Distributed Database Middleware

API Reference

Issue	01
Date	2024-04-01

HUAWEI TECHNOLOGIES CO., LTD.

Trademarks and Permissions


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

Notice

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

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

Huawei Technologies Co., Ltd.

Address: Huawei Industrial Base
Bantian, Longgang
Shenzhen 518129
People's Republic of China

Website: https://e.huawei.com

Security Declaration

Product Lifecycle

Huawei's regulations on product lifecycle are subject to the *Product End of Life Policy.* For details about this policy, visit the following web page:
https://support.huawei.com/ecolumnsweb/en/warranty-policy

Vulnerability

Huawei's regulations on product vulnerability management are subject to the *Vul. Response Process.* For details about this process, visit the following web page:
https://www.huawei.com/en/psirt/vul-response-process
For vulnerability information, enterprise customers can visit the following web page:
https://securitybulletin.huawei.com/enterprise/en/security-advisory

Preconfigured Digital Certificate

The digital certificates preconfigured on Huawei devices are subject to the *Rights and Responsibilities of Preconfigured Digital Certificates on Huawei Devices.* For details about this document, visit the following web page:
https://support.huawei.com/enterprise/en/bulletins-service/ENEWS2000015789

Huawei Enterprise End User License Agreement

This agreement is the end user license agreement between you (an individual, company, or any other entity) and Huawei for the use of the Huawei Software. Your use of the Huawei Software will be deemed as your acceptance of the terms mentioned in this agreement. For details about this agreement, visit the following web page:
https://e.huawei.com/en/about/eula

Lifecycle of Product Documentation

Huawei after-sales user documentation is subject to the *Product Documentation Lifecycle Policy.* For details about this policy, visit the following web page:
https://support.huawei.com/enterprise/en/bulletins-website/ENEWS2000017761

Contents

1 Before You Start

1.1 Overview

Welcome to Distributed Database Middleware (DDM). This document describes functions, syntax, parameters, and examples of DDM.

If you plan to access DDM using an API, ensure that you are familiar with DDM concepts.

This document describes how to use application programming interfaces (APIs) to perform creating, modifying, querying, and updating operations.

NOTE

- This document will be updated when APIs of new functions are added, for example, adding response parameters.
- To reduce impacts caused by API changes, DDM is backward compatible with existing APIs. When using DDM, you should accept and ignore unused parameters and parameter values in JSON responses.

1.2 API Calling

DDM supports Representational State Transfer (REST) APIs, allowing you to call APIs using HTTPS.

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 Constraints

- The number of DDM instances that you can create is determined by your quota.

- For more constraints, see API description.

1.5 Concepts

- Account

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

- IAM user

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

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

- Region

  A region is a geographic area in which cloud resources are deployed. Availability zones (AZs) in the same region can communicate with each other over an intranet, while AZs in different regions are isolated from each other. Deploying cloud resources in different regions can better suit certain user requirements or comply with local laws or regulations.

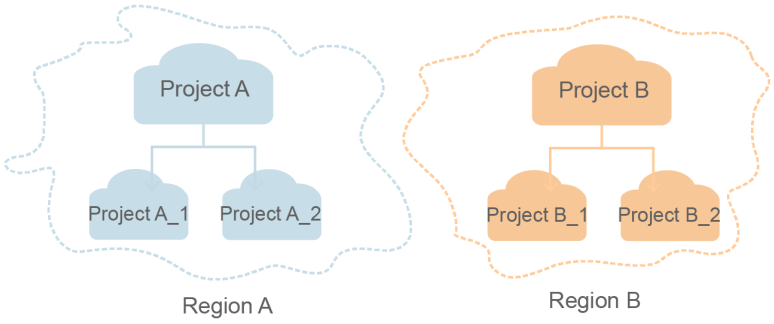
- AZ

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

- Project

  A project corresponds to a region. Projects group and isolate resources (including compute, storage, and network resources) across physical regions. Users can be granted permissions in a default project to access all resources in the region associated with the project. For more refined access control, create subprojects under a project and buy resources in the subprojects. Users can then be assigned permissions to access only specific resources in the subprojects.

Figure 1-1 Project isolating model



- 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 the enterprise project.

  For more information about enterprise projects and how to obtain enterprise project IDs, see ***Enterprise Management User Guide***.

2 API Overview

DDM provides REST APIs. With DDM APIs, you can use all DDM functions, including creating DDM instances and schemas.

Table 2-1 API description

Type	Subtype	Description
APIs for managing DDM instances	Buying a DDM instance	This API is used to buy a DDM instance. DDM instances created using this API are billed in pay-per-use mode. DDM runs in VPCs. Before you create a DDM instance, ensure that a VPC is available and a subnet and security group have been configured.
	Querying DDM instances	This API is used to query DDM instances.
	Querying details of a DDM instance	This API is used to query details about a DDM instance.
	Modifying the name of a DDM instance	This API is used to modify the name of an existing DDM instance.
	Modifying the security group of a DDM instance	This API is used to modify the security group of a DDM instance.
	Deleting a DDM instance	This API is used to delete a DDM instance to release all its resources.
	Restarting a DDM instance	This API is used to restart a DDM instance.
	Reloading table data	This API is used to reload table data of the destination DDM instance for cross-region DR.

Type	Subtype	Description
	Scaling out a DDM instance	This API is used to add nodes to a specified DDM instance.
	Scaling in a DDM instance	This API is used to remove nodes from a specified DDM instance.
	Modifying the read policy of the associated DB instance	This API is used to modify the read policy of the DB instance associated with a DDM instance.
	Synchronizing data node information	This API is used to synchronize configuration information of all data nodes that are associated with the current DDM instance.
	Querying nodes of a DDM instance	This API is used to query nodes of a DDM instance.
	Querying details of a DDM instance node	This API is used to query details of a DDM instance node.
	Querying parameters of a specified DDM instance	This API is used to query parameters of a specified DDM instance.
	Modifying parameters of a DDM instance	This API is used to modify parameters of a DDM instance.
	Querying DDM engine information	This API is used to query information about DDM engine.
	Querying DDM node classes available in an AZ	This API is used to query DDM node classes available in an AZ.
	Changing the node class of a DDM Instance	This API is used to change the node class of a DDM instance.
APIs for managing schemas	Creating a schema	This API is used to create a schema. Before creating a schema, ensure that there is an RDS instance available and that the instance is running properly and not associated with any DDM instance.

Type	Subtype	Description
	Querying schemas of a DDM instance	This API is used to query schemas of a DDM instance.
	Querying details of a schema	This API is used to query details about a schema.
	Deleting a schema	This API is used to delete a schema to release all its resources.
	Querying DB instances available for creating a schema	This API is used to query DB instances that can be used for creating a schema.
APIs for managing DDM accounts	Creating a DDM account	This API is used to create a DDM account. DDM accounts are used to connect to and manage schemas. One DDM account can be associated with multiple schemas.
	Querying DDM accounts	This API is used to query DDM accounts.
	Modifying a DDM account	This API is used to modify the permissions and associated schemas of a DDM account.
	Deleting a DDM account	This API is used to delete a DDM account. This operation will also disassociate the account from schemas if any.
	Resetting the password of a DDM account	This API is used to reset the password of a DDM account.
	Managing the administrator password	If it is the first time to call this API, it is used to create an administrator and reset its password for a DDM instance. Then this API can only be used to update the administrator password.
	Validating password strength	This API is used to validate whether an instance password is a weak password.
Monitoring	Monitoring slow query logs	This API is used to query the SQL statements that take a long time to execute on the DDM instance within a specified time range.
	Monitoring the read/write ratio (unavailable Soon)	This API is used to query reads and writes of a DDM instance in a specified time range.

3 Calling APIs

3.1 Making an API Request

This section describes the structure of a REST API and how to call an API. Before calling an API, you need to **obtain the user token** using the IAM API.

Request URI

A request URI is in the following format:

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

Although a request URI is a part of a request header, most programming languages or frameworks require the request URI to be separately transmitted, rather than being conveyed in a request message.

Table 3-1 URI parameter description

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 endpoint. The endpoint varies depending on the service and service region. *Endpoint*: Endpoint of the **Objective-func** function. For details, see **Regions and Endpoints**. For example, the endpoint of IAM in region **CN-Hong Kong** is **iam.ap-southeast-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 "Parameter name=Parameter value". For example, **? limit=10** indicates that a maximum of 10 data records will be displayed.

For example, to obtain an IAM token in the **CN-Hong Kong** region, obtain the endpoint of IAM (**iam.ap-southeast-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.ap-southeast-1.myhuaweicloud.com/v3/auth/tokens
```

NOTE

To simplify the URI display, each API is provided only with a resource-path and a request method. This is because the **URI-scheme** value of all APIs is **HTTPS**, and the endpoints in a region are the same. Therefore, the two parts are omitted.

Request Method

HTTP methods, which are also called operations or actions, specify the type of operations that you are requesting.

Table 3-2 HTTP methods

Method	Description
GET	Requests a server to return specified resources.
PUT	Requests a server to update specified resources.
POST	Requests the server to add a resource or perform special operations.
DELETE	Requests a server to delete specified resources, for example, an object.

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

```
POST https://iam.cn-north-1.myhuaweicloud.com/v3/auth/tokens
```

Request Headers

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.

You can also add additional fields to the request header, for example, the fields required by a specified URI and an HTTP method. **Table 3-3** lists common request header fields.

Table 3-3 Common request headers

Field	Description	Mandatory	Example
Host	Requested server information, which can be obtained from the URL of the service API. The value is in the *hostname[:port]* format. If the port number is not specified, the default port is used. The default port number for **https** is **443**.	No This parameter is mandatory for AK/SK authentication.	code.test.com or code.test.com:443
Content-Type	MIME type of the request body. You are advised to use the default value **application/json**. For APIs used to upload objects or images, the value varies depending on the flow type.	Yes	application/json
Content-Length	Length of the request body. The unit is byte.	This parameter is optional for POST requests, but must be left blank for GET requests.	3495

Field	Description	Mandatory	Example
X-Project-Id	Project ID. Obtain the project ID by following the instructions in **Obtaining a Project ID**.	No This parameter is mandatory for requests that use AK/SK authentication in the Dedicated Cloud (DeC) scenario or multi-project scenario.	e9993fc787d94b6c886cbaa340f9c0f4
X-Auth-Token	User token After the request is processed, the value of **X-Subject-Token** in the header is the token value.	No This parameter is mandatory for token authentication.	The following is part of an example token: MIIPAgYJKoZIhvcNAQc-Co...ggg1BBIINPXsidG9rZ

NOTE

In addition to supporting token-based authentication, APIs support authentication using access key ID/secret access key (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**.

The API used to **obtain a user token** does not require authentication. Therefore, this API only requires adding the **Content-Type** field. The request with the added **Content-Type** header is as follows:

```
POST https://iam.cn-north-1.myhuaweicloud.com/v3/auth/tokens
Content-Type: application/json
```

(Optional) Request Body

This part is optional. The body of a request is often sent in a structured format (for example, JSON or XML) as specified in the **Content-Type** header field. If the request body contains Chinese characters, these characters must be coded in UTF-8.

The request body varies depending on APIs. Some APIs do not require the request body, such as the APIs requested using GET and DELETE methods.

For the API used to **obtain a user token**, the request parameters and parameter description can be obtained in the API request. The following provides an example request with a body included. Replace ***username***, ***domainname***, ******** (login

password), and ***xxxxxxxxxxxxxxxxx*** (project name such as cn-north-1) with actual values. You can obtain the values 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.

```
POST https://iam.cn-north-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": "xxxxxxxxxxxxxxxxx"
            }
        }
    }
}
```

If all data required for the API request is available, you can send the request to call the API through **curl**, **Postman**, or coding. For the API of obtaining a user token, **x-subject-token** in the response header is the required user token. This token can then be used to authenticate the calling of other APIs.

Initiating Requests

You can initiate a request based on the constructed request message in one of the following ways:

- cURL

  cURL is a command-line tool used to perform URL operations and transmit information. It serves as an HTTP client that can send HTTP requests to the server and receive response messages. cURL is used for API debugging. For more information about cURL, visit **https://curl.haxx.se/**.

  > **NOTE**
  >
  > For security purposes, run the **curl** command on the server to query information, and then clear operation records, including but not limited to records in the **~/.bash_history** and **/var/log/messages** directories (if any).

- Code

  You can call APIs using code to assemble, send, and process request messages.

- REST client

Both Mozilla Firefox and Google Chrome provide a graphical browser plug-in, REST client, to send and process requests. For Mozilla Firefox, see **Firefox REST Client**. For Google Chrome, see **Chrome REST Client**.

3.2 Authentication

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

- Token-based authentication: Requests are authenticated using a token.
- AK/SK-based authentication: Requests are authenticated by encrypting the request body using an AK/SK pair. Authentication using AK/SK is recommended because it is more secure than authentication using tokens.

Token-based Authentication

NOTE

The validity period of a token is 24 hours. If a token is required, the system caches the token to avoid frequent calling.

A token specifies temporary permissions in a computer system. Token-based authentication adds a token in a request as its header during API calling to obtain the permissions for operating APIs on IAM.

When **calling an API 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": "xxxxxxxx"
            }
        }
    }
}
```

In **Making an API Request**, the process of calling the API used to **obtain a user token** is described.

After obtaining the token, add the **X-Auth-Token** header in a request 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:

```
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 requests for authentication.

- AK: access key ID, which is a unique identifier associated with a secret access key and is 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 or 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 Codes**.

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

Response Header

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

Figure 3-1 shows the response header for the API used to **obtain a user token**. The **x-subject-token** header field is the desired user token. This token can then be used to authenticate the calling of other APIs.

Figure 3-1 Header fields of the response to the request for obtaining a user token

(Optional) Response Body

This part is optional. The body of a response is often returned in structured format (for example, JSON or XML) as specified in the **Content-Type** header field. The response body transfers content except the response header.

For the API used to **obtain a user token**, the following information is returned. The following describes part of the request body.

```
{
    "token": {
        "expires_at": "2019-02-13T06:52:13.855000Z",
        "methods": [
            "password"
        ],
        "catalog": [
            {
                "endpoints": [
                    {
                        "region_id": "az-01",
.....
```

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

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

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

4 APIs (Recommended)

4.1 DDM Instances

4.1.1 Buying a DDM instance

Function

This API is used to buy a DDM instance.

Constraints

DDM runs in VPCs. Before you create a DDM instance, ensure that a VPC is available and a subnet and security group have been configured.

URI

POST /v1/{project_id}/instances

Table 4-1 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID of a tenant in a region

Request Parameters

Table 4-2 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token It can be obtained by calling an IAM API. The value of **X-Subject-Token** in the response header is the user token.

Table 4-3 Request body parameters

Parameter	Mandatory	Type	Description
instance	Yes	**CreateInstanceDetail** object	Instance information
extend_param	No	**CreateInstanceExtendParam** object	Extended parameter, which is available only when you buy a yearly/monthly instance

Table 4-4 CreateInstanceDetail

Parameter	Mandatory	Type	Description
name	Yes	String	Name of a DDM instance, which: • Can include 4 to 64 characters. • Must start with a letter. • Can contain only letters, digits, and hyphens (-). Minimum length: 4 characters Maximum length: 64 characters
flavor_id	Yes	String	Class ID
node_num	Yes	Integer	Number of nodes
engine_id	Yes	String	Engine ID
enterprise_project_id	No	String	Enterprise project ID

Parameter	Mandatory	Type	Description
available_zones	Yes	Array of strings	AZ code. The value cannot be empty. For details, see **Regions and Endpoints**.
vpc_id	Yes	String	VPC ID
security_group_id	Yes	String	Security group ID
subnet_id	Yes	String	Subnet ID
param_group_id	No	String	Parameter group ID
time_zone	No	String	UTC time zone. The default value is **UTC**. The value can be **UTC**, **UTC-12:00**, **UTC-11:00**, **UTC-10:00**, **UTC-09:00**, **UTC-08:00**, **UTC-07:00**, **UTC-06:00**, **UTC-05:00**, **UTC-04:00**, **UTC-03:00**, **UTC-02:00**, **UTC-01:00**, **UTC+01:00**, **UTC+02:00**, **UTC+03:00**, **UTC+05:00**, **UTC+06:00**, **UTC+07:00**, **UTC+08:00**, **UTC+09:00**, **UTC+10:00**, **UTC+11:00**, and **UTC+12:00**.
admin_user_name	No	String	Username of the administrator. The username: • Can include 1 to 32 characters. • Must start with a letter. • Can contain only letters, digits, and underscores (_).
admin_user_password	No	String	Password of the administrator. The password: • Can include 8 to 32 characters. • Must be a combination of uppercase letters, lowercase letters, digits, and the following special characters: ~!@#%^*-_=+? Must be a strong password to improve security and prevent security risks such as brute force cracking.

Table 4-5 CreateInstanceExtendParam

Parameter	Mandatory	Type	Description
charge_mode	No	String	Billing mode. The value can be: • **prePaid**: indicates the prepayment method. • **postPaid**: indicates the post-payment method. Default value: **postPaid** Default value: **postPaid** Enumerated values: • **prePaid** • **postPaid**
period_type	No	String	Subscription period. The value can be: • **month**: indicates the subscription time unit is month. • **year**: indicates the subscription time unit is year. This parameter is valid and mandatory if **charge_mode** is set to **prePaid**. Enumerated values: • **month** • **year**
period_num	No	Integer	Subscription duration. • If **period_type** is set to **month**, the value ranges from **1** to **9**. • If **period_type** is set to **year**, the value is **1**. This parameter is valid and mandatory if **charge_mode** is set to **prePaid**.

Parameter	Mandatory	Type	Description
is_auto_renew	No	String	Whether the instance is automatically renewed when it expires. The value can be: • **true**: The instance is automatically renewed when it expires. • **false**: The instance is not automatically renewed when it expires. This parameter is valid when **charge_mode** is set to **prePaid**. The instance is not automatically renewed by default if no value is specified. " Enumerated values: • **true** • **false**
is_auto_pay	No	String	Whether the order is paid from the customer's account balance. The value can be: • **true**: indicates that the order is automatically paid from the customer's account balance. • **false**: indicates that the order needs to be paid manually. This parameter is available if **charge_mode** is set to **prePaid**. The order needs to be paid manually by default if no value is specified. " Enumerated values: • **true** • **false**

Response Parameters

Status code: 200

Table 4-6 Response body parameters

Parameter	Type	Description
id	String	Instance ID. This parameter is returned only when a pay-per-use instance is created.
order_id	String	Order ID. This parameter is returned only when a yearly/monthly instance is created.

Status code: 400

Table 4-7 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Status code: 500

Table 4-8 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Example Request

- Example request (pay-per-use, without extended parameters)

```
POST https://{endpoint}/v1/{project_id}/instances

{
  "instance" : {
    "name" : "ddm-test-001",
    "flavor_id" : "8f2e696c-a9c1-30bd-af90-25522bc67606",
    "node_num" : 4,
    "engine_id" : "2325a707-0361-8be6-dd01-13474bbac437",
    "enterprise_project_id" : "0",
    "available_zones" : [ "az1xahz" ],
    "vpc_id" : "e1d886ec-cfe7-4cd4-b748-fc55a10b4172",
    "security_group_id" : "035b70ed-319b-4086-9fd7-62a2e8548b2e",
    "subnet_id" : "f942f970-1a02-4eee-8927-xxxxxxxx",
    "param_group_id" : "035b70ed-319b-4086-9fd7-xxxxxxxx"
  }
}
```

- Example request (pay-per-use, with extended parameters)

```
POST https://{endpoint}/v1/{project_id}/instances
```

```
{
  "instance" : {
    "name" : "ddm-test-002",
    "flavor_id" : "8f2e696c-a9c1-30bd-af90-25522bc67606",
    "node_num" : 4,
    "engine_id" : "2325a707-0361-8be6-dd01-13474bbac437",
    "enterprise_project_id" : "0",
    "available_zones" : [ "az1xahz" ],
    "vpc_id" : "e1d886ec-cfe7-4cd4-b748-fc55a10b4172",
    "security_group_id" : "035b70ed-319b-4086-9fd7-62a2e8548b2e",
    "subnet_id" : "f942f970-1a02-4eee-8927-e8670ce5a882"
  },
  "extend_param" : {
    "charge_mode" : "postPaid"
  }
}
```

- Example request (yearly/monthly)

```
POST https://{endpoint}/v1/{project_id}/instances

{
  "instance" : {
    "name" : "ddm-test-003",
    "flavor_id" : "8f2e696c-a9c1-30bd-af90-25522bc67606",
    "node_num" : 4,
    "engine_id" : "2325a707-0361-8be6-dd01-13474bbac437",
    "enterprise_project_id" : "0",
    "available_zones" : [ "az1xahz" ],
    "vpc_id" : "e1d886ec-cfe7-4cd4-b748-fc55a10b4172",
    "security_group_id" : "035b70ed-319b-4086-9fd7-62a2e8548b2e",
    "subnet_id" : "f942f970-1a02-4eee-8927-xxxxxxxx",
    "param_group_id" : "035b70ed-319b-4086-9fd7-xxxxxxxx"
  },
  "extend_param" : {
    "charge_mode" : "prePaid",
    "period_type" : "month",
    "period_num" : 1,
    "is_auto_renew" : "true",
    "is_auto_pay" : "true"
  }
}
```

Example Response

Status code: 200

OK

Buying a pay-per-use instance:

```
{
  "id": "e90bc6739a3c4666a577c3fa1524dac2in09",
  "order_id": null
}
```

Buying a yearly/monthly instance:

```
{
  "id" : null,
  "order_id" : "CS1810251738L8VVD"
}
```

Status code: 400

bad request

```
{
  "externalMessage" : "Parameter error.",
```



```
  "errCode" : "DBS.280001"
}
```

Status code: 500

server error

```
{
  "externalMessage" : "Server failure.",
  "errCode" : "DBS.200412"
}
```

Status Codes

Status Code	Description
200	OK
400	bad request
500	server error

Error Codes

For details, see **Error Codes**.

4.1.2 Querying DDM Instances

Function

This API is used to query DDM instances.

Constraints

None

URI

GET /v1/{project_id}/instances

Table 4-9 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID of a tenant in a region

Table 4-10 Query parameters

Parameter	Mandatory	Type	Description
offset	No	Integer	Which page the server starts returning items. The start value cannot be less than **0**. The default value is **0**. Minimum value: **0**
limit	No	Integer	Number of records displayed on each page. The value is greater than **0** and not greater than **128**. The default value is **10**. Minimum value: **1** Maximum value: **128**

Request Parameters

Table 4-11 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token It can be obtained by calling an IAM API. The value of **X-Subject-Token** in the response header is the user token.

Response Parameters

Status code: 200

Table 4-12 Response body parameters

Parameter	Type	Description
instances	Array of **ShowInstanceBeanResponse** objects	DDM instance information
instance_num	Integer	Number of DDM instances of a tenant
page_no	Integer	Current page
page_size	Integer	Data records on the current page

Parameter	Type	Description
total_record	Integer	Total records
total_page	Integer	Total pages

Table 4-13 ShowInstanceBeanResponse

Parameter	Type	Description
id	String	DDM instance ID
status	String	DDM instance status
name	String	Name of the created DDM instance
created	String	Time when the DDM instance is created. The time must be in the format of yyyy-mm-dd Thh:mm:ssZ. **T** indicates the separator between the calendar and the hourly notation of time. **Z** indicates Coordinated Universal Time (UTC).
updated	String	Time when the DDM instance is last updated, which is in the same format as **created**.
available_zone	String	AZ name
vpc_id	String	VPC ID
subnet_id	String	Subnet ID
security_group_id	String	Security group ID
node_count	Integer	Number of nodes
access_ip	String	Address for accessing the DDM instance
access_port	String	Port for accessing the DDM instance
core_count	String	Number of CPUs
ram_capacity	String	Memory size in GB
error_msg	String	Response message. This parameter is not returned if no abnormality occurs.
node_status	String	Node status
enterprise_project_id	String	Enterprise project ID
project_id	String	Project ID of a tenant in a region

Parameter	Type	Description
engine_version	String	Engine version (core version)
order_id	String	Order ID, which is returned if the DDM instance is billed on a yearly/monthly basis

Status code: 400

Table 4-14 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Status code: 500

Table 4-15 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Example Request

```
GET https://{endpoint}/v1/{project_id}/instances?offset={offset}&limit={limit}
```

Example Response

Status code: 200

OK

```
{
  "instance_num" : 26,
  "instances" : [ {
    "id" : "73f27bb0b26d4a79a9b0c58a4c700c4fin09,",
    "status" : "RUNNING,",
    "name" : "ddm-36c7,",
    "created" : "2021-12-14T07:46:22+0000,",
    "updated" : "2021-12-14T10:11:22+0000,",
    "available_zone" : "az1xahz,",
    "vpc_id" : "1d9441c5-1a4b-431d-9edb-bf3564a83e74,",
    "subnet_id" : "20c2d83c-755b-42a2-846a-4688eb35d1a7,",
    "security_group_id" : "fbdff89d-32fe-45a4-832b-115ac4bfaac9,",
    "node_count" : "1,",
```

```
    "access_ip" : "192.168.23.143",
    "access_port" : "5066,",
    "core_count" : "2,",
    "ram_capacity" : "4,",
    "node_status" : "RUNNING,",
    "enterprise_project_id" : "0,",
    "project_id" : "055d9f4ee780d4d42f96c01c1bc3c50c,",
    "engine_version" : "3.0.8"
  } ],
  "page_size" : 1,
  "total_record" : 26,
  "total_page" : 26,
  "page_no" : 3
}
```


Status code: 400

bad request


```
{
  "externalMessage" : "Parameter error.",
  "errCode" : "DBS.280001"
}
```


Status code: 500

server error


```
{
  "externalMessage" : "Server failure.",
  "errCode" : "DBS.200412"
}
```


Status Codes

Status Code	Description
200	OK
400	bad request
500	server error

Error Codes

For details, see **Error Codes**.

4.1.3 Querying Details of a DDM Instance

Function

This API is used to query details about a DDM instance.

Constraints

None

URI

GET /v1/{project_id}/instances/{instance_id}

Table 4-16 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID of a tenant in a region
instance_id	Yes	String	DDM instance ID

Request Parameters

Table 4-17 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token It can be obtained by calling an IAM API. The value of **X-Subject-Token** in the response header is the user token.

Response Parameters

Status code: 200

Table 4-18 Response body parameters

Parameter	Type	Description
id	String	DDM instance ID
status	String	DDM instance status
name	String	DDM instance name
created	String	Time when the DDM instance is created
updated	String	Time when the DDM instance is last updated
available_zone	String	Name of the AZ where the DDM instance is located
vpc_id	String	VPC ID
subnet_id	String	Subnet ID
security_group_id	String	Security group ID

Parameter	Type	Description
node_count	Integer	Number of nodes
access_ip	String	Address for accessing the DDM instance
access_port	String	Port for accessing the DDM instance
node_status	String	Node status
core_count	String	Number of CPUs
ram_capacity	String	Memory size in GB
error_msg	String	Response message. This parameter is not returned if no abnormality occurs.
project_id	String	Project ID
order_id	String	Order ID. No value is returned when a pay-per-use instance is created.
enterprise_project_id	String	Enterprise project ID
engine_version	String	Engine version (core version)
nodes	Array of **GetDetailfNodesInfo** objects	Node information
admin_user_name	String	Username of the administrator. The username: • Can include 1 to 32 characters. • Must start with a letter. • Can contain only letters, digits, and underscores (_).

Table 4-19 GetDetailfNodesInfo

Parameter	Type	Description
status	String	Status of the DDM instance node
port	String	Port of the DDM instance node
ip	String	IP address of the DDM instance node

Status code: 400

Table 4-20 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Status code: 500

Table 4-21 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Example Request

```
GET https://{endpoint}/v1/{project_id}/instances/{instance_id}
```

Example Response

Status code: 200

OK

```
{
  "nodes" : [ {
    "status" : "RUNNING",
    "port" : 5066,
    "ip" : "192.168.0.160"
  } ],
  "id" : "1f5c9fd6cd984056ba89c8c87cc03278in09",
  "status" : "RUNNING",
  "name" : "BUG-ddm2-lixingqiao-test",
  "created" : "2021-11-09T03:30:01+0000",
  "updated" : "2021-12-15T09:12:58+0000",
  "available_zone" : "az1xahz",
  "vpc_id" : "cfaa4024-0603-4aba-81d4-2203b4ad26fb",
  "subnet_id" : "48f270ef-af70-4ad9-bb1c-c28dd5b37f93",
  "security_group_id" : "bc28ef93-0083-4652-bce6-381e14284db6",
  "node_count" : "1",
  "access_ip" : "192.168.0.160",
  "access_port" : "5066",
  "core_count" : "2",
  "ram_capacity" : "8",
  "node_status" : "RUNNING",
  "enterprise_project_id" : "0",
  "project_id" : "070c071d8e80d58c2f42c0121b10cf9f",
  "engine_version" : "3.0.6",
  "admin_user_name" : "root"
}
```

Status code: 400

bad request

```
{
  "externalMessage" : "Parameter error.",
  "errCode" : "DBS.280001"
}
```

Status code: 500

server error

```
{
  "externalMessage" : "Server failure.",
  "errCode" : "DBS.200412"
}
```

Status Codes

Status Code	Description
200	OK
400	bad request
500	server error

Error Codes

For details, see **Error Codes**.

4.1.4 Modifying the Name of a DDM Instance

Function

This API is used to modify the name of a DDM instance.

Constraints

None

URI

PUT /v1/{project_id}/instances/{instance_id}/modify-name

Table 4-22 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID of a tenant in a region
instance_id	Yes	String	DDM instance ID

Request Parameters

Table 4-23 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token It can be obtained by calling an IAM API. The value of **X-Subject-Token** in the response header is the user token.

Table 4-24 Request body parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Name of a DDM instance, which: • Can include 4 to 64 characters. • Must start with a letter. • Can contain only letters, digits, and hyphens (-). Minimum length: 4 characters Maximum length: 64 characters

Response Parameters

Status code: 200

Table 4-25 Response body parameters

Parameter	Type	Description
name	String	DDM instance name

Status code: 400

Table 4-26 Response body parameters

Parameter	Type	Description
errCode	String	Service error code

Parameter	Type	Description
externalMessage	String	Error message

Status code: 500

Table 4-27 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Example Request

```
PUT https://{endpoint}/v1/{project_id}/instances/{instance_id}/modify-name

{
  "name" : "DDM-test-04"
}
```

Example Response

Status code: 200

OK

```
{
  "name" : "DDM-test-04"
}
```

Status code: 400

bad request

```
{
  "externalMessage" : "Parameter error.",
  "errCode" : "DBS.280001"
}
```

Status code: 500

server error

```
{
  "externalMessage" : "Server failure.",
  "errCode" : "DBS.200412"
}
```

Status Codes

Status Code	Description
200	OK
400	bad request
500	server error

Error Codes

For details, see **Error Codes**.

4.1.5 Modifying the Security Group of a DDM Instance

Function

This API is used to modify the security group of a DDM instance.

Constraints

None

URI

PUT /v1/{project_id}/instances/{instance_id}/modify-security-group

Table 4-28 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID of a tenant in a region
instance_id	Yes	String	DDM instance ID

Request Parameters

Table 4-29 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token It can be obtained by calling an IAM API. The value of **X-Subject-Token** in the response header is the user token.

Table 4-30 Request body parameters

Parameter	Mandatory	Type	Description
security_group_id	Yes	String	Security group ID. The default value is the original security group ID. You can change the value as required.

Response Parameters

Status code: 200

Table 4-31 Response body parameters

Parameter	Type	Description
security_group_id	String	Security group ID

Status code: 400

Table 4-32 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Status code: 500

Table 4-33 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Example Request

```
PUT https://{endpoint}/v1/{project_id}/instances/{instance_id}/modify-security-group

{
  "security_group_id" : "035b70ed-319b-4086-9fd7-62a2e8548b2e"
}
```

Example Response

Status code: 200

OK

```
{
  "security_group_id" : "035b70ed-319b-4086-9fd7-62a2e8548b2e"
}
```

Status code: 400

bad request

```
{
  "externalMessage" : "Parameter error.",
  "errCode" : "DBS.280001"
}
```

Status code: 500

server error

```
{
  "externalMessage" : "Server failure.",
  "errCode" : "DBS.200412"
}
```

Status Codes

Status Code	Description
200	OK
400	bad request
500	server error

Error Codes

For details, see **Error Codes**.

4.1.6 Deleting a DDM Instance

Function

This API is used to delete a DDM instance to release all its resources.

Constraints

None

URI

DELETE /v1/{project_id}/instances/{instance_id}

Table 4-34 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
instance_id	Yes	String	DDM instance ID

Table 4-35 Query parameters

Parameter	Mandatory	Type	Description
delete_rds_data	No	String	Whether data stored on the associated DB instances is deleted. The value can be: • **true**: indicates that the data stored on the associated DB instances is deleted. • **false**: indicates that the data stored on the associated DB instances is not deleted. It is left blank by default. Enumerated values: • **true** • **false**

Request Parameters

Table 4-36 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token It can be obtained by calling an IAM API. The value of **X-Subject-Token** in the response header is the user token.

Response Parameters

Status code: 200

Table 4-37 Response body parameters

Parameter	Type	Description
id	String	DDM instance ID

Status code: 400

Table 4-38 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Status code: 500

Table 4-39 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Example Request

- Request to delete a DDM instance (including the data stored on the associated DB instances)
  ```
  DELETE https://{endpoint}/v1/{project_id}/instances/{instance_id}?delete_rds_data=true
  ```
- Request to delete a DDM instance (excluding the data stored on the associated DB instances)
  ```
  DELETE https://{endpoint}/v1/{project_id}/instances/{instance_id}?delete_rds_data=false
  ```

Example Response

Status code: 200

OK

```
{
  "id" : "9608ce63-bf66-4342-be8a-44f9b6b15d54"
}
```

Status code: 400

bad request

```
{
  "externalMessage" : "Parameter error.",
```



```
  "errCode" : "DBS.280001"
}
```

Status code: 500

server error

```
{
  "externalMessage" : "Server failure.",
  "errCode" : "DBS.200412"
}
```

Status Codes

Status Code	Description
200	OK
400	bad request
500	server error

Error Codes

For details, see **Error Codes**.

4.1.7 Restarting a DDM Instance

Function

This API is used to restart a DDM instance.

Constraints

None

URI

POST /v1/{project_id}/instances/{instance_id}/action

Table 4-40 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID of a tenant in a region
instance_id	Yes	String	DDM instance ID

Request Parameters

Table 4-41 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token It can be obtained by calling an IAM API. The value of **X-Subject-Token** in the response header is the user token.

Table 4-42 Request body parameters

Parameter	Mandatory	Type	Description
restart	No	**RestarInstanceInfo** object	Restart-related parameter

Table 4-43 RestarInstanceInfo

Parameter	Mandatory	Type	Description
type	No	String	Restart type, which can be **soft** or **hard**. • **soft**: Only the process is restarted. • **hard**: The instance VM is forcibly restarted. Enumerated values: • **soft** • **hard**

Response Parameters

Status code: 200

Table 4-44 Response body parameters

Parameter	Type	Description
instanceId	String	DDM instance ID
instanceName	String	DDM instance name
jobId	String	Task ID

Status code: 400

Table 4-45 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Status code: 500

Table 4-46 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Example Request

```
POST https://{endpoint}/v1/{project_id}/instances/{instance_id}/action

{
  "restart" : {
    "type" : "soft"
  }
}
```

Example Response

Status code: 200

ok

```
{
  "instanceId" : "28e8841d0b9c4f6a9a30742ee60e1068in09",
  "instanceName" : "BUG-ddm-fb88-test",
  "jobId" : "1eb697c0-1842-43a3-8671-f562d0385cb9"
}
```

Status code: 400

bad request

```
{
  "externalMessage" : "Parameter error.",
  "errCode" : "DBS.280001"
}
```

Status code: 500

server error

```
{
  "externalMessage" : "Server failure.",
```

```
  "errCode" : "DBS.200412"
}
```

Status Codes

Status Code	Description
200	ok
400	bad request
500	server error

Error Codes

For details, see **Error Codes**.

4.1.8 Reloading Table Data

Function

This API is used to reload table data of the destination DDM instance for cross-region DR.

Constraints

None

URI

POST /v1/{project_id}/instances/{instance_id}/reload-config

Table 4-47 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID of a tenant in a region
instance_id	Yes	String	DDM instance ID

Request Parameters

Table 4-48 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token It can be obtained by calling an IAM API. The value of **X-Subject-Token** in the response header is the user token.

Response Parameters

Status code: 400

Table 4-49 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Status code: 500

Table 4-50 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Example Request

```
POST https://{endpoint}/v1/{project_id}/instances/{instance_id}/reload_config

{ }
```

Example Response

Status code: 200

ok

```
{ }
```

Status code: 400

bad request

```
{
  "externalMessage" : "Parameter error.",
  "errCode" : "DBS.280001"
}
```

Status code: 500

server error

```
{
  "externalMessage" : "Server failure.",
  "errCode" : "DBS.200412"
}
```

Status Codes

Status Code	Description
200	ok
400	bad request
500	server error

Error Codes

For details, see **Error Codes**.

4.1.9 Scaling Out a DDM Instance

Function

This API is used to scale out a specified DDM instance. Pay-per-use and yearly/monthly instances are all supported.

Constraints

Make sure that the associated RDS instances are available and not undergoing other operations.

URI

POST /v2/{project_id}/instances/{instance_id}/action/enlarge

Table 4-51 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID of a tenant in a region
instance_id	Yes	String	DDM instance ID

Request Parameters

Table 4-52 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token It can be obtained by calling an IAM API. The value of **X-Subject-Token** in the response header is the user token.

Table 4-53 Request body parameters

Parameter	Mandatory	Type	Description
flavor_id	Yes	String	Flavor ID of the VM for deploying the DDM instance that is to be scaled out
node_number	Yes	Integer	Number of nodes to be added
group_id	No	String	Group ID, which specifies the node group that is scaled out. This parameter must be specified if there are more than one node group.
is_auto_pay	No	Boolean	Whether the order is automatically paid when you change the node class of a yearly/monthly instance. This parameter does not affect the billing mode of automatic renewal. **true**: indicates that the order is automatically paid from your account. **false**: indicates that the order is manually paid from your account. The default value is **false**.

Response Parameters

Status code: 200

Table 4-54 Response body parameters

Parameter	Type	Description
instanceId	String	DDM instance ID
instanceName	String	DDM instance name
jobId	String	Task ID

Status code: 400

Table 4-55 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Status code: 500

Table 4-56 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Example Request

- Example request 1

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

{
  "flavor_id" : "8f2e696c-a9c1-30bd-af90-25522bc67606",
  "node_number" : 1
}
```

- Example request 2: Scaling out a DDM instance that has more than one group

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

{
  "flavor_id" : "8f2e696c-a9c1-30bd-af90-25522bc67606",
  "group_id" : "f080abf2010d45118068c28c8958f5fcgr09",
  "node_number" : 1
}
```

- Example request 3: Scaling out a yearly/monthly instance for which parameter **is_auto_pay** is set to **true**

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

```
{
  "flavor_id" : "8f2e696c-a9c1-30bd-af90-25522bc67606",
  "node_number" : 1,
  "is_auto_pay" : true
}
```

Example Response

Status code: 200

ok

```
{
  "instanceId" : "28e8841d0b9c4f6a9a30742ee60e1068in09",
  "instanceName" : "BUG-ddm-fb88-test",
  "jobId" : "1eb697c0-1842-43a3-8671-f562d0385cb9"
}
```

Status code: 400

bad request

```
{
  "externalMessage" : "Parameter error.",
  "errCode" : "DBS.280001"
}
```

Status code: 500

server error

```
{
  "externalMessage" : "Server failure.",
  "errCode" : "DBS.200412"
}
```

Status Codes

Status Code	Description
200	ok
400	bad request
500	server error

Error Codes

For details, see **Error Codes**.

4.1.10 Scaling in a DDM instance

Function

This API is used to remove nodes from a specified DDM instance.

Constraints

Make sure that the associated RDS instances are available and not undergoing other operations. Yearly/Monthly instances do not support this operation.

URI

POST /v2/{project_id}/instances/{instance_id}/action/reduce

Table 4-57 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID of a tenant in a region
instance_id	Yes	String	DDM instance ID

Request Parameters

Table 4-58 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token It can be obtained by calling an IAM API. The value of **X-Subject-Token** in the response header is the user token.

Table 4-59 Request body parameters

Parameter	Mandatory	Type	Description
node_number	Yes	Integer	Number of the nodes to be removed. The maximum value is the instance nodes minus 1.
group_id	No	String	Group ID, which specifies the group that is scaled out. This parameter must be specified if there are more than one node group.

Response Parameters

Status code: 200

Table 4-60 Response body parameters

Parameter	Type	Description
instanceId	String	DDM instance ID
instanceName	String	DDM instance name
jobId	String	Task ID

Status code: 400

Table 4-61 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Status code: 500

Table 4-62 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Example Request

- Example request 1

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

{
  "node_number" : 2
}
```

- Example request 2 where the DDM instance has more than one group

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

{
  "group_id" : "f080abf2010d45118068c28c8958f5fcgr09",
  "node_number" : 1
}
```

Example Response

Status code: 200

ok

```
{
  "instanceId" : "28e8841d0b9c4f6a9a30742ee60e1068in09",
  "instanceName" : "BUG-ddm-fb88-test",
  "jobId" : "1eb697c0-1842-43a3-8671-f562d0385cb9"
}
```

Status code: 400

bad request

```
{
  "externalMessage" : "Parameter error.",
  "errCode" : "DBS.280001"
}
```

Status code: 500

server error

```
{
  "externalMessage" : "Server failure.",
  "errCode" : "DBS.200412"
}
```

Status Codes

Status Code	Description
200	ok
400	bad request
500	server error

Error Codes

For details, see **Error Codes**.

4.1.11 Modifying the Read Policy of the Associated DB Instance

Function

This API is used to modify the read policy of the DB instance associated with a DDM instance.

Constraints

Make sure that the associated RDS instances are available and not undergoing other operations.

URI

PUT /v2/{project_id}/instances/{instance_id}/action/read-write-strategy

Table 4-63 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID of a tenant in a region
instance_id	Yes	String	DDM instance ID

Request Parameters

Table 4-64 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token It can be obtained by calling an IAM API. The value of **X-Subject-Token** in the response header is the user token.

Table 4-65 Request body parameters

Parameter	Mandatory	Type	Description
read_weight	Yes	Object	Read weights of the primary DB instance and its read replicas

Response Parameters

Status code: 200

Table 4-66 Response body parameters

Parameter	Type	Description
success	Boolean	Whether the operation is successful
instance_id	String	DDM instance ID

Status code: 400

Table 4-67 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Status code: 500

Table 4-68 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Example Request

```
PUT https://{endpoint}/v2/{project_id}/instances/{instance_id}/action/read-write-strategy

{
  "read_weight" : {
    "395298ae6fb9496d95939ed556474983in01" : 60,
    "38ef52c365a14b7caeb7333137900e96in01" : 50
  }
}
```

Example Response

Status code: 200

ok

```
{
  "success" : true,
  "instance_id" : "175f5aff-xxxx-xxxx-xxxx-d0858982bbec"
}
```

Status code: 400

bad request

```
{
  "externalMessage" : "Parameter error.",
  "errCode" : "DBS.280001"
}
```

Status code: 500

server error

```
{
  "externalMessage" : "Server failure.",
  "errCode" : "DBS.200412"
}
```

Status Codes

Status Code	Description
200	ok
400	bad request
500	server error

Error Codes

For details, see **Error Codes**.

4.1.12 Synchronizing Data Node Information

Function

This API is used to synchronize configuration information of all data nodes that are associated with a DDM instance.

Constraints

None

URI

POST /v1/{project_id}/instances/{instance_id}/rds/sync

Table 4-69 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID of a tenant in a region
instance_id	Yes	String	DDM instance ID

Request Parameters

Table 4-70 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token It can be obtained by calling an IAM API. The value of **X-Subject-Token** in the response header is the user token.

Response Parameters

Status code: 200

Table 4-71 Response body parameters

Parameter	Type	Description
instanceId	String	DDM instance ID
jobId	String	Task ID

Status code: 400

Table 4-72 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Status code: 500

Table 4-73 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Example Request

```
POST https://{endpoint}/v1/{project_id}/instances/{instance_id}/rds/sync

None
```

Example Response

Status code: 200

OK

```
{
  "instanceId" : "1f5c9fd6cd984056ba89c8c87cc03278in09",
  "jobId" : "6f1334ca-faa6-479e-837f-f3219192675e"
}
```

Status code: 400

bad request

```
{
  "externalMessage" : "Parameter error.",
  "errCode" : "DBS.280001"
}
```

Status code: 500

server error

```
{
  "externalMessage" : "Server failure.",
  "errCode" : "DBS.200412"
}
```

Status Codes

Status Code	Description
200	OK
400	bad request
500	server error

Error Codes

For details, see **Error Codes**.

4.1.13 Querying Nodes of a DDM Instance

Function

This API is used to query nodes of a DDM instance.

Constraints

None

URI

GET /v1/{project_id}/instances/{instance_id}/nodes

Table 4-74 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID of a tenant in a region
instance_id	Yes	String	DDM instance ID

Table 4-75 Query parameters

Parameter	Mandatory	Type	Description
offset	No	Integer	Which page the server starts returning items. The start value cannot be less than **0**. The default value is **0**. Minimum value: **0**
limit	No	Integer	Number of records displayed on each page. The value is greater than **0** and not greater than **128**. The default value is **128**. Minimum value: **1** Maximum value: **128**

Request Parameters

Table 4-76 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token It can be obtained by calling an IAM API. The value of **X-Subject-Token** in the response header is the user token.

Response Parameters

Status code: 200

Table 4-77 Response body parameters

Parameter	Type	Description
nodes	Array of **NodeList** objects	Instance node information
offset	Integer	Which page the server starts returning items.
limit	Integer	Number of records displayed on each page
total	Integer	Number of DDM instance nodes

Table 4-78 NodeList

Parameter	Type	Description
port	String	Port
status	String	Node status
node_id	String	Node ID
ip	String	ip

Status code: 400

Table 4-79 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Status code: 500

Table 4-80 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Example Request

```
GET https://{endpoint}/v1/{project_id}/instances/{instance_id}/nodes
```

Example Response

Status code: 200

OK

```
{
  "nodes" : [ {
    "port" : "5066",
    "status" : "normal,",
    "node_id" : "4a2b97b7f5e3462c9c78aae93b46ed83no09,",
    "ip" : "192.168.0.160"
  } ],
  "offset" : 0,
  "limit" : 128,
  "total" : 1
}
```

Status code: 400

bad request

```
{
  "externalMessage" : "Parameter error.",
  "errCode" : "DBS.280001"
}
```

Status code: 500

server error

```
{
  "externalMessage" : "Server failure.",
  "errCode" : "DBS.200412"
}
```

Status Codes

Status Code	Description
200	OK
400	bad request
500	server error

Error Codes

For details, see **Error Codes**.

4.1.14 Querying Details of a DDM Instance Node

Function

This API is used to query details of a DDM instance node.

Constraints

None

URI

GET /v1/{project_id}/instances/{instance_id}/nodes/{node_id}

Table 4-81 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID of a tenant in a region
instance_id	Yes	String	DDM instance ID
node_id	Yes	String	DDM instance node ID

Request Parameters

Table 4-82 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token It can be obtained by calling an IAM API. The value of **X-Subject-Token** in the response header is the user token.

Response Parameters

Status code: 200

Table 4-83 Response body parameters

Parameter	Type	Description
status	String	Node status
name	String	Node name
node_id	String	Node ID
private_ip	String	Private IP address of the node
floating_ip	String	Floating IP address of the node
server_id	String	VM ID

Parameter	Type	Description
subnet_name	String	Subnet name
datavolume_id	String	Data disk ID
res_subnet_ip	String	IP address provided by the resource subnet
systemvolume_id	String	System disk ID

Status code: 400

Table 4-84 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Status code: 500

Table 4-85 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Example Request

```
GET https://{endpoint}/v1/{project_id}/instances/{instance_id}/nodes/{node_id}
```

Example Response

Status code: 200

OK

```
{
  "status" : "normal,",
  "name" : "BUG-ddm2-lixingqiao-test_node_01,",
  "node_id" : "4a2b97b7f5e3462c9c78aae93b46ed83no09,",
  "private_ip" : "192.168.0.160,",
  "floating_ip" : "100.65.78.158,",
  "server_id" : "8bd4d0bd-f63e-489a-95b6-50351f9657e6,",
  "datavolume_id" : "30ade9fb-26de-4d1f-af08-c376974b9d86,",
```

```
  "res_subnet_ip" : "172.16.15.224,",
  "systemvolume_id" : "88d7de55-f886-4929-ae7c-04d842959700"
}
```


Status code: 400

bad request


```
{
  "externalMessage" : "Parameter error.",
  "errCode" : "DBS.280001"
}
```


Status code: 500

server error


```
{
  "externalMessage" : "Server failure.",
  "errCode" : "DBS.200412"
}
```


Status Codes

Status Code	Description
200	OK
400	bad request
500	server error

Error Codes

For details, see **Error Codes**.

4.1.15 Querying Parameters of a Specified DDM Instance

Function

This API is used to query parameters of a specified DDM instance.

Constraints

None

URI

GET /v3/{project_id}/instances/{instance_id}/configurations

Table 4-86 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID of a tenant in a region
instance_id	Yes	String	DDM instance ID

Table 4-87 Query parameters

Parameter	Mandatory	Type	Description
offset	No	Integer	Which page the server starts returning items. The start value cannot be less than **0**. The default value is **0**. Minimum value: **0**
limit	No	Integer	Number of records displayed on each page. The value is greater than **0** and not greater than **128**. The default value is **128**. Minimum value: **1** Maximum value: **128**

Request Parameters

Table 4-88 Request header parameters

Parameter	Mandatory	Type	Description
X-Language	No	String	Language. The value can be **zh-cn** or **en-us**. The default value is **zh-cn**.
X-Auth-Token	Yes	String	User token It can be obtained by calling an IAM API. The value of **X-Subject-Token** in the response header is the user token.

Response Parameters

Status code: 200

Table 4-89 Response body parameters

Parameter	Type	Description
updated	String	Time when DDM instance parameters are last updated
configuration_parameter	Array of **ConfigurationParameterList** objects	Information about DDM instance parameters
offset	Integer	Which page the server starts returning items
limit	Integer	Number of records displayed on each page
total	Integer	Total collections

Table 4-90 ConfigurationParameterList

Parameter	Type	Description
name	String	Parameter name
value	String	Parameter value
need_restart	String	Whether the instance needs to be restarted
read_only	String	Whether the parameter is read-only
value_range	String	Parameter value range
data_type	String	Parameter type
description	String	Parameter description

Status code: 400

Table 4-91 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Status code: 500

Table 4-92 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Example Request

```
GET https://{endpoint}/v3/{project_id}/instances/{instance_id}/configurations
```

Example Response

Status code: 200

OK

```
{
  "updated" : "2021-11-09 03:26:52",
  "configuration_parameter" : [ {
    "name" : "temp_table_size_limit,",
    "value" : "1000000,",
    "need_restart" : "0,",
    "read_only" : "0,",
    "value_range" : "500000-2000000000,",
    "data_type" : "integer,",
    "description" : "Maximum size of the temporary table."
  } ],
  "offset" : "0,",
  "limit" : "128,",
  "total" : 22
}
```

Status code: 400

bad request

```
{
  "externalMessage" : "Parameter error.",
  "errCode" : "DBS.280001"
}
```

Status code: 500

server error

```
{
  "externalMessage" : "Server failure.",
  "errCode" : "DBS.200412"
}
```

Status Codes

Status Code	Description
200	OK
400	bad request

Status Code	Description
500	server error

Error Codes

For details, see **Error Codes**.

4.1.16 Modifying Parameters of a DDM Instance

Function

This API is used to modify parameters of a DDM instance.

Constraints

None

URI

PUT /v3/{project_id}/instances/{instance_id}/configurations

Table 4-93 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID of a tenant in a region
instance_id	Yes	String	DDM instance ID

Request Parameters

Table 4-94 Request header parameters

Parameter	Mandatory	Type	Description
X-Language	No	String	Language. The value can be **zh-cn** or **en-us**. The default value is **zh-cn**.
X-Auth-Token	Yes	String	User token It can be obtained by calling an IAM API. The value of **X-Subject-Token** in the response header is the user token.

Table 4-95 Request body parameters

Parameter	Mandatory	Type	Description
values	Yes	**values** object	Parameters that need to be modified

Table 4-96 values

Parameter	Mandatory	Type	Description
bind_table	No	String	Data association among multiple sharded tables. The optimizer processes JOIN operations at the MySQL layer based on these associations. The format is [{tb.col1,tb2.col2}, {tb.col2,tb3.col1},...].
character_set_server	No	String	DDM server's character set. To store emoticons, set both this parameter and the character set on RDS to **utf8mb4**. To modify the character set, you must change the collation of the DDM server correspondingly. Enumerated values: • **gbk** • **utf8** • **utf8mb4**
collation_server	No	String	Collation on the DDM server. To modify the collation, you must modify the server's character set correspondingly. Enumerated values: • **utf8_unicode_ci** • **utf8_bin** • **gbk_chinese_ci** • **gbk_bin** • **utf8mb4_unicode_ci** • **utf8mb4_bin**

Parameter	Mandatory	Type	Description
concurrent_execution_level	No	String	Concurrency level of scanning table shards in a logical table. **DATA_NODE**: indicates that database shards are scanned in parallel and table shards in each database shard are scanned in serial. **RDS_INSTANCE**: indicates that RDS instances are scanned in parallel and shards in each DB instance are scanned in serial. **PHY_TABLE**: indicates that all table shards are scanned in parallel. Enumerated values: • **RDS_INSTANCE** • **DATA_NODE** • **PHY_TABLE**
connection_idle_timeout	No	String	Number of seconds the server waits for activity on a connection before closing it. The value ranges from **60** to **28800**. The default value is **28800**, indicating that the server waits for 28800 seconds before closing a connection.
enable_table_recycle	No	String	Whether the table recycle bin is enabled. Enumerated values: • **OFF** • **ON**
insert_to_load_data	No	String	Whether constant values can be inserted by executing the LOAD DATA statement. Enumerated values: • **OFF** • **ON**
live_transaction_timeout_on_shutdown	No	String	Timeout limit of an in-transit transaction, in seconds. The value ranges from **0** to **100**. The default value is **1**, indicating that the server waits for 1 second before closing the connection.

Parameter	Mandatory	Type	Description
long_query_time	No	String	Minimum duration of a query to be logged as slow, in seconds. The value ranges from **0.01** to **10**. The default value is **1**, indicating that the query is considered as a slow query if its execution duration is greater than or equal to 1 second.
max_allowed_packet	No	String	Maximum size of a packet or any generated intermediate string. The packet message buffer is initialized to **net_buffer_length** bytes, but can grow up to **max_allowed_packet** bytes if required. This value is small by default, to catch large (and possibly incorrect) packets. The value must be a multiple of **1024**. The value ranges from **1024** to **1073741824**. The default value is **16777216**.
max_backend_connections	No	String	Maximum of concurrent RDS client connections allowed per DDM instance. When this parameter is set to **0** (default), the maximum concurrent connections from a DDM node to an RDS instance is: (RDS instance's maximum connections - 20)/DDM nodes. The value ranges from **0** to **10000000**.
max_connections	No	String	Concurrent connections allowed per DDM instance, which depends on the class and quantity of associated RDS instances. The default value is **20000**. The value ranges from **10** to **40000**, indicating that the maximum of concurrent connections cannot exceed 40,000.

Parameter	Mandatory	Type	Description
min_backend_connections	No	String	Minimum concurrent connections from a DDM node to an RDS instance. The default value is **10**. The value ranges from **0** to **10000000**.
not_from_pushdown	No	String	Whether the SELECT statements that do not contain any FROM clauses are pushed down. Enumerated values: • **OFF** • **ON**
seconds_behind_master	No	String	Threshold in seconds of the replication lag between a primary RDS instance to its read replica. The value ranges from **0** to **7200**. The default value is **30**, indicating that the time for data replication between the primary RDS instance and its read replicas cannot exceed 30 seconds. If the time exceeds 30 seconds, the data read requests are no longer forwarded to the read replicas.
sql_audit	No	String	Whether SQL audit is enabled. Enumerated values: • **OFF** • **ON**
sql_execute_timeout	No	String	Number of seconds to wait for a SQL statement to execute before it times out. The value ranges from **100** to **28800**. The default value is **28800**, indicating that the SQL statement times out if its execution time is greater than or equal to 28,800 seconds.
support_ddl_binlog_hint	No	String	Whether a binlog hint is added to each DDL statement. Enumerated values: • **OFF** • **ON**

Parameter	Mandatory	Type	Description
transaction_policy	No	String	Transactions supported by DDM. XA transaction, which attempts to ensure atomicity and isolation. FREE transaction, which is a best-effort commit transaction that allows data to be written to multiple shards, without impacting performance. FREE transactions do not ensure atomicity. NO_DTX transaction, which is a single-shard transaction. Enumerated values: • **XA** • **FREE** • **NO_DTX**
ultimate_optimize	No	String	Whether the SQL execution plan is optimized based on parameter values. Enumerated values: • **OFF** • **ON**

Response Parameters

Status code: 200

Table 4-97 Response body parameters

Parameter	Type	Description
nodeList	String	DDM instance nodes
needRestart	Boolean	Whether the instance needs to be restarted
jobId	String	Task ID
configId	String	Parameter group ID
configName	String	Parameter group name

Status code: 400

Table 4-98 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Status code: 500

Table 4-99 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Example Request

```
PUT https://{endpoint}/v3/{project_id}/instances/{instance_id}/configurations

{
  "values" : {
    "long_query_time" : 2
  }
}
```

Example Response

Status code: 200

OK

```
{
  "nodeList" : null,
  "needRestart" : "false",
  "jobId" : "9fe84a77-6a6b-4b03-9a3e-db910a548657",
  "configId" : null,
  "configName" : null
}
```

Status code: 400

bad request

```
{
  "externalMessage" : "Parameter error.",
  "errCode" : "DBS.280001"
}
```

Status code: 500

server error

```
{
  "externalMessage" : "Server failure.",
```

```
  "errCode" : "DBS.200412"
}
```

Status Codes

Status Code	Description
200	OK
400	bad request
500	server error

Error Codes

For details, see **Error Codes**.

4.1.17 Querying DDM Engine Information

Function

This API is used to query information about DDM engine.

Constraints

None

URI

GET /v2/{project_id}/engines

Table 4-100 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID of a tenant in a region

Table 4-101 Query parameters

Parameter	Mandatory	Type	Description
offset	No	Integer	Which page the server starts returning items. The start value cannot be less than **0**. The default value is **0**. Minimum value: **0**

Parameter	Mandatory	Type	Description
limit	No	Integer	Number of records displayed on each page. The value is greater than **0** and not greater than **128**. The default value is **128**. Minimum value: **1** Maximum value: **128**

Request Parameters

Table 4-102 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token It can be obtained by calling an IAM API. The value of **X-Subject-Token** in the response header is the user token.

Response Parameters

Status code: 200

Table 4-103 Response body parameters

Parameter	Type	Description
engineGroups	Array of **EngineGroupsInfo** objects	Information of available engines
offset	Integer	Which page the server starts returning items
limit	Integer	Number of records displayed on each page
total	Integer	Number of engine versions

Table 4-104 EngineGroupsInfo

Parameter	Type	Description
id	String	Engine ID
name	String	Engine name

Parameter	Type	Description
version	String	Engine version
supportAzs	Array of SupportAzsInfo objects	AZs

Table 4-105 SupportAzsInfo

Parameter	Type	Description
code	String	AZ code
name	String	AZ name
favored	Boolean	Whether the current AZ is supported

Status code: 400

Table 4-106 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Status code: 500

Table 4-107 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Example Request

```
GET https://{endpoint}/v2/{project_id}/engines
```

Example Response

Status code: 200

OK

```
{
  "engineGroups" : [ {
    "id" : "b6907aa2-aacb-3ac9-9782-b90b152d456c,",
    "name" : "ddm,",
    "version" : "3.0.8,",
    "supportAzs" : [ {
      "code" : "az1xahz,",
      "name" : "az1xahz,",
      "favored" : false
    }, {
      "code" : "az2xahz,",
      "name" : "az2xahz,",
      "favored" : true
    } ]
  } ],
  "offset" : "0,",
  "limit" : "128,",
  "total" : 1
}
```


Status code: 400

bad request


```
{
  "externalMessage" : "Parameter error.",
  "errCode" : "DBS.280001"
}
```


Status code: 500

server error


```
{
  "externalMessage" : "Server failure.",
  "errCode" : "DBS.200412"
}
```


Status Codes

Status Code	Description
200	OK
400	bad request
500	server error

Error Codes

For details, see **Error Codes**.

4.1.18 Querying DDM Node Classes Available in an AZ

Function

This API is used to query DDM node classes available in an AZ.

Constraints

None

URI

GET /v2/{project_id}/flavors

Table 4-108 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID of a tenant in a region

Table 4-109 Query parameters

Parameter	Mandatory	Type	Description
engine_id	Yes	String	Engine ID, which can be obtained by calling the API for querying DDM engine information.
offset	No	Integer	Which page the server starts returning items. The start value cannot be less than **0**. The default value is **0**. Minimum value: **0**
limit	No	Integer	Number of records displayed on each page. The value is greater than **0** and not greater than **128**. The default value is **128**. Minimum value: **1** Maximum value: **128**

Request Parameters

Table 4-110 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token It can be obtained by calling an IAM API. The value of **X-Subject-Token** in the response header is the user token.

Response Parameters

Status code: 200

Table 4-111 Response body parameters

Parameter	Type	Description
computeFlavorGroups	Array of **ComputeFlavorGroupsInfo** objects	Compute flavor information

Table 4-112 ComputeFlavorGroupsInfo

Parameter	Type	Description
groupType	String	Compute resource architecture type. The value can be **x86** or **ARM**.
computeFlavors	Array of **ComputeFlavors** objects	Compute flavors
offset	Integer	Which page the server starts returning items
limit	Integer	Number of records displayed on each page
total	Integer	Total number of compute flavors

Table 4-113 ComputeFlavors

Parameter	Type	Description
id	String	Flavor ID
typeCode	String	Resource type code
code	String	VM flavor types recorded in DDM
iaasCode	String	VM flavor types recorded by the IaaS layer
cpu	String	Number of CPUs
mem	String	Memory size, in GB
maxConnections	String	Maximum number of connections
serverType	String	Compute resource type

Parameter	Type	Description
architecture	String	Compute resource architecture type. The value can be **x86** or **ARM**.
azStatus	Object	AZ status
regionStatus	String	Region status
groupType	String	Compute resource architecture type. The value can be **x86** or **ARM**.
dbType	String	Engine type
extendFields	Object	Extension field for storing AZ information

Status code: 400

Table 4-114 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Status code: 500

Table 4-115 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Example Request

```
GET https://{endpoint}/v2/{project_id}/flavors?engine_id={engine_id}
```

Example Response

Status code: 200

OK

```
{
  "computeFlavorGroups" : [ {
    "offset" : "0,",
    "limit" : "128,",
```

```
  "total" : "6,",
  "groupType" : "X86,",
  "computeFlavors" : [ {
    "id" : "8f2e696c-a9c1-30bd-af90-25522bc67606,",
    "typeCode" : "hws.resource.type.ddm,",
    "code" : "ddm.c3.large.2,",
    "iaasCode" : "c3.large.2,",
    "cpu" : "2,",
    "mem" : "4,",
    "maxConnections" : "null,",
    "serverType" : "KVM,",
    "architecture" : "X86,",
    "azStatus" : {
      "az2.dc0" : "unsupported,",
      "az3xahz" : "normal,",
      "az2xahz" : "normal,",
      "az1xahz" : "normal"
    },
    "regionStatus" : "normal,",
    "dbType" : "DDM,",
    "extendFields" : {
      "azCode" : "az2.dc0,az3xahz,az2xahz,az1xahz,",
      "azDescription" : "az2.dc0,az3xahz,az2xahz,az1xahz"
    }
  } ]
 }, {
  "offset" : "0,",
  "limit" : "128,",
  "total" : "3,",
  "groupType" : "ARM,",
  "computeFlavors" : [ {
    "id" : "87a6cb8b-0c56-3e16-a91d-9b680a1f8b7f,",
    "typeCode" : "hws.resource.type.ddm,",
    "code" : "ddm.rc6.large.2,",
    "iaasCode" : "kc1.xlarge.2,",
    "cpu" : "4,",
    "mem" : "8,",
    "maxConnections" : "null,",
    "serverType" : "KVM,",
    "architecture" : "ARM,",
    "azStatus" : {
      "az2.dc0" : "unsupported,",
      "az3xahz" : "normal,",
      "az2xahz" : "normal,",
      "az1xahz" : "normal"
    },
    "regionStatus" : "normal,",
    "groupType" : "null,",
    "dbType" : "DDM,",
    "extendFields" : {
      "azCode" : "az2.dc0,az3xahz,az2xahz,az1xahz,",
      "azDescription" : "az2.dc0,az3xahz,az2xahz,az1xahz"
    }
  } ]
 } ]
}
```


Status code: 400

bad request


```
{
  "externalMessage" : "Parameter error.",
  "errCode" : "DBS.280001"
}
```


Status code: 500

server error


```
{
  "externalMessage" : "Server failure.",
  "errCode" : "DBS.200412"
}
```

Status Codes

Status Code	Description
200	OK
400	bad request
500	server error

Error Codes

For details, see **Error Codes**.

4.1.19 Changing the Node Class of a DDM Instance

Function

This API is used to change the node class of a DDM instance.

Constraints

- The node class can be changed only when the corresponding DDM instance is normal.
- The new node class cannot be the same as the original one.
- Node classes of c6s series cannot be changed to those classes of c6 series.

URI

PUT /v3/{project_id}/instances/{instance_id}/flavor

Table 4-116 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID of a tenant in a region
instance_id	Yes	String	DDM instance ID

Request Parameters

Table 4-117 Request header parameters

Parameter	Mandatory	Type	Description
x-auth-token	Yes	String	User token It can be obtained by calling an IAM API. The value of **X-Subject-Token** in the response header is the user token.

Table 4-118 Request body parameters

Parameter	Mandatory	Type	Description
spec_code	Yes	String	Resource specification code of the new node class
group_id	No	String	This parameter is not required if the instance has only one node group. Each instance has one node group by default. If you need to create multiple node groups, set this parameter to the ID of the group whose node class you want to change.
is_auto_pay	No	Boolean	Whether the order is automatically paid when you change the node class of a yearly/monthly instance. This parameter does not affect the billing mode of automatic renewal. **true**: indicates that the order is automatically paid from the account. **false**: indicates that the order is manually paid from the account. The default value is **false**.

Response Parameters

Status code: 200

Table 4-119 Response body parameters

Parameter	Type	Description
job_id	String	ID of the task of changing node class. This parameter is returned only when you change the node class of a pay-per-use instance.
order_id	String	Order ID. This parameter is returned only when you change the node class of a yearly/monthly instance.

Status code: default

Table 4-120 Response body parameters

Parameter	Type	Description
errCode	String	Error code
externalMessage	String	Error message

Example Request

```
PUT https://{endpoint}/v3/{project_id}/instances/{instance_id}/flavor

{
  "spec_code" : "ddm.c6.4xlarge.2",
  "group_id" : "3e5a9063d3b84729b0a3310fad3a0942gr09",
  "is_auto_pay" : false
}
```

Example Response

Status code: 200

ok

```
{
  "job_id" : "2x414788a5112333a02390e2eb0ea227"
}
```

Status Codes

Status Code	Description
200	ok
default	Client or server error.

Error Codes

For details, see **Error Codes**.

4.2 Schemas

4.2.1 Creating a Schema

Function

This API is used to create a schema.

Constraints

Before creating a schema, ensure that you have associated RDS instances with your DDM instance and that the RDS instances are not associated with other DDM instances.

URI

POST /v1/{project_id}/instances/{instance_id}/databases

Table 4-121 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID of a tenant in a region
instance_id	Yes	String	DDM instance ID

Request Parameters

Table 4-122 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token It can be obtained by calling an IAM API. The value of **X-Subject-Token** in the response header is the user token.

Table 4-123 Request body parameters

Parameter	Mandatory	Type	Description
databases	Yes	Array of **CreateDatabaseDetail** objects	Schema information

Table 4-124 CreateDatabaseDetail

Parameter	Mandatory	Type	Description
name	Yes	String	Schema name, which: • Can include 2 to 48 characters. • Must start with a letter. • Contains only lowercase letters, digits, and underscores (_). • Cannot contain keywords **information_schema**, **mysql,** **performance_schema**, or **sys**. Minimum length: 2 characters Maximum length: 48 characters
shard_mode	Yes	String	Sharding mode of the schema. The value can be: • **cluster**: indicates that the schema is in sharded mode. • **single**: indicates that the schema is in unsharded mode. Enumerated values: • **cluster** • **single**

Parameter	Mandatory	Type	Description
shard_number	Yes	Integer	Number of shards in the same working mode • If **shard_unit** is not empty, the value is the product of **shard_unit** multiplied by the associated RDS instances. • If **shard_unit** is left blank, the value must be greater than the number of associated RDS instances and less than or equal to the product of the associated RDS instances multiplied by 64.
shard_unit	No	Integer	Number of shards per RDS instance This parameter is optional. • The value is **1** if the schema is unsharded. • The value ranges from **1** to **64** if the schema is sharded. Minimum value: **1** Maximum value: **64**
used_rds	Yes	Array of **DatabaseInstabcesParam** objects	RDS instances associated with the schema

Table 4-125 DatabaseInstabcesParam

Parameter	Mandatory	Type	Description
id	Yes	String	ID of the RDS instance associated with the schema
adminUser	Yes	String	Username for logging in to the associated RDS instance
adminPassword	Yes	String	Password for logging in to the associated RDS instance

Response Parameters

Status code: 200

Table 4-126 Response body parameters

Parameter	Type	Description
databases	Array of **CreateDatabaseDetailResponses** objects	Schema information

Table 4-127 CreateDatabaseDetailResponses

Parameter	Type	Description
name	String	Schema name

Status code: 400

Table 4-128 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Status code: 500

Table 4-129 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Example Request

The following is an example request of creating a schema and associating it with an existing DDM account.

```
POST https://{endpoint}/v1/{project_id}/instances/{instance_id}/databases
```

```
{
  "databases" : [ {
    "name" : "mytestdb",
    "shard_mode" : "cluster",
    "shard_number" : 8,
    "shard_unit" : 8,
    "used_rds" : [ {
      "id" : "f296c394f13f48449d715bf99af07e59in01",
      "adminUser" : "root",
      "adminPassword" : "PassWord_234"
    } ]
  } ]
}
```

Example Response

Status code: 200

OK

```
{
  "databases" : [ {
    "name" : "mytestdb"
  } ]
}
```

Status code: 400

bad request

```
{
  "externalMessage" : "Parameter error.",
  "errCode" : "DBS.280001"
}
```

Status code: 500

server error

```
{
  "externalMessage" : "Server failure.",
  "errCode" : "DBS.200412"
}
```

Status Codes

Status Code	Description
200	OK
400	bad request
500	server error

Error Codes

For details, see **Error Codes**.

4.2.2 Querying Schemas

Function

This API is used to query schemas of a DDM instance.

Constraints

None

URI

GET /v1/{project_id}/instances/{instance_id}/databases

Table 4-130 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
instance_id	Yes	String	DDM instance ID

Table 4-131 Query parameters

Parameter	Mandatory	Type	Description
offset	No	Integer	Which page the server starts returning items. The start value cannot be less than **0**. The default value is **0**. Minimum value: **0**
limit	No	Integer	Number of records displayed on each page. The value is greater than **0** and not greater than **128**. The default value is **128**. Minimum value: **1** Maximum value: **128**

Request Parameters

Table 4-132 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token It can be obtained by calling an IAM API. The value of **X-Subject-Token** in the response header is the user token.

Response Parameters

Status code: 200

Table 4-133 Response body parameters

Parameter	Type	Description
databases	Array of **GetDatabaseInfo** objects	Schema information
total	Integer	Total records

Table 4-134 GetDatabaseInfo

Parameter	Type	Description
name	String	Schema name
shard_mode	String	Sharding mode of the schema • **cluster**: indicates that the schema is in sharded mode. • **single**: indicates that the schema is in unsharded mode.
shard_number	Integer	Number of shards in the same working mode
status	String	Schema status
created	String	Time when the schema is created
used_rds	Array of **GetDatabaseUsedRds** objects	RDS instances associated with the schema
shard_unit	Integer	Number of shards per RDS instance

Table 4-135 GetDatabaseUsedRds

Parameter	Type	Description
id	String	Node ID of the associated RDS instance
name	String	Name of the associated RDS instance
status	String	Status of the associated RDS instance
error_msg	String	Response message. This parameter is not returned if no abnormality occurs.

Status code: 400

Table 4-136 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Status code: 500

Table 4-137 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Example Request

```
GET https://{endpoint}/v1/{project_id}/instances/{instance_id}/databases?offset={offset}&limit={limit}
```

Example Response

Status code: 200

OK

```
{
  "databases" : [ {
    "status" : "RUNNING,",
    "created" : "1642063713625,",
    "name" : "mytestdb170,",
    "shard_mode" : "cluster,",
    "shard_number" : "8,",
    "shard_unit" : "8,",
```

```
    "used_rds" : [ {
      "id" : "c6f68fed9e74478c8679479a07d7d568in01",
      "status" : "normal",
      "name" : "rds-test"
    } ]
  } ],
  "total" : 172
}
```

Status code: 400

bad request

```
{
  "externalMessage" : "Parameter error.",
  "errCode" : "DBS.280001"
}
```

Status code: 500

server error

```
{
  "externalMessage" : "Server failure.",
  "errCode" : "DBS.200412"
}
```

Status Codes

Status Code	Description
200	OK
400	bad request
500	server error

Error Codes

For details, see **Error Codes**.

4.2.3 Querying Details of a Schema

Function

This API is used to query details about a schema.

Constraints

None

URI

GET /v1/{project_id}/instances/{instance_id}/databases/{ddm_dbname}

Table 4-138 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
instance_id	Yes	String	DDM instance ID
ddm_dbname	Yes	String	Name of the schema to be queried, which is case-insensitive

Request Parameters

Table 4-139 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token It can be obtained by calling an IAM API. The value of **X-Subject-Token** in the response header is the user token.

Response Parameters

Status code: 200

Table 4-140 Response body parameters

Parameter	Type	Description
database	**GetDatabaseResponseBean** object	Schema information

Table 4-141 GetDatabaseResponseBean

Parameter	Type	Description
name	String	Schema name
created	String	Time when the schema is created
status	String	Schema status
updated	String	Time when the DDM instance is last updated

Parameter	Type	Description
databases	Array of **GetDatabases** objects	Sharding information of the schema
shard_mode	String	Sharding mode of the schema • **cluster**: indicates that the schema is in sharded mode. • **single**: indicates that the schema is in unsharded mode.
shard_number	Integer	Number of shards in the same working mode
shard_unit	Integer	Number of shards per RDS instance
dataVips	Array of strings	IP address and port number for connecting to the schema
used_rds	Array of **GetDatabaseUsedRds** objects	Associated RDS instances

Table 4-142 GetDatabases

Parameter	Type	Description
dbslot	Integer	Number of shards
name	String	Shard name
status	String	Shard status
created	String	Time when the shard is created
updated	String	Time when the shard is last updated
id	String	ID of the RDS instance where the shard is located
idName	String	Name of the physical database

Table 4-143 GetDatabaseUsedRds

Parameter	Type	Description
id	String	Node ID of the associated RDS instance
name	String	Name of the associated RDS instance
status	String	Status of the associated RDS instance

Parameter	Type	Description
error_msg	String	Response message. This parameter is not returned if no abnormality occurs.

Status code: 400

Table 4-144 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Status code: 500

Table 4-145 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Example Request

```
GET https://{endpoint} /v1/{project_id}/instances/{instance_id}/databases/{ddm_dbname}
```

Example Response

Status code: 200

OK

```
{
  "database" : {
    "name" : "db_7567",
    "created" : 1604631243234,
    "status" : "RUNNING",
    "updated" : 1604631243234,
    "databases" : [ {
      "id" : "e70a82534a364492b795c5080e3a1591in01",
      "name" : "db_7567_0000",
      "idName" : "db_7567_0000",
      "dbslot" : 0,
      "status" : "RUNNING",
      "created" : 1604631243234,
      "updated" : 1604631243234
    }, {
      "id" : "e70a82534a364492b795c5080e3a1591in01",
```

```
    "name" : "db_7567_0001",
    "idName" : "db_7567_0001",
    "dbslot" : 1,
    "status" : "RUNNING",
    "created" : 1604631243234,
    "updated" : 1604631243234
   }, {
    "id" : "e70a82534a364492b795c5080e3a1591in01",
    "name" : "db_7567_0002",
    "idName" : "db_7567_0002",
    "dbslot" : 2,
    "status" : "RUNNING",
    "created" : 1604631243234,
    "updated" : 1604631243234
   }, {
    "id" : "e70a82534a364492b795c5080e3a1591in01",
    "name" : "db_7567_0003",
    "idName" : "db_7567_0003",
    "dbslot" : 3,
    "status" : "RUNNING",
    "created" : 1604631243234,
    "updated" : 1604631243234
   }, {
    "id" : "e70a82534a364492b795c5080e3a1591in01",
    "name" : "db_7567_0004",
    "idName" : "db_7567_0004",
    "dbslot" : 4,
    "status" : "RUNNING",
    "created" : 1604631243234,
    "updated" : 1604631243234
   }, {
    "id" : "e70a82534a364492b795c5080e3a1591in01",
    "name" : "db_7567_0005",
    "idName" : "db_7567_0005",
    "dbslot" : 5,
    "status" : "RUNNING",
    "created" : 1604631243234,
    "updated" : 1604631243234
   }, {
    "id" : "e70a82534a364492b795c5080e3a1591in01",
    "name" : "db_7567_0006",
    "idName" : "db_7567_0006",
    "dbslot" : 6,
    "status" : "RUNNING",
    "created" : 1604631243234,
    "updated" : 1604631243234
   }, {
    "id" : "e70a82534a364492b795c5080e3a1591in01",
    "name" : "db_7567_0007",
    "idName" : "db_7567_0007",
    "dbslot" : 7,
    "status" : "RUNNING",
    "created" : 1604631243234,
    "updated" : 1604631243234
   } ],
   "shard_mode" : "cluster",
   "shard_number" : 8,
   "shard_unit" : 8,
   "dataVips" : [ {
    "id" : "192.168.185.97:5066"
   } ],
   "used_rds" : [ {
    "id" : "e70a82534a364492b795c5080e3a1591in01",
    "name" : "rds-5338",
    "status" : "normal"
   } ]
  }
}
```


Status code: 400

bad request

```
{
  "externalMessage" : "Parameter error.",
  "errCode" : "DBS.280001"
}
```

Status code: 500

server error

```
{
  "externalMessage" : "Server failure.",
  "errCode" : "DBS.200412"
}
```

Status Codes

Status Code	Description
200	OK
400	bad request
500	server error

Error Codes

For details, see **Error Codes**.

4.2.4 Deleting a Schema

Function

This API is used to delete a schema to release all its resources.

Constraints

None

URI

DELETE /v1/{project_id}/instances/{instance_id}/databases/{ddm_dbname}

Table 4-146 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
instance_id	Yes	String	DDM instance ID

Parameter	Mandatory	Type	Description
ddm_dbname	Yes	String	Name of the schema to be queried, which is case-insensitive

Table 4-147 Query parameters

Parameter	Mandatory	Type	Description
delete_rds_data	No	String	Whether data stored on the associated DB instances is deleted. The value can be: • **true**: indicates that the data stored on the associated DB instances is deleted. • **false**: indicates that the data stored on the associated DB instances is not deleted. It is left blank by default. Enumerated values: • **true** • **false**

Request Parameters

Table 4-148 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token It can be obtained by calling an IAM API. The value of **X-Subject-Token** in the response header is the user token.

Response Parameters

Status code: 400

Table 4-149 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Status code: 500

Table 4-150 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Example Request

- Request to delete a schema (including the data stored on associated DB instances)
  ```
  DELETE https://{endpoint}/v1/{project_id}/instances/{instance_id}/databases/{ddm_dbname}?delete_rds_data=true
  ```
- Request to delete a schema (excluding the data stored on associated DB instances)
  ```
  DELETE https://{endpoint}/v1/{project_id}/instances/{instance_id}/databases/{ddm_dbname}?delete_rds_data=false
  ```

Example Response

Status code: 200

OK

```
{ }
```

Status code: 400

bad request

```
{
  "externalMessage" : "Parameter error.",
  "errCode" : "DBS.280001"
}
```

Status code: 500

server error

```
{
  "externalMessage" : "Server failure.",
  "errCode" : "DBS.200412"
}
```

Status Codes

Status Code	Description
200	OK
400	bad request
500	server error

Error Codes

For details, see **Error Codes**.

4.2.5 Querying DB Instances Available for Creating a Schema

Function

This API is used to query DB instances that can be used for creating a schema.

Constraints

None

URI

GET /v1/{project_id}/instances/{instance_id}/rds

Table 4-151 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID of a tenant in a region
instance_id	Yes	String	DDM instance ID

Table 4-152 Query parameters

Parameter	Mandatory	Type	Description
offset	No	Integer	Which page the server starts returning items. The start value cannot be less than **0**. The default value is **0**. Minimum value: **0**

Parameter	Mandatory	Type	Description
limit	No	Integer	Number of records displayed on each page. The value is greater than **0** and not greater than **1000**. The default value is **128**. Minimum value: **1** Maximum value: **1000**

Request Parameters

Table 4-153 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token It can be obtained by calling an IAM API. The value of **X-Subject-Token** in the response header is the user token.

Response Parameters

Status code: 200

Table 4-154 Response body parameters

Parameter	Type	Description
instances	Array of **QueryAvailableRdsList** objects	DB instances that can be used for creating a schema
offset	Integer	Which page the server starts returning items
limit	Integer	Number of records displayed on each page
total	Integer	Total collections

Table 4-155 QueryAvailableRdsList

Parameter	Type	Description
id	String	DB instance ID

Parameter	Type	Description
projectId	String	Project ID of the tenant that the DB instance belongs to
status	String	DB instance status
name	String	DB instance name
engineName	String	Engine name of the DB instance
engineSoftwareVersion	String	Engine version of the DB instance
privateIp	String	Private IP address for connecting to the DB instance
mode	String	DB instance type (primary/standby or single-node)
port	Integer	Port for connecting to the DB instance
azCode	String	AZ
timeZone	String	Time zone

Status code: 400

Table 4-156 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Status code: 500

Table 4-157 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Example Request

```
GET https://{endpoint}/v1/{project_id}/instances/{instance_id}/rds
```

Example Response

Status code: 200

OK


```
{
  "instances" : [ {
    "id" : "c6f68fed9e74478c8679479a07d7d568in01",
    "projectId" : "055d9f4ee780d4d42f96c01c1bc3c50c",
    "status" : "normal",
    "name" : "test-ddm-no-delete-test01-00",
    "engineName" : "mysql",
    "engineSoftwareVersion" : 5.7,
    "privateIp" : "192.168.23.97",
    "mode" : "Ha",
    "port" : 3306,
    "azCode" : "az1xahzaz1xahz",
    "timeZone" : "UTC+08: 00"
  }, {
    "id" : "337e2598c2a64cb5935079f85996731din01",
    "projectId" : "055d9f4ee780d4d42f96c01c1bc3c50c",
    "status" : "normal",
    "name" : "test-ddm-no-delete-test01",
    "engineName" : "mysql",
    "engineSoftwareVersion" : 5.7,
    "privateIp" : "192.168.23.221",
    "mode" : "Ha",
    "port" : 3306,
    "azCode" : "az1xahzaz1xahz",
    "timeZone" : "UTC+08: 00"
  } ],
  "offset" : 0,
  "limit" : 6,
  "total" : 2
}
```


Status code: 400

bad request


```
{
  "externalMessage" : "Parameter error.",
  "errCode" : "DBS.280001"
}
```


Status code: 500

server error


```
{
  "externalMessage" : "Server failure.",
  "errCode" : "DBS.200412"
}
```


Status Codes

Status Code	Description
200	OK
400	bad request
500	server error

Error Codes

For details, see **Error Codes**.

4.3 DDM Accounts

4.3.1 Creating a DDM Account

Function

This API is used to create a DDM account. DDM accounts are used to connect to and manage schemas. A maximum of 100 DDM accounts can be created for each DDM instance, and one DDM account can be associated with multiple schemas.

Constraints

None

URI

POST /v1/{project_id}/instances/{instance_id}/users

Table 4-158 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID of a tenant in a region
instance_id	Yes	String	DDM instance ID

Request Parameters

Table 4-159 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token It can be obtained by calling an IAM API. The value of **X-Subject-Token** in the response header is the user token.

Table 4-160 Request body parameters

Parameter	Mandatory	Type	Description
users	Yes	Array of **CreateUsersInfo** objects	DDM account information

Table 4-161 CreateUsersInfo

Parameter	Mandatory	Type	Description
name	Yes	String	Username of the DDM account, which: • Can include 1 to 32 characters. • Must start with a letter. • Can contain only letters, digits, and underscores (_). Minimum length: 1 character Maximum length: 32 characters
password	Yes	String	DDM account password
base_authority	Yes	Array of strings	Basic permissions of the DDM account. The value can be **CREATE**, **DROP**, **ALTER**, **INDEX**, **INSERT**, **DELETE**, **UPDATE**, or **SELECT**. Enumerated values: • **CREATE** • **DROP** • **ALTER** • **INDEX** • **INSERT** • **DELETE** • **UPDATE** • **SELECT**
description	No	String	Description of the DDM account, which cannot exceed 256 characters. It is left blank by default. Maximum length: 256 characters

Parameter	Mandatory	Type	Description
databases	No	Array of **CreateUsersDatabases** objects	Associated schemas. The **databases** field is optional. You can create a DDM account without associating it with any schema.

Table 4-162 CreateUsersDatabases

Parameter	Mandatory	Type	Description
name	Yes	String	Name of the associated schema

Response Parameters

Status code: 200

Table 4-163 Response body parameters

Parameter	Type	Description
users	Array of **CreateUsersDetailResponses** objects	DDM account information

Table 4-164 CreateUsersDetailResponses

Parameter	Type	Description
name	String	Username of the DDM account

Status code: 400

Table 4-165 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Status code: 500

Table 4-166 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Example Request

```
POST https://{endpoint}/v1/{project_id}/instances/{instance_id}/users

{
  "users" : [ {
    "name" : "DDMuser1",
    "password" : "Axejs@98a",
    "base_authority" : [ "CREATE", "DROP", "ALTER", "INDEX", "INSERT", "DELETE", "UPDATE", "SELECT" ],
    "description" : "",
    "databases" : [ {
      "name" : "DDMdb1"
    } ]
  } ]
}
```

Example Response

Status code: 200

OK

```
{
  "users" : [ {
    "name" : "DDMuser1"
  } ]
}
```

Status code: 400

bad request

```
{
  "externalMessage" : "Parameter error.",
  "errCode" : "DBS.280001"
}
```

Status code: 500

server error

```
{
  "externalMessage" : "Server failure.",
  "errCode" : "DBS.200412"
}
```

Status Codes

Status Code	Description
200	OK
400	bad request
500	server error

Error Codes

For details, see **Error Codes**.

4.3.2 Querying DDM Accounts

Function

This API is used to query DDM accounts.

Constraints

None

URI

GET /v1/{project_id}/instances/{instance_id}/users

Table 4-167 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
instance_id	Yes	String	DDM instance ID

Table 4-168 Query parameters

Parameter	Mandatory	Type	Description
offset	No	Integer	Which page the server starts returning items. The start value cannot be less than **0**. The default value is **0**. Minimum value: **0**

Parameter	Mandatory	Type	Description
limit	No	Integer	Number of records displayed on each page. The value is greater than **0** and not greater than **128**. The default value is **128**. Minimum value: **1** Maximum value: **128**

Request Parameters

Table 4-169 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token It can be obtained by calling an IAM API. The value of **X-Subject-Token** in the response header is the user token.

Response Parameters

Status code: 200

Table 4-170 Response body parameters

Parameter	Type	Description
users	Array of **GetUsersListDetailResponses** objects	DDM account information
page_no	Integer	Current page
page_size	Integer	Data records on the current page
total_record	Integer	Total records
total_page	Integer	Total pages

Table 4-171 GetUsersListDetailResponses

Parameter	Type	Description
name	String	Username of the DDM account
status	String	Status of the DDM account
base_authority	Array of strings	Basic permissions of the DDM account. The value can be **CREATE**, **DROP**, **ALTER**, **INDEX**, **INSERT**, **DELETE**, **UPDATE**, or **SELECT**.
extend_authority	Array of strings	Extended permissions of the DDM account. This parameter is unavailable since August 2021 and was deleted in September 2021. The value can be **fulltableDelete**, **fulltableSelect**, or **fulltableUpdate**.
description	String	Description of the DDM account
created	Long	Time when the DDM account is created
databases	Array of **GetUsersListdatabase** objects	Associated schemas

Table 4-172 GetUsersListdatabase

Parameter	Type	Description
name	String	Name of the schema associated with the DDM account
description	String	Schema description

Status code: 400

Table 4-173 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Status code: 500

Table 4-174 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Example Request

```
GET https://{endpoint}/v1/{project_id}/instances/{instance_id}/users?offset={offset}&limit={limit}
```

Example Response

Status code: 200

OK

```
{
  "users" : [ {
    "name" : "ddmtest",
    "status" : "RUNNING",
    "base_authority" : [ "SELECT" ],
    "description" : "Account",
    "created" : "2019-10-30T11:01:24+0800",
    "databases" : [ {
      "name" : "zhxtest",
      "description" : "Schema"
    } ]
  } ],
  "page_no" : 1,
  "page_size" : 10,
  "total_record" : 1,
  "total_page" : 1
}
```

Status code: 400

bad request

```
{
  "externalMessage" : "Parameter error.",
  "errCode" : "DBS.280001"
}
```

Status code: 500

server error

```
{
  "externalMessage" : "Server failure.",
  "errCode" : "DBS.200412"
}
```

Status Codes

Status Code	Description
200	OK

Status Code	Description
400	bad request
500	server error

Error Codes

For details, see **Error Codes**.

4.3.3 Modifying a DDM Account

Function

This API is used to modify the permissions and associated schemas of a DDM account.

Constraints

At least one of the following parameters must be configured: **password**, **base_authority**, **role**, **description**, and **databases**.

URI

PUT /v1/{project_id}/instances/{instance_id}/users/{username}

Table 4-175 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
instance_id	Yes	String	DDM instance ID
username	Yes	String	Username of the DDM account

Request Parameters

Table 4-176 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token It can be obtained by calling an IAM API. The value of **X-Subject-Token** in the response header is the user token.

Table 4-177 Request body parameters

Parameter	Mandatory	Type	Description
user	Yes	**UpdateUserDetailReq** object	DDM account information

Table 4-178 UpdateUserDetailReq

Parameter	Mandatory	Type	Description
base_authority	No	Array of strings	Basic permissions of the DDM account. The default value is the original account permission. The value can be **CREATE**, **DROP**, **ALTER**, **INDEX**, **INSERT**, **DELETE**, **UPDATE**, or **SELECT**. Enumerated values: • **CREATE** • **DROP** • **ALTER** • **INDEX** • **INSERT** • **DELETE** • **UPDATE** • **SELECT**
description	No	String	Description of the DDM account, which cannot exceed 256 characters. It is left blank by default. Maximum length: 256 characters
databases	No	Array of **UpdateUsersDatabases** objects	DDM account information

Table 4-179 UpdateUsersDatabases

Parameter	Mandatory	Type	Description
name	No	String	Schema name, which is case-insensitive. The **databases** and **name** fields must be both or neither specified. The parameter is left blank by default.

Response Parameters

Status code: 200

Table 4-180 Response body parameters

Parameter	Type	Description
name	String	Username of the DDM account

Status code: 400

Table 4-181 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Status code: 500

Table 4-182 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Example Request

```
PUT https://{endpoint}/v1/{project_id}/instances/{instance_id}/users/{username}

{
```

```
 "user" : {
   "base_authority" : [ "SELECT" ],
   "description" : "test11",
   "databases" : [ {
     "name" : "db_7350"
   } ]
 }
}
```

Example Response

Status code: 200

OK

```
{
  "name" : "ddmtest"
}
```

Status code: 400

bad request

```
{
  "externalMessage" : "Parameter error.",
  "errCode" : "DBS.280001"
}
```

Status code: 500

server error

```
{
  "externalMessage" : "Server failure.",
  "errCode" : "DBS.200412"
}
```

Status Codes

Status Code	Description
200	OK
400	bad request
500	server error

Error Codes

For details, see **Error Codes**.

4.3.4 Deleting a DDM Account

Function

This API is used to delete a DDM account. This operation will also disassociate the account from associated schemas if any.

Constraints

None

URI

DELETE /v1/{project_id}/instances/{instance_id}/users/{username}

Table 4-183 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
instance_id	Yes	String	DDM instance ID
username	Yes	String	Username of the DDM account to be deleted

Request Parameters

Table 4-184 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token It can be obtained by calling an IAM API. The value of **X-Subject-Token** in the response header is the user token.

Response Parameters

Status code: 200

Table 4-185 Response body parameters

Parameter	Type	Description
name	String	Username of the DDM account

Status code: 400

Table 4-186 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Status code: 500

Table 4-187 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Example Request

```
DELETE https://{endpoint}/v1/{project_id}/instances/{instance_id}/users/{username}
```

Example Response

Status code: 200

OK

```
{
  "name" : "ddmtest"
}
```

Status code: 400

bad request

```
{
  "externalMessage" : "Parameter error.",
  "errCode" : "DBS.280001"
}
```

Status code: 500

server error

```
{
  "externalMessage" : "Server failure.",
  "errCode" : "DBS.200412"
}
```

Status Codes

Status Code	Description
200	OK
400	bad request
500	server error

Error Codes

For details, see **Error Codes**.

4.3.5 Resetting the Password of a DDM Account

Function

This API is used to reset the password of a DDM account.

Constraints

None

URI

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

Table 4-188 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
instance_id	Yes	String	DDM instance ID
username	Yes	String	Username of the DDM account to be modified

Request Parameters

Table 4-189 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token It can be obtained by calling an IAM API. The value of **X-Subject-Token** in the response header is the user token.

Table 4-190 Request body parameters

Parameter	Mandatory	Type	Description
password	Yes	String	New password

Response Parameters

Status code: 200

Table 4-191 Response body parameters

Parameter	Type	Description
success	Boolean	Whether the operation is successful
instance_id	String	DDM instance ID
user_name	String	Username of the DDM account

Status code: 400

Table 4-192 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Status code: 500

Table 4-193 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Example Request

```
POST https://{endpoint}/v2/{project_id}/instances/{instance_id}/users/{username}/password

{
  "password" : "GaussTest_234"
}
```

Example Response

Status code: 200

OK

```
{
  "name" : "ddmtest"
}
```

Status code: 400

bad request

```
{
  "externalMessage" : "Parameter error.",
  "errCode" : "DBS.280001"
}
```

Status code: 500

server error

```
{
  "externalMessage" : "Server failure.",
  "errCode" : "DBS.200412"
}
```

Status Codes

Status Code	Description
200	OK
400	bad request
500	server error

Error Codes

For details, see **Error Codes**.

4.3.6 Managing the Administrator Password

Function

If it is the first time to call this API, it is used to create an administrator and reset its password for a DDM instance. Then this API can only be used to update the administrator password.

URI

PUT /v3/{project_id}/instances/{instance_id}/admin-user

Table 4-194 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID of a tenant in a region
instance_id	Yes	String	DDM instance ID

Request Parameters

Table 4-195 Request header parameters

Parameter	Mandatory	Type	Description
x-auth-token	Yes	String	User token It can be obtained by calling an IAM API. The value of **X-Subject-Token** in the response header is the user token.

Table 4-196 Request body parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Username of the administrator. The username: • Can include 1 to 32 characters. • Must start with a letter. • Can contain only letters, digits, and underscores (_).

Parameter	Mandatory	Type	Description
password	Yes	String	Password of the administrator. The password: • Can include 8 to 32 characters. • Must be a combination of uppercase letters, lowercase letters, digits, and the following special characters: ~!@#%^*-_=+? Must be a strong password to improve security and prevent security risks such as brute force cracking.

Response Parameters

Status code: default

Table 4-197 Response body parameters

Parameter	Type	Description
error_code	String	Error code
error_msg	String	Error message

Example Request

```
POST https://{endpoint}/v3/{project_id}/instances/{instance_id}/admin-user

{
  "name" : "root",
  "password" : "xxx"
}
```

Example Response

None

Status Codes

Status Code	Description
200	OK
default	Client or server error.

Error Codes

For details, see **Error Codes**.

4.3.7 Validating Password Strength

Function

This API is used to check whether an instance password is a weak password.

URI

POST /v3/{project_id}/weak-password-verification

Table 4-198 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID of a tenant in a region

Request Parameters

Table 4-199 Request header parameters

Parameter	Mandatory	Type	Description
x-auth-token	Yes	String	User token It can be obtained by calling an IAM API. The value of **X-Subject-Token** in the response header is the user token.

Table 4-200 Request body parameters

Parameter	Mandatory	Type	Description
password	Yes	String	Character string to be validated

Response Parameters

Status code: 200

Table 4-201 Response body parameters

Parameter	Type	Description
is_weak_password	Boolean	Whether the password is a weak password. The value can be: **true**: indicating that the password is a weak password. Such a password is not recommended. **false**: indicating that the password is not a weak password. Such a password is recommended.

Status code: default

Table 4-202 Response body parameters

Parameter	Type	Description
error_code	String	Error code
error_msg	String	Error message

Example Request

```
POST https://{endpoint}/v3/{project_id}/weak-password-verification

{
  "password" : "xxx"
}
```

Example Response

Status code: 200

OK

```
{
  "is_weak_password" : true
}
```

Status Codes

Status Code	Description
200	OK
default	Client or server error.

Error Codes

For details, see **Error Codes**.

4.4 Monitoring

4.4.1 Monitoring Slow Query Logs

Function

This API is used to query the SQL statements that take a long time to execute on the DDM instance within a specified time range.

Constraints

None

URI

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

Table 4-203 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
instance_id	Yes	String	DDM instance ID

Table 4-204 Query parameters

Parameter	Mandatory	Type	Description
curPage	Yes	String	Which page the server starts returning items. The start value cannot be less than **0**.
perPage	Yes	String	Number of records displayed on each page
startDate	Yes	String	Start time in UTC, accurate to milliseconds
endDate	Yes	String	End time in UTC, accurate to milliseconds The interval between the start time and the end time must be no more than 7 days.

Request Parameters

Table 4-205 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token It can be obtained by calling an IAM API. The value of **X-Subject-Token** in the response header is the user token.

Response Parameters

Status code: 200

Table 4-206 Response body parameters

Parameter	Type	Description
totalRecord	Integer	Number of slow query logs
slowLogList	Array of **SlowLogList** objects	Information about slow query logs

Table 4-207 SlowLogList

Parameter	Type	Description
users	String	Username of the DDM account for executing the slow SQL statement
database	String	Name of the schema where the slow SQL statement is executed
querySample	String	Syntax for executing the slow SQL statement
logTime	String	Time when the slow SQL statement starts to be executed
time	String	Time for a SQL statement to execute, accurate to milliseconds
shards	String	Name of the physical shard
rowsExamined	String	Number of rows affected by the SQL statements that take a long time to execute

Parameter	Type	Description
host	String	Client IP address. This IP address may involve personal data. Anonymizing the IP address data is recommended.

Status code: 400

Table 4-208 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Status code: 500

Table 4-209 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Example Request

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/slowlog?
curPage={curPage}&perPage={perPage}&startDate={startDate}&endDate={endDate}
```

Example Response

Status code: 200

OK

```
{
  "totalRecord" : 2,
  "slowLogList" : [ {
    "users" : "testddm",
    "database" : "test1",
    "querySample" : "select id, sleep(3) from test",
    "logTime" : "2021-04-26T02:40:21",
    "time" : "12002",
    "shards" : "test1_0000",
    "rowsExamined" : "4",
    "host" : "192.168.16.18"
  } ]
}
```

Status code: 400

bad request

```
{
  "externalMessage" : "Parameter error.",
  "errCode" : "DBS.280001"
}
```

Status code: 500

server error

```
{
  "externalMessage" : "Server failure.",
  "errCode" : "DBS.200412"
}
```

Status Codes

Status Code	Description
200	OK
400	bad request
500	server error

Error Codes

For details, see **Error Codes**.

4.5 Application Examples

4.5.1 Querying DDM Instances

Scenarios

This section describes how to query all DDM instances of a tenant by configuring the pagination parameter in the required API for **Querying DDM Instances**.

Procedure

Step 1 Configure parameters **limit** and **offset** to query DDM instances.

- API Information

  URI format: GET /v1/{project_id}/instances

  For details, see **Querying DDM Instances**.

  - Example request

    GET: https://*{endpoint}*/v1/743b4c0428d945316666666666666666/instances?offset=0&limit=1

    Obtain the endpoint from **Regions and Endpoints**.

The **limit** value can be adjusted based on DDM instance data.

– Example response


```
{
"instance_num":10,
"instances":
[
{
"id":"cab932b426ed4215a8d76b9d71322661in09",
"status":"RUNNING",
"name":"ddm-20-single-2u4g-1-202010231552401522260",
"created":"2020-10-23T07:52:46+0000",
"updated":"2020-10-23T07:59:56+0000",
"available_zone":"az1xahz",
"vpc_id":"9cf0f8f5-9748-4ebb-9905-bbe429182bd6",
"subnet_id":"b35a4be7-65a5-4176-bec9-7a437493c498",
"security_group_id":"9d10da6d-38cc-4cf0-8f96-c34940a3fd15",
"node_count":1,
"access_ip":"192.168.60.13",
"access_port":"5066",
"core_count":"2",
"ram_capacity":"4",
"node_status":"RUNNING",
"enterprise_project_id":"0",
"project_id":"070c071d8e80d58c2f42c0121b10cf9f",
"engine_version":"2.5.10.10222119"
}
],
"page_no":1,
"page_size":1,
"total_record":10,
"total_page":10
}
```


Step 2 Collect query results.

1. Repeat step **1** until the returned query result is empty or the returned body does not contain the **instances** field. This indicates that all DDM instances have been queried.
2. The collected DDM instances are the desired query results.

----End

4.5.2 Modifying the Name of a DDM Instance

Scenarios

This section describes how to modify the name of a DDM instance using an API. For details, see **Modifying the Name of a DDM Instance**.

Procedure

Step 1 Modify the name of a DDM instance.

- API Information

  URI format: PUT /v1/{project_id}/instances/{instance_id}/modify_name

  For details, see **Modifying the Name of a DDM Instance**.

- Example request

PUT: https://*{endpoint}*/v1/743b4c0428d945316666666666666666/instances/modify_name

{"name": "ddm-testaa"}

Obtain the endpoint from **Regions and Endpoints**.

– Example response

```
{
"name":"ddm-testaa"
}
```

----End

4.5.3 Deleting a DDM instance

Scenarios

This section describes how to delete a DDM instance by calling an API. For details, see **Deleting a DDM Instance**.

Procedure

Step 1 Delete a DDM instance and delete the data stored in the associated RDS instances.

- API Information

  URI format: DELETE /v1/{project_id}/instances/{instance_id}?delete_rds_data=true

  For details, see **Deleting a DDM Instance**.

- Example request

  DELETE: https://*{endpoint}*/v1/743b4c0428d945316666666666666666/instances/d0b008c1ee95479d8799710d9f3a4097in09?delete_rds_data=true

  Obtain the endpoint from **Regions and Endpoints**.

  – Example response

  ```
  {
     "id":"d0b008c1ee95479d8799710d9f3a4097in09"
  }
  ```

----End

Step 1 Delete a DDM instance, but do not delete the data stored in the associated RDS instances.

- API Information

  URI format: DELETE /v1/{project_id}/instances/{instance_id}?delete_rds_data=false

  For details, see **Deleting a DDM Instance**.

- Example request

  Obtain the endpoint from **Regions and Endpoints**.

  – Example response

```
{
   "id":"d0b008c1ee95479d8799710d9f3a4097in09"
}
```

----End

5 APIs (Unavailable Soon)

5.1 Monitoring the Read/Write Ratio

Function

This API is used to query reads and writes of a DDM instance in a specified time range.

NOTICE

This API will become unavailable on July 30, 2023.

Constraints

None

URI

GET /v2/{project_id}/instances/{instance_id}/read-write-ratio

Table 5-1 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
instance_id	Yes	String	DDM instance ID

Table 5-2 Query parameters

Parameter	Mandatory	Type	Description
curPage	Yes	String	Which page the server starts returning items. The start value cannot be less than **0**.
perPage	Yes	String	Number of records displayed on each page
startDate	Yes	String	Start time in UTC, accurate to milliseconds
endDate	Yes	String	End time in UTC, accurate to milliseconds The interval between the start time and the end time must be no more than 1 month.

Request Parameters

Table 5-3 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token It can be obtained by calling an IAM API. The value of **X-Subject-Token** in the response header is the user token.

Response Parameters

Status code: 200

Table 5-4 Response body parameters

Parameter	Type	Description
totalRecord	Integer	Number of read/write ratio records on DDM
readWriteRatioList	Array of **ReadWriteRatioList** objects	Read/Write ratio records on DDM

Table 5-5 ReadWriteRatioList

Parameter	Type	Description
schema	String	Schema name
table	String	Logical table name
readCount	String	Reads
writeCount	String	Writes
relationTables	String	Associated table
lastUpdated	String	Time when the read/write ratio is last updated

Status code: 400

Table 5-6 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Status code: 500

Table 5-7 Response body parameters

Parameter	Type	Description
errCode	String	Service error code
externalMessage	String	Error message

Example Request

```
GET https://{endpoint}/v2/{project_id}/instances/{instance_id}/read-write-ratio?
curPage={curPage}&perPage={perPage}&startDate={startDate}&endDate={endDate}
```

Example Response

Status code: 200

OK

```
{
  "totalRecord" : 2,
  "readWriteRatioList" : [ {
    "schema" : "xxxx",
```

```
    "table" : "xxxx",
    "readCount" : "215",
    "writeCount" : "46",
    "relationTables" : "xxxx",
    "lastUpdated" : "1619404869724"
  } ]
}
```


Status code: 400

bad request


```
{
  "externalMessage" : "Parameter error.",
  "errCode" : "DBS.280001"
}
```


Status code: 500

server error


```
{
  "externalMessage" : "Server failure.",
  "errCode" : "DBS.200412"
}
```


Status Codes

Status Code	Description
200	OK
400	bad request
500	server error

Error Codes

For details, see **Error Codes**.

6 Permissions Policies and Supported Actions

6.1 Introduction

This section describes fine-grained permissions management for your DDM instance. If your account does not need individual IAM users, skip over this section.

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

NOTE

If you want to allow or deny the access to an API, fine-grained authorization is a good choice.

An account has all of the permissions required to call all APIs. If you want to send an API request using an IAM user of the account, ensure that the IAM user has 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 queries ECSs using an API, the user must have been granted permissions that allow the **ecs:servers:list** action.

Supported Actions

DDM 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. Actions supported by policies are specific to APIs. The following are common concepts related to policies:

- Permission: A statement in a policy that allows or denies certain operations.
- APIs: REST APIs that can be called in a custom policy
- Actions: Added to a custom policy to control permissions for specific operations

- Related actions: Actions which a specific action depends on to take effect. When assigning permissions for the action to a user, you also need to assign permissions for the dependent actions.
- IAM projects or enterprise projects: Type of projects in which policies can be used to grant permissions. A policy can be applied to IAM projects, 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 Project Management Service (EPS). Policies that only contain actions supporting IAM projects can be assigned to user groups and only take effect for IAM. Such policies will not take effect if they are assigned to user groups in Enterprise Management.

6.2 Instance Management

Table 6-1 DDM instance management

Permission	API	Action	IAM Project	Enterprise Project
Buying a pay-per-use instance Buying a yearly/ monthly DDM instance	POST /v1/ {project_id}/ instances	ddm:instance:create Before buying a DDM instance, obtain the following dependent permissions: • ecs:*:get* • ecs:*:list* • vpc:vpcs:list • vpc:securityGroups: get • vpc:subnets:get • ecs:cloudServerNics :update • ecs:serverInterfaces :use • Global or region-level **vpc:ports:*** • **BSS Finance** and **BSS Operator** policies The last permission is required only when you buy a yearly/monthly DDM instance.	√	√

Permission	API	Action	IAM Project	Enterprise Project
Querying DDM Instances	GET /v1/{project_id}/instances?offset={offset}&limit={limit}	ddm:instance:list	√	√
Querying Details of a DDM Instance	GET /v1/{project_id}/instances/{instance_id}	ddm:instance:get • vpc:*:get* • vpc:*:list*	√	√
Modifying a DDM Instance	PUT /v1/{project_id}/instances/{instance_id}	ddm:instance:modify To modify a security group, the following permissions must be configured: • vpc:*:get* • vpc:*:list* • vpc:ports:update • vpc:securityGroups:get	√	√
Restarting a DDM Instance	POST /v1/{project_id}/instances/{instance_id}/action	ddm:instance:reboot	√	√
Deleting a DDM Instance	DELETE /v1/{project_id}/instances/{instance_id}?delete_rds_data=true	ddm:instance:delete vpc:ports:delete	√	√

NOTE

The check mark (√) indicates that an action takes effect. The cross mark (x) indicates that an action does not take effect.

6.3 Schema Management

Table 6-2 Schema management actions

Permission	API	Actions	IAM Project	Enterprise Project
Creating a Schema	POST /v1/{project_id}/instances/{instance_id}/databases	ddm:database:create	√	√
Querying Schemas	GET /v1/{project_id}/instances/{instance_id}/databases?offset={offset}&limit={limit}	ddm:database:list	√	√
Querying Details of a Schema	GET /v1/{project_id}/instances/{instance_id}/databases/{ddm_dbname}	ddm:database:get	√	√
Deleting a Schema	DELETE /v1/{project_id}/instances/{instance_id}/databases/{ddm_dbname}?delete_rds_data=true	ddm:database:delete	√	√

6.4 Account Management

Table 6-3 Account management permissions

Permission	API	Actions	IAM Project	Enterprise Project
Creating a DDM Account	POST /v1/{project_id}/instances/{instance_id}/users	ddm:user:create	√	√
Querying DDM Accounts	GET /v1/{project_id}/instances/{instance_id}/users?offset={offset}&limit={limit}	ddm:user:list	√	√
Modifying a DDM Account	PUT /v1/{project_id}/instances/{instance_id}/users/{username}	ddm:user:modify	√	√
Deleting a DDM Account	DELETE /v1/{project_id}/instances/{instance_id}/users/{username}	ddm:user:delete	√	√

6.5 Reloading Table Data

Table 6-4 Permissions for managing logical tables

Permission	API	Action	IAM Project	Enterprise Project
Reloading table data	POST /v1/{project_id}/instances/{instance_id}/reload-config	ddm:instance:modify	√	√

7 Appendix

7.1 Abnormal Request Results

Abnormal Response

Table 7-1 Parameter description

Parameter	Type	Description
errCode	String	Returned error code when a task submission exception occurs. For details, see **Error Codes**.
externalMessage	String	Description of the error returned when a task submission exception occurs.

Example Response

```
{
    "errCode": "DBS.300101",
    "externalMessage": "Failed to delete the schema"
}
```

7.2 Status Codes

Normal

Table 7-2 lists the status codes that may be returned.

Table 7-2 Normal status codes

Status Code	Message	Description
200	OK	The request has been processed successfully.
202	OK	The asynchronous request is submitted successfully.

Abnormal

Table 7-3 lists the status codes that may be returned.

Table 7-3 Abnormal status codes

Status Code	Message	Description
400	Bad Request	The server fails to process the request. The possible causes are as follows: • The request could not be parsed by the server due to incorrect syntax. • Request parameters are incorrect.
401	Unauthorized	Necessary credentials (for example, username and password) required for user authentication are not provided.
403	Forbidden	You are forbidden to access the page requested.
404	Not Found	The request failed because the requested resource could not be found on the server.
405	Method Not Allowed	You are not allowed to use the method specified in the request.
409	Conflict	The request could not be processed due to a conflict with the current resource status.
413	Request Entity Too Large	The requested resource exceeds the resource quota.
415	Unsupported Media Type	**ContentType** contained in the request header is neither **application** nor **json**.
500	Internal Server Error	The request is not completed due to a service error.
501	Not Implemented	The request is not completed because the server does not support the requested function.

Status Code	Message	Description
503	Service Unavailable	The request could not be processed by the server because the server is being maintained or overloaded.

7.3 Error Codes

If an error code starting with **APIGW** is returned after you call an API, rectify the fault by referring to the instructions provided in **Error Codes**.

Status Code	Error Code	Error Message	Solution
400	DBS.280001	Parameter error.	Refresh the page, modify some parameters, and try again.
400	DBS.300002	Server failure.	Refresh the page and try again later.
400	DBS.300005	Failed to process the request. Contact technical support.	-
400	DBS.300100	Failed to create the schema.	Check whether the DDM instance is available, whether the instance is in the **Running** state, whether the RDS instance is being scaled, and whether the schema name is duplicated. If the schema name is duplicated, modify the name and send the request again.
400	DBS.300101	Failed to delete the schema.	Check whether the schema has been deleted and whether the associated RDS instance exists or becomes abnormal.
400	DBS.300102	The number of RDS instances associated with the schema is invalid.	Check whether the number of RDS instances associated with the schema is valid and try again.
400	DBS.300103	The RDS instance does not exist or it is not in the same VPC as the schema.	Check whether the RDS DB instance exists and whether the DB instance is in the same VPC as the required DDM instance, and then try again.
400	DBS.300104	The sharding rule of the schema is invalid.	Ensure that the sharding rule is valid and try again.

Status Code	Error Code	Error Message	Solution
400	DBS.300106	The number of shards per RDS instance is invalid.	Check whether the number of shards per RDS instance in the request is correct. If not, modify the number and try again.
400	DBS.300107	The accounts are duplicated.	Check whether the accounts configured in parameters are duplicated and try again.
400	DBS.300108	The RDS instances are duplicated.	Check whether there are duplicate RDS instances in the request. If yes, modify them and try again.
400	DBS.300109	The schema name is invalid.	Check whether the schema name is valid. If no, modify it and try again.
400	DBS.300110	The status of the RDS instance does not allow associating with the schema.	Resolve the RDS instance status exception and try again.
400	DBS.300112	This schema name already exists.	Check whether a schema with the same name exists. If yes, modify the name and try again.
400	DBS.300113	Failed to create a physical database.	Check whether the RDS instance administrator and password are correct. If no, modify them and try again.
400	DBS.300114	The administrator or password is incorrect.	Check whether the RDS administrator and password are correct. If no, modify them and try again.
400	DBS.300115	Failed to delete the shard.	To delete the schema, check whether there are RDS instances associated with it. If the associated RDS DB instances have been deleted, click **Synchronize DB Instance Data** on the **Basic Information** page and delete the schema again.
400	DBS.300116	Failed to query available RDS instances.	Check whether the RDS service is running properly and try again.
400	DBS.300117	Failed to obtain the associated RDS instances.	Check whether the RDS service is running properly and try again.

Status Code	Error Code	Error Message	Solution
400	DBS.300118	Failed to scale out the schema.	Check input parameters in the request.
400	DBS.300120	Failed to obtain task information.	Ensure that the DDM instance is running properly and try again.
400	DBS.300121	Failed to clear source data.	Ensure that the scale-out task exists and is in the valid state and that the administrator password of the destination RDS instance is correct, and try again.
400	DBS.300122	Failed to roll back the scale-out task.	Ensure that the scale-out task exists and is in the valid state and that the administrator password of the destination RDS instance is correct, and try again.
400	DBS.300123	Failed to retry the scale-out task.	Ensure that the scaling out task exists and its status is normal, and try again.
400	DBS.300125	Failed to synchronize RDS information.	Check whether the RDS service is running properly and try again.
400	DBS.300127	The number of shards in the RDS instances associated with the schema is invalid.	Ensure that the total number of shards in the associated RDS DB instances is valid and try again.
400	DBS.300128	Failed to query schema information.	Ensure that the DDM instance is running properly and try again.
400	DBS.300129	Failed to query schema information.	Ensure that the schema exists and try again.
400	DBS.300130	Invalid schema status.	Ensure that the schema is in the **Running** state and try again.
400	DBS.300131	Failed to update the schema SQL blacklist.	Ensure that the DDM instance is running properly and try again.
400	DBS.300132	Failed to query the schema SQL blacklist.	Ensure that the DDM instance is running properly and try again.

Status Code	Error Code	Error Message	Solution
400	DBS.300133	The RDS instance is being used to scale out a schema.	Ensure that the RDS DB instance status is not **Scaling out** or **Scaling failed** and try again.
400	DBS.300134	The schema status does not allow rolling back or canceling the scale-out task.	Refresh the page and try again later.
400	DBS.300135	Failed to update the relative read weight of the RDS instance.	Ensure that the DDM instance is running properly and try again.
400	DBS.300300	Failed to create a DDM account	Ensure that the DDM instance, schema, and account exist and try again.
400	DBS.300301	Failed to update the account.	Ensure that the account and associated schema exist and that the password meets the requirements, and try again.
400	DBS.300302	Failed to delete the account.	Ensure that the DDM instance is running properly and try again.
400	DBS.300305	Invalid DDM account name.	Ensure that the account name is valid and try again.
400	DBS.300306	The account password is invalid.	Ensure that the account name and password in the request body are correct and try again.
400	DBS.300307	The account description is invalid.	Ensure that the maximum length of account description does not exceed 256 characters and try again.
400	DBS.300309	The account basic permissions are invalid.	Ensure that basic permissions configured for the account are correct and try again.
400	DBS.300310	The account extended permissions are invalid.	Ensure that extended permissions configured for the account are correct and try again.
400	DBS.300311	The account already exists.	Ensure that the account name is unique and try again.
400	DBS.300312	Failed to query the account.	Ensure that the DDM instance is running properly and try again.

Status Code	Error Code	Error Message	Solution
400	DBS.300313	Failed to configure the SQL blacklist.	Ensure that the schema exists and is in the **Running** state, and try again.
400	DBS.300314	Failed to modify the read/write policy.	Ensure that the DDM instance is running properly and try again.
400	DBS.300315	Invalid read/write relative weight.	Ensure that the weight is greater than 0 and less than 100.
400	DBS.300316	Failed to query the read/write policy.	Ensure that the DDM instance is running properly and try again.
400	DBS.300322	Failed to save instance metadata.	Provide environment information, instance ID, involved operations, and symptom description and contact technical support.
400	DBS.300323	The associated RDS instance is unavailable or this operation cannot be performed when the RDS instance is in the current state.	Ensure that the associated RDS DB instance exists and is in normal state, and try again.
400	DBS.300327	The account does not exist.	Ensure that the account exists and try again.
400	DBS.300328	Failed to reset the password.	Ensure that the account exists and try again.
400	DBS.300329	Failed to obtain the step name.	Try again or roll back the scale-out task.
400	DBS.300330	Failed to convert the string from JSON to the required format.	Provide environment information, involved operations, instance ID, and symptom description and contact technical support.
400	DBS.300331	Data migration error.	Provide environment information, involved operations, instance ID, and symptom description and contact technical support.
400	DBS.300332	The scale-out task timed out.	Try again or roll back the scale-out task.

Status Code	Error Code	Error Message	Solution
400	DBS.300333	Failed to check the time zone.	Log in to the RDS console. In the instance list, locate the source RDS instance and click its name. In the navigation pane, choose **Parameters**, search for **time_zone** in the upper right text box to check its value. Then, perform these operations on the target RDS instance, and check whether its **time_zone** value is consistent with the source instance. If not, modify it to ensure that the two values are consistent and retry.
400	DBS.300334	**sql_mode** check failed.	Log in to the RDS console. In the instance list, locate the source RDS instance and click its name. In the navigation pane, choose **Parameters**, search for **sql_mode** in the upper right text box to check its value. Then, perform these operations on the target RDS instance, and check whether its **sql_mode** value is consistent with the source instance. If not, modify it to ensure that the two values are consistent and retry.
400	DBS.300335	Failed to obtain the RDS link.	Check whether the RDS service is running properly.
400	DBS.300336	Physical database not found.	Log in to the associated RDS instance and execute **SHOW DATABASES**. Check whether there is a physical database whose name starts with the schema name, and whether the number of physical databases must be the same as the number of shards in the schema.
400	DBS.300337	Failed to check the primary key.	Check whether there are tables without primary keys. If yes, add primary keys for those tables and retry.
400	DBS.300338	Failed to disable DDL or DML.	Provide environment information, instance ID, involved operations, and symptom description and contact technical support.
400	DBS.300339	Failed to insert a scale-out subtask.	Contact technical support.

Status Code	Error Code	Error Message	Solution
400	DBS.300340	Failed to change statuses of scale-out subtasks to **Initialization**.	Contact technical support.
400	DBS.300341	Failed to issue a scale-out subtask.	Contact technical support.
400	DBS.300342	Failed to obtain scale-out subtasks.	Contact technical support.
400	DBS.300343	The volume of remaining subtask data is invalid.	Provide environment information, involved operations, and symptom description and contact technical support.
400	DBS.300344	Status check for scale-out task failed.	Contact technical support.
400	DBS.300345	Failed to obtain information about the scale-out task.	Try again.
400	DBS.300346	Failed to obtain information about the DDM process.	Try again.
400	DBS.300347	Failed to disable the link.	Try again.
400	DBS.300348	Failed to obtain subtasks during data verification.	Contact technical support.
400	DBS.300349	Failed to update the schema status during route switching.	Contact technical support.
400	DBS.300350	The required data is not found.	Provide environment information, involved operations, and symptom description and contact technical support.
400	DBS.300351	Failed to change statuses of scale-out subtasks to **Error**.	Contact technical support.

Status Code	Error Code	Error Message	Solution
400	DBS.300352	Failed to change statuses of scale-out subtasks to **Stop**.	Provide environment information, involved operations, and symptom description and contact technical support.
400	DBS.300353	Data verification failed.	Provide environment information, involved operations, and symptom description and contact technical support.
400	DBS.300354	Failed to change statuses of scale-out subtasks to **Complete**.	Provide environment information, involved operations, and symptom description and contact technical support.
400	DBS.300355	Schema creation failed.	Provide environment information, involved operations, and symptom description and contact technical support.
400	DBS.300356	Failed to connect to the RDS DB instance.	Check whether the RDS service is running properly.
400	DBS.300357	The RDS instance administrator or password is incorrect.	If the RDS instance administrator is incorrect, log in to the RDS console, locate the required RDS instance and click its name, and view the administrator on the instance details page.
400	DBS.300358	There is a schema where source data is not cleared.	Log in to the DDM console, switch to the schema management page, locate the target schema, and click **Clear** in the **Operation** column.
400	DBS.300360	The associated instance does not support this operation.	Provide environment information, involved operations, and symptom description and contact technical support.
400	DBS.300362	The scaling method is invalid.	Select either one of rebalance and reshard.
400	DBS.300363	Pre-check failed.	Provide environment information, involved operations, and symptom description and contact technical support.

Status Code	Error Code	Error Message	Solution
400	DBS.300364	Failed to check disk space of the DB instance.	Perform the following operations to scale up disk space of the DB instance: 1. Log in to the management console. 2. Hover on the left menu to display **Service List** and choose **Database** > **Relational Database Service**. 3. On the **Instance Management** page, locate the required DB instance and click its name. 4. On the **Basic Information** page, locate the **Storage Space** part. 5. Click **Scale** and try again after the scaling is complete.
400	DBS.300365	Failed to update metadata.	Retry the scale-out task.
400	DBS.300366	Failed to rename physical tables during migration.	Retry the scale-out task.
400	DBS.300367	Failed to clear source data.	Try again.
400	DBS.300368	Parameter **lower_case_table_names** of the DB instance is invalid.	Perform the following operations to scale up disk space of the DB instance: 1. Log in to the management console. 2. Hover on the left menu to display **Service List** and choose **Database** > **Relational Database Service**. 3. On the **Instance Management** page, locate the required DB instance and click its name. 4. Choose **Parameters** in the left navigation pane, search for **lower_case_table_names** and verify that its value is **1** and retry.

Status Code	Error Code	Error Message	Solution
400	DBS.300369	Failed to configure access control. Contact technical support.	View DBS-ddm-instancemanager logs and DBS-resource-manager logs and modify as follows: • If error code APIG.xxxx is returned, indicating that you have no permission to access the API, contact ELB O&M engineers to grant the user the required permission. View the URI of the API in the resource management INFO log. • If an error code containing **ELB.***xxxx* is displayed, contact ELB O&M engineers to locate the fault.
400	DBS.300370	Load balancing is not enabled for the current DDM instance. To enable it, contact technical support.	If needed, go to DBS Operation System to enable access control for the required DDM instance.
400	DBS.300371	Creating a test DDM instance is not supported. To enable this function, contact technical support.	Creating DDM instances using the engine ID of a stable version is recommended. To create a test DDM instance, go to the DBS Operation System platform to add the user to the whitelist. View the DDM instance management logs to obtain the whitelist name.
400	DBS.300372	Invalid destination DDM instance.	Create a DDM instance as the destination instance.
400	DBS.300375	Invalid **{param}**.	Change the value of **{param}** and try again.
400	DBS.300376	The time zone is invalid.	Enter a correct time zone.
400	DBS.300377	Invalid time range.	1. Ensure that the format of the time range for route switching is HH:mm:ss. 2. Ensure that the route switching start time and end time are the same day and the interval is longer than 1 hour. 3. Retry after completing the above operations.

Status Code	Error Code	Error Message	Solution
400	DBS.300379	The operation is not allowed for this type of tasks.	Enter the ID of the required schema scale-out task.
400	DBS.300400	The task is not found.	Enter a valid task ID.
400	DBS.300401	Failed to modify the automatic route switching time range.	Provide environment information, involved operations, symptom description, and tenant node logs and contact technical support.
400	DBS.300402	Failed to switch the route manually.	Provide environment information, involved operations, symptom description, and tenant node logs and contact technical support.
400	DBS.300405	The schema name length is invalid.	Ensure that the schema name contains 2 to 24 characters.
400	DBS.300406	The schema name cannot contain keyword *keyword*.	Ensure that the schema name does not contain **information_schema**, **mysql**, **performance_schema**, and **sys**.
400	DBS.300407	The schema name cannot contain uppercase letters.	Enter a schema name that only consists of lowercase letters.
400	DBS.300408	Failed to check the RDS connectivity.	1. Upgrade Agent to 2.6.1 or later. 2. Provide environment information, involved operations, symptom description and contact O&M engineers to check whether the Agent is faulty.
400	DBS.300541	Insufficient permissions.	Use an account with the required permissions.

7.4 Instance Specifications

For details about DDM instance classes, see **Table 7-4**. The classes vary depending on actual situations.

Table 7-4 Supported DDM instance specifications

Specification Code	vCPUs	Memory (GB)	Architecture
ddm.c6.2xlarge.2	8	16	X86
ddm.c6.4xlarge.2	16	32	X86
ddm.c6.8xlarge.2	32	64	X86
ddm.kc1.2xlarge.2	8	16	ARM
ddm.kc1.4xlarge.2	16	32	ARM
ddm.kc1.8xlarge.2	32	64	ARM

7.5 Obtaining a Project ID

Scenarios

When calling APIs, you need to specify the project ID in some URLs. To do so, you need to obtain the project ID first Two methods are available:

- **Obtaining the Project ID by Calling an API**
- **Obtaining a Project ID from the Console**

Obtaining the Project ID by Calling an API

You can obtain the project ID by calling the API used to query **project information based on the specified criteria**.

The API used to obtain a project ID is **GET https://{Endpoint}/v3/projects**. **{Endpoint}** is the IAM endpoint and can be obtained from **Regions and Endpoints**. For details about API authentication, see **Authentication**.

The following is an example response. **id** indicates the project ID.

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

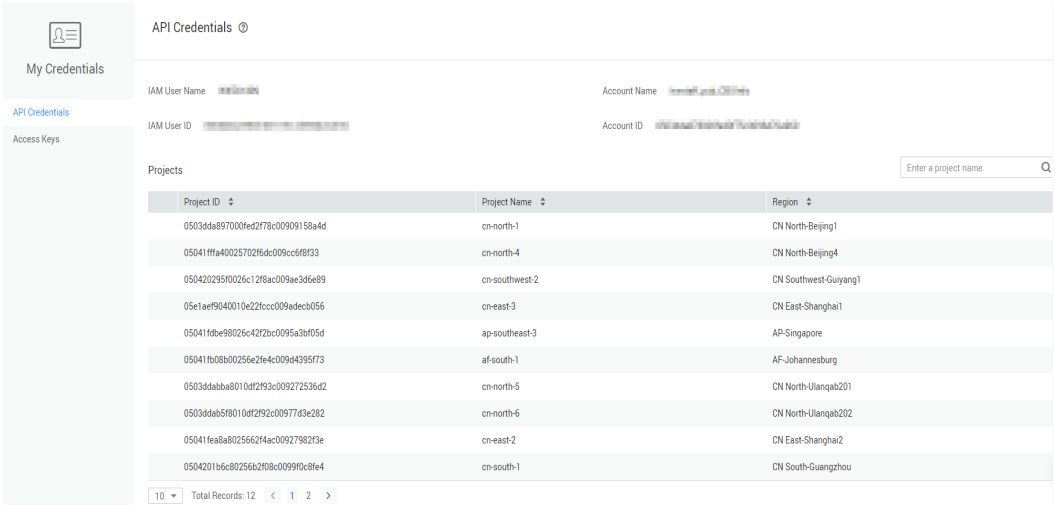
Obtaining a Project ID from the Console

Step 1 Sign up and log in to the management console.

Step 2 Hover the mouse over the username in the upper right corner and select **My Credentials** from the drop-down list.

On the **API Credentials** page, view the project ID in the project list.

Figure 7-1 Viewing project IDs



----End

7.6 Status Description

DDM Instance Statuses

Table 7-5 DDM instance statuses

Status	Description
CREATING	The DDM instance is being created.
CREATEFAILED	The DDM instance fails to be created.
RUNNING	The DDM instance is running and available.
ERROR	The DDM instance is faulty.
RESTARTING	The DDM instance is being restarted.
FREEZING	The DDM instance is being frozen.
FROZEN	The DDM instance is frozen.

Status	Description
UNFREEZING	The DDM instance is being unfrozen.
RESIZING	Class of the DDM instance fail to be changed.
UPGRADE_VERSION_INSTANCE	The version is being upgraded.
GROWING	A DDM instance is being scaled out.
REDUCING	A DDM instance is being scaled in.
data_disk_full	Disk space is full.
RESTORE	A DDM instance is being restored.
BACKUP	A DDM instance is being backed up.
ONDEMAND_TO_PERIOD	The billing mode of the DDM instance is being changed to yearly/monthly.
PERIOD_RESOURCE_SPEC_CHG	The change to the yearly/monthly DDM instance is being verified.

DDM Schema Statuses

Table 7-6 DDM schema statuses

Status	Description
CREATING	The schema is being created.
RUNNING	The schema is running and available.
CREATEFAILED	The schema fails to be created.
DELETING	The schema is being deleted.
Scaling out	The schema is being scaled out.
Scaling out failed	The schema fails to be scaled out.
Rolling back	The schema is being rolled back.

8 Change History

Table 8-1 Document change history

Released On	Description
2023-05-24	This is the twenty-eighth official release. Updated content in section "Scaling Out a DDM Instance."
2022-08-30	This is the twenty-seventh official release. • Added the description of changing the node class of a DDM instance. • Added the description of managing the administrator password. • Added the description of validating password strength.
2022-05-24	This issue is the twenty-sixth official release. Updated content in "API Overview."
2022-01-30	This issue is the twenty-fifth official release. Updated information about scaling out or in DDM instances. Updated the description of parameter **shard_unit** in section "Creating a Schema." Updated example responses.
2021-12-01	This issue is the twenty-fourth official release. Adjusted the section structure and optimized some sections.
2021-07-27	This issue is the twenty-third official release. Modified parameter information about account management.

Released On	Description
2021-07-12	This issue is the twenty-second official release. Modified query parameter table and response body parameter table in the following sections: • Querying DDM Engine Information • Querying DDM Node Classes Available in Each AZ • Querying Nodes of a DDM Instance • Querying DB Instances Available for Creating a Schema Modified the type of parameter **created** in table **GetUsersListDetailResponses**.
2021-05-10	This issue is the twenty-first official release. • Modified sections "Querying DDM Engine Information" and "Querying DDM Node Classes Available in Each AZ." • Modified sections "Scaling Out a DDM instance" and "Scaling In a DDM Instance." • Updated section "Modifying the Read Policy of the Associated DB Instance." • Added section "Synchronizing DB Instance Information." • Added section "Querying Nodes of a DDM Instance." • Added section "Querying Details of a DDM Instance." • Added section "Querying Parameters of a Specified DDM Instance." • Added section "Modifying Parameters of a DDM Instance." • Added section "Resetting the Password of an Account." • Added section "Monitoring Management."
2021-04-16	• Updated the description of parameter **name** in section "Creating a Schema." • Deleted the description of the API for scaling out a schema.
2020-03-08	This issue is the twentieth official release. • Added information about the API for scaling out a schema. • Added error codes for DDM schemas and DDM instances.

Released On	Description
2020-02-02	This issue is the nineteenth official release. Added fine-grained authorization items for creating and deleting DDM instances.
2020-12-29	This issue is the eighteenth official release. ● Optimized some table descriptions. ● Added description of DDM instance statuses.
2020-12-15	This issue is the seventeenth official release. ● Added parameter **timeZone** for buying a DDM instance. ● Modified URIs in sections "Modifying the Name of a DDM Instance", "Modifying the Security Group of a DDM Instance", and "Reloading Table Data". ● Deleted historical APIs.
2020-10-26	This issue is the sixteenth official release. Updated description of DDM2.0 parameters and added application examples.
2020-08-17	This issue is the fifteenth official release. Supplement information about DDM1.0 parameters and optimized the description of policies and supported actions.
2020-06-16	This issue is the fourteenth official release. Modified and optimized schema and DDM account description.
2020-05-10	This issue is the thirteenth official release. Added information of APIs used for DDM 2.0 Console.
2020-04-22	This issue is the twelfth official release. Added parameter **rdsmaster** for synchronizing RDS instance information.
2020-03-12	This issue is the eleventh official release. Added parameters for querying DDM instance information.
2019-12-16	This issue is the tenth official release. Added the description of querying AZs.
2019-03-31	This issue is the ninth official release. Added information about parameter **engine_version** involved in querying DDM instances.

Released On	Description
2019-03-24	This issue is the eighth official release. For query operations, added instructions for invoking APIs.
2019-01-18	This issue is the seventh official release. Optimized URLs.
2019-12-30	This issue is the sixth official release. Added the function of querying AZs for DDM and other optimization information.
2019-12-07	This issue is the fifth official release. Modified parameter information in schema management sections.
2019-10-30	This issue is the fourth official release. Added the section structure and optimized some sections.
2019-07-30	This issue is the third official release. Added error code and optimized some sections.
2019-05-31	This issue is the second official release. Added section "Permissions Policies and Supported Actions."
2019-01-30	This issue is the first official release.

