. Value: delete .
all_failure	No	String	When set to reliability , this parameter indicates all RocketMQ instances that fail to be created will be deleted.

Response Parameters

Status code: 200

Table 5-17 Response body parameters

Parameter	Type	Description
results	Array of results objects	Result of instance modification.

Table 5-18 results

Parameter	Type	Description
result	String	Operation result.
instance	String	Instance ID.

Example Requests

- Deleting instances in batches

```
POST https://{endpoint}/v2/{project_id}/instances/action
```

```
{
  "action": "delete",
  "instances": [ "54602a9d-5e22-4239-9123-77e350df4a34", "7166cdea-
dbad-4d79-9610-7163e6f8b640" ]
}
```

- Deleting all instances that fail to be created

```
POST https://{endpoint}/v2/{project_id}/instances/action
```

```
{
  "action": "delete",
  "all_failure": "reliability"
}
```

Example Responses

Status code: 200

Instances deleted.

```
{
  "results": [ {
    "result": "success",
    "instance": "019cacb7-4ff0-4d3c-9f33-f5f7b7fdc0e6"
  } ]
}
```

Status Codes

Status Code	Description
200	Instances deleted.

Error Codes

See [Error Codes](#).

5.1.7 Querying the Broker List

Function

This API is used to query the list of brokers.

URI

GET /v2/{project_id}/instances/{instance_id}/brokers

Table 5-19 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
instance_id	Yes	String	Instance ID.

Request Parameters

None

Response Parameters

Status code: 200

Table 5-20 Response body parameters

Parameter	Type	Description
brokers	Array of brokers objects	Broker list.

Table 5-21 brokers

Parameter	Type	Description
ids	Array of numbers	All broker IDs.
broker_name	String	Broker name.

Example Requests

Querying the broker list

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/brokers
```

Example Responses

Status code: 200

RocketMQ brokers queried successfully.

```
{  
  "brokers": [ {
```

```
"ids" : [ 0, 1, 2 ],
"broker_name" : "broker-0"
}]
}
```

Status Codes

Status Code	Description
200	RocketMQ brokers queried successfully.

Error Codes

See [Error Codes](#).

5.2 Consumer Group Management

5.2.1 Querying the Consumer Group List

Function

This API is used to query the consumer group list.

URI

GET /v2/{project_id}/instances/{instance_id}/groups

Table 5-22 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
instance_id	Yes	String	Instance ID.

Table 5-23 Query Parameters

Parameter	Mandatory	Type	Description
group	No	String	Consumer group name.
limit	No	Integer	Number of records to query. Default: 10

Parameter	Mandatory	Type	Description
offset	No	Integer	Offset, which is the position where the query starts. The value must be greater than or equal to 0. Default: 0

Request Parameters

None

Response Parameters

Status code: 200

Table 5-24 Response body parameters

Parameter	Type	Description
total	Number	Total number of consumer groups.
groups	Array of ConsumerGroup objects	Consumer group list.
max	Number	Maximum number of consumer groups that can be created.
remaining	Number	Number of consumer groups that can still be created.
next_offset	Number	Offset of the next page.
previous_offset	Number	Offset of the previous page.

Table 5-25 ConsumerGroup

Parameter	Type	Description
enabled	Boolean	Whether consumption is allowed.
broadcast	Boolean	Whether to broadcast.
brokers	Array of strings	List of associated brokers.
name	String	Consumer group name.

Parameter	Type	Description
retry_max_time	Number	Maximum number of retries.
from_beginning	Boolean	Whether consumption starts from the beginning.

Example Requests

Querying the consumer group list

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/groups
```

Example Responses

Status code: 200

Consumer groups queried successfully.

```
{
  "total": 1,
  "groups": [ {
    "enabled": true,
    "broadcast": false,
    "brokers": [ "broker-0" ],
    "name": "aaaaa",
    "retry_max_time": 16,
    "from_beginning": false
  } ],
  "max": 4000,
  "remaining": 3999,
  "next_offset": -1,
  "previous_offset": -1
}
```

Status Codes

Status Code	Description
200	Consumer groups queried successfully.

Error Codes

See [Error Codes](#).

5.2.2 Creating a Consumer Group or Batch Deleting Consumer Groups

Function

This API is used to create a consumer group or batch deleting consumer groups.

URI

POST /v2/{project_id}/instances/{instance_id}/groups

Table 5-26 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
instance_id	Yes	String	Instance ID.

Table 5-27 Query Parameters

Parameter	Mandatory	Type	Description
action	No	String	This parameter is used to delete topics in batches. If it is not set, a topic will be created.

Request Parameters

Table 5-28 Request body parameters

Parameter	Mandatory	Type	Description
groups	No	Array of strings	List of consumer groups to be deleted.
enabled	No	Boolean	Whether consumption is allowed.
broadcast	No	Boolean	Whether to broadcast.
brokers	No	Array of strings	List of associated brokers.
name	No	String	Consumer group name.
retry_max_time	No	Number	Maximum number of retries.
from_beginning	No	Boolean	Whether consumption starts from the beginning.

Response Parameters

Status code: 200

Table 5-29 Response body parameters

Parameter	Type	Description
job_id	String	ID of the job for deleting consumer groups.
name	String	ID of the consumer group that is successfully created.

Example Requests

- Creating a consumer group

```
POST https://{endpoint}/v2/{project_id}/instances/{instance_id}/groups
```

```
{
  "name": "consumer-group-test",
  "brokers": [ "broker-0" ],
  "from_beginning": false,
  "broadcast": false,
  "retry_max_time": 16
}
```

- Deleting consumer groups

```
POST https://{endpoint}/v2/{project_id}/instances/{instance_id}/groups?action=delete
```

```
{
  "groups": [ "consumer-group-test" ]
}
```

Example Responses

Status code: 200

A consumer group is created or multiple consumer groups are deleted in batches.

```
{
  "name": "consumer-group-test"
}
```

Status Codes

Status Code	Description
200	A consumer group is created or multiple consumer groups are deleted in batches.

Error Codes

See [Error Codes](#).

5.2.3 Batch Modifying Consumer Groups

Function

This API is used to modify consumer groups in batches.

URI

PUT /v2/{project_id}/instances/{instance_id}/groups

Table 5-30 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
instance_id	Yes	String	Instance ID.

Request Parameters

Table 5-31 Request body parameters

Parameter	Mandatory	Type	Description
groups	No	Array of ConsumerGroup objects	Consumer group list. This parameter is used when consumer groups are deleted in batches.

Table 5-32 ConsumerGroup

Parameter	Mandatory	Type	Description
enabled	No	Boolean	Whether consumption is allowed.
broadcast	No	Boolean	Whether to broadcast.
brokers	No	Array of strings	List of associated brokers.
name	No	String	Consumer group name.
retry_max_time	No	Number	Maximum number of retries.
from_beginning	No	Boolean	Whether consumption starts from the beginning.

Response Parameters

Status code: 200

Table 5-33 Response body parameters

Parameter	Type	Description
job_id	String	Job ID.

Example Requests

```
{
  "groups" : [ {
    "enabled" : true,
    "broadcast" : false,
    "brokers" : [ "broker-0" ],
    "name" : "group1",
    "retry_max_time" : 16,
    "from_beginning" : false
  } ]
}
```

Example Responses

Status code: 200

Consumer groups modified successfully.

```
{
  "job_id" : "8abfa7b27da211df017da340427b0979"
}
```

Status Codes

Status Code	Description
200	Consumer groups modified successfully.

Error Codes

See [Error Codes](#).

5.2.4 Deleting a Consumer Group

Function

This API is used to delete a specified consumer group.

URI

DELETE /v2/{project_id}/instances/{instance_id}/groups/{group}

Table 5-34 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. Minimum: 32 Maximum: 32
instance_id	Yes	String	Instance ID.
group	Yes	String	Consumer group name.

Request Parameters

None

Response Parameters

None

Example Requests

Deleting a consumer group

DELETE https://{endpoint}/v2/{project_id}/instances/{instance_id}/groups/{group}

Example Responses

None

Status Codes

Status Code	Description
204	The operation is successful.

Error Codes

See [Error Codes](#).

5.2.5 Querying a Consumer Group

Function

This API is used to query details about a specified consumer group.

URI

GET /v2/{project_id}/instances/{instance_id}/groups/{group}

Table 5-35 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
instance_id	Yes	String	Instance ID.
group	Yes	String	Consumer group name.

Request Parameters

None

Response Parameters

Status code: 200

Table 5-36 Response body parameters

Parameter	Type	Description
enabled	Boolean	Whether consumption is allowed.
broadcast	Boolean	Whether to broadcast.
brokers	Array of strings	List of associated brokers.
name	String	Consumer group name.
retry_max_time	Number	Maximum number of retries.
app_id	String	Application ID.
app_name	String	Application name.
permissions	Array of strings	Permission.

Example Requests

Querying a consumer group

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/groups/{group}
```

Example Responses

Status code: 200

Consumer group queried successfully.

```
{
  "name" : "test",
```

```

"enabled" : true,
"broadcast" : true,
"brokers" : [ "broker-0" ],
"permissions" : [ ],
"retry_max_time" : 10,
"app_id" : null,
"app_name" : null
}
    
```

Status Codes

Status Code	Description
200	Consumer group queried successfully.

Error Codes

See [Error Codes](#).

5.2.6 Modifying a Consumer Group

Function

This API is used to modify parameters of a specified consumer group.

URI

PUT /v2/{project_id}/instances/{instance_id}/groups/{group}

Table 5-37 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
instance_id	Yes	String	Instance ID.
group	Yes	String	Consumer group name.

Request Parameters

Table 5-38 Request body parameters

Parameter	Mandatory	Type	Description
enabled	No	Boolean	Whether consumption is allowed.
broadcast	No	Boolean	Whether to broadcast.

Parameter	Mandatory	Type	Description
brokers	No	Array of strings	List of associated brokers.
name	No	String	Consumer group name.
retry_max_time	No	Number	Maximum number of retries.
from_beginning	No	Boolean	Whether consumption starts from the beginning.

Response Parameters

None

Example Requests

Modifying a consumer group

```
PUT https://{endpoint}/v2/{project_id}/instances/{instance_id}/groups/{group}
```

```
{
  "name" : "consumer-group-test",
  "enabled" : true,
  "retry_max_time" : 16,
  "from_beginning" : true,
  "broadcast" : true
}
```

Example Responses

None

Status Codes

Status Code	Description
204	The operation is successful.

Error Codes

See [Error Codes](#).

5.2.7 Querying the Consumer Group List or Details

Function

This API is used to query the consumer group list or details.

URI

GET /v2/{project_id}/instances/{instance_id}/groups/{group}/topics

Table 5-39 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
instance_id	Yes	String	Instance ID.
group	Yes	String	Consumer group name.

Table 5-40 Query Parameters

Parameter	Mandatory	Type	Description
topic	No	String	Topic to be queried. If this parameter is not specified, the entire topic list is queried. If this parameter is specified, details of the topic are queried.

Request Parameters

None

Response Parameters

Status code: 200

Table 5-41 Response body parameters

Parameter	Type	Description
topics	Array of strings	Topic list.

Example Requests

Querying the consumer group list or details

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/groups/{group}/topics
```

Example Responses

Status code: 200

Consumer group list or details queried successfully.

```
{
  "topics" : [ "topic01", "topic02" ]
}
```

Status Codes

Status Code	Description
200	Consumer group list or details queried successfully.

Error Codes

See [Error Codes](#).

5.2.8 Resetting the Consumer Offset

Function

This API is used to reset the consumer offset.

URI

POST /v2/{engine}/{project_id}/instances/{instance_id}/groups/{group_id}/reset-message-offset

Table 5-42 Path Parameters

Parameter	Mandatory	Type	Description
engine	Yes	String	Engine type. Value: reliability . Default: reliability
project_id	Yes	String	Project ID.
instance_id	Yes	String	Instance ID.
group_id	Yes	String	Consumer group name.

Request Parameters

Table 5-43 Request body parameters

Parameter	Mandatory	Type	Description
topic	No	String	Topic to be reset.
timestamp	No	Number	Reset time.

Response Parameters

Status code: 200

Table 5-44 Response body parameters

Parameter	Type	Description
queues	Array of queues objects	Queues to be reset.

Table 5-45 queues

Parameter	Type	Description
broker_name	String	Broker where the queue is located.
queue_id	Integer	Queue ID.
timestamp_of_fset	Integer	Target offset.

Example Requests

Resetting the consumer offset

```
POST https://{endpoint}/v2/{engine}/{project_id}/instances/{instance_id}/groups/{group_id}/reset-message-offset
{
  "topic": "topic_01",
  "timestamp": 1662652800000
}
```

Example Responses

Status code: 200

Consumer offset reset.

```
{
  "queues": [ {
    "broker_name": "broker-1",
    "queue_id": 0,
    "timestamp_offset": 0
  } ]
}
```

Status Codes

Status Code	Description
200	Consumer offset reset.

Error Codes

See [Error Codes](#).

5.3 Topic Management

5.3.1 Creating a Topic or Deleting Topics in Batches

Function

This API is used to create a topic or delete topics in batches.

URI

POST /v2/{project_id}/instances/{instance_id}/topics

Table 5-46 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
instance_id	Yes	String	Instance ID.

Table 5-47 Query Parameters

Parameter	Mandatory	Type	Description
action	No	String	This parameter is used to delete topics in batches. If it is not set, a topic will be created.

Request Parameters

Table 5-48 Request body parameters

Parameter	Mandatory	Type	Description
name	No	String	Topic name.
brokers	No	Array of strings	Associated brokers.
queue_num	No	Number	Number of queues. Default: 3
permission	No	String	Permissions. Default: all
topics	No	Array of strings	Topic list. This parameter is used when topics are deleted in batches.

Response Parameters

Status code: 200

Table 5-49 Response body parameters

Parameter	Type	Description
id	String	Topic name.
job_id	String	Job ID for deleting topics.

Example Requests

Creating a topic

```
POST https://{endpoint}/v2/{project_id}/instances/{instance_id}/topics
```

```
{
  "name" : "topic-test",
  "brokers" : [ "broker-0" ],
  "queue_num" : 3,
  "permission" : "all"
}
```

Example Responses

Status code: 200

A topic is created or multiple topics are deleted in batches.

```
{  
  "id" : "topic-test"  
}
```

Status Codes

Status Code	Description
200	A topic is created or multiple topics are deleted in batches.

Error Codes

See [Error Codes](#).

5.3.2 Deleting a Topic

Function

This API is used to delete a specified topic.

URI

DELETE /v2/{project_id}/instances/{instance_id}/topics/{topic}

Table 5-50 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
instance_id	Yes	String	Instance ID.
topic	Yes	String	Topic name.

Request Parameters

None

Response Parameters

None

Example Requests

Deleting a topic

DELETE https://{endpoint}/v2/{project_id}/instances/{instance_id}/topics/{topic}

Example Responses

None

Status Codes

Status Code	Description
204	The operation is successful.

Error Codes

See [Error Codes](#).

5.3.3 Querying a Topic

Function

This API is used to query a topic.

URI

GET /v2/{project_id}/instances/{instance_id}/topics/{topic}

Table 5-51 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
instance_id	Yes	String	Instance ID.
topic	Yes	String	Topic name.

Request Parameters

None

Response Parameters

Status code: 200

Table 5-52 Response body parameters

Parameter	Type	Description
total_read_queue_num	Number	Total number of read queues.

Parameter	Type	Description
total_write_queue_num	Number	Total number of write queues.
permission	String	Permissions.
brokers	Array of brokers objects	Associated brokers.

Table 5-53 brokers

Parameter	Type	Description
broker_name	String	Broker name.
read_queue_num	Number	Number of read queues.
write_queue_num	Number	Number of write queues.

Example Requests

Querying a topic

GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/topics/{topic}

Example Responses

Status code: 200

Topic queried successfully.

```
{
  "total_read_queue_num" : 3,
  "total_write_queue_num" : 3,
  "permission" : "all",
  "brokers" : [ {
    "broker_name" : "broker-0",
    "read_queue_num" : 3,
    "write_queue_num" : 3
  } ]
}
```

Status Codes

Status Code	Description
200	Topic queried successfully.

Error Codes

See [Error Codes](#).

5.3.4 Modifying a Topic

Function

This API is used to modify a topic.

URI

PUT /v2/{project_id}/instances/{instance_id}/topics/{topic}

Table 5-54 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
instance_id	Yes	String	Instance ID.
topic	Yes	String	Topic name.

Request Parameters

Table 5-55 Request body parameters

Parameter	Mandatory	Type	Description
read_queue_num	No	Number	Total number of read queues.
write_queue_num	No	Number	Total number of write queues.
permission	No	String	Permissions.

Response Parameters

None

Example Requests

Modifying a topic

```
PUT https://{endpoint}/v2/{project_id}/instances/{instance_id}/topics/{topic}
{
  "read_queue_num" : 3,
  "write_queue_num" : 3,
  "permission" : "all"
}
```

Example Responses

None

Status Codes

Status Code	Description
204	The operation is successful.

Error Codes

See [Error Codes](#).

5.3.5 Querying the Topic Consumer Group List

Function

This API is used to query the topic consumer group list.

URI

GET /v2/{project_id}/instances/{instance_id}/topics/{topic}/groups

Table 5-56 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
instance_id	Yes	String	Instance ID.
topic	Yes	String	Topic name.

Request Parameters

None

Response Parameters

Status code: 200

Table 5-57 Response body parameters

Parameter	Type	Description
groups	Array of strings	Consumer group list.

Example Requests

Query the topic consumer group list

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/topics/{topic}/groups
```

Example Responses

Status code: 200

Topic consumer group list queried successfully.

```
{
  "groups" : [ "CID_JODIE_1", "test_consumer" ]
}
```

Status Codes

Status Code	Description
200	Topic consumer group list queried successfully.

Error Codes

See [Error Codes](#).

5.3.6 Querying the Number of Messages in a Topic

Function

This API is used to query the number of messages in a topic.

URI

```
GET /v2/{project_id}/instances/{instance_id}/topics/{topic}/status
```

Table 5-58 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
instance_id	Yes	String	Instance ID.
topic	Yes	String	Topic name.

Request Parameters

None

Response Parameters

Status code: 200

Table 5-59 Response body parameters

Parameter	Type	Description
brokers	Array of brokers objects	Brokers.

Table 5-60 brokers

Parameter	Type	Description
queues	Array of queues objects	Queue list.
broker_name	String	Broker name.

Table 5-61 queues

Parameter	Type	Description
id	Integer	Queue ID.
min_offset	Integer	Minimum offset.
max_offset	Integer	Maximum offset.
last_message_time	Integer	Time of the last message.

Example Requests

Querying the number of messages in a topic

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/topics/{topic}/status
```

Example Responses

Status code: 200

The number of messages in the topic queried successfully.

```
{
  "brokers": {
    "queues": [ {
      "id": 0,
      "min_offset": 0,
```

```

    "max_offset" : 2,
    "last_message_time" : 1662689877152
  }],
  "broker_name" : "broker-0"
}
}

```

Status Codes

Status Code	Description
200	The number of messages in the topic queried successfully.

Error Codes

See [Error Codes](#).

5.4 Message Management

5.4.1 Querying Messages

Function

This API is used to query messages.

URI

GET /v2/{engine}/{project_id}/instances/{instance_id}/messages

Table 5-62 Path Parameters

Parameter	Mandatory	Type	Description
engine	Yes	String	Message engine. Default: reliability
project_id	Yes	String	Project ID.
instance_id	Yes	String	Instance ID.

Table 5-63 Query Parameters

Parameter	Mandatory	Type	Description
topic	Yes	String	Topic name.
limit	No	String	Number of records to query.

Parameter	Mandatory	Type	Description
offset	No	String	Offset, which is the position where the query starts. The value must be greater than or equal to 0.
start_time	No	String	Start time. This parameter is mandatory when msg_id is not used for query.
end_time	No	String	End time. This parameter is mandatory when msg_id is not used for query.
msg_id	No	String	Message ID.

Request Parameters

None

Response Parameters

Status code: 200

Table 5-64 Response body parameters

Parameter	Type	Description
messages	Array of Message objects	Message list.
total	Number	Total number of messages.

Table 5-65 Message

Parameter	Type	Description
msg_id	String	Message ID.
instance_id	String	Instance ID.
topic	String	Topic name.
store_timestamp	Number	Time when the message is stored.
born_timestamp	Number	Time when the message is generated.

Parameter	Type	Description
reconsume_times	String	Number of retry times.
body	String	Message body.
body_crc	Number	Message body checksum.
store_size	Number	Storage size.
property_list	Array of property_list objects	Message attribute list.
born_host	String	IP address of the host that generates the message.
store_host	String	IP address of the host that stores the message.
queue_id	String	Queue ID.
queue_offset	String	Offset in the queue.

Table 5-66 property_list

Parameter	Type	Description
name	String	Attribute name.
value	String	Attribute value.

Example Requests

Querying messages

```
GET https://{endpoint}/v2/{engine}/{project_id}/instances/{instance_id}/messages
```

Example Responses

Status code: 200

Messages queried successfully.

```
{
  "messages": [ {
    "msg_id": "C0A807C9000027740000000000000030",
    "instance_id": "11c45539-xxxx-xxxx-xxxx-812c41f61f30",
    "topic": "topic-test",
    "store_timestamp": 1648888166319,
    "born_timestamp": 1648888166275,
    "reconsume_times": 0,
    "body": "xxxx",
    "body_crc": 1932557065,
    "store_size": 175,
    "property_list": [ {
      "name": "KEYS",
```



```

"value" : ""
}, {
  "name" : "UNIQ_KEY",
  "value" : "7F00000123DC6E0BE85808B037820000"
}, {
  "name" : "CLUSTER",
  "value" : "DmsCluster"
}, {
  "name" : "TAGS",
  "value" : ""
}
],
"born_host" : "192.168.0.66:50098",
"store_host" : "192.168.7.201:10100",
"queue_id" : 2,
"queue_offset" : 0
}],
"total" : 1
}

```

Status Codes

Status Code	Description
200	Messages queried successfully.

Error Codes

See [Error Codes](#).

5.4.2 Querying the Message Trace

Function

This API is used to query the message trace.

URI

GET /v2/{engine}/{project_id}/instances/{instance_id}/trace

Table 5-67 Path Parameters

Parameter	Mandatory	Type	Description
engine	Yes	String	Message engine. Default: reliability
project_id	Yes	String	Project ID.
instance_id	Yes	String	Instance ID.

Table 5-68 Query Parameters

Parameter	Mandatory	Type	Description
msg_id	No	String	Message ID.

Request Parameters

None

Response Parameters

Status code: 200

Table 5-69 Response body parameters

Parameter	Type	Description
trace	Array of trace objects	Message trace list.

Table 5-70 trace

Parameter	Type	Description
success	Boolean	Whether the request is successful.
trace_type	String	Trace type.
timestamp	Number	Time.
group_name	String	Producer group or consumer group.
cost_time	Number	Time spent.
request_id	String	Request ID.
consume_status	Number	Consumption status.
topic	String	Topic name.
msg_id	String	Message ID.
offset_msg_id	String	Offset message ID.
tags	String	Message tag.
keys	String	Message keys.
store_host	String	IP address of the host that stores the message.
client_host	String	IP address of the host that generates the message.

Parameter	Type	Description
retry_times	String	Number of retry times.
body_length	Number	Message body length.
msg_type	String	Message type.
transaction_state	String	Transaction status.
transaction_id	String	Transaction ID.
from_transaction_check	Boolean	Whether the response is a transaction check response.

Example Requests

Querying the message trace

```
GET https://{endpoint}/v2/{engine}/{project_id}/instances/{instance_id}/trace
```

Example Responses

Status code: 200

The query is successful.

```
[ {
  "success" : true,
  "trace_type" : "Pub",
  "timestamp" : 1634822858013,
  "group_name" : "ProducerGroupName",
  "cost_time" : 47,
  "request_id" : "644F0069C829287CBBF26B9A54390000",
  "consume_status" : 0,
  "topic" : "aaaaa",
  "msg_id" : "7F000001561018B4AC26B9A0D1D0004",
  "offset_msg_id" : "COA801170000277400000000000000BE12",
  "tags" : "TagA",
  "keys" : "OrderID188",
  "store_host" : "192.168.0.1:10101",
  "client_host" : "127.0.0.1",
  "retry_times" : 0,
  "body_length" : 11,
  "msg_type" : "Normal_Msg",
  "transaction_state" : null,
  "transaction_id" : null,
  "from_transaction_check" : false
} ]
```

Status Codes

Status Code	Description
200	The query is successful.

Error Codes

See [Error Codes](#).

5.4.3 Exporting Dead Letter Messages

Function

This API is used to export dead letter messages.

URI

POST /v2/{project_id}/instances/{instance_id}/messages/export

Table 5-71 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
instance_id	Yes	String	Instance ID.

Request Parameters

Table 5-72 Request body parameters

Parameter	Mandatory	Type	Description
topic	No	String	Topic name.
msg_id_list	No	Array of strings	Message ID list.
uniq_key_list	No	Array of strings	Unique key list.

Response Parameters

Status code: 200

Table 5-73 Response body parameters

Parameter	Type	Description
[items]	Array of Message objects	Message.

Table 5-74 Message

Parameter	Type	Description
msg_id	String	Message ID.
instance_id	String	Instance ID.
topic	String	Topic name.
store_timestamp	Number	Time when the message is stored.
born_timestamp	Number	Time when the message is generated.
reconsume_times	String	Number of retry times.
body	String	Message body.
body_crc	Number	Message body checksum.
store_size	Number	Storage size.
property_list	Array of property_list objects	Message attribute list.
born_host	String	IP address of the host that generates the message.
store_host	String	IP address of the host that stores the message.
queue_id	String	Queue ID.
queue_offset	String	Offset in the queue.

Table 5-75 property_list

Parameter	Type	Description
name	String	Attribute name.
value	String	Attribute value.

Example Requests

Exporting dead letter messages

```
POST https://{endpoint}/v2/{project_id}/instances/{instance_id}/messages/export
{
  "topic": "%DLQ%group1",
  "msg_id_list": [ "C0A8011700002774000000000013B19D", "C0A8011700002774000000000013B30F" ],
  "uniq_key_list": [ "7F000001001C18B4AAC26B8AED170010", "7F000001001C18B4AAC26B8AEE030015" ]
}
```

Example Responses

Status code: 200

Dead letter messages exported successfully.

```
{
  "topic": "topic_01",
  "body": "Hello world",
  "property_list": {
    "name": "ORIGIN_MESSAGE_ID",
    "value": "C0A8005B00002775000000000000EBAE"
  },
  "msg_id": "C0A8005B00002775000000000000133A2",
  "instance_id": "56055acb-3c3b-4481-aeab-10464086c2b4",
  "store_timestamp": 1662690563020,
  "born_timestamp": 1662690324415,
  "reconsume_times": 5,
  "body_crc": 198614610,
  "store_size": 317,
  "born_host": "10.58.233.224:63529",
  "store_host": "192.168.0.91:10101",
  "queue_id": 0,
  "queue_offset": 0
}
```

Status Codes

Status Code	Description
200	Dead letter messages exported successfully.

Error Codes

See [Error Codes](#).

5.5 User Management

5.5.1 Creating a User

Function

This API is used to create a user.

URI

POST /v2/{project_id}/instances/{instance_id}/users

Table 5-76 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.

Parameter	Mandatory	Type	Description
instance_id	Yes	String	Instance ID.

Request Parameters

Table 5-77 Request body parameters

Parameter	Mandatory	Type	Description
access_key	No	String	Username.
secret_key	No	String	Secret key.
white_remote_address	No	String	IP address whitelist.
admin	No	Boolean	Whether the user is an administrator.
default_topic_perm	No	String	Default topic permissions.
default_group_perm	No	String	Default consumer group permissions.
topic_perms	No	Array of topic_perms objects	Special topic permissions.
group_perms	No	Array of group_perms objects	Special consumer group permissions.

Table 5-78 topic_perms

Parameter	Mandatory	Type	Description
name	No	String	Topic name.
perm	No	String	Permissions.

Table 5-79 group_perms

Parameter	Mandatory	Type	Description
name	No	String	Consumer group name.
perm	No	String	Permissions.

Response Parameters

Status code: 200

Table 5-80 Response body parameters

Parameter	Type	Description
access_key	String	Username.
secret_key	String	Secret key.
white_remote_address	String	IP address whitelist.
admin	Boolean	Whether the user is an administrator.
default_topic_perm	String	Default topic permissions.
default_group_perm	String	Default consumer group permissions.
topic_perms	Array of topic_perms objects	Special topic permissions.
group_perms	Array of group_perms objects	Special consumer group permissions.

Table 5-81 topic_perms

Parameter	Type	Description
name	String	Topic name.
perm	String	Permissions.

Table 5-82 group_perms

Parameter	Type	Description
name	String	Consumer group name.
perm	String	Permissions.

Example Requests

```
{
  "access_key" : "user_name",
  "secret_key" : "O6bE3KjttD6Nc6ZtZP3axKLUHtOUaMnL",
  "white_remote_address" : ""
}
```



```
"admin" : false,  
"default_topic_perm" : "DENY",  
"default_group_perm" : "DENY",  
"topic_perms" : [ {  
  "name" : "topic1",  
  "perm" : "PUB|SUB"  
} ],  
"group_perms" : [ {  
  "name" : "group1",  
  "perm" : "PUB|SUB"  
} ]  
}
```

Example Responses

Status code: 200

The creation is successful.

```
{  
  "users" : {  
    "access_key" : "test_01",  
    "admin" : false,  
    "default_group_perm" : "DENY",  
    "default_topic_perm" : "SUB",  
    "group_perms" : [ ],  
    "secret_key" : "x1234567@",  
    "topic_perms" : [ ],  
    "white_remote_address" : ""  
  },  
  "total" : 1  
}
```

Status Codes

Status Code	Description
200	The creation is successful.

Error Codes

See [Error Codes](#).

5.5.2 Querying the User List

Function

This API is used to query the user list.

URI

GET /v2/{project_id}/instances/{instance_id}/users

Table 5-83 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
instance_id	Yes	String	Instance ID.

Table 5-84 Query Parameters

Parameter	Mandatory	Type	Description
offset	No	String	Offset, which is the position where the query starts. The value must be greater than or equal to 0.
limit	No	String	Number of records to query.

Request Parameters

None

Response Parameters

Status code: 200

Table 5-85 Response body parameters

Parameter	Type	Description
users	Array of User objects	User list.
total	Number	Total number of users.

Table 5-86 User

Parameter	Type	Description
access_key	String	Username.
secret_key	String	Secret key.
white_remote_address	String	IP address whitelist.
admin	Boolean	Whether the user is an administrator.

Parameter	Type	Description
default_topic_perm	String	Default topic permissions.
default_group_perm	String	Default consumer group permissions.
topic_perms	Array of topic_perms objects	Special topic permissions.
group_perms	Array of group_perms objects	Special consumer group permissions.

Table 5-87 topic_perms

Parameter	Type	Description
name	String	Topic name.
perm	String	Permissions.

Table 5-88 group_perms

Parameter	Type	Description
name	String	Consumer group name.
perm	String	Permissions.

Example Requests

Querying the user list

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/users?offset=0&limit=10
```

Example Responses

Status code: 200

Query result.

```
{
  "users": [ {
    "access_key": "test_01",
    "admin": false,
    "default_group_perm": "DENY",
    "default_topic_perm": "SUB",
    "group_perms": [ ],
    "secret_key": "x1234567@",
    "topic_perms": [ ],
  }
]
```

```
"white_remote_address" : ""
  } ],
  "total" : 1
}
```

Status Codes

Status Code	Description
200	Query result.

Error Codes

See [Error Codes](#).

5.5.3 Modifying User Parameters

Function

This API is used to modify user parameters.

URI

PUT /v2/{project_id}/instances/{instance_id}/users/{user_name}

Table 5-89 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
instance_id	Yes	String	Instance ID.
user_name	Yes	String	Username.

Request Parameters

Table 5-90 Request body parameters

Parameter	Mandatory	Type	Description
access_key	No	String	Username.
secret_key	No	String	Secret key.
white_remote_address	No	String	IP address whitelist.
admin	No	Boolean	Whether the user is an administrator.

Parameter	Mandatory	Type	Description
default_topic_perm	No	String	Default topic permissions.
default_group_perm	No	String	Default consumer group permissions.
topic_perms	No	Array of topic_perms objects	Special topic permissions.
group_perms	No	Array of group_perms objects	Special consumer group permissions.

Table 5-91 topic_perms

Parameter	Mandatory	Type	Description
name	No	String	Topic name.
perm	No	String	Permissions.

Table 5-92 group_perms

Parameter	Mandatory	Type	Description
name	No	String	Consumer group name.
perm	No	String	Permissions.

Response Parameters

Status code: 200

Table 5-93 Response body parameters

Parameter	Type	Description
access_key	String	Username.
secret_key	String	Secret key.
white_remote_address	String	IP address whitelist.
admin	Boolean	Whether the user is an administrator.

Parameter	Type	Description
default_topic_perm	String	Default topic permissions.
default_group_perm	String	Default consumer group permissions.
topic_perms	Array of topic_perms objects	Special topic permissions.
group_perms	Array of group_perms objects	Special consumer group permissions.

Table 5-94 topic_perms

Parameter	Type	Description
name	String	Topic name.
perm	String	Permissions.

Table 5-95 group_perms

Parameter	Type	Description
name	String	Consumer group name.
perm	String	Permissions.

Example Requests

```
{
  "access_key" : "user_name",
  "secret_key" : "O6bE3KjttD6Nc6ZtZP3axKLUHtOUaMnL",
  "white_remote_address" : "",
  "admin" : false,
  "default_topic_perm" : "DENY",
  "default_group_perm" : "DENY",
  "topic_perms" : [ {
    "name" : "topic1",
    "perm" : "PUB|SUB"
  } ],
  "group_perms" : [ {
    "name" : "group1",
    "perm" : "PUB|SUB"
  } ]
}
```

Example Responses

Status code: 200

The modification is successful.

```
{
  "access_key" : "test_01",
  "admin" : false,
  "default_group_perm" : "DENY",
  "default_topic_perm" : "SUB",
  "group_perms" : [ ],
  "secret_key" : "x1234567@",
  "topic_perms" : [ ],
  "white_remote_address" : ""
}
```

Status Codes

Status Code	Description
200	The modification is successful.

Error Codes

See [Error Codes](#).

5.5.4 Querying User Details

Function

This API is used to query user details.

URI

GET /v2/{project_id}/instances/{instance_id}/users/{user_name}

Table 5-96 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
instance_id	Yes	String	Instance ID.
user_name	Yes	String	Username.

Request Parameters

None

Response Parameters

Status code: 200

Table 5-97 Response body parameters

Parameter	Type	Description
access_key	String	Username.
secret_key	String	Secret key.
white_remote_address	String	IP address whitelist.
admin	Boolean	Whether the user is an administrator.
default_topic_perm	String	Default topic permissions.
default_group_perm	String	Default consumer group permissions.
topic_perms	Array of topic_perms objects	Special topic permissions.
group_perms	Array of group_perms objects	Special consumer group permissions.

Table 5-98 topic_perms

Parameter	Type	Description
name	String	Topic name.
perm	String	Permissions.

Table 5-99 group_perms

Parameter	Type	Description
name	String	Consumer group name.
perm	String	Permissions.

Example Requests

Querying user details

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/users/{user_name}?offset=0&limit=10
```

Example Responses

Status code: 200

User details queried successfully.

```
{
  "access_key" : "test_01",
  "admin" : false,
  "default_group_perm" : "DENY",
  "default_topic_perm" : "SUB",
  "group_perms" : [ ],
  "secret_key" : "x1234567@",
  "topic_perms" : [ ],
  "white_remote_address" : ""
}
```

Status Codes

Status Code	Description
200	User details queried successfully.

Error Codes

See [Error Codes](#).

5.5.5 Deleting a User

Function

This API is used to delete a user.

URI

DELETE /v2/{project_id}/instances/{instance_id}/users/{user_name}

Table 5-100 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
instance_id	Yes	String	Instance ID.
user_name	Yes	String	Username.

Request Parameters

None

Response Parameters

Status code: 200

Table 5-101 Response body parameters

Parameter	Type	Description
message	String	Information.

Example Requests

Deleting a consumer group

DELETE https://{endpoint}/v2/{project_id}/instances/{instance_id}/users/{user_name}

Example Responses

None

Status Codes

Status Code	Description
200	The deletion is successful.

Error Codes

See [Error Codes](#).

5.5.6 Querying Users Granted Permissions for a Topic

Function

This API is used to query the list of users that have been granted permissions for a topic.

URI

GET /v2/{project_id}/instances/{instance_id}/topics/{topic}/accesspolicy

Table 5-102 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
instance_id	Yes	String	Instance ID.
topic	Yes	String	Topic name.

Table 5-103 Query Parameters

Parameter	Mandatory	Type	Description
offset	No	String	Offset, which is the position where the query starts. The value must be greater than or equal to 0.
limit	No	String	Number of records to query.

Request Parameters

None

Response Parameters

Status code: 200

Table 5-104 Response body parameters

Parameter	Type	Description
policies	Array of policies objects	User list.
total	Number	Total number of users.
name	String	Name of the topic or consumer group.

Table 5-105 policies

Parameter	Type	Description
access_key	String	Username.
secret_key	String	Key.
white_remote_address	String	IP address whitelist.
admin	Boolean	Whether the user is an administrator.
perm	String	Permissions.

Example Requests

Querying the list of users granted permissions for a topic

GET https://{{endpoint}}/v2/{{project_id}}/instances/{{instance_id}}/topics/{{topic}}/accesspolicy

Example Responses

Status code: 200

List of users granted permissions for a topic queried successfully.

```
{
  "policies" : [],
  "total" : 0,
  "name" : "test"
}
```

Status Codes

Status Code	Description
200	List of users granted permissions for a topic queried successfully.

Error Codes

See [Error Codes](#).

5.5.7 Querying Users Granted Permissions for a Consumer Group

Function

This API is used to query the list of users that have been granted permissions for a consumer group.

URI

GET /v2/{engine}/{project_id}/instances/{instance_id}/groups/{group_id}/accesspolicy

Table 5-106 Path Parameters

Parameter	Mandatory	Type	Description
engine	Yes	String	Message engine. Default: reliability
project_id	Yes	String	Project ID.
instance_id	Yes	String	Instance ID.
group_id	Yes	String	Consumer group.

Table 5-107 Query Parameters

Parameter	Mandatory	Type	Description
offset	No	String	Offset, which is the position where the query starts. The value must be greater than or equal to 0.
limit	No	String	Number of records to query.

Request Parameters

None

Response Parameters

Status code: 200

Table 5-108 Response body parameters

Parameter	Type	Description
policies	Array of policies objects	User list.
total	Number	Total number of users.
name	String	Name of the topic or consumer group.

Table 5-109 policies

Parameter	Type	Description
access_key	String	Username.
secret_key	String	Key.
white_remote_address	String	IP address whitelist.
admin	Boolean	Whether the user is an administrator.
perm	String	Permissions.

Example Requests

Querying the list of users granted permissions for a consumer group

```
GET https://{endpoint}/v2/{engine}/{project_id}/instances/{instance_id}/groups/{group_id}/accesspolicy
```

Example Responses

Status code: 200

List of users granted permissions for a consumer group queried successfully.

```
{
  "policies" : {
    "access_key" : "test_01",
    "secret_key" : "AAAAAgAAAAAAAAAHAAAAQAAAdqH2a/vCXyEfkugiYc6nmu+g6CQPoaXhajh
+3pHjscvgAAAAAAAAAAAAAAAAEBSNBQuVKEq+9yvR6GbKcW4=",
    "white_remote_address" : "",
    "admin" : false,
    "perm" : ""
  },
  "total" : 1,
  "name" : "test"
}
```

Status Codes

Status Code	Description
200	List of users granted permissions for a consumer group queried successfully.

Error Codes

See [Error Codes](#).

5.6 Other APIs

5.6.1 Querying AZ Information

Function

This API is used to query the AZ ID for creating an instance.

URI

GET /v2/available-zones

Request Parameters

None

Response Parameters

Status code: 200

Table 5-110 Response body parameters

Parameter	Type	Description
region_id	String	Region ID.
available_zones	Array of available_zones objects	Array of AZs.

Table 5-111 available_zones

Parameter	Type	Description
soldOut	Boolean	Whether resources are sold out.
id	String	AZ ID.
code	String	AZ code.
name	String	AZ name.
port	String	AZ port.
resource_availability	String	Whether the AZ has available resources.
default_az	Boolean	Whether the AZ is the default AZ.
remain_time	Long	Remaining time.
ipv6_enable	Boolean	Whether IPv6 is supported.

Example Requests

```
GET https://{endpoint}/v2/available-zones?engine=reliability
```

Example Responses

Status code: 200

AZ information queried successfully.

```
{
  "region_id": "xxx",
  "available_zones": [ {
    "soldOut": false,
    "id": "8c90c2a4e2594c0782faa6b205afeca7",
    "code": "xxx",
    "name": "AZ 1",
    "port": "8002",
    "resource_availability": "true",
    "default_az": false,
    "remain_time": 9223372036854776000,
    "ipv6_enable": false
  }, {
    "soldOut": false,
    "id": "d539378ec1314c85b76fefa3f7071458",
```

```
"code" : "xxx",
"name" : "AZ 2",
"port" : "8003",
"resource_availability" : "true",
"default_az" : false,
"remain_time" : 9223372036854776000,
"ipv6_enable" : false
}, {
"soldOut" : false,
"id" : "9f1c5806706d4c1fb0eb72f0a9b18c77",
"code" : "xxx",
"name" : "AZ 3",
"port" : "443",
"resource_availability" : "true",
"default_az" : true,
"remain_time" : 9223372036854776000,
"ipv6_enable" : false
} ]
}
```

Status Codes

Status Code	Description
200	AZ information queried successfully.

Error Codes

See [Error Codes](#).

6 Permissions Policies and Supported Actions

This chapter describes fine-grained permissions management for your DMS for RocketMQ instances. If your cloud account does not require individual IAM users, you can skip 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 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 cloud resources based on the permissions.

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.

NOTE

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

An account has all of the permissions required to call all APIs, but IAM users must have the required permissions specifically assigned. The permissions required for calling an API are determined by the actions supported by the API. Only users who have been granted permissions allowing the actions can call the API successfully. For example, if an IAM user creates a RocketMQ instance using an API, the user must have been granted permissions that allow the **dms:instance:create** action.

Supported Actions

DMS for RocketMQ provides system-defined policies that can be directly used in IAM. 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: Statements in a policy that allow or deny certain operations.
- APIs: REST APIs that can be called by a user who has been granted specific permissions.

- Actions: Added to a custom policy to control permissions for specific operations.
- IAM projects or enterprise projects: A custom policy can be applied to IAM projects or enterprise projects or both. 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.

DMS for RocketMQ supports the following actions that can be defined in custom policies. Permissions must be obtained before calling APIs. For details on how to obtain permissions, visit the Identity and Access Management help center.

Table 6-1 DMS for RocketMQ actions

Permission	API	Action	IAM Project	Enterprise Project
Creating a Pay-per-Use Instance	POST /v2/{project_id}/instances	dms:instance:create	√	√
Querying the Instance List	GET /v2/{project_id}/instances	dms:instance:list	√	√
Querying an Instance	GET /v2/{project_id}/instances/{instance_id}	dms:instance:get	√	√
Deleting an Instance	DELETE /v2/{project_id}/instances/{instance_id}	dms:instance:delete	√	√
Modifying Instance Information	PUT /v2/{project_id}/instances/{instance_id}	dms:instance:modify	√	√
Batch Deleting Instances	POST /v2/{project_id}/instances/action	dms:instance:delete	√	√
Modifying a Topic	PUT /v2/{project_id}/instances/{instance_id}/topics/{topic}	dms:instance:modify	√	√
Querying the Topic Consumer Group List	GET /v2/{project_id}/instances/{instance_id}/topics/{topic}/groups	dms:instance:get	√	√

Permission	API	Action	IAM Project	Enterprise Project
Querying the Number of Messages in a Topic	GET /v2/{project_id}/instances/{instance_id}/topics/{topic}/status	dms:instance: get	√	√
Querying Messages	GET /v2/{engine}/{project_id}/instances/{instance_id}/messages	dms:instance: get	√	√
Creating a User	POST /v2/{project_id}/instances/{instance_id}/users	dms:instance: modify	√	√
Querying the User List	GET /v2/{project_id}/instances/{instance_id}/users	dms:instance: get	√	√
Querying User Details	GET /v2/{project_id}/instances/{instance_id}/users/{user_name}	dms:instance: get	√	√

7 Appendix

7.1 Status Code

[Table 7-1](#) lists status codes.

Table 7-1 Status codes

Status Code	Name	Description
100	Continue	The server has received the initial part of the request and the client should continue to send the remaining part.
101	Switching Protocols	The requester has asked the server to switch protocols and the server has agreed to do so. The target protocol must be more advanced than the source protocol. For example, the current HTTP protocol is switched to a later version of HTTP.
200	OK	Request sent successfully.
201	Created	The request has been fulfilled, resulting in the creation of a new resource.
202	Accepted	The request has been accepted for processing, but the processing has not been completed.
203	Non-Authoritative Information	The request has been fulfilled.
204	NoContent	The server has successfully processed the request, but is not returning any response body. The status code is returned in response to an HTTP OPTIONS request.

Status Code	Name	Description
205	Reset Content	The server has fulfilled the request, but the requester is required to reset the content.
206	Partial Content	The server has successfully processed a part of the GET request.
300	Multiple Choices	There are multiple options for the requested resource. For example, this code could be used to present a list of resource characteristics and addresses from which the client such as a browser may choose.
301	Moved Permanently	This and all future requests have been permanently moved to the given URI indicated in this response.
302	Found	The requested resource was temporarily moved.
303	See Other	The response to the request can be found under another URI using a GET or POST method.
304	Not Modified	The requested resource has not been modified. When the server returns this status code, it does not return any resources.
305	Use Proxy	The requested resource is available only through a proxy.
306	Unused	This HTTP status code is no longer used.
400	BadRequest	Invalid request. The client should modify the request instead of re-initiating it.
401	Unauthorized	The authorization information provided by the client is incorrect or invalid.
402	Payment Required	Reserved for future use.
403	Forbidden	The server has received the request and understood it, but the server is refusing to respond to it. The client should modify the request instead of re-initiating it.
404	NotFound	The requested resource cannot be found. The client should modify the request instead of re-initiating it.

Status Code	Name	Description
405	MethodNotAllowed	A request method is not supported for the requested resource. The client should modify the request instead of re-initiating it.
406	Not Acceptable	The server cannot fulfill the request based on the content characteristics of the request.
407	Proxy Authentication Required	This code is similar to 401, but indicates that the client must first authenticate itself with the proxy.
408	Request Time-out	The server timed out when waiting for the request. The client may re-initiate the request without any modification at any time.
409	Conflict	The request cannot be processed due to a conflict, such as an edit conflict between multiple simultaneous updates or the resource that the client attempts to create already exists.
410	Gone	The requested resource has been deleted permanently and will not be available again.
411	Length Required	The server refused to process the request because the request does not specify the length of its content.
412	Precondition Failed	The server does not meet one of the preconditions that the requester puts on the request.
413	Request Entity Too Large	The server refuses to process a request because the request is too large. The server may close the connection to prevent the client from continuing the request. If the server cannot process the request temporarily, the response will contain a Retry-After field.
414	Request-URI Too Large	The URI provided was too long for the server to process.
415	Unsupported Media Type	The server does not support the media type in the request.
416	Requested range not satisfiable	The requested range is invalid.
417	Expectation Failed	The server fails to meet the requirements of the Expect request-header field.

Status Code	Name	Description
422	UnprocessableEntity	The request is well-formed but is unable to be processed due to semantic errors.
429	TooManyRequests	The client has sent more requests than its rate limit is allowed within a given amount of time, or the server has received more requests than it is able to process within a given amount of time. In this case, the client should re-initiate requests after the time specified in the Retry-After header of the response expires.
500	InternalServerError	The server is able to receive the request but it could not understand the request.
501	Not Implemented	The server does not support the requested function.
502	Bad Gateway	The server was acting as a gateway or proxy and received an invalid request from a remote server.
503	ServiceUnavailable	The requested service is invalid. The client should modify the request instead of re-initiating it.
504	ServerTimeout	The request cannot be fulfilled within a given time. The response will reach the client only if the request carries the timeout parameter.
505	HTTP Version not supported	The server does not support the HTTP protocol version used in the request.

7.2 Error Codes

Status Code	Error Codes	Error Message	Description	Solution
400	DMS.00400002	The project ID format is invalid.	Invalid project ID.	Check the project ID format.
400	DMS.00400004	The request body is empty.	The request body is empty.	Check the request body.

Status Code	Error Codes	Error Message	Description	Solution
400	DMS.00400005	The message body is not in JSON format or contains invalid characters.	Check the project ID format.	Check the message body format.
400	DMS.00400007	Unsupported type.	Unsupported type.	Check the type.
400	DMS.00400008	Unsupported version.	Unsupported version.	Check the version.
400	DMS.00400009	Invalid product_id.	Invalid product_id in the request.	Check the product_id parameter.
400	DMS.00400010	Invalid instance name. The name must be 4 to 64 characters long. Only letters, digits, underscores (_), and hyphens (-) are allowed.	Invalid instance name. The name must be 4 to 64 characters long. Only letters, digits, underscores (_), and hyphens (-) are allowed.	Check the instance name.
400	DMS.00400011	The instance description can contain a maximum of 1024 characters.	The instance description can contain a maximum of 1024 characters.	Check the instance description.

Status Code	Error Codes	Error Message	Description	Solution
400	DMS.00400012	The password does not meet the complexity requirements. An instance password must: Be a string consisting of 8 to 32 characters. Contain at least two of the following character types: Lowercase letters Uppercase letters Digits Special characters `~!@#%&*()-_+=\ [{]}:;',<.>/?	The password does not meet the complexity requirements. A password: - Can contain 8 to 32 characters. - Must contain at least three of the following character types: letters, digits, and special characters `~!@#%&*()-_+=\ [{]}:;',<.>/? - Cannot be a weak password.	Check whether the password meets the requirements.
400	DMS.00400013	vpc_id in the request is empty.	Request parameter vpc_id is empty.	Check the vpc_id parameter.
400	DMS.00400014	security_group_id in the request is empty.	Request parameter security_group_id is empty.	Check the security_group_id parameter.
400	DMS.00400015	Invalid username. A username must be 4 to 64 characters long and consist of only letters, digits, and hyphens (-).	Invalid username. A username must be 4 to 64 characters long and consist of only letters, digits, and hyphens (-).	Check the username.

Status Code	Error Codes	Error Message	Description	Solution
400	DMS.00400016	subnet_id in the request is empty.	Request parameter subnet_id is empty.	Check the subnet_id parameter.
400	DMS.00400017	This DMS instance job task is still running.	A background task associated with this instance is running.	Try again later.
400	DMS.00400018	This subnet must exist in the VPC.	The subnet must exist in the VPC.	Check the subnet.
400	DMS.00400019	The password does not meet the complexity requirements.	The password does not meet the complexity requirements.	Check whether the password meets the requirements.
400	DMS.00400020	DHCP must be enabled for this subnet.	DHCP must be enabled for the subnet.	Check the DHCP status.
400	DMS.00400021	The isAutoRenew parameter in the request must be either 0 or 1.	Invalid isAutoRenew in the request.	Check the isAutoRenew parameter.
400	DMS.00400022	Engine does not match the product id.	The engine and product ID parameters do not match.	Check the engine parameter.
400	DMS.00400026	This operation is not allowed due to the instance status.	This operation is not allowed when the instance is in the current state.	Check the instance status.
400	DMS.00400028	Query advanced product, specCode not exists.	The specCode does not exist during the advanced feature query.	Check the origin_spec_code parameter.

Status Code	Error Codes	Error Message	Description	Solution
400	DMS.00400029	Query advanced product failed, can not find product for request.	The specCode does not exist during the advanced feature query.	Check the origin_spec_code parameter.
400	DMS.00400030	Invalid DMS instance id. The id must be a uuid.	Invalid instance ID.	Check the id parameter.
400	DMS.00400035	DMS instance quota of the tenant is insufficient.	Insufficient instance quota.	Apply for a higher quota.
400	DMS.00400037	The instanceParams parameter in the request contains invalid characters or is not in JSON format.	Request parameter instanceParams is not in JSON format or contains invalid characters.	Check the request parameter.
400	DMS.00400038	The periodNum parameter in the request must be an integer.	The periodNum parameter in the request must be an integer.	Check the periodNum parameter.
400	DMS.00400039	The quota limit has been reached.	The quota limit has been reached.	Apply for a higher quota.
400	DMS.00400042	The AZ does not exist.	The AZ does not exist.	Check the AZ.
400	DMS.00400045	The instance is not frozen and cannot be unfrozen.	The instance cannot be unfrozen because it is not frozen.	Check the instance status.
400	DMS.00400046	This security group does not exist.	The security group does not exist.	Check the security group.

Status Code	Error Codes	Error Message	Description	Solution
400	DMS.00400047	The periodType parameter in the request must be either 2 or 3.	Invalid periodType in the request.	Check the periodType parameter.
400	DMS.00400048	Invalid security group rules. Ensure that rules with the protocol being ANY are configured for both the inbound and outbound directions.	The security group must have both outbound and inbound rules with protocols set to ANY.	Check the security group rules.
400	DMS.00400049	The availability zone does not support ipv6.	The AZ does not support IPv6.	Select another AZ.
400	DMS.00400051	not found the new setup version tar to upgrade instance.	The package for upgrading the instance to the target version is not found.	Select another target version.
400	DMS.00400052	only the instance at running status can upgrade.	Only running instances can be upgraded.	Try again later.
400	DMS.00400053	the upgrade instance version equals to current version.	The target version is the same as the current version.	Select another target version.
400	DMS.00400055	Resource sold out.	Resources, such as ECS and volume resources, are insufficient.	Try again later.

Status Code	Error Codes	Error Message	Description	Solution
400	DMS.00400060	This instance name already exists.	The instance name already exists.	Check the instance name.
400	DMS.00400061	Invalid instance ID format.	Invalid instance ID.	Check the instance ID.
400	DMS.00400062	Invalid request parameter.	Invalid request parameters.	Check the request parameters.
400	DMS.00400063	Invalid configuration parameter {0}.	Invalid configuration parameter {0}.	Check the parameter.
400	DMS.00400064	The action parameter in the request must be delete or restart.	The action parameter in the request must be delete or restart.	Check the action parameter.
400	DMS.00400065	The instances parameter in the request is empty.	The instances parameter in the request is empty.	Check the instances parameter.
400	DMS.00400066	Invalid configuration parameter {0}.	Invalid configuration parameter {0}.	Check the parameter.
400	DMS.00400067	The available_zones parameter in the request must be an array that contains only one AZ ID.	Request parameter available_zones must be an array that contains only one AZ ID.	Check the available_zones parameter.
400	DMS.00400068	The VPC does not exist.	The VPC does not exist.	Check the VPC.
400	DMS.00400070	Invalid task ID format.	Invalid task ID.	Check the task ID.
400	DMS.00400081	Duplicate instance name.	The instance name already exists.	Check the instance name.

Status Code	Error Codes	Error Message	Description	Solution
400	DMS.00400082	Instance id is repeated.	The instance ID already exists.	Check the instance ID.
400	DMS.00400085	The message body contains invalid characters or is not in JSON format. The error key is <key>.	The message body is not in JSON format or contains invalid characters.	Check the message body.
400	DMS.00400099	The following instances in the Creating, Starting, Stopping, or Restarting state cannot be deleted.	Instances ({} in the Creating, Starting, Stopping, or Restarting state cannot be deleted.	Check the instance status.
400	DMS.00400100	The instances array can contain a maximum of 50 instance IDs.	The instances array can contain a maximum of 50 instance IDs.	Check the instance quantity.
400	DMS.00400101	The name of a Kafka topic must be 4 to 64 characters long and start with a letter. Only letters, digits, underscores (_), and hyphens (-) are allowed.	The name of a topic in a Kafka instance must be 4 to 64 characters long and start with a letter. Only letters, digits, underscores (_), and hyphens (-) are allowed.	Check the topic name.

Status Code	Error Codes	Error Message	Description	Solution
400	DMS.00400102	The number of partitions created for a Kafka topic must be within the range of 1-20.	The number of partitions created for a topic in a Kafka instance must be within the range of 1-20.	Check the number of partitions of the topic.
400	DMS.00400103	The number of replicas created for a Kafka topic must be within the range of 1-20.	The number of replicas created for a topic in a Kafka instance must be within the range of 1-20.	Check the number of replicas of the topic.
400	DMS.00400105	The message retention period of a Kafka topic must be within the range of 1-168.	The aging time of a topic in a Kafka instance must be within the range of 1-168.	Check the aging time of the topic.
400	DMS.00400106	Invalid maintenance time window.	Invalid maintenance time window.	Check the maintenance time window parameter.
400	DMS.00400107	The instance exists for unpaid scale up orders. Please process non payment orders first.	A specification modification order for the instance is pending.	Process the order first.
400	DMS.00400108	The Instance exists for processing scale up order. Please try again later.	A specification modification order for the instance is being processed.	Try again later.

Status Code	Error Codes	Error Message	Description	Solution
400	DMS.00400124	The maximum number of disk expansion times has been reached.	The maximum number of disk expansion times has been reached.	Check the maximum number of disk expansion times.
400	DMS.00400125	Invalid SPEC_CODE.	Invalid SPEC_CODE.	Check SPEC_CODE.
400	DMS.00400127	Instance not support to change retention_policy.	The instance does not support retention policy changes.	Contact technical support.
400	DMS.00400128	Invalid public access parameters.	Invalid public access parameters.	Check the public access parameters.
400	DMS.00400129	Current instance version is less than required.	The instance version does not support this operation.	Contact technical support.
400	DMS.00400134	There is another order need to pay first.	An unpaid order exists.	Pay for the order first.
400	DMS.00400135	Not support disk encrypted.	Disk encryption is not supported.	Do not enable disk encryption.
400	DMS.00400136	Disk encrypted key is null.	The disk encryption key is empty.	Check the disk encryption key.
400	DMS.00400137	Disk encrypted key state is not enabled.	The disk encryption key is not enabled.	Enable the disk encryption key.
400	DMS.00400142	Timestamp is invalid.	Timestamp is invalid.	Enter a correct timestamp.
400	DMS.00400500	Invalid disk space.	Invalid disk space.	Check the disk space.

Status Code	Error Codes	Error Message	Description	Solution
400	DMS.00400800	Duplicate topic name.Invalid request. You can create, delete, or query a topic only for a Kafka instance.Invalid parameter. Error information varies by scenario.	Invalid {0} in the request.	Check the instance.
400	DMS.00400861	Replication factor larger than available brokers.	The number of replicas in the topic to be created is greater than the number of available brokers.	Contact technical support.
400	DMS.00400969	The instance acl is off.	The instance acl is off.	Check the instance ssl status.
400	DMS.00400971	No message in the timestamp.	No message in the timestamp.	Please check the timestamp for message.
400	DMS.00404033	Does not support extend rabbitMQ disk space.	The RabbitMQ disk space cannot be expanded.	Scale out the RabbitMQ cluster.
400	DMS.10240002	The number of queried queues exceeds the upper limit.	The maximum number of queried queues has been reached.	Check the queue quantity.
400	DMS.10240004	The tag name is invalid.	Invalid tag name.	Check the tag name.
400	DMS.10240005	The project ID format is invalid.	Invalid project ID.	Check the project ID format.

Status Code	Error Codes	Error Message	Description	Solution
400	DMS.10240007	The name contains invalid characters.	The name contains invalid characters.	Check the name.
400	DMS.10240009	The message body is not in JSON format or contains invalid characters.	The message body is not in JSON format or contains invalid characters.	Check the message body.
400	DMS.10240010	The description contains invalid characters.	The description contains invalid characters.	Check the description.
400	DMS.10240011	The name length must be 1 to 64 characters.	The name can contain 1 to 64 characters.	Check the name length.
400	DMS.10240012	The name length must be 1 to 32 characters.	The name can contain 1 to 32 characters.	Check the name length.
400	DMS.10240013	The description length must not exceed 160 characters.	The description can contain a maximum of 160 characters.	Check the description length.
400	DMS.10240014	The number of consumable messages exceeds the maximum limit.	The number of consumable messages is not within the allowed range.	Check the number of consumable messages.
400	DMS.10240015	The queue ID format is invalid.	Invalid queue ID.	Check the queue ID.
400	DMS.10240016	The group ID format is invalid.	Invalid group ID.	Check the group ID.

Status Code	Error Codes	Error Message	Description	Solution
400	DMS.10240017	The queue already exists.	The queue already exists.	Check whether the queue exists.
400	DMS.10240018	The consumer group already exists.	The consumer group already exists.	Check whether the consumer group exists.
400	DMS.10240019	The number of consumer groups exceeds the upper limit.	The number of consumer groups exceeds the upper limit.	Check the number of consumer groups.
400	DMS.10240020	The quota is insufficient.	Insufficient quota.	Check the quota.
400	DMS.10240021	The value of time_wait is not within the value range of 1-60.	The value of time_wait is not within the range of 1-60.	Check the value of time_wait.
400	DMS.10240022	The value of max Consume Count must be within the range of 1-100.	The value of max Consume Count must be within the range of 1-100.	Check the value of max Consume Count.
400	DMS.10240027	The value of retention_hours must be an integer in the range of 1-72.	The value of retention_hours must be an integer in the range of 1-72.	Check the value of retention_hours.
400	DMS.10240028	Non-kafka queues do not support retention_hours.	Non-kafka queues do not support retention_hours.	Check whether the queue is a Kafka queue. If not, do not set retention_hours.
400	DMS.10240032	The queue is being created.	The queue is being created.	Check whether the queue is being created.
400	DMS.10240035	The tag key is empty or too long.	The tag key of the queue is empty or too long.	Check the tag key of the queue.

Status Code	Error Codes	Error Message	Description	Solution
400	DMS.10240036	The tag key contains invalid characters.	The tag key of the queue contains invalid characters.	Check the tag key of the queue.
400	DMS.10240038	The tag value is too long.	The tag value is too long.	Check the tag value of the queue.
400	DMS.10240039	The tag value contains invalid characters.	The tag value contains invalid characters.	Check the tag value of the queue.
400	DMS.10240040	You can only create or delete tags.	You can only create or delete tags.	Check whether the operation meets the requirements.
400	DMS.10240041	You can only filter or count tags.	You can only filter or count tags.	Check whether the operation meets the requirements.
400	DMS.10240042	The number of records on each page for pagination query exceeds the upper limit.	The number of records on each page for pagination query exceeds the upper limit.	Check the page size.
400	DMS.10240043	The number of skipped records for pagination query exceeds the upper limit.	The offset for pagination query exceeds the upper limit.	Check the paging offset.
400	DMS.10240044	A maximum of 10 tags can be created.	A maximum of 10 tags can be created.	Check the tag quantity.
400	DMS.10240045	The tag key has been used.	The tag key has been used.	Check whether the tag key has been used.
400	DMS.10540001	The message body contains invalid fields.	The message body contains invalid fields.	Check the message body.

Status Code	Error Codes	Error Message	Description	Solution
400	DMS.10540003	Message ack status must be either 'success' or 'fail'. It should not be '{status}'.	Message ack status must be either success or fail. It should not be {status}.	Check whether the status meets the requirements.
400	DMS.10540004	Request error	Request error. The queue or group name does not match the handler.	Check whether the queue or group name matches the handler.
400	DMS.10540010	The request format is incorrect	The request format is incorrect. {Error description}	Check the request format.
400	DMS.10540011	The message size is {message size}, larger than the size limit {max allowed size}.	The message size is {message size}, larger than the size limit {max allowed size}.	Check the request body size.
400	DMS.10540012	The message body is not in JSON format or contains invalid characters.	The message body is not in JSON format or contains invalid characters.	Check the message body format.
400	DMS.10540014	The URL contains invalid parameters.	The URL contains invalid parameters.	Check the URL parameters.
400	DMS.10540202	The request format is incorrect	The request format is incorrect. {Error description}	Check the request format.
400	DMS.10542204	Failed to consume messages due to {desc}.	Failed to consume messages. {Error description}	Check the error information and rectify the fault accordingly.

Status Code	Error Codes	Error Message	Description	Solution
400	DMS. 10542205	Failed to obtain the consumption instance because the handler does not exist. This may be because the consumer instance is released 1 minute after the message is consumed. As a result, the consumer instance fails to be obtained from the handler.	Failed to obtain the consumption instance because the handler does not exist. This may be because the consumer instance is released 1 minute after the message is consumed.	Check the handler.
400	DMS. 10542206	The value of ack_wait must be within the range of 15-300.	The value of ack_wait must be within the range of 15-300.	Check the value of ack_wait.
400	DMS. 10542209	The handler does not exist because the handler fails to be parsed, the message consumption times out, or the message consumption is repeatedly acknowledged.	The handler does not exist because the handler fails to be parsed, the message consumption times out, or the message consumption is repeatedly acknowledged.	Check whether the handler or consumption acknowledgment times out.
400	DMS. 10542214	The request format is incorrect	The request format is incorrect. {Error description}	Check the request format.
401	DMS. 10240101	Invalid token.	Invalid token.	Check whether the token is valid.

Status Code	Error Codes	Error Message	Description	Solution
401	DMS. 10240102	Expired token.	The token has expired.	Check whether the token has expired.
401	DMS. 10240103	Missing token.	The token is missing.	Check whether the token is missing.
401	DMS. 10240104	The project ID and token do not match.	The project ID and token do not match.	Check whether the project ID matches the token.
403	DMS. 00403002	A tenant has the read-only permission and cannot perform operations on DMS.	You cannot perform operations on DMS because you only have read permissions.	Check the tenant permission.
403	DMS. 00403003	This role does not have the permissions to perform this operation.	This role does not have the permissions to perform this operation.	Check the role permission.
403	DMS. 10240304	Change the quota of a queue or consumer group to a value smaller than the used quota.	The quota of a queue or consumer group cannot be smaller than the used amount.	Check the quota.
403	DMS. 10240306	The tenant has been frozen. You cannot perform operations on DMS.	The tenant has been frozen. You cannot perform operations on DMS.	Check the tenant status.
403	DMS. 10240307	The consumer group quota must be within the range of 1–10.	The consumer group quota must be within the range of 1–10.	Check whether the number of consumer groups exceeds the quota.

Status Code	Error Codes	Error Message	Description	Solution
403	DMS.10240308	The queue quota must be within the range of 1–20.	The queue quota must be within the range of 1–20.	Check whether the number of queues exceeds the quota.
403	DMS.10240309	Access denied. You cannot perform operations on DMS.	Access denied. You cannot perform operations on DMS.	Check whether you have the permission required to perform this operation.
403	DMS.10240310	A tenant has the read-only permission and cannot perform operations on DMS.	The tenant has read-only permissions and cannot perform operations on DMS.	Check the tenant permission.
403	DMS.10240311	This role does not have the permissions to perform this operation.	This role does not have the permissions required to perform operations on DMS.	Check the role permission.
403	DMS.10240312	The tenant is restricted and cannot perform operations on DMS.	The tenant is restricted and cannot perform operations on DMS.	Check the role permission.
404	DMS.00400970	The task has been deleted.	The task has been deleted.	Please check the existence of task.
404	DMS.00404001	The requested URL does not exist.	The requested URL does not exist.	Check the URL.
404	DMS.00404022	This instance does not exist.	The instance does not exist.	Check whether the instance exists.
404	DMS.10240401	The queue ID is incorrect or not found.	The queue ID is incorrect or is not found.	Check whether the queue ID exists and is correct.

Status Code	Error Codes	Error Message	Description	Solution
404	DMS. 10240405	The consumption group ID is incorrect or not found.	The consumption group ID is incorrect or is not found.	Check whether the consumer group ID exists and is correct.
404	DMS. 10240406	The URL or endpoint does not exist.	The URL or endpoint does not exist.	Check whether the URL or endpoint exists and is correct.
404	DMS. 10240407	The request is too frequent. Flow control is being performed. Please try again later.	The request is sent too frequently and flow control is being performed. Please try again later.	Try again later.
404	DMS. 10240426	No tag containing this key exists.	No tags containing this key exist.	Check the tag.
404	DMS. 10540401	The queue name does not exist.	The queue name does not exist.	Check whether the queue name exists.
405	DMS. 00405001	This request method is not allowed.	The request method is not allowed.	Check the request method.
500	DMS. 00500000	Internal service error.	Internal service error.	Contact technical support.
500	DMS. 00500006	Internal service error.	Internal service error.	Contact technical support.
500	DMS. 00500017	Internal service error.	Internal service error.	Contact technical support.
500	DMS. 00500024	Internal service error.	Internal service error.	Contact technical support.
500	DMS. 00500025	Internal service error.	Internal service error.	Contact technical support.
500	DMS. 00500041	Internal service error.	Internal service error.	Contact technical support.

Status Code	Error Codes	Error Message	Description	Solution
500	DMS.00500052	Internal service error.	Failed to submit the instance upgrade job.	Contact technical support.
500	DMS.00500053	Internal service error.	The specified instance node is not found.	Contact technical support.
500	DMS.00500054	Internal service error.	Failed to generate the password.	Contact technical support.
500	DMS.00500070	Internal service error.	Failed to configure the instance.	Contact technical support.
500	DMS.00500071	Internal service error.	Failed to create the instance backup policy.	Contact technical support.
500	DMS.00500094	Internal service error.	Internal service error.	Contact technical support.
500	DMS.00500106	Internal service error.	Internal service error.	Contact technical support.
500	DMS.10250002	Internal service error.	Internal service error.	Contact technical support.
500	DMS.10250003	Internal service error.	Internal service error.	Contact technical support.
500	DMS.10250004	Internal service error.	Internal service error.	Contact technical support.
500	DMS.10250005	Internal communication error.	Internal communication error.	Contact technical support.
500	DMS.10250006	Internal service error.	Internal service error.	Contact technical support.
500	DMS.10550035	tag_type must be either or or and.	tag_type must be either 'or' or 'and'.	Check tag_type.

7.3 Instance Statuses

Table 7-2 Instance status description

Status	Description
Creating	The instance is being created.
Create failed	The instance fails to be created.
Running	The instance is running properly. Instances in this state can provide services.
Faulty	The instance is not running properly.
Restarting	The instance is being restarted.
Changing	The instance specifications are being changed.
Change failed	The instance specifications failed to be changed.

7.4 Obtaining a Project ID

Scenario

A project ID is required for some URLs when an API is called. Obtain a project ID using either of the following methods:

- [Obtaining a Project ID by Calling an API](#)
- [Obtaining a Project ID on the Console](#)

Obtaining a Project ID by Calling an API

You can obtain a project ID by calling the IAM API used to query projects based on specified criteria.

The API used to obtain a project ID is **GET https://{Endpoint}/v3/projects**, where *{Endpoint}* indicates the IAM endpoint. You can obtain the IAM endpoint from [Regions and Endpoints](#). For details on API calling authentication, see [Authentication](#).

The following is an example response. The value of **id** in the **projects** section is the project ID:

```
{
  "projects": [
    {
      "domain_id": "65382450e8f64ac0870cd180d14e684b",
      "is_domain": false,
      "parent_id": "65382450e8f64ac0870cd180d14e684b",
      "name": "xxx-xxx-xxx",
      "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"
}
}
```

Obtaining a Project ID on the Console

A project ID is required for some URLs when an API is called. You can obtain a project ID on the console.

The following procedure describes how to obtain a project ID:

- Step 1** Log in to the management console.
- Step 2** Click the username and choose **My Credentials** from the drop-down list.

On the **My Credentials** page, view project IDs in the project list.

----End

7.5 Obtaining the Account Name and Account ID

When calling APIs, you need to specify your domain name and domain ID in certain URLs. To do so, you need to obtain domain name and domain ID on the console first. The following procedure describes how to obtain the domain and domain ID:

- Log in to the management console.
- Click the username and choose **My Credentials** from the drop-down list.

Viewing the account name and account ID

Figure 7-1 Viewing the domain name and domain ID



A Change History

Released On	Description
2023-01-09	This issue is the first official release.