MapReduce Service

API Reference

Date **2022-11-30**

Contents

1 Before You Start 1
1.1 Overview 1
1.2 API Calling 1
1.3 Endpoints 1
1.4 Constraints 1
1.5 Concepts 2
1.6 Selecting an API Type 3
2 API Overview 4
3 Calling APIs 7
3.1 Making an API Request 7
3.2 Authentication 11
3.3 Response 13
4 Application Cases 15
4.1 Creating an MRS Cluster 15
4.2 Scaling Out a Cluster 17
4.3 Scaling in a Cluster 18
4.4 Creating a Job 19
4.5 Terminating a Job 20
4.6 Terminating a Cluster 21
5 API V2 23
5.1 Cluster Management APIs 23
5.1.1 Creating a Cluster 23
5.1.2 Changing a Cluster Name 60
5.2 Job Object APIs 61
5.2.1 Adding and Executing a Job 61
5.2.2 Querying Information About a Job 67
5.2.3 Querying a List of Jobs 71
5.2.4 Terminating a Job 78
5.2.5 Obtaining SQL Results 79
5.2.6 Deleting Jobs in Batches 81
5.3 Auto Scaling APIs 82
5.3.1 Viewing Auto Scaling Policies 82

5.4 Cluster HDFS File API ... 89
5.4.1 Obtaining the List of Files from a Specified Directory ... 89
5.5 SQL APIs ... 93
5.5.1 Submitting a SQL Statement ... 93
5.5.2 Querying SQL Results ... 96
5.5.3 Canceling a SQL Execution Task ... 99
5.6 Agency Management ... 100
5.6.1 Querying the Mapping Between a User (Group) and an IAM Agency ... 101
5.6.2 Updating the Mapping Between a User (Group) and an IAM Agency ... 103
6 API V1.1 ... 106
6.1 Cluster Management APIs ... 106
6.1.1 Creating a Cluster and Executing a Job ... 106
6.1.2 Resizing a Cluster ... 148
6.1.3 Querying a Cluster List ... 154
6.1.4 Querying Cluster Details ... 169
6.1.5 Querying a Host List ... 183
6.1.6 Terminating a Cluster ... 187
6.2 Job Object APIs ... 188
6.3 Job Execution Object APIs ... 189
6.4 Auto Scaling APIs ... 189
6.4.1 Configuring an Auto Scaling Rule ... 189
6.5 Tag Management APIs ... 197
6.5.1 Adding Tags to a Specified Cluster ... 197
6.5.2 Querying Tags of a Specified Cluster ... 199
6.5.3 Deleting Tags from a Specified Cluster ... 200
6.5.4 Adding Tags to a Cluster in Batches ... 201
6.5.5 Deleting Tags from a Cluster in Batches ... 203
6.5.6 Querying All Tags ... 205
6.5.7 Querying a List of Clusters with Specified Tags ... 206
7 Out-of-Date APIs ... 213
7.1 Job API Management (Deprecated) ... 213
7.1.1 Adding and Executing a Job (Deprecated) ... 213
7.1.2 Querying the exe Object List of Jobs (Deprecated) ... 224
7.1.3 Querying exe Object Details (Deprecated) ... 229
7.1.4 Deleting a Job Execution Object (Deprecated) ... 232
8 Permissions Policies and Supported Actions ... 234
8.1 Introduction ... 234
9 Appendix ... 237
9.1 Status Codes ... 237
9.2 Error Codes ... 241
9.3 Obtaining a Project ID ... 257

9.4 257
9.5 Obtaining the MRS Cluster Information 257
9.6 Roles and components supported by MRS 258
9.7 Change History 260

1 Before You Start

1.1 Overview

Welcome to *MapReduce Service API Reference*. MapReduce Service (MRS) provides enterprise-level big data clusters on the cloud. Tenants can fully control the clusters and easily run big data components such as Hadoop, Spark, HBase, Kafka, and Storm in the clusters.

This document describes how to use application programming interfaces (APIs) to perform operations on MRS, such as creating or deleting clusters, adjusting nodes, as well as creating and executing jobs. For details about all supported operations, see **API Overview**.

If you plan to access MRS through an API, ensure that you are familiar with MRS concepts. For details, see **Service Overview** in the *MapReduce Service User Guide*.

MRS 3.*x* does not support V1.1 APIs. You need to use V2 APIs. If you still need to use V1.1 APIs, obtain them from **Out-of-Date APIs**.

1.2 API Calling

MRS support Representational State Transfer (REST) APIs, allowing you to call APIs using HTTPS. For details about API calling, see **Calling APIs**.

1.3 Endpoints

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

1.4 Constraints

- For more constraints, see API description.

1.5 Concepts

- Account

  An account is created upon successful registration with Huawei Cloud. The account has full access permissions on all the resources and services that are subscribed under it. It can also reset user passwords and grant permissions to users. For security purposes, create IAM users and grant them permissions for routine management.

- User

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

  The account name, username, and password will be required for API authentication.

- Region

  Regions are geographic areas isolated from each other. Resources are region-specific and cannot be used across regions through internal network connections. For low network latency and quick resource access, select the nearest region.

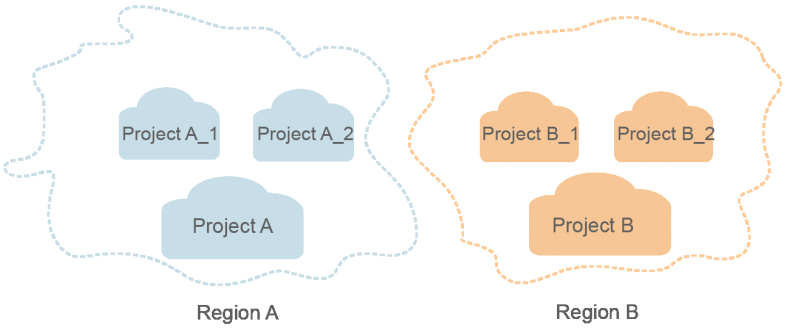
- AZ

  An AZ contains one or more physical data centers. Each AZ has independent cooling, fire extinguishing, moisture-proof, and electricity facilities. Within an AZ, computing, network, storage, and other resources are logically divided into multiple clusters. AZs within a region are interconnected using high-speed optical fibers to support cross-AZ high-availability systems.

- Project

  Projects group and isolate compute, storage, and network resources across physical regions. A default project is provided for each region, and subprojects can be created under each default project. Users can be granted permissions to access all resources in a specific project. For more refined access control, create sub-projects under a project and resources in the sub-projects. Users can then be assigned permissions to access only specific resources in the sub-projects.

  Figure 1-1 Project isolation model



- Checkpoint

  When an application consumes data, the latest SN of the consumed data is recorded as a checkpoint. When the data is reconsumed, the consumption can be continued based on this checkpoint.

- App

  Multiple applications can consume data in the same stream, and their checkpoints are distinguished by App.

1.6 Selecting an API Type

Currently, MRS provides two types (V1.1 and V2) of APIs for cloud services with customized specifications. Only part of V2 APIs is available and mainly used for submitting jobs and SQL statements. If the API functions are the same, you are advised to use the V2 APIs first.

Based on the V1.1 APIs, the V2 APIs have the following enhancements:

- Jobs can be submitted in a security cluster.
- The HiveSQL, Spark Python, and Flink jobs are supported.
- The SparkSQL and SparkScript results can be queried.

For details about the APIs and their functions, see **API Overview**.

2 API Overview

MRS provides APIs that meet RESTful API design standards, as listed in **Table 2-1** and **Table 2-2**.

Some APIs are gradually discarded during MRS version evolution. For details, see **Table 2-3**.

Table 2-1 V2 APIs

API	Function	API URI
Cluster management APIs	**Creating a Cluster**	POST /v2/{project_id}/clusters
Job object APIs	**Adding and Executing a Job**	POST /v2/{project_id}/clusters/{cluster_id}/job-executions
	Querying Information About a Job	GET /v2/{project_id}/clusters/{cluster_id}/job-executions/{job_execution_id}
	Querying a List of Jobs	GET /v2/{project_id}/clusters/{cluster_id}/job-executions
	Terminating a Job	POST /v2/{project_id}/clusters/{cluster_id}/job-executions/{job_execution_id}/kill
	Deleting Jobs in Batches	POST /v2/{project_id}/clusters/{cluster_id}/job-executions/batch-delete
	Obtaining SQL Results	GET /v2/{project_id}/clusters/{cluster_id}/job-executions/{job_execution_id}/sql-result
SQL APIs	**Submitting a SQL Statement**	POST /v2/{project_id}/clusters/{cluster_id}/sql-execution

API	Function	API URI
	Querying SQL Results	GET /v2/{project_id}/clusters/{cluster_id}/sql-execution/{sql_id}
	Canceling a SQL Execution Task	POST /v2/{project_id}/clusters/{cluster_id}/sql-execution/{sql_id}/cancel
Cluster HDFS file APIs (V2)	Obtaining the List of Files from a Specified Directory	GET /v2/{project_id}/clusters/{cluster_id}/files?path={directory}&offset={offset}&limit={limit}&sort_key={sort_key}&order={order}
Agency management APIs (V2)	Querying the Mapping Between a User (Group) and an IAM Agency	GET /v2/{project_id}/clusters/{cluster_id}/agency-mapping
	Updating the Mapping Between a User (Group) and an IAM Agency	PUT /v2/{project_id}/clusters/{cluster_id}/agency-mapping

Table 2-2 V1.1 APIs

API	Function	API URI
Cluster management APIs	Creating a Cluster and Executing a Job	POST /v1.1/{project_id}/run-job-flow
	Resizing a Cluster	PUT /v1.1/{project_id}/cluster_infos/{cluster_id}
	Querying a Cluster List	GET /v1.1/{project_id}/cluster_infos
	Querying Cluster Details	GET /v1.1/{project_id}/cluster_infos/{cluster_id}
	Terminating a Cluster	DELETE /v1.1/{project_id}/clusters/{cluster_id}
	Querying a Host List	GET /v1.1/{project_id}/clusters/{cluster_id}/hosts
Job object APIs	Adding and Executing a Job (Deprecated)	POST /v1.1/{project_id}/jobs/submit-job
	Querying the exe Object List of Jobs (Deprecated)	GET /v1.1/{project_id}/job-exes

API	Function	API URI
	Querying exe Object Details (Deprecated)	GET /v1.1/{project_id}/job-exes/{job_exe_id}
Job execution object APIs	Deleting a Job Execution Object (Deprecated)	DELETE /v1.1/{project_id}/job-executions/{job_execution_id}
Auto scaling APIs	Configuring an Auto Scaling Rule	POST /v1.1/{project_id}/autoscaling-policy/{cluster_id}
Tag management APIs	Adding Tags to a Specified Cluster	POST /v1.1/{project_id}/clusters/{cluster_id}/tags
	Deleting Tags from a Specified Cluster	DELETE /v1.1/{project_id}/clusters/{cluster_id}/tags/{key}
	Querying Tags of a Specified Cluster	GET /v1.1/{project_id}/clusters/{cluster_id}/tags
	Adding Tags to a Cluster in Batches	POST /v1.1/{project_id}/clusters/{cluster_id}/tags/action
	Querying All Tags	GET /v1.1/{project_id}/clusters/tags
	Querying a List of Clusters with Specified Tags	POST /v1.1/{project_id}/clusters/resource_instances/action

Table 2-3 Historical APIs

API	Function	API URI
V1.1 Job Management API	Adding and Executing a Job	POST /v1.1/{project_id}/jobs/submit-job
	Querying the exe Object List	GET /v1.1/{project_id}/job-exes
	Querying exe Object Details	GET /v1.1/{project_id}/job-exes/{job_exe_id}
	Deleting a Job Execution Object	DELETE /v1.1/{project_id}/job-executions/{job_execution_id}

3 Calling APIs

3.1 Making an API Request

This section describes the structure of a REST API, and uses the IAM API for as an example to demonstrate how to call an API. The obtained token is used to authenticate the calling of other APIs.

Request URI

A request URI is in the following format:

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

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

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. For details about endpoints, see **Endpoints**.
resource-path	Access path of an API for performing a specified operation. Obtain the path from the URI of an API. For example, the **resource-path** of the API used to obtain a user token is **/v3/auth/tokens**.
query-string	Query parameter, which is optional. Ensure that a question mark (?) is included before each query parameter that is in the format of "*Parameter name=Parameter value*". For example, **?limit=10** indicates that a maximum of 10 data records will be displayed.

NOTE

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

Request Methods

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

Table 3-2 HTTP methods

Method	Description
GET	Requests the server to return specified resources.
PUT	Requests the server to update specified resources.
POST	Requests the server to add resources or perform special operations.
DELETE	Requests the server to delete specified resources, for example, an object.
HEAD	Requests the server resource header.
PATCH	Requests the server to update partial content of a specified resource. If the resource does not exist, a new resource will be created.

In the URI of the API to obtain a user token, you can see that the request method is **POST**. The request is as follows:

```
POST https://{{endpoint}}/v3/auth/tokens
```

Request Header

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

Table 3-3 lists common request header fields.

Table 3-3 Common request header fields

Name	Description	Mandatory	Example
X-Sdk-Date	Time when the request is sent. The time is in *YYYYMMDD*T*HHMMSS*Z format. The value is the current Greenwich Mean Time (GMT) of the system.	This field is mandatory for AK/SK-based authentication.	20150907T101459Z
Host	Server information of the resource being requested. The value can be obtained from the URL of a service API. The value is **hostname[:port]**. If the port number is not specified, the default port is used. The default port number for HTTPS is **443**.	This field is mandatory for AK/SK-based authentication.	code.test.com or code.test.com:443
Content-Type	MIME type of the request body This field is mandatory and its default value is **application/json**. Other values of this field will be provided for specific APIs if any.	Yes	application/json
Content-Length	Length of the request body. The unit is byte.	This field is mandatory for POST and PUT requests, but must be left blank for GET requests.	3495
X-Project-Id	Project ID. This field is used to obtain the token for each project.	No	e9993fc787d94b6c886cbaa340f9c0f4

Name	Description	Mandatory	Example
X-Auth-Token	User token. It is the response to the API used to obtain a user token. This API is the only one that does not require authentication. The token is the value of **X-Subject-Token** in the response header.	No This field is mandatory for token-based authentication.	-
X-Language	Request language. Possible values: • **zh-cn**: Chinese • **en-us**: English	No	en-us
X-Domain-Id	Account ID.	No	-

NOTE

In addition to supporting token-based authentication, APIs also 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 authentication) and **X-Sdk-Date** (time when the request is sent) header fields are automatically added to the request.

For more information, see **AK/SK-based Authentication** in **Authentication**.

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

```
POST https://{{endpoint}}/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. The request body transfers content except the request header. If the request body contains Chinese characters, these characters must be coded in UTF-8.

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

In the case of the API used to obtain a user token, the request parameters and parameter description can be obtained from the API request. The following provides an example request with a body included. Replace *username*, *domainname*, ******** (login password), and *xxxxxxxxxxxxxxxxx* (project ID) with the actual values. To learn how to obtain a project ID, see **Obtaining a Project ID**.

> **NOTE**
>
> The **scope** parameter specifies where a token takes effect. You can set **scope** to an account or a project under an account. In the following example, the token takes effect only for the resources in a specified project. For more information about this API, see Obtaining a User Token of the IAM service.

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

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

All data required for the API request is available. In the response to the API used to obtain a user token, **x-subject-token** is the desired user token. This token can then be used to authenticate the calling of other APIs.

3.2 Authentication

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

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

Token-based Authentication

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

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

The token can be obtained by calling the API in Obtaining a User Token. A project-level token is required for calling this service API, that is, when calling the API for obtaining a user token, set the value of **auth.scope** in the request body to **project**.

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

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

```
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 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.

NOTICE

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

3.3 Response

Status Code

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

A status code is a group of digits, ranging from 1*xx* to 5*xx*. It indicates the status of a request. For more information, see **Status Codes**.

For the API to obtain a user token, if the status code **201** is returned after the API is called, 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 fields for the API used to obtain a user token. The **x-subject-token** header field is the desired user token. This token can then be used to authenticate the calling of other APIs.

Figure 3-1 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.

The following shows the response body for the API to obtain a user token. For the sake of space, only part of the content is displayed here.

```
{
    "token": {
```

```
        "expires_at": "2019-02-13T06:52:13.855000Z",
        "methods": [
            "password"
        ],
        "catalog": [
            {
                "endpoints": [
                    {
        "region_id": "aaa"//The region ID "aaa" is used as an example.
...
```

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

```
{
    "error_msg": "Invalid cluster name.",
    "error_code": "12000002"
}
```

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

4 Application Cases

4.1 Creating an MRS Cluster

Scenario

This section describes how to create an MRS analysis cluster using APIs. For details on how to call APIs, see **Making an API Request**.

Constraints

- A VPC and subnet have been created using the VPC service in the region where the cluster is to be created. For details about how to create a VPC, see . For details about how to create a subnet, see .
- You have obtained the region and AZ information of the cluster to be created. For details, see **Endpoints**.
- You have obtained the project ID of the region where the cluster is to be created. For details, see **Obtaining a Project ID**.
- You have determined the version of the cluster to be created and the components supported by the version.
- In this example, an analysis cluster is created.

Procedure

- API

  URI format: POST /v2/{project_id}/clusters

  For details, see **Creating a Cluster**.
- Example request

  POST: https://*{endpoint}*/v2/*{project_id}*/clusters
  - For details about **{endpoint}**, see **Endpoints**.
  - For details about **{project_id}**, see **Obtaining a Project ID**.
  - Obtain the value of **node_size** from the cluster creation page on the MRS console.

  Request body:

```
{
 "cluster_version": "MRS 3.1.0",
 "cluster_name": "mrs_Demo",
 "cluster_type": "ANALYSIS",
 "charge_info": {
    "charge_mode": "postPaid"
 },
 "region": "",
 "availability_zone": "",
 "vpc_name": "vpc-37cd",
 "subnet_name": "subnet-ed99",
 "components": "Hadoop,Spark2x,HBase,Hive,Hue,Flink,Oozie,Ranger,Tez",
 "safe_mode": "KERBEROS",
 "manager_admin_password": "Mrs@1234",
 "login_mode": "PASSWORD",
 "node_root_password": "Mrs@1234",
 "log_collection": 1,
 "mrs_ecs_default_agency": "MRS_ECS_DEFAULT_AGENCY",
 "tags": [
  {
    "key": "tag1",
    "value": "111"
  },
  {
    "key": "tag2",
    "value": "222"
  }
 ],
 "node_groups": [
  {
    "group_name": "master_node_default_group",
    "node_num": 2,
    "node_size": "rc3.4xlarge.4.linux.bigdata",
    "root_volume": {
      "type": "SAS",
      "size": 480
    },
    "data_volume": {
      "type": "SAS",
      "size": 600
    },
    "data_volume_count": 1
  },
  {
    "group_name": "core_node_analysis_group",
    "node_num": 3,
    "node_size": "rc3.4xlarge.4.linux.bigdata",
    "root_volume": {
      "type": "SAS",
      "size": 480
    },
    "data_volume": {
      "type": "SAS",
      "size": 600
    },
    "data_volume_count": 1
  },
  {
    "group_name": "task_node_analysis_group",
    "node_num": 3,
    "node_size": "rc3.4xlarge.4.linux.bigdata",
    "root_volume": {
      "type": "SAS",
      "size": 480
    },
    "data_volume": {
      "type": "SAS",
      "size": 600
    },
```

```
    "data_volume_count": 1,
    "auto_scaling_policy": {
            "auto_scaling_enable": true,
            "min_capacity": 0,
            "max_capacity": 1,
            "resources_plans": [],
            "exec_scripts": [],
            "rules": [
               {
                  "name": "default-expand-1",
                  "description": "",
                  "adjustment_type": "scale_out",
                  "cool_down_minutes": 5,
                  "scaling_adjustment": "1",
                  "trigger": {
                     "metric_id": 2003,
                     "metric_name": "StormSlotAvailablePercentage",
                     "metric_value": 100,
                     "comparison_operator_id": 2003,
                     "comparison_operator": "LTOE",
                     "evaluation_periods": "1"
                  }
               }
            ]
         }
   }
 ]
}
```

For details about the parameters, see **Creating a Cluster**.

- Example response

```
{
   "cluster_id": "da1592c2-bb7e-468d-9ac9-83246e95447a"
}
```

4.2 Scaling Out a Cluster

Scenario

After a cluster is created, add Core or Task nodes to the cluster. After an MRS cluster is created, the number of Master nodes cannot be adjusted. That is, Master nodes cannot be scaled in or out. For details on how to call APIs, see **Making an API Request**.

Constraints

- A cluster has been created and is in the **Running** state.
- You have obtained the project ID of the region where the cluster is to be created. For details, see **Obtaining a Project ID**.
- You have obtained the cluster ID, that is, the value of **cluster_id** in the command output returned after the cluster is successfully created. For details about how to obtain the cluster ID, see **Obtaining a Cluster ID**.
- This section uses the Core node as an example.

Procedure

- API

  URI format: PUT /v1.1/{project_id}/cluster_infos/{cluster_id}

  For details, see **Resizing a Cluster**.

- Example request

  PUT: https://*{endpoint}*/v1.1/*{project_id}*/cluster_infos/*{cluster_id}*

  – For details about **{endpoint}**, see **Endpoints**.

  – For details about **{project_id}**, see **Obtaining a Project ID**.

  – **{cluster_id}** indicates the value of **cluster_id** in the command output returned after the cluster is successfully created. You can also obtain the value of **cluster_id** by referring to **Obtaining a Cluster ID**.

  Request body:

```
{
   "service_id": "",
   "plan_id": "",
   "parameters": {
      "order_id": "",
      "scale_type": "scale_out",
      "node_id": "node_orderadd",
      "node_group": "core_node_default_group",
      "instances": "1",
     "skip_bootstrap_scripts":false,
     "scale_without_start":false
   },
   "previous_values": {
      "plan_id": ""
   }
}
```

  For details about the parameters, see **Resizing a Cluster**.

- Example response

```
{
   "result": "succeeded"
}
```

4.3 Scaling in a Cluster

Scenario

This section describes how to scale in a Core or Task node in the cluster after it is created. After an MRS cluster is created, the number of Master nodes cannot be adjusted. That is, Master nodes cannot be scaled in or out. For details on how to call APIs, see **Making an API Request**.

Constraints

- A cluster has been created and is in the **Running** state.
- You have obtained the project ID of the region where the cluster is to be created. For details, see **Obtaining a Project ID**.
- You have obtained the cluster ID, that is, the value of **cluster_id** in the command output returned after the cluster is successfully created. For details about how to obtain the cluster ID, see **Obtaining a Cluster ID**.
- The Core node is used as an example.

Procedure

- API

  URI format: PUT /v1.1/{project_id}/cluster_infos/{cluster_id}

For details, see **Resizing a Cluster**.

- Example request

  PUT: https://*{endpoint}*/v1.1/*{project_id}*/cluster_infos/*{cluster_id}*

  - For details about **{endpoint}**, see **Endpoints**.
  - For details about **{project_id}**, see **Obtaining a Project ID**.
  - **{cluster_id}** indicates the value of **cluster_id** in the command output returned after the cluster is successfully created. You can also obtain the value of **cluster_id** by referring to **Obtaining a Cluster ID**.

  Request body:

```
{
   "service_id": "",
   "plan_id": "",
   "parameters": {
      "order_id": "",
      "scale_type": "scale_in",
      "node_id": "node_orderadd",
      "node_group": "core_node_default_group",
      "instances": "1"


   },
   "previous_values": {
      "plan_id": ""
   }
}
```

  For details about the parameters, see **Resizing a Cluster**.

- Example response

```
{
   "result": "succeeded"
}
```

4.4 Creating a Job

Scenario

This API is used to add and submit a job in an MRS cluster. For details on how to call APIs, see **Making an API Request**.

Constraints

- A cluster has been created and is in the **Running** state.
- You have obtained the project ID of the region where the cluster is to be created. For details, see **Obtaining a Project ID**.
- You have obtained the cluster ID, that is, the value of **cluster_id** in the command output returned after the cluster is successfully created. For details about how to obtain the cluster ID, see **Obtaining a Cluster ID**.
- IAM users have been synchronized. On the Overview tab page of the cluster details page, click **Click to synchronize** on the right of **IAM User Sync** to synchronize IAM users.
- The job-related programs and input files have been stored in OBS.
- In this example, a MapReduce job is added.

Procedure

- API

  URI format: POST /v2/{project_id}/clusters/{cluster_id}/job-executions

  For details, see **Adding and Executing a Job**.

- Example request

  POST: https://*{endpoint}*/v2/*{project_id}*/clusters/*{cluster_id}*/job-executions

  - For details about **{endpoint}**, see **Endpoints**.
  - For details about **{project_id}**, see **Obtaining a Project ID**.
  - **{cluster_id}** indicates the value of **cluster_id** in the command output returned after the cluster is successfully created. You can also obtain the value of **cluster_id** by referring to **Obtaining a Cluster ID**.

  Request body:

```
{
   "job_name":"MapReduceTest",
   "job_type":"MapReduce",
   "arguments":[
      "obs://obs-test/program/hadoop-mapreduce-examples-x.x.x.jar",
      "wordcount",
      "obs://obs-test/input/",
      "obs://obs-test/job/mapreduce/output"
   ],
   "properties":{
      "fs.obs.endpoint":"obs endpoint",
      "fs.obs.access.key":"xxx",
      "fs.obs.secret.key":"yyy"
   }
}
```

  For details about the parameters, see **Adding and Executing a Job**.

- Example response

```
{
  "job_submit_result":{
     "job_id":"44b37a20-ffe8-42b1-b42b-78a5978d7e40",
     "state":"COMPLETE"
  }
}
```

4.5 Terminating a Job

Scenario

This API is used to manually terminate the job if a job is not completed after being submitted. For details on how to call APIs, see **Making an API Request**.

Constraints

- A cluster has been created and is in the **Running** state.
- You have obtained the project ID of the region where the cluster is to be created. For details, see **Obtaining a Project ID**.
- You have obtained the cluster ID, that is, the value of **cluster_id** in the command output returned after the cluster is successfully created. For details about how to obtain the cluster ID, see **Obtaining a Cluster ID**.

- You have obtained the job ID, that is, the value of **job_id** in the returned result after the job is successfully submitted. For details about how to obtain the job ID, see **Obtaining a Job ID**.
- IAM users have been synchronized. On the Overview tab page of the cluster details page, click **Click to synchronize** on the right of **IAM User Sync** to synchronize IAM users.
- The job-related programs and input files have been stored in OBS.
- In this example, a MapReduce job is added.

Procedure

- API

  URI format: POST /v2/{project_id}/clusters/{cluster_id}/job-executions/{job_execution_id}/kill

  For details, see **Terminating a Job**.
- Example request

  POST: https://*{endpoint}*/v2/*{project_id}*/clusters/*{cluster_id}*/job-executions/*{job_execution_id}*/kill

  - For details about **{endpoint}**, see **Endpoints**.
  - For details about **{project_id}**, see **Obtaining a Project ID**.
  - **{cluster_id}** indicates the value of **cluster_id** in the command output returned after the cluster is successfully created. You can also obtain the value of **cluster_id** by referring to **Obtaining a Cluster ID**.
  - **{job_execution_id}** indicates the job ID returned after the job is successfully submitted. You can also obtain the job ID by referring to **Obtaining a Job ID**.

  Request body: None
- Example response

  None

4.6 Terminating a Cluster

Scenario

This API is used to delete a cluster after data processing and analysis are completed or the cluster is abnormal.

Clusters in any of the following states cannot be terminated:

- **Scaling-out**: The cluster is being scaled out.
- **Scaling-in**: The cluster is being scaled in.
- **Starting**: The cluster is being started.
- **Terminating**: The cluster is being deleted.
- **Terminated**: The cluster has been terminated.
- **Failed**: The cluster is failed.

For details on how to call APIs, see **Making an API Request**.

Constraints

- You have obtained the project ID of the region where the cluster is to be created. For details, see **Obtaining a Project ID**.
- You have obtained the cluster ID, that is, the value of **cluster_id** in the command output returned after the cluster is successfully created. For details about how to obtain the cluster ID, see **Obtaining a Cluster ID**.

Procedure

- API

  URI format: DELETE /V1.1/{project_id}/clusters/{cluster_id}

  For details, see **Terminating a Cluster**.

- Example request

  DELETE: https://*{endpoint}*/v1.1/*{project_id}*/clusters/*{cluster_id}*

  - For details about **{endpoint}**, see **Endpoints**.
  - For details about **{project_id}**, see **Obtaining a Project ID**.
  - **{cluster_id}** indicates the value of **cluster_id** in the command output returned after the cluster is successfully created. You can also obtain the value of **cluster_id** by referring to **Obtaining a Cluster ID**.

  Request body: None

- Example response

```
{
    "result": "succeeded"
}
```

5 API V2

5.1 Cluster Management APIs

5.1.1 Creating a Cluster

Function

This API is used to create an MRS cluster.

Before using the API, you need to obtain the resources listed in **Table 5-1**.

Table 5-1 Obtaining resources

Resource	How to Obtain
VPC	See operation instructions in in the *VPC API Reference*.
Subnet	See operation instructions in in the *VPC API Reference*.
Key Pair	See operation instructions in in the *ECS API Reference*.
Zone	See **Endpoints** for details about regions and AZs.
Version	Currently, MRS 2.1.1, 3.0.5, and 3.1.0 are supported.

Resource	How to Obtain
Component	• MRS 3.1.0 supports the following components: – An analysis cluster contains the following components: Hadoop, Spark2x, HBase, Hive, Hue, Flink, Oozie, ZooKeeper, Ranger, Tez, Impala, Presto, and Kudu. – A streaming cluster contains the following components: Kafka, Flume, ZooKeeper, and Ranger. – A hybrid cluster contains the following components: Hadoop, Spark2x, HBase, Hive, Hue, Flink, Oozie, ZooKeeper, Ranger, Tez, Impala, Presto, Kudu, Kafka, and Flume. – A custom cluster contains the following components: Hadoop, Spark2x, HBase, Hive, Hue, Kafka, Flume, Flink, Oozie, ZooKeeper, Ranger, Tez, Impala, Presto, ClickHouse, and Kudu. • MRS 3.0.5 supports the following components: – An analysis cluster contains the following components: Hadoop, Spark2x, HBase, Hive, Hue, Loader, Flink, Oozie, ZooKeeper, Ranger, Tez, Impala, Presto, Kudu, and Alluxio. – A streaming cluster contains the following components: Kafka, Storm, Flume, ZooKeeper, and Ranger. – A hybrid cluster contains the following components: Hadoop, Spark2x, HBase, Hive, Hue, Loader, Flink, Oozie, ZooKeeper, Ranger, Tez, Impala, Presto, Kudu, Alluxio, Kafka, Storm, and Flume. – A custom cluster contains the following components: Hadoop, Spark2x, HBase, Hive, Hue, Loader, Kafka, Storm, Flume, Flink, Oozie, ZooKeeper, Ranger, Tez, Impala, Presto, ClickHouse, Kudu, and Alluxio.

URI

POST /v2/{project_id}/clusters

Table 5-2 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.

Request Parameters

Table 5-3 Request parameters

Parameter	Mandatory	Type	Description
cluster_version	Yes	String	Cluster version. Possible values: • MRS 3.0.5 • MRS 3.1.0
cluster_name	Yes	String	Cluster name. It must be unique. A cluster name can contain only 1 to 64 characters. Only letters, numbers, hyphens (-), and underscores (_) are allowed.
cluster_type	Yes	String	Cluster type. Possible values: • **ANALYSIS**: analysis cluster • **STREAMING**: streaming cluster • **MIXED**: hybrid cluster • **CUSTOM**: custom cluster, which is supported only by MRS 3.*x*.
region	Yes	String	Information about the region where the cluster is located. For details, see **Endpoints**.
vpc_name	Yes	String	Name of the VPC where the subnet locates. Perform the following operations to obtain the VPC name from the VPC management console: 1. Log in to the management console. 2. Click **Virtual Private Cloud** and select **Virtual Private Cloud** from the left list. On the **Virtual Private Cloud** page, obtain the VPC name from the list.

Parameter	Mandatory	Type	Description
subnet_id	No	String	Subnet ID, which can be obtained by performing the following operations on the VPC management console: 1. Log in to the VPC management console. 2. Click **Virtual Private Cloud** and select **Virtual Private Cloud** from the left list. 3. Locate the row that contains the target VPC and click the number in the **Subnets** column to view the subnet information. 4. Click the subnet name to obtain the network ID. At least one of **subnet_id** and **subnet_name** must be configured. If the two parameters are configured but do not match the same subnet, the cluster fails to create. **subnet_id** is recommended.

Parameter	Mandatory	Type	Description
subnet_name	Yes	String	Subnet name. Perform the following operations to obtain the subnet name from the VPC management console: 1. Log in to the management console. 2. Click **Virtual Private Cloud** and select **Virtual Private Cloud** from the left list. 3. Locate the row that contains the target VPC and click the number in the **Subnets** column to obtain the subnet name. At least one of **subnet_id** and **subnet_name** must be configured. If the two parameters are configured but do not match the same subnet, the cluster fails to create. If only **subnet_name** is configured and subnets with the same name exist in the VPC, the first subnet name in the VPC is used when a cluster is created. **subnet_id** is recommended.
components	Yes	String	List of component names, which are separated by commas (,). For details about the component names, see the component list of each version in **Table 5-1**.
availability_zone	Yes	String	AZ name. Multi-AZ clusters are not supported. See **Endpoints** for details about AZs.

Parameter	Mandatory	Type	Description
security_groups_id	No	String	Security group ID of the cluster. ● If this parameter is left blank, MRS automatically creates a security group, whose name starts with **mrs_{cluster_name}**. ● If this parameter is not left blank, a fixed security group is used to create a cluster. The transferred ID must be the security group ID owned by the current tenant. The security group must include an inbound rule in which all protocols and all ports are allowed and the source is the IP address of the specified node on the management plane.
safe_mode	Yes	String	Running mode of an MRS cluster. ● **SIMPLE**: normal cluster. In a normal cluster, Kerberos authentication is disabled, and users can use all functions provided by the cluster. ● **KERBEROS**: security cluster. In a security cluster, Kerberos authentication is enabled, and common users cannot use the file management and job management functions of an MRS cluster or view cluster resource usage and the job records of Hadoop and Spark. To use more cluster functions, the users must contact the Manager administrator to assign more permissions.

Parameter	Mandatory	Type	Description
manager_admin_password	Yes	String	Password of the MRS Manager administrator. The password must meet the following requirements: ● Must contain 8 to 26 characters. ● Must contain at least four of the following: uppercase letters, lowercase letters, digits, and special characters (!@$%^-_=+[{}]:,./?), but must not contain spaces. ● Cannot be the username or the username spelled backwards.
login_mode	Yes	String	Node login mode. ● **PASSWORD**: password-based login. If this value is selected, **node_root_password** cannot be left blank. ● **KEYPAIR**: specifies the key pair used for login. If this value is selected, **node_keypair_name** cannot be left blank.
node_root_password	No	String	Password of user **root** for logging in to a cluster node. A password must meet the following requirements: ● Must be 8 to 26 characters long. ● Must contain at least four of the following: uppercase letters, lowercase letters, digits, and special characters (!@$%^-_=+[{}]:,./?), but must not contain spaces. ● Cannot be the username or the username spelled backwards.
node_keypair_name	No	String	Name of a key pair You can use a key pair to log in to the Master node in the cluster.

Parameter	Mandatory	Type	Description
enterprise_project_id	No	String	Enterprise project ID. When creating a cluster, associate the enterprise project ID with the cluster. The default value is **0**, indicating the **default** enterprise project.
eip_address	No	String	An EIP bound to an MRS cluster can be used to access MRS Manager. The EIP must have been created and must be in the same region as the cluster.
eip_id	No	String	ID of the bound EIP. This parameter is mandatory when **eip_address** is configured. To obtain the EIP ID, log in to the VPC console, choose **Network > Elastic IP and Bandwidth > Elastic IP**, click the EIP to be bound, and obtain the ID in the Basic Information area.
mrs_ecs_default_agency	No	String	Name of the agency bound to a cluster node by default. The value is fixed to **MRS_ECS_DEFAULT_AGENCY**. An agency allows ECS or BMS to manage MRS resources. You can configure an agency of the ECS type to automatically obtain the AK/SK to access OBS. The **MRS_ECS_DEFAULT_AGENCY** agency has the OBS OperateAccess permission of OBS and the CES FullAccess (for users who have enabled fine-grained policies), CES Administrator, and KMS Administrator permissions in the region where the cluster is located.

Parameter	Mandatory	Type	Description
template_id	No	String	Template used for node deployment when the cluster type is CUSTOM. • **mgmt_control_combined_v2**: template for jointly deploying the management and control nodes. The management and control roles are co-deployed on the Master node, and data instances are deployed in the same node group. This deployment mode applies to scenarios where the number of control nodes is less than 100, reducing costs. • **mgmt_control_separated_v2**: The management and control roles are deployed on different master nodes, and data instances are deployed in the same node group. This deployment mode is applicable to a cluster with 100 to 500 nodes and delivers better performance in high-concurrency load scenarios. • **mgmt_control_data_separated_v2**: The management role and control role are deployed on different Master nodes, and data instances are deployed in different node groups. This deployment mode is applicable to a cluster with more than 500 nodes. Components can be deployed separately, which can be used for a larger cluster scale.
tags	No	Array of tag objects	Cluster tag For more parameter description, see **Table 5-4**. A maximum of 10 tags can be added to a cluster.

Parameter	Mandatory	Type	Description
log_collection	No	Integer	Specifies whether to collect logs when cluster creation fails: The default value is **1**, indicating that OBS buckets will be created and only used to collect logs that record MRS cluster creation failures. Possible values: • **0**: Do not collect. • **1**: Collect. The default value is **1**, indicating that OBS buckets will be created and only used to collect logs that record MRS cluster creation failures.
node_groups	Yes	Array of NodeGroupV2 objects	Information about the node groups in the cluster. For details about the parameters, see **Table 5-5**.
bootstrap_scripts	No	Array of bootstrap_script objects	Bootstrap action script information. For more parameter description, see **Table 5-7**. MRS 3.x does not support this parameter.
add_jobs	No	Array of add_jobs objects	Jobs can be submitted when a cluster is created. Currently, only one job can be created. For details about job parameters, see **Table 5-8**. MRS 3.x does not support this parameter.

Table 5-4 Tag parameters

Parameter	Mandatory	Type	Description
key	Yes	String	Tag key. • It contains a maximum of 36 Unicode characters and cannot be an empty string. • The tag **key** cannot start or end with spaces or contain non-printable ASCII characters (0–31) and the following special characters: =*<>\,|/ • The tag key of a resource must be unique.
value	Yes	String	Tag value. • The value can contain 0 to 43 unicode characters that can be blank. • The value cannot start or end with spaces or contain non-printable ASCII characters (0–31) and the following special characters: =*<>\,|/

Table 5-5 NodeGroup parameters

Parameter	Mandatory	Type	Description
group_name	Yes	String	Node group name. The value can contain a maximum of 64 characters, including uppercase and lowercase letters, digits and underscores (_). The rules for configuring node groups are as follows: • **master_node_default_group**: Master node group, which must be included in all cluster types. • **core_node_analysis_group**: analysis Core node group, which must be contained in the analysis cluster and hybrid cluster. • **core_node_streaming_group**: indicates the streaming Core node group, which must be included in both streaming and hybrid clusters. • **task_node_analysis_group**: Analysis Task node group. This node group can be selected for analysis clusters and hybrid clusters as required. • **task_node_streaming_group**: streaming Task node group. This node group can be selected for streaming clusters and hybrid clusters as required. • **node_group{x}**: node group of the customized cluster. You can add multiple node groups as required. A maximum of nine node groups can be added.
node_num	Yes	Integer	Number of nodes. The value ranges from 0 to 500. The maximum number of Core and Task nodes is 500.

Parameter	Mandatory	Type	Description
node_size	Yes	String	Instance specifications of a node. MRS supports host specifications determined by CPU, memory, and disk space. You are advised to obtain the value of this parameter from the cluster creation page on the MRS console.
root_volume	No	Volume object	System disk information of the node. This parameter is optional for some VMs or the system disk of the BMS. This parameter is mandatory in other cases. For details about the parameter description, see **Table 5-6**.
data_volume	No	Volume object	Data disk information. This parameter is mandatory when **data_volume_count** is not **0**.
data_volume_count	No	Integer	Number of data disks of a node. Value range: 0 to 10
auto_scaling_policy	No	auto_scaling_policy object	Autoscaling rule corresponding to the node group. For details about the parameters, see **Table 5-9**.

Parameter	Mandatory	Type	Description
assigned_roles	No	Array of strings	This parameter is mandatory when the cluster type is CUSTOM. You can specify the roles deployed in a node group. This parameter is a character string array. Each character string represents a role expression. Role expression definition: • If the role is deployed on all nodes in the node group, set this parameter to *<role name>*, for example, **DataNode**. • If the role is deployed on a specified subscript node in the node group: *<role name>:<index1>,<index2>..., <indexN>*, for example, **NameNode:1,2**. The subscript starts from 1. • Some roles support multi-instance deployment (that is, multiple instances of the same role are deployed on a node): *<role name>[<instance count>]*, for example, **EsNode[9]**. For details about available roles, see **Roles and components supported by MRS**.

Table 5-6 Volume parameters

Parameter	Mandatory	Type	Description
type	Yes	String	Disk type. The following disk types are supported: • **SATA**: common I/O disk • **SAS**: high I/O disk • **SSD**: ultra-high I/O disk
size	Yes	Integer	Specifies the data disk size, in GB. The value range is **10** to **32768**.

Table 5-7 bootstrap_script parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Name of a bootstrap action script. It must be unique in a cluster. The value can contain only digits, letters, spaces, hyphens (-), and underscores (_) and must not start with a space. The value can contain 1 to 64 characters.
uri	Yes	String	Path of a bootstrap action script. Set this parameter to an OBS bucket path or a local VM path. • OBS bucket path: Enter a script path manually. For example, enter the path of the public sample script provided by MRS. Example: **s3a://bootstrap/presto/presto-install.sh**. If **dualroles** is installed, the parameter of the **presto-install.sh** script is **dualroles**. If **worker** is installed, the parameter of the **presto-install.sh** script is **worker**. Based on the Presto usage habit, you are advised to install **dualroles** on the active Master nodes and **worker** on the Core nodes. • Local VM path: Enter a script path. The script path must start with a slash (/) and end with **.sh**.
parameters	No	String	Bootstrap action script parameters.
nodes	Yes	Array of strings	Name of a node group where the bootstrap action script is executed. The value can be **Master**, **Core**, or **Task**.
active_master	No	Boolean	Whether the bootstrap action script runs only on active Master nodes. The default value is **false**, indicating that the bootstrap action script can run on all Master nodes.

Parameter	Mandatory	Type	Description
before_component_start	No	Boolean	Time when the bootstrap action script is executed. Currently, the following two options are available: **Before component start** and **After component start** The default value is **false**, indicating that the bootstrap action script is executed after the component is started.
fail_action	Yes	String	Whether to continue executing subsequent scripts and creating a cluster after the bootstrap action script fails to be executed. ● **continue**: Continue to execute subsequent scripts. ● **errorout**: Stop the action. The default value is **errorout**, indicating that the action is stopped. **NOTE** You are advised to set this parameter to **continue** in the commissioning phase so that the cluster can continue to be installed and started no matter whether the bootstrap action is successful.
action_stages	No	Array of strings	Select the time when the bootstrap action script is executed. ● **BEFORE_COMPONENT_FIRST_START**: before initial component starts ● **AFTER_COMPONENT_FIRST_START**: after initial component starts ● **BEFORE_SCALE_IN**: before scale-in ● **AFTER_SCALE_IN**: after scale-in ● **BEFORE_SCALE_OUT**: before scale-out ● **AFTER_SCALE_OUT**: after scale-out

Table 5-8 add_jobs parameters

Parameter	Mandatory	Type	Description
job_type	Yes	Integer	Job type code. ● 1: MapReduce ● 2: Spark ● 3: Hive Script ● 4: HiveQL (not supported currently) ● 5: DistCp, importing and exporting data (not supported currently) ● 6: Spark Script ● 7: Spark SQL, submitting Spark SQL statements (not supported currently).
job_name	Yes	String	Job name. It contains 1 to 64 characters. Only letters, digits, hyphens (-), and underscores (_) are allowed. **NOTE** Identical job names are allowed but not recommended.
jar_path	No	String	Path of the JAR or SQL file for program execution. The parameter must meet the following requirements: ● Contains a maximum of 1023 characters, excluding special characters such as ;|&><'$. The parameter value cannot be empty or full of spaces. ● Files can be stored in HDFS or OBS. The path varies depending on the file system. – OBS: The path must start with **s3a://**. Files or programs encrypted by KMS are not supported. – HDFS: The path starts with a slash (**/**). ● Spark Script must end with **.sql** while MapReduce and Spark Jar must end with **.jar**. **sql** and **jar** are case-insensitive.

Parameter	Mandatory	Type	Description
arguments	No	String	Key parameter for program execution. The parameter is specified by the function of the user's program. MRS is only responsible for loading the parameter. The parameter contains a maximum of 2047 characters, excluding special characters such as ;|&>'<$, and can be left blank.
input	No	String	Address for inputting data. Files can be stored in HDFS or OBS. The path varies depending on the file system. ● OBS: The path must start with **s3a://**. Files or programs encrypted by KMS are not supported. ● HDFS: The path starts with a slash (**/**). The parameter contains a maximum of 1023 characters, excluding special characters such as ;|&>'<$, and can be left blank.
output	No	String	Address for outputting data. Files can be stored in HDFS or OBS. The path varies depending on the file system. ● OBS: The path must start with **s3a://**. ● HDFS: The path starts with a slash (**/**). If the specified path does not exist, the system will automatically create it. The parameter contains a maximum of 1023 characters, excluding special characters such as ;|&>'<$, and can be left blank.

Parameter	Mandatory	Type	Description
job_log	No	String	Path for storing job logs that record job running status. Files can be stored in HDFS or OBS. The path varies depending on the file system. ● OBS: The path must start with **s3a://**. ● HDFS: The path starts with a slash (**/**). The parameter contains a maximum of 1023 characters, excluding special characters such as ;|&>'<$, and can be left blank.
shutdown_cluster	No	Bool	Whether to delete the cluster after the job execution is complete. ● **true**: Yes ● **false**: No
file_action	No	String	Data import and export. ● import ● export
submit_job_once_cluster_run	Yes	Bool	● **true**: Submit a job during cluster creation. ● **false**: Submit a job after the cluster is created. Set this parameter to **true** in this example.
hql	No	String	HiveQL statement

Parameter	Mandatory	Type	Description
hive_script_path	Yes	String	SQL program path. This parameter is needed by Spark Script and Hive Script jobs only, and must meet the following requirements: • Contains a maximum of 1023 characters, excluding special characters such as ;|&><'$. The address cannot be empty or full of spaces. • Files can be stored in HDFS or OBS. The path varies depending on the file system. – OBS: The path must start with **s3a://**. Files or programs encrypted by KMS are not supported. – HDFS: The path starts with a slash (**/**). • Ends with **.sql**. **sql** is case-insensitive.

Table 5-9 auto_scaling_policy parameters

Parameter	Mandatory	Type	Description
auto_scaling_enable	Yes	Boolean	Whether to enable the auto scaling rule.
min_capacity	Yes	Integer	Minimum number of nodes left in the node group. Value range: 0 to 500
max_capacity	Yes	Integer	Maximum number of nodes in the node group. Value range: 0 to 500
resources_plans	No	Array of resources_plan objects	Resource plan list. For details, see **Table 5-10**. If this parameter is left blank, the resource plan is disabled. When auto scaling is enabled, either a resource plan or an auto scaling rule must be configured.

Parameter	Mandatory	Type	Description
exec_scripts	No	Array of scale_script objects	List of custom scaling automation scripts. For details, see **Table 5-11**. If this parameter is left blank, a hook script is disabled.
rules	No	Array of rules objects	List of auto scaling rules. For details, see **Table 5-12**. When auto scaling is enabled, either a resource plan or an auto scaling rule must be configured.

Table 5-10 resources_plan parameters

Parameter	Mandatory	Type	Description
period_type	Yes	String	Cycle type of a resource plan. Currently, only the following cycle type is supported: • daily
start_time	Yes	String	Start time of a resource plan. The value is in the format of **hour:minute**, indicating that the time ranges from 0:00 to 23:59.
end_time	Yes	String	End time of a resource plan. The value is in the same format as that of **start_time**. The interval between **end_time** and **start_time** must be greater than or equal to 30 minutes.
min_capacity	Yes	Integer	Minimum number of the preserved nodes in a node group in a resource plan. Value range: 0 to 500
max_capacity	Yes	Integer	Maximum number of the preserved nodes in a node group in a resource plan. Value range: 0 to 500

Table 5-11 scale_script parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Name of a custom automation script. It must be unique in a same cluster. The value can contain only digits, letters, spaces, hyphens (-), and underscores (_) and must not start with a space. The value can contain 1 to 64 characters.
uri	Yes	String	Path of a custom automation script. Set this parameter to an OBS bucket path or a local VM path. ● OBS bucket path: Enter a script path manually. for example, **s3a://*XXX*/scale.sh**. ● Local VM path: Enter a script path. The script path must start with a slash (/) and end with **.sh**.

Parameter	Mandatory	Type	Description
parameters	No	String	Parameters of a custom automation script. ● Multiple parameters are separated by space. ● The following predefined system parameters can be transferred: – *${mrs_scale_node_num}*: Number of the nodes to be added or removed – *${mrs_scale_type}*: Scaling type. The value can be **scale_out** or **scale_in**. – *${mrs_scale_node_hostnames}*: Host names of the nodes to be added or removed – *${mrs_scale_node_ips}*: IP addresses of the nodes to be added or removed – *${mrs_scale_rule_name}*: Name of the rule that triggers auto scaling ● Other user-defined parameters are used in the same way as those of common shell scripts. Parameters are separated by space.
nodes	Yes	List<String>	Type of a node where the custom automation script is executed. The node type can be Master, Core, or Task.
active_master	No	Boolean	Whether the custom automation script runs only on the active Master node. The default value is **false**, indicating that the custom automation script can run on all Master nodes.

Parameter	Mandatory	Type	Description
action_stage	Yes	String	Time when a script is executed. The following four options are supported: • **before_scale_out**: before scale-out • **before_scale_in**: before scale-in • **after_scale_out**: after scale-out • **after_scale_in**: after scale-in
fail_action	Yes	String	Whether to continue to execute subsequent scripts and create a cluster after the custom automation script fails to be executed. • **continue**: Continue to execute subsequent scripts. • **errorout**: Stop the action. **NOTE** • You are advised to set this parameter to **continue** in the commissioning phase so that the cluster can continue to be installed and started no matter whether the custom automation script is executed successfully. • The scale-in operation cannot be undone. Therefore, **fail_action** must be set to **continue** for the scripts that are executed after scale-in.

Table 5-12 rules parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Name of an auto scaling rule. A cluster name can contain only 1 to 64 characters. Only letters, digits, hyphens (-), and underscores (_) are allowed. Rule names must be unique in a node group.

Parameter	Mandatory	Type	Description
description	No	String	Description about an auto scaling rule. It contains a maximum of 1024 characters.
adjustment_type	Yes	String	Auto scaling rule adjustment type. Possible values: • **scale_out**: cluster scale-out • **scale_in**: cluster scale-in
cool_down_minutes	Yes	Integer	Cluster cooling time after an auto scaling rule is triggered, when no auto scaling operation is performed. The unit is minute. Value range: 0 to 10,080. One week is equal to 10,080 minutes.
scaling_adjustment	Yes	Integer	Number of nodes that can be adjusted once. Value range: 1 to 100
trigger	Yes	Trigger object	Condition for triggering a rule. For details, see **Table 5-13**.

Table 5-13 trigger parameters

Parameter	Mandatory	Type	Description
metric_name	Yes	String	Metric name. This triggering condition makes a judgment according to the value of the metric. A metric name contains a maximum of 64 characters.
metric_value	Yes	String	Metric threshold to trigger a rule. The value must be an integer or a number with two decimal places.
comparison_operator	No	String	Metric judgment logic operator. Possible values: • **LT**: less than • **GT**: greater than • **LTOE**: less than or equal to • **GTOE**: greater than or equal to

Parameter	Mandatory	Type	Description
evaluation_periods	Yes	Integer	Number of consecutive five-minute periods, during which a metric threshold is reached. Value range: 1 to 288

Response Parameters

Status code: 200

Table 5-14 Response parameters

Parameter	Type	Description
cluster_id	String	Cluster ID, which is returned by the system after the cluster is created.

Example Request

- Creating an analysis cluster

```
POST /v2/{project_id}/clusters

{
  "cluster_version": "MRS 3.1.0",
  "cluster_name": "mrs_DyJA_dm",
  "cluster_type": "ANALYSIS",
  "charge_info": {
     "charge_mode": "postPaid"
  },
  "region": "",
  "availability_zone": "",
  "vpc_name": "vpc-37cd",
  "subnet_id" : "1f8c5ca6-1f66-4096-bb00-baf175954f6e",
  "subnet_name": "subnet",
  "components": "Hadoop,Spark2x,HBase,Hive,Hue,Flink,Oozie,Ranger,Tez",
  "safe_mode": "KERBEROS",
  "manager_admin_password": "Mrs@1234",
  "login_mode": "PASSWORD",
  "node_root_password": "Mrs@1234",
  "log_collection": 1,
  "mrs_ecs_default_agency": "MRS_ECS_DEFAULT_AGENCY",
  "tags": [
    {
      "key": "tag1",
      "value": "111"
    },
    {
      "key": "tag2",
      "value": "222"
    }
  ],
  "node_groups": [
    {
      "group_name": "master_node_default_group",
      "node_num": 2,
      "node_size": "rc3.4xlarge.4.linux.bigdata",
```

```
    "root_volume": {
      "type": "SAS",
      "size": 480
    },
    "data_volume": {
      "type": "SAS",
      "size": 600
    },
    "data_volume_count": 1
  },
  {
    "group_name": "core_node_analysis_group",
    "node_num": 3,
    "node_size": "rc3.4xlarge.4.linux.bigdata",
    "root_volume": {
      "type": "SAS",
      "size": 480
    },
    "data_volume": {
      "type": "SAS",
      "size": 600
    },
    "data_volume_count": 1
  },
  {
    "group_name": "task_node_analysis_group",
    "node_num": 3,
    "node_size": "rc3.4xlarge.4.linux.bigdata",
    "root_volume": {
      "type": "SAS",
      "size": 480
    },
    "data_volume": {
      "type": "SAS",
      "size": 600
    },
    "data_volume_count": 1,
   "auto_scaling_policy": {
              "auto_scaling_enable": true,
              "min_capacity": 0,
              "max_capacity": 1,
              "resources_plans" : [ {
                "period_type" : "daily",
                "start_time" : "12:00",
                "end_time" : "13:00",
                "min_capacity" : 2,
                "max_capacity" : 3
              } ],
              "exec_scripts" : [ {
                "name" : "test",
                "uri" : "s3a://obs-mrstest/bootstrap/basic_success.sh",
                "parameters" : "",
                "nodes" : [ "master_node_default_group", "core_node_analysis_group",
"task_node_analysis_group" ],
                "active_master" : false,
                "action_stage" : "before_scale_out",
                "fail_action" : "continue"
              } ],
              "rules": [
                 {
                    "name": "default-expand-1",
                    "description": "",
                    "adjustment_type": "scale_out",
                    "cool_down_minutes": 5,
                    "scaling_adjustment": "1",
                    "trigger": {
                       "metric_name": "StormSlotAvailablePercentage",
                       "metric_value": 100,
                       "comparison_operator": "LTOE",
```

```
                    "evaluation_periods": "1"
                }
            }
        ]
    }
  }
 ]
}
```

- Creating a streaming cluster

```
POST /v2/{project_id}/clusters

{
  "cluster_version": "MRS 3.1.0",
  "cluster_name": "mrs_Dokle_dm",
  "cluster_type": "STREAMING",
  "charge_info": {
     "charge_mode": "postPaid"
  },
  "region": "",
  "availability_zone": "",
  "vpc_name": "vpc-37cd",
   "subnet_id" : "1f8c5ca6-1f66-4096-bb00-baf175954f6e",
  "subnet_name": "subnet",
  "components": ""Storm,Kafka,Flume,Ranger",
  "safe_mode": "KERBEROS",
  "manager_admin_password": "Mrs@1234",
  "login_mode": "PASSWORD",
  "node_root_password": "Mrs@1234",
  "log_collection": 1,
  "mrs_ecs_default_agency": "MRS_ECS_DEFAULT_AGENCY",
  "tags": [
   {
     "key": "tag1",
     "value": "111"
   },
   {
     "key": "tag2",
     "value": "222"
   }
  ],
  "node_groups": [
   {
     "group_name": "master_node_default_group",
     "node_num": 2,
     "node_size": "rc3.4xlarge.4.linux.bigdata",
     "root_volume": {
       "type": "SAS",
       "size": 480
     },
     "data_volume": {
       "type": "SAS",
       "size": 600
     },
     "data_volume_count": 1
   },
   {
     "group_name": "core_node_streaming_group",
     "node_num": 3,
     "node_size": "rc3.4xlarge.4.linux.bigdata",
     "root_volume": {
       "type": "SAS",
       "size": 480
     },
     "data_volume": {
       "type": "SAS",
       "size": 600
     },
     "data_volume_count": 1,
   },
```

```
  {
    "group_name": "task_node_streaming_group",
    "node_num": 0,
    "node_size": "rc3.4xlarge.4.linux.bigdata",
    "root_volume": {
      "type": "SAS",
      "size": 480
    },
    "data_volume": {
      "type": "SAS",
      "size": 600
    },
    "data_volume_count": 1,
   "auto_scaling_policy": {
            "auto_scaling_enable": true,
            "min_capacity": 0,
            "max_capacity": 1,
            "resources_plans" : [ {
              "period_type" : "daily",
              "start_time" : "12:00",
              "end_time" : "13:00",
              "min_capacity" : 2,
              "max_capacity" : 3
            } ],
            "rules": [
               {
                  "name": "default-expand-1",
                  "description": "",
                  "adjustment_type": "scale_out",
                  "cool_down_minutes": 5,
                  "scaling_adjustment": "1",
                  "trigger": {
                     "metric_id": 2003,
                     "metric_name": "StormSlotAvailablePercentage",
                     "metric_value": 100,
                     "comparison_operator": "LTOE",
                     "evaluation_periods": "1"
                  }
               }
            ]
         }
   }
 ]
}
```

- Creating a hybrid cluster

```
POST /v2/{project_id}/clusters

{
 "cluster_version" : "MRS 3.1.0",
 "cluster_name" : "mrs_onmm_dm",
 "cluster_type" : "MIXED",
 "charge_info" : {
    "charge_mode" : "postPaid"
 },
 "region" : "",
 "availability_zone" : "",
 "vpc_name" : "vpc-37cd",
 "subnet_id" : "1f8c5ca6-1f66-4096-bb00-baf175954f6e",
 "subnet_name" : "subnet",
 "components" : "Hadoop,Spark2x,HBase,Hive,Hue,Loader,Kafka,Storm,Flume,Flink,Oozie,Ranger,Tez",
 "safe_mode" : "KERBEROS",
 "manager_admin_password" : "your password",
 "login_mode" : "PASSWORD",
 "node_root_password" : "your password",
 "log_collection" : 1,
 "mrs_ecs_default_agency" : "MRS_ECS_DEFAULT_AGENCY",
 "tags" : [ {
  "key" : "tag1",
  "value" : "111"
```

```
}, {
  "key" : "tag2",
  "value" : "222"
} ],
"node_groups" : [ {
  "group_name" : "master_node_default_group",
  "node_num" : 2,
  "node_size" : "Sit3.4xlarge.4.linux.bigdata",
  "root_volume" : {
    "type" : "SAS",
    "size" : 480
  },
  "data_volume" : {
    "type" : "SAS",
    "size" : 600
  },
  "data_volume_count" : 1
}, {
  "group_name" : "core_node_streaming_group",
  "node_num" : 3,
  "node_size" : "Sit3.4xlarge.4.linux.bigdata",
  "root_volume" : {
    "type" : "SAS",
    "size" : 480
  },
  "data_volume" : {
    "type" : "SAS",
    "size" : 600
  },
  "data_volume_count" : 1
}, {
  "group_name" : "core_node_analysis_group",
  "node_num" : 3,
  "node_size" : "Sit3.4xlarge.4.linux.bigdata",
  "root_volume" : {
    "type" : "SAS",
    "size" : 480
  },
  "data_volume" : {
    "type" : "SAS",
    "size" : 600
  },
  "data_volume_count" : 1
}, {
  "group_name" : "task_node_analysis_group",
  "node_num" : 1,
  "node_size" : "Sit3.4xlarge.4.linux.bigdata",
  "root_volume" : {
    "type" : "SAS",
    "size" : 480
  },
  "data_volume" : {
    "type" : "SAS",
    "size" : 600
  },
  "data_volume_count" : 1
}, {
  "group_name" : "task_node_streaming_group",
  "node_num" : 0,
  "node_size" : "Sit3.4xlarge.4.linux.bigdata",
  "root_volume" : {
    "type" : "SAS",
    "size" : 480
  },
  "data_volume" : {
    "type" : "SAS",
    "size" : 600
  },
  "data_volume_count" : 1
```

```
 } ]
}
```

- Creating a custom cluster with management and controller nodes co-deployed

```
POST /v2/{project_id}/clusters

{
  "cluster_version": "MRS 3.1.0",
  "cluster_name": "mrs_heshe_dm",
  "cluster_type": "CUSTOM",
  "charge_info": {
     "charge_mode": "postPaid"
  },
  "region": "",
  "availability_zone": "",
  "vpc_name": "vpc-37cd",
  "subnet_id" : "1f8c5ca6-1f66-4096-bb00-baf175954f6e",
  "subnet_name": "subnet-ed99",
  "components":
"Hadoop,Spark2x,HBase,Hive,Hue,Kafka,Flume,Flink,Oozie,HetuEngine,Ranger,Tez,ZooKeeper,ClickHous
e",
  "safe_mode": "KERBEROS",
  "manager_admin_password": "Mrs@1234",
  "login_mode": "PASSWORD",
  "node_root_password": "Mrs@1234",
  "mrs_ecs_default_agency": "MRS_ECS_DEFAULT_AGENCY",
  "template_id": "mgmt_control_combined_v2",
  "log_collection": 1,
  "tags": [
   {
     "key": "tag1",
     "value": "111"
   },
   {
     "key": "tag2",
     "value": "222"
   }
  ],
  "node_groups": [
   {
     "group_name": "master_node_default_group",
     "node_num": 3,
     "node_size": "Sit3.4xlarge.4.linux.bigdata",
     "root_volume": {
       "type": "SAS",
       "size": 480
     },
     "data_volume": {
       "type": "SAS",
       "size": 600
     },
     "data_volume_count": 1,
     "assigned_roles": [
               "OMSServer:1,2",
               "SlapdServer:1,2",
               "KerberosServer:1,2",
               "KerberosAdmin:1,2",
               "quorumpeer:1,2,3",
               "NameNode:2,3",
               "Zkfc:2,3",
               "JournalNode:1,2,3",
               "ResourceManager:2,3",
               "JobHistoryServer:2,3",
               "DBServer:1,3",
               "Hue:1,3",
               "MetaStore:1,2,3",
               "WebHCat:1,2,3",
               "HiveServer:1,2,3",
               "HMaster:2,3",
```

```
                "MonitorServer:1,2",
                "Nimbus:1,2",
                "UI:1,2",
                "JDBCServer2x:1,2,3",
                "JobHistory2x:2,3",
                "SparkResource2x:1,2,3",
                "oozie:2,3",
                "LoadBalancer:2,3",
                "TezUI:1,3",
                "TimelineServer:3",
                "RangerAdmin:1,2",
                "UserSync:2",
                "TagSync:2",
                "KerberosClient",
                "SlapdClient",
                "meta",
                "HSConsole:2,3",
                "FlinkResource:1,2,3",
                "DataNode:1,2,3",
                "NodeManager:1,2,3",
                "IndexServer2x:1,2",
                "ThriftServer:1,2,3",
                "RegionServer:1,2,3",
                "ThriftServer1:1,2,3",
                "RESTServer:1,2,3",
                "Broker:1,2,3",
                "Supervisor:1,2,3",
                "Logviewer:1,2,3",
                "Flume:1,2,3",
                "HSBroker:1,2,3"
]
   },
   {
     "group_name": "node_group_1",
     "node_num": 3,
     "node_size": "Sit3.4xlarge.4.linux.bigdata",
     "root_volume": {
       "type": "SAS",
       "size": 480
     },
     "data_volume": {
       "type": "SAS",
       "size": 600
     },
     "data_volume_count": 1,
     "assigned_roles": [
                "DataNode",
                "NodeManager",
                "RegionServer",
                "Flume:1",
                "Broker",
                "Supervisor",
                "Logviewer",
                "HBaseIndexer",
                "KerberosClient",
                "SlapdClient",
                "meta",
                "HSBroker:1,2",
                "ThriftServer",
                "ThriftServer1",
                "RESTServer",
                "FlinkResource"]
   },
   {
     "group_name": "node_group_2",
     "node_num": 1,
     "node_size": "Sit3.4xlarge.4.linux.bigdata",
     "root_volume": {
       "type": "SAS",
```

```
      "size": 480
    },
    "data_volume": {
      "type": "SAS",
      "size": 600
    },
    "data_volume_count": 1,
    "assigned_roles": [
            "NodeManager",
            "KerberosClient",
            "SlapdClient",
            "meta",
            "FlinkResource"]
  }
 ]
}
```

- Creating a custom cluster with management and controller nodes deployed separately

```
POST /v2/{project_id}/clusters

{
 "cluster_version": "MRS 3.1.0",
 "cluster_name": "mrs_jdRU_dm01",
 "cluster_type": "CUSTOM",
 "charge_info": {
    "charge_mode": "postPaid"
 },
 "region": "",
 "availability_zone": "",
 "vpc_name": "vpc-37cd",
 "subnet_id" : "1f8c5ca6-1f66-4096-bb00-baf175954f6e",
 "subnet_name": "subnet-ed99",
 "components":
"Hadoop,Spark2x,HBase,Hive,Hue,Kafka,Flume,Flink,Oozie,HetuEngine,Ranger,Tez,Ranger,Tez,ZooKeeper,ClickHouse",
 "safe_mode": "KERBEROS",
 "manager_admin_password": "Mrs@1234",
 "login_mode": "PASSWORD",
 "node_root_password": "Mrs@1234",
 "mrs_ecs_default_agency": "MRS_ECS_DEFAULT_AGENCY",
 "log_collection": 1,
 "template_id": "mgmt_control_separated_v2",
 "tags": [
  {
    "key": "aaa",
    "value": "111"
  },
  {
    "key": "bbb",
    "value": "222"
  }
 ],
 "node_groups": [
  {
    "group_name": "master_node_default_group",
    "node_num": 5,
    "node_size": "rc3.4xlarge.4.linux.bigdata",
    "root_volume": {
      "type": "SAS",
      "size": 480
    },
    "data_volume": {
      "type": "SAS",
      "size": 600
    },
    "data_volume_count": 1,
    "assigned_roles": [
            "OMSServer:1,2",
            "SlapdServer:3,4",
```

```
            "KerberosServer:3,4",
            "KerberosAdmin:3,4",
            "quorumpeer:3,4,5",
            "NameNode:4,5",
            "Zkfc:4,5",
            "JournalNode:1,2,3,4,5",
            "ResourceManager:4,5",
            "JobHistoryServer:4,5",
            "DBServer:3,5",
            "Hue:1,2",
            "MetaStore:1,2,3,4,5",
            "WebHCat:1,2,3,4,5",
            "HiveServer:1,2,3,4,5",
            "HMaster:4,5",
            "MonitorServer:1,2",
            "Nimbus:1,2",
            "UI:1,2",
            "JDBCServer2x:1,2,3,4,5",
            "JobHistory2x:4,5",
            "SparkResource2x:1,2,3,4,5",
            "oozie:1,2",
            "LoadBalancer:1,2",
            "TezUI:1,2",
            "TimelineServer:5",
            "RangerAdmin:1,2",
            "KerberosClient",
            "SlapdClient",
            "meta",
            "HSConsole:1,2",
            "FlinkResource:1,2,3,4,5",
            "DataNode:1,2,3,4,5",
            "NodeManager:1,2,3,4,5",
            "IndexServer2x:1,2",
            "ThriftServer:1,2,3,4,5",
            "RegionServer:1,2,3,4,5",
            "ThriftServer1:1,2,3,4,5",
            "RESTServer:1,2,3,4,5",
            "Broker:1,2,3,4,5",
            "Supervisor:1,2,3,4,5",
            "Logviewer:1,2,3,4,5",
            "Flume:1,2,3,4,5",
            "HBaseIndexer:1,2,3,4,5",
            "TagSync:1",
            "UserSync:1"]
    },
    {
      "group_name": "node_group_1",
      "node_num": 3,
      "node_size": "rc3.4xlarge.4.linux.bigdata",
      "root_volume": {
        "type": "SAS",
        "size": 480
      },
      "data_volume": {
        "type": "SAS",
        "size": 600
      },
      "data_volume_count": 1,
      "assigned_roles": [
            "DataNode",
            "NodeManager",
            "RegionServer",
            "Flume:1",
            "Broker",
            "Supervisor",
            "Logviewer",
            "HBaseIndexer",
            "KerberosClient",
            "SlapdClient",
```

```
            "meta",
            "HSBroker:1,2",
            "ThriftServer",
            "ThriftServer1",
            "RESTServer",
            "FlinkResource"]
  }
 ]
}
```

- Creating a custom data cluster

```
POST /v2/{project_id}/clusters

{
  "cluster_version": "MRS 3.1.0",
  "cluster_name": "mrs_jdRU_dm02",
  "cluster_type": "CUSTOM",
  "charge_info": {
     "charge_mode": "postPaid"
  },
  "region": "",
  "availability_zone": "",
  "vpc_name": "vpc-37cd",
  "subnet_id" : "1f8c5ca6-1f66-4096-bb00-baf175954f6e",
  "subnet_name": "subnet-ed99",
  "components":
"Hadoop,Spark2x,HBase,Hive,Hue,Kafka,Flume,Flink,Oozie,Ranger,Tez,Ranger,Tez,ZooKeeper,ClickHouse
",
  "safe_mode": "KERBEROS",
  "manager_admin_password": "Mrs@1234",
  "login_mode": "PASSWORD",
  "node_root_password": "Mrs@1234",
  "mrs_ecs_default_agency": "MRS_ECS_DEFAULT_AGENCY",
  "template_id": "mgmt_control_data_separated_v2",
  "log_collection": 1,
  "tags": [
    {
      "key": "aaa",
      "value": "111"
    },
    {
      "key": "bbb",
      "value": "222"
    }
  ],
  "node_groups": [
    {
      "group_name": "master_node_default_group",
      "node_num": 9,
      "node_size": "rc3.4xlarge.4.linux.bigdata",
      "root_volume": {
        "type": "SAS",
        "size": 480
      },
      "data_volume": {
        "type": "SAS",
        "size": 600
      },
      "data_volume_count": 1,
      "assigned_roles": [
            "OMSServer:1,2",
            "SlapdServer:5,6",
            "KerberosServer:5,6",
            "KerberosAdmin:5,6",
            "quorumpeer:5,6,7,8,9",
            "NameNode:3,4",
            "Zkfc:3,4",
            "JournalNode:5,6,7",
            "ResourceManager:8,9",
            "JobHistoryServer:8",
```

```
            "DBServer:8,9",
            "Hue:8,9",
            "FlinkResource:3,4",
            "MetaStore:8,9",
            "WebHCat:5",
            "HiveServer:8,9",
            "HMaster:8,9",
            "MonitorServer:3,4",
            "Nimbus:8,9",
            "UI:8,9",
            "JDBCServer2x:8,9",
            "JobHistory2x:8,9",
            "SparkResource2x:5,6,7",
            "oozie:4,5",
            "LoadBalancer:8,9",
            "TezUI:5,6",
            "TimelineServer:5",
            "RangerAdmin:4,5",
            "UserSync:5",
            "TagSync:5",
            "KerberosClient",
            "SlapdClient",
            "meta",
            "HSBroker:5",
            "HSConsole:3,4",
            "FlinkResource:3,4"]
    },
    {
      "group_name": "node_group_1",
      "node_num": 3,
      "node_size": "rc3.4xlarge.4.linux.bigdata",
      "root_volume": {
        "type": "SAS",
        "size": 480
      },
      "data_volume": {
        "type": "SAS",
        "size": 600
      },
      "data_volume_count": 1,
      "assigned_roles": [
            "DataNode",
            "NodeManager",
            "RegionServer",
            "Flume:1",
            "GraphServer",
            "KerberosClient",
            "SlapdClient",
            "meta",
            "HSBroker:1,2"
]
    },
    {
      "group_name": "node_group_2",
      "node_num": 3,
      "node_size": "rc3.4xlarge.4.linux.bigdata",
      "root_volume": {
        "type": "SAS",
        "size": 480
      },
      "data_volume": {
        "type": "SAS",
        "size": 600
      },
      "data_volume_count": 1,
      "assigned_roles": [
            "HBaseIndexer",
            "SolrServer[3]",
            "EsNode[2]",
```

```
                "KerberosClient",
                "SlapdClient",
                "meta"
]
    },
    {
      "group_name": "node_group_3",
      "node_num": 3,
      "node_size": "rc3.4xlarge.4.linux.bigdata",
      "root_volume": {
        "type": "SAS",
        "size": 480
      },
      "data_volume": {
        "type": "SAS",
        "size": 600
      },
      "data_volume_count": 1,
      "assigned_roles": [
                "Redis[2]",
                "KerberosClient",
                "SlapdClient",
                "meta"]
    },
    {
      "group_name": "node_group_4",
      "node_num": 3,
      "node_size": "rc3.4xlarge.4.linux.bigdata",
      "root_volume": {
        "type": "SAS",
        "size": 480
      },
      "data_volume": {
        "type": "SAS",
        "size": 600
      },
      "data_volume_count": 1,
      "assigned_roles": [
                "Broker",
                "Supervisor",
                "Logviewer",
                "KerberosClient",
                "SlapdClient",
                "meta"]
    }
  ]
}
```

Example Response

- Example of a successful response

```
{
    "cluster_id": "da1592c2-bb7e-468d-9ac9-83246e95447a"
}
```

- Example of a failed response

```
{
    "error_code": "MRS.0002",
    "error_msg": "The parameter is invalid."
}
```

Status Codes

See **Status Codes**.

Error Codes

See **Error Codes**.

5.1.2 Changing a Cluster Name

Function

Changing a Cluster Name

URI

PUT /v2/{project_id}/clusters/{cluster_id}/cluster-name

Table 5-15 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	The project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.
cluster_id	Yes	String	The cluster ID. For details about how to obtain the cluster ID, see **Obtaining a Cluster ID**.

Request Parameters

Table 5-16 Request body parameter

Parameter	Mandatory	Type	Description
cluster_name	Yes	String	The new cluster name.

Response Parameters

Status code: 200

Table 5-17 Response body parameter

Parameter	Type	Description
result	String	The operation result of the mapping update request. Value **succeeded** indicates that the operation is successful, and value **failed** indicates that the operation fails.

Status code: 400

Table 5-18 Response body parameters

Parameter	Type	Description
error_code	String	The error code.
error_msg	String	The error message.

Example Request

```
{
  "cluster_name" : "mrs_jdRU_dm01"
}
```

Example Response

Status code: 200

The cluster name is changed.

```
{
  "result" : "succeeded"
}
```

Status Codes

See **Status Codes**.

Error Codes

See **Error Codes**.

5.2 Job Object APIs

5.2.1 Adding and Executing a Job

Function

This API is used to add and submit a job in an MRS cluster.

NOTE

- On the **Dashboard** tab page of the cluster details page, click **Click to synchronize** on the right side of **IAM User Sync** to synchronize IAM users. Then submit a job through this API.

URI

POST /v2/{project_id}/clusters/{cluster_id}/job-executions

Table 5-19 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.
cluster_id	Yes	String	Cluster ID. For details on how to obtain the cluster ID, see **Obtaining a Cluster ID**.

Request Parameters

Table 5-20 Request parameters

Parameter	Mandatory	Type	Description
job_type	Yes	String	Type of a job. • MapReduce • SparkSubmit • SparkPython: Example request of a SparkPython job (Jobs of this type will be converted to SparkSubmit jobs for submission. The job type is displayed as SparkSubmit on the MRS console. Select SparkSubmit when you call an API to query the job list.) • HiveScript • HiveSql • DistCp, importing and exporting data • SparkScript • SparkSql • Flink **NOTE** Spark, Hive, and Flink jobs can be added to only clusters that include Spark, Hive, and Flink components.

Parameter	Mandatory	Type	Description
job_name	Yes	String	Job name. It contains 1 to 64 characters. Only letters, digits, hyphens (-), and underscores (_) are allowed. **NOTE** Identical job names are allowed but not recommended.
arguments	No	Array of strings	Key parameter for program execution. The parameter is specified by the function of the user's program. MRS is only responsible for loading the parameter. The value can contain a maximum of 150,000 characters. Special characters (;|&>'<$!\\) are not allowed. This parameter can be left blank. **NOTE** • If you enter a parameter with sensitive information (such as the login password), the parameter may be exposed in the job details display and log printing. Exercise caution when performing this operation. • For MRS 3.x or later, a file path on OBS can start with **obs://**. To use this format to submit HiveScript or HiveSQL jobs, choose **Components** > **Hive** > **Service Configuration** on the cluster details page. Set **Basic** to **All**, and search for **core.site.customized.configs**. Add the endpoint configuration item **fs.obs.endpoint** of OBS and enter the endpoint corresponding to OBS in **Value**. For details, see **Endpoints**.
properties	No	Map<String,String>	Program system parameter. The parameter contains a maximum of 2,048 characters, excluding special characters such as ><|'`&!\, and can be left blank.

Response Parameters

Status code: 200

Table 5-21 Response body parameter

Parameter	Type	Description
job_submit_result	JobSubmitResult object	The job execution result. For details about the parameters, see **Table 5-22**.

Table 5-22 JobSubitResult parameters

Parameter	Type	Description
job_id	String	Job ID
state	String	Job submission status. • **COMPLETE**: The job is submitted. • **JOBSTAT_SUBMIT_FAILED**: Failed to submit the job.

Status code: 500

Table 5-23 Response body parameters

Parameter	Type	Description
error_code	String	The error code.
error_msg	String	The error message.

Example Request

You must have prepared the OBS paths, sample files, endpoints, and AKs/SKs when submitting a request.

- Example request of a MapReduce job:

```
POST https://{endpoint}/v2/{project_id}/clusters/{cluster_id}/job-executions

{
    "job_name":"MapReduceTest",
    "job_type":"MapReduce",
    "arguments":[
        "obs://obs-test/program/hadoop-mapreduce-examples-x.x.x.jar",
        "wordcount",
        "obs://obs-test/input/",
        "obs://obs-test/job/mapreduce/output"
    ],
    "properties":{
        "fs.obs.endpoint":"obs endpoint",
        "fs.obs.access.key":"xxx",
        "fs.obs.secret.key":"yyy"
    }
}
```

- Example request of a SparkSubmit job:

```
POST https://{endpoint}/v2/{project_id}/clusters/{cluster_id}/job-executions
```

```
{
   "job_name":"SparkSubmitTest",
   "job_type":"SparkSubmit",
   "arguments":[
      "--master",
      "yarn",
      "--deploy-mode",
      "cluster",
      "--py-files",
      "obs://obs-test/a.py",
      "--conf",
      "spark.yarn.appMasterEnv.PYTHONPATH=/tmp:$PYTHONPATH",
      "--conf",
      "spark.yarn.appMasterEnv.aaa=aaaa",
      "--conf",
      "spark.executorEnv.aaa=executoraaa",
      "--properties-file",
      "obs://obs-test/test-spark.conf",
      "obs://obs-test/pi.py",
      "100000"
   ],
   "properties":{
      "fs.obs.access.key":"xxx",
      "fs.obs.secret.key":"yyy"
   }
}
```

- Example request of a HiveScript job:

```
POST https://{endpoint}/v2/{project_id}/clusters/{cluster_id}/job-executions

{
   "job_name":"HiveScriptTest",
   "job_type":"HiveScript",
   "arguments":[
      "obs://obs-test/sql/test_script.sql"
   ],
   "properties":{
      "fs.obs.endpoint":"obs endpoint",
      "fs.obs.access.key":"xxx",
      "fs.obs.secret.key":"yyy"
   }
}
```

- Example request of a HiveSql job:

```
POST https://{endpoint}/v2/{project_id}/clusters/{cluster_id}/job-executions

{
  "job_name" : "HiveSqlTest",
  "job_type" : "HiveSql",
  "arguments" : [ "DROP TABLE IF EXISTS src_wordcount;\ncreate external table src_wordcount(line string) row format delimited fields terminated by \"\\n\" stored as textfile location \"obs://donotdel-gxc/input/\";\ninsert into src_wordcount values(\"v1\")" ],
  "properties" : {
    "fs.obs.endpoint" : "obs endpoint",
    "fs.obs.access.key" : "xxx",
    "fs.obs.secret.key" : "yyy"
  }
}
```

- Example request of a DistCp job:

```
POST https://{endpoint}/v2/{project_id}/clusters/{cluster_id}/job-executions

{
   "job_name":"DistCpTest",
   "job_type":"DistCp",
   "arguments":[
      "obs://obs-test/DistcpJob/",
      "/user/test/sparksql/"
   ],
   "properties":{
      "fs.obs.endpoint":"obs endpoint",
```

```
        "fs.obs.access.key":"xxx",
        "fs.obs.secret.key":"yyy"
    }
}
```

- Example request of a SparkScript job:

```
POST https://{endpoint}/v2/{project_id}/clusters/{cluster_id}/job-executions

{
    "job_name":"SparkScriptTest",
    "job_type":"SparkScript",
    "arguments":[
        "op-key1",
        "op-value1",
        "op-key2",
        "op-value2",
        "obs://obs-test/sql/test_script.sql"
    ],
    "properties":{
        "fs.obs.access.key":"xxx",
        "fs.obs.secret.key":"yyy"
    }
}
```

- Example request of a SparkSql job:

```
POST https://{endpoint}/v2/{project_id}/clusters/{cluster_id}/job-executions

{
    "job_name":"SparkSqlTest",
    "job_type":"SparkSql",
    "arguments":[
        "op-key1",
        "op-value1",
        "op-key2",
        "op-value2",
        "create table student_info3 (id string,name string,gender string,age int,addr string);"
    ],
    "properties":{
        "fs.obs.access.key":"xxx",
        "fs.obs.secret.key":"yyy"
    }
}
```

- Example request of a Flink job:

```
POST https://{endpoint}/v2/{project_id}/clusters/{cluster_id}/job-executions

{
    "job_name":"FlinkTest",
    "job_type":"Flink",
    "arguments":[
        "run",
        "-d",
        "-ynm",
        "testExcutorejobhdfsbatch",
        "-m",
        "yarn-cluster",
        "hdfs://test/examples/batch/WordCount.jar"
    ],
    "properties":{
        "fs.obs.endpoint":"obs endpoint",
        "fs.obs.access.key":"xxx",
        "fs.obs.secret.key":"yyy"
    }
}
```

- The following shows an example request of a SparkPython job (Jobs of this type will be converted to SparkSubmit jobs for submission. The job type is displayed as SparkSubmit on the MRS console. Select SparkSubmit when you call an API to query the job list.)

```
POST https://{endpoint}/v2/{project_id}/clusters/{cluster_id}/job-executions

{
  "job_name" : "SparkPythonTest",
  "job_type" : "SparkPython",
  "arguments" : [ "--master", "yarn", "--deploy-mode", "cluster", "--py-files", "obs://obs-test/a.py", "--conf", "spark.yarn.appMasterEnv.PYTHONPATH=/tmp:$PYTHONPATH", "--conf", "spark.yarn.appMasterEnv.aaa=aaaa", "--conf", "spark.executorEnv.aaa=executoraaa", "--properties-file", "obs://obs-test/test-spark.conf", "obs://obs-test/pi.py", "100000" ],
  "properties" : {
    "fs.obs.access.key" : "xxx",
    "fs.obs.secret.key" : "yyy"
  }
}
```

Example Response

Status code: 200

- Example of a successful response

```
{
  "job_submit_result":{
      "job_id":"44b37a20-ffe8-42b1-b42b-78a5978d7e40",
      "state":"COMPLETE"
  }
}
```

Status code: 500

- Example of a failed response

```
{
"error_msg": Hive jobs cannot be submitted.
"error_code":"0168"
}
```

Status Codes

See **Status Codes**.

Error Codes

See **Error Codes**.

5.2.2 Querying Information About a Job

Function

This API is used to query information about a specified job in an MRS cluster.

URI

- Format

  GET /v2/{project_id}/clusters/{cluster_id}/job-executions/{job_execution_id}

- Parameter description

Table 5-24 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	The project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.
cluster_id	Yes	String	The cluster ID. For details on how to obtain the cluster ID, see **Obtaining a Cluster ID**.
job_execution_id	Yes	String	The job ID. For details about how to obtain the job ID, see **Obtaining a Job ID**.

Request Parameters

Request parameters

None

Response Parameters

Status code: 202

Table 5-25 Response parameter

Parameter	Type	Description
job_detail	JobQueryBean object	Job details. For details about the parameter, see **Table 5-26**.

Table 5-26 Job parameters

Parameter	Type	Description
job_id	String	The job ID.
user	String	The name of the user who submits a job.
job_name	String	The job name.

Parameter	Type	Description
job_result	String	The final result of a job. Possible values: • **FAILED**: indicates that the job fails to be executed. • **KILLED**: indicates that the job is manually terminated during execution. • **UNDEFINED**: indicates that the job is being executed. • **SUCCEEDED**: indicates that the job has been successfully executed.
job_state	String	The job execution status. Possible values: • **FAILED**: indicates that the job failed. • **KILLED**: indicates that the job is terminated. • **New**: indicates that the job is created. • **NEW_SAVING**: indicates that the job has been created and is being saved. • **SUBMITTED**: indicates that the job is submitted. • **ACCEPTED**: indicates that the job is accepted. • **RUNNING**: indicates that the job is running. • **FINISHED**: indicates that the job is completed.
job_progress	Float	The job execution progress.
job_type	String	The job type. Possible values: • **MapReduce** • **SparkSubmit**: Select this value when you call an API to query SparkPython jobs. • **HiveScript** • **HiveSql** • **DistCp**: imports and exports data. • **SparkScript** • **SparkSql** • **Flink**

Parameter	Type	Description
started_time	Long	The time when a job starts to execute, in milliseconds.
submitted_time	Long	The time when a job is submitted, in milliseconds.
finished_time	Long	The time when a job execution is completed, in milliseconds.
elapsed_time	Long	The job execution duration, in milliseconds.
arguments	String	Run parameters.
properties	String	The configuration parameter, which is used to configure **-d** parameters. The value can contain a maximum of 2,048 characters. Special characters (;|&>'<$!\\) are not allowed. This parameter can be left blank.
launcher_id	String	The launcher job ID.
app_id	String	The actual job ID.
tracking_url	String	The URL for accessing logs. Currently, only SparkSubmit jobs support this parameter. This parameter accesses the Yarn Web UI via the EIP bound to the cluster. If the EIP is unbound from the cluster on the VPC console, the MRS service data is not updated in a timely manner and the access fails. In this case, you can bind the EIP to the cluster again to rectify the fault.
queue	String	The type of the resource queue used by a job.

Status code: 500

Table 5-27 Response body parameters

Parameter	Type	Description
error_code	String	The error code.
error_msg	String	The error message.

Example

- Example request

  Querying information about a job

  ```
  GET https://{endpoint}/v2/{project_id}/clusters/{cluster_id}/job-executions/{job_execution_id}
  ```

- Example response
  - Example of a successful response

    ```
    {
        "job_detail": {
            "job_id": "431b135e-c090-489f-b1db-0abe3822b855",
            "user": "xxxx",
            "job_name": "pyspark1",
            "job_result": "SUCCEEDED",
            "job_state": "FINISHED",
            "job_progress": "100",
            "job_type": "SparkSubmit",
            "started_time": "1564626578817",
            "submitted_time": 1564626561541,
            "finished_time": 1564626664930,
            "elapsed_time": 86113,
            "queue": "default",
            "arguments": "[--class, org.apache.spark.examples.SparkPi, --driver-memory, 512MB, --num-executors, 1, --executor-cores, 1, --master, yarn-cluster, obs://obs-test/jobs/spark/spark-examples_2.11-2.1.0.jar, 10000]",
            "launcher_id": "application_1564622673393_0006",
            "app_id": "application_1564622673393_0007",
            "properties": "{}"
        }
    }
    ```

  - Example of a failed response

    ```
    {
    "error_msg": "Failed to query the job."
       "error_code":"0162"
    }
    ```

Status Codes

See **Status Codes**.

Error Codes

See **Error Codes**.

5.2.3 Querying a List of Jobs

Function

This API is used to query the job list in a specified MRS cluster.

URI

- Format

  GET /v2/{project_id}/clusters/{cluster_id}/job-executions

- Parameter description

Table 5-28 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	The project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.
cluster_id	Yes	String	The cluster ID. For details about how to obtain the cluster ID, see **Obtaining a Cluster ID**.

Request Parameters

Table 5-29 Request parameters

Parameter	Mandatory	Type	Description
job_name	No	String	The job name. The value can contain 1 to 128 characters. Only letters, numbers, hyphens (-), and underscores (_) are allowed.
job_id	No	String	The job ID. The value can contain 1 to 64 characters. Only letters, numbers, and hyphens (-) are allowed.
user	No	String	The username. The value can contain 1 to 32 characters. Only letters, numbers, hyphens (-), underscores (_), and periods (.) are allowed, but the value cannot start with a number.
job_type	No	String	The job type. Possible values: • **MapReduce** • **SparkSubmit** • **SparkSubmit**: Select this value when you call an API to query SparkPython jobs. • **HiveScript** • **HiveSql** • **DistCp**: imports and exports data. • **SparkScript** • **SparkSql** • **Flink**

Parameter	Mandatory	Type	Description
job_state	No	String	The job execution status. Possible values: • **FAILED**: indicates that the job fails to be executed. • **KILLED**: indicates that the job is terminated. • **New**: indicates that the job is created. • **NEW_SAVING**: indicates that the job has been created and is being saved. • **SUBMITTED**: indicates that the job is submitted. • **ACCEPTED**: indicates that the job is accepted. • **RUNNING**: indicates that the job is running. • **FINISHED**: indicates that the job is completed.
job_result	No	String	The job execution result. Possible values: • **FAILED**: indicates that the job fails to be executed. • **KILLED**: indicates that the job is manually terminated during execution. • **UNDEFINED**: indicates that the job is being executed. • **SUCCEEDED**: indicates that the job has been successfully executed.
queue	No	String	The type of the resource queue used by a job. The value can contain 1 to 64 characters. Only letters, numbers, and hyphens (-) are allowed.
limit	No	String	The number of records displayed on each page in the returned result. The default value is **10**.
offset	No	String	The default offset from which the job list starts to be queried is **1**.

Parameter	Mandatory	Type	Description
sort_by	No	String	The sorting mode of returned results. The default value is **desc**. Possible values: • **asc**: indicates that the returned results are ranked in ascending order. • **desc**: indicates that the returned results are ranked in descending order.
submitted_time_begin	No	Long	The UTC timestamp after which a job is submitted, in milliseconds, for example, **1562032041362**.
submitted_time_end	No	Long	The UTC timestamp before which a job is submitted, in milliseconds, for example, **1562032041362**.

Response Parameters

Status code: 202

Table 5-30 Response parameter

Parameter	Type	Description
job_list	Array of JobQueryBean objects	The job list. For details about the parameters, see **Table 5-31**.

Table 5-31 Job parameters

Parameter	Type	Description
job_id	String	The job ID.
user	String	The name of the user who submits a job.
job_name	String	The job name.

Parameter	Type	Description
job_result	String	The final result of a job. Possible values: • **FAILED**: indicates that the job fails to be executed. • **KILLED**: indicates that the job is manually terminated during execution. • **UNDEFINED**: indicates that the job is being executed. • **SUCCEEDED**: indicates that the job has been successfully executed.
job_state	String	The job execution status. Possible values: • **FAILED**: indicates that the job fails to be executed. • **KILLED**: indicates that the job is terminated. • **New**: indicates that the job is created. • **NEW_SAVING**: indicates that the job has been created and is being saved. • **SUBMITTED**: indicates that the job is submitted. • **ACCEPTED**: indicates that the job is accepted. • **RUNNING**: indicates that the job is running. • **FINISHED**: indicates that the job is completed.
job_progress	Float	The job execution progress.
job_type	String	The job type. Possible values: • **MapReduce** • **SparkSubmit**: Select this value when you call an API to query SparkPython jobs. • **HiveScript** • **HiveSql** • **DistCp**: imports and exports data. • **SparkScript** • **SparkSql** • **Flink**

Parameter	Type	Description
started_time	Long	The time when a job starts to execute, in milliseconds.
submitted_time	Long	The time when a job is submitted, in milliseconds.
finished_time	Long	The time when a job execution is completed, in milliseconds.
elapsed_time	Long	The job execution duration, in milliseconds.
arguments	String	The running parameters.
properties	String	The configuration parameter, which is used to configure **-d** parameters. The value can contain a maximum of 2,048 characters. Special characters (;|&>'<$!\\) are not allowed. This parameter can be left blank.
launcher_id	String	The actual job ID.
app_id	String	The actual job ID.
tracking_url	String	The URL for accessing logs. Currently, only SparkSubmit jobs support this parameter. This parameter accesses the Yarn Web UI via the EIP bound to the cluster. If the EIP is unbound from the cluster on the VPC console, the MRS service data is not updated in a timely manner and the access fails. In this case, you can bind the EIP to the cluster again to rectify the fault.
queue	String	The type of the resource queue used by a job.

Status code: 500

Table 5-32 Response body parameters

Parameter	Type	Description
error_code	String	The error code.
error_msg	String	The error message.

Example

- Example request

  Querying a list of jobs

  ```
  GET https://{endpoint}/v2/{project_id}/clusters/{cluster_id}/job-executions
  ```

- Example response
  - Example of a successful response

    ```
    {
       "job_list": [{
            "job_id": "981374c1-85da-44ee-be32-edfb4fba776c",
            "user": "xxxx",
            "job_name": "SparkSubmitTset",
            "job_result": "UNDEFINED",
            "job_state": "ACCEPTED",
            "job_progress": "0",
            "job_type": "SparkSubmit",
            "started_time": "0",
            "submitted_time": 1564714763119,
            "finished_time": "0",
            "elapsed_time": "0",
            "queue": "default",
            "arguments": "[--class, --driver-memory, --executor-cores, --master, yarn-cluster, obs://
    obs-test/hadoop-mapreduce-examples-3.1.1.jar, dddd]",
            "launcher_id": "application_1564622673393_0613",
            "properties": "{}"
        },
        {
            "job_id": "c54c8aa0-c277-4f83-8acc-521d85cfa32b",
            "user": "xxxx",
            "job_name": "SparkSubmitTset2",
            "job_result": "UNDEFINED",
            "job_state": "ACCEPTED",
            "job_progress": "0",
            "job_type": "SparkSubmit",
            "started_time": "0",
            "submitted_time": 1564714020099,
            "finished_time": "0",
            "elapsed_time": "0",
            "queue": "default",
            "arguments": "[--conf, yujjsjhe, --driver-memory, yueujdjjd, --master, yarn-cluster, obs://
    obs-test/hadoop-mapreduce-examples-3.1.1.jar]",
            "launcher_id": "application_1564622673393_0611",
            "properties": "{}"
        }
       ]
    }
    ```

  - Example of a failed response

    ```
    {
    "error_msg": "Failed to query the job list."
    "error_code":"0166"
    }
    ```

Status Codes

See **Status Codes**.

Error Codes

See **Error Codes**.

5.2.4 Terminating a Job

Function

This API is used to terminate a specified job in an MRS cluster.

URI

- Format

  POST /v2/{project_id}/clusters/{cluster_id}/job-executions/{job_execution_id}/kill

- Parameter description

  Table 5-33 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	The project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.
cluster_id	Yes	String	The cluster ID. For details about how to obtain the cluster ID, see **Obtaining a Cluster ID**.
job_execution_id	Yes	String	The job ID. For details about how to obtain the job ID, see **Obtaining a Job ID**.

Request Parameters

Request parameters

None

Response Parameters

Response parameters

None

Example

- Example request

  POST https://{endpoint}/v2/{project_id}/clusters/{cluster_id}/job-executions/{job_execution_id}/kill

- Example response
  - Example of a successful response

    None
  - Example of a failed response

```
{
"error_msg": "Failed to terminate the job."
  "error_code":"0175"
}
```

Status Codes

See **Status Codes**.

Error Codes

See **Error Codes**.

5.2.5 Obtaining SQL Results

Function

This API is used to obtain results returned after the SQL statements for querying SparkSQL and SparkScript jobs in an MRS cluster are executed.

URI

- Format

  GET /v2/{project_id}/clusters/{cluster_id}/job-executions/{job_execution_id}/sql-result

- Parameter description

  Table 5-34 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	The project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.
cluster_id	Yes	String	The cluster ID. For details about how to obtain the cluster ID, see **Obtaining a Cluster ID**.
job_execution_id	Yes	String	The job ID. For details about how to obtain the job ID, see **Obtaining a Job ID**.

Request Parameters

Request parameters

None

Response Parameters

Status code: 202

Table 5-35 Response parameter

Parameter	Type	Description
sql-results	Object	The query results of a SQL statement.

Status code: 500

Table 5-36 Response body parameters

Parameter	Type	Description
error_code	String	The error code.
error_msg	String	The error message.

Example Request

Example request for obtaining the SQL result

```
GET https://{endpoint}/v2/{project_id}/clusters/{cluster_id}/job-executions/{job_execution_id}/sql-result
```

Example Response

Status code: 202

The SQL statement execution result is obtained.

```
{
  "sql_results" : {
    "0" : [ {
      "result" : "succeed"
    } ],
    "1" : [ {
      "database" : "default",
      "isTemporary" : "false",
      "tableName" : "src_wordcount"
    } ],
    "2" : [ {
      "result" : "succeed"
    } ],
    "3" : [ {
      "result" : "succeed"
    } ],
    "4" : [ {
      "name" : "a",
      "id" : 1
    }, {
      "name" : "b",
      "id" : 2
    } ]
  }
}
```

Status code: 500

Failed to obtain the SQL statement execution result.

```
{
  "error_msg" : "Failed to obtain SQL job results.",
  "error_code" : "0172"
}
```

Status Codes

See **Status Codes**.

Error Codes

See **Error Codes**.

5.2.6 Deleting Jobs in Batches

Function

This API is used to delete jobs in batches.

URI

- Format
  POST /v2/{project_id}/clusters/{cluster_id}/job-executions/batch-delete
- Parameter description

Table 5-37 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	The project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.
cluster_id	Yes	String	The cluster ID. For details about how to obtain the cluster ID, see **Obtaining a Cluster ID**.

Request Parameters

Table 5-38 Request parameter

Parameter	Mandatory	Type	Description
job_id_list	Yes	Array of strings	The job ID list. For details about how to obtain the job ID, see **Obtaining a Job ID**.

Response Parameters

Response parameters

None

Example

- Example request

```
POST https://{endpoint}/v2/{project_id}/clusters/{cluster_id}/job-executions/batch-delete

{
  "job_id_list" : [ "48c45725-b699-4aa9-9bfd-f7ff87eb6fe8", "af846665-dd32-4349-a8b5-561e109c383c" ]
}
```

- Example response

  None

Status Codes

See **Status Codes**.

Error Codes

See **Error Codes**.

5.3 Auto Scaling APIs

5.3.1 Viewing Auto Scaling Policies

Function

This API is used to view all auto scaling policies of a specified cluster.

URI

GET /v2/{project_id}/autoscaling-policy/{cluster_id}

Table 5-39 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	The project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.
cluster_id	Yes	String	The cluster ID. For details about how to obtain the cluster ID, see **Obtaining a Cluster ID**.

Request Parameters

None

Response Parameters

Status code: 200

Table 5-40 Response body parameter

Parameter	Type	Description
[Array element]	Array of AutoScalingPolicyV2 objects	The auto scaling policy list. For details, see **Table 5-41**.

Table 5-41 AutoScalingPolicyV2

Parameter	Type	Description
node_group_name	String	The node group name.
resource_pool_name	String	The resource plan name.
auto_scaling_policy	AutoScalingPolicy object	The auto scaling policy. For details, see **Table 5-42**.

Table 5-42 AutoScalingPolicy

Parameter	Type	Description
auto_scaling_enable	Boolean	Whether to enable the auto scaling policy.

Parameter	Type	Description
min_capacity	Integer	The minimum number of nodes reserved in the node group. The value ranges from 0 to 500.
max_capacity	Integer	The maximum number of nodes in the node group. The value ranges from 0 to 500.
resources_plans	Array of ResourcesPlan objects	The resource plan list. If this parameter is left blank, the resource plan is disabled. When **auto_scaling_enable** is set to **true**, either this parameter or **resources_plans** must be configured. For details, see **Table 5-43**.
rules	Array of Rule objects	List of auto scaling rules. When **auto_scaling_enable** is set to **true**, either this parameter or **resources_plans** must be configured. For details, see **Table 5-44**.
exec_scripts	Array of ScaleScript objects	The list of custom scaling automation scripts. If this parameter is left blank, the automation script is disabled. For details, see **Table 5-46**.

Table 5-43 ResourcesPlan

Parameter	Type	Description
period_type	String	The cycle type of a resource plan. Currently, only the following cycle type is supported: daily
start_time	String	The start time of a resource plan. The value is in the format of **hour:minute**, indicating that the time ranges from 00:00 to 23:59.
end_time	String	The end time of a resource plan. The value is in the same format as that of **start_time**. The interval between **end_time** and **start_time** must be greater than or equal to 30 minutes.
min_capacity	Integer	The minimum number of reserved nodes in a node group in a resource plan. The value ranges from 0 to 500.

Parameter	Type	Description
max_capacity	Integer	The maximum number of reserved nodes in a node group in a resource plan. The value ranges from 0 to 500.

Table 5-44 Rule

Parameter	Type	Description
name	String	The name of an auto scaling rule. The name can contain only 1 to 64 characters. Only letters, numbers, hyphens (-), and underscores (_) are allowed. Rule names must be unique in a node group.
description	String	The description about an auto scaling rule. It contains a maximum of 1,024 characters.
adjustment_type	String	Auto scaling rule adjustment type. Possible values: • **scale_out**: cluster scale-out • **scale_in**: cluster scale-in
cool_down_minutes	Integer	The cluster cooling time after an auto scaling rule is triggered, when no auto scaling operation is performed. The unit is minute. The value ranges from 0 to 10080. One week is equal to 10,080 minutes.
scaling_adjustment	Integer	The number of cluster nodes that can be adjusted at a time. The value ranges from 1 to 100.
trigger	Trigger object	The condition for triggering a rule. For details, see **Table 5-45**.

Table 5-45 Trigger

Parameter	Type	Description
metric_name	String	The metric name. This triggering condition makes a judgment according to the value of the metric. A metric name contains a maximum of 64 characters.

Parameter	Type	Description
metric_value	String	The metric threshold to trigger a rule. The value must be an integer or a number with two decimal places.
comparison_operator	String	The metric judgment logic operator. Possible values: • **LT**: less than • **GT**: greater than • **LTOE**: less than or equal to • **GTOE**: greater than or equal to
evaluation_periods	Integer	The number of consecutive five-minute periods, during which a metric threshold is reached. The value ranges from 1 to 288.

Table 5-46 ScaleScript

Parameter	Type	Description
name	String	The name of a custom automation script. Names must be unique in a cluster. The value can contain only numbers, letters, spaces, hyphens (-), and underscores (_) and cannot start with a space. The value can contain 1 to 64 characters.
uri	String	The path of a custom automation script. Set this parameter to an OBS bucket path or a local VM path. • OBS bucket path: Enter a script path, for example, **obs://*XXX*/scale.sh**. • Local VM path: Enter a script path. The script path must start with a slash (/) and end with **.sh**.

Parameter	Type	Description
parameters	String	Parameters of a custom automation script. Multiple parameters are separated by space. The following predefined system parameters can be transferred: ● *${mrs_scale_node_num}*: The number of nodes to be added or removed ● *${mrs_scale_type}*: The scaling type. The value can be **scale_out** or **scale_in**. ● *${mrs_scale_node_hostnames}*: Host names of the nodes to be added or removed ● *${mrs_scale_node_ips}*: IP addresses of the nodes to be added or removed ● *${mrs_scale_rule_name}*: The name of the rule that triggers scaling. Other user-defined parameters are used in the same way as those of common shell scripts. Parameters are separated by spaces.
nodes	Array of strings	The name of the node group where the custom automation script is executed.
active_master	Boolean	Whether the custom automation script runs only on the active master node. The default value is **false**, indicating that the custom automation script can run on all master nodes.

Parameter	Type	Description
fail_action	String	Whether to continue executing subsequent scripts and creating a cluster after the custom automation script fails to execute. Notes: • You are advised to set this parameter to **continue** in the commissioning phase so the cluster can continue to be installed and started no matter whether the custom automation script is executed successfully. • The scale-in operation cannot be undone. **fail_action** must be set to **continue** for the scripts that are executed after scale-in. Possible values: • **continue**: Continue to execute subsequent scripts. • **errorout**: Stop the action.
action_stage	String	The time when a script is executed. Possible values: • **before_scale_out**: before scale-out • **before_scale_in**: before scale-in • **after_scale_out**: after scale-out • **after_scale_in**: after scale-in

Status code: 400

Table 5-47 Response body parameters

Parameter	Type	Description
error_code	String	Error codes.
error_msg	String	The error message.

Example Request

None

Example Response

Status code: 200

Auto scaling policies are displayed.

```
[ {
  "node_group_name" : "task_node_analysis_group",
  "resource_pool_name" : "default",
  "auto_scaling_policy" : {
    "auto_scaling_enable" : true,
    "min_capacity" : 0,
    "max_capacity" : 1,
    "resources_plans" : [ {
      "period_type" : "daily",
      "start_time" : "12:00",
      "end_time" : "13:00",
      "min_capacity" : 2,
      "max_capacity" : 3
    } ],
    "rules" : [ {
      "name" : "default-expand-1",
      "description" : "",
      "adjustment_type" : "scale_out",
      "cool_down_minutes" : 5,
      "scaling_adjustment" : 1,
      "trigger" : {
        "metric_name" : "YARNAppRunning",
        "metric_value" : 100,
        "comparison_operator" : "GTOE",
        "evaluation_periods" : 1
      }
    } ]
  }
} ]
```

Status Codes

See **Status Codes**.

Error Codes

See **Error Codes**.

5.4 Cluster HDFS File API

5.4.1 Obtaining the List of Files from a Specified Directory

Function

This API is used to obtain the list of files from a specified directory in an MRS cluster.

URI

- Format

  GET /v2/{project_id}/clusters/{cluster_id}/files
- Parameter description

Table 5-48 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	The project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.
cluster_id	Yes	String	The cluster ID. For details about how to obtain the cluster ID, see **Obtaining a Cluster ID**.
path	Yes	String	The file directory. For example, to access the **/tmp/test** directory list, the value must be a directory. The overall URI is as follows: **/v2/{project_id}/clusters/{cluster_id}/files?path=%2Ftmp%2Ftest** A single-level directory must comply with the following rules: 1. The directory path cannot be left blank. 2. The value cannot start or end with a period (.). 3. The value cannot contain the following characters: /:*?"<>|\;&,'`!{}[]$%+ 4. The value cannot exceed 255 characters.
offset	No	String	The pagination parameter. The file list is queried from the offset. The default value is **0**.
limit	No	String	The pagination parameter, indicating the maximum number of records on a page. The default value is **100** and the maximum value is **1000**.

Parameter	Mandatory	Type	Description
sort_key	No	String	The list is sorted by this attribute. The default value is **path_suffix**. The following attributes are supported: • **path_suffix**: file or directory name • **length**: file size • **modification_time**: modification time
order	No	String	The list sorting mode. Possible values: • **desc**: Files are displayed in descending order. • **asc**: Files are displayed in ascending order. The default value is **desc**.

Request Parameters

Request parameters

None

Response Parameters

Table 5-49 Response parameters

Parameter	Type	Description
total_count	Integer	The total number of files, which is irrelevant to pagination.
files	Array of FileStatusV2 objects	The file list. For details, see **Table 5-50**.

Table 5-50 FileStatus parameters

Parameter	Type	Description
path_suffix	String	The file name extension in the current directory. For example, if you obtain the **/tmp/test** file in the **/tmp** directory, the value of **path_suffix** is **test**.

Parameter	Type	Description
owner	String	The file owner.
group	String	The file owner group.
permission	String	The permission information.
replication	Integer	The number of replicas.
block_size	Integer	The block size.
length	Integer	The file length.
type	String	The file type. Possible values: ● **FILE**: file ● **DIRECTORY**: directory
children_num	Integer	The number of files in the directory.
access_time	Long	The file access time.
modification_time	Long	The file modification time.

Example

- Example request
  ```
  GET /v2/{project_id}/clusters/{cluster_id}/files?
  path={directory}&offset={offset}&limit={limit}&sort_key={sort_key}&order={order}
  ```
- Example response
  - Example of a successful response
    ```
    {
       "total_count": 2,
       "files": [
          {
             "access_time": 0,
             "block_size": 0,
             "children_num": 0,
             "group": "hadoop",
             "length": 0,
             "modification_time": 1587179516623,
             "owner": "hdfs",
             "path_suffix": "app-logs",
             "permission": "777",
             "replication": 0,
             "type": "DIRECTORY"
          },
          {
             "access_time": 1587267212761,
             "block_size": 134217728,
             "children_num": 0,
             "group": "hadoop",
             "length": 23666188,
             "modification_time": 1587222156003,
             "owner": "root",
             "path_suffix": "data-m-00000",
             "permission": "644",
             "replication": 3,
             "type": "FILE"
          }
    ```

```
  ]
}
```

Status Codes

See **Status Codes**.

Error Codes

See **Error Codes**.

5.5 SQL APIs

5.5.1 Submitting a SQL Statement

Function

This API is used to submit and execute a SQL statement in an MRS cluster.

URI

- Format
  POST /v2/{project_id}/clusters/{cluster_id}/sql-execution
- Parameter description

Table 5-51 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	The project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.
cluster_id	Yes	String	The cluster ID. For details about how to obtain the cluster ID, see **Obtaining a Cluster ID**.

Request Parameters

Table 5-52 Request parameters

Parameter	Mandatory	Type	Description
sql_type	Yes	String	The SQL type. Currently, only the SQL of the presto type is supported. **NOTE** • The SQL of the presto type can be submitted and executed only in clusters containing the Presto component. • This parameter is available only for MRS 2.0.6.1, 2.1.0.7, 3.1.2, or later clusters.
sql_content	Yes	String	The SQL statement to be executed. **NOTE** Currently, only a single SQL statement can be executed at a time, and the statement cannot contain a semicolon (;).
database	No	String	The database where the SQL statement to be executed is located. The default value is **default**.
archive_path	No	String	The directory for storing the dumped SQL execution results. **NOTE** Only the **SELECT** statement dumps query results. Currently, the query results can be dumped only to OBS.

Response Parameters

Status code: 200

Table 5-53 Response parameters

Parameter	Type	Description
id	String	The execution ID of a SQL statement. The ID is generated only when the **SELECT**, **SHOW**, or **DESC** statement is executed. For other operations, the ID is empty.

Parameter	Type	Description
message	String	The error message.
statement	String	The SQL statement to be executed.
status	String	The execution status of a SQL statement. Possible values: • **QUEUED** • **WAITING_FOR_RESOURCES** • **PLANNING** • **STARTING** • **RUNNING** • **FINISHING** • **FINISHED** • **FAILED**
result_location	String	The path for archiving the final results of the SQL query statement. **NOTE** Only the **SELECT** statement dumps the SQL execution results to **result_location**.
content	Array<Array<String>>	The execution result of the SQL statement. **NOTE** Only non-SELECT statements return results in **content**. If the SQL statement does not return results, **content** is empty.

Status code: 400

Table 5-54 Response body parameters

Parameter	Type	Description
error_code	String	The error code.
error_msg	String	The error message.

Example Request

Example request for submitting a SQL statement

```
POST https://{endpoint}/v2/{project_id}/clusters/{cluster_id}/sql-execution

{
  "sql_type" : "presto",
  "sql_content" : "show tables",
  "database" : "default",
```

```
 "archive_path" : "obs://my-bucket/path"
}
```

Example Response

Status code: 200

The SQL statement is submitted successfully.

```
{
 "id" : "20190909_011820_00151_xxxxx",
 "statement" : "show tables",
 "status" : "FINISHED",
 "result_location" : " obs://my_bucket/uuid_date/xxxx.csv",
 "content" : [ [ "t1", null ], [ null, "t2" ], [ null, "t3" ] ]
}
```

Status code: 400

Failed to submit the SQL statement.

```
{
 "error_code" : "MRS.0011",
  "message": "Failed to submit SQL to the executor. The cluster ID is xxxx"
}
```

Status Codes

See **Status Codes**.

Error Codes

See **Error Codes**.

5.5.2 Querying SQL Results

Function

This API is used to query the execution result of a SQL statement in an MRS cluster.

URI

- Format

  GET /v2/{project_id}/clusters/{cluster_id}/sql-execution/{sql_id}

- Parameter description

  Table 5-55 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	The project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.

Parameter	Mandatory	Type	Description
cluster_id	Yes	String	Cluster ID. For details on how to obtain the cluster ID, see **Obtaining a Cluster ID**.
sql_id	Yes	String	The execution ID of a SQL statement, that is, **sql_id** in the return result in **Submitting a SQL Statement**.

Request Parameters

Request parameters

None

Response Parameters

Status code: 200

Table 5-56 Response parameters

Parameter	Type	Description
id	String	The execution ID of a SQL statement. The ID is generated only when the **SELECT**, **SHOW**, or **DESC** statement is executed. For other operations, the ID is empty.
message	String	The error message.
statement	String	The SQL statement to be executed.
status	String	The execution status of a SQL statement. Possible values: • **QUEUED** • **WAITING_FOR_RESOURCES** • **PLANNING** • **STARTING** • **RUNNING** • **FINISHING** • **FINISHED** • **FAILED**

Parameter	Type	Description
result_location	String	The path for archiving the final results of the SQL query statement. **NOTE** Only the **SELECT** statement dumps the SQL execution results to **result_location**.
content	Array	The execution result of the SQL statement. **NOTE** Only non-SELECT statements return results in **content**. If the SQL statement does not return results, **content** is empty.

Status code: 400

Table 5-57 Response body parameters

Parameter	Type	Description
error_code	String	The error code.
error_msg	String	The error message.

Example Request

Example request for querying the SQL execution result

```
GET https://{endpoint}/v2/{project_id}/clusters/{cluster_id}/sql-execution/{sql_id}
```

Example Response

Status code: 200

Querying the SQL execution result is successful.

```
{
  "id" : "20190909_011820_00151_xxxxx",
  "statement" : "show tables",
  "status" : "FINISHED",
  "result_location" : "obs://my_bucket/uuid_date/xxxx.csv",
  "content" : [ [ "t1", null ], [ null, "t2" ], [ null, "t3" ] ]
}
```

Status code: 400

Failed to query the SQL execution result.

```
{
  "error_code" : "MRS.0011",
  "message": "Failed to submit SQL to the executor. The cluster ID is xxxx"
}
```

Status Codes

See **Status Codes**.

Error Codes

See **Error Codes**.

5.5.3 Canceling a SQL Execution Task

Function

This API is used to cancel the execution task of a SQL statement in an MRS cluster.

URI

- Format
  POST /v2/{project_id}/clusters/{cluster_id}/sql-execution/{sql_id}/cancel
- Parameter description

Table 5-58 URI parameters

Parameter	Mandatory	Description
project_id	Yes	The project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.
cluster_id	Yes	The cluster ID. For details about how to obtain the cluster ID, see **Obtaining a Cluster ID**.
sql_id	Yes	The execution ID of a SQL statement, that is, **sql_id** in the return result in **Submitting a SQL Statement**.

Request Parameters

Request parameters

None

Response Parameters

Table 5-59 Response parameters

Parameter	Type	Description
status	String	The result of canceling the execution of the SQL statement. Possible values: • **SUCCEED**: successful • **FAILED**: failed **NOTE** By default, **SUCCEED** is returned. **SUCCEED** is also returned when the task is completed. Only when the running SQL statement fails to be canceled, **FAILED** is returned.
message	String	The error message.

Example

- Example request
  ```
  POST https://{endpoint}/v2/{project_id}/clusters/{cluster_id}/sql-execution/{sql_id}/cancel
  ```
- Example response
  - Example of a successful response
    ```
    {
       "status":"SUCCEED"
    }
    ```
  - Example of a failed response
    ```
    {
       "status":"FAILED",
       "message":"Cancel sql error"
    }
    ```

Status Codes

See **Status Codes**.

Error Codes

See **Error Codes**.

5.6 Agency Management

5.6.1 Querying the Mapping Between a User (Group) and an IAM Agency

Function

This API is used to obtain details about the mapping between a user or user group and an IAM agency.

URI

- URI format

  GET /v2/{project_id}/clusters/{cluster_id}/agency-mapping

- Parameter description

  Table 5-60 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	The project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.
cluster_id	Yes	String	The cluster ID. For details about how to obtain the cluster ID, see **Obtaining a Cluster ID**.

Request Parameters

Request parameters

None

Response Parameters

Status code: 200

Table 5-61 Response parameter

Parameter	Type	Description
agency_mappings	Array of AgencyMapping objects	The mapping between users or user groups and agencies. For details, see **Table 5-62**.

Table 5-62 agency_mappings parameters

Parameter	Type	Description
agency	String	The name of the IAM agency bound to the mapping.
identifier_type	String	The agency type. Possible values: • **User**: indicates that the mapping is for users. The user name list is displayed in **identifiers**. • **Group**: indicates that the mapping is for user groups. The user group name list is displayed in **identifiers**.
identifiers	Array of String	The list of users or user groups mapped to the IAM agency. Log in to the IAM management console and choose **Users** or **User Groups** in the left navigation pane to obtain the user or user group name list.
agency_id	String	The unique ID of the agency bound to the mapping. Log in to the IAM management console and choose **Agencies** in the left navigation pane. On the **Agencies** page that is displayed, move your cursor over the agency name to obtain the agency ID.

Status code: 400

Table 5-63 Response parameters

Parameter	Type	Description
error_code	String	The error code.
error_msg	String	The error message.

Example Request

None

Example Response

Status code: 200

Querying the mapping between a user or user group and an IAM agency is successful.

```
{
  "agency_mappings" : [ {
```

```
    "agency" : "agency01",
    "identifier_type" : "User",
    "identifiers" : [ "user01" ],
    "agency_id" : "092adc623c00d2ea4fdac01d4b637f0b"
  }, {
    "agency" : "agency02",
    "identifier_type" : "User",
    "identifiers" : [ "user02" ],
    "agency_id" : "065239307e00d3ae4f80c01d4bdafdfd"
  }, {
    "agency" : "groupAgency",
    "identifier_type" : "Group",
    "identifiers" : [ "group01", "group02", "group03" ],
    "agency_id" : "08467a446200d5ac4ff9c01d56670c3b"
  } ]
}
```

Status Codes

See **Status Codes**.

Error Codes

See **Error Codes**.

5.6.2 Updating the Mapping Between a User (Group) and an IAM Agency

Function

This API is used to update the mapping between a user or user group and an IAM agency.

URI

- URI format

  PUT /v2/{project_id}/clusters/{cluster_id}/agency-mapping

- Parameter description

  Table 5-64 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	The project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.
cluster_id	Yes	String	The cluster ID. For details about how to obtain the cluster ID, see **Obtaining a Cluster ID**.

Request Parameters

Table 5-65 Request parameter

Parameter	Mandatory	Type	Description
agency_mappings	Yes	Array of AgencyMapping objects	The mapping between users or user groups and agencies. For details, see **Table 5-66**.

Table 5-66 agency_mappings parameters

Parameter	Mandatory	Type	Description
agency	Yes	String	The name of the IAM agency bound to the mapping.
identifier_type	Yes	String	The agency type. Possible values: • **User**: indicates that the mapping is for users. The user name list is displayed in **identifiers**. • **Group**: indicates that the mapping is for user groups. The user group name list is displayed in **identifiers**.
identifiers	Yes	Array of String	The list of users or user groups mapped to the IAM agency. Log in to the IAM management console and choose **Users** or **User Groups** in the left navigation pane to obtain the user or user group name list.
agency_id	Yes	String	The ID of the agency bound to the mapping. Log in to the IAM management console and choose **Agencies** in the left navigation pane. On the **Agencies** page that is displayed, move your cursor over the agency name to obtain the agency ID.

Response Parameters

Status code: 200

Table 5-67 Response body parameter

Parameter	Type	Description
result	String	The operation result of the request for updating a mapping. Possible values: • **succeeded**: The operation is successful. • **failed**: The operation failed.

Status code: 400

Table 5-68 Response body parameters

Parameter	Type	Description
error_code	String	The error code.
error_msg	String	The error description.

Example

- Example request

  Example request for updating the mapping between a user or user group and an IAM agency

```
PUT https://{endpoint}/v2/{project_id}/clusters/{cluster_id}/agency-mapping

{
  "agency_mappings" : [ {
    "agency" : "agency01",
    "identifier_type" : "User",
    "identifiers" : [ "test" ],
    "agency_id" : "xxxx"
  } ]
}
```

- Example response

  Updating the mapping between a user or user group and an IAM agency is successful.

```
{
  "result" : "succeeded"
}
```

Status Codes

See **Status Codes**.

Error Codes

See **Error Codes**.

6 API V1.1

6.1 Cluster Management APIs

6.1.1 Creating a Cluster and Executing a Job

Function

This API is used to create an MRS cluster and submit a job in the cluster. This API is incompatible with Sahara.

A maximum of 10 clusters can be concurrently created.

Before using the API, you need to obtain the resources listed in **Table 6-1**.

Table 6-1 Obtaining resources

Resource	How to Obtain
VPC	See operation instructions in in the *VPC API Reference*.
Subnet	See operation instructions in in the *VPC API Reference*.
Key Pair	See operation instructions in in the *ECS API Reference*.
Zone	See **Endpoints** for details about regions and AZs.
Version	Currently, MRS 2.1.1, 3.0.5, and 3.1.0 are supported.

Resource	How to Obtain
Component	• MRS 3.1.0 supports the following components: – An analysis cluster contains the following components: Hadoop, Spark2x, HBase, Hive, Hue, Flink, Oozie, ZooKeeper, Ranger, Tez, Impala, Presto, and Kudu. – A streaming cluster contains the following components: Kafka, Flume, ZooKeeper, and Ranger. – A hybrid cluster contains the following components: Hadoop, Spark2x, HBase, Hive, Hue, Flink, Oozie, ZooKeeper, Ranger, Tez, Impala, Presto, Kudu, Kafka, and Flume. – A custom cluster contains the following components: Hadoop, Spark2x, HBase, Hive, Hue, Kafka, Flume, Flink, Oozie, ZooKeeper, Ranger, Tez, Impala, Presto, ClickHouse, and Kudu. • MRS 3.0.5 supports the following components: – An analysis cluster contains the following components: Hadoop, Spark2x, HBase, Hive, Hue, Loader, Flink, Oozie, ZooKeeper, Ranger, Tez, Impala, Presto, Kudu, and Alluxio. – A streaming cluster contains the following components: Kafka, Storm, Flume, ZooKeeper, and Ranger. – A hybrid cluster contains the following components: Hadoop, Spark2x, HBase, Hive, Hue, Loader, Flink, Oozie, ZooKeeper, Ranger, Tez, Impala, Presto, Kudu, Alluxio, Kafka, Storm, and Flume. – A custom cluster contains the following components: Hadoop, Spark2x, HBase, Hive, Hue, Loader, Kafka, Storm, Flume, Flink, Oozie, ZooKeeper, Ranger, Tez, Impala, Presto, ClickHouse, Kudu, and Alluxio.

URI

POST /v1.1/{project_id}/run-job-flow

Table 6-2 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	The project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.

Request Parameters

Table 6-3 Request parameters

Parameter	Mandatory	Type	Description
cluster_version	Yes	String	The cluster version. Possible values: • MRS 3.0.5 • MRS 3.1.0
cluster_name	Yes	String	The cluster name, which must be unique. A cluster name can contain only 1 to 64 characters. Only letters, numbers, hyphens (-), and underscores (_) are allowed.
master_node_num	Yes	Integer	The number of master nodes. If cluster HA is enabled, set this parameter to **2**. If cluster HA is disabled, set this parameter to **1**. This parameter cannot be set to **1** in MRS 3.*x*.
core_node_num	Yes	Integer	The number of core nodes. Value range: 1 to 500 The default maximum number of core nodes is **500**. If more than 500 core nodes are required, apply for a higher quota.
billing_type	Yes	Integer	The cluster billing mode.
data_center	Yes	String	The information about the region where the cluster is located. For details, see **Endpoints**.
vpc	Yes	String	The name of the VPC where the subnet is located. Obtain the VPC name by performing the following operations on the VPC management console: 1. Log in to the management console. 2. Click **Virtual Private Cloud** and select **Virtual Private Cloud** from the left list. On the **Virtual Private Cloud** page, obtain the VPC name from the list.

Parameter	Mandatory	Type	Description
master_node_size	Yes	String	The instance specifications of master nodes, for example, **c3.4xlarge.2.linux.bigdata**. The host specifications supported by MRS are determined by CPU, memory, and disk space. You are advised to obtain the specifications supported by the corresponding version in the corresponding region from the cluster creation page on the MRS console.
core_node_size	Yes	String	The instance specifications of core nodes, for example, **c3.4xlarge.2.linux.bigdata**. You are advised to obtain the specifications supported by the corresponding version in the corresponding region from the cluster creation page on the MRS console.
component_list	Yes	Array of component_list objects	The list of service components to be installed. For details about the parameters, see **Table 6-4**.
available_zone_id	Yes	String	The AZ ID.
vpc_id	Yes	String	The ID of the VPC where the subnet is located. Obtain the VPC ID by performing the following operations on the VPC management console: 1. Log in to the management console. 2. Click **Virtual Private Cloud** and select **Virtual Private Cloud** from the left list. On the **Virtual Private Cloud** page, obtain the VPC ID from the list.

Parameter	Mandatory	Type	Description
subnet_id	Yes	String	The subnet ID. Obtain the subnet ID by performing the following operations on the VPC management console: 1. Log in to the management console. 2. Click **Virtual Private Cloud** and select **Virtual Private Cloud** from the left list. 3. Locate the row that contains the target VPC and click the number in the **Subnets** column to view the subnet information. 4. Click the subnet name to obtain the network ID. At least one of **subnet_id** and **subnet_name** must be configured. If the two parameters are configured but do not match the same subnet, the cluster fails to create. **subnet_id** is recommended.

Parameter	Mandatory	Type	Description
subnet_name	Yes	String	The subnet name. Obtain the subnet name by performing the following operations on the VPC management console: 1. Log in to the management console. 2. Click **Virtual Private Cloud** and select **Virtual Private Cloud** from the left list. 3. Locate the row that contains the target VPC and click the number in the **Subnets** column to obtain the subnet name. At least one of **subnet_id** and **subnet_name** must be configured. If the two parameters are configured but do not match the same subnet, the cluster fails to create. If only **subnet_name** is configured and subnets with the same name exist in the VPC, the first subnet name in the VPC is used when a cluster is created. **subnet_id** is recommended.
security_groups_id	No	String	The ID of the security group configured for the cluster. ● If this parameter is left blank, MRS automatically creates a security group, whose name starts with **mrs_{cluster_name}**. ● If this parameter is not left blank, a fixed security group is used to create a cluster. The transferred ID must be the security group ID owned by the current tenant. The security group must include an inbound rule in which all protocols and all ports are allowed and the source is the IP address of the specified node on the management plane.

Parameter	Mandatory	Type	Description
add_jobs	No	Array of add_jobs objects	Jobs can be submitted when a cluster is created. Currently, only one job can be created. For details about the parameters, see **Table 6-5**.
volume_size	No	Integer	The data disk storage space of master and core nodes. To increase the data storage capacity, you can add disks when creating a cluster. Select a proper disk storage space based on the following application scenarios: • Storage-compute decoupling: Data is stored in the OBS system. Costs of clusters are relatively low but computing performance is poor. The clusters can be deleted at any time. It is recommended when data computing is infrequently performed. • Storage-compute integration: Data is stored in the HDFS system. Costs of clusters are relatively high but computing performance is good. The clusters cannot be deleted in a short term. It is recommended when data computing is frequently performed. Value range: 100 GB to 32,000 GB. When assigning a value to this parameter, you only need to pass a number, without the unit. This parameter is not recommended. For details, see the description of the **volume_type** parameter.

Parameter	Mandatory	Type	Description
volume_type	No	String	The data disk storage type of master and core nodes. Currently, SATA, SAS, SSD, and GPSSD are supported. Disk parameters can be represented by **volume_type** and **volume_size**, or multi-disk parameters. If the **volume_type** and **volume_size** parameters coexist with the multi-disk parameters, the system reads the **volume_type** and **volume_size** parameters first. You are advised to use the multi-disk parameters. Possible values: ● **SATA**: common I/O ● **SAS**: high I/O ● **SSD**: ultra-high I/O ● **GPSSD**: general-purpose SSD
master_data_volume_type	No	String	This parameter is a multi-disk parameter, indicating the data disk storage type of the master node. Currently, SATA, SAS, SSD, and GPSSD are supported.
master_data_volume_size	No	Integer	This parameter is a multi-disk parameter, indicating the data disk storage space of master nodes. To increase the data storage capacity, you can add disks when creating a cluster. Value range: 100 GB to 32,000 GB. When assigning a value to this parameter, you only need to pass a number, without the unit.
master_data_volume_count	No	Integer	This parameter is a multi-disk parameter, indicating the number of data disks of the master nodes. The value can only be **1**.
core_data_volume_type	No	String	This parameter is a multi-disk parameter, indicating the data disk storage type of core nodes. Currently, SATA, SAS, SSD, and GPSSD are supported.

Parameter	Mandatory	Type	Description
core_data_volume_size	No	Integer	This parameter is a multi-disk parameter, indicating the data disk storage space of core nodes. To increase the data storage capacity, you can add disks when creating a cluster. Value range: 100 GB to 32,000 GB. When assigning a value to this parameter, you only need to pass a number, without the unit.
core_data_volume_count	No	Integer	This parameter is a multi-disk parameter, indicating the number of data disks of the core nodes. Value range: 1 to 10
task_node_groups	No	Array of task_node_groups objects	The list of task nodes. For details about the parameters, see **Table 6-6**.
bootstrap_scripts	No	Array of bootstrap_script objects	The Bootstrap action script information. For details about the parameters, see **Table 6-8**.
node_public_cert_name	No	String	The name of a key pair. You can use a key pair to log in to a cluster node. If **login_mode** is set to **1**, the request body contains the **node_public_cert_name** field.
cluster_admin_secret	No	String	The password of the MRS Manager administrator. A password must meet the following complexity requirements: • Must contain 8 to 26 characters. • Cannot be the username or the username spelled backwards. • Must contain every type of the following: – Lowercase letters – Uppercase letters – Numbers – Special characters (!@$%^-_=+[{}]:,./?)

Parameter	Mandatory	Type	Description
cluster_master_secret	Yes	String	The password of user **root** for logging in to a cluster node. If **login_mode** is set to **0**, the request body contains the **cluster_master_secret** field. A password must meet the following complexity requirements: • Must be 8 to 26 characters long. • Must contain every type of the following: uppercase letters, lowercase letters, numbers, and special characters (!@$%^-_=+[{}]:,./?), but must not contain spaces. • Cannot be the username or the username spelled backwards.
safe_mode	Yes	Integer	The running mode of an MRS cluster. Possible values: • **0**: normal cluster. In a normal cluster, Kerberos authentication is disabled, and users can use all functions provided by the cluster. • **1**: security cluster. In a security cluster, Kerberos authentication is enabled, and common users cannot use the file management and job management functions of an MRS cluster or view cluster resource usage and the job records of Hadoop and Spark. To use these functions, the users must obtain the relevant permissions from the MRS Manager administrator.
tags	No	Array of tag objects	The cluster tags. For details about the parameters, see **Table 6-9**. • A cluster allows a maximum of 10 tags. A tag name (key) must be unique in a cluster. • A tag key or value cannot contain the following special characters: =*<>\,|/

Parameter	Mandatory	Type	Description
cluster_type	No	Integer	The cluster type. Possible values: ● **0**: analysis cluster ● **1**: streaming cluster The default value is **0**.
log_collection	No	Integer	Whether to collect logs when cluster creation fails. The default value is **1**, indicating that OBS buckets are created only for collecting logs when an MRS cluster fails to create. Possible values: ● **0**: Do not collect logs. ● **1**: Collect logs.
enterprise_project_id	No	String	The enterprise project ID. When you create a cluster, associate the enterprise project ID with the cluster. The default value is **0**, indicating the **default** enterprise project.
login_mode	No	Integer	The cluster login mode. The default value is **1**. ● If **login_mode** is set to **0**, the request body contains the **cluster_master_secret** field. ● If **login_mode** is set to **1**, the request body contains the **node_public_cert_name** field. Possible values: ● **0**: password ● **1**: key pair

Parameter	Mandatory	Type	Description
node_groups	No	Array of NodeGroupV11 objects	The list of nodes. For details about the parameters, see **Table 6-10**. Note: Configure either this parameter or the following parameters: **master_node_num**, **master_node_size**, **core_node_num**, **core_node_size**, **master_data_volume_type**, **master_data_volume_size**, **master_data_volume_count**, **core_data_volume_type**, **core_data_volume_size**, **core_data_volume_count**, **volume_type**, **volume_size**, **task_node_groups**

Table 6-4 component_list parameter

Parameter	Mandatory	Type	Description
component_name	Yes	String	The component name. For details, see the component information in **Table 6-1**.

Table 6-5 add_jobs parameters

Parameter	Mandatory	Type	Description
job_type	Yes	Integer	The job type code. Possible values: ● **1**: MapReduce ● **2**: Spark ● **3**: Hive Script ● **4**: HiveQL (not supported currently) ● **5**: DistCp for importing and exporting data (not supported currently) ● **6**: Spark Script ● **7**: Spark SQL for submitting Spark SQL statements (not supported currently) **NOTE** Spark and Hive jobs can be created only in clusters that where Spark and Hive are installed.
job_name	Yes	String	The job name. The value can contain 1 to 64 characters. Only letters, numbers, hyphens (-), and underscores (_) are allowed. **NOTE** Identical job names are allowed but not recommended.

Parameter	Mandatory	Type	Description
jar_path	No	String	The path of the JAR or SQL file for program execution. The value must meet the following requirements: ● The value contains a maximum of 1,023 characters. It cannot contain special characters (;|&>,<'$) and cannot be left blank or all spaces. ● Files can be stored in HDFS or OBS. The path varies depending on the file system. – OBS: The path must start with **s3a://**. Files or programs encrypted by KMS are not supported. – HDFS: The path starts with a slash (**/**). ● Spark Script must end with **.sql** while MapReduce and Spark Jar must end with **.jar**. **sql** and **jar** are case-insensitive.
arguments	No	String	The key parameter for program execution. The parameter is specified by the function of the user's program. MRS is only responsible for loading the parameter. The parameter can contain 0 to 150,000 characters, but special characters (;|&>'<$) are not allowed.
input	No	String	The data input path. Files can be stored in HDFS or OBS. The path varies depending on the file system. ● OBS: The path must start with **s3a://**. Files or programs encrypted by KMS are not supported. ● HDFS: The path starts with a slash (**/**). The value can contain 0 to 1,023 characters, but special characters (;|&>'<$) are not allowed.

Parameter	Mandatory	Type	Description
output	No	String	The data output path. Files can be stored in HDFS or OBS. The path varies depending on the file system. ● OBS: The path must start with **s3a://**. ● HDFS: The path starts with a slash (**/**). If the specified path does not exist, the system will automatically create it. The value can contain 0 to 1,023 characters, but special characters (;|&>'<$) are not allowed.
job_log	No	String	The path for storing job logs that record job running status. Files can be stored in HDFS or OBS. The path varies depending on the file system. ● OBS: The path must start with **s3a://**. ● HDFS: The path starts with a slash (**/**). The value can contain 0 to 1,023 characters, but special characters (;|&>'<$) are not allowed.
shutdown_cluster	No	Boolean	Whether to terminate the cluster after the job execution is complete. ● **true**: Terminate the cluster. ● **false**: Do not terminate the cluster.
file_action	No	String	The action to be performed on a file. Possible values: ● **import** ● **export**
submit_job_once_cluster_run	Yes	Boolean	● **true**: Submit a job during cluster creation. ● **false**: Submit a job after the cluster is created. Set this parameter to **true** in this example.

Parameter	Mandatory	Type	Description
hql	No	String	The HQL script statement.
hive_script_path	Yes	String	The SQL program path. This parameter is required by Spark Script and Hive Script jobs only, and the value must meet the following requirements: ● The value contains a maximum of 1,023 characters. It cannot contain special characters (;| &><'$) and cannot be left blank or all spaces. ● Files can be stored in HDFS or OBS. The path varies depending on the file system. – OBS: The path must start with **s3a://**. Files or programs encrypted by KMS are not supported. – HDFS: The path starts with a slash (**/**). ● Ends with **.sql**. **sql** is case-insensitive.

Table 6-6 task_node_groups parameters

Parameter	Mandatory	Type	Description
node_num	Yes	Integer	The number of task nodes. The value ranges from 0 to 500, and the total number of core and task nodes cannot exceed 500.
node_size	Yes	String	The instance specifications of task nodes, for example, **c6.4xlarge.4.linux.mrs**. Obtain the instance specifications of the corresponding version in the corresponding region from the cluster createion page of the MRS management console.

Parameter	Mandatory	Type	Description
data_volume_type	Yes	String	The data disk storage type of task nodes. Possible values: • **SATA**: common I/O • **SAS**: high I/O • **SSD**: ultra-high I/O • **GPSSD**: general-purpose SSD
data_volume_count	Yes	Integer	The number of data disks of a task node. Value range: 0 to 10
data_volume_size	Yes	Integer	The data disk storage space of a task node. Value range: 100 GB to 32,000 GB You only need to enter a number and do not need to enter the unit GB.
auto_scaling_policy	No	auto_scaling_policy object	The auto scaling policy. For details, see **Table 6-7**.

Table 6-7 auto_scaling_policy parameters

Parameter	Mandatory	Type	Description
auto_scaling_enable	Yes	Boolean	Whether to enable the auto scaling policy.
min_capacity	Yes	Integer	The minimum number of nodes reserved in the node group. Value range: 0 to 500
max_capacity	Yes	Integer	The maximum number of nodes in the node group. Value range: 0 to 500
resources_plans	No	Array of resources_plan objects	The resource plan list. For details, see **Table 6-12**. If this parameter is left blank, the resource plan is disabled. When **auto_scaling_enable** is set to **true**, either this parameter or **rules** must be configured.

Parameter	Mandatory	Type	Description
exec_scripts	No	Array of scale_script objects	The list of custom scaling automation scripts. For details, see **Table 6-15**. If this parameter is left blank, a hook script is disabled.
rules	No	Array of rules objects	The list of auto scaling rules. For details, see **Table 6-13**. When **auto_scaling_enable** is set to **true**, either this parameter or **resources_plans** must be configured.

Table 6-8 bootstrap_script parameters

Parameter	Mandatory	Type	Description
name	Yes	String	The name of a bootstrap action script, which must be unique in a cluster. The value can contain only numbers, letters, spaces, hyphens (-), and underscores (_) and cannot start with a space. The value can contain 1 to 64 characters.

Parameter	Mandatory	Type	Description
uri	Yes	String	The path of a Bootstrap action script. Set this parameter to an OBS bucket path or a local VM path. ● OBS bucket path: Enter a script path. For example, enter the path of the public sample script provided by MRS. Example: **s3a://bootstrap/presto/presto-install.sh**. If **dualroles** is installed, the parameter of the **presto-install.sh** script is **dualroles**. If **worker** is installed, the parameter of the **presto-install.sh** script is **worker**. Based on the Presto usage habit, you are advised to install **dualroles** on the active Master nodes and **worker** on the Core nodes. ● Local VM path: Enter a script path. The script path must start with a slash (/) and end with **.sh**.
parameters	No	String	The bootstrap action script parameters.
nodes	Yes	Array of strings	The type of a node where the bootstrap action script is executed. The value can be **Master**, **Core**, or **Task**. The node type must be represented in lowercase letters.
active_master	No	Boolean	Whether the bootstrap action script runs only on active master nodes. The default value is **false**, indicating that the Bootstrap action script can run on all Master nodes.

Parameter	Mandatory	Type	Description
before_component_start	No	Boolean	Time when the bootstrap action script is executed. Currently, the following two options are available: **Before component start** and **After component start** The default value is **false**, indicating that the bootstrap action script is executed after the component is started.
fail_action	Yes	String	Whether to continue executing subsequent scripts and creating a cluster after the Bootstrap action script fails to be executed. ● **continue**: Continue to execute subsequent scripts. ● **errorout**: Stop the action. The default value is **errorout**, indicating that the action is stopped. **NOTE** You are advised to set this parameter to **continue** in the commissioning phase so that the cluster can continue to be installed and started no matter whether the bootstrap action is successful.
action_stages	No	Array of strings	Select the time when the bootstrap action script is executed. ● **BEFORE_COMPONENT_FIRST_START**: before initial component starts ● **AFTER_COMPONENT_FIRST_START**: after initial component starts ● **BEFORE_SCALE_IN**: before scale-in ● **AFTER_SCALE_IN**: after scale-in ● **BEFORE_SCALE_OUT**: before scale-out ● **AFTER_SCALE_OUT**: after scale-out

Table 6-9 tag parameters

Parameter	Mandatory	Type	Description
key	Yes	String	The tag key. • It contains a maximum of 36 Unicode characters and cannot be an empty string. • The tag key cannot start or end with spaces or contain non-printable ASCII characters (0–31) and special characters (=*<> \,|/). • The tag key of a resource must be unique.
value	Yes	String	The value. • The value contains a maximum of 43 Unicode characters and can be an empty string. • The tag value cannot start or end with spaces or contain non-printable ASCII characters (0–31) and special characters (=*<> \,|/).

Table 6-10 NodeGroupV11 parameters

Parameter	Mandatory	Type	Description
group_name	Yes	String	The node group name. Possible values: • **master_node_default_group** • **core_node_analysis_group** • **core_node_streaming_group** • **task_node_analysis_group** • **task_node_streaming_group**
node_num	Yes	Integer	The number of nodes. The value ranges from 0 to 500, and the default value is **0**. The total number of core and task nodes cannot exceed 500.

Parameter	Mandatory	Type	Description
node_size	Yes	String	The instance specifications of cluster nodes, for example, . Obtain the instance specifications of the corresponding version in the corresponding region from the cluster createion page of the MRS management console.
root_volume_size	Yes	String	The system disk storage space of a node.
root_volume_type	Yes	String	The system disk storage type of a node. Possible values: • **SATA**: common I/O • **SAS**: high I/O • **SSD**: ultra-high I/O • **GPSSD**: general-purpose SSD
data_volume_type	Yes	String	The data disk storage type of a node. Possible values: • **SATA**: common I/O • **SAS**: high I/O • **SSD**: ultra-high I/O • **GPSSD**: general-purpose SSD
data_volume_count	Yes	Integer	The number of data disks of a node. Value range: 0 to 10
data_volume_size	Yes	Integer	The data disk storage space of a node. Value range: 100 GB to 32,000 GB
auto_scaling_policy	No	auto_scaling_policy object	The auto scaling rule information. This parameter is available only when **group_name** is set to **task_node_analysis_group** or **task_node_streaming_group**. For details, see **Table 6-11**. For details about the parameters, see **Table 6-7**.

Table 6-11 Node configuration parameters

Parameter	Mandatory	Type	Description
master_node_num	Yes	Integer	The number of master nodes. If cluster HA is enabled, set this parameter to **2**. If cluster HA is disabled, set this parameter to **1**. This parameter cannot be set to **1** in MRS 3.*x*.
master_node_size	Yes	String	The instance specifications of master nodes, for example, . The host specifications supported by MRS are determined by CPU, memory, and disk space. Obtain the instance specifications of the corresponding version in the corresponding region from the cluster createion page of the MRS management console.
core_node_num	Yes	Integer	The number of core nodes. Value range: 1 to 500 A maximum of 500 core nodes are supported by default. If more than 500 core nodes are required, .

Parameter	Mandatory	Type	Description
core_node_size	Yes	String	The instance specifications of core nodes, for example, . Obtain the instance specifications of the corresponding version in the corresponding region from the cluster createion page of the MRS management console.
master_data_volume_type	No	String	This parameter is a multi-disk parameter, indicating the data disk storage type of master nodes. Currently, SATA, SAS, and SSD are supported.
master_data_volume_size	No	Integer	This parameter is a multi-disk parameter, indicating the data disk storage space of master nodes. To increase the data storage capacity, you can add disks when creating a cluster. Value range: 100 GB to 32,000 GB
master_data_volume_count	No	Integer	This parameter is a multi-disk parameter, indicating the number of data disks of the master nodes. The value can only be **1**.

Parameter	Mandatory	Type	Description
core_data_volume_type	No	String	This parameter is a multi-disk parameter, indicating the data disk storage type of core nodes. Currently, SATA, SAS, and SSD are supported.
core_data_volume_size	No	Integer	This parameter is a multi-disk parameter, indicating the data disk storage space of core nodes. To increase the data storage capacity, you can add disks when creating a cluster. Value range: 100 GB to 32,000 GB
core_data_volume_count	No	Integer	This parameter is a multi-disk parameter, indicating the number of data disks of core nodes. Value range: 1 to 10

Parameter	Mandatory	Type	Description
volume_type	No	String	The data disk storage type of the master and core nodes. Currently, SATA, SAS, and SSD are supported. Disk parameters can be represented by **volume_type** and **volume_size**, or multi-disk parameters. If the **volume_type** and **volume_size** parameters coexist with the multi-disk parameters, the system reads the **volume_type** and **volume_size** parameters first. You are advised to use the multi-disk parameters. Possible values: ● **SATA**: common I/O ● **SAS**: high I/O ● **SSD**: ultra-high I/O

Parameter	Mandatory	Type	Description
volume_size	No	Integer	The data disk storage space of master and core nodes. To increase the data storage capacity, you can add disks when creating a cluster. Value range: 100 GB to 32,000 GB This parameter is not recommended. For details, see the description of the **volume_type** parameter.
task_node_groups	No	Array	The list of task nodes. For more parameter description, see **Table 6-6**.

Table 6-12 resources_plan parameters

Parameter	Mandatory	Type	Description
period_type	Yes	String	Cycle type of a resource plan. Currently, only the following cycle type is supported: **daily**
start_time	Yes	String	Start time of a resource plan. The value is in the format of **hour:minute**, indicating that the time ranges from 0:00 to 23:59.
end_time	Yes	String	End time of a resource plan. The value is in the same format as that of **start_time**. The interval between **end_time** and **start_time** must be greater than or equal to 30 minutes.

Parameter	Mandatory	Type	Description
min_capacity	Yes	Integer	Minimum number of the preserved nodes in a node group in a resource plan. Value range: 0 to 500
max_capacity	Yes	Integer	Maximum number of the preserved nodes in a node group in a resource plan. Value range: 0 to 500

Table 6-13 rules parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Name of an auto scaling rule. It contains only 1 to 64 characters. Only letters, digits, hyphens (-), and underscores (_) are allowed. Rule names must be unique in a node group.
description	No	String	Description about an auto scaling rule. It contains a maximum of 1024 characters.
adjustment_type	Yes	String	Auto scaling rule adjustment type. Possible values: • **scale_out**: cluster scale-out • **scale_in**: cluster scale-in
cool_down_minutes	Yes	Integer	Cluster cooling time after an auto scaling rule is triggered, when no auto scaling operation is performed. The unit is minute. Value range: 0 to 10,080. One week is equal to 10,080 minutes.
scaling_adjustment	Yes	Integer	Number of nodes that can be adjusted once. Value range: 1 to 100
trigger	Yes	trigger object	Condition for triggering a rule. For details, see **Table 6-14**.

Table 6-14 trigger parameters

Parameter	Mandatory	Type	Description
metric_name	Yes	String	Metric name. This triggering condition makes a judgment according to the value of the metric. A metric name contains a maximum of 64 characters. **Table 6-16** lists the supported metric names.
metric_value	Yes	String	Metric threshold to trigger a rule The parameter value must be an integer or number with two decimal places only. **Table 6-16** provides value types and ranges corresponding to **metric_name**.
comparison_operator	No	String	Metric judgment logic operator. Possible values: • **LT**: less than • **GT**: greater than • **LTOE**: less than or equal to • **GTOE**: greater than or equal to
evaluation_periods	Yes	Integer	Number of consecutive five-minute periods, during which a metric threshold is reached Value range: 1 to 288

Table 6-15 scale_script parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Name of a custom automation script. It must be unique in a same cluster. The value can contain only digits, letters, spaces, hyphens (-), and underscores (_) and cannot start with a space. The value can contain 1 to 64 characters.

Parameter	Mandatory	Type	Description
uri	Yes	String	Path of a custom automation script. Set this parameter to an OBS bucket path or a local VM path. ● OBS bucket path: Enter a script path manually, for example, **s3a://*XXX*/scale.sh**. ● Local VM path: Enter a script path. The script path must start with a slash (/) and end with **.sh**.
parameters	No	String	Parameters of a custom automation script. ● Multiple parameters are separated by space. ● The following predefined system parameters can be transferred: – *${mrs_scale_node_num}*: Number of the nodes to be added or removed – *${mrs_scale_type}*: Scaling type. The value can be **scale_out** or **scale_in**. – *${mrs_scale_node_hostnames}*: Host names of the nodes to be added or removed – *${mrs_scale_node_ips}*: IP addresses of the nodes to be added or removed – *${mrs_scale_rule_name}*: Name of the rule that triggers auto scaling ● Other user-defined parameters are used in the same way as those of common shell scripts. Parameters are separated by space.
nodes	Yes	Array of string	Type of a node where the custom automation script is executed. The node type can be Master, Core, or Task.

Parameter	Mandatory	Type	Description
active_master	No	Boolean	Whether the custom automation script runs only on the active Master node. The default value is **false**, indicating that the custom automation script can run on all Master nodes.
action_stage	Yes	String	Time when a script is executed. The following four options are supported: ● **before_scale_out**: before scale-out ● **before_scale_in**: before scale-in ● **after_scale_out**: after scale-out ● **after_scale_in**: after scale-in
fail_action	Yes	String	Whether to continue to execute subsequent scripts and create a cluster after the custom automation script fails to be executed. ● **continue**: Continue to execute subsequent scripts. ● **errorout**: Stop the action. **NOTE** ● You are advised to set this parameter to **continue** in the commissioning phase so that the cluster can continue to be installed and started no matter whether the custom automation script is executed successfully. ● The scale-in operation cannot be undone. Therefore, **fail_action** must be set to **continue** for the scripts that are executed after scale-in.

Table 6-16 Auto scaling metrics

Cluster Type	Metric Name	Value Type	Description
Streaming cluster	StormSlotAvailable	Integer	Number of available Storm slots. Value range: 0 to 2147483646

Cluster Type	Metric Name	Value Type	Description
	StormSlotAvailablePercentage	Percentage	Percentage of available Storm slots, that is, the proportion of the available slots to total slots. Value range: 0 to 100
	StormSlotUsed	Integer	Number of the used Storm slots. Value range: 0 to 2147483646
	StormSlotUsedPercentage	Percentage	Percentage of the used Storm slots, that is, the proportion of the used slots to total slots. Value range: 0 to 100
	StormSupervisorMemAverageUsage	Integer	Average memory usage of the Supervisor process of Storm. Value range: 0 to 2147483646
	StormSupervisorMemAverageUsagePercentage	Percentage	Average percentage of the used memory of the Supervisor process of Storm to the total memory of the system. Value range: 0 to 100
	StormSupervisorCPUAverageUsagePercentage	Percentage	Average percentage of the used CPUs of the Supervisor process of Storm to the total CPUs. Value range: 0 to 6000
Analysis cluster	YARNAppPending	Integer	Number of pending tasks on Yarn. Value range: 0 to 2147483646
	YARNAppPendingRatio	Ratio	Ratio of pending tasks on Yarn, that is, the ratio of pending tasks to running tasks on Yarn. Value range: 0 to 2147483646
	YARNAppRunning	Integer	Number of running tasks on Yarn. Value range: 0 to 2147483646
	YARNContainerAllocated	Integer	Number of containers allocated to Yarn. Value range: 0 to 2147483646
	YARNContainerPending	Integer	Number of pending containers on Yarn. Value range: 0 to 2147483646

Cluster Type	Metric Name	Value Type	Description
	YARNContainerPendingRatio	Ratio	Ratio of pending containers on Yarn, that is, the ratio of pending containers to running containers on Yarn. Value range: 0 to 2147483646
	YARNCPUAllocated	Integer	Number of virtual CPUs (vCPUs) allocated to Yarn Value range: 0 to 2147483646
	YARNCPUAvailable	Integer	Number of available vCPUs on Yarn. Value range: 0 to 2147483646
	YARNCPUAvailablePercentage	Percentage	Percentage of available vCPUs on Yarn, that is, the proportion of available vCPUs to total vCPUs. Value range: 0 to 100
	YARNCPUPending	Integer	Number of pending vCPUs on Yarn. Value range: 0 to 2147483646
	YARNMemoryAllocated	Integer	Memory allocated to Yarn. The unit is MB. Value range: 0 to 2147483646
	YARNMemoryAvailable	Integer	Available memory on Yarn. The unit is MB. Value range: 0 to 2147483646
	YARNMemoryAvailablePercentage	Percentage	Percentage of available memory on Yarn, that is, the proportion of available memory to total memory on Yarn. Value range: 0 to 100
	YARNMemoryPending	Integer	Pending memory on Yarn. Value range: 0 to 2147483646

NOTE

When the value type is percentage or ratio in **Table 6-16**, the valid value can be accurate to percentile. The percentage metric value is a decimal value with a percent sign (%) removed. For example, 16.80 represents 16.80%.

Response Parameters

Status code: 200

Table 6-17 Response body parameters

Parameter	Type	Description
cluster_id	String	Cluster ID, which is returned by the system after the cluster is created.
result	Boolean	Operation result. ● **true**: The operation is successful. ● **false**: The operation failed.
msg	String	System message, which can be empty.

Example Request

- Creating a cluster with **Cluster HA** enabled (using the **node_groups** parameter group)

```
POST https://{endpoint}/v1.1/{project_id}/run-job-flow

{
   "billing_type": 12,
   "data_center": ,
   "available_zone_id": ,
   "cluster_name": "mrs_HEbK",
   "cluster_version": "MRS 3.1.0",
   "safe_mode": 0,
   "cluster_type": 0,
"component_list": [
      {
         "component_name": "Hadoop"
      },
      {
         "component_name": "Spark2x"
      },
      {
         "component_name": "HBase"
      },
      {
         "component_name": "Hive"
      },
      {
         "component_name": "Tez"
      },
      {
          "component_name": "Hue"
      },
      {
         "component_name": "Flink"
       }
],
   "vpc": "vpc-4b1c",
   "vpc_id": "4a365717-67be-4f33-80c5-98e98a813af8",
   "subnet_id": "67984709-e15e-4e86-9886-d76712d4e00a",
   "subnet_name": "subnet-4b44",
   "security_groups_id": "4820eace-66ad-4f2c-8d46-cf340e3029dd",
   "tags": [{
      "key": "key1",
      "value": "value1"
```

```
}, {
    "key": "key2",
    "value": "value2"
}],
"node_groups": [{
        "group_name": "master_node_default_group",
        "node_num": 2,
        "node_size": "",
        "root_volume_size": 480,
        "root_volume_type": "SATA",
        "data_volume_type": "SATA",
        "data_volume_count": 1,
        "data_volume_size": 600
    }, {
        "group_name": "core_node_analysis_group",
        "node_num": 3,
        "node_size": "",
        "root_volume_size": 480,
        "root_volume_type": "SATA",
        "data_volume_type": "SATA",
        "data_volume_count": 1,
        "data_volume_size": 600
    }, {
        "group_name": "task_node_analysis_group",
        "node_num": 2,
        "node_size": "",
        "root_volume_size": 480,
        "root_volume_type": "SATA",
        "data_volume_type": "SATA",
        "data_volume_count": 0,
        "data_volume_size": 600,
        "auto_scaling_policy": {
            "auto_scaling_enable": true,
            "min_capacity": 1,
            "max_capacity": "3",
            "resources_plans": [{
                "period_type": "daily",
                "start_time": "9:50",
                "end_time": "10:20",
                "min_capacity": 2,
                "max_capacity": 3
            }, {
                "period_type ": "daily",
                "start_time ": "10:20",
                "end_time ": "12:30",
                "min_capacity ": 0,
                "max_capacity ": 2
            }],
            "exec_scripts": [{
                "name": "before_scale_out",
                "uri": "s3a://XXX/zeppelin_install.sh ",
                "parameters": "${mrs_scale_node_num} ${mrs_scale_type} xxx",
                "nodes": ["master", "core", "task"],
                "active_master": "true",
                "action_stage": "before_scale_out",
                "fail_sction": "continue"
            }, {
                "name": "after_scale_out",
                "uri": "s3a://XXX/storm_rebalance.sh",
                "parameters": "${mrs_scale_node_hostnames} ${mrs_scale_node_ips}",
                "nodes": ["master", "core", "task"],
                "active_master": "true",
                "action_stage": "after_scale_out",
                "fail_action": "continue"
            }],
            "rules": [{
                "name": "default-expand-1",
                "adjustment_type": "scale_out",
                "cool_down_minutes": 5,
```

```
                "scaling_adjustment": 1,
                "trigger": {
                    "metric_name": "YARNMemoryAvailablePercentage",
                    "metric_value": "25",
                    "comparison_operator": "LT",
                    "evaluation_periods": 10
                }
            }, {
                "name": "default-shrink-1",
                "adjustment_type": "scale_in",
                "cool_down_minutes": 5,
                "scaling_adjustment": 1,
                "trigger": {
                    "metric_name": "YARNMemoryAvailablePercentage",
                    "metric_value": "70",
                    "comparison_operator": "GT",
                    "evaluation_periods": 10
                }
            }]
        }
    }
  ],
  "login_mode": 1,
  "cluster_master_secret": "",
  "cluster_admin_secret": "",
  "log_collection": 1,
  "add_jobs": [{
    "job_type": 1,
    "job_name": "tenji111",
    "jar_path": "s3a://bigdata/program/hadoop-mapreduce-examples-xxx.jar",
    "arguments": "wordcount",
    "input": "s3a://bigdata/input/wd_1k/",
    "output": "s3a://bigdata/output/",
    "job_log": "s3a://bigdata/log/",
    "shutdown_cluster": true,
    "file_action": "",
    "submit_job_once_cluster_run": true,
    "hql": "",
    "hive_script_path": ""
  }],
  "bootstrap_scripts": [{
    "name": "Modify os config",
    "uri": "s3a://XXX/modify_os_config.sh",
    "parameters": "param1 param2",
    "nodes": ["master", "core", "task"],
    "active_master": "false",
    "before_component_start": "true",
          "start_time" : "1667892101",
          "state" : "IN_PROGRESS",
    "fail_action": "continue"
     "action_stages" : [ "BEFORE_COMPONENT_FIRST_START", "BEFORE_SCALE_IN" ]
  }, {
    "name": "Install zepplin",
    "uri": "s3a://XXX/zeppelin_install.sh",
    "parameters": "",
    "nodes": ["master"],
    "active_master": "true",
    "before_component_start": "false",
          "start_time" : "1667892101",
          "state" : "IN_PROGRESS",
    "fail_action": "continue"
        "action_stages" : [ "AFTER_SCALE_IN", "AFTER_SCALE_OUT" ]
  }]
}
```

- Creating a cluster with **Cluster HA** enabled (without using the **node_groups** parameter group)

```
POST https://{endpoint}/v1.1/{project_id}/run-job-flow

{
```

```
"billing_type": 12,
"data_center": ,
"master_node_num": 2,
"master_node_size": "",
"core_node_num": 3,
"core_node_size": "",
"available_zone_id": ,
"cluster_name": "newcluster",
"vpc": "vpc1",
"vpc_id": "5b7db34d-3534-4a6e-ac94-023cd36aaf74",
"subnet_id": "815bece0-fd22-4b65-8a6e-15788c99ee43",
"subnet_name": "subnet",
"security_groups_id": "845bece1-fd22-4b45-7a6e-14338c99ee43",
"tags": [
  {
    "key": "key1",
    "value": "value1"
  },
  {
    "key": "key2",
    "value": "value2"
  }
],
"cluster_version": "MRS 3.1.0",
"cluster_type": 0,
"master_data_volume_type": "SATA",
"master_data_volume_size": 600,
"master_data_volume_count": 1,
"core_data_volume_type": "SATA",
"core_data_volume_size": 600,
"core_data_volume_count": 2,
"node_public_cert_name": "SSHkey-bba1",
"safe_mode": 0,
"log_collection": 1,
"task_node_groups": [
  {

    "node_num": 2,
    "node_size": "",
    "data_volume_type": "SATA",
    "data_volume_count": 1,
    "data_volume_size": 600,
    "auto_scaling_policy":
     {
       "auto_scaling_enable": true,
       "min_capacity": "1",
       "max_capacity": "3",
       "resources_plans": [{
         "period_type": "daily",
         "start_time": "9:50",
         "end_time": "10:20",
         "min_capacity": "2",
         "max_capacity": "3"
       },{
         "period_type": "daily",
         "start_time": "10:20",
         "end_time": "12:30",
         "min_capacity": "0",
         "max_capacity": "2"
       }],
       "exec_scripts": [{
         "name": "before_scale_out",
         "uri": "s3a://XXX/zeppelin_install.sh",
         "parameters": "${mrs_scale_node_num}${mrs_scale_type}xxx",
         "nodes": [
           "master",
           "core",
           "task"
         ],
```

```
            "active_master": "true",
            "action_stage": "before_scale_out",
            "fail_action": "continue"
          },{
            "name": "after_scale_out",
            "uri": "s3a://XXX/storm_rebalance.sh",
            "parameters": "${mrs_scale_node_hostnames}${mrs_scale_node_ips}",
            "nodes": [
              "master",
              "core",
              "task"
            ],
            "active_master": "true",
            "action_stage": "after_scale_out",
            "fail_action": "continue"
          }],
          "rules": [
           {
            "name": "default-expand-1",
            "adjustment_type": "scale_out",
            "cool_down_minutes": 5,
            "scaling_adjustment": 1,
            "trigger": {
              "metric_name": "YARNMemoryAvailablePercentage",
              "metric_value": "25",
              "comparison_operator": "LT",
              "evaluation_periods": 10
             }
            },
          {
            "name": "default-shrink-1",
            "adjustment_type": "scale_in",
            "cool_down_minutes": 5,
            "scaling_adjustment": 1,
            "trigger": {
              "metric_name": "YARNMemoryAvailablePercentage",
              "metric_value": "70",
              "comparison_operator": "GT",
              "evaluation_periods": 10
            }
          }
        ]
      }
    }
  ],
"component_list": [
  {
      "component_name": "Hadoop"
  },
  {
      "component_name": "Spark2x"
  },
  {
      "component_name": "HBase"
  },
  {
      "component_name": "Hive"
  },
  {
      "component_name": "Presto"
  },
  {
      "component_name": "Tez"
  },
  {
       "component_name": "Hue"
  },
  {
      "component_name": "Flink"
```

```
        }
    ],
    "add_jobs": [
        {
            "job_type": 1,
            "job_name": "tenji111",
            "jar_path": "s3a://bigdata/program/hadoop-mapreduce-examples-XXX.jar",
            "arguments": "wordcount",
            "input": "s3a://bigdata/input/wd_1k/",
            "output": "s3a://bigdata/output/",
            "job_log": "s3a://bigdata/log/",
            "shutdown_cluster": false,
            "file_action": "",
            "submit_job_once_cluster_run": true,
            "hql": "",
            "hive_script_path": ""
        }
    ],
"bootstrap_scripts": [
        {
            "name":"Modifyosconfig",
            "uri":"s3a://XXX/modify_os_config.sh",
            "parameters":"param1 param2",
            "nodes":[
                "master",
                "core",
                "task"
            ],
            "active_master":"false",
            "before_component_start":"true",
            "fail_action":"continue"
        },
        {
            "name":"Install zepplin",
            "uri":"s3a://XXX/zeppelin_install.sh",
            "parameters":"",
            "nodes":[
                "master"
            ],
            "active_master":"true",
            "before_component_start":"false",
            "fail_action":"continue"
        }
    ]
}
```

- Disabling the **Cluster HA** function and creating a cluster with the minimum specifications (using the **node_groups** parameter group)

```
POST https://{endpoint}/v1.1/{project_id}/run-job-flow

{
    "billing_type": 12,
    "data_center": ,
    "available_zone_id": ,
    "cluster_name": "mrs_HEbK",
    "cluster_version": "MRS 3.1.0",
    "safe_mode": 0,
    "cluster_type": 0,
"component_list": [
        {
            "component_name": "Hadoop"
        },
        {
            "component_name": "Spark2x"
        },
        {
            "component_name": "HBase"
        },
        {
            "component_name": "Hive"
```

```
        },
        {
            "component_name": "Tez"
        },
        {
             "component_name": "Hue"
        },
        {
            "component_name": "Flink"
         }
],
    "vpc": "vpc-4b1c",
    "vpc_id": "4a365717-67be-4f33-80c5-98e98a813af8",
    "subnet_id": "67984709-e15e-4e86-9886-d76712d4e00a",
    "subnet_name": "subnet-4b44",
    "security_groups_id": "4820eace-66ad-4f2c-8d46-cf340e3029dd",
    "tags": [{
        "key": "key1",
        "value": "value1"
    }, {
        "key": "key2",
        "value": "value2"
    }],
    "node_groups": [{
            "group_name": "master_node_default_group",
            "node_num": 1,
            "node_size": "",
            "root_volume_size": 480,
            "root_volume_type": "SATA",
            "data_volume_type": "SATA",
            "data_volume_count": 1,
            "data_volume_size": 600
        }, {
            "group_name": "core_node_analysis_group",
            "node_num": 1,
            "node_size": "",
            "root_volume_size": 480,
            "root_volume_type": "SATA",
            "data_volume_type": "SATA",
            "data_volume_count": 1,
            "data_volume_size": 600
        }
    ],
    "login_mode": 1,
    "cluster_master_secret": "",
    "cluster_admin_secret": "",
    "log_collection": 1,
    "add_jobs": [{
        "job_type": 1,
        "job_name": "tenji111",
        "jar_path": "s3a://bigdata/program/hadoop-mapreduce-examples-xxx.jar",
        "arguments": "wordcount",
        "input": "s3a://bigdata/input/wd_1k/",
        "output": "s3a://bigdata/output/",
        "job_log": "s3a://bigdata/log/",
        "shutdown_cluster": true,
        "file_action": "",
        "submit_job_once_cluster_run": true,
        "hql": "",
        "hive_script_path": ""
    }],
    "bootstrap_scripts": [{
        "name": "Modify os config",
        "uri": "s3a://XXX/modify_os_config.sh",
        "parameters": "param1 param2",
        "nodes": ["master", "core", "task"],
        "active_master": "false",
        "before_component_start": "true",
        "fail_action": "continue"
```

```
  }, {
      "name": "Install zepplin",
      "uri": "s3a://XXX/zeppelin_install.sh",
      "parameters": "",
      "nodes": ["master"],
      "active_master": "true",
      "before_component_start": "false",
      "fail_action": "continue"
  }]
}
```

- Disabling the **Cluster HA** function and creating a cluster with the minimum specifications (without using the **node_groups** parameter group)

```
POST https://{endpoint}/v1.1/{project_id}/run-job-flow

{
   "billing_type": 12,
   "data_center": ,
   "master_node_num": 1
   "master_node_size": "",
   "core_node_num": 1,
   "core_node_size": "",
   "available_zone_id": ,
   "cluster_name": "newcluster",
   "vpc": "vpc1",
   "vpc_id": "5b7db34d-3534-4a6e-ac94-023cd36aaf74",
   "subnet_id": "815bece0-fd22-4b65-8a6e-15788c99ee43",
   "subnet_name": "subnet",
   "security_groups_id": "",
   "tags": [
     {
        "key": "key1",
        "value":"value1"
     },
     {
        "key": "key2",
        "value": "value2"
     }
   ],
   "cluster_version": "MRS 3.1.0",
   "cluster_type": 0,
   "master_data_volume_type": "SATA",
   "master_data_volume_size": 600,
   "master_data_volume_count": 1,
   "core_data_volume_type": "SATA",
   "core_data_volume_size": 600,
   "core_data_volume_count": 1,
   "login_mode": 1,
   "node_public_cert_name": "SSHkey-bba1",
   "safe_mode": 0,
   "cluster_admin_secret":"******",
   "log_collection": 1,
   "component_list": [
      {
        "component_name": "Hadoop"
      },
      {
        "component_name": "Spark2x"
      },
      {
        "component_name": "HBase"
      },
      {
         "component_name": "Hive"
      },
      {
         "component_name": "Tez"
      },
      {
         "component_name": "Hue"
```

```
        },
        {
            "component_name": "Flink"
        }
                    ],
    "add_jobs": [
        {
            "job_type": 1,
            "job_name": "tenji111",
            "jar_path": "s3a://bigdata/program/hadoop-mapreduce-examples-XXX.jar",
            "arguments": "wordcount",
            "input": "s3a://bigdata/input/wd_1k/",
            "output": "s3a://bigdata/output/",
            "job_log": "s3a://bigdata/log/",
            "shutdown_cluster": false,
            "file_action": "",
            "submit_job_once_cluster_run": true,
            "hql": "",
            "hive_script_path": ""
        }
    ],
"bootstrap_scripts": [
        {
            "name":"Install zepplin",
            "uri":"s3a://XXX/zeppelin_install.sh",
            "parameters":"",
            "nodes":[
            "master"
            ],
            "active_master":"false",
            "before_component_start":"false",
            "fail_action":"continue"
        }
    ]
}
```

Example Response

Status code: 200

The cluster is created.

```
{
  "cluster_id" : "da1592c2-bb7e-468d-9ac9-83246e95447a",
  "result" : true,
  "msg" : ""
}
```

Status Codes

Table 6-18 describes the status code.

Table 6-18 Status code

Status Code	Description
200	The cluster has been created.

See **Status Codes**.

Error Codes

See **Error Codes**.

6.1.2 Resizing a Cluster

Function

This API is used to manually scale out or scale in Core or Task nodes in a cluster that has been created. After an MRS cluster is created, the number of Master nodes cannot be adjusted. That is, Master nodes cannot be scaled in or out. This API is incompatible with Sahara.

Only clusters in the **Running** state can be scaled out or in.

The APIs described in this section support only streaming, analysis, and hybrid clusters.

URI

- Format

  PUT /v1.1/{project_id}/cluster_infos/{cluster_id}

- Parameter description

Table 6-19 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.
cluster_id	Yes	String	Cluster ID For details about how to obtain the cluster ID, see **Obtaining the MRS Cluster Information**.

Request Parameters

Table 6-20 Request parameters

Parameter	Mandatory	Type	Description
service_id	No	String	Service ID. This parameter is reserved for extension. You do not need to set this parameter.
plan_id	No	String	Plan ID. This parameter is reserved for extension. You do not need to set this parameter.

Parameter	Mandatory	Type	Description
parameters	Yes	Object	Core parameters. For details, see **Table 6-21**.
previous_values	No	Map<String,String>	This parameter is an extended API and needs to be reserved. You do not need to set this parameter.

Table 6-21 parameters description

Parameter	Mandatory	Type	Description
order_id	No	String	Order ID obtained by the system during scale-out or scale-in. You do not need to set the parameter.
scale_type	Yes	String	• **scale_in**: cluster scale-in • **scale_out**: cluster scale-out
node_id	Yes	String	ID of the newly added or removed node. The parameter value is fixed to **node_orderadd**.
node_group	No	String	Node group to be scaled out or in • If the value of **node_group** is **core_node_default_group**, the node group is a Core node group. • If the value of **node_group** is **task_node_default_group**, the node group is a Task node group. If it is left blank, the default value **core_node_default_group** is used.
task_node_info	No	Object	Task node specifications. For more parameter description, see **Table 6-22**. • When the number of Task nodes is **0**, this parameter is used to specify Task node specifications. • When the number of Task nodes is greater than **0**, this parameter is unavailable.

Parameter	Mandatory	Type	Description
instances	Yes	Integer	Number of nodes to be added or removed ● The maximum number of nodes to be added is 500 minus the number of Core and Task nodes. For example, the current number of Core nodes is 3, the number of nodes to be added must be less than or equal to 497. A maximum of 500 Core and Task nodes are supported by default. If more than 500 Core and Task nodes are required, contact technical support engineers or call a background API to modify the database. ● Nodes can be deleted for cluster scale-out when the number of Core nodes is greater than 3 or the number of Task nodes is greater than 0. For example, if there are 5 Core nodes and 5 Task nodes in a cluster, only 2 (5 minus 3) Core nodes are available for deletion and 5 or fewer than 5 Task nodes can be deleted.
skip_bootstrap_scripts	No	String	This parameter is valid only when a bootstrap action is configured during cluster creation and takes effect during scale-out. It indicates whether the bootstrap action specified during cluster creation is performed on nodes added during scale-out. The default value is **false**, indicating that the bootstrap action is performed.
scale_without_start	No	Boolean	Whether to start components on the added nodes after cluster scale-out ● **true**: Do not start components after scale-out. ● **false**: Start components after scale-out.

Parameter	Mandatory	Type	Description
server_ids	No	Array of strings	ID list of Task nodes to be deleted during task node scale-in. • This parameter does not take effect when **scale_type** is set to **scale-out**. • If **scale_type** is set to **scale-in** and cannot be left blank, the system deletes the specified Task nodes. • When **scale_type** is set to **scale-in** and **server_ids** is left blank, the system automatically deletes the Task nodes based on the system rules.

Table 6-22 task_node_info parameters

Parameter	Mandatory	Type	Description
node_size	Yes	String	Instance specifications of a Task node, Example:
data_volume_type	Yes	String	Data disk storage type of the Task node, supporting SATA, SAS, and SSD currently. • **SATA**: common I/O • **SAS**: high I/O • **SSD**: ultra-high I/O • **GPSSD**: general-purpose SSD
data_volume_count	Yes	Integer	Number of data disks of a Task node Value range: 0 to 10
data_volume_size	Yes	Integer	Data disk storage space of a Task node Value range: 100 GB to 32,000 GB. When assigning a value to this parameter, you only need to pass a number, without the unit.

Response Parameters

Response parameters

Table 6-23 describes the response parameter.

Table 6-23 Response parameter

Parameter	Type	Description
result	String	Operation result • **succeeded**: The operation is successful. • Error Codes describes the error codes returned upon operation failures.

Example

- Example request

  Example of adding core nodes

```
PUT /v1.1/{project_id}/cluster_infos/{cluster_id}

{
   "service_id": "",
   "plan_id": "",
   "parameters": {
      "order_id": "",
      "scale_type": "scale_out",
      "node_id": "node_orderadd",
      "node_group": "core_node_default_group",
      "instances": "1",
     "skip_bootstrap_scripts":false,
     "scale_without_start":false
   },
   "previous_values": { }
}
```

  Example of adding task nodes when the number of existing task nodes is greater than zero

```
PUT /v1.1/{project_id}/cluster_infos/{cluster_id}

{
   "service_id": "",
   "plan_id": "",
   "parameters": {
      "order_id": "",
      "scale_type": "scale_out",
      "node_id": "node_orderadd",
      "node_group": "task_node_default_group",
      "instances": "1",
      "skip_bootstrap_scripts":false,
      "scale_without_start":false
   },
   "previous_values": { }
}
```

  Example of adding task nodes when the number of existing task nodes is zero

```
PUT /v1.1/{project_id}/cluster_infos/{cluster_id}

{
   "service_id": "",
   "plan_id": "",
   "parameters": {
      "order_id": "",
```

```
        "scale_type": "scale_out",
        "node_id": "node_orderadd",
        "node_group": "task_node_default_group",
        "task_node_info": {
                "node_size": "",
                "data_volume_type":"SATA",
                "data_volume_count":2,
                "data_volume_size":600
                },
        "instances": "1",
        "scale_without_start":false


    },
    "previous_values": { }
}
```

Example of adding core nodes

```
PUT /v1.1/{project_id}/cluster_infos/{cluster_id}

{
    "service_id": "",
    "plan_id": "",
    "parameters": {
        "order_id": "",
        "scale_type": "scale_in",
        "node_id": "node_orderadd",
        "node_group": "core_node_default_group",
        "instances": "1"


    },
    "previous_values": { }
}
```

Example of removing task nodes

```
PUT /v1.1/{project_id}/cluster_infos/{cluster_id}

{
    "service_id": "",
    "plan_id": "",
    "parameters": {
        "order_id": "",
        "scale_type": "scale_in",
        "node_id": "node_orderadd",
        "node_group": "task_node_default_group",
        "instances": "1"

    },
    "previous_values": { }
}
```

Example of removing specified task nodes

```
PUT /v1.1/{project_id}/cluster_infos/{cluster_id}

{
    "service_id": "",
    "plan_id": "",
    "parameters": {
        "order_id": "",
        "scale_type": "scale_in",
        "node_id": "node_orderadd",
        "node_group": "task_node_default_group",
        "instances": "2",
        "server_ids": ["c9573435-7814-4b2c-9131-ad78b814414c",
"a4951009-6a0f-4e7b-9c81-9d4bd1f8c537"]
    },
    "previous_values": { }
}
```

- Example response

```
{
    "result": "succeeded"
}
```

Status Codes

- **Table 6-24** describes the status code.

  Table 6-24 Status code

Status Code	Description
200	The core or task nodes have been added or removed.

- See **Status Codes**.

Error Codes

See **Error Codes**.

6.1.3 Querying a Cluster List

Function

This API is used to query a list of clusters created by a user. This API is incompatible with Sahara.

URI

- Format

  GET /v1.1/{project_id}/cluster_infos

- Parameter description

  Table 6-25 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	The project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.

Table 6-26 Query parameters

Parameter	Mandatory	Type	Description
pageSize	No	String	Maximum number of clusters displayed on a page Value range: 1 to 2147483646
currentPage	No	String	Current page number
clusterName	No	String	The cluster name.
clusterState	No	String	You can query a cluster list by cluster status. • **existing**: Query a list of existing clusters except the terminated clusters. • **history**: Query a list of history clusters, including the terminated clusters, clusters that fail to be terminated, clusters that fail to delete VMs, and clusters that fail to terminate a database update. • **starting**: Query a list of clusters that are being started. • **running**: Query a list of running clusters. • **terminated**: Query a list of terminated clusters. • **failed**: Query a list of failed clusters. • **abnormal**: Query a list of abnormal clusters. • **terminating**: Query a list of clusters that are being terminated. • **frozen**: Query a list of frozen clusters. • **scaling-out**: Query a list of clusters that are being scaled out. • **scaling-in**: Query a list of clusters that are being scaled in.

Parameter	Mandatory	Type	Description
tags	No	String	You can search for a cluster by its tag. If you specify multiple tags, the relationship between them is AND. • The format of the **tags** parameter is **tags=k1*v1,k2*v2,k3*v3**. • When the values of some tags are null, the format is **tags=k1,k2,k3*v3**.
enterpriseProjectId	No	String	The enterprise project ID used to query clusters in a specified enterprise project. The default value is **0**, indicating the **default** enterprise project.

Request Parameters

None

Response Parameters

Table 6-27 Response parameters

Parameter	Type	Description
clusterTotal	String	Total number of clusters in a list
clusters	Array of Cluster objects	Cluster parameters. For details, see **Table 6-28**.

Table 6-28 cluster parameters

Parameter	Type	Description
clusterId	String	Cluster ID.
clusterName	String	Cluster name.
masterNodeNum	String	Number of Master nodes deployed in a cluster.
coreNodeNum	String	Number of Core nodes deployed in a cluster.
totalNodeNum	String	Total number of nodes deployed in a cluster.

Parameter	Type	Description
clusterState	String	Cluster status. Valid values include: • **starting**: The cluster is being started. • **running**: The cluster is running. • **terminated**: The cluster has been terminated. • **failed**: The cluster fails. • **abnormal**: The cluster is abnormal. • **terminating**: The cluster is being terminated. • **frozen**: The cluster has been frozen. • **scaling-out**: The cluster is being scaled out. • **scaling-in**: The cluster is being scaled in.
createAt	String	Cluster creation time, which is a 10-bit timestamp.
updateAt	String	Cluster update time, which is a 10-bit timestamp.
dataCenter	String	Cluster work region.
vpc	String	VPC name.
vpcId	String	VPC ID.
hadoopVersion	String	Hadoop version.
masterNodeSize	String	Instance specifications of a Master node.
coreNodeSize	String	Instance specifications of a Core node.
componentList	Array	Component list. For details, see **Table 6-29**.
externalIp	String	External IP address.
externalAlterna-teIp	String	Backup external IP address.
internalIp	String	Internal IP address.
deploymentId	String	Cluster deployment ID.
remark	String	Cluster remarks.
orderId	String	Cluster creation order ID.
azId	String	AZ ID.

Parameter	Type	Description
masterNodeProductId	String	Product ID of a Master node.
masterNodeSpecId	String	Specification ID of a Master node.
coreNodeProductId	String	Product ID of a Core node.
coreNodeSpecId	String	Specification ID of a Core node.
azName	String	AZ name.
azCode	String	AZ name (en).
availabilityZoneId	String	The AZ.
instanceId	String	Instance ID.
vnc	String	URI for remotely logging in to an ECS.
tenantId	String	Project ID.
volumeSize	Integer	Disk storage space.
volumeType	String	Disk type.
subnetId	String	Subnet ID.
clusterType	String	Cluster type.
subnetName	String	Subnet name.
securityGroupsId	String	Security group ID.
slaveSecurityGroupsId	String	Security group ID of a non-Master node. Currently, one MRS cluster uses only one security group. Therefore, this field has been discarded.
bootstrapScripts	Array of Table BootstrapScript objects	The bootstrap action script information.

Parameter	Type	Description
stageDesc	String	Cluster operation progress description. The cluster installation progress includes: • Verifying cluster parameters: Cluster parameters are being verified. • Applying for cluster resources: Cluster resources are being applied for. • Creating VMs: The VMs are being created. • Initializing VMs: The VMs are being initialized. • Installing MRS Manager: MRS Manager is being installed. • Deploying the cluster: The cluster is being deployed. • Cluster installation failed: Failed to install the cluster. The cluster scale-out progress includes: • Preparing for scale-out: Cluster scale-out is being prepared. • Creating VMs: The VMs are being created. • Initializing VMs: The VMs are being initialized. • Adding nodes to the cluster: The nodes are being added to the cluster. • Scale-out failed: Failed to scale out the cluster. The cluster scale-in progress includes: • Preparing for scale-in: Cluster scale-in is being prepared. • Decommissioning instance: The instance is being decommissioned. • Deleting VMs: The VMs are being deleted. • Deleting nodes from the cluster: The nodes are being deleted from the cluster. • Scale-in failed: Failed to scale in the cluster.

Parameter	Type	Description
		If the cluster installation, scale-out, or scale-in fails, **stageDesc** will display the failure cause.
ismrsManagerFinish	Boolean	Whether MRS Manager installation is finished during cluster creation. • **true**: MRS Manager installation is finished. • **false**: MRS Manager installation is not finished.
safeMode	Integer	Running mode of an MRS cluster. • **0**: Normal cluster • **1**: Security cluster
clusterVersion	String	Cluster version.
nodePublicCertName	String	Name of the key file.
masterNodeIp	String	IP address of a Master node.
privateIpFirst	String	Preferred private IP address.
errorInfo	String	Error message.
tags	String	The tag information.
logCollection	Integer	Whether to collect logs when cluster installation fails. • **0**: Do not collect. • **1**: Collect.
taskNodeGroups	List<NodeGroup>	List of Task nodes. For more parameter description, see **Table 6-31**.
nodeGroups	List<NodeGroup>	List of Master, Core and Task nodes. For more parameter description, see **Table 6-31**.
masterDataVolumeType	String	Data disk storage type of the Master node. Currently, SATA, SAS, and SSD are supported.
masterDataVolumeSize	Integer	Data disk storage space of the Master node To increase data storage capacity, you can add disks at the same time when creating a cluster. Value range: 100 GB to 32,000 GB

Parameter	Type	Description
masterDataVolumeCount	Integer	Number of data disks of the Master node The value can be set to **1** only.
coreDataVolumeType	String	Data disk storage type of the Core node. Currently, SATA, SAS, and SSD are supported.
coreDataVolumeSize	Integer	Data disk storage space of the Core node. To increase data storage capacity, you can add disks at the same time when creating a cluster. Value range: 100 GB to 32,000 GB
coreDataVolumeCount	Integer	Number of data disks of the Core node. Value range: 1 to 10
scale	String	The node change status. If this parameter is left blank, the cluster nodes are not changed. Possible values: • **Scaling-out**: The cluster is being scaled out. • **Scaling-in**: The cluster is being scaled in. • **scaling-error**: The cluster is in the running state and fails to be scaled in or out or the specifications fail to be scaled up for the last time. • **scaling-up**: The master node specifications are being scaled up. • **scaling_up_first**: The standby master node specifications are being scaled up. • **scaled_up_first**: The standby master node specifications have been scaled up. • **scaled-up-success**: The master node specifications have been scaled up.

Table 6-29 componentList parameters

Parameter	Type	Description
componentId	String	Component ID For example, the **component_id** of Hadoop is MRS 3.1.0_001.
componentName	String	Component name
componentVersion	String	Component version
componentDesc	String	Component description

Table 6-30 BootstrapScript parameters

Parameter	Type	Description
name	String	The name of a bootstrap action script, which must be unique in a cluster. The value can contain only numbers, letters, spaces, hyphens (-), and underscores (_) and cannot start with a space. The value can contain 1 to 64 characters.
uri	String	The path of a bootstrap action script. Set this parameter to an OBS bucket path or a local VM path. • OBS bucket path: Enter a script path. For example, enter the path of the public sample script provided by MRS. Example: **s3a://bootstrap/presto/presto-install.sh**. If **dualroles** is installed, the parameter of the **presto-install.sh** script is **dualroles**. If **worker** is installed, the parameter of the **presto-install.sh** script is **worker**. Based on the Presto usage habit, you are advised to install **dualroles** on the active master nodes and **worker** on the core nodes. • Local VM path: Enter a script path. The script path must start with a slash (/) and end with **.sh**.
parameters	String	The bootstrap action script parameters.

Parameter	Type	Description
nodes	Array of strings	Type of a node where the Bootstrap action script is executed. The value can be **master**, **core**, or **task**. The node type must be represented in lowercase letters.
active_master	Boolean	Whether the bootstrap action script runs only on active master nodes. The default value is **false**, indicating that the Bootstrap action script can run on all master nodes.
fail_action	String	Whether to continue executing subsequent scripts and creating a cluster after the bootstrap action script fails to execute. The default value is **errorout**, indicating that the action is stopped. Note: You are advised to set this parameter to **continue** in the commissioning phase so that the cluster can continue to be installed and started no matter whether the bootstrap action is successful. Possible values: • **continue**: Continue to execute subsequent scripts. • **errorout**: Stop the action.
before_component_start	Boolean	Time when the bootstrap action script is executed. Currently, the following two options are available: **Before component start** and **After component start**. The default value is **false**, indicating that the bootstrap action script is executed after the component is started.

Parameter	Type	Description
action_stages	Array of strings	Select the time when the bootstrap action script is executed. • **BEFORE_COMPONENT_FIRST_START**: before initial component starts • **AFTER_COMPONENT_FIRST_START**: after initial component starts • **BEFORE_SCALE_IN**: before scale-in • **AFTER_SCALE_IN**: after scale-in • **BEFORE_SCALE_OUT**: before scale-out • **AFTER_SCALE_OUT**: after scale-out

Table 6-31 NodeGroup parameters

Parameter	Type	Description
GroupName	String	Node group name.
NodeNum	Integer	Number of nodes. The value ranges from 0 to 500. The minimum number of Master and Core nodes is 1 and the total number of Core and Task nodes cannot exceed 500.
NodeSize	String	Instance specifications of a node.
NodeSpecId	String	Instance specification ID of a node
NodeProductId	String	Instance product ID of a node
VmProductId	String	VM product ID of a node
VmSpecCode	String	VM specifications of a node
RootVolumeSize	Integer	System disk size of a node. This parameter is not configurable and its default value is **40 GB**.
RootVolumeProductId	String	System disk product ID of a node
RootVolumeType	String	System disk type of a node
RootVolumeResourceSpecCode	String	System disk product specifications of a node
RootVolumeResourceType	String	System disk product type of a node

Parameter	Type	Description
DataVolumeType	String	Data disk storage type of a node. Currently, SATA, SAS, and SSD are supported. • SATA: Common I/O • SAS: High I/O • SSD: Ultra-high I/O
DataVolumeCount	Integer	Number of data disks of a node.
DataVolumeSize	Integer	Data disk storage space of a node.
DataVolumeProductId	String	Data disk product ID of a node
DataVolumeResourceSpecCode	String	Data disk product specifications of a node
DataVolumeResourceType	String	Data disk product type of a node

Example

- Example request
  ```
  GET /v1.1/{project_id}/cluster_infos?
  pageSize={page_size}&currentPage={current_page}&clusterState={cluster_state}&tags={tags}
  ```
- Example response
  ```
  {
     "clusterTotal": 1,
     "clusters": [
        {
           "clusterId": "bc134369-294c-42b7-a707-b2036ba38524",
           "clusterName": "mrs_D0zW",
           "masterNodeNum": "2",
           "coreNodeNum": "3",
           "clusterState": "terminated",
           "createAt": "1498272043",
           "updateAt": "1498636753",
           "chargingStartTime": "1498273733",
           "logCollection": 1,
           "billingType": "Metered",
           "dataCenter": ,
           "vpc": null,
           "duration": "0",
           "fee": null,
           "hadoopVersion": null,
           "masterNodeSize": null,
           "coreNodeSize": null,
           "componentList": [{
           "id": null,
           "componentId": "MRS 3.1.0_001",
           "componentName": "Hadoop",
           "componentVersion": "3.1.1",
           "external_datasources": null,
           "componentDesc": "A distributed data processing framework for big data sets",
           "componentDescEn": null
        },
        {
           "id": null,
  ```

```
            "componentId": "MRS 3.1.0_002",
            "componentName": "HBase",
            "componentVersion": "2.2.3",
            "external_datasources": null,
            "componentDesc": "HBase is a column-based distributed storage system that features high reliability, performance, and scalability",
            "componentDescEn": null
        },
        {
            "id": null,
            "componentId": "MRS 3.1.0_003",
            "componentName": "Hive",
            "componentVersion": "3.1.0",
            "external_datasources": null,
            "componentDesc": "A data warehouse software that facilitates query and management of big data sets stored in distributed storage systems"
            "componentDescEn": null
        },
        {
            "id": null,
            "componentId": "MRS 3.1.0_004",
            "componentName": "Spark2x",
            "componentVersion": "2.4.5",
            "external_datasources": null,
            "componentDesc": "Spark2x is a fast general-purpose engine for large-scale data processing. It is developed based on the open-source Spark2.x version.",
            "componentDescEn": null
        },
        {
            "id": null,
            "componentId": "MRS 3.1.0_005",
            "componentName": "Tez",
            "componentVersion": "0.9.2",
            "external_datasources": null,
            "componentDesc": "An application framework which allows for a complex directed-acyclic-graph of tasks for processing data.",
            "componentDescEn": null
        },
        {
            "id": null,
            "componentId": "MRS 3.1.0_006",
            "componentName": "Flink",
            "componentVersion": "1.12.0",
            "external_datasources": null,
            "componentDesc": "Flink is an open-source message processing system that integrates streams in batches.",
            "componentDescEn": null
        },
        {
            "id": null,
            "componentId": "MRS 3.1.0_008",
            "componentName": "Kafka",
            "componentVersion": "2.11-2.4.0",
            "external_datasources": null,
            "componentDesc": "Kafka is a distributed message release and subscription system.",
            "componentDescEn": null
        },
        {
            "id": null,
            "componentId": "MRS 3.1.0_009",
            "componentName": "Flume",
            "componentVersion": "1.9.0",
            "external_datasources": null,
            "componentDesc": "Flume is a distributed, reliable, and highly available service for efficiently collecting, aggregating, and moving large amounts of log data",
            "componentDescEn": null
        },
        {
            "id": null,
```

```
            "componentId": "MRS 3.1.0_014",
            "componentName": "Hue",
            "componentVersion": "4.7.0",
            "external_datasources": null,
            "componentDesc": "Apache Hadoop UI",
            "componentDescEn": null
        },
        {
            "id": null,
            "componentId": "MRS 3.1.0_015",
            "componentName": "Oozie",
            "componentVersion": "5.1.0",
            "external_datasources": null,
            "componentDesc": "A Hadoop job scheduling system",
            "componentDescEn": null
        },
        {
            "id": null,
            "componentId": "MRS 3.1.0_022",
            "componentName": "Ranger",
            "componentVersion": "2.0.0",
            "external_datasources": null,
            "componentDesc": "Ranger is a centralized framework based on the Hadoop platform. It
provides permission control interfaces such as monitoring, operation, and management interfaces for
complex data.",
            "componentDescEn": null
        }],
            "externalIp": null,
            "externalAlternateIp": null,
            "internalIp": null,
            "deploymentId": null,
            "remark": "",
            "orderId": null,
            "azId": null,
            "masterNodeProductId": null,
            "masterNodeSpecId": null,
            "coreNodeProductId": null,
            "coreNodeSpecId": null,
            "azName": ,
            "instanceId": null,
            "vnc": "v2/5a3314075bfa49b9ae360f4ecd333695/servers/
e2cda891-232e-4703-995e-3b1406add01d/action",
            "tenantId": null,
            "volumeSize": 0,
            "volumeType": null,
            "subnetId": null,
            "subnetName": null,
            "securityGroupsId": null,
            "slaveSecurityGroupsId": null,
            "mrsManagerFinish": false,
            "stageDesc": "Installing MRS Manager",
            "safeMode": 0,
            "clusterVersion": null,
            "nodePublicCertName": null,
            "masterNodeIp": "unknown",
            "privateIpFirst": null,
            "errorInfo": "",
            "clusterType": 0,
            "nodeGroups": [
              {
                "groupName": "master_node_default_group",
                "nodeNum": 1,
                "nodeSize": "",
                "nodeSpecId": "cdc6035a249a40249312f5ef72a23cd7",
                "vmProductId": "",
                "vmSpecCode": null,
                "nodeProductId": "dc970349d128460e960a0c2b826c427c",
                "rootVolumeSize": 480,
                "rootVolumeProductId": "16c1dcf0897249758b1ec276d06e0572",
```

```
            "rootVolumeType": "SATA",
            "rootVolumeResourceSpecCode": "",
            "rootVolumeResourceType": "",
            "dataVolumeType": "SATA",
            "dataVolumeCount": 1,
            "dataVolumeSize": 600,
            "dataVolumeProductId": "16c1dcf0897249758b1ec276d06e0572",
            "dataVolumeResourceSpecCode": "",
            "dataVolumeResourceType": "",
          },
          {
            "groupName": "core_node_analysis_group",
            "nodeNum": 1,
            "nodeSize": "",
            "nodeSpecId": "cdc6035a249a40249312f5ef72a23cd7",
            "vmProductId": "",
            "vmSpecCode": null,
            "nodeProductId": "dc970349d128460e960a0c2b826c427c",
            "rootVolumeSize": 480,
            "rootVolumeProductId": "16c1dcf0897249758b1ec276d06e0572",
            "rootVolumeType": "SATA",
            "rootVolumeResourceSpecCode": "",
            "rootVolumeResourceType": "",
            "dataVolumeType": "SATA",
            "dataVolumeCount": 1,
            "dataVolumeSize": 600,
            "dataVolumeProductId": "16c1dcf0897249758b1ec276d06e0572",
            "dataVolumeResourceSpecCode": "",
            "dataVolumeResourceType": "",
          },
          {
            "groupName": "task_node_analysis_group",
            "nodeNum": 1,
            "nodeSize": "",
            "nodeSpecId": "cdc6035a249a40249312f5ef72a23cd7",
            "vmProductId": "",
            "vmSpecCode": null,
            "nodeProductId": "dc970349d128460e960a0c2b826c427c",
            "rootVolumeSize": 480,
            "rootVolumeProductId": "16c1dcf0897249758b1ec276d06e0572",
            "rootVolumeType": "SATA",
            "rootVolumeResourceSpecCode": "",
            "rootVolumeResourceType": "",
            "dataVolumeType": "SATA",
            "dataVolumeCount": 1,
            "dataVolumeSize": 600,
            "dataVolumeProductId": "16c1dcf0897249758b1ec276d06e0572",
            "dataVolumeResourceSpecCode": "",
            "dataVolumeResourceType": "",
          }

        ],
        "taskNodeGroups": [
          {
            "groupName": "task_node_default_group",
            "nodeNum": 1,
            "nodeSize": "",
            "nodeSpecId": "cdc6035a249a40249312f5ef72a23cd7",
            "vmProductId": "",
            "vmSpecCode": null,
            "nodeProductId": "dc970349d128460e960a0c2b826c427c",
            "rootVolumeSize": 480,
            "rootVolumeProductId": "16c1dcf0897249758b1ec276d06e0572",
            "rootVolumeType": "SATA",
            "rootVolumeResourceSpecCode": "",
            "rootVolumeResourceType": "",
            "dataVolumeType": "SATA",
            "dataVolumeCount": 1,
            "dataVolumeSize": 600,
```

```
            "dataVolumeProductId": "16c1dcf0897249758b1ec276d06e0572",
            "dataVolumeResourceSpecCode": "",
            "dataVolumeResourceType": "",
          }
        ],
      "masterDataVolumeType": "SATA",
      "masterDataVolumeSize": 600,
      "masterDataVolumeCount": 1,
      "coreDataVolumeType": "SATA",
      "coreDataVolumeSize": 600,
      "coreDataVolumeCount": 1,
    }
  ]
}
```


Status Codes

Table 6-32 describes the status code.

Table 6-32 Status code

Status Code	Description
200	The cluster list information has been queried.

See **Status Codes**.

Error Codes

See **Error Codes**.

6.1.4 Querying Cluster Details

Function

This API is used to query details about a specified cluster. This API is incompatible with Sahara.

URI

- Format

  GET /v1.1/{project_id}/cluster_infos/{cluster_id}

- Parameter description

Table 6-33 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.

Parameter	Mandatory	Type	Description
cluster_id	Yes	String	Cluster ID For details about how to obtain the value, see **Obtaining the MRS Cluster Information**.

Request Parameters

Request parameters

None

Response Parameters

Table 6-34 Response body parameter

Parameter	Type	Description
cluster	Cluster object	Cluster parameters. For details, see **Table 6-35**.

Table 6-35 Response parameters

Parameter	Type	Description
clusterId	String	Cluster ID
clusterName	String	Cluster name
masterNodeNum	String	Number of Master nodes deployed in a cluster
coreNodeNum	String	Number of Core nodes deployed in a cluster
totalNodeNum	String	Total number of nodes deployed in a cluster

Parameter	Type	Description
clusterState	String	Cluster status. Valid values include: • **starting**: The cluster is being started. • **running**: The cluster is running. • **terminated**: The cluster has been terminated. • **failed**: The cluster fails. • **abnormal**: The cluster is abnormal. • **terminating**: The cluster is being terminated. • **frozen**: The cluster has been frozen. • **scaling-out**: The cluster is being scaled out. • **scaling-in**: The cluster is being scaled in.
createAt	String	Cluster creation time, which is a 10-bit timestamp
updateAt	String	Cluster update time, which is a 10-bit timestamp
dataCenter	String	Cluster work region
vpc	String	VPC name
vpcId	String	VPC ID
hadoopVersion	String	Hadoop version
masterNodeSize	String	Instance specifications of a Master node
coreNodeSize	String	Instance specifications of a Core node
componentList	Array	Component list. For details, see **Table 6-36**.
externalIp	String	External IP address
externalAlterna-teIp	String	Backup external IP address
internalIp	String	Internal IP address
deploymentId	String	Cluster deployment ID
remark	String	Cluster remarks
orderId	String	Cluster creation order ID
azId	String	AZ ID

Parameter	Type	Description
masterNodeProductId	String	Product ID of a Master node
masterNodeSpecId	String	Specification ID of a Master node
coreNodeProductId	String	Product ID of a Core node
coreNodeSpecId	String	Specification ID of a Core node
azName	String	AZ name
azCode	String	AZ name (en).
availabilityZoneId	String	AZ.
instanceId	String	Instance ID
vnc	String	URI for remotely logging in to an ECS
tenantId	String	Project ID
volumeSize	Integer	Disk storage space
volumeType	String	Disk type.
subnetId	String	Subnet ID
subnetName	String	Subnet name
securityGroupsId	String	Security group ID
slaveSecurityGroupsId	String	Security group ID of a non-Master node. Currently, one MRS cluster uses only one security group. Therefore, this field has been discarded. This field returns the same value as **securityGroupsId** does for compatibility consideration.
bootstrapscripts	Array	Bootstrap action script information. For more parameter description, see **Table 6-38**.

Parameter	Type	Description
stageDesc	String	Cluster operation progress description. The cluster installation progress includes: ● Verifying cluster parameters: Cluster parameters are being verified. ● Applying for cluster resources: Cluster resources are being applied for. ● Creating VMs: The VMs are being created. ● Initializing VMs: The VMs are being initialized. ● Installing MRS Manager: MRS Manager is being installed. ● Deploying the cluster: The cluster is being deployed. ● Cluster installation failed: Failed to install the cluster. The cluster scale-out progress includes: ● Preparing for scale-out: Cluster scale-out is being prepared. ● Creating VMs: The VMs are being created. ● Initializing VMs: The VMs are being initialized. ● Adding nodes to the cluster: The nodes are being added to the cluster. ● Scale-out failed: Failed to scale out the cluster. The cluster scale-in progress includes: ● Preparing for scale-in: Cluster scale-in is being prepared. ● Decommissioning instance: The instance is being decommissioned. ● Deleting VMs: The VMs are being deleted. ● Deleting nodes from the cluster: The nodes are being deleted from the cluster. ● Scale-in failed: Failed to scale in the cluster.

Parameter	Type	Description
		If the cluster installation, scale-out, or scale-in fails, **stageDesc** will display the failure cause.
isMrsManagerFinish	Boolean	Whether MRS Manager installation is finished during cluster creation. • **true**: MRS Manager installation is finished. • **false**: MRS Manager installation is not finished.
safeMode	Integer	Running mode of an MRS cluster • **0**: Normal cluster • **1**: Security cluster
clusterVersion	String	Cluster version
nodePublicCertName	String	Name of the public key file
masterNodeIp	String	IP address of a Master node
privateIpFirst	String	Preferred private IP address
errorInfo	String	Error message
tags	String	Tag information
clusterType	Integer	Cluster type
logCollection	Integer	Whether to collect logs when cluster installation fails • **0**: Do not collect. • **1**: Collect.
taskNodeGroups	List<NodeGroup>	List of Task nodes. For more parameter description, see **Table 6-37**.
nodeGroups	List<NodeGroup>	List of Master, Core and Task nodes. For more parameter description, see **Table 6-37**.
masterDataVolumeType	String	Data disk storage type of the Master node. Currently, SATA, SAS, and SSD are supported.
masterDataVolumeSize	Integer	Data disk storage space of the Master node. To increase data storage capacity, you can add disks at the same time when creating a cluster. Value range: 100 GB to 32,000 GB

Parameter	Type	Description
masterDataVolumeCount	Integer	Number of data disks of the Master node. The value can be set to **1** only.
coreDataVolumeType	String	Data disk storage type of the Core node. Currently, SATA, SAS, and SSD are supported.
coreDataVolumeSize	Integer	Data disk storage space of the Core node. To increase data storage capacity, you can add disks at the same time when creating a cluster. Value range: 100 GB to 32,000 GB
coreDataVolumeCount	Integer	Number of data disks of the Core node. Value range: 1 to 10
scale	String	Node change status. If this parameter is left blank, the cluster nodes are not changed. Possible values: ● **scaling-out**: The cluster is being scaled out. ● **scaling-in**: The cluster is being scaled in. ● **scaling-error**: The cluster is in the running state and fails to be scaled in or out or the specifications fail to be scaled up for the last time. ● **scaling-up**: The Master node specifications are being scaled up. ● **scaling_up_first**: The standby Master node specifications are being scaled up. ● **scaled_up_first**: The standby Master node specifications have been scaled up successfully. ● **scaled-up-success**: The Master node specifications have been scaled up successfully.

Table 6-36 componentList parameters

Parameter	Type	Description
componentId	String	Component ID For example, the **component_id** of Hadoop is MRS 3.1.0_001.
componentName	String	Component name
componentVersion	String	Component version
componentDesc	String	Component description

Table 6-37 NodeGroup parameters

Parameter	Type	Description
GroupName	String	Node group name.
NodeNum	Integer	Number of nodes. The value ranges from 0 to 500. The minimum number of Master and Core nodes is 1 and the total number of Core and Task nodes cannot exceed 500.
NodeSize	String	Instance specifications of a node.
NodeSpecId	String	Instance specification ID of a node
NodeProductId	String	Instance product ID of a node
VmProductId	String	VM product ID of a node
VmSpecCode	String	VM specifications of a node
RootVolumeSize	Integer	System disk size of a node. This parameter is not configurable and its default value is **40 GB**.
RootVolumeProductId	String	System disk product ID of a node
RootVolumeType	String	System disk type of a node
RootVolumeResourceSpecCode	String	System disk product specifications of a node
RootVolumeResourceType	String	System disk product type of a node

Parameter	Type	Description
DataVolumeType	String	Data disk storage type of a node. Currently, SATA, SAS, and SSD are supported. • SATA: Common I/O • SAS: High I/O • SSD: Ultra-high I/O
DataVolumeCount	Integer	Number of data disks of a node.
DataVolumeSize	Integer	Data disk storage space of a node.
DataVolumeProductId	String	Data disk product ID of a node
DataVolumeResourceSpecCode	String	Data disk product specifications of a node
DataVolumeResourceType	String	Data disk product type of a node

Table 6-38 bootstrapscripts parameters

Parameter	Type	Description
name	String	Name of a bootstrap action script. It must be unique in a cluster. The value can contain only digits, letters, spaces, hyphens (-), and underscores (_) and cannot start with a space. The value can contain 1 to 64 characters.

Parameter	Type	Description
uri	String	Path of the shell script. Set this parameter to an OBS bucket path or a local VM path. • OBS bucket path: Enter a script path manually. For example, enter the path of the public sample script provided by MRS. Example: **s3a://bootstrap/presto/presto-install.sh**. If **dualroles** is installed, the parameter of the **presto-install.sh** script is **dualroles**. If **worker** is installed, the parameter of the **presto-install.sh** script is **worker**. Based on the Presto usage habit, you are advised to install **dualroles** on the active Master nodes and **worker** on the Core nodes. • Local VM path: Enter a script path. The script path must start with a slash (/) and end with **.sh**.
parameters	String	Bootstrap action script parameters
nodes	Array String	Type of a node where the bootstrap action script is executed. The value can be Master, Core, or Task.
active_master	Boolean	Whether the bootstrap action script runs only on active Master nodes. The default value is **false**, indicating that the bootstrap action script can run on all Master nodes.
before_component_start	Boolean	Time when the bootstrap action script is executed. Currently, the following two options are available: **Before component start** and **After component start** The default value is **false**, indicating that the bootstrap action script is executed after the component is started.

Parameter	Type	Description
fail_action	String	Whether to continue executing subsequent scripts and creating a cluster after the bootstrap action script fails to be executed. • **continue**: Continue to execute subsequent scripts. • **errorout**: Stop the action. The default value is **errorout**, indicating that the action is stopped. **NOTE** You are advised to set this parameter to **continue** in the commissioning phase so that the cluster can continue to be installed and started no matter whether the bootstrap action is successful.
action_stages	Array of strings	Select the time when the bootstrap action script is executed. • **BEFORE_COMPONENT_FIRST_START**: before initial component starts • **AFTER_COMPONENT_FIRST_START**: after initial component starts • **BEFORE_SCALE_IN**: before scale-in • **AFTER_SCALE_IN**: after scale-in • **BEFORE_SCALE_OUT**: before scale-out • **AFTER_SCALE_OUT**: after scale-out

Example

- Example request
  ```
  GET /v1.1/{project_id}/cluster_infos/{cluster_id}
  ```
- Example response
  ```
  {
      "cluster":{
          "clusterId":"bdb064ff-2855-4624-90d5-e9a6376abd6e",
          "clusterName":"c17022001",
          "masterNodeNum":"2",
          "coreNodeNum":"3",
          "clusterState":"scaling-in",
          "stageDesc": null,
          "createAt":"1487570757",
          "updateAt":"1487668974",
          "billingType":"Metered",
          "dataCenter":,
          "vpc": "vpc-autotest",
          "vpcId": "e2978efd-ca12-4058-9332-1ca0bfbab592",
          "duration":"0",
          "fee":"0",
          "hadoopVersion":"",
          "masterNodeSize":"",
          "coreNodeSize":"",
  ```

```
        "componentList": [{
          "id": null,
          "componentId": "MRS 3.1.0_001",
          "componentName": "Hadoop",
          "componentVersion": "3.1.1",
          "external_datasources": null,
          "componentDesc": "A distributed data processing framework for big data sets",
          "componentDescEn": null
        },
        {
          "id": null,
          "componentId": "MRS 3.1.0_002",
          "componentName": "HBase",
          "componentVersion": "2.2.3",
          "external_datasources": null,
          "componentDesc": "HBase is a column-based distributed storage system that features high
reliability, performance, and scalability",
          "componentDescEn": null
        },
        {
          "id": null,
          "componentId": "MRS 3.1.0_003",
          "componentName": "Hive",
          "componentVersion": "3.1.0",
          "external_datasources": null,
          "componentDesc": "A data warehouse software that facilitates query and management of big
data sets stored in distributed storage systems"
          "componentDescEn": null
        },
        {
          "id": null,
          "componentId": "MRS 3.1.0_004",
          "componentName": "Spark2x",
          "componentVersion": "2.4.5",
          "external_datasources": null,
          "componentDesc": "Spark2x is a fast general-purpose engine for large-scale data processing.
It is developed based on the open-source Spark2.x version.",
          "componentDescEn": null
        },
        {
          "id": null,
          "componentId": "MRS 3.1.0_005",
          "componentName": "Tez",
          "componentVersion": "0.9.2",
          "external_datasources": null,
          "componentDesc": "An application framework which allows for a complex directed-acyclic-
graph of tasks for processing data.",
          "componentDescEn": null
        },
        {
          "id": null,
          "componentId": "MRS 3.1.0_006",
          "componentName": "Flink",
          "componentVersion": "1.12.0",
          "external_datasources": null,
          "componentDesc": "Flink is an open-source message processing system that integrates
streams in batches.",
          "componentDescEn": null
        },
        {
          "id": null,
          "componentId": "MRS 3.1.0_008",
          "componentName": "Kafka",
          "componentVersion": "2.11-2.4.0",
          "external_datasources": null,
          "componentDesc": "Kafka is a distributed message release and subscription system.",
          "componentDescEn": null
        },
        {
```

```
            "id": null,
            "componentId": "MRS 3.1.0_009",
            "componentName": "Flume",
            "componentVersion": "1.9.0",
            "external_datasources": null,
            "componentDesc": "Flume is a distributed, reliable, and highly available service for efficiently collecting, aggregating, and moving large amounts of log data",
            "componentDescEn": null
        },
        {
            "id": null,
            "componentId": "MRS 3.1.0_013",
            "componentName": "Loader",
            "componentVersion": "1.99.3",
            "external_datasources": null,
            "componentDesc": "Loader is a tool designed for efficiently transmitting a large amount of data between Apache Hadoop and structured databases (such as relational databases).",
            "componentDescEn": null
        },
        {
            "id": null,
            "componentId": "MRS 3.1.0_014",
            "componentName": "Hue",
            "componentVersion": "4.7.0",
            "external_datasources": null,
            "componentDesc": "Apache Hadoop UI",
            "componentDescEn": null
        },
        {
            "id": null,
            "componentId": "MRS 3.1.0_015",
            "componentName": "Oozie",
            "componentVersion": "5.1.0",
            "external_datasources": null,
            "componentDesc": "A Hadoop job scheduling system",
            "componentDescEn": null
        },
        {
            "id": null,
            "componentId": "MRS 3.1.0_022",
            "componentName": "Ranger",
            "componentVersion": "2.0.0",
            "external_datasources": null,
            "componentDesc": "Ranger is a centralized framework based on the Hadoop platform. It provides permission control interfaces such as monitoring, operation, and management interfaces for complex data.",
            "componentDescEn": null
        }],
        "externalIp":"100.XXX.XXX.XXX",
        "externalAlternateIp":"100.XXX.XXX.XXX",
        "internalIp":"192.XXX.XXX.XXX",
        "deploymentId":"4ac46ca7-a488-4b91-82c2-e4d7aa9c40c2",
        "remark":"",
        "orderId":"null",
        "azId":"null",
        "masterNodeProductId":"b35cf2d2348a445ca74b32289a160882",
        "masterNodeSpecId":"8ab05e503b4c42abb304e2489560063b",
        "coreNodeProductId":"dc970349d128460e960a0c2b826c427c",
        "coreNodeSpecId":"cdc6035a249a40249312f5ef72a23cd7",
        "azName":,
        "instanceId":"4ac46ca7-a488-4b91-82c2-e4d7aa9c40c2",
        "vnc":null,
        "tenantId":"3f99e3319a8943ceb15c584f3325d064",
        "volumeSize":600,
        "volumeType":"SATA",
        "subnetId": "6b96eec3-4f8d-4c83-93e2-6ec625001d7c",
        "subnetName":"subnet-ftest",
        "securityGroupsId":"930e34e2-195d-401f-af07-0b64ea6603f8",
        "slaveSecurityGroupsId":"2ef3343e-3477-4a0d-80fe-4d874e4f81b8",
```

```
    "stageDesc": "Installing MRS Manager",
    "mrsManagerFinish": false,
    "safeMode":1,
    "clusterVersion":"MRS 3.1.0",
    "nodePublicCertName":"myp",
    "masterNodeIp":"192.XXX.XXX.XXX",
    "privateIpFirst":"192.XXX.XXX.XXX",
    "errorInfo":null,
    "tags":"k1=v1,k2=v2,k3=v3",
    "clusterType": 0,
    "logCollection": 1,
     "nodeGroups": [
       {
           "groupName": "master_node_default_group",
           "nodeNum": 1,
           "nodeSize": "",
           "nodeSpecId": "cdc6035a249a40249312f5ef72a23cd7",
           "vmProductId": "",
           "vmSpecCode": null,
           "nodeProductId": "dc970349d128460e960a0c2b826c427c",
           "rootVolumeSize": 480,
           "rootVolumeProductId": "16c1dcf0897249758b1ec276d06e0572",
           "rootVolumeType": "SATA",
           "rootVolumeResourceSpecCode": "",
           "rootVolumeResourceType": "",
           "dataVolumeType": "SATA",
           "dataVolumeCount": 1,
           "dataVolumeSize": 600,
           "dataVolumeProductId": "16c1dcf0897249758b1ec276d06e0572",
           "dataVolumeResourceSpecCode": "",
           "dataVolumeResourceType": ""
         },
         {
           "groupName": "core_node_analysis_group",
           "nodeNum": 1,
           "nodeSize": "",
           "nodeSpecId": "cdc6035a249a40249312f5ef72a23cd7",
           "vmProductId": "",
           "vmSpecCode": null,
           "nodeProductId": "dc970349d128460e960a0c2b826c427c",
           "rootVolumeSize": 480,
           "rootVolumeProductId": "16c1dcf0897249758b1ec276d06e0572",
           "rootVolumeType": "SATA",
           "rootVolumeResourceSpecCode": "",
           "rootVolumeResourceType": "",
           "dataVolumeType": "SATA",
           "dataVolumeCount": 1,
           "dataVolumeSize": 600,
           "dataVolumeProductId": "16c1dcf0897249758b1ec276d06e0572",
           "dataVolumeResourceSpecCode": "",
           "dataVolumeResourceType": ""
         },
         {
           "groupName": "task_node_analysis_group",
           "nodeNum": 1,
           "nodeSize": "",
           "nodeSpecId": "cdc6035a249a40249312f5ef72a23cd7",
           "vmProductId": "",
           "vmSpecCode": null,
           "nodeProductId": "dc970349d128460e960a0c2b826c427c",
           "rootVolumeSize": 480,
           "rootVolumeProductId": "16c1dcf0897249758b1ec276d06e0572",
           "rootVolumeType": "SATA",
           "rootVolumeResourceSpecCode": "",
           "rootVolumeResourceType": "",
           "dataVolumeType": "SATA",
           "dataVolumeCount": 1,
           "dataVolumeSize": 600,
           "dataVolumeProductId": "16c1dcf0897249758b1ec276d06e0572",
```

```
            "dataVolumeResourceSpecCode": "",
            "dataVolumeResourceType": ""
          }

        ],
      "taskNodeGroups": [
        {
          "groupName": "task_node_default_group",
          "nodeNum": 1,
          "nodeSize": "",
          "nodeSpecId": "cdc6035a249a40249312f5ef72a23cd7",
          "vmProductId": "",
          "vmSpecCode": null,
          "nodeProductId": "dc970349d128460e960a0c2b826c427c",
          "rootVolumeSize": 480,
          "rootVolumeProductId": "16c1dcf0897249758b1ec276d06e0572",
          "rootVolumeType": "SATA",
          "rootVolumeResourceSpecCode": "",
          "rootVolumeResourceType": "",
          "dataVolumeType": "SATA",
          "dataVolumeCount": 1,
          "dataVolumeSize": 600,
          "dataVolumeProductId": "16c1dcf0897249758b1ec276d06e0572",
          "dataVolumeResourceSpecCode": "",
          "dataVolumeResourceType": "",
          "AutoScalingPolicy": null
          }
        ],
       "masterDataVolumeType": "SATA",
       "masterDataVolumeSize": 600,
       "masterDataVolumeCount": 1,
       "coreDataVolumeType": "SATA",
       "coreDataVolumeSize": 600,
       "coreDataVolumeCount": 1,
    }
}
```


Status Codes

Table 6-39 describes the status code.

Table 6-39 Status code

Status Code	Description
200	Cluster details have been queried.

See **Status Codes**.

Error Codes

See **Error Codes**.

6.1.5 Querying a Host List

Function

This API is used to query a host list of a specified cluster.

URI

- Format

  GET /v1.1/{project_id}/clusters/{cluster_id}/hosts

- Parameter description

  Table 6-40 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.
cluster_id	Yes	String	Cluster ID For details about how to obtain the value, see **Obtaining the MRS Cluster Information**.

Request Parameters

Table 6-41 Request parameters

Parameter	Mandatory	Type	Description
pageSize	No	Integer	Maximum number of clusters displayed on a page Value range: [1-2147483646]. The default value is **10**.
currentPage	No	Integer	Current page number The default value is **1**.

Response Parameters

Table 6-42 Response parameters

Parameter	Type	Description
total	Integer	Total number of hosts in a list
hosts	Array	Host parameters For details, see **Table 6-43**.

Table 6-43 Hosts parameters

Parameter	Type	Description
id	String	VM ID
ip	String	VM IP address
availability_zone_id	String	The AZ.
tags	Array of TagPlain objects	The tag list. For details, see **Table 6-44**.
flavor	String	VM flavor ID
type	String	VM type Currently, MasterNode, CoreNode, and TaskNode are supported.
name	String	VM name
status	String	Current VM state
mem	String	Memory
cpu	String	Number of CPU cores
root_volume_size	String	OS disk capacity
data_volume_type	String	Data disk type
data_volume_size	Integer	Data disk capacity
data_volume_count	Integer	Number of data disks

Table 6-44 TagPlain parameters

Parameter	Type	Description
key	String	The tag key. A tag key cannot contain special characters (=*<>\,|/) or start or end with spaces.
value	String	The value. A tag value cannot contain special characters (=*<>\,|/) or start or end with spaces.

Example

- Example request
  ```
  GET /v1.1/{project_id}/clusters/{cluster_id}/hosts
  ```

- Example response


```
{
 "total": 5,
 "hosts": [
  {
   "id": "063d1d47-ae91-4a48-840c-b3cfe4efbcf0",
   "name": "a78e161c-d14f-4b68-8c2d-0219920ce844_node_core_IQhiC",
   "ip": "192.168.0.169",
   "status": "ACTIVE",
   "flavor": "c6.4xlarge.4.linux.mrs",
   "type": "Core",
   "mem": "16384",
   "cpu": "8",
   "root_volume_size": "480",
   "data_volume_type": "SATA",
   "data_volume_size": 600,
   "data_volume_count": 1
  },
  {
   "id": "dc5c6208-faa2-4727-a65a-2b1ce235d350",
   "name": "a78e161c-d14f-4b68-8c2d-0219920ce844_node_master1_ASzkl",
   "ip": "192.168.0.156",
   "status": "ACTIVE",
   "flavor": "c2.4xlarge.linux.mrs",
   "type": "Master",
   "mem": "32768",
   "cpu": "16",
   "root_volume_size": "480",
   "data_volume_type": "SATA",
   "data_volume_size": 600,
   "data_volume_count": 1
  },
  {
   "id": "c0ce793d-848b-448a-835b-ea0cac534b09",
   "name": "a78e161c-d14f-4b68-8c2d-0219920ce844_node_core_ANnRN",
   "ip": "192.168.0.243",
   "status": "ACTIVE",
   "flavor": "c6.4xlarge.4.linux.mrs",
   "type": "Core",
   "mem": "16384",
   "cpu": "8",
   "root_volume_size": "480",
   "data_volume_type": "SATA",
   "data_volume_size": 600,
   "data_volume_count": 1
  },
  {
   "id": "95c23e43-ef6e-4732-b6ed-a5f1c7779fae",
   "name": "a78e161c-d14f-4b68-8c2d-0219920ce844_node_core_uRRiA",
   "ip": "192.168.0.126",
   "status": "ACTIVE",
   "flavor": "c6.4xlarge.4.linux.mrs",
   "type": "Core",
   "mem": "16384",
   "cpu": "8",
   "root_volume_size": "480",
   "data_volume_type": "SATA",
   "data_volume_size": 600,
   "data_volume_count": 1
  },
  {
   "id": "63bdbf75-1133-4a94-8c27-1fa12c8b9e70",
   "name": "a78e161c-d14f-4b68-8c2d-0219920ce844_node_master2_StqFu",
   "ip": "192.168.0.22",
   "status": "ACTIVE",
   "flavor": "c2.4xlarge.linux.mrs",
   "type": "Master",
   "mem": "32768",
   "cpu": "16",
```

```
    "root_volume_size": "480",
    "data_volume_type": "SATA",
    "data_volume_size": 600,
    "data_volume_count": 1
   }
 ]
}
```

Status Codes

Table 6-45 describes the status code.

Table 6-45 Status code

Status Code	Description
200	The host list information has been queried.

See **Status Codes**.

Error Codes

See **Error Codes**.

6.1.6 Terminating a Cluster

Function

This API is used to terminate a cluster after data processing and analysis are complete or the cluster is abnormal. This API is compatible with Sahara.

Clusters in any of the following states cannot be terminated:

- scaling-out
- scaling-in
- starting
- terminating
- terminated
- failed

URI

- Format

  DELETE /v1.1/{project_id}/clusters/{cluster_id}
- Parameter description

Table 6-46 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.
cluster_id	Yes	String	Cluster ID For details about how to obtain the cluster ID, see **Obtaining the MRS Cluster Information**.

Request Parameters

None

Response Parameters

None

Example

- Example request
  ```
  DELETE /v1.1/{project_id}/clusters/{cluster_id}
  ```
- Example response

  None

Status Codes

Table 6-47 describes the status codes.

Table 6-47 Status codes

Status Code	Description
200	The request is executed.
204	The cluster has been terminated.

See **Status Codes**.

Error Codes

See **Error Codes**.

6.2 Job Object APIs

6.3 Job Execution Object APIs

6.4 Auto Scaling APIs

6.4.1 Configuring an Auto Scaling Rule

Function

This API is used to configure auto scaling rules.

The API used for cluster creation and job execution can also be used to create an auto scaling rule.

URI

- Format

  POST /v1.1/{project_id}/autoscaling-policy/{cluster_id}

- Parameter description

  Table 6-48 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.
cluster_id	Yes	String	Cluster ID For details about how to obtain the cluster ID, see **Obtaining the MRS Cluster Information**.

Request Parameters

Table 6-49 Request body parameters

Parameter	Mandatory	Type	Description
node_group	Yes	String	Type of the node to which an auto scaling rule applies. Currently, only Task nodes support auto scaling rules, that is, the request value is **task_node_default_group**.

Parameter	Mandatory	Type	Description
auto_scaling_policy	Yes	AutoScalingPolicy object	The auto scaling policy. **Table 6-50** describes these parameters.

Table 6-50 AutoScalingPolicy object parameters

Parameter	Mandatory	Type	Description
auto_scaling_enable	Yes	Boolean	Whether to enable the auto scaling rule.
min_capacity	Yes	Integer	Minimum number of nodes left in the node group. Value range: 0 to 500
max_capacity	Yes	Integer	Maximum number of nodes in the node group. Value range: 0 to 500
resources_plans	No	List	Resource plan list. For details, see **Table 6-51**. If this parameter is left blank, the resource plan is disabled. When auto scaling is enabled, either a resource plan or an auto scaling rule must be configured.
exec_scripts	No	List	List of custom scaling automation scripts. For details, see **Table 6-52**. If this parameter is left blank, a hook script is disabled.
rules	No	List	List of auto scaling rules. For details, see **Table 6-53**. When auto scaling is enabled, either a resource plan or an auto scaling rule must be configured.

Table 6-51 resources_plan parameters

Parameter	Mandatory	Type	Description
period_type	Yes	String	Cycle type of a resource plan. Currently, only the following cycle type is supported: daily
start_time	Yes	String	Start time of a resource plan. The value is in the format of **hour:minute**, indicating that the time ranges from 0:00 to 23:59.
end_time	Yes	String	End time of a resource plan. The value is in the same format as that of **start_time**. The interval between **end_time** and **start_time** must be greater than or equal to 30 minutes.
min_capacity	Yes	Integer	Minimum number of the preserved nodes in a node group in a resource plan. Value range: 0 to 500
max_capacity	Yes	Integer	Maximum number of the preserved nodes in a node group in a resource plan. Value range: 0 to 500

Table 6-52 exec_script parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Name of a custom automation script. It must be unique in a same cluster. The value can contain only digits, letters, spaces, hyphens (-), and underscores (_) and must not start with a space. The value can contain 1 to 64 characters.

Parameter	Mandatory	Type	Description
uri	Yes	String	Path of a custom automation script. Set this parameter to an OBS bucket path or a local VM path. ● OBS bucket path: Enter a script path manually. for example, **s3a://*XXX*/scale.sh**. ● Local VM path: Enter a script path. The script path must start with a slash (/) and end with **.sh**.
parameters	No	String	Parameters of a custom automation script. ● Multiple parameters are separated by space. ● The following predefined system parameters can be transferred: – *${mrs_scale_node_num}*: Number of the nodes to be added or removed – *${mrs_scale_type}*: Scaling type. The value can be **scale_out** or **scale_in**. – *${mrs_scale_node_hostnames}*: Host names of the nodes to be added or removed – *${mrs_scale_node_ips}*: IP addresses of the nodes to be added or removed – *${mrs_scale_rule_name}*: Name of the rule that triggers auto scaling ● Other user-defined parameters are used in the same way as those of common shell scripts. Parameters are separated by space.
nodes	Yes	Array of strings	Type of a node where the custom automation script is executed. The node type can be Master, Core, or Task.

Parameter	Mandatory	Type	Description
active_master	No	Boolean	Whether the custom automation script runs only on the active Master node. The default value is **false**, indicating that the custom automation script can run on all Master nodes.
action_stage	Yes	String	Time when a script is executed. The following four options are supported: ● **before_scale_out**: before scale-out ● **before_scale_in**: before scale-in ● **after_scale_out**: after scale-out ● **after_scale_in**: after scale-in
fail_action	Yes	String	Whether to continue to execute subsequent scripts and create a cluster after the custom automation script fails to be executed. ● **continue**: Continue to execute subsequent scripts. ● **errorout**: Stop the action. **NOTE** ● You are advised to set this parameter to **continue** in the commissioning phase so that the cluster can continue to be installed and started no matter whether the custom automation script is executed successfully. ● The scale-in operation cannot be undone. Therefore, **fail_action** must be set to **continue** for the scripts that are executed after scale-in.

Table 6-53 rules parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Name of an auto scaling rule. A cluster name can contain only 1 to 64 characters. Only letters, digits, hyphens (-), and underscores (_) are allowed. Rule names must be unique in a node group.
description	No	String	Description about an auto scaling rule. It contains a maximum of 1,024 characters.
adjustment_type	Yes	String	Auto scaling rule adjustment type. Possible values: • **scale_out**: cluster scale-out • **scale_in**: cluster scale-in
cool_down_minutes	Yes	Integer	Cluster cooling time after an auto scaling rule is triggered, when no auto scaling operation is performed. The unit is minute. Value range: 0 to 10,080. One week is equal to 10,080 minutes.
scaling_adjustment	Yes	Integer	Number of nodes that can be adjusted once. Value range: 1 to 100
trigger	Yes	Trigger object	Condition for triggering a rule. For details, see **Table 6-54**.

Table 6-54 trigger parameters

Parameter	Mandatory	Type	Description
metric_name	Yes	String	Metric name. This triggering condition makes a judgment according to the value of the metric. A metric name contains a maximum of 64 characters.

Parameter	Mandatory	Type	Description
metric_value	Yes	String	Metric threshold to trigger a rule The parameter value can be an integer or number with two decimal places only.
comparison_operator	No	String	Metric judgment logic operator. Possible values: • **LT**: less than • **GT**: greater than • **LTOE**: less than or equal to • **GTOE**: greater than or equal to
evaluation_periods	Yes	Integer	Number of consecutive five-minute periods, during which a metric threshold is reached Value range: 1 to 288

Response Parameters

Table 6-55 Response parameter

Parameter	Type	Description
result	String	Operation result • succeeded: The operation is successful.

Example

- Example request

```
{
  "node_group":"task_node_default_group",
  "auto_scaling_policy": {
      "auto_scaling_enable": true,
      "min_capacity": "1",
      "max_capacity": "3",
      "resources_plans": [{
         "period_type": "daily",
         "start_time": "9:50",
         "end_time": "10:20",
         "min_capacity": "2",
         "max_capacity": "3"
         },{
         "period_type": "daily",
         "start_time": "10:20",
         "end_time": "12:30",
         "min_capacity": "0",
         "max_capacity": "2"
       }],
       "exec_scripts": [{
```

```
      "name": "before_scale_out",
      "uri": "s3a://XXX/zeppelin_install.sh",
      "parameters": "",
      "nodes": [
        "master",
        "core",
        "task"
      ],
      "active_master": "true",
      "action_stage": "before_scale_out",
      "fail_action": "continue"
      },{
      "name": "after_scale_out",
      "uri": "s3a://XXX/storm_rebalance.sh",
      "parameters": "",
      "nodes": [
        "master",
        "core",
        "task"
      ],
      "active_master": "true",
      "action_stage": "after_scale_out",
      "fail_action": "continue"
    }],
    "rules": [{
      "name": "default-expand-1",
      "adjustment_type": "scale_out",
      "cool_down_minutes": 5,
      "scaling_adjustment": 1,
      "trigger": {
        "metric_name": "YARNMemoryAvailablePercentage",
        "metric_value": "25",
        "comparison_operator": "LT",
        "evaluation_periods": 10
        }
      },
      {
       "name": "default-shrink-1",
       "adjustment_type": "scale_in",
       "cool_down_minutes": 5,
       "scaling_adjustment": 1,
       "trigger": {
          "metric_name": "YARNMemoryAvailablePercentage",
          "metric_value": "70",
          "comparison_operator": "GT",
          "evaluation_periods": 10
          }
       }]
   }
}
```

NOTE

A new auto scaling rule will overwrite the auto scaling rule saved in the original database. If you want to modify the original rule, query the original rule first, modify the rule, and submit a modification task. For details, see **Querying Cluster Details**.

- Example response

```
{      "result": "succeeded"  }
```

Status Codes

Table 6-56 describes the status code.

Table 6-56 Status code

Status Code	Description
200	The cluster has been created.

See **Status Codes**.

Error Codes

See **Error Codes**.

6.5 Tag Management APIs

6.5.1 Adding Tags to a Specified Cluster

Function

This API is used to add tags to a specified cluster.

A cluster has a maximum of 10 tags. This API is idempotent. If a tag to be created has the same key as an existing tag, the tag will overwrite the existing one.

URI

- Format

  POST /v1.1/{project_id}/clusters/{cluster_id}/tags
- Parameter description

  Table 6-57 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.
cluster_id	Yes	String	Cluster ID For details about how to obtain the cluster ID, see **Obtaining the MRS Cluster Information**.

Request Parameters

Table 6-58 Request body parameter

Parameter	Mandatory	Type	Description
tag	Yes	Tag object	The cluster tag.

Table 6-59 tags parameters

Parameter	Mandatory	Type	Description
key	Yes	String	Key.
value	Yes	String	Tag value.

Response Parameters

None

Example

- Example request

```
{
   "tag":
      {
         "key":"DEV",
         "value":"DEV1"
      }
}
```

- Example response

  None

Status Codes

Table 6-60 describes the status code.

Table 6-60 Status code

Status Code	Description
204	The operation is successful.

Error Codes

See **Error Codes**.

6.5.2 Querying Tags of a Specified Cluster

Function

This API is used to query tags of a specified cluster.

URI

- Format

  GET /v1.1/{project_id}/clusters/{cluster_id}/tags
- Parameter description

  Table 6-61 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.
cluster_id	Yes	String	Cluster ID For details about how to obtain the cluster ID, see **Obtaining the MRS Cluster Information**.

Request Parameters

None

Response Parameters

Table 6-62 Response parameter

Parameter	Type	Description
tags	Array of TagPlain objects	Tag list. For details, see **Table 6-63**.

Table 6-63 TagPlain parameters

Parameter	Type	Description
key	String	Key.
value	String	Tag value.

Example

- Example request
  ```
  GET /v1.1/{project_id}/clusters/{cluster_id}/tags
  ```
- Example response
  ```
  {
      "tags": [
       {
          "key": "key1",
          "value": "value1"
       },
       {
          "key": "key2",
          "value": "value3"
       }
    ]
  }
  ```

Status Codes

Table 6-64 describes the status code.

Table 6-64 Status code

Status Code	Description
200	The operation is successful.

Error Codes

See **Error Codes**.

6.5.3 Deleting Tags from a Specified Cluster

Function

This API is used to delete tags from a specified cluster.

URI

- Format

  DELETE /v1.1/{project_id}/clusters/{cluster_id}/tags/{key}
- Parameter description

Table 6-65 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.

Parameter	Mandatory	Type	Description
cluster_id	Yes	String	Cluster ID For details about how to obtain the cluster ID, see **Obtaining a Cluster ID**.
key	Yes	String	The tag key.

Request Parameters

Request parameters

None

Response Parameters

Response parameters

None

Example

- Example request
  DELETE /v1.1/{project_id}/clusters/{cluster_id}/tags/{key}
- Example response

  None

Status Codes

See **Status Codes**.

Error Codes

See **Error Codes**.

6.5.4 Adding Tags to a Cluster in Batches

Function

This API is used to add tags to a specified cluster in batches.

A cluster can have a maximum of 10 tags.

This API is idempotent. If a tag to be created has the same key as an existing tag in a cluster, the tag will overwrite the existing one.

URI

POST /v1.1/{project_id}/clusters/{cluster_id}/tags/action

Table 6-66 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.
cluster_id	Yes	String	Cluster ID For details about how to obtain the cluster ID, see **Obtaining a Cluster ID**.

Request Parameters

Table 6-67 Request parameters

Parameter	Mandatory	Type	Description
action	Yes	String	Operation to be performed. The value can be set to **create** or **delete** only.
tags	Yes	Array of tag objects	Tag list. For details about the parameter, see **Table 6-68**.

Table 6-68 tags parameters

Parameter	Mandatory	Type	Description
key	Yes	String	The tag key. • It contains a maximum of 36 Unicode characters and cannot be an empty string. • The tag key of a resource must be unique.
value	Yes	String	The value. • The value contains a maximum of 43 Unicode characters and can be an empty string.

Response Parameters

None

Example Request

Add tags to a cluster in batches.

```
POST /v1.1/{project_id}/clusters/{cluster_id}/tags/action

{
  "action" : "create",
  "tags" : [ {
    "key" : "DEV1",
    "value" : "DEV1"
  }, {
    "key" : "DEV2",
    "value" : "DEV2"
  } ]
}
```

Example Response

None

Status Codes

See **Status Codes**.

Error Codes

See **Error Codes**.

6.5.5 Deleting Tags from a Cluster in Batches

Function

This API is used to delete tags from a specified cluster in batches.

A cluster can have a maximum of 10 tags.

This API is idempotent.

- When tags are being deleted and some tags do not exist, the operation is considered to be successful by default. The character set of the tags will not be checked. A key and a value can respectively contain up to 36 and 43 Unicode characters. The tags structure cannot be missing. The key cannot be empty or an empty string.

URI

POST /v1.1/{project_id}/clusters/{cluster_id}/tags/action

Table 6-69 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	The project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.

Parameter	Mandatory	Type	Description
cluster_id	Yes	String	The cluster ID. For details about how to obtain the cluster ID, see **Obtaining a Cluster ID**.

Request Parameters

Table 6-70 Request parameters

Parameter	Mandatory	Type	Description
action	Yes	String	Operation to be performed. The value can be **create** or **delete** only.
tags	Yes	Array of tag objects	The tag list. For details about the parameter, see **Table 6-71**.

Table 6-71 tags parameters

Parameter	Mandatory	Type	Description
key	Yes	String	The tag key. • It contains a maximum of 36 Unicode characters and cannot be an empty string. • The tag key of a resource must be unique.
value	Yes	String	The value. • The value contains a maximum of 43 Unicode characters and can be an empty string.

Response Parameters

None

Example Request

Delete tags from a cluster in batches.

```
POST /v1.1/{project_id}/clusters/{cluster_id}/tags/action

{
```

```
 "action" : "delete",
 "tags" : [ {
   "tag" : null,
   "key" : "DEV1",
   "value" : "DEV1"
 }, {
   "tag" : null,
   "key" : "DEV2",
   "value" : "DEV2"
 } ],
 "x-request-examples-text-1" : "{\n    \"action\":\"create\",\n    \"tags\":[\n        {\n            \"key\":\"DEV1\",
\n            \"value\":\"DEV1\"\n        }.{\n            \"key\":\"DEV2\",\n            \"value\":\"DEV2\"\n        }]\n}"
}
```

Example Response

None

Status Codes

See **Status Codes**.

Error Codes

See **Error Codes**.

6.5.6 Querying All Tags

Function

This API is used to query all tags of a specified region.

URI

- Format
  GET /v1.1/{project_id}/clusters/tags
- Parameter description

Table 6-72 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.

Request Parameters

Request parameters

None

Response Parameters

Table 6-73 Response parameter

Parameter	Type	Description
tags	Array of TagWithMultiValue objects	Tag list. For details, see **Table 6-74**.

Table 6-74 tags parameters

Parameter	Type	Description
key	String	Tag key.
value	Array of strings	Tag value.

Example

- Example request
  ```
  GET https://{endpoint}/v1.1/{project_id}/clusters/tags
  ```
- Example response
  ```
  {
    "tags": [
    {
      "key": "key1",
      "values": [
        "value1",
        "value2"
      ]
      },
      {
      "key": "key2",
      "values": [
        "value1",
        "value2"
      ]
      }
    ]
  }
  ```

Status Codes

See **Status Codes**.

Error Codes

See **Error Codes**.

6.5.7 Querying a List of Clusters with Specified Tags

Function

This API is used to filter clusters by tag.

By default, clusters and tags are sorted in descending order of creation time.

URI

- Format
  POST /v1.1/{project_id}/clusters/resource_instances/action
- Parameter description

Table 6-75 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	The project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.

Request Parameters

Table 6-76 Request parameters

Parameter	Mandatory	Type	Description
tags	No	Array of TagWithMultiValue objects	The return result contains resources corresponding to all tags in this parameter. This parameter contains a maximum of 10 keys, and each key contains a maximum of 10 values. The structure body cannot be missing, and the key cannot be left blank or set to an empty string. For details about the parameters, see **Table 6-77**.
tags_any	No	Array of TagWithMultiValue objects	The return result contains resources corresponding to any tag in this parameter. This parameter contains a maximum of 10 keys, and each key contains a maximum of 10 values. The structure body cannot be missing, and the key cannot be left blank or set to an empty string. Keys must be unique and values of a key must be unique. For details about the parameters, see **Table 6-77**.

Parameter	Mandatory	Type	Description
not_tags	No	Array of TagWithMultiValue objects	The return result does not contain resources corresponding to all tags in this parameter. This parameter contains a maximum of 10 keys, and each key contains a maximum of 10 values. The structure body cannot be missing, and the key cannot be left blank or set to an empty string. Keys must be unique and values of a key must be unique.
not_tags_any	No	Array of TagWithMultiValue objects	The return result does not contain resources corresponding to any tag in this parameter. This parameter contains a maximum of 10 keys, and each key contains a maximum of 10 values. The structure body cannot be missing, and the key cannot be left blank or set to an empty string. Keys must be unique and values of a key must be unique. For details about the parameters, see **Table 6-77**.
limit	No	Integer	Number of records. This parameter is not available when **action** is set to **count**. The default value is **1000** when **action** is set to **filter**. The maximum value is **1000**, and the minimum value is **1**. The value cannot be a negative number.
offset	No	Integer	Index position. The query starts from the next piece of data specified by the **offset** parameter. This parameter is not required when you query data on the first page. The value in the response body returned for querying data on the previous page will be included in this parameter for querying data on subsequent pages. This parameter is not available when **action** is set to **count**. If **action** is set to **filter**, the value must be a number, and the default value is **0**. The value cannot be a negative number.

Parameter	Mandatory	Type	Description
action	Yes	String	Operation to be performed. The value can be **filter** or **count**. The value **filter** indicates pagination query. The value **count** indicates that the total number of query results meeting the search criteria will be returned.
matches	No	Array of Match objects	Search field. **key** indicates the field to be matched, for example, **resource_name**. **value** indicates the matched value. This field is a fixed dictionary value. Determine whether fuzzy match is required based on different fields. For example, if **key** is **resource_name**, fuzzy search is used by default. If **value** is an empty string, exact match is used. For details about the parameters, see **Table 6-78**.

Table 6-77 tag parameters

Parameter	Mandatory	Type	Description
key	Yes	String	Key. A tag key cannot contain special characters (=*<>\,|/) or start or end with spaces.
values	No	Array of strings	List of values. A tag value cannot contain special characters (=*<>\,|/) or start or end with spaces.

Table 6-78 match parameters

Parameter	Mandatory	Type	Description
key	No	String	Key. Currently, only **resource_name** is available, indicating the cluster name. Other keys will be supported later.

Parameter	Mandatory	Type	Description
value	No	String	Value. A value contains a maximum of 64 Unicode characters.

Response

Table 6-79 Response parameters

Parameter	Type	Description
resources	Array of MRSResource objects	Resource details. For details, see **Table 6-80**.
total_count	Integer	Total number of resources.

Table 6-80 resources parameters

Parameter	Type	Description
resource_detail	String	Resource details.
resource_id	String	Resource ID.
resource_name	String	Resource name.
tags	Array of TagPlain objects	Tag list. For details, see **Table 6-81**.

Table 6-81 tags parameters

Parameter	Type	Description
key	String	Key.
value	String	Tag value.

Example

- Example request

  Request body when **action** is set to **filter**

```
{
  "offset": "100",
  "limit": "100",
"action": "filter",
```

```
 "matches":[
{
       "key": "resource_name",
       "value": "clusterA"
      }
],
   "not_tags": [
    {
      "key": "key1",
      "values": [
        "value1",
        "value2"
      ]
    }
  ],
  "tags": [
    {
      "key": "key1",
      "values": [
        "value1",
        "value2"
      ]
    }
  ],
  "tags_any": [
    {
      "key": "key1",
      "values": [
        "value1",
        "value2"
      ]
    }
  ],
"not_tags_any": [
    {
      "key": "key1",
      "values": [
        "value1",
        "value2"
      ]
    }
  ]
}
```

Request body when **action** is set to **count**

```
{
  "action": "count",
  "not_tags": [
    {
      "key": "key1",
      "values": [
        "value1",
        "value2"
      ]
    }
  ],
  "tags": [
    {
      "key": "key1",
      "values": [
        "value1",
        "value2"
      ]
    },
  {
      "key": "key2",
      "values": [
        "value1",
        "value2"
      ]
```

```
    }
  ],
  "tags_any": [
    {
      "key": "key1",
      "values": [
        "value1",
        "value2"
      ]
    }
  ],
"not_tags_any": [
    {
      "key": "key1",
      "values": [
        "value1",
        "value2"
      ]
    }
  ],
"matches":[
{
       "key": "resource_name",
       "value": "clusterA"
      }
]
}
```

- Example response

  Response body when **action** is set to **filter**

```
  {
    "resources": [
      {
         "resource_detail": null,
         "resource_id": "cdfs_cefs_wesas_12_dsad",
         "resource_name": "clusterA"
      }
    ]
    "total_count": 1000
}
```

  Response body when **action** is set to **count**

```
{
     "total_count": 1000
}
```

Status Codes

See **Status Codes**.

Error Codes

See **Error Codes**.

7 Out-of-Date APIs

7.1 Job API Management (Deprecated)

7.1.1 Adding and Executing a Job (Deprecated)

Function

This API is used to add a job to an MRS cluster and execute the job. This API is incompatible with Sahara.

URI

- Format

  POST /v1.1/{project_id}/jobs/submit-job
- Parameter description

  Table 7-1 URI parameter

Parameter	Mandatory	Description
project_id	Yes	The project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.

Request Parameters

Table 7-2 Request parameters

Parameter	Mandatory	Type	Description
job_type	Yes	Integer	Job type code • 1: MapReduce • 2: Spark • 3: Hive Script • 4: HiveQL (not supported currently) • 5: DistCp, importing and exporting data. For details, see **Table 7-3**. • 6: Spark Script • 7: Spark SQL, submitting Spark SQL statements. For details, see **Table 7-4**. (Not supported in this API currently.) **NOTE** Spark and Hive jobs can be added to only clusters that include Spark and Hive components.
job_name	Yes	String	Job name Contains only 1 to 64 letters, digits, hyphens (-), and underscores (_). **NOTE** Identical job names are allowed but not recommended.
cluster_id	Yes	String	Cluster ID

Parameter	Mandatory	Type	Description
jar_path	Yes	String	Path of the JAR or SQL file for program execution The parameter must meet the following requirements: ● Contains a maximum of 1,023 characters, excluding special characters such as ;|&><'$. The address cannot be empty or full of spaces. ● Starts with **/** or **s3a://**. The OBS path does not support files or programs encrypted by KMS. ● Spark Script must end with **.sql** while MapReduce and Spark Jar must end with **.jar**.**sql** and **jar** are case-insensitive.
arguments	No	String	Key parameter for program execution. The parameter is specified by the function of the user's program. MRS is only responsible for loading the parameter. The parameter contains a maximum of 2,047 characters, excluding special characters such as ;|&>'<$, and can be left blank. **NOTE** When entering a parameter containing sensitive information (for example, login password), you can add an at sign (@) before the parameter name to encrypt the parameter value. This prevents the sensitive information from being persisted in plaintext. Therefore, when you view job information on the MRS, sensitive information will be displayed as asterisks (*). For example, username=admin @password=admin_123.

Parameter	Mandatory	Type	Description
input	No	String	Path for inputting data, which must start with **/** or **s3a://**. Set this parameter to a correct OBS path. The OBS path does not support files or programs encrypted by KMS. The parameter contains a maximum of 1,023 characters, excluding special characters such as ;|&>'<$, and can be left blank.
output	No	String	Path for outputting data, which must start with **/** or **s3a://**. A correct OBS path is required. If the path does not exist, the system automatically creates it. The parameter contains a maximum of 1,023 characters, excluding special characters such as ;|&>'<$, and can be left blank.
job_log	No	String	Path for storing job logs that record job running status. The path must start with **/** or **s3a://**. A correct OBS path is required. The parameter contains a maximum of 1,023 characters, excluding special characters such as ;|&>'<$, and can be left blank.
hive_script_path	Yes	String	SQL program path This parameter is needed by Spark Script and Hive Script jobs only, and must meet the following requirements: ● Contains a maximum of 1,023 characters, excluding special characters such as ;|&><'$. The address cannot be empty or full of spaces. ● The path must start with **/** or **s3a://**. The OBS path does not support files or programs encrypted by KMS. ● The path must end with **.sql**.**sql** is case-insensitive.

Table 7-3 DistCp parameters

Parameter	Mandatory	Type	Description
job_name	Yes	String	Job name Contains only 1 to 64 letters, digits, hyphens (-), and underscores (_). **NOTE** Identical job names are allowed but not recommended.
input	No	String	Data source path ● When you import data, the parameter is set to an OBS path. Files or programs encrypted by KMS are not supported. ● When you export data, the parameter is set to an HDFS path.
output	No	String	Data receiving path ● When you import data, the parameter is set to an HDFS path. ● When you export data, the parameter is set to an OBS path.
file_action	Yes	String	Types of file operations, including: ● export: Export data from HDFS to OBS. ● import: Import data from OBS to HDFS.

Table 7-4 Spark SQL parameters

Parameter	Mandatory	Type	Description
hql	Yes	String	Spark SQL statement, which needs Base64 encoding and decoding. **ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789+/** is a standard encoding table. MRS uses **ABCDEFGHILKJMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789+/** for Base64 encoding. The value of the **hql** parameter is generated by adding any letter to the beginning of the encoded character string. The Spark SQL statement is generated by decoding the value in the background. Example: 1. Obtain the Base64 encoding tool. 2. Enter the **show tables;** Spark SQL statement in the encoding tool to perform Base64 encoding. 3. Obtain the encoded character string **c2hvdyB0YWLsZXM7**. 4. At the beginning of **c2hvdyB0YWLsZXM7**, add any letter, for example, **g**. Then, the character string becomes **gc2hvdyB0YWLsZXM7**, that is, the value of the **hql** parameter.
job_name	Yes	String	Job name. It contains 1 to 64 characters. Only letters, digits, hyphens (-), and underscores (_) are allowed. **NOTE** Identical job names are allowed but not recommended.

Response Parameters

Table 7-5 Response parameter

Parameter	Type	Description
job_execution	Object	Job details. For details, see **Table 7-6**.

Table 7-6 job_execution parameters

Parameter	Type	Description
templated	Bool	Whether job execution objects are generated by job templates.
created_at	Integer	Creation time, which is a 10-bit timestamp.
updated_at	Integer	Update time, which is a 10-bit timestamp.
id	String	Job ID
tenant_id	String	Project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.
job_id	String	Job application ID
job_name	String	Job name
input_id	String	Data input ID
output_id	String	Data output ID
start_time	Integer	Start time of job execution, which is a 10-bit timestamp.
end_time	Integer	End time of job execution, which is a 10-bit timestamp.
cluster_id	String	Cluster ID
engine_job_id	String	Workflow ID of Oozie
return_code	Integer	Returned code for an execution result
is_public	Bool	Whether a job is public • true • false The current version does not support this function.

Parameter	Type	Description
is_protected	Bool	Whether a job is protected • true • false The current version does not support this function.
group_id	String	Group ID of a job
jar_path	String	Path of the **.jar** file for program execution
input	String	Address for inputting data
output	String	Address for outputting data
job_log	String	Address for storing job logs
job_type	Integer	Job type code • 1: MapReduce • 2: Spark • 3: Hive Script • 4: HiveQL (not supported currently) • 5: DistCp • 6: Spark Script • 7: Spark SQL (not supported in this API currently)
file_action	String	Data import and export
arguments	String	Key parameter for program execution. The parameter is specified by the function of the user's internal program. MRS is only responsible for loading the parameter. This parameter can be empty.
job_state	Integer	Job status code • -1: Terminated • 1: Starting • 2: Running • 3: Completed • 4: Abnormal • 5: Error

Parameter	Type	Description
job_final_status	Integer	Final job status • 0: unfinished • 1: terminated due to an execution error • 2: executed successfully • 3: canceled
hive_script_path	String	Address of the Hive script
create_by	String	User ID for creating jobs This parameter is not used in the current version, but is retained for compatibility with earlier versions.
finished_step	Integer	Number of completed steps This parameter is not used in the current version, but is retained for compatibility with earlier versions.
job_main_id	String	Main ID of a job This parameter is not used in the current version, but is retained for compatibility with earlier versions.
job_step_id	String	Step ID of a job This parameter is not used in the current version, but is retained for compatibility with earlier versions.
postpone_at	Integer	Delay time, which is a 10-bit timestamp. This parameter is not used in the current version, but is retained for compatibility with earlier versions.
step_name	String	Step name of a job This parameter is not used in the current version, but is retained for compatibility with earlier versions.
step_num	Integer	Number of steps This parameter is not used in the current version, but is retained for compatibility with earlier versions.
task_num	Integer	Number of tasks This parameter is not used in the current version, but is retained for compatibility with earlier versions.

Parameter	Type	Description
update_by	String	User ID for updating jobs
credentials	String	Token The current version does not support this function.
user_id	String	User ID for creating jobs This parameter is not used in the current version, but is retained for compatibility with earlier versions.
job_configs	String	Key-value pair set for saving job running configurations
extra	String	Authentication information The current version does not support this function.
data_source_urls	String	Data source URL
info	String	Key-value pair set, containing job running information returned by Oozie

Example

- Example request

  The following is an example of a MapReduce job request:

```
{
    "job_type": 1,
    "job_name": "mrs_test_jobone_20170602_141106",
    "cluster_id": "e955a7a3-d334-4943-a39a-994976900d56",
    "jar_path": "s3a://mrs-opsadm/jarpath/hadoop-mapreduce-examples-2.7.2.jar",
    "arguments": "wordcount",
    "input": "s3a://mrs-opsadm/input/",
    "output": "s3a://mrs-opsadm/output/",
    "job_log": "s3a://mrs-opsadm/log/",
    "file_action": "",
    "hql": "",
    "hive_script_path": ""
}
```

  Example request of a Spark job

```
{
    "job_type": 2,
    "job_name": "mrs_test_sparkjob_20170602_141106",
    "cluster_id": "e955a7a3-d334-4943-a39a-994976900d56",
    "jar_path": "s3a://mrs-opsadm/jarpath/spark-test.jar",
    "arguments": "org.apache.spark.examples.SparkPi 10",
    "input": "",
    "output": "s3a://mrs-opsadm/output/",
    "job_log": "s3a://mrs-opsadm/log/",
    "file_action": "",
    "hql": "",
    "hive_script_path": ""
}
```

  Example request of a Hive Script job

```
{
    "job_type": 3,
    "job_name": "mrs_test_SparkScriptJob_20170602_141106",
    "cluster_id": "e955a7a3-d334-4943-a39a-994976900d56",
    "jar_path": "s3a://mrs-opsadm/jarpath/Hivescript.sql",
    "arguments": "",
    "input": "s3a://mrs-opsadm/input/",
    "output": "s3a://mrs-opsadm/output/",
    "job_log": "s3a://mrs-opsadm/log/",
    "file_action": "",
    "hql": "",
    "hive_script_path": "s3a://mrs-opsadm/jarpath/Hivescript.sql"
}
```

Example request of a DistCp job for import

```
{
    "job_type": 5,
    "job_name": "mrs_test_importjob_20170602_141106",
    "cluster_id": "e955a7a3-d334-4943-a39a-994976900d56",
    "input": "s3a://mrs-opsadm/jarpath/hadoop-mapreduce-examples-2.7.2.jar",
    "output": "/user",
    "file_action": "import"
}
```

Example request for exporting a DistCp job

```
{
    "job_type": 5,
    "job_name": "mrs_test_exportjob_20170602_141106",
    "cluster_id": "e955a7a3-d334-4943-a39a-994976900d56",
    "input": "/user/hadoop-mapreduce-examples-2.7.2.jar",
    "output": "s3a://mrs-opsadm/jarpath/",
    "file_action": "export"
}
```

Example request of a Spark Script job

```
{
    "job_type": 6,
    "job_name": "mrs_test_sparkscriptjob_20170602_141106",
    "cluster_id": "e955a7a3-d334-4943-a39a-994976900d56",
    "jar_path": "s3a://mrs-opsadm/jarpath/sparkscript.sql",
    "arguments": "",
    "input": "s3a://mrs-opsadm/input/",
    "output": "s3a://mrs-opsadm/output/",
    "job_log": "s3a://mrs-opsadm/log/",
    "file_action": "",
    "hql": "",
    "hive_script_path": "s3a://mrs-opsadm/jarpath/sparkscript.sql"
}
```

- Example response

```
{
  "job_execution": {
    "templated": false,
    "created_at": 1496387588,
    "updated_at": 1496387588,
    "id": "12ee9ae4-6ee1-48c6-bb84-fb0b4f76cf03",
    "tenant_id": "c71ad83a66c5470496c2ed6e982621cc",
    "job_id": "",
    "job_name": "mrs_test_jobone_20170602_141106",
    "input_id": null,
    "output_id": null,
    "start_time": 1496387588,
    "end_time": null,
    "cluster_id": "e955a7a3-d334-4943-a39a-994976900d56",
    "engine_job_id": null,
    "return_code": null,
    "is_public": null,
    "is_protected": false,
    "group_id": "12ee9ae4-6ee1-48c6-bb84-fb0b4f76cf03",
    "jar_path": "s3a://mrs-opsadm/jarpath/hadoop-mapreduce-examples-2.7.2.jar",
```

```
    "input": "s3a://mrs-opsadm/input/",
    "output": "s3a://mrs-opsadm/output/",
    "job_log": "s3a://mrs-opsadm/log/",
    "job_type": 1,
    "file_action": "",
    "arguments": "wordcount",
    "hql": "",
    "job_state": 2,
    "job_final_status": 0,
    "hive_script_path": "",
    "create_by": "b67132be2f054a45b247365647e05af0",
    "finished_step": 0,
    "job_main_id": "",
    "job_step_id": "",
    "postpone_at": 1496387588,
    "step_name": "",
    "step_num": 0,
    "task_num": 0,
    "update_by": "b67132be2f054a45b247365647e05af0",
    "credentials": "",
    "user_id": "b67132be2f054a45b247365647e05af0",
    "job_configs": null,
    "extra": null,
    "data_source_urls": null,
    "info": null
  }
}
```


Status Codes

Table 7-7 describes the status code.

Table 7-7 Status code

Status Code	Description
200	The job has been added.

See **Status Codes**.

7.1.2 Querying the exe Object List of Jobs (Deprecated)

Function

This API is used to query the exe object list of all jobs. This API is incompatible with Sahara.

URI

- Format

  GET /v1.1/{project_id}/job-exes

- Parameter description

Table 7-8 URI parameter

Parameter	Mandatory	Description
project_id	Yes	The project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.

Request Parameters

Table 7-9 Request parameters

Parameter	Mandatory	Type	Description
cluster_id	Yes	String	Cluster ID
id	No	String	Job execution object ID
page_size	No	Integer	Maximum number of jobs displayed on a page Value range: 1 to 100
current_page	No	Integer	Current page number
job_name	No	String	Job name
state	No	Integer	Job status code • -1: Terminated • 2: Running • 3: Completed • 4: Abnormal

Response Parameters

Table 7-10 Response parameters

Parameter	Type	Description
totalRecord	Integer	The total number of jobs in the job list.
job_executions	Array	Job list parameter For details, see **Table 7-11**.

Table 7-11 job_executions parameters

Parameter	Type	Description
id	String	Job ID
create_at	Integer	Creation time, which is a 13-bit timestamp.
update_at	Integer	Update time, which is a 13-bit timestamp.
tenant_id	String	Project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.
job_id	String	Job ID of the YARN
job_name	String	Job name
start_time	Integer	Start time of job execution, which is a 13-bit timestamp.
end_time	Integer	End time of job execution, which is a 13-bit timestamp.
cluster_id	String	Cluster ID of a job
group_id	String	Group ID of a job
jar_path	String	Path of the **.jar** file or **.sql** file for program execution
input	String	Address for inputting data
output	String	Address for outputting data
job_log	String	Address for storing job logs
job_type	Integer	Job type code • 1: MapReduce • 2: Spark • 3: Hive Script • 4: HiveQL (not supported currently) • 5: DistCp • 6: Spark Script • 7: Spark SQL (not supported in this API currently)
file_action	String	Data import and export

Parameter	Type	Description
arguments	String	Key parameter for program execution. The parameter is specified by the function of the user's internal program. MRS is only responsible for loading the parameter. This parameter can be empty.
hql	String	HiveQL statement
job_state	Integer	Job status code ● -1: Terminated ● 2: Running ● 3: Completed ● 4: Abnormal
job_final_status	Integer	Final job status. ● 0: unfinished ● 1: terminated due to an execution error ● 2: executed successfully ● 3: canceled
hive_script_path	String	Address of the Hive script
create_by	String	User ID for creating jobs
finished_step	Integer	Number of completed steps
job_main_id	String	Main ID of a job
job_step_id	String	Step ID of a job
postpone_at	Integer	Delay time, which is a 13-bit timestamp.
step_name	String	Step name of a job
step_num	Integer	Number of steps
task_num	Integer	Number of tasks
update_by	String	User ID for updating jobs
spend_time	Integer	Duration of job execution (unit: s)
step_seq	Integer	Step sequence of a job
progress	String	Job execution progress

Example

- Example request

```
GET/v1.1/{project_id}/job-exes?
page_size=10&current_page=1&state=3&job_name=myfirstjob&clusterId=20ca8601-72a2-4570-b788-2a20fec81a95
```

- Example response

```
{
   "totalRecord": 14,
   "job_executions": [
      {
         "id": "669476bd-89d2-45aa-8f1a-872d16de377e",
         "create_at": 1484641003707,
         "update_at": 1484641003707,
         "tenant_id": "3f99e3319a8943ceb15c584f3325d064",
         "job_id": "",
         "job_name": "myfirstjob",
         "start_time": 1484641003707,
         "end_time": null,
         "cluster_id": "2b460e01-3351-4170-b0a7-57b9dd5ffef3",
         "group_id": "669476bd-89d2-45aa-8f1a-872d16de377e",
         "jar_path": "s3a://jp-test1/program/hadoop-mapreduce-examples-2.4.1.jar",
         "input": "s3a://jp-test1/input/",
         "output": "s3a://jp-test1/output/",
         "job_log": "s3a://jp-test1/joblogs/",
         "job_type": 1,
         "file_action": "",
         "arguments": "wordcount",
         "hql": "",
         "job_state": 2,
         "job_final_status": 1,
         "hive_script_path": null,
         "create_by": "3f99e3319a8943ceb15c584f3325d064",
         "finished_step": 0,
         "job_main_id": "",
         "job_step_id": "",
         "postpone_at": 1484641003174,
         "step_name": "",
         "step_num": 0,
         "task_num": 0,
         "update_by": "3f99e3319a8943ceb15c584f3325d064",
         "spend_time": null,
         "step_seq": 222,
         "progress": "first progress"
      }
   ]
}
```

Status Codes

Table 7-12 describes the status code.

Table 7-12 Status code

Status Code	Description
200	The exe object list of jobs is queried.

See **Status Codes**.

7.1.3 Querying exe Object Details (Deprecated)

Function

This API is used to query detailed information about the exe object of a job. This API is incompatible with Sahara.

URI

- Format

  GET /v1.1/{project_id}/job-exes/{job_exe_id}

- Parameter description

  Table 7-13 URI parameters

Parameter	Mandatory	Description
project_id	Yes	The project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.
job_exe_id	Yes	The job ID.

Request Parameters

Request parameters

None

Response Parameters

Table 7-14 Response parameter

Parameter	Type	Description
job_execution	Object	Job details. For details, see **Table 7-15**.

Table 7-15 job_execution parameters

Parameter	Type	Description
id	String	Job ID
create_at	Integer	Creation time, which is a 13-bit timestamp.
update_at	Integer	Update time, which is a 13-bit timestamp.

Parameter	Type	Description
tenant_id	String	Project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.
job_id	String	Job ID
job_name	String	Job name
start_time	Integer	Start time of job execution, which is a 13-bit timestamp.
end_time	Integer	End time of job execution, which is a 13-bit timestamp.
cluster_id	String	Cluster ID of a job
group_id	String	Group ID of a job
jar_path	String	Path of the **.jar** file or **.sql** file for program execution
input	String	Address for inputting data
output	String	Address for outputting data
job_log	String	Address for storing job logs
job_type	Integer	Job type code • 1: MapReduce • 2: Spark • 3: Hive Script • 4: HiveQL (not supported currently) • 5: DistCp • 6: Spark Script • 7: Spark SQL (not supported in this API currently)
file_action	String	Data import and export
arguments	String	Key parameter for program execution. The parameter is specified by the function of the user's program. MRS is only responsible for loading the parameter. This parameter can be empty.
hql	String	HiveQL statement

Parameter	Type	Description
job_state	Integer	Job status code ● -1: Terminated ● 1: Starting ● 2: Running ● 3: Completed ● 4: Abnormal ● 5: Error
job_final_status	Integer	Final job status ● 0: unfinished ● 1: terminated due to an execution error ● 2: executed successfully ● 3: canceled
hive_script_path	String	Address of the Hive script
create_by	String	User ID for creating jobs
finished_step	Integer	Number of completed steps
job_main_id	String	Main ID of a job
job_step_id	String	Step ID of a job
postpone_at	Integer	Delay time, which is a 13-bit timestamp.
step_name	String	Step name of a job
step_num	Integer	Number of steps
task_num	Integer	Number of tasks
update_by	String	User ID for updating jobs
spend_time	Integer	Duration of job execution (unit: s)
step_seq	Integer	Step sequence of a job
progress	String	Job execution progress

Example

- Example request

  None.

- Example response

```
{
    "job_execution": {
        "id": "632863d5-15d4-4691-9dc1-1a72340cb097",
        "create_at": 1484240559176,
```

```
    "update_at": 1484240559176,
    "tenant_id": "3f99e3319a8943ceb15c584f3325d064",
    "job_id": "632863d5-15d4-4691-9dc1-1a72340cb097",
    "job_name": "hive_script",
    "start_time": 1484240559176,
    "end_time": null,
    "cluster_id": "8b1d55f6-150e-45e2-8347-b2ca608d366b",
    "group_id": "632863d5-15d4-4691-9dc1-1a72340cb097",
    "jar_path": "s3a://jp-test1/program/Hivescript.sql",
    "input": "s3a://jp-test1/input/",
    "output": "s3a://jp-test1/output/",
    "job_log": "s3a://jp-test1/joblogs/",
    "job_type": 3,
    "file_action": "",
    "arguments": "wordcount",
    "hql": null,
    "job_state": 3,
    "job_final_status": 1,
    "hive_script_path": "s3a://jp-test1/program/Hivescript.sql",
    "create_by": "3f99e3319a8943ceb15c584f3325d064",
    "finished_step": 0,
    "job_main_id": "",
    "job_step_id": "",
    "postpone_at": 1484240558705,
    "step_name": "",
    "step_num": 0,
    "task_num": 0,
    "update_by": "3f99e3319a8943ceb15c584f3325d064",
    "spend_time": null,
    "step_seq": 222,
    "progress": "first progress"
  }
}
```


Status Code

Table 7-16 describes the status code of this API.

Table 7-16 Status code

Status code	Description
200	The exe object details are queried successfully.

For the description about error status codes, see **Status Codes**.

7.1.4 Deleting a Job Execution Object (Deprecated)

Function

This API is used to delete a job execution object. This API is compatible with Sahara.

URI

- Format

  DELETE /v1.1/{project_id}/job-executions/{job_execution_id}
- Parameter description

Table 7-17 URI parameters

Parameter	Mandatory	Description
project_id	Yes	The project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.
job_execution_id	Yes	The job ID.

Request Parameters

Request parameters

None

Response Parameters

Response parameters

None

Example

- Example request
  None
- Example response
  None

Status Codes

Table 7-18 describes the status code.

Table 7-18 Status code

Status Code	Description
204	The job execution object is deleted.

See **Status Codes**.

8 Permissions Policies and Supported Actions

8.1 Introduction

Create IAM users and assign permissions to the users so you can perform fine-grained permissions management for your MRS resources. If your account meets service requirements, you do not need to create IAM users. In this case, skip this section.

By default, new IAM users do not have permissions assigned. You need to add them to one or more groups, and attach permissions policies or roles to these groups. Users inherit permissions from the groups to which they are added and can perform specified operations on cloud services based on the permissions.

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

NOTE

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

An account has all the permissions required to call all APIs, but IAM users must be assigned the required permissions. The permissions required for calling an API are determined by the actions supported by the API. Only users who have been granted permissions allowing the actions can call the API successfully. For example, if an IAM user queries MRS clusters using an API, the user must have been granted permissions that allow the **mrs:cluster:list** action.

Supported Actions

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

- Permission: A statement in a policy that allows or denies certain operations.
- API: REST APIs that can be called by a user who has been granted specific permissions.
- Action: Specific operations that are allowed or denied.
- Related actions: Actions on which a specific action depends to take effect. When assigning permissions for the action to a user, you also need to assign permissions for the related actions.
- IAM or enterprise projects: Type of projects for which an action will take effect. Policies that contain actions for both IAM and enterprise projects can be used and take effect for both IAM and Enterprise Management. Policies that only contain actions for IAM projects can be used and only take effect for IAM.

  NOTE

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

Table 8-1 Actions

Permissions	API	Action	IAM Project	Enterprise Project
Creating a Cluster and Executing a Job	POST /v1.1/{project_id}/run-job-flow	mrs:cluster:create	√	√
Resizing a Cluster	PUT /v1.1/{project_id}/cluster_infos/{cluster_id}	mrs:cluster:resize	√	√
Querying a Cluster List	GET /v1.1/{project_id}/cluster_infos	mrs:cluster:list	√	√
Querying Cluster Details	GET /v1.1/{project_id}/cluster_infos/{cluster_id}	mrs:cluster:get	√	√
Deleting a Cluster	DELETE /v1.1/{project_id}/clusters/{cluster_id}	mrs:cluster:delete	√	√
Querying a Host List	GET /v1.1/{project_id}/clusters/{cluster_id}/hosts	mrs:host:list	√	√
Adding and Executing a Job (V1)	POST /v1.1/{project_id}/jobs/submit-job	mrs:job:submit	√	√
Querying the exe Object List of Jobs (V1)	GET /v1.1/{project_id}/job-exes	mrs:job:list	√	√

Permissions	API	Action	IAM Project	Enterprise Project
Querying exe Object Details (V1)	GET /v1.1/{project_id}/job-exes/{job_exe_id}	mrs:job:get	√	√
Configuring an Auto Scaling Rule	POST /v1.1/{project_id}/autoscaling-policy/{cluster_id}	mrs:cluster:policy	√	√
Querying Tags of a Specified Cluster	GET /v1.1/{project_id}/clusters/{cluster_id}/tags	mrs:tag:list	√	√
Adding or Deleting Cluster Tags in Batches	POST /v1.1/{project_id}/clusters/{cluster_id}/tags/action	mrs:tag:batchOperate	√	√
Querying All Tags	GET /v1.1/{project_id}/clusters/tags	mrs:tag:list	√	√
Querying a List of Clusters with Specified Tags	POST /v1.1/{project_id}/clusters/resource_instances/action	mrs:tag:listResource	√	×
Adding and Executing a Job (V2)	POST /v2/{project_id}/clusters/{cluster_id}/job-executions	mrs:job:submit	√	√
Querying Information About a Job (V2)	GET /v2/{project_id}/clusters/{cluster_id}/job-executions/{job_execution_id}	mrs:job:list	√	√
Querying a List of Jobs (V2)	GET /v2/{project_id}/clusters/{cluster_id}/job-executions	mrs:job:list	√	√
Terminating a Job (V2)	POST /v2/{project_id}/clusters/{cluster_id}/job-executions/{job_execution_id}/kill	mrs:job:stop	√	√
Deleting Jobs in Batches (V2)	POST /v2/{project_id}/clusters/{cluster_id}/job-executions/batch-delete	mrs:job:batchDelete	√	√
Obtaining the SQL Result (V2)	GET /v2/{project_id}/clusters/{cluster_id}/job-executions/{job_execution_id}/sql-result	mrs:job:list	√	√

9 Appendix

9.1 Status Codes

Table 9-1 describes status codes.

Table 9-1 Status codes

Status Code	Message	Description
100	Continue	The client should continue with its request. This interim response is used to inform the client that the initial part of the request has been received and has not yet been rejected by the server.
101	Switching Protocols	The protocol should be switched. The protocol can only be switched to a newer protocol. For example, the current HTTPS protocol is switched to a later version.
200	OK	The request has been fulfilled.
201	Created	The request has been fulfilled and a new resource has been created.
202	Accepted	The request has been accepted, but the processing has not been completed.
203	Non-Authoritative Information	The server has successfully processed the request, but is returning information that may be from another source.

Status Code	Message	Description
204	NoContent	The request has been fulfilled, but the HTTPS response does not contain a response body. The status code is returned in response to an HTTPS OPTIONS request.
205	Reset Content	The server has fulfilled the request, but the requester is required to reset the content.
206	Partial Content	The server has successfully processed the partial GET request.
300	Multiple Choices	There are multiple options for the location of the requested resource. The response contains a list of resource characteristics and addresses from which a user terminal (such as a browser) can choose the most appropriate one.
301	Moved Permanently	The requested resource has been assigned a new permanent URI, and the new URI is contained in the response.
302	Found	The requested resource resides temporarily under a different URI.
303	See Other	The response to the request can be found under a different URI. It can be retrieved by using a GET or POST method.
304	Not Modified	The requested resource has not been modified. When the server returns this status code, it does not return any resources.
305	Use Proxy	The requested resource is available only through a proxy.
306	Unused	The HTTPS status code is no longer used.
400	BadRequest	The request is invalid. The client should not repeat the request without modifications.

Status Code	Message	Description
401	Unauthorized	This status code is returned after the client provides the authentication information, indicating that the authentication information is incorrect or invalid.
402	Payment Required	This status code is reserved for future use.
403	Forbidden	The server understood the request, but is refusing to fulfill it. The client should not repeat the request without modifications.
404	NotFound	The requested resource cannot be found. The client should not repeat the request without modifications.
405	MethodNotAllowed	A request method is not supported for the requested resource. The client should not repeat the request without modifications.
406	Not Acceptable	The server cannot fulfill the request according to the content characteristics of the request.
407	Proxy Authentication Required	This status code is similar to 401, but indicates that the client must first authenticate itself with the proxy.
408	Request Time-out	The server has timed out waiting for the request. The client may repeat the request without modifications at a later time.
409	Conflict	The request could not be processed due to a conflict with the current state of the resource. This status code indicates that the resource that the client is attempting to create already exists, or that the request has failed to be processed because of the update of the conflict request.
410	Gone	The requested resource has been deleted permanently and is no longer available.

Status Code	Message	Description
411	Length Required	The server is refusing to process the request without a defined **Content-Length**.
412	Precondition Failed	The server did not meet one of the preconditions that the requester put on the request.
413	Request Entity Too Large	The server is refusing to process a request because the request entity is too large for the server to process. The server may close the connection to prevent the client from continuing the request. If the server is only temporarily unable to process the request, the response will contain a **Retry-After** header field.
414	Request-URI Too Large	The Request-URI is too long for the server to process.
415	Unsupported Media Type	The server is unable to process the media format in the request.
416	Requested range not satisfiable	The requested range is invalid.
417	Expectation Failed	The server has failed to meet the requirements of the **Expect** request-header field.
422	UnprocessableEntity	The request is well-formed but is unable to be processed due to semantic errors.
429	TooManyRequests	The client has sent excessive number of requests to the server within a given time (exceeding the limit on the access frequency of the client), or the server has received an excessive number of requests within a given time (beyond its processing capability). In this case, the client should resend the request after the time specified in the **Retry-After** header of the response has elapsed.
500	InternalServerError	The server is able to receive the request but unable to understand it.
501	Not Implemented	The server does not support the function required to fulfill the request.

Status Code	Message	Description
502	Bad Gateway	The server was acting as a gateway or proxy and received an invalid request from the remote server.
503	ServiceUnavailable	The requested service is invalid. The client should not repeat the request without modifications.
504	ServerTimeout	The request cannot be fulfilled within a given time. This status code is returned to the client only if the **Timeout** parameter is specified in the request.
505	HTTPS Version not supported	The server does not support the HTTPS protocol version used in the request.

9.2 Error Codes

Status Code	Error Code	Error Message	Measure
400	0171	Failed to collect job log details.	-
400	12000002	The parameter is invalid.	-
400	12000003	The cluster does not exist.	-
400	12000009	The parameter is invalid.	-
400	12000012	Tasks are being executed in the cluster. Subcontract period cluster is not allowed for clusters that are in this state. Scale-out or scale-in is not allowed for clusters that are not in the running state.	-

Status Code	Error Code	Error Message	Measure
400	12000013	Failed to scale in the cluster (ID: xxx). The type and the quantity of nodes to be deleted are xxx and xxx, respectively. The task node does not exist.	-
400	12000014	Failed to scale out the cluster (ID: xxx). The type and the quantity of nodes to be added are xxx and xxx, respectively.	-
400	12000018	Scale-out or scale-in cannot be performed again because it is in progress.	-
400	12000019	Failed to obtain hosts of the cluster.	-
400	12000021	Clusters in the xxx state cannot be terminated.	-
400	120000212	Failed to obtain the AZs that you have permission to access.	-
400	12000023	Failed to obtain cluster details.	-
400	12000027	Failed to verify the subnet when creating the cluster xxx.	-
400	12000028	The cluster has a maximum of xxx Core and Task nodes.	-
400	12000029	Failed to obtain the quota.	-

Status Code	Error Code	Error Message	Measure
400	12000030	The requested number of nodes in the cluster exceeds the available quota.	-
400	12000031	The requested number of vCPUs in the cluster exceeds the available quota.	-
400	12000032	The requested memory of the cluster exceeds the available quota.	-
400	12000033	The requested number of disks in the cluster exceeds the available quota.	-
400	12000034	The requested disk capacity of the cluster exceeds the available quota.	-
400	12000036	Failed to obtain product information.	-
400	12000038	Failed to obtain the security group.	-
400	12000041	Failed to obtain the cluster list.	-
400	12000042	Failed to create a cluster.	-
400	12000043	Duplicate cluster name: xxx.	-
400	12000044	The minimum memory of a Master node is xxx GB.	-

Status Code	Error Code	Error Message	Measure
400	12000045	Insufficient quota of the security group.	-
400	12000046	Insufficient quota of the security group rule.	-
400	12000048	Product specification xxx does not exist.	-
400	12000050	Incorrect certificate.	-
400	12000052	No access rights.	-
400	12000053	Invalid billing type.	-
400	12000054	The operation is not supported.	-
400	12000059	Key pair xxx does not exist.	-
400	12000061	Failed to submit the job (cluster ID: xxx; job name: xxx; job ID: xxx).	-
400	12000062	Jobs cannot be submitted to a cluster in the xxx state.	-
400	12000063	Spark jobs cannot be submitted.	-
400	12000064	Jobs cannot be submitted or inquired to a security cluster by API.	-
400	12000080	The status of some nodes is not running in the cluster. Try again later.	-

Status Code	Error Code	Error Message	Measure
400	12000081	The jar_path parameter cannot be left blank if the job type is MapReduce or Spark.	-
400	12000082	Node groups cannot be deleted in the cluster that is being scaled in or scaled out.	-
400	12000085	This interface does not support the cluster of this version.Please use the /v2/{project_id}/clusters/{cluster_id}/job-executions job submit interface.	-
400	12000086	This cluster version not support s3[an]: schema. Please use the obs: schema.	-
400	12000087	Failed to get billing records.	-
400	12000090	Products of the xxx specifications are no longer available in selected AZ.	-
400	12000092	Failed to get metadata of version xxx.	-
400	12000093	Metadata of version xxx not found.	-
400	12000094	The xxx in xxx version does not support the xxx flavor.	-

Status Code	Error Code	Error Message	Measure
400	12000095	Patch xxx is unavailable.	-
400	12000099	Topology template of version xxx not found.	-
400	12000100	Failed to stop the cluster. Only running or abnormal clusters can be stopped.	-
400	12000101	Failed to start the cluster. Only stopped clusters can be started.	-
400	12000104	VPC|ECS|DSS|DCC|OpenStack service error.	-
400	12000105	VPC xxx does not exist.	-
400	12000106	Key pair xxx does not exist.	-
400	12000107	Invalid project ID: xxx.	-
400	12000108	Failed to verify the EIP when creating the cluster xxx.	-
400	12000109	Failed to bind the EIP to cluster xxx.	-
400	12000110	Failed to unbind the EIP from cluster xxx.	-
400	12000111	Failed to bind the EIP. The EIP xxx has been bound to another resource.	-
400	12000112	The EIP xxx does not exist.	-

Status Code	Error Code	Error Message	Measure
400	12000113	Failed to update the EIP.	-
400	12000114	The hive_script_path parameter cannot be left blank if the job type is Hive or SparkScript.	-
400	12000115	The ECS group quota is insufficient.	-
400	12000116	The VPC quota is insufficient. Select the existing VPC or increase the quota.	-
400	12000117	The subnet quota is insufficient. Select the existing subnet or increase the quota.	-
400	12000118	Failed to create the security group rule.	-
400	12000119	The security group rule already exists.	-
400	12000121	Failed to submit a request to scale out the Yearly/Monthly cluster. Scale-out is not allowed because the cluster has an unpaid order. Scale out the cluster again after you pay the order.	-
400	12000122	EPS service error.	-

Status Code	Error Code	Error Message	Measure
400	12000123	Failed to update Task node information because the number of Task nodes is not 0.	-
400	12000124	In the cluster xxx, the number of Task nodes can be adjusted only using auto scaling.	-
400	12000125	Failed to update Task node information, because the cluster state is scaling out or scaling in.	-
400	12000126	Failed to obtain authentication information.	-
400	12000127	Failed to lock cluster operation.	-
400	12000128	Failed to unlock cluster operation.	-
400	12000129	Master node specifications cannot be scaled up for a cluster that is not in the running state.	-
400	12000130	Specifications available for scale-up not found.	-
400	12000131	Master node specifications cannot be scaled up for a non-HA cluster.	-

Status Code	Error Code	Error Message	Measure
400	12000132	vCPUs and memory cannot be reduced in the specification scale-up.	-
400	12000133	Specification scale-up is not available for this type of nodes.	-
400	12000134	Failed to scale up the Master node specifications.	-
400	12000135	Master nodes available for specification scale-up not found	-
400	12000138	Failed to get max server Group members.	-
400	12000139	All evs volume type is sellout, please try again later.	-
400	12000140	Evs volume type:xxx is sellout, please try again later.	-
400	12000141	The disk size of a node cannot be less than {value} GB.	-
400	12000142	The disk size cannot exceed 32,000 GB.	-
400	12000209	Either a VPC ID or name is required.	-
400	12000210	Either a subnet ID or name is required.	-

Status Code	Error Code	Error Message	Measure
400	12000233	Insufficient resources for flavor xxx. Reduce the purchase quantity and try again. Alternatively, select another instance type or flavor, or switch the AZ or region to select your desired product.	-
400	12000234	Insufficient resources for the flavor xxx of the node to be scaled out. Reduce the purchase quantity and try again.	-
400	12000360	Cluster name cannot be updated for a cluster that is in the terminating or terminated state.	-
400	12000361	Only offline cluster is allowed to modify manager access ip.	-
400	12000362	Cluster id is already exist or not universally unique identifier format.	-
400	12000363	The offline cluster with same management node IPs[xxx] already exists.	-
400	12005001	The number of tags in one cluster must not exceed xxx.	-

Status Code	Error Code	Error Message	Measure
400	12005002	Tag xxx does not comply with the specifications. The tag key and value cannot start or end with a space and cannot contain any of the following characters: =*<>\ \,|/.	-
400	13000046	Security group xxx does not exist.	-
400	MRS.1010	The RDS instance whose ID is xxx is not found. The status (xxx) of RDS instance xxx is abnormal.	-
401	12000001	Invalid authentication	-
401	12000136	Permission denied. Error message: Policy doesn't allow bss:order:update to be performed.	-
404	12000057	Failed to obtain the file list.	-
404	12005003	The tag key xxx does not exist in cluster xxx.	-
500	0023	Failed to obtain cluster details.	-
500	0056	Jobs cannot be submitted to a cluster in the xxx state.	-
500	0057	Spark jobs cannot be submitted.	-
500	0093	Metadata of version xxx not found.	-

Status Code	Error Code	Error Message	Measure
500	0160	Failed to kill the job.	-
500	0161	Failed to delete jobs in batches.	-
500	0162	Failed to query the job.	-
500	0165	Failed to verify the SQL statement.	-
500	0166	Failed to query a job list.	-
500	0167	The v2 job API cannot be accessed.	-
500	0168	Hive jobs cannot be submitted.	-
500	0169	Flink jobs cannot be submitted.	-
500	0170	Failed to collect job log directory information.	-
500	0172	Failed to collect the SQL job result.	-
500	0173	Failed to submit the job (cluster ID: xxx; job name: xxx).	-
500	0174	Failed to query the job.	-
500	0175	Failed to kill the job.	-
500	0176	The job does not exist.	-
500	0177	The number of jobs running in each cluster cannot exceed 10.	-
500	0178	The job ID cannot be left blank.	-

Status Code	Error Code	Error Message	Measure
500	0179	The job type must be SparkSql or SparkScript.	-
500	0180	The job is being submitted.	-
500	0181	The SQL job result collection is empty.	-
500	0182	Failed to read the SQL job result.	-
500	0183	The job is running.	-
500	0185	The log type does not exist.	-
500	0187	The log aggregation path is empty. Logs cannot be queried.	-
500	0188	The job query result is empty. Logs cannot be obtained.	-
500	0189	Failed to delete the job list.	-
500	0190	The user who submits the job cannot be empty.	-
500	0191	Failed to query the user who submits the job on MRS Manager.	-
500	0192	The current user does not exist on MRS Manager. Grant the user sufficient permissions on IAM and then perform IAM user synchronization on the Dashboard tab page.	-

Status Code	Error Code	Error Message	Measure
500	0193	Failed to operate the database job records.	-
500	0194	Failed to start the launcher and submit the job.	-
500	0199	Failed to delete the jobs.	-
500	0200	Failed to kill the jobs on Yarn.	-
500	0201	The job does not exist.	-
500	0202	Too many jobs are being submitted, please try again later.	-
500	0211	The maximum number of bound security groups has been reached. A maximum of four security groups can be bound, excluding the default security group that is automatically created.	-
500	12000004	Internal server error.	-
500	12000020	Failed to terminate the cluster.	-
500	120000213	The region [xxx] does not exist.	-
500	120000214	No permission to access the AZs [xxx]. The following AZs [xxx] can be accessed.	-

Status Code	Error Code	Error Message	Measure
500	12000055	Failed to open the file.	-
500	12000060	The number of running jobs per cluster cannot exceed xxx.	-
500	12000068	If the job type is Hive or Spark Script, the value of the mains should not be left blank.	-
500	12000069	If the job type is MapReduce or Spark, the value of the libs should not be left blank.	-
500	12000070	An error occurred while accessing Knox.	-
500	12000071	The Executor server has an internal error.	-
500	12000073	Failed to access the HDFS directory.	-
500	12000102	Failed to stop the cluster.	-
500	12000103	Failed to start the cluster.	-
500	12000154	IAM synchronization is in progress and cannot be triggered again in the same cluster. Cluster ID: xxx	-
500	12000156	Failed to query Iam group	-
500	12000157	Failed to query Iam user or role	-

Status Code	Error Code	Error Message	Measure
500	12000163	Failed to query Manager user.	-
500	12000164	Failed to query Manager user group.	-
500	130000002	The token is invalid.	-
500	MRS.0010	Dataconnector error.	-
500	MRS.0011	SQL typed xxx can not run on the cluster which not installed dependent components. Cannot execute SQL on a cluster in the xxx state. Not support sql execution in cluster version xxx. Request with multiple SQL is not support. Failed to submit the SQL request to Executor (cluster ID: xxx).	-
500	MRS.0205	Failed to sync agency mapping configuration to cluster.	-
500	MRS.0206	Updating agency mapping task is running.	-
500	MRS.0207	Parse Json format failed.	-
500	MRS.0208	Create or modify policy failed.	-
500	MRS.0209	Assign policy to agency failed.	-
500	MRS.0210	No secu_admin policy.	-

Status Code	Error Code	Error Message	Measure
500	MRS.0211	Failed to obtain new agency or new agency does not exist.	-
500	MRS.0212	Updating ECS agency task is running.	-
500	MRS.0216	Failed to update ECS metadata.	-
500	MRS.0217	Failed to unbind policy.	-
500	MRS.0218	Failed to check whether policies are exist or not.	-

9.3 Obtaining a Project ID

Obtaining a Project ID from the Management Console

A project ID (**project_id**) is required for some URLs when an API is called. To obtain a project ID, perform the following operations:

1. Log in to the management console.

If there are multiple projects in one region, expand **Region** and view subproject IDs in the **Project ID** column.

9.4

(**domain-id**) is required for some URLs when an API is called. To obtain on the management console, perform the following steps:

1. Log in to the management console.

9.5 Obtaining the MRS Cluster Information

Components Supported by MRS

- MRS 3.1.0 supports the following components:
  - An analysis cluster contains the following components: Hadoop, Spark2x, HBase, Hive, Hue, Flink, Oozie, ZooKeeper, Ranger, Tez, Impala, Presto, and Kudu.
  - A streaming cluster contains the following components: Kafka, Flume, ZooKeeper, and Ranger.

  - A hybrid cluster contains the following components: Hadoop, Spark2x, HBase, Hive, Hue, Flink, Oozie, ZooKeeper, Ranger, Tez, Impala, Presto, Kudu, Kafka, and Flume.
  - A custom cluster contains the following components: Hadoop, Spark2x, HBase, Hive, Hue, Kafka, Flume, Flink, Oozie, ZooKeeper, Ranger, Tez, Impala, Presto, ClickHouse, and Kudu.
- MRS 3.0.5 supports the following components:
  - An analysis cluster contains the following components: Hadoop, Spark2x, HBase, Hive, Hue, Loader, Flink, Oozie, ZooKeeper, Ranger, Tez, Impala, Presto, Kudu, and Alluxio.
  - A streaming cluster contains the following components: Kafka, Storm, Flume, ZooKeeper, and Ranger.
  - A hybrid cluster contains the following components: Hadoop, Spark2x, HBase, Hive, Hue, Loader, Flink, Oozie, ZooKeeper, Ranger, Tez, Impala, Presto, Kudu, Alluxio, Kafka, Storm, and Flume.
  - A custom cluster contains the following components: Hadoop, Spark2x, HBase, Hive, Hue, Loader, Kafka, Storm, Flume, Flink, Oozie, ZooKeeper, Ranger, Tez, Impala, Presto, ClickHouse, Kudu, and Alluxio.

Obtaining a Cluster ID

A cluster ID (**cluster_id**) is required for some URLs when an API is called. To obtain a cluster ID, perform the following operations:

1. Log in to the MRS management console.
2. Choose **Clusters** > **Active Clusters**, and click the name of the cluster to be operated. The cluster details page is displayed.
3. Click the **Dashboard** tab and obtain the cluster ID in the **Basic Information** area.

Obtaining a Job ID

A job ID (**job_execution_id**) is required for some URLs when an API is called. To obtain a job ID, perform the following operations:

1. Log in to the MRS management console.
2. Choose **Clusters** > **Active Clusters**, and click the name of the cluster to be operated. The cluster details page is displayed.
3. Click the **Jobs** tab and obtain the ID of the job to be operated from the job list.

9.6 Roles and components supported by MRS

Table 9-2 Roles and components supported by MRS

Role Name	Component
OMSServer	OMSServer

Role Name	Component
NameNode	HDFS
Zkfc	HDFS
JournalNode	HDFS
DataNode	HDFS
ResourceManager	Yarn
NodeManager	Yarn
JobHistoryServer	MapReduce
quorumpeer	ZooKeeper
HMaster	HBase
ThriftServer	HBase
RegionServer	HBase
SlapdServer	LdapServer
KerberosServer	KrbServer
KerberosAdmin	KrbServer
Hue	Hue
LoaderServer	Loader
JDBCServer	Spark
JobHistory	Spark
SparkResource	Spark
JDBCServer2x	Spark2x
JobHistory2x	Spark2x
SparkResource2x	Spark2x
MetaStore	Hive
WebHCat	Hive
HiveServer	Hive
MonitorServer	Flume
Flume	Flume
oozie	Oozie
KerberosClient	KrbClient
SlapdClient	LdapClient

Role Name	Component
meta	meta
DBServer	DBService
Broker	Kafka
Supervisor	Storm
Logviewer	Storm
Nimbus	Storm
UI	Storm
FlinkResource	Flink
ClickHouseServer	ClickHouse
ClickHouseBalancer	ClickHouse

9.7 Change History

Release Date	What's New
2022-09-30	This issue is the fifth official release. Added error code reference. For details, see **Error Codes**.
2022-07-15	This issue is the fourth official release. Gradually deprecated some APIs during MRS version evolution. For details, see **Out-of-Date APIs**.
2022-04-30	This issue is the third official release. Added the support for MRS 3.1.0.
2021-03-30	This issue is the second official issue. Added the support for MRS 3.0.5.
2020-02-26	This issue is the first official release.