Cloud Data Migration

API Reference

Issue 02

Date 2022-12-22

HUAWEI TECHNOLOGIES CO., LTD.

Trademarks and Permissions


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

Notice

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

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

Contents

1 Before You Start

1.1 Overview

Welcome to Cloud Data Migration API Reference. Cloud Data Migration (CDM) implements data mobility by enabling batch data migration among homogeneous and heterogeneous data sources. It supports on-premises and cloud file systems, relational databases, data warehouses, NoSQL, big data, and object storage.

You can use the APIs provided in this document to perform operations on CDM, such as creating a cluster and creating a migration job. For details about all supported operations, see **API Overview**.

If you plan to access CDM through an API, ensure that you are familiar with CDM concepts. For details, see **Service Overview**.

If you plan to access CDM through an API, ensure that you are familiar with CDM concepts. For details, see "Introduction" in the *CDM User Guide*.

1.2 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**.

Table 1-1 lists CDM endpoints. Select a desired one based on the service requirements.

Table 1-1 CDM endpoints

Region Name	Region ID	Endpoint	Protocol
CN-Hong Kong	ap-southeast-1	cdm.ap-southeast-1.myhuaweicloud.com	HTTPS
AP-Bangkok	ap-southeast-2	cdm.ap-southeast-2.myhuaweicloud.com	HTTPS

Region Name	Region ID	Endpoint	Protocol
AP-Singapore	ap-southeast-3	cdm.ap-southeast-3.myhuaweicloud.com	HTTPS
LA-Sao Paulo1	sa-brazil-1	cdm.sa-brazil-1.myhuaweicloud.com	HTTPS
LA-Santiago	la-south-2	cdm.la-south-2.myhuaweicloud.com	HTTPS
LA-Mexico City1	na-mexico-1	cdm.na-mexico-1.myhuaweicloud.com	HTTPS

1.3 Project ID and Account ID

Obtaining a Project ID and Account ID

A project is a group of tenant resources, and an account ID corresponds to the current account. The IAM ID corresponds to the current user. You can view the project IDs, account IDs, and user IDs in different regions on the corresponding pages.

1. Register with and log in to the management console.
2. Hover the cursor on the username in the upper right corner and select **My Credentials** from the drop-down list.
3. On the **API Credentials** page, obtain the account name, account ID, IAM username, and IAM user ID, and obtain the project and its ID from the project list.

Obtaining a Project ID by Calling an API

You can obtain the project ID by calling the API to **query project information**. The API is GET https://{Endpoint}/v3/projects, in which {Endpoint} is the IAM endpoint and can be obtained from IAM documentation.

For details about API authentication, see **Authentication**.

The following is a response example. The value of **id** under **projects** is the project ID. If multiple IDs are returned, obtain the desired one based on your actual region (name).

```
{
    "projects": [
        {
            "domain_id": "65382450e8f64ac0870cd180d14e684b",
            "is_domain": false,
            "parent_id": "65382450e8f64ac0870cd180d14e684b",
            "name": "region-name",
            "description": "",
            "links": {
                "next": null,
                "previous": null,
                "self": "https://www.example.com/v3/projects/a4a5d4098fb4474fa22cd05f897d6b99"
            },
```



```
            "id": "a4a5d4098fb4474fa22cd05f897d6b99",
            "enabled": true
        }
    ],
    "links": {
        "next": null,
        "previous": null,
        "self": "https://www.example.com/v3/projects"
    }
}
```


1.4 Concepts

- Account

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

- IAM user

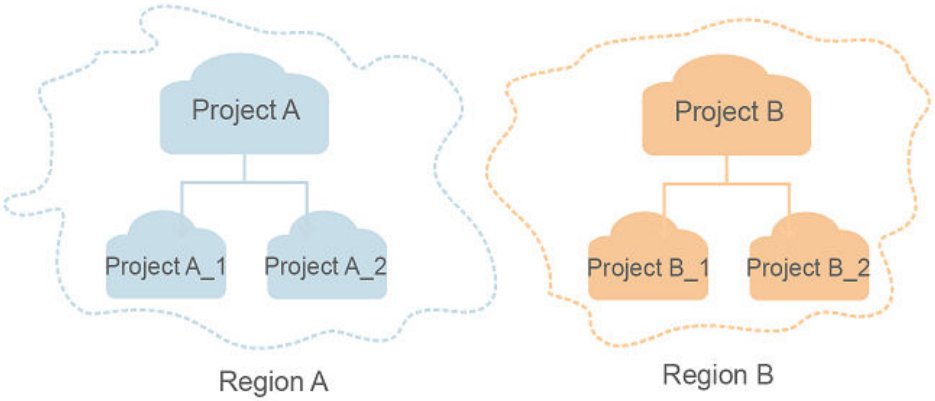
  An IAM user is created using an account to use cloud services. Each IAM user has its own identity credentials (password and access keys).

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

- Project

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

  Figure 1-1 Project isolating model



2 API Overview

CDM provides self-developed APIs. You can perform the following operations with CDM APIs.

Table 2-1 CDM APIs

Type	API	Description	Maximum Number of Requests Made Per User Per Minute
Cluster Management	Creating a Cluster	This API is used to create a cluster.	5
	Querying the Cluster List	This API is used to query and display the cluster list.	120
	Querying Cluster Details	This API is used to query the details of a cluster.	120
	Restarting a Cluster	This API is used to restart a CDM cluster.	20
	Deleting a Cluster	This API is used to delete a specified CDM cluster.	20
	Stopping a Cluster	This API is used to stop a specified CDM cluster.	20
	Querying All AZs	This API is used to query all AZs.	20
	Querying Supported Versions	This API is used to query supported cluster versions.	20

Type	API	Description	Maximum Number of Requests Made Per User Per Minute
	Querying a Flavor	This API is used to query the flavor of a cluster version.	20
	Querying Details About a Flavor	This API is used to query the details of a cluster flavor.	20
	Querying the Enterprise Project IDs of All Clusters	This API is used to query the enterprise project IDs of all clusters.	20
	Querying the Enterprise Project ID of a Specified Cluster	This API is used to query the enterprise project ID of a specified cluster.	20
	Query a Specified Instance in a Cluster	This API is used to query a specified instance in a cluster.	20
	Modifying a Cluster	This API is used to modify a specified CDM cluster.	20
	Starting a Cluster	This API is used to start a specified CDM cluster.	20
Link Management	Creating a Link	This API is used to connect to a specified data source.	120
	Querying a Link	This API is used to query a link.	120
	Modifying a Link	This API is used to modify a link.	120
	Deleting a Link	This API is used to delete a specified link.	120

Type	API	Description	Maximum Number of Requests Made Per User Per Minute
Job Management	Creating a Job in a Specific Cluster	This API is used to create a data migration job in a specific CDM cluster. The job will not automatically start.	1,200
	Creating and Executing a Job in a Random Cluster	This API is used to create and run a data migration job in a random CDM cluster.	120
	Querying the Job List	This API is used to query the job list.	120
	Modifying a Job	This API is used to modify a job.	120
	Starting a Job	This API is used to start a data migration job.	1,200
	Stopping a Job	This API is used to stop a running job.	1,200
	Querying Job Status	This API is used to query the status of a job.	120
	Querying Job Execution History	This API is used to query the execution history of a job.	120
	Deleting a Job	This API is used to delete a job.	120

Precautions

- CDM jobs carry large volumes of data, which increases the database load. You are advised to periodically delete unused jobs.
- If a large number of jobs are delivered in a short period, cluster resources may be exhausted.
- CDM is a batch offline migration tool. You are advised not to create a large number of small jobs using CDM.

3 Calling APIs

3.1 Making an API Request

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

Request URI

A request URI is in the following format:

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

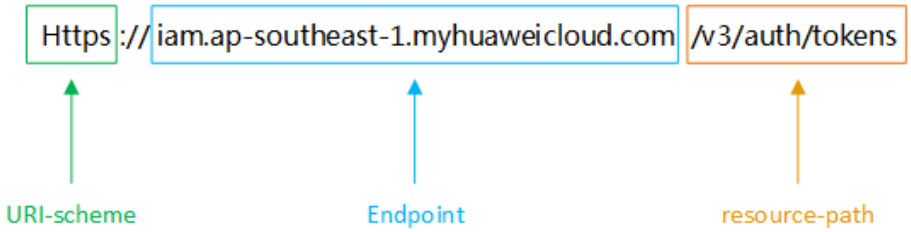
Table 3-1 Request URI

Parameter	Description
URI-scheme	Protocol used to transmit requests. All APIs use HTTPS.
Endpoint	Domain name or IP address of the server bearing the REST service. The endpoint varies between services in different regions. It can be obtained from **Endpoints**. For example, the endpoint of IAM in region **CN-Hong Kong** is **iam.ap-southeast-1.myhuaweicloud.com**. For example, the endpoint of IAM in the **eu-west-0** region is **iam.eu-west-0.prod-cloud-ocb.orange-business.com**.
resource-path	Access path of an API for performing a specified operation. Obtain the value 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 a query parameter that is in the format of "*Parameter name=Parameter value*". For example, **? limit=10** indicates that a maximum of 10 data records will be displayed.

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

```
https://iam.ap-southeast-1.myhuaweicloud.com/v3/auth/tokens
```

Figure 3-1 Example URI



NOTE

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

Request Methods

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

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	Same as GET except that the server must return only the response 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.

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

```
POST https://iam.ap-southeast-1.myhuaweicloud.com/v3/auth/tokens
```


Request Header

You can also add additional 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 describes common request headers.

Table 3-3 Common request headers

Name	Description	Mandatory	Example Value
Content-type	Request body type or format. Its default value is **application/json**.	Yes	application/json
Content-Length	Length of the request body. The unit is byte.	No	3495
X-Project-Id	Project ID. This parameter is used to obtain the token for the project.	No	e9993fc787d94b6c886cbaa340f9c0f4
X-Auth-Token	User token, which is a response to the API used to **obtain a user token**. This API is the only one that does not require authentication.	Yes	-
X-Language	Request language.	Yes	en_us
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) time of the system.	No	20190407T101459Z
Host	Information about the requested server. The value can be obtained from the URL of the service API in *hostname[:port]* format. If the port number is not specified, the default port is used. The default port number for **https** is **443**.	No	code.test.com or code.test.com:443

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


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

Request Body

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

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.

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

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

NOTE

The **scope** parameter specifies where a token takes effect. Its value can be **project** or **domain**. In the example, the value is **project**, indicating that the obtained token takes effect only for the resources in a specified project. If the value is **domain**, the obtained token takes effect for all resources of the specified account.

For more information about this parameter, see **Obtaining a User Token**.

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

3.2 Authentication

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.

NOTE

An IAM user can pass the authentication and access DataArts Studio through an API or SDK only if **Programmatic access** is selected for **Access Type** during the creation of the IAM user.

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.
- Ensure that the token is valid when you use it. Using a token that will soon expire may cause API calling failures.

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

When calling the API to **obtain a user token**, you must set **auth.scope** in the request body to **project**.

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

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, the API used to query a connection list. For example, if the token is **ABCDEFJ....**, **X-Auth-Token: ABCDEFJ....** can be added to a request as follows:

```
GET https://iam.ap-southeast-1.myhuaweicloud.com/v1/{project_id}/connections
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, use token-based authentication.
- You can use the AK/SK in a permanent or temporary access key. The **X-Security-Token** field must be configured if the AK/SK in a temporary access key is used, and the field value is **security_token** of the temporary access key.

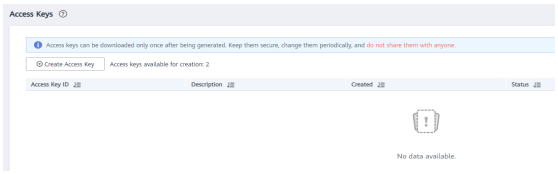
In AK/SK-based authentication, AK/SK is used to sign requests and the signature is then added to the requests for authentication.

- 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.

To obtain an access key, perform the following steps:

1. Log in to the management console, move the cursor to the username in the upper right corner, and select **My Credentials** from the drop-down list.
2. On the **My Credentials** page, choose **Access Keys**, and click **Create Access Key**. See **Figure 3-2**.

   Figure 3-2 Clicking Create Access Key

3. Click **OK** and save the access key file as prompted. The access key file will be saved to your browser's configured download location. Open the **credentials.csv** file to view **Access Key Id** and **Secret Access Key**.

   NOTE

   - Only two access keys can be added for each user.
   - To ensure access key security, the access key is automatically downloaded only when it is generated for the first time and cannot be obtained from the management console later. Keep them properly.

In AK/SK-based authentication, you can use an AK/SK to sign requests based on the signature algorithm or use the signing SDK to sign requests. For details about how to sign requests and use the signature SDK, see **API Request Signing Guide**.

NOTICE

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

3.3 Response

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

Status Code

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 Code**.

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

Response Header

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

Table 3-4 Common response headers

Parameter	Description	Mandatory
Content-Type	Media type of the message body sent to a receiver Type: string Default value: **application/json; charset=UTF-8**	Yes
X-request-id	This field carries the request ID for task tracing. Type: string **request_id-timestamp-hostname** (The **request_id** is the UUID generated on the server. **timestamp** indicates the current timestamp, and **hostname** is the name of the server that processes the current API.) Default value: none	No
X-ratelimit	This field carries the total number of flow control requests. Type: integer Default value: none	No
X-ratelimit-used	This field carries the number of remaining requests. Type: integer Default value: none	No
X-ratelimit-window	This field carries the flow control unit. Type: string The unit is minute, hour, or day. Default value: hour	No

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

Response Body

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

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

```
{
    "token": {
        "expires_at": "2019-02-13T06:52:13.855000Z",
        "methods": [
            "password"
        ],
        "catalog": [
            {
                "endpoints": [
                    {
                        "region_id": "ap-southeast-1",
.....
```

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

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

In the response body, **error_code** is an error code, and **error_msg** provides information about the error. For details, see **Error Code**.

4 Application Example

This section describes how to use cURL to call CDM APIs to migrate data from a local MySQL database to DWS in the cloud.

1. **Obtaining a Token**

   Call the API to obtain the user token, which will be put into the request header for authentication in a subsequent request.

2. **Creating a CDM Cluster**

   - If you have created a CDM cluster, skip this step and directly use the ID of the created cluster.
   - If you want to use a new cluster for migration, call the API in **Creating a Cluster** to create a CDM cluster.

3. **Creating Links**

   Call the API in **Creating a Link** to create the MySQL and DWS links.

4. **Creating a Migration Job**

   Call the API in **Creating a Job in a Specified Cluster** to create a job for migrating data from MySQL to DWS.

5. **Viewing Job Result**

   Call the API in **Starting a Job** to execute the job.

Preparing Data

Before calling an API, prepare the following data.

Table 4-1 Preparing data

Item	Name	Description	Example
Account information	Project name	Name of the project where CDM resides	Project Name
	Project ID	ID of the project where CDM resides	1551c7f6c808414d8e9f3c514a170f2e

Item	Name	Description	Example
	Account name	Name of an enterprise account to which a user belongs	Account Name
	Username	Username for using a cloud service. The user must have operation permissions on CDM.	Username
	Password	User password	password
VPC information	VPC ID	The VPC where CDM resides must be the same as that of DWS.	6b47302a-bf79-4b20-bf7a-80987408e196
	Subnet ID	The subnet where CDM resides must be the same as that of DWS.	63bdc3cb-a4e7-486f-82ee-d9bf208c8f8c
	Security group ID	The security group where CDM resides must be the same as that of DWS.	005af77a-cce5-45ac-99c7-2ea50ea8addf
Endpoint	IAM endpoint	An endpoint is the **request address** for calling an API. Endpoints vary depending on services and regions. You can obtain endpoints from **Regions and Endpoints**.	iam_endpoint
	CDM endpoint	An endpoint is the **request address** for calling an API. Endpoints vary depending on services and regions. You can obtain endpoints of the service from **Endpoints**.	cdm_endpoint
MySQL database	IP address	IP address of the local MySQL database, which allows CDM to access the MySQL database using a public IP address	1xx.120.85.24
	Port	MySQL database port	3306

Item	Name	Description	Example
	Database name	Name of the MySQL database from which data is to be exported	DB_name
	Username	Username for accessing the MySQL database. The user must have the read, write, and delete permissions on the MySQL database.	username
	Password	Password for accessing the MySQL database	DB_password
DWS database	IP address	IP address of the DWS database. CDM can access the IP address through the internal network.	10.120.85.24
	Port	DWS database port	3306
	Database name	Name of the DWS database to which data is written	DWS
	Username	Username for accessing the DWS database. The user must have the read, write, and delete permissions on the DWS database.	user_dws
	Password	Password for accessing the DWS database	dws_password

Obtaining a Token

1. Before calling other APIs, obtain the token and set it as an environment variable.

```
curl -H "Content-Type:application/json" https://{iam_endpoint}/v3/auth/tokens -X POST -d '
{
    "auth": {
        "identity": {
            "methods": [
                "password"
            ],
            "password": {
                "user": {
                    "name": "Username",
```



```
                "password": "password",
                "domain": {
                    "name": "Account Name"
                }
            }
        }
    },
    "scope": {
        "project": {
            "id": "1551c7f6c808414d8e9f3c514a170f2e"
        }
    }
  }
}
' -v -k
```

The value of **X-Subject-Token** in the response header is the token.

```
X-Subject-Token:MIIDkgYJKoZIhvcNAQcCoIIDgzCCA38CAQExDTALBglghkgBZQMEAgEwgXXXXX...
```

2. Run the following command to set the token as an environment variable for future use:

```
export Token = MIIDkgYJKoZIhvcNAQcCoIIDgzCCA38CAQExDTALBglghkgBZQMEAgEwgXXXXX...
```

Creating a CDM Cluster

1. Call the API in **Creating a Cluster** to create a cluster. The following values are examples:
   - Cluster name: **cdm-ab82**
   - Cluster flavor: **cdm.medium**
   - The VPC, subnet, and security group are the same as those of DWS, and the EIP is automatically bound.

   If status code **200** is returned, the cluster is successfully created.

```
curl -X POST -H 'Content-Type:application/json;charset=utf-8' -H "X-Auth-Token:$Token" -d '
{
  "cluster": {
    "name": "cdm-ab82",
    "vpcId": "6b47302a-bf79-4b20-bf7a-80987408e196",
    "instances": [{
          "flavorRef": "fb8fe666-6734-4b11-bc6c-43d11db3c745",
          "nics": [{
              "net-id": "63bdc3cb-a4e7-486f-82ee-d9bf208c8f8c",
              "securityGroupId": "005af77a-cce5-45ac-99c7-2ea50ea8addf"
          }],
          "availability_zone": "Project Name",
          "type": "cdm"
      }],
    "datastore": {
          "version": "1.8.5",
          "type": "cdm"
      },
    "isScheduleBootOff": false,
    "scheduleBootTime": "null",
    "scheduleOffTime": "null",
    "isAutoOff": false,
    "sys_tags": [{
          "key": "_sys_enterprise_project_id",
          "value": "1ce45885-4033-40d2-bdde-d4dbaceb387d"
      }]
    },
  "autoRemind": false,
  "phoneNum": "null",
  "email": "null"
}'
https://{cdm_endpoint}/v1.1/1551c7f6c808414d8e9f3c514a170f2e/clusters -v -k
```


2. Call the API in **Querying the Cluster List** to query cluster information, obtain the cluster ID, and set the cluster ID to a global variable.

```
curl -X GET -H 'Content-Type:application/json;charset=utf-8' -H "X-Auth-Token:$Token" https://
{cdm_endpoint}/v1.1/1551c7f6c808414d8e9f3c514a170f2e/clusters -k -v
```

The response is as follows:

```
{
  "clusters": [{
    "version": "x.x.x",
    "updated": "2018-09-05T08:38:25",
    "name": "cdm-ab82",
    "created": "2018-09-05T08:38:25",
    "id": "bae65496-643e-47ca-84af-948672de7eeb",
    "status": "200",
    "isFrozen": "0",
    "statusDetail": "Normal",
    "actionProgress": {},
    "config_status": "In-Sync"
  }]
}
```

If the value of **status** is **200**, the cluster is successfully created. The cluster ID is **bae65496-643e-47ca-84af-948672de7eeb**.

3. Run the following command to set the cluster ID to a global variable for future use:

```
export ID = bae65496-643e-47ca-84af-948672de7eeb
```

Creating Links

1. Call the API in **Creating a Link** to create the MySQL link **mysql_link**. The following values are examples:
   - IP address: **1***xx***.120.85.24**
   - Port number: **3306**
   - Database name: **DB**_*name*
   - Login username: *username*
   - Password: **DB**_*password*

   If status code **200** is returned, the link is successfully created.

```
curl -X POST -H "Content-Type:application/json" -H "X-Auth-Token:$Token" -d '{
  "links": [{
    "enabled": true,
    "update-user": null,
    "name": "mysql_link",
    "link-config-values": {
      "configs": [
                {
                    "name": "linkConfig",
                    "inputs": [
                        {
                            "name": "linkConfig.databaseType",
                            "value": "MYSQL"
                        },
                        {
                            "name": "linkConfig.host",
                            "value": "1xx.120.85.24"
                        },
                        {
                            "name": "linkConfig.port",
                            "value": "3306"
                        },
                        {
                            "name": "linkConfig.database",
                            "value": "DB_name"
```



```
                    },
                    {
                        "name": "linkConfig.username",
                        "value": "username"
                    },
                    {
                        "name": "linkConfig.password",
                        "value": "DB_password"
                    },
                    {
                        "name": "linkConfig.fetchSize",
                        "value": "100000"
                    },
                    {
                        "name": "linkConfig.usingNative",
                        "value": "true"
                    }
                ]
            }
        ]
  },
  "connector-name": "generic-jdbc-connector",
  "creation-date": 1536654788622,
  "update-date": 1536654788622,
  "creation-user": null
 }]
}'
https://{cdm_endpoint}/v1.1/1551c7f6c808414d8e9f3c514a170f2e/clusters/
bae65496-643e-47ca-84af-948672de7eeb/cdm/link -k -v
```

2. Call the API in **Creating a Link** to create the DWS link **dws_link**. The following values are examples:
   - IP address of the database: **10.120.85.24**
   - Port number: **3306**
   - Database name: **DWS**
   - Login username: **user_dws**
   - Password: **dws_password**

```
curl -X POST -H "Content-Type:application/json" -H "X-Auth-Token:$Token" -d '{
 "links": [{
  "enabled": true,
  "update-user": null,
  "name": "dws_link",
  "link-config-values": {
   "configs": [
            {
                "name": "linkConfig",
                "inputs": [
                    {
                        "name": "linkConfig.databaseType",
                        "value": "DWS"
                    },
                    {
                        "name": "linkConfig.host",
                        "value": "10.120.85.24"
                    },
                    {
                        "name": "linkConfig.port",
                        "value": "3306"
                    },
                    {
                        "name": "linkConfig.database",
                        "value": "DWS"
                    },
                    {
                        "name": "linkConfig.username",
                        "value": "user_dws"
```



```
                    },
                    {
                        "name": "linkConfig.password",
                        "value": "dws_password"
                    },
                    {
                        "name": "linkConfig.fetchSize",
                        "value": "100000"
                    },
                    {
                        "name": "linkConfig.usingNative",
                        "value": "true"
                    }
                ]
            }
        ]
    },
    "connector-name": "generic-jdbc-connector",
    "creation-date": 1536654788622,
    "update-date": 1536654788622,
    "creation-user": null
  }]
}'
https://{cdm_endpoint}/v1.1/1551c7f6c808414d8e9f3c514a170f2e/clusters/
bae65496-643e-47ca-84af-948672de7eeb/cdm/link -k -v
```

Creating a Migration Job

1. After the links are created, call the API in **Creating a Job in a Specified Cluster** to create a migration job. The following is a sample job:
   - The job name is **mysql2dws**.
   - The name of the MySQL database from which data is exported is **default**, and the name of the exported table is **mysql_tbl**. The job is split into multiple tasks by **id** and the tasks are executed concurrently.
   - The name of the database on DWS to which the data is imported is **public**, and the table name is **cdm_all_type**. Do not clear the data in the table before import.
   - If no table in the local MySQL database exists in the database on DWS, CDM automatically creates the table on DWS.
   - The field list loaded to DWS is **id&gid&name**.
   - When the job extracts data, three extractors are concurrently executed.

   If status code **200** is returned, the job is successfully created.

```
curl -X POST -H "Content-Type:application/json" -H "X-Cluster-ID:$ID" -H "X-Auth-Token:$Token" -d '{
  "jobs": [{
    "job_type": "NORMAL_JOB",
    "name": "mysql2dws",
    "from-link-name": "mysql_link",
    "from-connector-name": "generic-jdbc-connector",
    "to-link-name": "dws_link",
    "to-connector-name": "generic-jdbc-connector",
    "from-config-values": {
      "configs": [{
        "name": "fromJobConfig",
        "inputs": [{
          "name": "fromJobConfig.schemaName",
          "value": "default"
        },
        {
          "name": "fromJobConfig.tableName",
          "value": "mysql_tbl"
        },
```

```
      {
        "name": "fromJobConfig.partitionColumn",
        "value": "id"
      }]
    }]
  },
"to-config-values": {
          "configs": [
             {
                "inputs": [
                   {
                      "name": "toJobConfig.schemaName",
                      "value": "public"
                   },
                   {
                      "name": "toJobConfig.tablePreparation",
                      "value": "CREATE_WHEN_NOT_EXIST"
                   },
                   {
                      "name": "toJobConfig.tableName",
                      "value": "cdm_all_type"
                   },
                   {
                      "name": "toJobConfig.columnList",
                      "value": "id&gid&name"
                   },
                   {
                      "name": "toJobConfig.shouldClearTable",
                      "value": "false"
                   }
                ],
                "name": "toJobConfig"
             }
          ]
       },
  "driver-config-values": {
    "configs": [{
      "name": "throttlingConfig",
      "inputs": [{
        "name": "throttlingConfig.numExtractors",
        "value": "3"
      }]
    }]
  }
 }]
}' https://{cdm_endpoint}/v1.1/1551c7f6c808414d8e9f3c514a170f2e/clusters/
bae65496-643e-47ca-84af-948672de7eeb/cdm/job -k -v
```

2. Call the API in **Starting a Job** to execute the job.

```
curl -X GET -H 'Content-Type:application/json;charset=utf-8' -H "X-Cluster-ID:$ID" -H "X-Auth-
Token:$Token" https://{cdm_endpoint}/v1.1/1551c7f6c808414d8e9f3c514a170f2e/clusters/
bae65496-643e-47ca-84af-948672de7eeb/cdm/job/mysql2dws/start -k -v
```

The response is as follows:

```
{
 "submissions": [{
   "progress": 1,
   "job-name": "mysql2dws",
   "status": "BOOTING",
   "creation-date": 1536654788622,
   "creation-user": "cdm"
 }]
}
```

Viewing Job Result

1. Call the API in **Querying Job Status** to query the job status.


```
curl -X GET -H 'Content-Type:application/json;charset=utf-8' -H "X-Cluster-ID:$ID" -H "X-Auth-Token:$Token" https://{cdm_endpoint}/v1.1/1551c7f6c808414d8e9f3c514a170f2e/clusters/6ec9a0a4-76be-4262-8697-e7af1fac7920/cdm/job/mysql2dws/status -k -v
```

2. View the job execution result. The response to successful job execution is as follows:

```
{
  "submissions": [{
    "progress": 0,
    "job-name": "mysql2dws",
    "status": "SUCCEEDED",
    "creation-date": 1536654788622,
    "creation-user": "cdm",
    "isStopingIncrement": "",
    "last-update-date": 1536654888622,
    "is-execute-auto": false,
    "last-update-user": "cdm",
    "isDeleteJob": false,
    "isIncrementing": false,
    "external-id": "job_local1127970451_0009",
    "counters": {
      "org.apache.sqoop.submission.counter.SqoopCounters": {
        "BYTES_WRITTEN": -1,
        "TOTAL_FILES": -1,
        "BYTES_READ": -1,
        "FILES_WRITTEN": -1,
        "TOTAL_SIZE": -1,
        "FILES_READ": -1,
        "ROWS_WRITTEN": 80,
        "ROWS_READ": 80
      }
    }
  }]
}
```

NOTE

- **BYTES_WRITTEN**: number of written bytes
- **BYTES_READ**: number of read bytes
- **TOTAL_FILES**: total number of files
- **FILES_WRITTEN**: number of written files
- **FILES_READ**: number of read files
- **ROWS_WRITTEN**: number of rows that are successfully written
- **ROWS_READ**: number of rows that are successfully read

5 API

5.1 Cluster Management

5.1.1 Querying Cluster Details

Function

This API is used to query cluster details.

Calling Method

For details, see **Calling APIs**.

URI

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

Table 5-1 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain it, see **Project ID and Account ID**.
cluster_id	Yes	String	Cluster ID

Request Parameters

Table 5-2 Request header parameters

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

Response Parameters

Status code: 200

Table 5-3 Response body parameters

Parameter	Type	Description
publicEndpoint	String	EIP bound to the cluster
instances	Array of **ClusterDetailInstance** objects	Cluster node information. For details, see the descriptions of **instances** parameters.
security_group_id	String	Security group ID
subnet_id	String	Subnet ID
vpc_id	String	VPC ID
customerConfig	**CustomerConfig** object	User configuration
datastore	**Datastore** object	CDM information
isAutoOff	Boolean	Automatic shutdown
publicEndpointDomainName	String	Domain name for the EIP bound to the cluster
bakExpectedStartTime	String	Start time
bakKeepDay	Integer	Retention period
maintainWindow	**maintainWindow** object	Maintenance window

Parameter	Type	Description
recentEvent	Integer	Number of events
flavorName	String	Flavor name
azName	String	AZ name
endpointDomainName	String	Peer domain name
publicEndpointStatus	**publicEndpointStatus** object	EIP status
isScheduleBootOff	Boolean	Whether to enable scheduled startup/shutdown. The scheduled startup/shutdown and auto shutdown functions cannot be enabled at the same time.
namespace	String	Namespace
eipId	String	EIP ID
failedReasons	**FailedReasons** object	Failure cause. If this parameter is left empty, the cluster is in normal state.
dbuser	String	Database user
links	Array of **ClusterLinks** objects	Cluster link information
clusterMode	String	Cluster mode: sharding
task	**ClusterTask** object	Task information
created	String	Cluster creation time in ISO 8601 format: YYYY-MM-DDThh:mm:ssZ
statusDetail	String	Cluster status: normal
config_status	String	Cluster configuration status. • **In-Sync**: The configuration has been synchronized. • **Applying**: The cluster is being configured. • **Sync-Failure**: The configuration failed.
actionProgress	**ActionProgress** object	Cluster operation progress, which consists of a key and a value. The key indicates an ongoing task, and the value indicates the progress of the ongoing task. An example is "action_progress":{"SNAPSHOTTING":"16%"}.
name	String	Cluster name

Parameter	Type	Description
id	String	Cluster ID
isFrozen	String	Whether the cluster is frozen. The value can be **0** (not frozen) or **1** (frozen).
actions	Array of strings	Cluster configuration status. Options: - **In-Sync**: The cluster configuration has been synchronized. - **Applying**: The cluster is being configured. - **Sync-Failure**: The cluster configuration failed.
updated	String	Cluster update time in ISO 8601 format: YYYY-MM-DDThh:mm:ssZ
status	String	Cluster status. The options are as follows: • **100**: creating • **200**: normal • **300**: failed • **303**: creation failed • **800**: frozen • **900**: stopped • **910**: stopping • **920**: starting

Table 5-4 ClusterDetailInstance

Parameter	Type	Description
flavor	**flavor** object	VM flavor of a node. For details, see the descriptions of **flavor** parameters.
volume	**volume** object	Disk information of a node. For details, see the descriptions of **volume** parameters.
status	String	Node status • **100**: creating • **200**: normal • **300**: failed • **303**: failed to be created • **400**: deleted • **800**: frozen

Parameter	Type	Description
actions	Array of strings	Node operation status • **REBOOTING**: restarting • **RESTORING**: restoring • **REBOOT_FAILURE**: failed to restart
type	String	Node type. Currently, only **cdm** is available.
id	String	Node VM ID
name	String	Name of the VM on the node
isFrozen	String	Whether the node is frozen. The value can be **0** (not frozen) or **1** (frozen).
components	String	Components
config_status	String	Node configuration status. The value is null when the cluster list is queried. • **In-Sync**: The configuration has been synchronized. • **Applying**: The configuration is in progress. • **Sync-Failure**: The configuration fails.
role	String	Instance role
group	String	Group
links	Array of **ClusterLinks** objects	Links (null is returned for querying the cluster list)
paramsGroupId	String	Group ID
publicIp	String	Public IP address
manageIp	String	Management IP address
trafficIp	String	Traffic IP address
shard_id	String	Slice ID
manage_fix_ip	String	Management fix IP address
private_ip	String	Private IP address
internal_ip	String	Internal IP address
resource	Array of **Resource** objects	Resource information (null is returned for querying the cluster list)

Table 5-5 flavor

Parameter	Type	Description
id	String	VM flavor ID
links	Array of **ClusterLinks** objects	Links

Table 5-6 volume

Parameter	Type	Description
type	String	Type of disks on the node. Only local disks are supported.
size	Long	Size of the disk on the node (GB)

Table 5-7 Resource

Parameter	Type	Description
resource_id	String	Resource ID
resource_type	String	Resource type: server

Table 5-8 CustomerConfig

Parameter	Type	Description
failureRemind	String	Failure notification
clusterName	String	Cluster type
serviceProvider	String	Service provider
localDisk	String	Whether the disk is a local disk
ssl	String	Whether to enable SSL
createFrom	String	Source
resourceId	String	Resource ID
flavorType	String	Flavor type
workSpaceId	String	Workspace ID
trial	String	Trial

Table 5-9 Datastore

Parameter	Type	Description
type	String	Type. Generally, the value is **cdm**.
version	String	Cluster version

Table 5-10 maintainWindow

Parameter	Type	Description
day	String	Day of a week
startTime	String	Start time
endTime	String	End time

Table 5-11 publicEndpointStatus

Parameter	Type	Description
status	String	Status
errorMessage	String	Error message

Table 5-12 FailedReasons

Parameter	Type	Description
CREATE_FAILED	**CREATE_FAILED** object	Cause of the cluster creation failure

Table 5-13 CREATE_FAILED

Parameter	Type	Description
errorCode	String	Error code
errorMsg	String	Failure cause

Table 5-14 ClusterLinks

Parameter	Type	Description
rel	String	Relationship

Parameter	Type	Description
href	String	Link address

Table 5-15 ClusterTask

Parameter	Type	Description
description	String	Task description
id	String	Task ID
name	String	Task name

Table 5-16 ActionProgress

Parameter	Type	Description
CREATING	String	Cluster creation progress, for example, 29%
GROWING	String	Cluster expansion progress, for example, 29%
RESTORING	String	Cluster restoration progress, for example, 29%
SNAPSHOTTING	String	Cluster snapshotting progress, for example, 29%
REPAIRING	String	Cluster repairing progress, for example, 29%

Example Requests

```
GET /v1.1/1551c7f6c808414d8e9f3c514a170f2e/clusters/bae65496-643e-47ca-84af-948672de7eeb
```

Example Responses

Status code: 200

Request succeeded.


```
{
  "publicEndpoint" : "49.xx.xx.10",
  "instances" : [ {
    "flavor" : {
      "id" : "fb8fe666-6734-4b11-bc6c-43d11db3c745"
    },
    "volume" : {
      "size" : "100",
      "type" : "LOCAL_DISK"
    },
    "name" : "cdm-c018",
    "id" : "635dce67-3df8-4756-b4c7-90e45e687367",
    "isFrozen" : "0",
    "type" : "cdm",
    "actions" : "REBOOTING",
    "config_status" : "In-Sync",
```



```
    "status" : "200"
  } ],
  "created" : "2018-09-05T08:38:25",
  "statusDetail" : "Normal",
  "actionProgress" : { },
  "name" : "cdm-c018",
  "id" : "bae65496-643e-47ca-84af-948672de7eeb",
  "isFrozen" : "0",
  "actions" : "REBOOTING",
  "updated" : "2018-09-05T08:38:25",
  "status" : "200"
}
```


Status Codes

Status Code	Description
200	Request succeeded.
400	Request error.
401	Authentication failed.
403	No operation permissions.
404	No resources found.
500	Internal service error. For details about the returned error code, see Error Codes.
503	Service unavailable.

Error Codes

See **Error Codes**.

5.1.2 Deleting a Cluster

Function

This API is used to delete a cluster.

Calling Method

For details, see **Calling APIs**.

URI

DELETE /v1.1/{project_id}/clusters/{cluster_id}

Table 5-17 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain it, see **Project ID and Account ID**.
cluster_id	Yes	String	Cluster ID

Request Parameters

Table 5-18 Request header parameters

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

Table 5-19 Request body parameters

Parameter	Mandatory	Type	Description
keep_last_manual_backup	Yes	Integer	Number of backup log files. Retain the default value **0**.

Response Parameters

Status code: 202

Table 5-20 Response body parameters

Parameter	Type	Description
jobId	String	Job ID

Example Requests

```
DELETE /v1.1/1551c7f6c808414d8e9f3c514a170f2e/clusters/6ec9a0a4-76be-4262-8697-e7af1fac7920

{
  "keep_last_manual_backup" : 0
}
```

Example Responses

Status code: 202

Accepted.

```
{
  "jobId" : "ff8080815e55125a015e552eddba001a"
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cdm.v1.region.CdmRegion;
import com.huaweicloud.sdk.cdm.v1.*;
import com.huaweicloud.sdk.cdm.v1.model.*;


public class DeleteClusterSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CdmClient client = CdmClient.newBuilder()
                .withCredential(auth)
                .withRegion(CdmRegion.valueOf("<YOUR REGION>"))
                .build();
        DeleteClusterRequest request = new DeleteClusterRequest();
        request.withClusterId("{cluster_id}");
        CdmDeleteClusterReq body = new CdmDeleteClusterReq();
        body.withKeepLastManualBackup(0);
        request.withBody(body);
        try {
            DeleteClusterResponse response = client.deleteCluster(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
```



```
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcdm.v1.region.cdm_region import CdmRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcdm.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CdmClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CdmRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DeleteClusterRequest()
        request.cluster_id = "{cluster_id}"
        request.body = CdmDeleteClusterReq(
            keep_last_manual_backup=0
        )
        response = client.delete_cluster(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cdm "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
```



```
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cdm.NewCdmClient(
        cdm.CdmClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.DeleteClusterRequest{}
    request.ClusterId = "{cluster_id}"
    request.Body = &model.CdmDeleteClusterReq{
        KeepLastManualBackup: int32(0),
    }
    response, err := client.DeleteCluster(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in **API Explorer**. SDK sample code can be automatically generated.

Status Codes

Status Code	Description
202	Accepted.
400	Request error.
401	Authorization failed.
403	No operation permissions.
404	No resources found.
500	Internal service error. For details about the returned error code, see Error Codes.
503	Service unavailable.

Error Codes

See **Error Codes**.

5.1.3 Querying All AZs

Function

This API is used to query all AZs of a CDM cluster.

Calling Method

For details, see **Calling APIs**.

URI

GET /v1.1/{project_id}/regions/{region_id}/availability_zones

Table 5-21 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain the project ID, see **Project ID and Account ID**.
region_id	Yes	String	Region ID. You can obtain it from the response message of the "Querying the Region List" API of the IAM service.

Request Parameters

Table 5-22 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token. This parameter is mandatory when **token authentication** is used. You can obtain it from the value of **X-Subject-Token** in the response message header returned by the "Obtaining a User Token" API of the IAM service.

Response Parameters

Status code: 200

Table 5-23 Response body parameters

Parameter	Type	Description
regionId	String	Region ID.
defaultAZ	String	Specifies the default AZ.

Parameter	Type	Description
availableZones	Array of **CdmClusterAvailabilityZone** objects	Indicates the AZ.

Table 5-24 CdmClusterAvailabilityZone

Parameter	Type	Description
availableZoneId	String	ID of the AZ
availableZoneName	String	AZ name
availableZoneCode	String	Indicates the AZ code.
azStatus	String	AZ status.
type	String	Indicates the AZ type.
tags	Object	AZ tag

Example Requests

```
GET /v1.1/1551c7f6c808414d8e9f3c514a170f2e/regions/xxx-xxx-xxx/availability_zones
```

Example Responses

Status code: 200

The request is successful.

```
{
  "regionId" : "xxx-xxx-xxx",
  "defaultAZ" : "xxx-xxx-xxx",
  "availableZones" : [ {
    "availableZoneId" : "xxx-xxx-xxx",
    "availableZoneName" : "xxx-xxx-xxx",
    "availableZoneCode" : "xxx-xxx-xxx",
    "azStatus" : "Available",
    "type" : null,
    "tags" : null
  } ]
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;
```



```
import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cdm.v1.region.CdmRegion;
import com.huaweicloud.sdk.cdm.v1.*;
import com.huaweicloud.sdk.cdm.v1.model.*;


public class ShowAvailabilityZonesSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CdmClient client = CdmClient.newBuilder()
                .withCredential(auth)
                .withRegion(CdmRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowAvailabilityZonesRequest request = new ShowAvailabilityZonesRequest();
        request.withRegionId("{region_id}");
        try {
            ShowAvailabilityZonesResponse response = client.showAvailabilityZones(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcdm.v1.region.cdm_region import CdmRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcdm.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"
```



```
    credentials = BasicCredentials(ak, sk, projectId)

    client = CdmClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CdmRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowAvailabilityZonesRequest()
        request.region_id = "{region_id}"
        response = client.show_availability_zones(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cdm "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cdm.NewCdmClient(
        cdm.CdmClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowAvailabilityZonesRequest{}
    request.RegionId = "{region_id}"
    response, err := client.ShowAvailabilityZones(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in **API Explorer**. SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	The request is successful.
400	Request error.
401	Authentication failed.
403	No operation permission.
404	No resources found.
500	Internal service error. For details about the returned error code, see Error Codes.
503	Service unavailable.

Error Codes

See **Error Codes**.

5.1.4 Querying Supported Versions

Function

This API is used to query the versions supported by CDM clusters.

Calling Method

For details, see **Calling APIs**.

URI

GET /v1.1/{project_id}/datastores

Table 5-25 Path Parameters

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

Request Parameters

Table 5-26 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token. This parameter is mandatory when **token authentication** is used. You can obtain it from the value of **X-Subject-Token** in the response message header returned by the "Obtaining a User Token" API of the IAM service.

Response Parameters

Status code: 200

Table 5-27 Response body parameters

Parameter	Type	Description
datastores	Array of **CdmClusterDatastore** objects	Specifies the database list.

Table 5-28 CdmClusterDatastore

Parameter	Type	Description
id	String	Service ID, which is used to differentiate services.
name	String	Service name
bigclusterEnable	Boolean	Specifies whether large-scale clusters are supported.
defaultVersion	String	Default version.
versions	Array of **CdmClusterVersion** objects	Indicates the version.

Table 5-29 CdmClusterVersion

Parameter	Type	Description
active	String	Version status.
id	String	Version ID.
image	String	Version image.
name	String	Version name.
packages	String	Version package.
datastore	String	Service ID, which is used to differentiate services.
links	Array of **ClusterLinks** objects	URL information.

Table 5-30 ClusterLinks

Parameter	Type	Description
rel	String	Relationship
href	String	Link address

Example Requests

```
GET /v1.1/1551c7f6c808414d8e9f3c514a170f2e/datastores
```

Example Responses

Status code: 200

The request is successful.

```
[ {
  "id" : "736270b9-27c7-4f03-823b-447d8245e1c2",
  "name" : "cdm",
  "bigclusterEnable" : false,
  "defaultVersion" : "2.9.3.300",
  "links" : null,
  "versions" : [ {
    "active" : "1",
    "id" : "e8a8b8cc-63f8-4fb5-8d4a-24c502317b11",
    "image" : null,
    "links" : [ {
      "rel" : "self",
      "href" : "https://10.63.25.93:443/rds/v1.0/datastores/736270b9-27c7-4f03-823b-447d8245e1c2"
    }, {
      "rel" : "bookmark",
      "href" : "https://10.63.25.93:443/datastores/736270b9-27c7-4f03-823b-447d8245e1c2"
    } ],
    "name" : "2.9.3.300",
    "packages" : "cdm",
```



```
   "datastore" : "736270b9-27c7-4f03-823b-447d8245e1c2"
  } ]
} ]
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cdm.v1.region.CdmRegion;
import com.huaweicloud.sdk.cdm.v1.*;
import com.huaweicloud.sdk.cdm.v1.model.*;


public class ShowDatastoresSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CdmClient client = CdmClient.newBuilder()
                .withCredential(auth)
                .withRegion(CdmRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowDatastoresRequest request = new ShowDatastoresRequest();
        try {
            ShowDatastoresResponse response = client.showDatastores(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
```



```
from huaweicloudsdkcdm.v1.region.cdm_region import CdmRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcdm.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CdmClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CdmRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowDatastoresRequest()
        response = client.show_datastores(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cdm "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cdm.NewCdmClient(
        cdm.CdmClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowDatastoresRequest{}
    response, err := client.ShowDatastores(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
```



```
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in **API Explorer**. SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	The request is successful.
400	Request error.
401	Authentication failed.
403	No operation permission.
404	No resources found.
500	Internal service error. For details about the returned error code, see Error Codes.
503	Service unavailable.

Error Codes

See **Error Codes**.

5.1.5 Querying Version Specifications

Function

This API is used to query all compatible specifications by version ID.

Calling Method

For details, see **Calling APIs**.

URI

GET /v1.1/{project_id}/datastores/{datastore_id}/flavors

Table 5-31 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain it, see **Project ID and Account ID**.
datastore_id	Yes	String	Version ID. For details about how to obtain the version ID, see **Versions Supported by CDM**.

Request Parameters

Table 5-32 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token. This parameter is mandatory when **token authentication** is used. You can obtain it from the value of **X-Subject-Token** in the response message header returned by the "Obtaining a User Token" API of the IAM service.

Response Parameters

Status code: 200

Table 5-33 Response body parameters

Parameter	Type	Description
id	String	Service ID, which is used to differentiate services.
dbname	String	Database name. Generally, the value is cdm.
versions	Array of **CdmClusterDatastoreVersion** objects	API version list

Table 5-34 CdmClusterDatastoreVersion

Parameter	Type	Description
id	String	Specifies the version ID.
name	String	Version name.
flavors	Array of **CdmClusterFlavor** objects	Flavor information.

Table 5-35 CdmClusterFlavor

Parameter	Type	Description
cpu	Integer	CPU.
ram	Integer	Indicates the memory.
name	String	Flavor name
region	String	region.
typename	String	Type.
clusterMode	String	multinode cluster
status	String	Flavor status. Options:
str_id	String	Indicates the flavor ID.

Example Requests

```
GET /v1.1/1551c7f6c808414d8e9f3c514a170f2e/datastores/736270b9-27c7-4f03-823b-447d8245e1c2/flavors
```

Example Responses

Status code: 200

The request is successful.

```
{
  "id" : "736270b9-27c7-4f03-823b-447d8245e1c2",
  "dbname" : "cdm",
  "versions" : [ {
    "id" : "e8a8b8cc-63f8-4fb5-8d4a-24c502317b11",
    "name" : "2.9.3.300",
    "flavors" : [ {
      "cpu" : 4,
      "ram" : 8,
      "name" : "cdm.small",
      "region" : "xxx-xxx-xxx",
      "typename" : "cdm",
      "clusterMode" : "sharding",
      "status" : "abandon",
      "str_id" : "a79fd5ae-1833-448a-88e8-3ea2b913e1f6"
```



```
  } ]
 } ]
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cdm.v1.region.CdmRegion;
import com.huaweicloud.sdk.cdm.v1.*;
import com.huaweicloud.sdk.cdm.v1.model.*;


public class ShowFlavorsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CdmClient client = CdmClient.newBuilder()
                .withCredential(auth)
                .withRegion(CdmRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowFlavorsRequest request = new ShowFlavorsRequest();
        request.withDatastoreId("{datastore_id}");
        try {
            ShowFlavorsResponse response = client.showFlavors(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
```



```
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcdm.v1.region.cdm_region import CdmRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcdm.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CdmClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CdmRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowFlavorsRequest()
        request.datastore_id = "{datastore_id}"
        response = client.show_flavors(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cdm "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cdm.NewCdmClient(
        cdm.CdmClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowFlavorsRequest{}
    request.DatastoreId = "{datastore_id}"
```



```
    response, err := client.ShowFlavors(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in **API Explorer**. SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	The request is successful.
400	Request error
401	Authorization failed.
403	No operation permissions.
404	No resources found.
500	Internal service error. For details about the returned error code, see Error Codes.
503	Service unavailable.

Error Codes

See **Error Codes**.

5.1.6 Querying Details About a Flavor

Function

This API is used to query the details of a flavor with a specified ID.

Calling Method

For details, see **Calling APIs**.

URI

GET /v1.1/{project_id}/flavors/{flavor_id}

Table 5-36 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain the project ID, see **Project ID and Account ID**.
flavor_id	Yes	String	Flavor ID. For details about how to obtain it, see **Querying Version Flavors**.

Request Parameters

Table 5-37 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token. This parameter is mandatory when **token authentication** is used. You can obtain it from the value of **X-Subject-Token** in the response message header returned by the "Obtaining a User Token" API of the IAM service.

Response Parameters

Status code: 200

Table 5-38 Response body parameters

Parameter	Type	Description
name	String	Flavor name
str_id	String	Flavor ID
flavor_detail	Array of **flavorAttribute** objects	List of flavor details

Table 5-39 flavorAttribute

Parameter	Type	Description
name	String	Flavor attribute name, such as **mem** and **cpu**
value	String	Flavor attribute value

Example Requests

```
GET /v1.1/1551c7f6c808414d8e9f3c514a170f2e/flavors/a79fd5ae-1833-448a-88e8-3ea2b913e1f6
```

Example Responses

Status code: 200

Request succeeded.

```
{
  "str_id" : "a79fd5ae-1833-448a-88e8-3ea2b913e1f6",
  "name" : "cdm.large",
  "flavor_detail" : [ {
    "name" : "cpu",
    "value" : 8
  }, {
    "name" : "mem",
    "value" : 16
  }, {
    "name" : "volumeType",
    "value" : "SATA"
  }, {
    "name" : "flavor",
    "value" : "s6.2xlarge.2"
  } ]
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cdm.v1.region.CdmRegion;
import com.huaweicloud.sdk.cdm.v1.*;
import com.huaweicloud.sdk.cdm.v1.model.*;


public class ShowFlavorDetailSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
```



```
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CdmClient client = CdmClient.newBuilder()
                .withCredential(auth)
                .withRegion(CdmRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowFlavorDetailRequest request = new ShowFlavorDetailRequest();
        request.withFlavorId("{flavor_id}");
        try {
            ShowFlavorDetailResponse response = client.showFlavorDetail(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcdm.v1.region.cdm_region import CdmRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcdm.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CdmClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CdmRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowFlavorDetailRequest()
        request.flavor_id = "{flavor_id}"
        response = client.show_flavor_detail(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```


Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cdm "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cdm.NewCdmClient(
        cdm.CdmClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowFlavorDetailRequest{}
    request.FlavorId = "{flavor_id}"
    response, err := client.ShowFlavorDetail(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in **API Explorer**. SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Request succeeded.
400	Request error.
401	Authentication failed.
403	No operation permissions.
404	No resources found.

Status Code	Description
500	Internal service error. For details about the returned error code, see Error Codes.
503	Service unavailable.

Error Codes

See **Error Codes**.

5.1.7 Querying the Enterprise Project IDs of All Clusters

Function

This API is used to query the enterprise project IDs of all clusters in the current project.

Calling Method

For details, see **Calling APIs**.

URI

GET /v1.1/{project_id}/enterprise-projects

Table 5-40 Path Parameters

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

Request Parameters

Table 5-41 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token. This parameter is mandatory when **token authentication** is used. You can obtain it from the value of **X-Subject-Token** in the response message header returned by the "Obtaining a User Token" API of the IAM service.

Response Parameters

Status code: 200

Table 5-42 Response body parameters

Parameter	Type	Description
resources	Array of **CdmClusterEnterpriseProject** objects	Enterprise projects of the cluster

Table 5-43 CdmClusterEnterpriseProject

Parameter	Type	Description
cluster_id	String	Cluster ID
sys_tags	Array of **sys_tags** objects	Enterprise projects

Table 5-44 sys_tags

Parameter	Type	Description
value	String	Enterprise project ID
key	String	The value is fixed at **_sys_enterprise_project_id**.

Example Requests

```
GET /v1.1/1551c7f6c808414d8e9f3c514a170f2e/enterprise-projects
```

Example Responses

Status code: 200

Request succeeded.

```
{
  "resources" : [ {
    "cluster_id" : "2b2a676f-0b91-4ad5-8c24-ec61be586fd1",
    "sys_tags" : [ {
      "key" : "_sys_enterprise_project_id",
      "value" : "1ce45885-4033-40d2-bdde-d4dbaceb387d"
    } ]
  }, {
    "cluster_id" : "387f8b24-c4b0-4211-97fe-745d0e88dc88",
    "sys_tags" : [ {
      "key" : "_sys_enterprise_project_id",
      "value" : 0
    } ]
  } ]
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cdm.v1.region.CdmRegion;
import com.huaweicloud.sdk.cdm.v1.*;
import com.huaweicloud.sdk.cdm.v1.model.*;


public class ShowEnterpriseProjectsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CdmClient client = CdmClient.newBuilder()
                .withCredential(auth)
                .withRegion(CdmRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowEnterpriseProjectsRequest request = new ShowEnterpriseProjectsRequest();
        try {
```

```
            ShowEnterpriseProjectsResponse response = client.showEnterpriseProjects(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcdm.v1.region.cdm_region import CdmRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcdm.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CdmClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CdmRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowEnterpriseProjectsRequest()
        response = client.show_enterprise_projects(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cdm "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
```



```
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cdm.NewCdmClient(
        cdm.CdmClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowEnterpriseProjectsRequest{}
    response, err := client.ShowEnterpriseProjects(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in **API Explorer**. SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Request succeeded.
400	Request error.
401	Authentication failed.
403	No operation permissions.
404	No resources found.
500	Internal service error. For details about the returned error code, see Error Codes.
503	Service unavailable.

Error Codes

See **Error Codes**.

5.1.8 Querying the Enterprise Project ID of a Specified Cluster

Function

This API is used to query the enterprise project ID of a specified cluster.

Calling Method

For details, see **Calling APIs**.

URI

GET /v1.1/{project_id}/clusters/{cluster_id}/enterprise-projects

Table 5-45 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain the project ID, see **Project ID and Account ID**.
cluster_id	Yes	String	CDM cluster ID. For details about how to obtain it, see **Querying the Cluster List**.

Request Parameters

Table 5-46 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token. This parameter is mandatory when **token authentication** is used. You can obtain it from the value of **X-Subject-Token** in the response message header returned by the "Obtaining a User Token" API of the IAM service.

Response Parameters

Status code: 200

Table 5-47 Response body parameters

Parameter	Type	Description
sys_tags	Array of **sys_tags** objects	Enterprise projects

Table 5-48 sys_tags

Parameter	Type	Description
value	String	Enterprise project ID
key	String	The value is fixed at **_sys_enterprise_project_id**.

Example Requests

```
GET /v1.1/1551c7f6c808414d8e9f3c514a170f2e/clusters/2b2a676f-0b91-4ad5-8c24-ec61be586fd1/
enterprise-projects
```

Example Responses

Status code: 200

Request succeeded.

```
{
  "sys_tags" : [ {
    "key" : "_sys_enterprise_project_id",
    "value" : "1ce45885-4033-40d2-bdde-d4dbaceb387d"
  } ]
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cdm.v1.region.CdmRegion;
import com.huaweicloud.sdk.cdm.v1.*;
import com.huaweicloud.sdk.cdm.v1.model.*;


public class ShowClusterEnterpriseProjectsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
```



```
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CdmClient client = CdmClient.newBuilder()
                .withCredential(auth)
                .withRegion(CdmRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowClusterEnterpriseProjectsRequest request = new ShowClusterEnterpriseProjectsRequest();
        request.withClusterId("{cluster_id}");
        try {
            ShowClusterEnterpriseProjectsResponse response = client.showClusterEnterpriseProjects(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcdm.v1.region.cdm_region import CdmRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcdm.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CdmClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CdmRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowClusterEnterpriseProjectsRequest()
        request.cluster_id = "{cluster_id}"
        response = client.show_cluster_enterprise_projects(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
```



```
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cdm "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cdm.NewCdmClient(
        cdm.CdmClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowClusterEnterpriseProjectsRequest{}
    request.ClusterId = "{cluster_id}"
    response, err := client.ShowClusterEnterpriseProjects(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in **API Explorer**. SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Request succeeded.
400	Request error.
401	Authentication failed.

Status Code	Description
403	No operation permissions.
404	No resources found.
500	Internal service error. For details about the returned error code, see Error Codes.
503	Service unavailable.

Error Codes

See **Error Codes**.

5.1.9 Query a Specified Instance in a Cluster

Function

This API is used to query a specified instance in a cluster.

Calling Method

For details, see **Calling APIs**.

URI

GET /v1.1/{project_id}/instances/{instance_id}

Table 5-49 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain the project ID, see **Project ID and Account ID**.
instance_id	Yes	String	Instance ID. For details about how to obtain it, see **Querying the Cluster List**.

Request Parameters

Table 5-50 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token. This parameter is mandatory when **token authentication** is used. You can obtain it from the value of **X-Subject-Token** in the response message header returned by the "Obtaining a User Token" API of the IAM service.

Response Parameters

Status code: 200

Table 5-51 Response body parameters

Parameter	Type	Description
instance	**CdmQueryClusterInstanceDetail** object	Instance information

Table 5-52 CdmQueryClusterInstanceDetail

Parameter	Type	Description
configurationStatus	String	Cluster configuration status. The options are as follows: • **In-Sync**: The configuration has been synchronized. • **Applying**: The cluster is being configured. • **Sync-Failure**: The configuration failed.
paramsGroupId	String	Configuration ID
type	String	Service type, which is **cdm**
role	String	Instance mode, which is **Standalone**
subnetid	String	Subnet ID of the instance
securegroup	String	Security group ID

Parameter	Type	Description
vpc	String	VPC ID of the instance
azcode	String	AZ name
region	String	Site name
created	String	Instance creation time in ISO 8601 format: YYYY-MM-DDThh:mm:ssZ
updated	String	Instance update time in ISO 8601 format: YYYY-MM-DDThh:mm:ssZ
name	String	Instance name
id	String	Instance ID
flavor	**flavor** object	VM flavor of a node. For details, see the descriptions of **flavor** parameters.
datastore	**Datastore** object	Cluster information. For details, see the descriptions of **datastore** parameters.
dbuser	String	Database user, which is **cdm**
payModel	Integer	Payment mode • **0**: pay-per-use • **1**: yearly/monthly
publicIp	String	Public IP address bound to the cluster
trafficIp	String	Private IP address of the cluster
trafficIpv6	String	Private IPv6 address of the cluster
cluster_id	String	Cluster ID

Table 5-53 flavor

Parameter	Type	Description
id	String	VM flavor ID
links	Array of **ClusterLinks** objects	Links

Table 5-54 ClusterLinks

Parameter	Type	Description
rel	String	Relationship

Parameter	Type	Description
href	String	Link address

Table 5-55 Datastore

Parameter	Type	Description
type	String	Type. Generally, the value is **cdm**.
version	String	Cluster version

Example Requests

```
GET /v1.1/1551c7f6c808414d8e9f3c514a170f2e/instances/2c529048-ed06-4bcb-a48e-bf1800e1f496
```

Example Responses

Status code: 200

Request succeeded.

```
{
  "instance" : {
    "configurationStatus" : "In-Sync",
    "paramsGroupId" : "26084bb9-e74b-47d5-8be6-c0fbee9449d5",
    "type" : "cdm",
    "subnetid" : "9e4049b5-19a6-48fe-b5a2-2857c842fe56",
    "securegroup" : "560a3642-ddb1-4e93-a4bb-e484ae975127",
    "vpc" : "f35aee01-c4a3-47c1-8d92-9df430537de4",
    "azcode" : "xxx-xxx-xxxa",
    "region" : "xxx-xxx-xxx",
    "created" : "2018-09-05T08:38:25",
    "updated" : "2018-09-05T08:38:25",
    "name" : "test-cdm-dn-1-1",
    "id" : "2c529048-ed06-4bcb-a48e-bf1800e1f496",
    "flavor" : {
      "id" : "a79fd5ae-1833-448a-88e8-3ea2b913e1f6",
      "links" : [ ]
    },
    "datastore" : {
      "type" : "cdm",
      "version" : "2.9.3.300"
    },
    "dbuser" : "cdm",
    "payModel" : 0,
    "publicIp" : "49.xx.xx.10",
    "trafficIp" : "192.168.0.128",
    "trafficIpv6" : null,
    "cluster_id" : "2d9ac57e-3ebf-4557-86d5-89ae750ff61c"
  }
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cdm.v1.region.CdmRegion;
import com.huaweicloud.sdk.cdm.v1.*;
import com.huaweicloud.sdk.cdm.v1.model.*;


public class ShowInstanceDetailSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CdmClient client = CdmClient.newBuilder()
                .withCredential(auth)
                .withRegion(CdmRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowInstanceDetailRequest request = new ShowInstanceDetailRequest();
        request.withInstanceId("{instance_id}");
        try {
            ShowInstanceDetailResponse response = client.showInstanceDetail(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcdm.v1.region.cdm_region import CdmRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcdm.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
```



```
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CdmClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CdmRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowInstanceDetailRequest()
        request.instance_id = "{instance_id}"
        response = client.show_instance_detail(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cdm "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cdm.NewCdmClient(
        cdm.CdmClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowInstanceDetailRequest{}
    request.InstanceId = "{instance_id}"
    response, err := client.ShowInstanceDetail(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```


More

For SDK sample code of more programming languages, see the Sample Code tab in **API Explorer**. SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Request succeeded.
400	Request error.
401	Authentication failed.
403	No operation permissions.
404	No resources found.
500	Internal service error. For details about the returned error code, see Error Codes.
503	Service unavailable.

Error Codes

See **Error Codes**.

5.1.10 Modifying a Cluster

Function

Modify the CDM cluster configuration.

Calling Method

For details, see **Calling APIs**.

URI

POST /v1.1/{project_id}/cluster/modify/{cluster_id}

Table 5-56 Path Parameters

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

Parameter	Mandatory	Type	Description
cluster_id	Yes	String	CDM cluster ID. For details about how to obtain the CDM cluster ID, see **Querying the Cluster List**.

Request Parameters

Table 5-57 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Type (format) of the message body. This parameter is mandatory if the message body exists. If the message body does not exist, leave this parameter blank. If the request body contains Chinese characters, use charset=utf8 to specify the Chinese character set, for example, application/json;charset=utf8.
X-Auth-Token	Yes	String	User token. This parameter is mandatory when **token authentication** is used. You can obtain it from the value of **X-Subject-Token** in the response message header returned by the "Obtaining a User Token" API of the IAM service.
X-Language	Yes	String	Request language.

Table 5-58 Request body parameters

Parameter	Mandatory	Type	Description
autoOff	No	Boolean	The phone automatically shuts down.
scheduleBootOff	No	Boolean	Scheduled shutdown.
scheduleBootTime	No	String	Scheduled startup.

Parameter	Mandatory	Type	Description
scheduleOffTime	No	String	Scheduled shutdown time.
autoRemind	No	Boolean	Message notification.
phoneNum	No	String	Mobile number. A maximum of 20 mobile numbers can be entered and separated by commas (,).
email	No	String	Email address. A maximum of 20 email addresses can be entered and separated by commas (,).

Response Parameters

None

Example Requests

Modify the cluster configuration.

```
POST /v1.1/1551c7f6c808414d8e9f3c514a170f2e/cluster/modify/bae65496-643e-47ca-84af-948672de7eeb

{
  "autoOff" : false,
  "scheduleBootOff" : true,
  "scheduleBootTime" : "00:00:00",
  "scheduleOffTime" : "10:00:00",
  "autoRemind" : true,
  "phoneNum" : "xxx",
  "email" : "xxx@xxx.com"
}
```

Example Responses

None

SDK Sample Code

The SDK sample code is as follows.

Java

Modify the cluster configuration.

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cdm.v1.region.CdmRegion;
```



```
import com.huaweicloud.sdk.cdm.v1.*;
import com.huaweicloud.sdk.cdm.v1.model.*;


public class ModifyClusterSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CdmClient client = CdmClient.newBuilder()
                .withCredential(auth)
                .withRegion(CdmRegion.valueOf("<YOUR REGION>"))
                .build();
        ModifyClusterRequest request = new ModifyClusterRequest();
        request.withClusterId("{cluster_id}");
        CdmModifyClusterReq body = new CdmModifyClusterReq();
        body.withEmail("xxx@xxx.com");
        body.withPhoneNum("xxx");
        body.withAutoRemind(true);
        body.withScheduleOffTime("10:00:00");
        body.withScheduleBootTime("00:00:00");
        body.withScheduleBootOff(true);
        body.withAutoOff(false);
        request.withBody(body);
        try {
            ModifyClusterResponse response = client.modifyCluster(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

Modify the cluster configuration.

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcdm.v1.region.cdm_region import CdmRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcdm.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
```



```
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CdmClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CdmRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ModifyClusterRequest()
        request.cluster_id = "{cluster_id}"
        request.body = CdmModifyClusterReq(
            email="xxx@xxx.com",
            phone_num="xxx",
            auto_remind=True,
            schedule_off_time="10:00:00",
            schedule_boot_time="00:00:00",
            schedule_boot_off=True,
            auto_off=False
        )
        response = client.modify_cluster(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

Modify the cluster configuration.

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cdm "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cdm.NewCdmClient(
        cdm.CdmClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())
```



```
    request := &model.ModifyClusterRequest{}
    request.ClusterId = "{cluster_id}"
    emailCdmModifyClusterReq:= "xxx@xxx.com"
    phoneNumCdmModifyClusterReq:= "xxx"
    autoRemindCdmModifyClusterReq:= true
    scheduleOffTimeCdmModifyClusterReq:= "10:00:00"
    scheduleBootTimeCdmModifyClusterReq:= "00:00:00"
    scheduleBootOffCdmModifyClusterReq:= true
    autoOffCdmModifyClusterReq:= false
    request.Body = &model.CdmModifyClusterReq{
        Email: &emailCdmModifyClusterReq,
        PhoneNum: &phoneNumCdmModifyClusterReq,
        AutoRemind: &autoRemindCdmModifyClusterReq,
        ScheduleOffTime: &scheduleOffTimeCdmModifyClusterReq,
        ScheduleBootTime: &scheduleBootTimeCdmModifyClusterReq,
        ScheduleBootOff: &scheduleBootOffCdmModifyClusterReq,
        AutoOff: &autoOffCdmModifyClusterReq,
    }
    response, err := client.ModifyCluster(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in **API Explorer**. SDK sample code can be automatically generated.

Status Codes

Status Code	Description
202	The request is successful.
400	Request error.
401	Authentication failed.
403	No operation permission.
404	No resources found.
500	The internal API of the service is abnormal.
503	Service unavailable.

Error Codes

See **Error Codes**.

5.1.11 Restarting a Cluster

Function

This API is used to restart a cluster.

Calling Method

For details, see **Calling APIs**.

URI

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

Table 5-59 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain it, see **Project ID and Account ID**.
cluster_id	Yes	String	Cluster ID

Request Parameters

Table 5-60 Request header parameters

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

Table 5-61 Request body parameters

Parameter	Mandatory	Type	Description
restart	Yes	**restart** object	Cluster restart. For details about how to define the cluster to restart, see the descriptions of **restart** parameters.

Table 5-62 restart

Parameter	Mandatory	Type	Description
restartDelayTime	No	Integer	Restart delay, in seconds

Parameter	Mandatory	Type	Description
restartMode	No	String	Restart mode. • **IMMEDIATELY**: immediate restart • **FORCELY**: forcible restart • **SOFTLY**: normal restart The default value is **IMMEDIATELY**. Forcibly restarting service processes will interrupt services and restart VMs in the cluster.
restartLevel	No	String	Restart level. The options are as follows: • **SERVICE**: service restart • **VM**: VM restart The default value is **SERVICE**.
type	Yes	String	Type of the cluster node. The value can only be **cdm**.
instance	No	String	This parameter is reserved. When **restartLevel** is set to **SERVICE**, this parameter is mandatory. Set it to an empty string.
group	No	String	This parameter is reserved. When **restartLevel** is set to **SERVICE**, this parameter is mandatory. Set it to an empty string.

Response Parameters

Status code: 200

Table 5-63 Response body parameters

Parameter	Type	Description
jobId	String	Job ID

Example Requests

Restart a cluster.

```
POST /v1.1/1551c7f6c808414d8e9f3c514a170f2e/clusters/bae65496-643e-47ca-84af-948672de7eeb/action
```



```
{
  "restart" : {
    "instance" : "",
    "type" : "cdm",
    "group" : ""
  }
}
```

Example Responses

Status code: 200

Request succeeded.

```
{
  "jobId" : "ff8080815e59d92d015e5b27ccb0004d"
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Restart a cluster.

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cdm.v1.region.CdmRegion;
import com.huaweicloud.sdk.cdm.v1.*;
import com.huaweicloud.sdk.cdm.v1.model.*;


public class RestartClusterSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CdmClient client = CdmClient.newBuilder()
                .withCredential(auth)
                .withRegion(CdmRegion.valueOf("<YOUR REGION>"))
                .build();
        RestartClusterRequest request = new RestartClusterRequest();
        request.withClusterId("{cluster_id}");
        CdmRestartClusterReq body = new CdmRestartClusterReq();
        CdmRestartClusterReqRestart restartbody = new CdmRestartClusterReqRestart();
        restartbody.withType("cdm")
            .withInstance("")
            .withGroup("");
```



```
        body.withRestart(restartbody);
        request.withBody(body);
        try {
            RestartClusterResponse response = client.restartCluster(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

Restart a cluster.

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcdm.v1.region.cdm_region import CdmRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcdm.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CdmClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CdmRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RestartClusterRequest()
        request.cluster_id = "{cluster_id}"
        restartbody = CdmRestartClusterReqRestart(
            type="cdm",
            instance="",
            group=""
        )
        request.body = CdmRestartClusterReq(
            restart=restartbody
        )
        response = client.restart_cluster(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```


Go

Restart a cluster.

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cdm "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cdm.NewCdmClient(
        cdm.CdmClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RestartClusterRequest{}
    request.ClusterId = "{cluster_id}"
    instanceRestart:= ""
    groupRestart:= ""
    restartbody := &model.CdmRestartClusterReqRestart{
        Type: "cdm",
        Instance: &instanceRestart,
        Group: &groupRestart,
    }
    request.Body = &model.CdmRestartClusterReq{
        Restart: restartbody,
    }
    response, err := client.RestartCluster(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in **API Explorer**. SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Request succeeded.
400	Request error.
401	Authentication failed.
403	No operation permissions.
404	No resources found.
500	Internal service error. For details about the returned error code, see Error Codes.
503	Service unavailable.

Error Codes

See **Error Codes**.

5.1.12 Starting a Cluster

Function

This API is used to start a cluster.

Calling Method

For details, see **Calling APIs**.

URI

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

Table 5-64 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain it, see **Project ID and Account ID**.
cluster_id	Yes	String	Cluster ID

Request Parameters

Table 5-65 Request header parameters

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

Table 5-66 Request body parameters

Parameter	Mandatory	Type	Description
start	Yes	Object	Starting a cluster. This parameter is an empty object.

Response Parameters

Status code: 200

Table 5-67 Response body parameters

Parameter	Type	Description
jobId	Array of strings	Job ID

Example Requests

Start a cluster.

```
POST /v1.1/1551c7f6c808414d8e9f3c514a170f2e/clusters/bae65496-643e-47ca-84af-948672de7eeb/action

{
  "start" : { }
}
```

Example Responses

Status code: 200

Request succeeded.

```
{
  "jobId" : [ "ff8080815e59d92d015e5b27ccb0004d" ]
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Start a cluster.

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cdm.v1.region.CdmRegion;
import com.huaweicloud.sdk.cdm.v1.*;
import com.huaweicloud.sdk.cdm.v1.model.*;


public class StartClusterSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CdmClient client = CdmClient.newBuilder()
                .withCredential(auth)
                .withRegion(CdmRegion.valueOf("<YOUR REGION>"))
                .build();
        StartClusterRequest request = new StartClusterRequest();
        request.withClusterId("{cluster_id}");
        CdmStartClusterReq body = new CdmStartClusterReq();
        body.withStart(new Object());
        request.withBody(body);
        try {
            StartClusterResponse response = client.startCluster(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

Start a cluster.


```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcdm.v1.region.cdm_region import CdmRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcdm.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CdmClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CdmRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = StartClusterRequest()
        request.cluster_id = "{cluster_id}"
        request.body = CdmStartClusterReq(
            start={}
        )
        response = client.start_cluster(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

Start a cluster.

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cdm "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()
```



```
    client := cdm.NewCdmClient(
        cdm.CdmClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.StartClusterRequest{}
    request.ClusterId = "{cluster_id}"
    var startCdmStartClusterReq interface{} = make(map[string]string)
    request.Body = &model.CdmStartClusterReq{
        Start: &startCdmStartClusterReq,
    }
    response, err := client.StartCluster(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in **API Explorer**. SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Request succeeded.
400	Request error.
401	Authentication failed.
403	No operation permissions.
404	No resources found.
500	Internal service error. For details about the returned error code, see Error Codes.
503	Service unavailable.

Error Codes

See **Error Codes**.

5.1.13 Stopping a Cluster (To Be Taken Offline)

Function

This API is used to stop a cluster.

Calling Method

For details, see **Calling APIs**.

URI

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

Table 5-68 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain it, see **Project ID and Account ID**.
cluster_id	Yes	String	Cluster ID

Request Parameters

Table 5-69 Request header parameters

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

Table 5-70 Request body parameters

Parameter	Mandatory	Type	Description
stop	Yes	**stop** object	Cluster stop operation. For details, see the descriptions of **stop** parameters.

Table 5-71 stop

Parameter	Mandatory	Type	Description
stopMode	Yes	String	Stop mode. • **IMMEDIATELY**: immediate stop • **GRACEFULLY**: graceful stop

Parameter	Mandatory	Type	Description
delayTime	No	Integer	Shutdown delay, in seconds. This parameter is valid only when **stopMode** is set to **GRACEFULLY**. If the value is **-1**, the system waits until all jobs are complete and stops accepting new jobs. If the value is greater than **0**, the system stops the cluster after the specified delay and stops accepting new jobs.

Response Parameters

Status code: 200

Table 5-72 Response body parameters

Parameter	Type	Description
jobId	Array of strings	Job ID

Example Requests

Stopping a Cluster

```
POST /v1.1/1551c7f6c808414d8e9f3c514a170f2e/clusters/bae65496-643e-47ca-84af-948672de7eeb/action

{
  "stop" : {
    "stopMode" : "GRACEFULLY",
    "delayTime" : -1
  }
}
```

Example Responses

Status code: 200

Request succeeded.

```
{
  "jobId" : [ "ff8080815e59d92d015e5b27ccb0004d" ]
}
```

Status Codes

Status Code	Description
200	Request succeeded.
400	Request error.
401	Authentication failed.
403	No operation permissions.
404	No resources found.
500	Internal service error. For details about the returned error code, see Error Codes.
503	Service unavailable.

Error Codes

See **Error Codes**.

5.1.14 Creating a Cluster

Function

This API is used to create a cluster.

Calling Method

For details, see **Calling APIs**.

URI

POST /v1.1/{project_id}/clusters

Table 5-73 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain it, see **Project ID and Account ID**.

Request Parameters

Table 5-74 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token. It can be obtained by calling the IAM API (value of X-Subject-Token in the response header).
X-Language	Yes	String	Request language

Table 5-75 Request body parameters

Parameter	Mandatory	Type	Description
cluster	Yes	**cluster** object	Cluster object. For details, see the descriptions of **cluster** parameters.
auto_remind	No	Boolean	Whether to enable message notifications. If the function is enabled, you can configure a maximum of 20 mobile numbers or email addresses. You will be notified of job failures (only table/file migration jobs) and EIP exceptions by SMS message or email.
phone_num	No	String	Mobile number for receiving notifications
email	No	String	Email address for receiving notifications

Table 5-76 cluster

Parameter	Mandatory	Type	Description
scheduleBootTime	No	String	Time for scheduled startup of a CDM cluster. The CDM cluster starts at this time every day.

Parameter	Mandatory	Type	Description
isScheduleBootOff	No	Boolean	Whether to enable scheduled startup/shutdown. The scheduled startup/shutdown and auto shutdown functions cannot be enabled at the same time.
instances	No	Array of **instance** objects	Node list. For details, see the descriptions of **instances** parameters.
datastore	No	**Datastore** object	Cluster information. For details, see the descriptions of **datastore** parameters.
extended_properties	No	**ExtendedProperties** object	Extended attribute. For details, see the descriptions of **extended_properties** parameters.
scheduleOffTime	No	String	Time for scheduled shutdown of a CDM cluster. The CDM cluster shuts down directly at this time every day without waiting for unfinished jobs to complete.
vpcId	No	String	VPC ID, which is used for configuring a network for the cluster.
name	No	String	Cluster name
sys_tags	No	Array of **sys_tags** objects	Enterprise project information. For details, see the descriptions of **sys_tags** parameters.
isAutoOff	No	Boolean	Whether to enable auto shutdown. The auto shutdown and scheduled startup/shutdown functions cannot be enabled at the same time. After auto shutdown is enabled, if no job is running in the cluster and no scheduled job is created, a cluster will automatically shut down 15 minutes later to reduce costs.

Table 5-77 instance

Parameter	Mandatory	Type	Description
availability_zone	Yes	String	AZ where a cluster is located. It can be obtained from https://developer.huaweicloud.com/intl/en-us/endpoint.
nics	Yes	Array of **nics** objects	NIC list. A maximum of two NICs are supported. For details, see the descriptions of **nics** parameters.
flavorRef	Yes	String	Instance flavor. • **a79fd5ae-1833-448a-88e8-3ea2b913e1f6**: cdm.small with 2 vCPUs and 4 GB memory applicable to Proof of Concept (PoC) verification and development tests • **fb8fe666-6734-4b11-bc6c-43d11db3c745**: cdm.medium with 4 vCPUs and 8 GB memory applicable to the migration of a single database table with fewer than 10 million pieces of data • **5ddb1071-c5d7-40e0-a874-8a032e81a697**: cdm.large with 8 vCPUs and 16 GB memory applicable to the migration of a single database table with 10 million pieces of data or more • **6ddb1072-c5d7-40e0-a874-8a032e81a698**: cdm.xlarge with 16 vCPUs and 32 GB memory applicable to TB-level data migration requiring 10GE high-speed bandwidth
type	Yes	String	Node type. Currently, only **cdm** is available.

Table 5-78 nics

Parameter	Mandatory	Type	Description
securityGroupId	Yes	String	Security group ID
net-id	Yes	String	Subnet ID

Table 5-79 Datastore

Parameter	Mandatory	Type	Description
type	No	String	Type. Generally, the value is **cdm**.
version	No	String	Cluster version

Table 5-80 ExtendedProperties

Parameter	Mandatory	Type	Description
workSpaceId	No	String	Workspace ID
resourceId	No	String	Resource ID
trial	No	String	Whether the cluster is a trial cluster

Table 5-81 sys_tags

Parameter	Mandatory	Type	Description
value	Yes	String	Enterprise project ID
key	Yes	String	The value is fixed at **_sys_enterprise_project_id**.

Response Parameters

Status code: 202

Table 5-82 Response body parameters

Parameter	Type	Description
name	String	Cluster name

Parameter	Type	Description
id	String	Cluster ID
task	**Task** object	Task information
datastore	**Datastore** object	Cluster information
instances	Array of **ClusterInstance** objects	Cluster node information

Table 5-83 Task

Parameter	Type	Description
id	String	Task ID
name	String	Task name

Table 5-84 Datastore

Parameter	Type	Description
type	String	Type. Generally, the value is **cdm**.
version	String	Cluster version

Table 5-85 ClusterInstance

Parameter	Type	Description
id	String	Node VM ID
name	String	Name of the VM on the node
type	String	Node type. Currently, only **cdm** is available.
shard_id	String	Shard ID

Example Requests

Creating a CDM cluster whose version is 1.8.10 and name is **cdm-ab82**

```
POST /v1.1/1551c7f6c808414d8e9f3c514a170f2e/clusters

{
  "cluster" : {
    "scheduleBootTime" : "",
    "isScheduleBootOff" : false,
```

```
  "instances" : [ {
    "availability_zone" : "xx-xxx",
    "nics" : [ {
      "securityGroupId" : "c37852d2-2d12-41cb-af47-65c80e995c80",
      "net-id" : "2d120298-6130-44d4-a438-454912fff901"
    } ],
    "flavorRef" : "5ddb1071-c5d7-40e0-a874-8a032e81a697",
    "type" : "cdm"
  } ],
  "datastore" : {
    "type" : "cdm",
    "version" : "1.8.10"
  },
  "scheduleOffTime" : "",
  "vpcId" : "67c06084-2212-4242-bcd4-d2144c2385a9",
  "name" : "cdm-ab82",
  "sys_tags" : [ {
    "value" : "1ce45885-4033-40d2-bdde-d4dbaceb387d",
    "key" : "_sys_enterprise_project_id"
  } ],
  "isAutoOff" : false
 },
 "auto_remind" : false,
 "phone_num" : "",
 "email" : ""
}
```

Example Responses

Status code: 202

Accepted.

```
{
 "id" : "befc862c-9286-46a0-a1d6-300d98b63aad",
 "name" : "cdm-4ef213",
 "task" : {
   "id" : "2c9080047f1b1185017f1ef6ad0500ac",
   "name" : "rdsCreateBackupJob"
 },
 "datastore" : {
   "type" : "cdm",
   "version" : "2.9.1.100"
 },
 "instances" : [ {
   "id" : "b2672e7d-2faf-423f-96bb-0664cd743cfd",
   "name" : "cdm-4ef213-cdm-dn-1-1",
   "type" : "cdm",
   "shard_id" : "dn-1"
 } ]
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Creating a CDM cluster whose version is 1.8.10 and name is **cdm-ab82**

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
```



```
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cdm.v1.region.CdmRegion;
import com.huaweicloud.sdk.cdm.v1.*;
import com.huaweicloud.sdk.cdm.v1.model.*;

import java.util.List;
import java.util.ArrayList;

public class CreateClusterSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CdmClient client = CdmClient.newBuilder()
                .withCredential(auth)
                .withRegion(CdmRegion.valueOf("<YOUR REGION>"))
                .build();
        CreateClusterRequest request = new CreateClusterRequest();
        CdmCreateClusterReq body = new CdmCreateClusterReq();
        List<SysTags> listClusterSysTags = new ArrayList<>();
        listClusterSysTags.add(
            new SysTags()
                .withValue("1ce45885-4033-40d2-bdde-d4dbaceb387d")
                .withKey("_sys_enterprise_project_id")
        );
        Datastore datastoreCluster = new Datastore();
        datastoreCluster.withType("cdm")
            .withVersion("1.8.10");
        List<Nics> listInstancesNics = new ArrayList<>();
        listInstancesNics.add(
            new Nics()
                .withSecurityGroupId("c37852d2-2d12-41cb-af47-65c80e995c80")
                .withNetId("2d120298-6130-44d4-a438-454912fff901")
        );
        List<Instance> listClusterInstances = new ArrayList<>();
        listClusterInstances.add(
            new Instance()
                .withAvailabilityZone("xx-xxx")
                .withNics(listInstancesNics)
                .withFlavorRef("5ddb1071-c5d7-40e0-a874-8a032e81a697")
                .withType("cdm")
        );
        CdmCreateClusterReqCluster clusterbody = new CdmCreateClusterReqCluster();
        clusterbody.withScheduleBootTime("")
            .withIsScheduleBootOff(false)
            .withInstances(listClusterInstances)
            .withDatastore(datastoreCluster)
            .withScheduleOffTime("")
            .withVpcId("67c06084-2212-4242-bcd4-d2144c2385a9")
            .withName("cdm-ab82")
            .withSysTags(listClusterSysTags)
            .withIsAutoOff(false);
        body.withEmail("");
        body.withPhoneNum("");
        body.withAutoRemind(false);
        body.withCluster(clusterbody);
        request.withBody(body);
```



```
        try {
            CreateClusterResponse response = client.createCluster(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

Creating a CDM cluster whose version is 1.8.10 and name is **cdm-ab82**

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcdm.v1.region.cdm_region import CdmRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcdm.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CdmClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CdmRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateClusterRequest()
        listSysTagsCluster = [
            SysTags(
                value="1ce45885-4033-40d2-bdde-d4dbaceb387d",
                key="_sys_enterprise_project_id"
            )
        ]
        datastoreCluster = Datastore(
            type="cdm",
            version="1.8.10"
        )
        listNicsInstances = [
            Nics(
                security_group_id="c37852d2-2d12-41cb-af47-65c80e995c80",
                net_id="2d120298-6130-44d4-a438-454912fff901"
            )
        ]
        listInstancesCluster = [
            Instance(
                availability_zone="xx-xxx",
                nics=listNicsInstances,
                flavor_ref="5ddb1071-c5d7-40e0-a874-8a032e81a697",
```



```
            type="cdm"
        )
    ]
    clusterbody = CdmCreateClusterReqCluster(
        schedule_boot_time="",
        is_schedule_boot_off=False,
        instances=listInstancesCluster,
        datastore=datastoreCluster,
        schedule_off_time="",
        vpc_id="67c06084-2212-4242-bcd4-d2144c2385a9",
        name="cdm-ab82",
        sys_tags=listSysTagsCluster,
        is_auto_off=False
    )
    request.body = CdmCreateClusterReq(
        email="",
        phone_num="",
        auto_remind=False,
        cluster=clusterbody
    )
    response = client.create_cluster(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

Creating a CDM cluster whose version is 1.8.10 and name is **cdm-ab82**

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cdm "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cdm.NewCdmClient(
        cdm.CdmClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreateClusterRequest{}
    var listSysTagsCluster = []model.SysTags{
        {
            Value: "1ce45885-4033-40d2-bdde-d4dbaceb387d",
            Key: "_sys_enterprise_project_id",
```



```
        },
    }
    typeDatastore:= "cdm"
    versionDatastore:= "1.8.10"
    datastoreCluster := &model.Datastore{
        Type: &typeDatastore,
        Version: &versionDatastore,
    }
    var listNicsInstances = []model.Nics{
        {
            SecurityGroupId: "c37852d2-2d12-41cb-af47-65c80e995c80",
            NetId: "2d120298-6130-44d4-a438-454912fff901",
        },
    }
    var listInstancesCluster = []model.Instance{
        {
            AvailabilityZone: "xx-xxx",
            Nics: listNicsInstances,
            FlavorRef: "5ddb1071-c5d7-40e0-a874-8a032e81a697",
            Type: "cdm",
        },
    }
    scheduleBootTimeCluster:= ""
    isScheduleBootOffCluster:= false
    scheduleOffTimeCluster:= ""
    vpcIdCluster:= "67c06084-2212-4242-bcd4-d2144c2385a9"
    nameCluster:= "cdm-ab82"
    isAutoOffCluster:= false
    clusterbody := &model.CdmCreateClusterReqCluster{
        ScheduleBootTime: &scheduleBootTimeCluster,
        IsScheduleBootOff: &isScheduleBootOffCluster,
        Instances: &listInstancesCluster,
        Datastore: datastoreCluster,
        ScheduleOffTime: &scheduleOffTimeCluster,
        VpcId: &vpcIdCluster,
        Name: &nameCluster,
        SysTags: &listSysTagsCluster,
        IsAutoOff: &isAutoOffCluster,
    }
    emailCdmCreateClusterReq:= ""
    phoneNumCdmCreateClusterReq:= ""
    autoRemindCdmCreateClusterReq:= false
    request.Body = &model.CdmCreateClusterReq{
        Email: &emailCdmCreateClusterReq,
        PhoneNum: &phoneNumCdmCreateClusterReq,
        AutoRemind: &autoRemindCdmCreateClusterReq,
        Cluster: clusterbody,
    }
    response, err := client.CreateCluster(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in **API Explorer**. SDK sample code can be automatically generated.

Status Codes

Status Code	Description
202	Accepted.
400	Request error.
401	Authorization failed.
403	No operation permissions.
404	No resources found.
500	Internal interface error.
503	Service unavailable.

Error Codes

See **Error Codes**.

5.1.15 Querying the Cluster List

Function

This API is used to query the cluster list.

Calling Method

For details, see **Calling APIs**.

URI

GET /v1.1/{project_id}/clusters

Table 5-86 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain it, see **Project ID and Account ID**.

Request Parameters

Table 5-87 Request header parameters

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

Response Parameters

Status code: 200

Table 5-88 Response body parameters

Parameter	Type	Description
clusters	Array of **clusters** objects	Cluster list. For details, see the descriptions of **clusters** parameters.

Table 5-89 clusters

Parameter	Type	Description
customerConfig	**CustomerConfig** object	User configuration
datastore	**Datastore** object	CDM information
instances	Array of **ClusterDetailInstance** objects	Cluster node information. For details, see the descriptions of **instances** parameters.
azName	String	AZ name
dbuser	String	Database user
flavorName	String	Flavor name
recentEvent	Integer	Number of events
isAutoOff	Boolean	Automatic shutdown

Parameter	Type	Description
isScheduleBootOff	Boolean	Whether to enable scheduled startup/shutdown. The scheduled startup/shutdown and auto shutdown functions cannot be enabled at the same time.
clusterMode	String	Cluster mode: sharding
namespace	String	Namespace
task	**ClusterTask** object	Task information
publicEndpoint	String	EIP bound to the cluster
actionProgress	**ActionProgress** object	Cluster operation progress, which consists of a key and a value. The key indicates an ongoing task, and the value indicates the progress of the ongoing task. An example is "action_progress":{"SNAPSHOTTING":"16%"}.
created	String	Cluster creation time in ISO 8601 format: YYYY-MM-DDThh:mm:ssZ
bakExpectedStartTime	String	Start time
bakKeepDay	Integer	Retention period
name	String	Cluster name
statusDetail	String	Cluster status: normal
id	String	Cluster ID
isFrozen	String	Whether the cluster is frozen. The value can be **0** (not frozen) or **1** (frozen).
updated	String	Cluster update time in ISO 8601 format: YYYY-MM-DDThh:mm:ssZ
status	String	Cluster status. The options are as follows: • **100**: creating • **200**: normal • **300**: failed • **303**: creation failed • **500**: restarting • **800**: frozen • **900**: stopped • **910**: stopping • **920**: starting

Parameter	Type	Description
failedReasons	FailedReasons object	Failure cause. If this parameter is left empty, the cluster is in normal state.

Table 5-90 CustomerConfig

Parameter	Type	Description
failureRemind	String	Failure notification
clusterName	String	Cluster type
serviceProvider	String	Service provider
localDisk	String	Whether the disk is a local disk
ssl	String	Whether to enable SSL
createFrom	String	Source
resourceId	String	Resource ID
flavorType	String	Flavor type
workSpaceId	String	Workspace ID
trial	String	Trial

Table 5-91 Datastore

Parameter	Type	Description
type	String	Type. Generally, the value is **cdm**.
version	String	Cluster version

Table 5-92 ClusterDetailInstance

Parameter	Type	Description
flavor	flavor object	VM flavor of a node. For details, see the descriptions of **flavor** parameters.
volume	volume object	Disk information of a node. For details, see the descriptions of **volume** parameters.

Parameter	Type	Description
status	String	Node status • **100**: creating • **200**: normal • **300**: failed • **303**: failed to be created • **400**: deleted • **800**: frozen
actions	Array of strings	Node operation status • **REBOOTING**: restarting • **RESTORING**: restoring • **REBOOT_FAILURE**: failed to restart
type	String	Node type. Currently, only **cdm** is available.
id	String	Node VM ID
name	String	Name of the VM on the node
isFrozen	String	Whether the node is frozen. The value can be **0** (not frozen) or **1** (frozen).
components	String	Components
config_status	String	Node configuration status. The value is null when the cluster list is queried. • **In-Sync**: The configuration has been synchronized. • **Applying**: The configuration is in progress. • **Sync-Failure**: The configuration fails.
role	String	Instance role
group	String	Group
links	Array of **ClusterLinks** objects	Links (null is returned for querying the cluster list)
paramsGroupId	String	Group ID
publicIp	String	Public IP address
manageIp	String	Management IP address
trafficIp	String	Traffic IP address
shard_id	String	Slice ID
manage_fix_ip	String	Management fix IP address

Parameter	Type	Description
private_ip	String	Private IP address
internal_ip	String	Internal IP address
resource	Array of **Resource** objects	Resource information (null is returned for querying the cluster list)

Table 5-93 flavor

Parameter	Type	Description
id	String	VM flavor ID
links	Array of **ClusterLinks** objects	Links

Table 5-94 volume

Parameter	Type	Description
type	String	Type of disks on the node. Only local disks are supported.
size	Long	Size of the disk on the node (GB)

Table 5-95 ClusterLinks

Parameter	Type	Description
rel	String	Relationship
href	String	Link address

Table 5-96 Resource

Parameter	Type	Description
resource_id	String	Resource ID
resource_type	String	Resource type: server

Table 5-97 ClusterTask

Parameter	Type	Description
description	String	Task description
id	String	Task ID
name	String	Task name

Table 5-98 ActionProgress

Parameter	Type	Description
CREATING	String	Cluster creation progress, for example, 29%
GROWING	String	Cluster expansion progress, for example, 29%
RESTORING	String	Cluster restoration progress, for example, 29%
SNAPSHOTTING	String	Cluster snapshotting progress, for example, 29%
REPAIRING	String	Cluster repairing progress, for example, 29%

Table 5-99 FailedReasons

Parameter	Type	Description
CREATE_FAILED	**CREATE_FAILED** object	Cause of the cluster creation failure

Table 5-100 CREATE_FAILED

Parameter	Type	Description
errorCode	String	Error code
errorMsg	String	Failure cause

Example Requests

```
GET /v1.1/1551c7f6c808414d8e9f3c514a170f2e/clusters
```

Example Responses

Status code: 200

Request succeeded.

```
{
  "clusters" : [ {
```



```
    "publicEndpoint" : "49.xx.xx.10",
    "actionProgress" : { },
    "created" : "2018-09-05T08:38:25",
    "name" : "cdm-c018",
    "statusDetail" : "Normal",
    "id" : "bae65496-643e-47ca-84af-948672de7eeb",
    "isFrozen" : "0",
    "updated" : "2018-09-05T08:38:25",
    "status" : "200"
  } ]
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cdm.v1.region.CdmRegion;
import com.huaweicloud.sdk.cdm.v1.*;
import com.huaweicloud.sdk.cdm.v1.model.*;


public class ListClustersSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CdmClient client = CdmClient.newBuilder()
                .withCredential(auth)
                .withRegion(CdmRegion.valueOf("<YOUR REGION>"))
                .build();
        ListClustersRequest request = new ListClustersRequest();
        try {
            ListClustersResponse response = client.listClusters(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```


Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcdm.v1.region.cdm_region import CdmRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcdm.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CdmClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CdmRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListClustersRequest()
        response = client.list_clusters(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cdm "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cdm.NewCdmClient(
        cdm.CdmClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
```



```
            Build())

    request := &model.ListClustersRequest{}
    response, err := client.ListClusters(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in **API Explorer**. SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Request succeeded.
400	Request error.
401	Authorization failed.
403	No operation permissions.
404	No resources found.
500	Internal service error. For details about the returned error code, see Error Codes.
503	Service unavailable.

Error Codes

See **Error Codes**.

5.2 Job Management

5.2.1 Querying a Job

Function

This API is used to query jobs.

Calling Method

For details, see **Calling APIs**.

URI

GET /v1.1/{project_id}/clusters/{cluster_id}/cdm/job/{job_name}

Table 5-101 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain it, see **Project ID and Account ID**.
cluster_id	Yes	String	Cluster ID
job_name	Yes	String	Job name. When this parameter is set to **all**, all jobs are to be queried.

Table 5-102 Query Parameters

Parameter	Mandatory	Type	Description
filter	No	String	When **job_name** is **all**, this parameter is used for fuzzy job filtering.
page_no	No	Integer	Page number
page_size	No	Integer	Number of jobs on each page. The value ranges from 10 to 100.
jobType	No	String	Type of the jobs to be queried • **jobType=NORMAL_JOB**: table/file migration job • **jobType=BATCH_JOB**: entire DB migration job • **jobType=SCENARIO_JOB**: scenario migration job If this parameter is not specified, only table/file migration jobs are queried by default.

Request Parameters

Table 5-103 Request header parameters

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

Response Parameters

Status code: 200

Table 5-104 Response body parameters

Parameter	Type	Description
total	Integer	Number of jobs. The value is **0** for querying a single job.
jobs	Array of **Job** objects	Job list. For details, see the descriptions of **jobs** parameters.
page_no	Integer	Page number. Jobs on the specified page will be returned.
page_size	Integer	Number of jobs on each page

Table 5-105 Job

Parameter	Type	Description
job_type	String	Job type • **NORMAL_JOB**: table/file migration • **BATCH_JOB**: entire DB migration • **SCENARIO_JOB**: scenario migration

Parameter	Type	Description
from-connector-name	String	Source link type. The corresponding link parameters are as follows: • **generic-jdbc-connector**: link to relational database • **obs-connector**: link to OBS • **hdfs-connector**: link to HDFS • **hbase-connector**: link to HBase and link to CloudTable • **hive-connector**: link to Hive • **ftp-connector/sftp-connector**: link to an FTP or SFTP server • **mongodb-connector**: link to MongoDB • **redis-connector**: link to Redis/DCS • **kafka-connector**: link to Kafka • **dis-connector**: link to DIS • **elasticsearch-connector**: link to Elasticsearch/Cloud Search Service (CSS) • [**dli-connector**: link to DLI] (tag:nohcs) • **http-connector**: link to HTTP/HTTPS servers (No link parameters are required.) • **dms-kafka-connector**: link to DMS Kafka
to-config-values	**ConfigValues** object	Destination link parameters, which vary depending on the destination. For details, see **Destination Job Parameters**.
to-link-name	String	Name of the destination link, that is, the name of the link created through the API used to create a link
driver-config-values	**ConfigValues** object	Job parameters, such as **Retry upon Failure** and **Concurrent Extractors**. For details, see **Job Parameter Description**.
from-config-values	**ConfigValues** object	Source link parameters, which vary depending on the source. For details, see **Source Job Parameters**.

Parameter	Type	Description
to-connector-name	String	Destination link type. The corresponding link parameters are as follows: • **generic-jdbc-connector**: link to relational database • **obs-connector**: link to OBS • **hdfs-connector**: link to HDFS • **hbase-connector**: link to HBase and link to CloudTable • **hive-connector**: link to Hive • **ftp-connector/sftp-connector**: link to an FTP or SFTP server • **mongodb-connector**: link to MongoDB • **redis-connector**: link to Redis/DCS • **kafka-connector**: link to Kafka • **dis-connector**: link to DIS • **elasticsearch-connector**: link to Elasticsearch/Cloud Search Service (CSS) • [**dli-connector**: link to DLI] (tag:nohcs) • **http-connector**: link to HTTP/HTTPS servers (No link parameters are required.) • **dms-kafka-connector**: link to DMS Kafka
name	String	Job name, which contains 1 to 240 characters
from-link-name	String	Name of the source link, that is, the name of the link created through the API used to create a link
creation-user	String	User who created the job. The value is generated by the system.
creation-date	Long	Time when the job was created, accurate to millisecond. The value is generated by the system.
update-date	Long	Time when the job was last updated, accurate to millisecond. The value is generated by the system.
is_incre_job	Boolean	Whether the job is an incremental job. This parameter has been discarded.
flag	Integer	Whether the job is a scheduled job. If yes, the value is **1**. Otherwise, the value is **0**. The value is generated by the system based on the scheduled task configuration.

Parameter	Type	Description
files_read	Integer	Number of read files. The value is generated by the system.
update-user	String	User who last updated the job. The value is generated by the system.
external_id	String	ID of the job to be executed. For a local job, the value is in the format of **job_local1202051771_0002** . For a DLI job, the value is the DLI job ID, for example, **"12345"**. The value is generated by the system and does not need to be set.
type	String	Job type. The value of this parameter is the same as that of **job_type**. The options are as follows: • **NORMAL_JOB**: table/file migration job • **BATCH_JOB**: entire DB migration job • **SCENARIO_JOB**: scenario migration job
execute_start_date	Long	Time when the last task was started, accurate to millisecond. The value is generated by the system.
delete_rows	Integer	Number of rows deleted by an incremental job. This parameter is deprecated.
enabled	Boolean	Whether the link is enabled. The value is generated by the system.
bytes_written	Long	Number of bytes written by the job. The value is generated by the system.
id	Integer	Job ID, which is generated by the system
is_use_sql	Boolean	Whether to use SQL statements. The value is generated by the system based on whether SQL statements are used for source data extraction. You do not need to set this parameter.
update_rows	Integer	Number of updated rows in an incremental job. This parameter is deprecated.
group_name	String	Group name
bytes_read	Long	Number of bytes read by the job. The value is generated by the system.
execute_update_date	Long	Time when the last task was updated, accurate to millisecond. The value is generated by the system.

Parameter	Type	Description
write_rows	Integer	Number of rows written by an incremental job. This parameter is deprecated.
rows_written	Integer	Number of rows written by the job. The value is generated by the system.
rows_read	Long	Number of rows read by the job. The value is generated by the system.
files_written	Integer	Number of written files. The value is generated by the system.
is_incrementing	Boolean	Whether the job is an incremental job. Similar to parameter **is_incre_job**, this parameter is deprecated.
execute_create_date	Long	Time when the last task was created, accurate to millisecond. The value is generated by the system.
status	String	Job execution status • **BOOTING**: The job is starting. • **RUNNING**: The job is running. • **SUCCEEDED**: The job was successfully executed. • **FAILED**: The job execution failed. • **NEW**: The job was not executed.

Table 5-106 ConfigValues

Parameter	Type	Description
configs	Array of **configs** objects	The data structures of source link parameters, destination link parameters, and job parameters are the same. However, the **inputs** parameter varies. For details, see the descriptions of **configs** parameters.
extended-configs	**extended-configs** object	Extended configuration. For details, see the descriptions of **extended-configs** parameters. The extended configuration is not open to external systems. You do not need to set it.

Table 5-107 configs

Parameter	Type	Description
inputs	Array of **Input** objects	Input parameter list. Each element in the list is in *name,value* format. For details, see the descriptions of **inputs** parameters. In the **from-config-values** data structure, the value of this parameter varies with the source link type. For details, see section "Source Job Parameters" in the *Cloud Data Migration User Guide*. In the **to-config-values** data structure, the value of this parameter varies with the destination link type. For details, see section "Destination Job Parameters" in the *Cloud Data Migration User Guide*. For details about the **inputs** parameter in the **driver-config-values** data structure, see the job parameter descriptions.
name	String	Configuration name. The value is **fromJobConfig** for a source job, **toJobConfig** for a destination job, and **linkConfig** for a link.
id	Integer	Configuration ID, which is generated by the system. You do not need to set this parameter.
type	String	Configuration type, which is generated by the system. You do not need to set this parameter. The value can be **LINK** (for link management APIs) or **JOB** (for job management APIs).

Table 5-108 Input

Parameter	Type	Description
name	String	Parameter name. • For link management APIs, parameter names start with **linkConfig.**. The parameters vary depending on the link type. For details, see the parameter descriptions of the corresponding link in **Link Parameters**. • For job management APIs, source link parameter names start with **fromJobConfig.**. For details, see the source job parameters in **Source Job Parameters**. Destination link parameter names start with **toJobConfig.**. For details, see **Destination Job Parameters**. For details about job parameters, see the task parameter descriptions in **Job Parameters**.

Parameter	Type	Description
value	Object	Parameter value, which must be a string.
type	String	Value type, such as **STRING** and **INTEGER**. The value is set by the system.

Table 5-109 extended-configs

Parameter	Type	Description
name	String	Extended configuration name. This parameter is unavailable for external systems and does not need to be set.
value	String	Extended configuration value. This parameter is unavailable for external systems and does not need to be set.

Example Requests

```
GET /v1.1/1551c7f6c808414d8e9f3c514a170f2e/clusters/6ec9a0a4-76be-4262-8697-e7af1fac7920/cdm/job/all?jobType=NORMAL_JOB
```

Example Responses

Status code: 200

Request succeeded.

```
{
  "total" : 1,
  "jobs" : [ {
    "job_type" : "NORMAL_JOB",
    "from-connector-name" : "elasticsearch-connector",
    "to-config-values" : {
      "configs" : [ {
        "inputs" : [ {
          "name" : "toJobConfig.streamName",
          "value" : "dis-lkGm"
        }, {
          "name" : "toJobConfig.separator",
          "value" : "|"
        }, {
          "name" : "toJobConfig.columnList",
          "value" : "1&2&3"
        } ],
        "name" : "toJobConfig"
      } ]
    },
    "to-link-name" : "dis",
    "driver-config-values" : {
      "configs" : [ {
        "inputs" : [ {
          "name" : "throttlingConfig.numExtractors",
          "value" : "1"
        }, {
          "name" : "throttlingConfig.submitToCluster",
          "value" : "false"
```



```
      }, {
        "name" : "throttlingConfig.numLoaders",
        "value" : "1"
      }, {
        "name" : "throttlingConfig.recordDirtyData",
        "value" : "false"
      } ],
      "name" : "throttlingConfig"
    }, {
      "inputs" : [ ],
      "name" : "jarConfig"
    }, {
      "inputs" : [ {
        "name" : "schedulerConfig.isSchedulerJob",
        "value" : "false"
      }, {
        "name" : "schedulerConfig.disposableType",
        "value" : "NONE"
      } ],
      "name" : "schedulerConfig"
    }, {
      "inputs" : [ ],
      "name" : "transformConfig"
    }, {
      "inputs" : [ {
        "name" : "retryJobConfig.retryJobType",
        "value" : "NONE"
      } ],
      "name" : "retryJobConfig"
    } ]
  },
  "from-config-values" : {
    "configs" : [ {
      "inputs" : [ {
        "name" : "fromJobConfig.index",
        "value" : "52est"
      }, {
        "name" : "fromJobConfig.type",
        "value" : "est_array"
      }, {
        "name" : "fromJobConfig.columnList",
        "value" : "array_f1_int:long&array_f2_text:string&array_f3_object:nested"
      }, {
        "name" : "fromJobConfig.splitNestedField",
        "value" : "false"
      } ],
      "name" : "fromJobConfig"
    } ]
  },
  "to-connector-name" : "dis-connector",
  "name" : "es_css",
  "from-link-name" : "css"
 } ],
 "page_no" : 1,
 "page_size" : 10
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
```



```
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cdm.v1.region.CdmRegion;
import com.huaweicloud.sdk.cdm.v1.*;
import com.huaweicloud.sdk.cdm.v1.model.*;


public class ShowJobsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CdmClient client = CdmClient.newBuilder()
                .withCredential(auth)
                .withRegion(CdmRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowJobsRequest request = new ShowJobsRequest();
        request.withClusterId("{cluster_id}");
        request.withJobName("{job_name}");
        try {
            ShowJobsResponse response = client.showJobs(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcdm.v1.region.cdm_region import CdmRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcdm.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)
```



```
    client = CdmClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CdmRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowJobsRequest()
        request.cluster_id = "{cluster_id}"
        request.job_name = "{job_name}"
        response = client.show_jobs(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cdm "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cdm.NewCdmClient(
        cdm.CdmClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowJobsRequest{}
    request.ClusterId = "{cluster_id}"
    request.JobName = "{job_name}"
    response, err := client.ShowJobs(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in **API Explorer**. SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Request succeeded.

Error Codes

See **Error Codes**.

5.2.2 Deleting a Job

Function

This API is used to delete a job.

Calling Method

For details, see **Calling APIs**.

URI

DELETE /v1.1/{project_id}/clusters/{cluster_id}/cdm/job/{job_name}

Table 5-110 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain it, see **Project ID and Account ID**.
cluster_id	Yes	String	Cluster ID
job_name	Yes	String	Job name

Request Parameters

Table 5-111 Request header parameters

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

Response Parameters

Status code: 500

Table 5-112 Response body parameters

Parameter	Type	Description
errCode	String	Error code
externalMessage	String	Error message

Example Requests

```
DELETE /v1.1/1551c7f6c808414d8e9f3c514a170f2e/clusters/6ec9a0a4-76be-4262-8697-e7af1fac7920/cdm/job/jdbc2hive
```

Example Responses

Status code: 500

Internal service error. For details about the returned error code, see Error Codes.

```
{
  "errCode" : "Cdm.0100",
  "externalMessage" : "Job[jdbc2hive] doesn't exist."
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cdm.v1.region.CdmRegion;
import com.huaweicloud.sdk.cdm.v1.*;
import com.huaweicloud.sdk.cdm.v1.model.*;


public class DeleteJobSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
```



```
            .withAk(ak)
            .withSk(sk);

    CdmClient client = CdmClient.newBuilder()
            .withCredential(auth)
            .withRegion(CdmRegion.valueOf("<YOUR REGION>"))
            .build();
    DeleteJobRequest request = new DeleteJobRequest();
    request.withClusterId("{cluster_id}");
    request.withJobName("{job_name}");
    try {
        DeleteJobResponse response = client.deleteJob(request);
        System.out.println(response.toString());
    } catch (ConnectionException e) {
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
  }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcdm.v1.region.cdm_region import CdmRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcdm.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CdmClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CdmRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DeleteJobRequest()
        request.cluster_id = "{cluster_id}"
        request.job_name = "{job_name}"
        response = client.delete_job(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```


Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cdm "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cdm.NewCdmClient(
        cdm.CdmClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.DeleteJobRequest{}
    request.ClusterId = "{cluster_id}"
    request.JobName = "{job_name}"
    response, err := client.DeleteJob(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in **API Explorer**. SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Request succeeded.
400	Request error.
401	Authentication failed.
403	No operation permissions.
404	No resources found.

Status Code	Description
500	Internal service error. For details about the returned error code, see Error Codes.

Error Codes

See **Error Codes**.

5.2.3 Modifying a Job

Function

This API is used to modify a job.

Calling Method

For details, see **Calling APIs**.

URI

PUT /v1.1/{project_id}/clusters/{cluster_id}/cdm/job/{job_name}

Table 5-113 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain it, see **Project ID and Account ID**.
cluster_id	Yes	String	Cluster ID
job_name	Yes	String	Job name

Request Parameters

Table 5-114 Request header parameters

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

Table 5-115 Request body parameters

Parameter	Mandatory	Type	Description
jobs	Yes	Array of **Job** objects	Job list. For details, see the descriptions of **jobs** parameters.

Table 5-116 Job

Parameter	Mandatory	Type	Description
job_type	Yes	String	Job type • **NORMAL_JOB**: table/file migration • **BATCH_JOB**: entire DB migration • **SCENARIO_JOB**: scenario migration

Parameter	Mandatory	Type	Description
from-connector-name	Yes	String	Source link type. The corresponding link parameters are as follows: • **generic-jdbc-connector**: link to relational database • **obs-connector**: link to OBS • **hdfs-connector**: link to HDFS • **hbase-connector**: link to HBase and link to CloudTable • **hive-connector**: link to Hive • **ftp-connector/sftp-connector**: link to an FTP or SFTP server • **mongodb-connector**: link to MongoDB • **redis-connector**: link to Redis/DCS • **kafka-connector**: link to Kafka • **dis-connector**: link to DIS • **elasticsearch-connector**: link to Elasticsearch/Cloud Search Service (CSS) • [**dli-connector**: link to DLI](tag:nohcs) • **http-connector**: link to HTTP/HTTPS servers (No link parameters are required.) • **dms-kafka-connector**: link to DMS Kafka
to-config-values	Yes	**ConfigValues** object	Destination link parameters, which vary depending on the destination. For details, see **Destination Job Parameters**.
to-link-name	Yes	String	Name of the destination link, that is, the name of the link created through the API used to create a link

Parameter	Mandatory	Type	Description
driver-config-values	Yes	**ConfigValues** object	Job parameters, such as **Retry upon Failure** and **Concurrent Extractors**. For details, see **Job Parameter Description**.
from-config-values	Yes	**ConfigValues** object	Source link parameters, which vary depending on the source. For details, see **Source Job Parameters**.
to-connector-name	Yes	String	Destination link type. The corresponding link parameters are as follows: • **generic-jdbc-connector**: link to relational database • **obs-connector**: link to OBS • **hdfs-connector**: link to HDFS • **hbase-connector**: link to HBase and link to CloudTable • **hive-connector**: link to Hive • **ftp-connector/sftp-connector**: link to an FTP or SFTP server • **mongodb-connector**: link to MongoDB • **redis-connector**: link to Redis/DCS • **kafka-connector**: link to Kafka • **dis-connector**: link to DIS • **elasticsearch-connector**: link to Elasticsearch/Cloud Search Service (CSS) • [**dli-connector**: link to DLI] (tag:nohcs) • **http-connector**: link to HTTP/HTTPS servers (No link parameters are required.) • **dms-kafka-connector**: link to DMS Kafka
name	Yes	String	Job name, which contains 1 to 240 characters

Parameter	Mandatory	Type	Description
from-link-name	Yes	String	Name of the source link, that is, the name of the link created through the API used to create a link
creation-user	No	String	User who created the job. The value is generated by the system.
creation-date	No	Long	Time when the job was created, accurate to millisecond. The value is generated by the system.
update-date	No	Long	Time when the job was last updated, accurate to millisecond. The value is generated by the system.
is_incre_job	No	Boolean	Whether the job is an incremental job. This parameter has been discarded.
flag	No	Integer	Whether the job is a scheduled job. If yes, the value is **1**. Otherwise, the value is **0**. The value is generated by the system based on the scheduled task configuration.
files_read	No	Integer	Number of read files. The value is generated by the system.
update-user	No	String	User who last updated the job. The value is generated by the system.
external_id	No	String	ID of the job to be executed. For a local job, the value is in the format of **job_local1202051771_0002** . For a DLI job, the value is the DLI job ID, for example, **"12345"**. The value is generated by the system and does not need to be set.

Parameter	Mandatory	Type	Description
type	No	String	Job type. The value of this parameter is the same as that of **job_type**. The options are as follows: • **NORMAL_JOB**: table/file migration job • **BATCH_JOB**: entire DB migration job • **SCENARIO_JOB**: scenario migration job
execute_start_date	No	Long	Time when the last task was started, accurate to millisecond. The value is generated by the system.
delete_rows	No	Integer	Number of rows deleted by an incremental job. This parameter is deprecated.
enabled	No	Boolean	Whether the link is enabled. The value is generated by the system.
bytes_written	No	Long	Number of bytes written by the job. The value is generated by the system.
id	No	Integer	Job ID, which is generated by the system
is_use_sql	No	Boolean	Whether to use SQL statements. The value is generated by the system based on whether SQL statements are used for source data extraction. You do not need to set this parameter.
update_rows	No	Integer	Number of updated rows in an incremental job. This parameter is deprecated.
group_name	No	String	Group name
bytes_read	No	Long	Number of bytes read by the job. The value is generated by the system.
execute_update_date	No	Long	Time when the last task was updated, accurate to millisecond. The value is generated by the system.

Parameter	Mandatory	Type	Description
write_rows	No	Integer	Number of rows written by an incremental job. This parameter is deprecated.
rows_written	No	Integer	Number of rows written by the job. The value is generated by the system.
rows_read	No	Long	Number of rows read by the job. The value is generated by the system.
files_written	No	Integer	Number of written files. The value is generated by the system.
is_incrementing	No	Boolean	Whether the job is an incremental job. Similar to parameter **is_incre_job**, this parameter is deprecated.
execute_create_date	No	Long	Time when the last task was created, accurate to millisecond. The value is generated by the system.
status	No	String	Job execution status • **BOOTING**: The job is starting. • **RUNNING**: The job is running. • **SUCCEEDED**: The job was successfully executed. • **FAILED**: The job execution failed. • **NEW**: The job was not executed.

Table 5-117 ConfigValues

Parameter	Mandatory	Type	Description
configs	Yes	Array of **configs** objects	The data structures of source link parameters, destination link parameters, and job parameters are the same. However, the **inputs** parameter varies. For details, see the descriptions of **configs** parameters.
extended-configs	No	**extended-configs** object	Extended configuration. For details, see the descriptions of **extended-configs** parameters. The extended configuration is not open to external systems. You do not need to set it.

Table 5-118 configs

Parameter	Mandatory	Type	Description
inputs	Yes	Array of **Input** objects	Input parameter list. Each element in the list is in *name,value* format. For details, see the descriptions of **inputs** parameters. In the **from-config-values** data structure, the value of this parameter varies with the source link type. For details, see section "Source Job Parameters" in the *Cloud Data Migration User Guide*. In the **to-config-values** data structure, the value of this parameter varies with the destination link type. For details, see section "Destination Job Parameters" in the *Cloud Data Migration User Guide*. For details about the **inputs** parameter in the **driver-config-values** data structure, see the job parameter descriptions.

Parameter	Mandatory	Type	Description
name	Yes	String	Configuration name. The value is **fromJobConfig** for a source job, **toJobConfig** for a destination job, and **linkConfig** for a link.
id	No	Integer	Configuration ID, which is generated by the system. You do not need to set this parameter.
type	No	String	Configuration type, which is generated by the system. You do not need to set this parameter. The value can be **LINK** (for link management APIs) or **JOB** (for job management APIs).

Table 5-119 Input

Parameter	Mandatory	Type	Description
name	Yes	String	Parameter name. • For link management APIs, parameter names start with **linkConfig.**. The parameters vary depending on the link type. For details, see the parameter descriptions of the corresponding link in **Link Parameters**. • For job management APIs, source link parameter names start with **fromJobConfig.**. For details, see the source job parameters in **Source Job Parameters**. Destination link parameter names start with **toJobConfig.**. For details, see **Destination Job Parameters**. For details about job parameters, see the task parameter descriptions in **Job Parameters**.

Parameter	Mandatory	Type	Description
value	Yes	Object	Parameter value, which must be a string.
type	No	String	Value type, such as **STRING** and **INTEGER**. The value is set by the system.

Table 5-120 extended-configs

Parameter	Mandatory	Type	Description
name	No	String	Extended configuration name. This parameter is unavailable for external systems and does not need to be set.
value	No	String	Extended configuration value. This parameter is unavailable for external systems and does not need to be set.

Response Parameters

Status code: 200

Table 5-121 Response body parameters

Parameter	Type	Description
validation-result	Array of **JobValidationResult** objects	Validation result. - If a job fails to be modified, the failure cause is returned. - If a job is successfully modified, an empty list is returned.

Table 5-122 JobValidationResult

Parameter	Type	Description
message	String	Error description
status	String	Error level, for example, **ERROR** or **WARNING**

Status code: 400

Table 5-123 Response body parameters

Parameter	Type	Description
code	String	Return code
errCode	String	Error code
message	String	Error message
externalMessage	String	Additional information

Example Requests

Modifying a data migration job whose name is **es_css**, source link is an Elasticsearch link, and destination link is a DIS link

```
PUT /v1.1/1551c7f6c808414d8e9f3c514a170f2e/cluster/6ec9a0a4-76be-4262-8697-e7af1fac7920/cdm/job/es_css

{
  "jobs" : [ {
    "job_type" : "NORMAL_JOB",
    "from-connector-name" : "elasticsearch-connector",
    "to-config-values" : {
      "configs" : [ {
        "inputs" : [ {
          "name" : "toJobConfig.streamName",
          "value" : "dis-lkGm"
        }, {
          "name" : "toJobConfig.separator",
          "value" : "|"
        }, {
          "name" : "toJobConfig.columnList",
          "value" : "1&2&3"
        } ],
        "name" : "toJobConfig"
      } ]
    },
    "to-link-name" : "dis",
    "driver-config-values" : {
      "configs" : [ {
        "inputs" : [ {
          "name" : "throttlingConfig.numExtractors",
          "value" : "1"
        }, {
          "name" : "throttlingConfig.submitToCluster",
          "value" : "false"
        }, {
          "name" : "throttlingConfig.numLoaders",
          "value" : "1"
        }, {
          "name" : "throttlingConfig.recordDirtyData",
          "value" : "false"
        } ],
        "name" : "throttlingConfig"
      }, {
        "inputs" : [ ],
        "name" : "jarConfig"
      }, {
        "inputs" : [ {
          "name" : "schedulerConfig.isSchedulerJob",
          "value" : "false"
```



```
      }, {
        "name" : "schedulerConfig.disposableType",
        "value" : "NONE"
      } ],
      "name" : "schedulerConfig"
    }, {
      "inputs" : [ ],
      "name" : "transformConfig"
    }, {
      "inputs" : [ {
        "name" : "retryJobConfig.retryJobType",
        "value" : "NONE"
      } ],
      "name" : "retryJobConfig"
    } ]
  },
  "from-config-values" : {
    "configs" : [ {
      "inputs" : [ {
        "name" : "fromJobConfig.index",
        "value" : "52est"
      }, {
        "name" : "fromJobConfig.type",
        "value" : "est_array"
      }, {
        "name" : "fromJobConfig.columnList",
        "value" : "array_f1_int:long&array_f2_text:string&array_f3_object:nested"
      }, {
        "name" : "fromJobConfig.splitNestedField",
        "value" : "false"
      } ],
      "name" : "fromJobConfig"
    } ]
  },
  "to-connector-name" : "dis-connector",
  "name" : "es_css",
  "from-link-name" : "css"
 } ]
}
```

Example Responses

Status code: 200

Request succeeded.

```
{
 "validation-result" : [ { }, { }, { } ]
}
```

Status code: 400

Error code

```
{
 "code" : "Cdm.0095",
 "errCode" : "Cdm.00095",
 "message" : "A job with the name obs-obs does not exist.",
 "externalMessage" : "A job with the name obs-obs does not exist."
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Modifying a data migration job whose name is **es_css**, source link is an Elasticsearch link, and destination link is a DIS link

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cdm.v1.region.CdmRegion;
import com.huaweicloud.sdk.cdm.v1.*;
import com.huaweicloud.sdk.cdm.v1.model.*;

import java.util.List;
import java.util.ArrayList;

public class UpdateJobSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CdmClient client = CdmClient.newBuilder()
                .withCredential(auth)
                .withRegion(CdmRegion.valueOf("<YOUR REGION>"))
                .build();
        UpdateJobRequest request = new UpdateJobRequest();
        request.withClusterId("{cluster_id}");
        request.withJobName("{job_name}");
        CdmUpdateJobJsonReq body = new CdmUpdateJobJsonReq();
        List<Input> listConfigsInputs = new ArrayList<>();
        listConfigsInputs.add(
            new Input()
                .withName("fromJobConfig.index")
                .withValue("52est")
        );
        listConfigsInputs.add(
            new Input()
                .withName("fromJobConfig.type")
                .withValue("est_array")
        );
        listConfigsInputs.add(
            new Input()
                .withName("fromJobConfig.columnList")
                .withValue("array_f1_int:long&array_f2_text:string&array_f3_object:nested")
        );
        listConfigsInputs.add(
            new Input()
                .withName("fromJobConfig.splitNestedField")
                .withValue("false")
        );
        List<Configs> listFromConfigValuesConfigs = new ArrayList<>();
        listFromConfigValuesConfigs.add(
            new Configs()
                .withInputs(listConfigsInputs)
```



```
        .withName("fromJobConfig")
);
ConfigValues fromconfigvaluesJobs = new ConfigValues();
fromconfigvaluesJobs.withConfigs(listFromConfigValuesConfigs);
List<Input> listConfigsInputs1 = new ArrayList<>();
listConfigsInputs1.add(
    new Input()
        .withName("retryJobConfig.retryJobType")
        .withValue("NONE")
);
List<Input> listConfigsInputs2 = new ArrayList<>();
listConfigsInputs2.add(
    new Input()
        .withName("schedulerConfig.isSchedulerJob")
        .withValue("false")
);
listConfigsInputs2.add(
    new Input()
        .withName("schedulerConfig.disposableType")
        .withValue("NONE")
);
List<Input> listConfigsInputs3 = new ArrayList<>();
listConfigsInputs3.add(
    new Input()
        .withName("throttlingConfig.numExtractors")
        .withValue("1")
);
listConfigsInputs3.add(
    new Input()
        .withName("throttlingConfig.submitToCluster")
        .withValue("false")
);
listConfigsInputs3.add(
    new Input()
        .withName("throttlingConfig.numLoaders")
        .withValue("1")
);
listConfigsInputs3.add(
    new Input()
        .withName("throttlingConfig.recordDirtyData")
        .withValue("false")
);
List<Configs> listDriverConfigValuesConfigs = new ArrayList<>();
listDriverConfigValuesConfigs.add(
    new Configs()
        .withInputs(listConfigsInputs1)
        .withName("retryJobConfig")
);
ConfigValues driverconfigvaluesJobs = new ConfigValues();
driverconfigvaluesJobs.withConfigs(listDriverConfigValuesConfigs);
List<Input> listConfigsInputs4 = new ArrayList<>();
listConfigsInputs4.add(
    new Input()
        .withName("toJobConfig.streamName")
        .withValue("dis-lkGm")
);
listConfigsInputs4.add(
    new Input()
        .withName("toJobConfig.separator")
        .withValue("|")
);
listConfigsInputs4.add(
    new Input()
        .withName("toJobConfig.columnList")
        .withValue("1&2&3")
);
List<Configs> listToConfigValuesConfigs = new ArrayList<>();
listToConfigValuesConfigs.add(
    new Configs()
```



```
            .withInputs(listConfigsInputs4)
            .withName("toJobConfig")
    );
    ConfigValues toconfigvaluesJobs = new ConfigValues();
    toconfigvaluesJobs.withConfigs(listToConfigValuesConfigs);
    List<Job> listbodyJobs = new ArrayList<>();
    listbodyJobs.add(
        new Job()
            .withJobType(Job.JobTypeEnum.fromValue("NORMAL_JOB"))
            .withFromConnectorName("elasticsearch-connector")
            .withToConfigValues(toconfigvaluesJobs)
            .withToLinkName("dis")
            .withDriverConfigValues(driverconfigvaluesJobs)
            .withFromConfigValues(fromconfigvaluesJobs)
            .withToConnectorName("dis-connector")
            .withName("es_css")
            .withFromLinkName("css")
    );
    body.withJobs(listbodyJobs);
    request.withBody(body);
    try {
        UpdateJobResponse response = client.updateJob(request);
        System.out.println(response.toString());
    } catch (ConnectionException e) {
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
  }
}
```

Python

Modifying a data migration job whose name is **es_css**, source link is an Elasticsearch link, and destination link is a DIS link

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcdm.v1.region.cdm_region import CdmRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcdm.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CdmClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CdmRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateJobRequest()
```



```
request.cluster_id = "{cluster_id}"
request.job_name = "{job_name}"
listInputsConfigs = [
    Input(
        name="fromJobConfig.index",
        value="52est"
    ),
    Input(
        name="fromJobConfig.type",
        value="est_array"
    ),
    Input(
        name="fromJobConfig.columnList",
        value="array_f1_int:long&array_f2_text:string&array_f3_object:nested"
    ),
    Input(
        name="fromJobConfig.splitNestedField",
        value="false"
    )
]
listConfigsFromconfigvalues = [
    Configs(
        inputs=listInputsConfigs,
        name="fromJobConfig"
    )
]
fromconfigvaluesJobs = ConfigValues(
    configs=listConfigsFromconfigvalues
)
listInputsConfigs1 = [
    Input(
        name="retryJobConfig.retryJobType",
        value="NONE"
    )
]
listInputsConfigs2 = [
    Input(
        name="schedulerConfig.isSchedulerJob",
        value="false"
    ),
    Input(
        name="schedulerConfig.disposableType",
        value="NONE"
    )
]
listInputsConfigs3 = [
    Input(
        name="throttlingConfig.numExtractors",
        value="1"
    ),
    Input(
        name="throttlingConfig.submitToCluster",
        value="false"
    ),
    Input(
        name="throttlingConfig.numLoaders",
        value="1"
    ),
    Input(
        name="throttlingConfig.recordDirtyData",
        value="false"
    )
]
listConfigsDriverconfigvalues = [
    Configs(
        inputs=listInputsConfigs1,
        name="retryJobConfig"
    )
]
```



```
        driverconfigvaluesJobs = ConfigValues(
            configs=listConfigsDriverconfigvalues
        )
        listInputsConfigs4 = [
            Input(
                name="toJobConfig.streamName",
                value="dis-lkGm"
            ),
            Input(
                name="toJobConfig.separator",
                value="|"
            ),
            Input(
                name="toJobConfig.columnList",
                value="1&2&3"
            )
        ]
        listConfigsToconfigvalues = [
            Configs(
                inputs=listInputsConfigs4,
                name="toJobConfig"
            )
        ]
        toconfigvaluesJobs = ConfigValues(
            configs=listConfigsToconfigvalues
        )
        listJobsbody = [
            Job(
                job_type="NORMAL_JOB",
                from_connector_name="elasticsearch-connector",
                to_config_values=toconfigvaluesJobs,
                to_link_name="dis",
                driver_config_values=driverconfigvaluesJobs,
                from_config_values=fromconfigvaluesJobs,
                to_connector_name="dis-connector",
                name="es_css",
                from_link_name="css"
            )
        ]
        request.body = CdmUpdateJobJsonReq(
            jobs=listJobsbody
        )
        response = client.update_job(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

Modifying a data migration job whose name is **es_css**, source link is an Elasticsearch link, and destination link is a DIS link

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cdm "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
```



```
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cdm.NewCdmClient(
        cdm.CdmClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateJobRequest{}
    request.ClusterId = "{cluster_id}"
    request.JobName = "{job_name}"
    valueInputs:= "52est"
    var valueInputsInterface interface{} = valueInputs
    valueInputs1:= "est_array"
    var valueInputsInterface1 interface{} = valueInputs1
    valueInputs2:= "array_f1_int:long&array_f2_text:string&array_f3_object:nested"
    var valueInputsInterface2 interface{} = valueInputs2
    valueInputs3:= "false"
    var valueInputsInterface3 interface{} = valueInputs3
    var listInputsConfigs = []model.Input{
        {
            Name: "fromJobConfig.index",
            Value: &valueInputsInterface,
        },
        {
            Name: "fromJobConfig.type",
            Value: &valueInputsInterface1,
        },
        {
            Name: "fromJobConfig.columnList",
            Value: &valueInputsInterface2,
        },
        {
            Name: "fromJobConfig.splitNestedField",
            Value: &valueInputsInterface3,
        },
    }
    var listConfigsFromConfigValues = []model.Configs{
        {
            Inputs: listInputsConfigs,
            Name: "fromJobConfig",
        },
    }
    fromconfigvaluesJobs := &model.ConfigValues{
        Configs: listConfigsFromConfigValues,
    }
    valueInputs4:= "NONE"
    var valueInputsInterface4 interface{} = valueInputs4
    var listInputsConfigs1 = []model.Input{
        {
            Name: "retryJobConfig.retryJobType",
            Value: &valueInputsInterface4,
        },
    }
    valueInputs5:= "false"
    var valueInputsInterface5 interface{} = valueInputs5
    valueInputs6:= "NONE"
    var valueInputsInterface6 interface{} = valueInputs6
    var listInputsConfigs2 = []model.Input{
```



```
    {
        Name: "schedulerConfig.isSchedulerJob",
        Value: &valueInputsInterface5,
    },
    {
        Name: "schedulerConfig.disposableType",
        Value: &valueInputsInterface6,
    },
}
valueInputs7:= "1"
var valueInputsInterface7 interface{} = valueInputs7
valueInputs8:= "false"
var valueInputsInterface8 interface{} = valueInputs8
valueInputs9:= "1"
var valueInputsInterface9 interface{} = valueInputs9
valueInputs10:= "false"
var valueInputsInterface10 interface{} = valueInputs10
var listInputsConfigs3 = []model.Input{
    {
        Name: "throttlingConfig.numExtractors",
        Value: &valueInputsInterface7,
    },
    {
        Name: "throttlingConfig.submitToCluster",
        Value: &valueInputsInterface8,
    },
    {
        Name: "throttlingConfig.numLoaders",
        Value: &valueInputsInterface9,
    },
    {
        Name: "throttlingConfig.recordDirtyData",
        Value: &valueInputsInterface10,
    },
}
var listConfigsDriverConfigValues = []model.Configs{
    {
        Inputs: listInputsConfigs1,
        Name: "retryJobConfig",
    },
}
driverconfigvaluesJobs := &model.ConfigValues{
    Configs: listConfigsDriverConfigValues,
}
valueInputs11:= "dis-lkGm"
var valueInputsInterface11 interface{} = valueInputs11
valueInputs12:= "|"
var valueInputsInterface12 interface{} = valueInputs12
valueInputs13:= "1&2&3"
var valueInputsInterface13 interface{} = valueInputs13
var listInputsConfigs4 = []model.Input{
    {
        Name: "toJobConfig.streamName",
        Value: &valueInputsInterface11,
    },
    {
        Name: "toJobConfig.separator",
        Value: &valueInputsInterface12,
    },
    {
        Name: "toJobConfig.columnList",
        Value: &valueInputsInterface13,
    },
}
var listConfigsToConfigValues = []model.Configs{
    {
        Inputs: listInputsConfigs4,
        Name: "toJobConfig",
    },
```



```
    }
    toconfigvaluesJobs := &model.ConfigValues{
        Configs: listConfigsToConfigValues,
    }
    var listJobsbody = []model.Job{
        {
            JobType: model.GetJobJobTypeEnum().NORMAL_JOB,
            FromConnectorName: "elasticsearch-connector",
            ToConfigValues: toconfigvaluesJobs,
            ToLinkName: "dis",
            DriverConfigValues: driverconfigvaluesJobs,
            FromConfigValues: fromconfigvaluesJobs,
            ToConnectorName: "dis-connector",
            Name: "es_css",
            FromLinkName: "css",
        },
    }
    request.Body = &model.CdmUpdateJobJsonReq{
        Jobs: listJobsbody,
    }
    response, err := client.UpdateJob(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in **API Explorer**. SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Request succeeded.
400	Error code

Error Codes

See **Error Codes**.

5.2.4 Creating and Executing a Job in a Random Cluster

Function

This API is used to create and execute a job in a random cluster.

Calling Method

For details, see **Calling APIs**.

URI

POST /v1.1/{project_id}/clusters/job

Table 5-124 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain it, see **Project ID and Account ID**.

Request Parameters

Table 5-125 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token. It can be obtained by calling the IAM API (value of X-Subject-Token in the response header).
X-Language	Yes	String	Request language

Table 5-126 Request body parameters

Parameter	Mandatory	Type	Description
jobs	Yes	Array of **Job** objects	Job list. For details, see the descriptions of **jobs** parameters.
clusters	Yes	Array of strings	IDs of CDM clusters. The system selects a random cluster in running state from the specified clusters and creates and executes a migration job in the cluster.

Table 5-127 Job

Parameter	Mandatory	Type	Description
job_type	Yes	String	Job type • **NORMAL_JOB**: table/file migration • **BATCH_JOB**: entire DB migration • **SCENARIO_JOB**: scenario migration
from-connector-name	Yes	String	Source link type. The corresponding link parameters are as follows: • **generic-jdbc-connector**: link to relational database • **obs-connector**: link to OBS • **hdfs-connector**: link to HDFS • **hbase-connector**: link to HBase and link to CloudTable • **hive-connector**: link to Hive • **ftp-connector/sftp-connector**: link to an FTP or SFTP server • **mongodb-connector**: link to MongoDB • **redis-connector**: link to Redis/DCS • **kafka-connector**: link to Kafka • **dis-connector**: link to DIS • **elasticsearch-connector**: link to Elasticsearch/Cloud Search Service (CSS) • [**dli-connector**: link to DLI](tag:nohcs) • **http-connector**: link to HTTP/HTTPS servers (No link parameters are required.) • **dms-kafka-connector**: link to DMS Kafka

Parameter	Mandatory	Type	Description
to-config-values	Yes	**ConfigValues** object	Destination link parameters, which vary depending on the destination. For details, see **Destination Job Parameters**.
to-link-name	Yes	String	Name of the destination link, that is, the name of the link created through the API used to create a link
driver-config-values	Yes	**ConfigValues** object	Job parameters, such as **Retry upon Failure** and **Concurrent Extractors**. For details, see **Job Parameter Description**.
from-config-values	Yes	**ConfigValues** object	Source link parameters, which vary depending on the source. For details, see **Source Job Parameters**.

Parameter	Mandatory	Type	Description
to-connector-name	Yes	String	Destination link type. The corresponding link parameters are as follows: • **generic-jdbc-connector**: link to relational database • **obs-connector**: link to OBS • **hdfs-connector**: link to HDFS • **hbase-connector**: link to HBase and link to CloudTable • **hive-connector**: link to Hive • **ftp-connector/sftp-connector**: link to an FTP or SFTP server • **mongodb-connector**: link to MongoDB • **redis-connector**: link to Redis/DCS • **kafka-connector**: link to Kafka • **dis-connector**: link to DIS • **elasticsearch-connector**: link to Elasticsearch/Cloud Search Service (CSS) • [**dli-connector**: link to DLI](tag:nohcs) • **http-connector**: link to HTTP/HTTPS servers (No link parameters are required.) • **dms-kafka-connector**: link to DMS Kafka
name	Yes	String	Job name, which contains 1 to 240 characters
from-link-name	Yes	String	Name of the source link, that is, the name of the link created through the API used to create a link
creation-user	No	String	User who created the job. The value is generated by the system.

Parameter	Mandatory	Type	Description
creation-date	No	Long	Time when the job was created, accurate to millisecond. The value is generated by the system.
update-date	No	Long	Time when the job was last updated, accurate to millisecond. The value is generated by the system.
is_incre_job	No	Boolean	Whether the job is an incremental job. This parameter has been discarded.
flag	No	Integer	Whether the job is a scheduled job. If yes, the value is **1**. Otherwise, the value is **0**. The value is generated by the system based on the scheduled task configuration.
files_read	No	Integer	Number of read files. The value is generated by the system.
update-user	No	String	User who last updated the job. The value is generated by the system.
external_id	No	String	ID of the job to be executed. For a local job, the value is in the format of **job_local1202051771_0002** . For a DLI job, the value is the DLI job ID, for example, **"12345"**. The value is generated by the system and does not need to be set.
type	No	String	Job type. The value of this parameter is the same as that of **job_type**. The options are as follows: • **NORMAL_JOB**: table/file migration job • **BATCH_JOB**: entire DB migration job • **SCENARIO_JOB**: scenario migration job

Parameter	Mandatory	Type	Description
execute_start_date	No	Long	Time when the last task was started, accurate to millisecond. The value is generated by the system.
delete_rows	No	Integer	Number of rows deleted by an incremental job. This parameter is deprecated.
enabled	No	Boolean	Whether the link is enabled. The value is generated by the system.
bytes_written	No	Long	Number of bytes written by the job. The value is generated by the system.
id	No	Integer	Job ID, which is generated by the system
is_use_sql	No	Boolean	Whether to use SQL statements. The value is generated by the system based on whether SQL statements are used for source data extraction. You do not need to set this parameter.
update_rows	No	Integer	Number of updated rows in an incremental job. This parameter is deprecated.
group_name	No	String	Group name
bytes_read	No	Long	Number of bytes read by the job. The value is generated by the system.
execute_update_date	No	Long	Time when the last task was updated, accurate to millisecond. The value is generated by the system.
write_rows	No	Integer	Number of rows written by an incremental job. This parameter is deprecated.
rows_written	No	Integer	Number of rows written by the job. The value is generated by the system.
rows_read	No	Long	Number of rows read by the job. The value is generated by the system.

Parameter	Mandatory	Type	Description
files_written	No	Integer	Number of written files. The value is generated by the system.
is_incrementing	No	Boolean	Whether the job is an incremental job. Similar to parameter **is_incre_job**, this parameter is deprecated.
execute_create_date	No	Long	Time when the last task was created, accurate to millisecond. The value is generated by the system.
status	No	String	Job execution status • **BOOTING**: The job is starting. • **RUNNING**: The job is running. • **SUCCEEDED**: The job was successfully executed. • **FAILED**: The job execution failed. • **NEW**: The job was not executed.

Table 5-128 ConfigValues

Parameter	Mandatory	Type	Description
configs	Yes	Array of **configs** objects	The data structures of source link parameters, destination link parameters, and job parameters are the same. However, the **inputs** parameter varies. For details, see the descriptions of **configs** parameters.
extended-configs	No	**extended-configs** object	Extended configuration. For details, see the descriptions of **extended-configs** parameters. The extended configuration is not open to external systems. You do not need to set it.

Table 5-129 configs

Parameter	Mandatory	Type	Description
inputs	Yes	Array of **Input** objects	Input parameter list. Each element in the list is in *name,value* format. For details, see the descriptions of **inputs** parameters. In the **from-config-values** data structure, the value of this parameter varies with the source link type. For details, see section "Source Job Parameters" in the *Cloud Data Migration User Guide*. In the **to-config-values** data structure, the value of this parameter varies with the destination link type. For details, see section "Destination Job Parameters" in the *Cloud Data Migration User Guide*. For details about the **inputs** parameter in the **driver-config-values** data structure, see the job parameter descriptions.
name	Yes	String	Configuration name. The value is **fromJobConfig** for a source job, **toJobConfig** for a destination job, and **linkConfig** for a link.
id	No	Integer	Configuration ID, which is generated by the system. You do not need to set this parameter.
type	No	String	Configuration type, which is generated by the system. You do not need to set this parameter. The value can be **LINK** (for link management APIs) or **JOB** (for job management APIs).

Table 5-130 Input

Parameter	Mandatory	Type	Description
name	Yes	String	Parameter name. • For link management APIs, parameter names start with **linkConfig.**. The parameters vary depending on the link type. For details, see the parameter descriptions of the corresponding link in **Link Parameters**. • For job management APIs, source link parameter names start with **fromJobConfig.**. For details, see the source job parameters in **Source Job Parameters**. Destination link parameter names start with **toJobConfig.**. For details, see **Destination Job Parameters**. For details about job parameters, see the task parameter descriptions in **Job Parameters**.
value	Yes	Object	Parameter value, which must be a string.
type	No	String	Value type, such as **STRING** and **INTEGER**. The value is set by the system.

Table 5-131 extended-configs

Parameter	Mandatory	Type	Description
name	No	String	Extended configuration name. This parameter is unavailable for external systems and does not need to be set.
value	No	String	Extended configuration value. This parameter is unavailable for external systems and does not need to be set.

Response Parameters

Status code: 200

Table 5-132 Response body parameters

Parameter	Type	Description
submissions	Array of StartJobSubmission objects	Job running information. For details, see the descriptions of **submission** parameters.

Table 5-133 StartJobSubmission

Parameter	Type	Description
isIncrementing	Boolean	Whether the job migrates incremental data
delete_rows	Integer	Number of deleted rows
update_rows	Integer	Number of updated rows
write_rows	Integer	Number of written rows
submission-id	Integer	Job submission ID
job-name	String	Job name
creation-user	String	User who started the job
creation-date	Long	Job creation time, accurate to millisecond
execute-date	Long	Execution time
progress	Float	Job progress. If a job fails, the value is **-1**. Otherwise, the value ranges from 0 to 100.
status	String	Job status • **BOOTING**: The job is starting. • **FAILURE_ON_SUBMIT**: The job failed to be submitted. • **RUNNING**: The job is running. • **SUCCEEDED**: The job was executed successfully. • **FAILED**: The job failed. • **UNKNOWN**: The job status is unknown. • **NEVER_EXECUTED**: The job was not executed.

Parameter	Type	Description
isStopingIncrement	String	Whether to stop incremental data migration
is-execute-auto	Boolean	Whether to execute the job as scheduled
last-update-date	Long	Time when the job was last updated
last-udpate-user	String	User who last updated the job status
isDeleteJob	Boolean	Whether to delete the job after it is executed

Example Requests

Randomly selecting a CDM cluster and creating a job named **es_css** to migrate tables from Elasticsearch to DIS

```
POST /v1.1/1551c7f6c808414d8e9f3c514a170f2e/clusters/job

{
  "jobs" : [ {
    "job_type" : "NORMAL_JOB",
    "from-connector-name" : "elasticsearch-connector",
    "to-config-values" : {
      "configs" : [ {
        "inputs" : [ {
          "name" : "toJobConfig.streamName",
          "value" : "dis-lkGm"
        }, {
          "name" : "toJobConfig.separator",
          "value" : "|"
        }, {
          "name" : "toJobConfig.columnList",
          "value" : "1&2&3"
        } ],
        "name" : "toJobConfig"
      } ]
    },
    "to-link-name" : "dis",
    "driver-config-values" : {
      "configs" : [ {
        "inputs" : [ {
          "name" : "throttlingConfig.numExtractors",
          "value" : "1"
        }, {
          "name" : "throttlingConfig.submitToCluster",
          "value" : "false"
        }, {
          "name" : "throttlingConfig.numLoaders",
          "value" : "1"
        }, {
          "name" : "throttlingConfig.recordDirtyData",
          "value" : "false"
        } ],
        "name" : "throttlingConfig"
      }, {
        "inputs" : [ ],
        "name" : "jarConfig"
      }, {
```



```
      "inputs" : [ {
        "name" : "schedulerConfig.isSchedulerJob",
        "value" : "false"
      }, {
        "name" : "schedulerConfig.disposableType",
        "value" : "NONE"
      } ],
      "name" : "schedulerConfig"
    }, {
      "inputs" : [ ],
      "name" : "transformConfig"
    }, {
      "inputs" : [ {
        "name" : "retryJobConfig.retryJobType",
        "value" : "NONE"
      } ],
      "name" : "retryJobConfig"
    } ]
  },
  "from-config-values" : {
    "configs" : [ {
      "inputs" : [ {
        "name" : "fromJobConfig.index",
        "value" : "52est"
      }, {
        "name" : "fromJobConfig.type",
        "value" : "est_array"
      }, {
        "name" : "fromJobConfig.columnList",
        "value" : "array_f1_int:long&array_f2_text:string&array_f3_object:nested"
      }, {
        "name" : "fromJobConfig.splitNestedField",
        "value" : "false"
      } ],
      "name" : "fromJobConfig"
    } ]
  },
  "to-connector-name" : "dis-connector",
  "name" : "es_css",
  "from-link-name" : "css"
 } ],
 "clusters" : [ "b0791496-e111-4e75-b7ca-9277aeab9297", "c2db1191-eb6c-464a-a0d3-b434e6c6df26",
"c2db1191-eb6c-464a-a0d3-b434e6c6df26" ]
}
```

Example Responses

Status code: 200

Request succeeded.

```
{
 "submissions" : [ {
  "isIncrementing" : false,
  "job-name" : "obs2obs-03",
  "submission-id" : 13,
  "isStopingIncrement" : "",
  "last-update-date" : 1635909057030,
  "is-execute-auto" : false,
  "delete_rows" : 0,
  "write_rows" : 0,
  "isDeleteJob" : false,
  "creation-user" : "cdmUser",
  "progress" : 0,
  "creation-date" : 1635909057030,
  "update_rows" : 0,
  "status" : "PENDING",
  "execute-date" : 1635909057030
```



```
 }]
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Randomly selecting a CDM cluster and creating a job named **es_css** to migrate tables from Elasticsearch to DIS

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cdm.v1.region.CdmRegion;
import com.huaweicloud.sdk.cdm.v1.*;
import com.huaweicloud.sdk.cdm.v1.model.*;

import java.util.List;
import java.util.ArrayList;

public class CreateAndStartRandomClusterJobSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CdmClient client = CdmClient.newBuilder()
                .withCredential(auth)
                .withRegion(CdmRegion.valueOf("<YOUR REGION>"))
                .build();
        CreateAndStartRandomClusterJobRequest request = new CreateAndStartRandomClusterJobRequest();
        CdmRandomCreateAndStartJobJsonReq body = new CdmRandomCreateAndStartJobJsonReq();
        List<String> listbodyClusters = new ArrayList<>();
        listbodyClusters.add("b0791496-e111-4e75-b7ca-9277aeab9297");
        listbodyClusters.add("c2db1191-eb6c-464a-a0d3-b434e6c6df26");
        listbodyClusters.add("c2db1191-eb6c-464a-a0d3-b434e6c6df26");
        List<Input> listConfigsInputs = new ArrayList<>();
        listConfigsInputs.add(
            new Input()
                .withName("fromJobConfig.index")
                .withValue("52est")
        );
        listConfigsInputs.add(
            new Input()
                .withName("fromJobConfig.type")
                .withValue("est_array")
        );
        listConfigsInputs.add(
            new Input()
                .withName("fromJobConfig.columnList")
```



```
        .withValue("array_f1_int:long&array_f2_text:string&array_f3_object:nested")
);
listConfigsInputs.add(
    new Input()
        .withName("fromJobConfig.splitNestedField")
        .withValue("false")
);
List<Configs> listFromConfigValuesConfigs = new ArrayList<>();
listFromConfigValuesConfigs.add(
    new Configs()
        .withInputs(listConfigsInputs)
        .withName("fromJobConfig")
);
ConfigValues fromconfigvaluesJobs = new ConfigValues();
fromconfigvaluesJobs.withConfigs(listFromConfigValuesConfigs);
List<Input> listConfigsInputs1 = new ArrayList<>();
listConfigsInputs1.add(
    new Input()
        .withName("retryJobConfig.retryJobType")
        .withValue("NONE")
);
List<Input> listConfigsInputs2 = new ArrayList<>();
listConfigsInputs2.add(
    new Input()
        .withName("schedulerConfig.isSchedulerJob")
        .withValue("false")
);
listConfigsInputs2.add(
    new Input()
        .withName("schedulerConfig.disposableType")
        .withValue("NONE")
);
List<Input> listConfigsInputs3 = new ArrayList<>();
listConfigsInputs3.add(
    new Input()
        .withName("throttlingConfig.numExtractors")
        .withValue("1")
);
listConfigsInputs3.add(
    new Input()
        .withName("throttlingConfig.submitToCluster")
        .withValue("false")
);
listConfigsInputs3.add(
    new Input()
        .withName("throttlingConfig.numLoaders")
        .withValue("1")
);
listConfigsInputs3.add(
    new Input()
        .withName("throttlingConfig.recordDirtyData")
        .withValue("false")
);
List<Configs> listDriverConfigValuesConfigs = new ArrayList<>();
listDriverConfigValuesConfigs.add(
    new Configs()
        .withInputs(listConfigsInputs1)
        .withName("retryJobConfig")
);
ConfigValues driverconfigvaluesJobs = new ConfigValues();
driverconfigvaluesJobs.withConfigs(listDriverConfigValuesConfigs);
List<Input> listConfigsInputs4 = new ArrayList<>();
listConfigsInputs4.add(
    new Input()
        .withName("toJobConfig.streamName")
        .withValue("dis-lkGm")
);
listConfigsInputs4.add(
    new Input()
```



```
                .withName("toJobConfig.separator")
                .withValue("|")
        );
        listConfigsInputs4.add(
            new Input()
                .withName("toJobConfig.columnList")
                .withValue("1&2&3")
        );
        List<Configs> listToConfigValuesConfigs = new ArrayList<>();
        listToConfigValuesConfigs.add(
            new Configs()
                .withInputs(listConfigsInputs4)
                .withName("toJobConfig")
        );
        ConfigValues toconfigvaluesJobs = new ConfigValues();
        toconfigvaluesJobs.withConfigs(listToConfigValuesConfigs);
        List<Job> listbodyJobs = new ArrayList<>();
        listbodyJobs.add(
            new Job()
                .withJobType(Job.JobTypeEnum.fromValue("NORMAL_JOB"))
                .withFromConnectorName("elasticsearch-connector")
                .withToConfigValues(toconfigvaluesJobs)
                .withToLinkName("dis")
                .withDriverConfigValues(driverconfigvaluesJobs)
                .withFromConfigValues(fromconfigvaluesJobs)
                .withToConnectorName("dis-connector")
                .withName("es_css")
                .withFromLinkName("css")
        );
        body.withClusters(listbodyClusters);
        body.withJobs(listbodyJobs);
        request.withBody(body);
        try {
            CreateAndStartRandomClusterJobResponse response =
client.createAndStartRandomClusterJob(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

Randomly selecting a CDM cluster and creating a job named **es_css** to migrate tables from Elasticsearch to DIS

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcdm.v1.region.cdm_region import CdmRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcdm.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
```



```
ak = os.environ["CLOUD_SDK_AK"]
sk = os.environ["CLOUD_SDK_SK"]
projectId = "{project_id}"

credentials = BasicCredentials(ak, sk, projectId)

client = CdmClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(CdmRegion.value_of("<YOUR REGION>")) \
    .build()

try:
    request = CreateAndStartRandomClusterJobRequest()
    listClustersbody = [
        "b0791496-e111-4e75-b7ca-9277aeab9297",
        "c2db1191-eb6c-464a-a0d3-b434e6c6df26",
        "c2db1191-eb6c-464a-a0d3-b434e6c6df26"
    ]
    listInputsConfigs = [
        Input(
            name="fromJobConfig.index",
            value="52est"
        ),
        Input(
            name="fromJobConfig.type",
            value="est_array"
        ),
        Input(
            name="fromJobConfig.columnList",
            value="array_f1_int:long&array_f2_text:string&array_f3_object:nested"
        ),
        Input(
            name="fromJobConfig.splitNestedField",
            value="false"
        )
    ]
    listConfigsFromconfigvalues = [
        Configs(
            inputs=listInputsConfigs,
            name="fromJobConfig"
        )
    ]
    fromconfigvaluesJobs = ConfigValues(
        configs=listConfigsFromconfigvalues
    )
    listInputsConfigs1 = [
        Input(
            name="retryJobConfig.retryJobType",
            value="NONE"
        )
    ]
    listInputsConfigs2 = [
        Input(
            name="schedulerConfig.isSchedulerJob",
            value="false"
        ),
        Input(
            name="schedulerConfig.disposableType",
            value="NONE"
        )
    ]
    listInputsConfigs3 = [
        Input(
            name="throttlingConfig.numExtractors",
            value="1"
        ),
        Input(
            name="throttlingConfig.submitToCluster",
            value="false"
```



```
            ),
            Input(
                name="throttlingConfig.numLoaders",
                value="1"
            ),
            Input(
                name="throttlingConfig.recordDirtyData",
                value="false"
            )
        ]
        listConfigsDriverconfigvalues = [
            Configs(
                inputs=listInputsConfigs1,
                name="retryJobConfig"
            )
        ]
        driverconfigvaluesJobs = ConfigValues(
            configs=listConfigsDriverconfigvalues
        )
        listInputsConfigs4 = [
            Input(
                name="toJobConfig.streamName",
                value="dis-lkGm"
            ),
            Input(
                name="toJobConfig.separator",
                value="|"
            ),
            Input(
                name="toJobConfig.columnList",
                value="1&2&3"
            )
        ]
        listConfigsToconfigvalues = [
            Configs(
                inputs=listInputsConfigs4,
                name="toJobConfig"
            )
        ]
        toconfigvaluesJobs = ConfigValues(
            configs=listConfigsToconfigvalues
        )
        listJobsbody = [
            Job(
                job_type="NORMAL_JOB",
                from_connector_name="elasticsearch-connector",
                to_config_values=toconfigvaluesJobs,
                to_link_name="dis",
                driver_config_values=driverconfigvaluesJobs,
                from_config_values=fromconfigvaluesJobs,
                to_connector_name="dis-connector",
                name="es_css",
                from_link_name="css"
            )
        ]
        request.body = CdmRandomCreateAndStartJobJsonReq(
            clusters=listClustersbody,
            jobs=listJobsbody
        )
        response = client.create_and_start_random_cluster_job(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```


Go

Randomly selecting a CDM cluster and creating a job named **es_css** to migrate tables from Elasticsearch to DIS

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cdm "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cdm.NewCdmClient(
        cdm.CdmClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreateAndStartRandomClusterJobRequest{}
    var listClustersbody = []string{
        "b0791496-e111-4e75-b7ca-9277aeab9297",
        "c2db1191-eb6c-464a-a0d3-b434e6c6df26",
        "c2db1191-eb6c-464a-a0d3-b434e6c6df26",
    }
    valueInputs:= "52est"
    var valueInputsInterface interface{} = valueInputs
    valueInputs1:= "est_array"
    var valueInputsInterface1 interface{} = valueInputs1
    valueInputs2:= "array_f1_int:long&array_f2_text:string&array_f3_object:nested"
    var valueInputsInterface2 interface{} = valueInputs2
    valueInputs3:= "false"
    var valueInputsInterface3 interface{} = valueInputs3
    var listInputsConfigs = []model.Input{
        {
            Name: "fromJobConfig.index",
            Value: &valueInputsInterface,
        },
        {
            Name: "fromJobConfig.type",
            Value: &valueInputsInterface1,
        },
        {
            Name: "fromJobConfig.columnList",
            Value: &valueInputsInterface2,
        },
        {
            Name: "fromJobConfig.splitNestedField",
            Value: &valueInputsInterface3,
        },
    }
```



```
var listConfigsFromConfigValues = []model.Configs{
    {
        Inputs: listInputsConfigs,
        Name: "fromJobConfig",
    },
}
fromconfigvaluesJobs := &model.ConfigValues{
    Configs: listConfigsFromConfigValues,
}
valueInputs4:= "NONE"
var valueInputsInterface4 interface{} = valueInputs4
var listInputsConfigs1 = []model.Input{
    {
        Name: "retryJobConfig.retryJobType",
        Value: &valueInputsInterface4,
    },
}
valueInputs5:= "false"
var valueInputsInterface5 interface{} = valueInputs5
valueInputs6:= "NONE"
var valueInputsInterface6 interface{} = valueInputs6
var listInputsConfigs2 = []model.Input{
    {
        Name: "schedulerConfig.isSchedulerJob",
        Value: &valueInputsInterface5,
    },
    {
        Name: "schedulerConfig.disposableType",
        Value: &valueInputsInterface6,
    },
}
valueInputs7:= "1"
var valueInputsInterface7 interface{} = valueInputs7
valueInputs8:= "false"
var valueInputsInterface8 interface{} = valueInputs8
valueInputs9:= "1"
var valueInputsInterface9 interface{} = valueInputs9
valueInputs10:= "false"
var valueInputsInterface10 interface{} = valueInputs10
var listInputsConfigs3 = []model.Input{
    {
        Name: "throttlingConfig.numExtractors",
        Value: &valueInputsInterface7,
    },
    {
        Name: "throttlingConfig.submitToCluster",
        Value: &valueInputsInterface8,
    },
    {
        Name: "throttlingConfig.numLoaders",
        Value: &valueInputsInterface9,
    },
    {
        Name: "throttlingConfig.recordDirtyData",
        Value: &valueInputsInterface10,
    },
}
var listConfigsDriverConfigValues = []model.Configs{
    {
        Inputs: listInputsConfigs1,
        Name: "retryJobConfig",
    },
}
driverconfigvaluesJobs := &model.ConfigValues{
    Configs: listConfigsDriverConfigValues,
}
valueInputs11:= "dis-lkGm"
var valueInputsInterface11 interface{} = valueInputs11
valueInputs12:= "|"
```



```
    var valueInputsInterface12 interface{} = valueInputs12
    valueInputs13:= "1&2&3"
    var valueInputsInterface13 interface{} = valueInputs13
    var listInputsConfigs4 = []model.Input{
        {
            Name: "toJobConfig.streamName",
            Value: &valueInputsInterface11,
        },
        {
            Name: "toJobConfig.separator",
            Value: &valueInputsInterface12,
        },
        {
            Name: "toJobConfig.columnList",
            Value: &valueInputsInterface13,
        },
    }
    var listConfigsToConfigValues = []model.Configs{
        {
            Inputs: listInputsConfigs4,
            Name: "toJobConfig",
        },
    }
    toconfigvaluesJobs := &model.ConfigValues{
        Configs: listConfigsToConfigValues,
    }
    var listJobsbody = []model.Job{
        {
            JobType: model.GetJobJobTypeEnum().NORMAL_JOB,
            FromConnectorName: "elasticsearch-connector",
            ToConfigValues: toconfigvaluesJobs,
            ToLinkName: "dis",
            DriverConfigValues: driverconfigvaluesJobs,
            FromConfigValues: fromconfigvaluesJobs,
            ToConnectorName: "dis-connector",
            Name: "es_css",
            FromLinkName: "css",
        },
    }
    request.Body = &model.CdmRandomCreateAndStartJobJsonReq{
        Clusters: listClustersbody,
        Jobs: listJobsbody,
    }
    response, err := client.CreateAndStartRandomClusterJob(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in **API Explorer**. SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Request succeeded.

Error Codes

See **Error Codes**.

5.2.5 Stopping a Job

Function

This API is used to stop a job.

Calling Method

For details, see **Calling APIs**.

URI

PUT /v1.1/{project_id}/clusters/{cluster_id}/cdm/job/{job_name}/stop

Table 5-134 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain it, see **Project ID and Account ID**.
cluster_id	Yes	String	Cluster ID
job_name	Yes	String	Job name

Request Parameters

Table 5-135 Request header parameters

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

Response Parameters

Status code: 200

Table 5-136 Response body parameters

Parameter	Type	Description
validation-result	Array of **JobValidationResult** objects	Validation result. If the job fails to be stopped, the failure cause is returned. For details, see the **validation-result** parameter. If the job is stopped, an empty list is returned.

Table 5-137 JobValidationResult

Parameter	Type	Description
message	String	Error description
status	String	Error level, for example, **ERROR** or **WARNING**

Example Requests

```
PUT /v1.1/1551c7f6c808414d8e9f3c514a170f2e/clusters/6ec9a0a4-76be-4262-8697-e7af1fac7920/cdm/job/jdbc2hive/stop
```

Example Responses

Status code: 200

Request succeeded.

```
{ }
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cdm.v1.region.CdmRegion;
import com.huaweicloud.sdk.cdm.v1.*;
import com.huaweicloud.sdk.cdm.v1.model.*;


public class StopJobSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
```



```
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CdmClient client = CdmClient.newBuilder()
                .withCredential(auth)
                .withRegion(CdmRegion.valueOf("<YOUR REGION>"))
                .build();
        StopJobRequest request = new StopJobRequest();
        request.withClusterId("{cluster_id}");
        request.withJobName("{job_name}");
        try {
            StopJobResponse response = client.stopJob(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcdm.v1.region.cdm_region import CdmRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcdm.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CdmClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CdmRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = StopJobRequest()
        request.cluster_id = "{cluster_id}"
        request.job_name = "{job_name}"
        response = client.stop_job(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```


Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cdm "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cdm.NewCdmClient(
        cdm.CdmClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.StopJobRequest{}
    request.ClusterId = "{cluster_id}"
    request.JobName = "{job_name}"
    response, err := client.StopJob(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in **API Explorer**. SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Request succeeded.

Error Codes

See **Error Codes**.

5.2.6 Creating a Job in a Specified Cluster

Function

This API is used to create a job in a specified cluster.

Calling Method

For details, see **Calling APIs**.

URI

POST /v1.1/{project_id}/clusters/{cluster_id}/cdm/job

Table 5-138 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain it, see **Project ID and Account ID**.
cluster_id	Yes	String	Cluster ID

Request Parameters

Table 5-139 Request header parameters

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

Table 5-140 Request body parameters

Parameter	Mandatory	Type	Description
jobs	Yes	Array of **Job** objects	Job list. For details, see the descriptions of **jobs** parameters.

Table 5-141 Job

Parameter	Mandatory	Type	Description
job_type	Yes	String	Job type • **NORMAL_JOB**: table/file migration • **BATCH_JOB**: entire DB migration • **SCENARIO_JOB**: scenario migration
from-connector-name	Yes	String	Source link type. The corresponding link parameters are as follows: • **generic-jdbc-connector**: link to relational database • **obs-connector**: link to OBS • **hdfs-connector**: link to HDFS • **hbase-connector**: link to HBase and link to CloudTable • **hive-connector**: link to Hive • **ftp-connector/sftp-connector**: link to an FTP or SFTP server • **mongodb-connector**: link to MongoDB • **redis-connector**: link to Redis/DCS • **kafka-connector**: link to Kafka • **dis-connector**: link to DIS • **elasticsearch-connector**: link to Elasticsearch/Cloud Search Service (CSS) • [**dli-connector**: link to DLI](tag:nohcs) • **http-connector**: link to HTTP/HTTPS servers (No link parameters are required.) • **dms-kafka-connector**: link to DMS Kafka

Parameter	Mandatory	Type	Description
to-config-values	Yes	**ConfigValues** object	Destination link parameters, which vary depending on the destination. For details, see **Destination Job Parameters**.
to-link-name	Yes	String	Name of the destination link, that is, the name of the link created through the API used to create a link
driver-config-values	Yes	**ConfigValues** object	Job parameters, such as **Retry upon Failure** and **Concurrent Extractors**. For details, see **Job Parameter Description**.
from-config-values	Yes	**ConfigValues** object	Source link parameters, which vary depending on the source. For details, see **Source Job Parameters**.

Parameter	Mandatory	Type	Description
to-connector-name	Yes	String	Destination link type. The corresponding link parameters are as follows: • **generic-jdbc-connector**: link to relational database • **obs-connector**: link to OBS • **hdfs-connector**: link to HDFS • **hbase-connector**: link to HBase and link to CloudTable • **hive-connector**: link to Hive • **ftp-connector/sftp-connector**: link to an FTP or SFTP server • **mongodb-connector**: link to MongoDB • **redis-connector**: link to Redis/DCS • **kafka-connector**: link to Kafka • **dis-connector**: link to DIS • **elasticsearch-connector**: link to Elasticsearch/Cloud Search Service (CSS) • [**dli-connector**: link to DLI](tag:nohcs) • **http-connector**: link to HTTP/HTTPS servers (No link parameters are required.) • **dms-kafka-connector**: link to DMS Kafka
name	Yes	String	Job name, which contains 1 to 240 characters
from-link-name	Yes	String	Name of the source link, that is, the name of the link created through the API used to create a link
creation-user	No	String	User who created the job. The value is generated by the system.

Parameter	Mandatory	Type	Description
creation-date	No	Long	Time when the job was created, accurate to millisecond. The value is generated by the system.
update-date	No	Long	Time when the job was last updated, accurate to millisecond. The value is generated by the system.
is_incre_job	No	Boolean	Whether the job is an incremental job. This parameter has been discarded.
flag	No	Integer	Whether the job is a scheduled job. If yes, the value is **1**. Otherwise, the value is **0**. The value is generated by the system based on the scheduled task configuration.
files_read	No	Integer	Number of read files. The value is generated by the system.
update-user	No	String	User who last updated the job. The value is generated by the system.
external_id	No	String	ID of the job to be executed. For a local job, the value is in the format of **job_local1202051771_0002** . For a DLI job, the value is the DLI job ID, for example, **"12345"**. The value is generated by the system and does not need to be set.
type	No	String	Job type. The value of this parameter is the same as that of **job_type**. The options are as follows: • **NORMAL_JOB**: table/file migration job • **BATCH_JOB**: entire DB migration job • **SCENARIO_JOB**: scenario migration job

Parameter	Mandatory	Type	Description
execute_start_date	No	Long	Time when the last task was started, accurate to millisecond. The value is generated by the system.
delete_rows	No	Integer	Number of rows deleted by an incremental job. This parameter is deprecated.
enabled	No	Boolean	Whether the link is enabled. The value is generated by the system.
bytes_written	No	Long	Number of bytes written by the job. The value is generated by the system.
id	No	Integer	Job ID, which is generated by the system
is_use_sql	No	Boolean	Whether to use SQL statements. The value is generated by the system based on whether SQL statements are used for source data extraction. You do not need to set this parameter.
update_rows	No	Integer	Number of updated rows in an incremental job. This parameter is deprecated.
group_name	No	String	Group name
bytes_read	No	Long	Number of bytes read by the job. The value is generated by the system.
execute_update_date	No	Long	Time when the last task was updated, accurate to millisecond. The value is generated by the system.
write_rows	No	Integer	Number of rows written by an incremental job. This parameter is deprecated.
rows_written	No	Integer	Number of rows written by the job. The value is generated by the system.
rows_read	No	Long	Number of rows read by the job. The value is generated by the system.

Parameter	Mandatory	Type	Description
files_written	No	Integer	Number of written files. The value is generated by the system.
is_incrementing	No	Boolean	Whether the job is an incremental job. Similar to parameter **is_incre_job**, this parameter is deprecated.
execute_create_date	No	Long	Time when the last task was created, accurate to millisecond. The value is generated by the system.
status	No	String	Job execution status • **BOOTING**: The job is starting. • **RUNNING**: The job is running. • **SUCCEEDED**: The job was successfully executed. • **FAILED**: The job execution failed. • **NEW**: The job was not executed.

Table 5-142 ConfigValues

Parameter	Mandatory	Type	Description
configs	Yes	Array of **configs** objects	The data structures of source link parameters, destination link parameters, and job parameters are the same. However, the **inputs** parameter varies. For details, see the descriptions of **configs** parameters.
extended-configs	No	**extended-configs** object	Extended configuration. For details, see the descriptions of **extended-configs** parameters. The extended configuration is not open to external systems. You do not need to set it.

Table 5-143 configs

Parameter	Mandatory	Type	Description
inputs	Yes	Array of **Input** objects	Input parameter list. Each element in the list is in *name,value* format. For details, see the descriptions of **inputs** parameters. In the **from-config-values** data structure, the value of this parameter varies with the source link type. For details, see section "Source Job Parameters" in the *Cloud Data Migration User Guide*. In the **to-config-values** data structure, the value of this parameter varies with the destination link type. For details, see section "Destination Job Parameters" in the *Cloud Data Migration User Guide*. For details about the **inputs** parameter in the **driver-config-values** data structure, see the job parameter descriptions.
name	Yes	String	Configuration name. The value is **fromJobConfig** for a source job, **toJobConfig** for a destination job, and **linkConfig** for a link.
id	No	Integer	Configuration ID, which is generated by the system. You do not need to set this parameter.
type	No	String	Configuration type, which is generated by the system. You do not need to set this parameter. The value can be **LINK** (for link management APIs) or **JOB** (for job management APIs).

Table 5-144 Input

Parameter	Mandatory	Type	Description
name	Yes	String	Parameter name. • For link management APIs, parameter names start with **linkConfig.**. The parameters vary depending on the link type. For details, see the parameter descriptions of the corresponding link in **Link Parameters**. • For job management APIs, source link parameter names start with **fromJobConfig.**. For details, see the source job parameters in **Source Job Parameters**. Destination link parameter names start with **toJobConfig.**. For details, see **Destination Job Parameters**. For details about job parameters, see the task parameter descriptions in **Job Parameters**.
value	Yes	Object	Parameter value, which must be a string.
type	No	String	Value type, such as **STRING** and **INTEGER**. The value is set by the system.

Table 5-145 extended-configs

Parameter	Mandatory	Type	Description
name	No	String	Extended configuration name. This parameter is unavailable for external systems and does not need to be set.
value	No	String	Extended configuration value. This parameter is unavailable for external systems and does not need to be set.

Response Parameters

Status code: 200

Table 5-146 Response body parameters

Parameter	Type	Description
name	String	Job name
validation-result	Array of **JobValidationResult** objects	Check result • If the modification fails, the failure cause is returned. • If the modification is successful, an empty list is returned.

Table 5-147 JobValidationResult

Parameter	Type	Description
message	String	Error description
status	String	Error level, for example, **ERROR** or **WARNING**

Status code: 400

Table 5-148 Response body parameters

Parameter	Type	Description
code	String	Return code
errCode	String	Error code
message	String	Error message
externalMessage	String	Additional information

Example Requests

Creating a data migration job whose name is **es_css**, source link is an Elasticsearch link, and destination link is a DIS link

```
POST /v1.1/1551c7f6c808414d8e9f3c514a170f2e/clusters/6ec9a0a4-76be-4262-8697-e7af1fac7920/cdm/job

{
  "jobs" : [ {
    "job_type" : "NORMAL_JOB",
    "from-connector-name" : "elasticsearch-connector",
    "to-config-values" : {
```

```
    "configs" : [ {
      "inputs" : [ {
        "name" : "toJobConfig.streamName",
        "value" : "dis-lkGm"
      }, {
        "name" : "toJobConfig.separator",
        "value" : "|"
      }, {
        "name" : "toJobConfig.columnList",
        "value" : "1&2&3"
      } ],
      "name" : "toJobConfig"
    } ]
  },
  "to-link-name" : "dis",
  "driver-config-values" : {
    "configs" : [ {
      "inputs" : [ {
        "name" : "throttlingConfig.numExtractors",
        "value" : "1"
      }, {
        "name" : "throttlingConfig.submitToCluster",
        "value" : "false"
      }, {
        "name" : "throttlingConfig.numLoaders",
        "value" : "1"
      }, {
        "name" : "throttlingConfig.recordDirtyData",
        "value" : "false"
      } ],
      "name" : "throttlingConfig"
    }, {
      "inputs" : [ ],
      "name" : "jarConfig"
    }, {
      "inputs" : [ {
        "name" : "schedulerConfig.isSchedulerJob",
        "value" : "false"
      }, {
        "name" : "schedulerConfig.disposableType",
        "value" : "NONE"
      } ],
      "name" : "schedulerConfig"
    }, {
      "inputs" : [ ],
      "name" : "transformConfig"
    }, {
      "inputs" : [ {
        "name" : "retryJobConfig.retryJobType",
        "value" : "NONE"
      } ],
      "name" : "retryJobConfig"
    } ]
  },
  "from-config-values" : {
    "configs" : [ {
      "inputs" : [ {
        "name" : "fromJobConfig.index",
        "value" : "52est"
      }, {
        "name" : "fromJobConfig.type",
        "value" : "est_array"
      }, {
        "name" : "fromJobConfig.columnList",
        "value" : "array_f1_int:long&array_f2_text:string&array_f3_object:nested"
      }, {
        "name" : "fromJobConfig.splitNestedField",
        "value" : "false"
      } ],
```



```
      "name" : "fromJobConfig"
    } ]
  },
  "to-connector-name" : "dis-connector",
  "name" : "es_css",
  "from-link-name" : "css"
 } ]
}
```

Example Responses

Status code: 200

Request succeeded.

```
{
 "name" : "mysql2hive"
}
```

Status code: 400

Request error.

```
{
 "code" : "Cdm.0104",
 "errCode" : "Cdm.0104",
 "message" : "Job name already exist or created by other.",
 "ternalMessage" : "Job name already exist or created by other."
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Creating a data migration job whose name is **es_css**, source link is an Elasticsearch link, and destination link is a DIS link

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cdm.v1.region.CdmRegion;
import com.huaweicloud.sdk.cdm.v1.*;
import com.huaweicloud.sdk.cdm.v1.model.*;

import java.util.List;
import java.util.ArrayList;

public class CreateJobSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
```



```
        .withProjectId(projectId)
        .withAk(ak)
        .withSk(sk);

CdmClient client = CdmClient.newBuilder()
        .withCredential(auth)
        .withRegion(CdmRegion.valueOf("<YOUR REGION>"))
        .build();
CreateJobRequest request = new CreateJobRequest();
request.withClusterId("{cluster_id}");
CdmCreateJobJsonReq body = new CdmCreateJobJsonReq();
List<Input> listConfigsInputs = new ArrayList<>();
listConfigsInputs.add(
    new Input()
        .withName("fromJobConfig.index")
        .withValue("52est")
);
listConfigsInputs.add(
    new Input()
        .withName("fromJobConfig.type")
        .withValue("est_array")
);
listConfigsInputs.add(
    new Input()
        .withName("fromJobConfig.columnList")
        .withValue("array_f1_int:long&array_f2_text:string&array_f3_object:nested")
);
listConfigsInputs.add(
    new Input()
        .withName("fromJobConfig.splitNestedField")
        .withValue("false")
);
List<Configs> listFromConfigValuesConfigs = new ArrayList<>();
listFromConfigValuesConfigs.add(
    new Configs()
        .withInputs(listConfigsInputs)
        .withName("fromJobConfig")
);
ConfigValues fromconfigvaluesJobs = new ConfigValues();
fromconfigvaluesJobs.withConfigs(listFromConfigValuesConfigs);
List<Input> listConfigsInputs1 = new ArrayList<>();
listConfigsInputs1.add(
    new Input()
        .withName("retryJobConfig.retryJobType")
        .withValue("NONE")
);
List<Input> listConfigsInputs2 = new ArrayList<>();
listConfigsInputs2.add(
    new Input()
        .withName("schedulerConfig.isSchedulerJob")
        .withValue("false")
);
listConfigsInputs2.add(
    new Input()
        .withName("schedulerConfig.disposableType")
        .withValue("NONE")
);
List<Input> listConfigsInputs3 = new ArrayList<>();
listConfigsInputs3.add(
    new Input()
        .withName("throttlingConfig.numExtractors")
        .withValue("1")
);
listConfigsInputs3.add(
    new Input()
        .withName("throttlingConfig.submitToCluster")
        .withValue("false")
);
listConfigsInputs3.add(
```



```
        new Input()
            .withName("throttlingConfig.numLoaders")
            .withValue("1")
    );
    listConfigsInputs3.add(
        new Input()
            .withName("throttlingConfig.recordDirtyData")
            .withValue("false")
    );
    List<Configs> listDriverConfigValuesConfigs = new ArrayList<>();
    listDriverConfigValuesConfigs.add(
        new Configs()
            .withInputs(listConfigsInputs1)
            .withName("retryJobConfig")
    );
    ConfigValues driverconfigvaluesJobs = new ConfigValues();
    driverconfigvaluesJobs.withConfigs(listDriverConfigValuesConfigs);
    List<Input> listConfigsInputs4 = new ArrayList<>();
    listConfigsInputs4.add(
        new Input()
            .withName("toJobConfig.streamName")
            .withValue("dis-lkGm")
    );
    listConfigsInputs4.add(
        new Input()
            .withName("toJobConfig.separator")
            .withValue("|")
    );
    listConfigsInputs4.add(
        new Input()
            .withName("toJobConfig.columnList")
            .withValue("1&2&3")
    );
    List<Configs> listToConfigValuesConfigs = new ArrayList<>();
    listToConfigValuesConfigs.add(
        new Configs()
            .withInputs(listConfigsInputs4)
            .withName("toJobConfig")
    );
    ConfigValues toconfigvaluesJobs = new ConfigValues();
    toconfigvaluesJobs.withConfigs(listToConfigValuesConfigs);
    List<Job> listbodyJobs = new ArrayList<>();
    listbodyJobs.add(
        new Job()
            .withJobType(Job.JobTypeEnum.fromValue("NORMAL_JOB"))
            .withFromConnectorName("elasticsearch-connector")
            .withToConfigValues(toconfigvaluesJobs)
            .withToLinkName("dis")
            .withDriverConfigValues(driverconfigvaluesJobs)
            .withFromConfigValues(fromconfigvaluesJobs)
            .withToConnectorName("dis-connector")
            .withName("es_css")
            .withFromLinkName("css")
    );
    body.withJobs(listbodyJobs);
    request.withBody(body);
    try {
        CreateJobResponse response = client.createJob(request);
        System.out.println(response.toString());
    } catch (ConnectionException e) {
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
```



```
        }
    }
}
```

Python

Creating a data migration job whose name is **es_css**, source link is an Elasticsearch link, and destination link is a DIS link

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcdm.v1.region.cdm_region import CdmRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcdm.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CdmClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CdmRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateJobRequest()
        request.cluster_id = "{cluster_id}"
        listInputsConfigs = [
            Input(
                name="fromJobConfig.index",
                value="52est"
            ),
            Input(
                name="fromJobConfig.type",
                value="est_array"
            ),
            Input(
                name="fromJobConfig.columnList",
                value="array_f1_int:long&array_f2_text:string&array_f3_object:nested"
            ),
            Input(
                name="fromJobConfig.splitNestedField",
                value="false"
            )
        ]
        listConfigsFromconfigvalues = [
            Configs(
                inputs=listInputsConfigs,
                name="fromJobConfig"
            )
        ]
        fromconfigvaluesJobs = ConfigValues(
            configs=listConfigsFromconfigvalues
        )
        listInputsConfigs1 = [
            Input(
                name="retryJobConfig.retryJobType",
                value="NONE"
            )
```



```
    ]
    listInputsConfigs2 = [
        Input(
            name="schedulerConfig.isSchedulerJob",
            value="false"
        ),
        Input(
            name="schedulerConfig.disposableType",
            value="NONE"
        )
    ]
    listInputsConfigs3 = [
        Input(
            name="throttlingConfig.numExtractors",
            value="1"
        ),
        Input(
            name="throttlingConfig.submitToCluster",
            value="false"
        ),
        Input(
            name="throttlingConfig.numLoaders",
            value="1"
        ),
        Input(
            name="throttlingConfig.recordDirtyData",
            value="false"
        )
    ]
    listConfigsDriverconfigvalues = [
        Configs(
            inputs=listInputsConfigs1,
            name="retryJobConfig"
        )
    ]
    driverconfigvaluesJobs = ConfigValues(
        configs=listConfigsDriverconfigvalues
    )
    listInputsConfigs4 = [
        Input(
            name="toJobConfig.streamName",
            value="dis-lkGm"
        ),
        Input(
            name="toJobConfig.separator",
            value="|"
        ),
        Input(
            name="toJobConfig.columnList",
            value="1&2&3"
        )
    ]
    listConfigsToconfigvalues = [
        Configs(
            inputs=listInputsConfigs4,
            name="toJobConfig"
        )
    ]
    toconfigvaluesJobs = ConfigValues(
        configs=listConfigsToconfigvalues
    )
    listJobsbody = [
        Job(
            job_type="NORMAL_JOB",
            from_connector_name="elasticsearch-connector",
            to_config_values=toconfigvaluesJobs,
            to_link_name="dis",
            driver_config_values=driverconfigvaluesJobs,
            from_config_values=fromconfigvaluesJobs,
```



```
            to_connector_name="dis-connector",
            name="es_css",
            from_link_name="css"
        )
    ]
    request.body = CdmCreateJobJsonReq(
        jobs=listJobsbody
    )
    response = client.create_job(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

Creating a data migration job whose name is **es_css**, source link is an Elasticsearch link, and destination link is a DIS link

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cdm "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cdm.NewCdmClient(
        cdm.CdmClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreateJobRequest{}
    request.ClusterId = "{cluster_id}"
    valueInputs:= "52est"
    var valueInputsInterface interface{} = valueInputs
    valueInputs1:= "est_array"
    var valueInputsInterface1 interface{} = valueInputs1
    valueInputs2:= "array_f1_int:long&array_f2_text:string&array_f3_object:nested"
    var valueInputsInterface2 interface{} = valueInputs2
    valueInputs3:= "false"
    var valueInputsInterface3 interface{} = valueInputs3
    var listInputsConfigs = []model.Input{
        {
            Name: "fromJobConfig.index",
            Value: &valueInputsInterface,
        },
        {
```



```
        Name: "fromJobConfig.type",
        Value: &valueInputsInterface1,
    },
    {
        Name: "fromJobConfig.columnList",
        Value: &valueInputsInterface2,
    },
    {
        Name: "fromJobConfig.splitNestedField",
        Value: &valueInputsInterface3,
    },
}
var listConfigsFromConfigValues = []model.Configs{
    {
        Inputs: listInputsConfigs,
        Name: "fromJobConfig",
    },
}
fromconfigvaluesJobs := &model.ConfigValues{
    Configs: listConfigsFromConfigValues,
}
valueInputs4:= "NONE"
var valueInputsInterface4 interface{} = valueInputs4
var listInputsConfigs1 = []model.Input{
    {
        Name: "retryJobConfig.retryJobType",
        Value: &valueInputsInterface4,
    },
}
valueInputs5:= "false"
var valueInputsInterface5 interface{} = valueInputs5
valueInputs6:= "NONE"
var valueInputsInterface6 interface{} = valueInputs6
var listInputsConfigs2 = []model.Input{
    {
        Name: "schedulerConfig.isSchedulerJob",
        Value: &valueInputsInterface5,
    },
    {
        Name: "schedulerConfig.disposableType",
        Value: &valueInputsInterface6,
    },
}
valueInputs7:= "1"
var valueInputsInterface7 interface{} = valueInputs7
valueInputs8:= "false"
var valueInputsInterface8 interface{} = valueInputs8
valueInputs9:= "1"
var valueInputsInterface9 interface{} = valueInputs9
valueInputs10:= "false"
var valueInputsInterface10 interface{} = valueInputs10
var listInputsConfigs3 = []model.Input{
    {
        Name: "throttlingConfig.numExtractors",
        Value: &valueInputsInterface7,
    },
    {
        Name: "throttlingConfig.submitToCluster",
        Value: &valueInputsInterface8,
    },
    {
        Name: "throttlingConfig.numLoaders",
        Value: &valueInputsInterface9,
    },
    {
        Name: "throttlingConfig.recordDirtyData",
        Value: &valueInputsInterface10,
    },
}
```



```
    var listConfigsDriverConfigValues = []model.Configs{
        {
            Inputs: listInputsConfigs1,
            Name: "retryJobConfig",
        },
    }
    driverconfigvaluesJobs := &model.ConfigValues{
        Configs: listConfigsDriverConfigValues,
    }
    valueInputs11:= "dis-lkGm"
    var valueInputsInterface11 interface{} = valueInputs11
    valueInputs12:= "|"
    var valueInputsInterface12 interface{} = valueInputs12
    valueInputs13:= "1&2&3"
    var valueInputsInterface13 interface{} = valueInputs13
    var listInputsConfigs4 = []model.Input{
        {
            Name: "toJobConfig.streamName",
            Value: &valueInputsInterface11,
        },
        {
            Name: "toJobConfig.separator",
            Value: &valueInputsInterface12,
        },
        {
            Name: "toJobConfig.columnList",
            Value: &valueInputsInterface13,
        },
    }
    var listConfigsToConfigValues = []model.Configs{
        {
            Inputs: listInputsConfigs4,
            Name: "toJobConfig",
        },
    }
    toconfigvaluesJobs := &model.ConfigValues{
        Configs: listConfigsToConfigValues,
    }
    var listJobsbody = []model.Job{
        {
            JobType: model.GetJobJobTypeEnum().NORMAL_JOB,
            FromConnectorName: "elasticsearch-connector",
            ToConfigValues: toconfigvaluesJobs,
            ToLinkName: "dis",
            DriverConfigValues: driverconfigvaluesJobs,
            FromConfigValues: fromconfigvaluesJobs,
            ToConnectorName: "dis-connector",
            Name: "es_css",
            FromLinkName: "css",
        },
    }
    request.Body = &model.CdmCreateJobJsonReq{
        Jobs: listJobsbody,
    }
    response, err := client.CreateJob(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in **API Explorer**. SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Request succeeded.
400	Request error.

Error Codes

See **Error Codes**.

5.2.7 Starting a Job

Function

This API is used to start a job.

Calling Method

For details, see **Calling APIs**.

URI

PUT /v1.1/{project_id}/clusters/{cluster_id}/cdm/job/{job_name}/start

Table 5-149 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain it, see **Project ID and Account ID**.
cluster_id	Yes	String	Cluster ID
job_name	Yes	String	Job name

Request Parameters

Table 5-150 Request header parameters

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

Table 5-151 Request body parameters

Parameter	Mandatory	Type	Description
variables	No	Object	Start a job and configure variable parameters. If no variable is configured for the job, the object is empty.

Response Parameters

Status code: 200

Table 5-152 Response body parameters

Parameter	Type	Description
submissions	Array of **StartJobSubmission** objects	Job running information. For details, see the descriptions of **submission** parameters.

Table 5-153 StartJobSubmission

Parameter	Type	Description
isIncrementing	Boolean	Whether the job migrates incremental data
delete_rows	Integer	Number of deleted rows
update_rows	Integer	Number of updated rows
write_rows	Integer	Number of written rows
submission-id	Integer	Job submission ID

Parameter	Type	Description
job-name	String	Job name
creation-user	String	User who started the job
creation-date	Long	Job creation time, accurate to millisecond
execute-date	Long	Execution time
progress	Float	Job progress. If a job fails, the value is **-1**. Otherwise, the value ranges from 0 to 100.
status	String	Job status • **BOOTING**: The job is starting. • **FAILURE_ON_SUBMIT**: The job failed to be submitted. • **RUNNING**: The job is running. • **SUCCEEDED**: The job was executed successfully. • **FAILED**: The job failed. • **UNKNOWN**: The job status is unknown. • **NEVER_EXECUTED**: The job was not executed.
isStopingIncrement	String	Whether to stop incremental data migration
is-execute-auto	Boolean	Whether to execute the job as scheduled
last-update-date	Long	Time when the job was last updated
last-udpate-user	String	User who last updated the job status
isDeleteJob	Boolean	Whether to delete the job after it is executed

Example Requests

Start a job with empty parameters.

```
PUT /v1.1/1551c7f6c808414d8e9f3c514a170f2e/clusters/6ec9a0a4-76be-4262-8697-e7af1fac7920/cdm/job/jdbc2hive/start

{
  "variables" : { }
}
```

Example Responses

Status code: 200

Request succeeded.


```
{
  "submissions" : [ {
    "job-name" : "jdbc2hive",
    "creation-user" : "cdm",
    "creation-date" : "1536905778725",
    "progress" : 1,
    "status" : "BOOTING"
  } ]
}
```

Status Codes

Status Code	Description
200	Request succeeded.

Error Codes

See **Error Codes**.

5.2.8 Querying Job Status

Function

This API is used to query the job status.

Calling Method

For details, see **Calling APIs**.

URI

GET /v1.1/{project_id}/clusters/{cluster_id}/cdm/job/{job_name}/status

Table 5-154 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain it, see **Project ID and Account ID**.
cluster_id	Yes	String	Cluster ID
job_name	Yes	String	Job name

Request Parameters

Table 5-155 Request header parameters

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

Response Parameters

Status code: 200

Table 5-156 Response body parameters

Parameter	Type	Description
submissions	Array of **Submission** objects	Job running information. For details, see the descriptions of **submission** parameters.

Table 5-157 Submission

Parameter	Type	Description
isIncrementing	Boolean	Whether the job migrates incremental data
job-name	String	Job name
counters	**counters** object	Job running result statistics. This parameter is available only when **status** is **SUCCEEDED**. For details, see the description of the **counters** parameter.
isStopingIncrement	String	Whether to stop incremental data migration
is-execute-auto	Boolean	Whether to execute the job as scheduled
last-update-date	Long	Time when the job was last updated
last-udpate-user	String	User who last updated the job status
isDeleteJob	Boolean	Whether to delete the job after it is executed

Parameter	Type	Description
creation-user	String	User who submitted the job
creation-date	Long	Creation time
external-id	String	Job ID
progress	Float	Job progress. If a job fails, the value is **-1**. Otherwise, the value ranges from 0 to 100.
submission-id	Integer	Job submission ID
delete_rows	Integer	Number of deleted rows
update_rows	Integer	Number of updated rows
write_rows	Integer	Number of written rows
execute-date	Long	Execution time
status	String	Job status • **BOOTING**: The job is starting. • **FAILURE_ON_SUBMIT**: The job failed to be submitted. • **RUNNING**: The job is running. • **SUCCEEDED**: The job was executed successfully. • **FAILED**: The job failed. • **UNKNOWN**: The job status is unknown. • **NEVER_EXECUTED**: The job was not executed.
error-details	String	Error details. This parameter is available only when **status** is **FAILED**.
error-summary	String	Error summary. This parameter is available only when **status** is **FAILED**.

Table 5-158 counters

Parameter	Type	Description
org.apache.sqoop.submission.counter.SqoopCounters	**counter** object	Job running result statistics. For details, see the descriptions of **counter** parameters.

Table 5-159 counter

Parameter	Type	Description
BYTES_WRITTEN	Long	Number of bytes written
TOTAL_FILES	Integer	Total number of files
ROWS_READ	Long	Number of rows read
BYTES_READ	Long	Number of bytes read
ROWS_WRITTEN	Long	Number of rows written
FILES_WRITTEN	Integer	Number of files written
FILES_READ	Integer	Number of files read
TOTAL_SIZE	Long	Total number of bytes
FILES_SKIPPED	Integer	Number of files skipped
ROWS_WRITTEN_SKIPPED	Long	Number of rows skipped

Example Requests

```
GET /v1.1/1551c7f6c808414d8e9f3c514a170f2e/clusters/6ec9a0a4-76be-4262-8697-e7af1fac7920/cdm/job/jdbc2hive/status
```

Example Responses

Status code: 200

Request succeeded.

```
{
  "submissions" : [ {
    "job-name" : "jdbc2hive",
    "creation-user" : "cdm",
    "creation-date" : "1536905778725",
    "progress" : 1,
    "status" : "BOOTING"
  } ]
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
```



```
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cdm.v1.region.CdmRegion;
import com.huaweicloud.sdk.cdm.v1.*;
import com.huaweicloud.sdk.cdm.v1.model.*;


public class ShowJobStatusSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CdmClient client = CdmClient.newBuilder()
                .withCredential(auth)
                .withRegion(CdmRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowJobStatusRequest request = new ShowJobStatusRequest();
        request.withClusterId("{cluster_id}");
        request.withJobName("{job_name}");
        try {
            ShowJobStatusResponse response = client.showJobStatus(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcdm.v1.region.cdm_region import CdmRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcdm.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"
```



```
    credentials = BasicCredentials(ak, sk, projectId)

    client = CdmClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CdmRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowJobStatusRequest()
        request.cluster_id = "{cluster_id}"
        request.job_name = "{job_name}"
        response = client.show_job_status(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cdm "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cdm.NewCdmClient(
        cdm.CdmClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowJobStatusRequest{}
    request.ClusterId = "{cluster_id}"
    request.JobName = "{job_name}"
    response, err := client.ShowJobStatus(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```


More

For SDK sample code of more programming languages, see the Sample Code tab in **API Explorer**. SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Request succeeded.

Error Codes

See **Error Codes**.

5.2.9 Querying Job Execution History

Function

This API is used to query the job execution history.

Calling Method

For details, see **Calling APIs**.

URI

GET /v1.1/{project_id}/clusters/{cluster_id}/cdm/submissions

Table 5-160 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain it, see **Project ID and Account ID**.
cluster_id	Yes	String	Cluster ID

Table 5-161 Query Parameters

Parameter	Mandatory	Type	Description
jname	Yes	String	Job name

Request Parameters

Table 5-162 Request header parameters

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

Response Parameters

Status code: 200

Table 5-163 Response body parameters

Parameter	Type	Description
submissions	Array of **Submission** objects	Job running information. For details, see the descriptions of **submission** parameters.
total	Integer	Total number of historical records for a job
page_no	Integer	Page number
page_size	Integer	Number of records on each page. The default value is **10**.

Table 5-164 Submission

Parameter	Type	Description
isIncrementing	Boolean	Whether the job migrates incremental data
job-name	String	Job name
counters	**counters** object	Job running result statistics. This parameter is available only when **status** is **SUCCEEDED**. For details, see the description of the **counters** parameter.
isStopingIncrement	String	Whether to stop incremental data migration
is-execute-auto	Boolean	Whether to execute the job as scheduled

Parameter	Type	Description
last-update-date	Long	Time when the job was last updated
last-udpate-user	String	User who last updated the job status
isDeleteJob	Boolean	Whether to delete the job after it is executed
creation-user	String	User who submitted the job
creation-date	Long	Creation time
external-id	String	Job ID
progress	Float	Job progress. If a job fails, the value is **-1**. Otherwise, the value ranges from 0 to 100.
submission-id	Integer	Job submission ID
delete_rows	Integer	Number of deleted rows
update_rows	Integer	Number of updated rows
write_rows	Integer	Number of written rows
execute-date	Long	Execution time
status	String	Job status • **BOOTING**: The job is starting. • **FAILURE_ON_SUBMIT**: The job failed to be submitted. • **RUNNING**: The job is running. • **SUCCEEDED**: The job was executed successfully. • **FAILED**: The job failed. • **UNKNOWN**: The job status is unknown. • **NEVER_EXECUTED**: The job was not executed.
error-details	String	Error details. This parameter is available only when **status** is **FAILED**.
error-summary	String	Error summary. This parameter is available only when **status** is **FAILED**.

Table 5-165 counters

Parameter	Type	Description
org.apache.sqoop.submission.counter.SqoopCounters	**counter** object	Job running result statistics. For details, see the descriptions of **counter** parameters.

Table 5-166 counter

Parameter	Type	Description
BYTES_WRITTEN	Long	Number of bytes written
TOTAL_FILES	Integer	Total number of files
ROWS_READ	Long	Number of rows read
BYTES_READ	Long	Number of bytes read
ROWS_WRITTEN	Long	Number of rows written
FILES_WRITTEN	Integer	Number of files written
FILES_READ	Integer	Number of files read
TOTAL_SIZE	Long	Total number of bytes
FILES_SKIPPED	Integer	Number of files skipped
ROWS_WRITTEN_SKIPPED	Long	Number of rows skipped

Example Requests

```
GET /v1.1/1551c7f6c808414d8e9f3c514a170f2e/clusters/6ec9a0a4-76be-4262-8697-e7af1fac7920/cdm/submissions?jname=jdbc2hive
```

Example Responses

Status code: 200

Request succeeded.

```
{
  "submissions" : [ {
    "job-name" : "jdbc2hive",
    "creation-user" : "cdm",
    "creation-date" : "1536905778725",
    "progress" : 1,
    "status" : "BOOTING"
```



```
 } ]
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cdm.v1.region.CdmRegion;
import com.huaweicloud.sdk.cdm.v1.*;
import com.huaweicloud.sdk.cdm.v1.model.*;


public class ShowSubmissionsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CdmClient client = CdmClient.newBuilder()
                .withCredential(auth)
                .withRegion(CdmRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowSubmissionsRequest request = new ShowSubmissionsRequest();
        request.withClusterId("{cluster_id}");
        try {
            ShowSubmissionsResponse response = client.showSubmissions(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
```



```
from huaweicloudsdkcdm.v1.region.cdm_region import CdmRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcdm.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CdmClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CdmRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowSubmissionsRequest()
        request.cluster_id = "{cluster_id}"
        response = client.show_submissions(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cdm "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cdm.NewCdmClient(
        cdm.CdmClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowSubmissionsRequest{}
    request.ClusterId = "{cluster_id}"
    response, err := client.ShowSubmissions(request)
```



```
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in **API Explorer**. SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Request succeeded.

Error Codes

See **Error Codes**.

5.3 Link Management

5.3.1 Creating a Link

Function

This API is used to create a link.

Calling Method

For details, see **Calling APIs**.

URI

POST /v1.1/{project_id}/clusters/{cluster_id}/cdm/link

Table 5-167 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain it, see **Project ID and Account ID**.
cluster_id	Yes	String	Cluster ID

Table 5-168 Query Parameters

Parameter	Mandatory	Type	Description
validate	No	String	When the parameter is set to **true**, the API only validates whether the parameters are correctly configured, but does not create any link.

Request Parameters

Table 5-169 Request header parameters

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

Table 5-170 Request body parameters

Parameter	Mandatory	Type	Description
links	Yes	Array of **links** objects	Link list. For details, see the descriptions of **links** parameters.

Table 5-171 links

Parameter	Mandatory	Type	Description
link-config-values	Yes	**link-config-values** object	Link parameters. For details, see the descriptions of **link-config-values** parameters.
creation-user	No	String	User who created a link
name	Yes	String	Link name
id	No	Integer	Link ID
creation-date	No	Long	Time when a link was created

Parameter	Mandatory	Type	Description
connector-name	Yes	String	Connector name. The corresponding link parameter is **generic-jdbc-connector**, which indicates a relational database link. **obs-connector**: link to OBS **hdfs-connector**: link to HDFS **hbase-connector**: link to HBase and link to CloudTable **hive-connector**: link to Hive **ftp-connector/sftp-connector**: link to an FTP or SFTP server **mongodb-connector**: link to MongoDB **redis-connector**: link to Redis/DCS **kafka-connector**: link to Kafka **dis-connector**: link to DIS **elasticsearch-connector**: link to Elasticsearch/Cloud Search Service (CSS) [**dli-connector**: link to DLI] (tag:nohcs)**http-connector**: link to HTTP/ HTTPS (No link parameters are required.) **dms-kafka-connector**: link to DMSKafka
update-date	No	Long	Time when a link was updated
enabled	No	Boolean	Whether to activate the link. The default value is **true**.
update-user	No	String	User who updated a link

Table 5-172 link-config-values

Parameter	Mandatory	Type	Description
configs	Yes	Array of **configs** objects	Data structure of link parameters. For details, see the descriptions of **configs** parameters.
extended-configs	No	**extended-configs** object	Extended configuration. For details, see the descriptions of **extended-configs** parameters.
validators	No	Array of strings	Validator

Table 5-173 configs

Parameter	Mandatory	Type	Description
inputs	Yes	Array of **Input** objects	Input parameter list. Each element in the list is in *name,value* format. For details, see the descriptions of **inputs** parameters. In the **from-config-values** data structure, the value of this parameter varies with the source link type. For details, see section "Source Job Parameters" in the *Cloud Data Migration User Guide*. In the **to-config-values** data structure, the value of this parameter varies with the destination link type. For details, see section "Destination Job Parameters" in the *Cloud Data Migration User Guide*. For details about the **inputs** parameter in the **driver-config-values** data structure, see the job parameter descriptions.
name	Yes	String	Configuration name. The value is **fromJobConfig** for a source job, **toJobConfig** for a destination job, and **linkConfig** for a link.
id	No	Integer	Configuration ID, which is generated by the system. You do not need to set this parameter.
type	No	String	Configuration type, which is generated by the system. You do not need to set this parameter. The value can be **LINK** (for link management APIs) or **JOB** (for job management APIs).

Table 5-174 Input

Parameter	Mandatory	Type	Description
name	Yes	String	Parameter name. • For link management APIs, parameter names start with **linkConfig.**. The parameters vary depending on the link type. For details, see the parameter descriptions of the corresponding link in **Link Parameters**. • For job management APIs, source link parameter names start with **fromJobConfig.**. For details, see the source job parameters in **Source Job Parameters**. Destination link parameter names start with **toJobConfig.**. For details, see **Destination Job Parameters**. For details about job parameters, see the task parameter descriptions in **Job Parameters**.
value	Yes	Object	Parameter value, which must be a string.
type	No	String	Value type, such as **STRING** and **INTEGER**. The value is set by the system.

Table 5-175 extended-configs

Parameter	Mandatory	Type	Description
name	No	String	Name
value	No	String	Value

Response Parameters

Status code: 200

Table 5-176 Response body parameters

Parameter	Type	Description
name	String	Link name
validation-result	Array of **validationResult** objects	Validation structure. If a link fails to be created, the failure cause is returned. For details, see the descriptions of **validation-result** parameters. If a link is successfully created, an empty list is returned.

Table 5-177 validationResult

Parameter	Type	Description
linkConfig	Array of **validationLinkConfig** objects	Validation result of link creation or update. For details, see the descriptions of **validationLinkConfig** parameters.

Table 5-178 validationLinkConfig

Parameter	Type	Description
message	String	Error message
status	String	Error level, for example, **ERROR** or **WARNING**

Status code: 400

Table 5-179 Response body parameters

Parameter	Type	Description
code	String	Return code
errCode	String	Error code
message	String	Error message
externalMessage	String	Additional information

Status code: 500

Table 5-180 Response body parameters

Parameter	Type	Description
message	String	Error message
status	String	Error level, for example, **ERROR** or **WARNING**

Example Requests

Creating a link named **mysql_link**

```
POST /v1.1/1551c7f6c808414d8e9f3c514a170f2e/clusters/6ec9a0a4-76be-4262-8697-e7af1fac7920/cdm/link

{
  "links" : [ {
    "link-config-values" : {
      "configs" : [ {
        "inputs" : [ {
          "name" : "linkConfig.databaseType",
          "value" : "MYSQL"
        }, {
          "name" : "linkConfig.host",
          "value" : "100.94.8.163"
        }, {
          "name" : "linkConfig.port",
          "value" : "3306"
        }, {
          "name" : "linkConfig.database",
          "value" : "DB_name"
        }, {
          "name" : "linkConfig.username",
          "value" : "username"
        }, {
          "name" : "linkConfig.password",
          "value" : "DB_password"
        }, {
          "name" : "linkConfig.fetchSize",
          "value" : "100000"
        }, {
          "name" : "linkConfig.usingNative",
          "value" : "false"
        } ],
        "name" : "linkConfig"
      } ]
    },
    "name" : "mysql_link",
    "creation-date" : 1496654788622,
    "connector-name" : "generic-jdbc-connector",
    "update-date" : 1496654788622,
    "enabled" : true
  } ]
}
```

Example Responses

Status code: 200

Request succeeded.

```
{
  "name" : "rdb_link",
  "validation-result" : [ { } ]
}
```


Status code: 400

Request error.

```
{
  "code" : "Cdm.0315",
  "errCode" : "Cdm.0315",
  "message" : "Link name [ftp_link] already exist or created by other user.",
  "externalMessage" : "Link name [ftp_link] already exist or created by other user."
}
```

Status code: 500

Internal service error. For details about the returned error code, see Error Codes.

```
{
  "validation-result" : [ {
    "linkConfig" : [ {
      "message" : "Can't connect to the database with given credentials: The authentication type 12 is not supported. Check that you have configured the pg_hba.conf file to include the client's IP address or subnet, and that it is using an authentication scheme supported by the driver.",
      "status" : "ERROR"
    } ]
  } ]
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Creating a link named **mysql_link**

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cdm.v1.region.CdmRegion;
import com.huaweicloud.sdk.cdm.v1.*;
import com.huaweicloud.sdk.cdm.v1.model.*;

import java.util.List;
import java.util.ArrayList;

public class CreateLinkSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CdmClient client = CdmClient.newBuilder()
                .withCredential(auth)
```



```
        .withRegion(CdmRegion.valueOf("<YOUR REGION>"))
        .build();
CreateLinkRequest request = new CreateLinkRequest();
request.withClusterId("{cluster_id}");
CdmCreateAndUpdateLinkReq body = new CdmCreateAndUpdateLinkReq();
List<Input> listConfigsInputs = new ArrayList<>();
listConfigsInputs.add(
    new Input()
        .withName("linkConfig.databaseType")
        .withValue("MYSQL")
);
listConfigsInputs.add(
    new Input()
        .withName("linkConfig.host")
        .withValue("100.94.8.163")
);
listConfigsInputs.add(
    new Input()
        .withName("linkConfig.port")
        .withValue("3306")
);
listConfigsInputs.add(
    new Input()
        .withName("linkConfig.database")
        .withValue("DB_name")
);
listConfigsInputs.add(
    new Input()
        .withName("linkConfig.username")
        .withValue("username")
);
listConfigsInputs.add(
    new Input()
        .withName("linkConfig.password")
        .withValue("DB_password")
);
listConfigsInputs.add(
    new Input()
        .withName("linkConfig.fetchSize")
        .withValue("100000")
);
listConfigsInputs.add(
    new Input()
        .withName("linkConfig.usingNative")
        .withValue("false")
);
List<Configs> listLinkConfigValuesConfigs = new ArrayList<>();
listLinkConfigValuesConfigs.add(
    new Configs()
        .withInputs(listConfigsInputs)
        .withName("linkConfig")
);
LinksLinkconfigvalues linkConfigValuesLinks = new LinksLinkconfigvalues();
linkConfigValuesLinks.withConfigs(listLinkConfigValuesConfigs);
List<Links> listbodyLinks = new ArrayList<>();
listbodyLinks.add(
    new Links()
        .withLinkConfigValues(linkConfigValuesLinks)
        .withName("mysql_link")
        .withCreationDate(1496654788622L)
        .withConnectorName("generic-jdbc-connector")
        .withUpdateDate(1496654788622L)
        .withEnabled(true)
);
body.withLinks(listbodyLinks);
request.withBody(body);
try {
    CreateLinkResponse response = client.createLink(request);
    System.out.println(response.toString());
```



```
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

Creating a link named **mysql_link**

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcdm.v1.region.cdm_region import CdmRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcdm.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CdmClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CdmRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateLinkRequest()
        request.cluster_id = "{cluster_id}"
        listInputsConfigs = [
            Input(
                name="linkConfig.databaseType",
                value="MYSQL"
            ),
            Input(
                name="linkConfig.host",
                value="100.94.8.163"
            ),
            Input(
                name="linkConfig.port",
                value="3306"
            ),
            Input(
                name="linkConfig.database",
                value="DB_name"
            ),
            Input(
                name="linkConfig.username",
                value="username"
            ),
            Input(
                name="linkConfig.password",
```



```
            value="DB_password"
        ),
        Input(
            name="linkConfig.fetchSize",
            value="100000"
        ),
        Input(
            name="linkConfig.usingNative",
            value="false"
        )
    ]
    listConfigsLinkConfigValues = [
        Configs(
            inputs=listInputsConfigs,
            name="linkConfig"
        )
    ]
    linkConfigValuesLinks = LinksLinkconfigvalues(
        configs=listConfigsLinkConfigValues
    )
    listLinksbody = [
        Links(
            link_config_values=linkConfigValuesLinks,
            name="mysql_link",
            creation_date=1496654788622,
            connector_name="generic-jdbc-connector",
            update_date=1496654788622,
            enabled=True
        )
    ]
    request.body = CdmCreateAndUpdateLinkReq(
        links=listLinksbody
    )
    response = client.create_link(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

Creating a link named **mysql_link**

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cdm "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
```



```
    Build()

client := cdm.NewCdmClient(
    cdm.CdmClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.CreateLinkRequest{}
request.ClusterId = "{cluster_id}"
valueInputs:= "MYSQL"
var valueInputsInterface interface{} = valueInputs
valueInputs1:= "100.94.8.163"
var valueInputsInterface1 interface{} = valueInputs1
valueInputs2:= "3306"
var valueInputsInterface2 interface{} = valueInputs2
valueInputs3:= "DB_name"
var valueInputsInterface3 interface{} = valueInputs3
valueInputs4:= "username"
var valueInputsInterface4 interface{} = valueInputs4
valueInputs5:= "DB_password"
var valueInputsInterface5 interface{} = valueInputs5
valueInputs6:= "100000"
var valueInputsInterface6 interface{} = valueInputs6
valueInputs7:= "false"
var valueInputsInterface7 interface{} = valueInputs7
var listInputsConfigs = []model.Input{
    {
        Name: "linkConfig.databaseType",
        Value: &valueInputsInterface,
    },
    {
        Name: "linkConfig.host",
        Value: &valueInputsInterface1,
    },
    {
        Name: "linkConfig.port",
        Value: &valueInputsInterface2,
    },
    {
        Name: "linkConfig.database",
        Value: &valueInputsInterface3,
    },
    {
        Name: "linkConfig.username",
        Value: &valueInputsInterface4,
    },
    {
        Name: "linkConfig.password",
        Value: &valueInputsInterface5,
    },
    {
        Name: "linkConfig.fetchSize",
        Value: &valueInputsInterface6,
    },
    {
        Name: "linkConfig.usingNative",
        Value: &valueInputsInterface7,
    },
}
var listConfigsLinkConfigValues = []model.Configs{
    {
        Inputs: listInputsConfigs,
        Name: "linkConfig",
    },
}
linkConfigValuesLinks := &model.LinksLinkconfigvalues{
    Configs: listConfigsLinkConfigValues,
}
```



```
    creationDateLinks:= int64(1496654788622)
    updateDateLinks:= int64(1496654788622)
    enabledLinks:= true
    var listLinksbody = []model.Links{
        {
            LinkConfigValues: linkConfigValuesLinks,
            Name: "mysql_link",
            CreationDate: &creationDateLinks,
            ConnectorName: "generic-jdbc-connector",
            UpdateDate: &updateDateLinks,
            Enabled: &enabledLinks,
        },
    }
    request.Body = &model.CdmCreateAndUpdateLinkReq{
        Links: listLinksbody,
    }
    response, err := client.CreateLink(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in **API Explorer**. SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Request succeeded.
400	Request error.
401	Authentication failed.
403	No operation permissions.
404	No resources found.
500	Internal service error. For details about the returned error code, see Error Codes.
503	Service unavailable.

Error Codes

See **Error Codes**.

5.3.2 Querying a Link

Function

This API is used to query a link.

Calling Method

For details, see **Calling APIs**.

URI

GET /v1.1/{project_id}/clusters/{cluster_id}/cdm/link/{link_name}

Table 5-181 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain it, see **Project ID and Account ID**.
cluster_id	Yes	String	Cluster ID
link_name	Yes	String	Link name

Request Parameters

Table 5-182 Request header parameters

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

Response Parameters

Status code: 200

Table 5-183 Response body parameters

Parameter	Type	Description
links	Array of **links** objects	Link list. For details, see the descriptions of **links** parameters.
fromTo-unMapping	String	Source and destination data sources not supported by table/file migration
batchFromTo-mapping	String	Source and destination data sources supported by entire DB migration

Table 5-184 links

Parameter	Type	Description
link-config-values	**link-config-values** object	Link parameters. For details, see the descriptions of **link-config-values** parameters.
creation-user	String	User who created a link
name	String	Link name
id	Integer	Link ID
creation-date	Long	Time when a link was created
connector-name	String	Connector name. The corresponding link parameter is **generic-jdbc-connector**, which indicates a relational database link. **obs-connector**: link to OBS **hdfs-connector**: link to HDFS **hbase-connector**: link to HBase and link to CloudTable **hive-connector**: link to Hive **ftp-connector/sftp-connector**: link to an FTP or SFTP server **mongodb-connector**: link to MongoDB **redis-connector**: link to Redis/DCS **kafka-connector**: link to Kafka **dis-connector**: link to DIS **elasticsearch-connector**: link to Elasticsearch/Cloud Search Service (CSS) [**dli-connector**: link to DLI] (tag:nohcs)**http-connector**: link to HTTP/HTTPS (No link parameters are required.) **dms-kafka-connector**: link to DMSKafka
update-date	Long	Time when a link was updated
enabled	Boolean	Whether to activate the link. The default value is **true**.
update-user	String	User who updated a link

Table 5-185 link-config-values

Parameter	Type	Description
configs	Array of **configs** objects	Data structure of link parameters. For details, see the descriptions of **configs** parameters.
extended-configs	**extended-configs** object	Extended configuration. For details, see the descriptions of **extended-configs** parameters.
validators	Array of strings	Validator

Table 5-186 configs

Parameter	Type	Description
inputs	Array of **Input** objects	Input parameter list. Each element in the list is in *name,value* format. For details, see the descriptions of **inputs** parameters. In the **from-config-values** data structure, the value of this parameter varies with the source link type. For details, see section "Source Job Parameters" in the *Cloud Data Migration User Guide*. In the **to-config-values** data structure, the value of this parameter varies with the destination link type. For details, see section "Destination Job Parameters" in the *Cloud Data Migration User Guide*. For details about the **inputs** parameter in the **driver-config-values** data structure, see the job parameter descriptions.
name	String	Configuration name. The value is **fromJobConfig** for a source job, **toJobConfig** for a destination job, and **linkConfig** for a link.
id	Integer	Configuration ID, which is generated by the system. You do not need to set this parameter.
type	String	Configuration type, which is generated by the system. You do not need to set this parameter. The value can be **LINK** (for link management APIs) or **JOB** (for job management APIs).

Table 5-187 Input

Parameter	Type	Description
name	String	Parameter name. • For link management APIs, parameter names start with **linkConfig.**. The parameters vary depending on the link type. For details, see the parameter descriptions of the corresponding link in **Link Parameters**. • For job management APIs, source link parameter names start with **fromJobConfig.**. For details, see the source job parameters in **Source Job Parameters**. Destination link parameter names start with **toJobConfig.**. For details, see **Destination Job Parameters**. For details about job parameters, see the task parameter descriptions in **Job Parameters**.

Parameter	Type	Description
value	Object	Parameter value, which must be a string.
type	String	Value type, such as **STRING** and **INTEGER**. The value is set by the system.

Table 5-188 extended-configs

Parameter	Type	Description
name	String	Name
value	String	Value

Example Requests

```
GET /v1.1/1551c7f6c808414d8e9f3c514a170f2e/clusters/6ec9a0a4-76be-4262-8697-e7af1fac7920/cdm/link/sftplink
```

Example Responses

Status code: 200

Request succeeded.

```
{
  "links" : [ {
    "link-config-values" : {
      "configs" : [ {
        "inputs" : [ {
          "name" : "linkConfig.server",
          "type" : "STRING",
          "value" : "100.94.8.163"
        }, {
          "name" : "linkConfig.port",
          "type" : "INTEGER",
          "value" : 22
        }, {
          "name" : "linkConfig.username",
          "type" : "STRING",
          "value" : "root"
        }, {
          "name" : "linkConfig.password",
          "type" : "STRING",
          "value" : "Add password here"
        } ],
        "name" : "linkConfig"
      } ]
    },
    "creation-user" : "cdm",
    "name" : "sftp_link",
    "creation-date" : 1516674482640,
    "connector-name" : "sftp-connector",
    "update-date" : 1516674476022,
    "enabled" : true,
    "update-user" : "cdm"
  } ]
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cdm.v1.region.CdmRegion;
import com.huaweicloud.sdk.cdm.v1.*;
import com.huaweicloud.sdk.cdm.v1.model.*;


public class ShowLinkSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CdmClient client = CdmClient.newBuilder()
                .withCredential(auth)
                .withRegion(CdmRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowLinkRequest request = new ShowLinkRequest();
        request.withClusterId("{cluster_id}");
        request.withLinkName("{link_name}");
        try {
            ShowLinkResponse response = client.showLink(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcdm.v1.region.cdm_region import CdmRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcdm.v1 import *
```



```
if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CdmClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CdmRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowLinkRequest()
        request.cluster_id = "{cluster_id}"
        request.link_name = "{link_name}"
        response = client.show_link(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cdm "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cdm.NewCdmClient(
        cdm.CdmClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowLinkRequest{}
    request.ClusterId = "{cluster_id}"
    request.LinkName = "{link_name}"
    response, err := client.ShowLink(request)
    if err == nil {
```



```
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in **API Explorer**. SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Request succeeded.
400	Request error.
401	Authentication failed.
403	No operation permissions.
404	No resources found.
500	Internal service error. For details about the returned error code, see Error Codes.
503	Service unavailable.

Error Codes

See **Error Codes**.

5.3.3 Deleting a Link

Function

This API is used to delete a link.

Calling Method

For details, see **Calling APIs**.

URI

DELETE /v1.1/{project_id}/clusters/{cluster_id}/cdm/link/{link_name}

Table 5-189 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain it, see **Project ID and Account ID**.
cluster_id	Yes	String	Cluster ID
link_name	Yes	String	Name of the link to be deleted

Request Parameters

Table 5-190 Request header parameters

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

Response Parameters

Status code: 500

Table 5-191 Response body parameters

Parameter	Type	Description
errCode	String	Error code
externalMessage	String	Error message

Example Requests

```
DELETE /v1.1/1551c7f6c808414d8e9f3c514a170f2e/clusters/6ec9a0a4-76be-4262-8697-e7af1fac7920/cdm/link/jdbclink
```

Example Responses

Status code: 500

Internal service error. For details about the returned error code, see Error Codes.

```
{
  "errCode" : "Cdm.0021",
```

```
  "externalMessage" : "Given link name is in use"
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cdm.v1.region.CdmRegion;
import com.huaweicloud.sdk.cdm.v1.*;
import com.huaweicloud.sdk.cdm.v1.model.*;


public class DeleteLinkSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CdmClient client = CdmClient.newBuilder()
                .withCredential(auth)
                .withRegion(CdmRegion.valueOf("<YOUR REGION>"))
                .build();
        DeleteLinkRequest request = new DeleteLinkRequest();
        request.withClusterId("{cluster_id}");
        request.withLinkName("{link_name}");
        try {
            DeleteLinkResponse response = client.deleteLink(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

import os
```



```
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcdm.v1.region.cdm_region import CdmRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcdm.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"

    credentials = BasicCredentials(ak, sk, projectId)

    client = CdmClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CdmRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DeleteLinkRequest()
        request.cluster_id = "{cluster_id}"
        request.link_name = "{link_name}"
        response = client.delete_link(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cdm "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cdm.NewCdmClient(
        cdm.CdmClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.DeleteLinkRequest{}
```



```
    request.ClusterId = "{cluster_id}"
    request.LinkName = "{link_name}"
    response, err := client.DeleteLink(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in **API Explorer**. SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Request succeeded.
400	Request error.
401	Authentication failed.
403	No operation permissions.
404	No resources found.
500	Internal service error. For details about the returned error code, see Error Codes.
503	Service unavailable.

Error Codes

See **Error Codes**.

5.3.4 Modifying a Link

Function

This API is used to modify a link.

Calling Method

For details, see **Calling APIs**.

URI

PUT /v1.1/{project_id}/clusters/{cluster_id}/cdm/link/{link_name}

Table 5-192 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details about how to obtain it, see **Project ID and Account ID**.
cluster_id	Yes	String	Cluster ID
link_name	Yes	String	Link name

Request Parameters

Table 5-193 Request header parameters

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

Table 5-194 Request body parameters

Parameter	Mandatory	Type	Description
links	Yes	Array of **links** objects	Link list. For details, see the descriptions of **links** parameters.

Table 5-195 links

Parameter	Mandatory	Type	Description
link-config-values	Yes	**link-config-values** object	Link parameters. For details, see the descriptions of **link-config-values** parameters.
creation-user	No	String	User who created a link
name	Yes	String	Link name
id	No	Integer	Link ID
creation-date	No	Long	Time when a link was created

Parameter	Mandatory	Type	Description
connector-name	Yes	String	Connector name. The corresponding link parameter is **generic-jdbc-connector**, which indicates a relational database link. **obs-connector**: link to OBS **hdfs-connector**: link to HDFS **hbase-connector**: link to HBase and link to CloudTable **hive-connector**: link to Hive **ftp-connector/sftp-connector**: link to an FTP or SFTP server **mongodb-connector**: link to MongoDB **redis-connector**: link to Redis/DCS **kafka-connector**: link to Kafka **dis-connector**: link to DIS **elasticsearch-connector**: link to Elasticsearch/Cloud Search Service (CSS) [**dli-connector**: link to DLI] (tag:nohcs)**http-connector**: link to HTTP/HTTPS (No link parameters are required.) **dms-kafka-connector**: link to DMSKafka
update-date	No	Long	Time when a link was updated
enabled	No	Boolean	Whether to activate the link. The default value is **true**.
update-user	No	String	User who updated a link

Table 5-196 link-config-values

Parameter	Mandatory	Type	Description
configs	Yes	Array of **configs** objects	Data structure of link parameters. For details, see the descriptions of **configs** parameters.
extended-configs	No	**extended-configs** object	Extended configuration. For details, see the descriptions of **extended-configs** parameters.
validators	No	Array of strings	Validator

Table 5-197 configs

Parameter	Mandatory	Type	Description
inputs	Yes	Array of **Input** objects	Input parameter list. Each element in the list is in *name,value* format. For details, see the descriptions of **inputs** parameters. In the **from-config-values** data structure, the value of this parameter varies with the source link type. For details, see section "Source Job Parameters" in the *Cloud Data Migration User Guide*. In the **to-config-values** data structure, the value of this parameter varies with the destination link type. For details, see section "Destination Job Parameters" in the *Cloud Data Migration User Guide*. For details about the **inputs** parameter in the **driver-config-values** data structure, see the job parameter descriptions.
name	Yes	String	Configuration name. The value is **fromJobConfig** for a source job, **toJobConfig** for a destination job, and **linkConfig** for a link.
id	No	Integer	Configuration ID, which is generated by the system. You do not need to set this parameter.
type	No	String	Configuration type, which is generated by the system. You do not need to set this parameter. The value can be **LINK** (for link management APIs) or **JOB** (for job management APIs).

Table 5-198 Input

Parameter	Mandatory	Type	Description
name	Yes	String	Parameter name. • For link management APIs, parameter names start with **linkConfig.**. The parameters vary depending on the link type. For details, see the parameter descriptions of the corresponding link in **Link Parameters**. • For job management APIs, source link parameter names start with **fromJobConfig.**. For details, see the source job parameters in **Source Job Parameters**. Destination link parameter names start with **toJobConfig.**. For details, see **Destination Job Parameters**. For details about job parameters, see the task parameter descriptions in **Job Parameters**.
value	Yes	Object	Parameter value, which must be a string.
type	No	String	Value type, such as **STRING** and **INTEGER**. The value is set by the system.

Table 5-199 extended-configs

Parameter	Mandatory	Type	Description
name	No	String	Name
value	No	String	Value

Response Parameters

Status code: 200

Table 5-200 Response body parameters

Parameter	Type	Description
validation-result	Array of **validationResult** objects	Validation structure. If a link fails to be created, the failure cause is returned. For details, see the descriptions of **validation-result** parameters. If a link is successfully created, an empty list is returned.

Table 5-201 validationResult

Parameter	Type	Description
linkConfig	Array of **validationLinkConfig** objects	Validation result of link creation or update. For details, see the descriptions of **validationLinkConfig** parameters.

Table 5-202 validationLinkConfig

Parameter	Type	Description
message	String	Error message
status	String	Error level, for example, **ERROR** or **WARNING**

Status code: 500

Table 5-203 Response body parameters

Parameter	Type	Description
message	String	Error message
status	String	Error level, for example, **ERROR** or **WARNING**

Example Requests

Modifying a link named **mysql_link**

```
PUT /v1.1/1551c7f6c808414d8e9f3c514a170f2e/clusters/6ec9a0a4-76be-4262-8697-e7af1fac7920/cdm/link/rdb_link

{
  "links" : [ {
    "link-config-values" : {
      "configs" : [ {
        "inputs" : [ {
          "name" : "linkConfig.databaseType",
```

```
        "value" : "MYSQL"
      }, {
        "name" : "linkConfig.host",
        "value" : "100.94.8.163"
      }, {
        "name" : "linkConfig.port",
        "value" : "3306"
      }, {
        "name" : "linkConfig.database",
        "value" : "DB_name"
      }, {
        "name" : "linkConfig.username",
        "value" : "username"
      }, {
        "name" : "linkConfig.password",
        "value" : "DB_password"
      }, {
        "name" : "linkConfig.fetchSize",
        "value" : "100000"
      }, {
        "name" : "linkConfig.usingNative",
        "value" : "false"
      } ],
      "name" : "linkConfig"
    } ]
  },
  "name" : "mysql_link",
  "creation-date" : 1496654788622,
  "connector-name" : "generic-jdbc-connector",
  "update-date" : 1496654788622,
  "enabled" : true
 } ]
}
```

Example Responses

Status code: 200

Request succeeded.

```
{
 "validation-result" : [ { } ]
}
```

Status code: 500

Internal service error. For details about the returned error code, see Error Codes.

```
{
 "validation-result" : [ {
  "linkConfig" : [ {
   "message" : "Can't connect to the database with given credentials: The authentication type 12 is not supported. Check that you have configured the pg_hba.conf file to include the client's IP address or subnet, and that it is using an authentication scheme supported by the driver.",
   "status" : "ERROR"
  } ]
 } ]
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Modifying a link named **mysql_link**


```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.cdm.v1.region.CdmRegion;
import com.huaweicloud.sdk.cdm.v1.*;
import com.huaweicloud.sdk.cdm.v1.model.*;

import java.util.List;
import java.util.ArrayList;

public class UpdateLinkSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        String projectId = "{project_id}";

        ICredential auth = new BasicCredentials()
                .withProjectId(projectId)
                .withAk(ak)
                .withSk(sk);

        CdmClient client = CdmClient.newBuilder()
                .withCredential(auth)
                .withRegion(CdmRegion.valueOf("<YOUR REGION>"))
                .build();
        UpdateLinkRequest request = new UpdateLinkRequest();
        request.withClusterId("{cluster_id}");
        request.withLinkName("{link_name}");
        CdmCreateAndUpdateLinkReq body = new CdmCreateAndUpdateLinkReq();
        List<Input> listConfigsInputs = new ArrayList<>();
        listConfigsInputs.add(
            new Input()
                .withName("linkConfig.databaseType")
                .withValue("MYSQL")
        );
        listConfigsInputs.add(
            new Input()
                .withName("linkConfig.host")
                .withValue("100.94.8.163")
        );
        listConfigsInputs.add(
            new Input()
                .withName("linkConfig.port")
                .withValue("3306")
        );
        listConfigsInputs.add(
            new Input()
                .withName("linkConfig.database")
                .withValue("DB_name")
        );
        listConfigsInputs.add(
            new Input()
                .withName("linkConfig.username")
                .withValue("username")
        );
        listConfigsInputs.add(
            new Input()
                .withName("linkConfig.password")
                .withValue("DB_password")
```



```
        );
        listConfigsInputs.add(
            new Input()
                .withName("linkConfig.fetchSize")
                .withValue("100000")
        );
        listConfigsInputs.add(
            new Input()
                .withName("linkConfig.usingNative")
                .withValue("false")
        );
        List<Configs> listLinkConfigValuesConfigs = new ArrayList<>();
        listLinkConfigValuesConfigs.add(
            new Configs()
                .withInputs(listConfigsInputs)
                .withName("linkConfig")
        );
        LinksLinkconfigvalues linkConfigValuesLinks = new LinksLinkconfigvalues();
        linkConfigValuesLinks.withConfigs(listLinkConfigValuesConfigs);
        List<Links> listbodyLinks = new ArrayList<>();
        listbodyLinks.add(
            new Links()
                .withLinkConfigValues(linkConfigValuesLinks)
                .withName("mysql_link")
                .withCreationDate(1496654788622L)
                .withConnectorName("generic-jdbc-connector")
                .withUpdateDate(1496654788622L)
                .withEnabled(true)
        );
        body.withLinks(listbodyLinks);
        request.withBody(body);
        try {
            UpdateLinkResponse response = client.updateLink(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

Modifying a link named **mysql_link**

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcdm.v1.region.cdm_region import CdmRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcdm.v1 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]
    projectId = "{project_id}"
```



```
credentials = BasicCredentials(ak, sk, projectId)

client = CdmClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(CdmRegion.value_of("<YOUR REGION>")) \
    .build()

try:
    request = UpdateLinkRequest()
    request.cluster_id = "{cluster_id}"
    request.link_name = "{link_name}"
    listInputsConfigs = [
        Input(
            name="linkConfig.databaseType",
            value="MYSQL"
        ),
        Input(
            name="linkConfig.host",
            value="100.94.8.163"
        ),
        Input(
            name="linkConfig.port",
            value="3306"
        ),
        Input(
            name="linkConfig.database",
            value="DB_name"
        ),
        Input(
            name="linkConfig.username",
            value="username"
        ),
        Input(
            name="linkConfig.password",
            value="DB_password"
        ),
        Input(
            name="linkConfig.fetchSize",
            value="100000"
        ),
        Input(
            name="linkConfig.usingNative",
            value="false"
        )
    ]
    listConfigsLinkConfigValues = [
        Configs(
            inputs=listInputsConfigs,
            name="linkConfig"
        )
    ]
    linkConfigValuesLinks = LinksLinkconfigvalues(
        configs=listConfigsLinkConfigValues
    )
    listLinksbody = [
        Links(
            link_config_values=linkConfigValuesLinks,
            name="mysql_link",
            creation_date=1496654788622,
            connector_name="generic-jdbc-connector",
            update_date=1496654788622,
            enabled=True
        )
    ]
    request.body = CdmCreateAndUpdateLinkReq(
        links=listLinksbody
    )
    response = client.update_link(request)
```



```
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

Modifying a link named **mysql_link**

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    cdm "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/cdm/v1/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    projectId := "{project_id}"

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        WithProjectId(projectId).
        Build()

    client := cdm.NewCdmClient(
        cdm.CdmClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateLinkRequest{}
    request.ClusterId = "{cluster_id}"
    request.LinkName = "{link_name}"
    valueInputs:= "MYSQL"
    var valueInputsInterface interface{} = valueInputs
    valueInputs1:= "100.94.8.163"
    var valueInputsInterface1 interface{} = valueInputs1
    valueInputs2:= "3306"
    var valueInputsInterface2 interface{} = valueInputs2
    valueInputs3:= "DB_name"
    var valueInputsInterface3 interface{} = valueInputs3
    valueInputs4:= "username"
    var valueInputsInterface4 interface{} = valueInputs4
    valueInputs5:= "DB_password"
    var valueInputsInterface5 interface{} = valueInputs5
    valueInputs6:= "100000"
    var valueInputsInterface6 interface{} = valueInputs6
    valueInputs7:= "false"
    var valueInputsInterface7 interface{} = valueInputs7
    var listInputsConfigs = []model.Input{
        {
            Name: "linkConfig.databaseType",
            Value: &valueInputsInterface,
        },
        {
            Name: "linkConfig.host",
```



```
            Value: &valueInputsInterface1,
        },
        {
            Name: "linkConfig.port",
            Value: &valueInputsInterface2,
        },
        {
            Name: "linkConfig.database",
            Value: &valueInputsInterface3,
        },
        {
            Name: "linkConfig.username",
            Value: &valueInputsInterface4,
        },
        {
            Name: "linkConfig.password",
            Value: &valueInputsInterface5,
        },
        {
            Name: "linkConfig.fetchSize",
            Value: &valueInputsInterface6,
        },
        {
            Name: "linkConfig.usingNative",
            Value: &valueInputsInterface7,
        },
    }
    var listConfigsLinkConfigValues = []model.Configs{
        {
            Inputs: listInputsConfigs,
            Name: "linkConfig",
        },
    }
    linkConfigValuesLinks := &model.LinksLinkconfigvalues{
        Configs: listConfigsLinkConfigValues,
    }
    creationDateLinks:= int64(1496654788622)
    updateDateLinks:= int64(1496654788622)
    enabledLinks:= true
    var listLinksbody = []model.Links{
        {
            LinkConfigValues: linkConfigValuesLinks,
            Name: "mysql_link",
            CreationDate: &creationDateLinks,
            ConnectorName: "generic-jdbc-connector",
            UpdateDate: &updateDateLinks,
            Enabled: &enabledLinks,
        },
    }
    request.Body = &model.CdmCreateAndUpdateLinkReq{
        Links: listLinksbody,
    }
    response, err := client.UpdateLink(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in **API Explorer**. SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Request succeeded.
400	Request error.
401	Authentication failed.
403	No operation permissions.
404	No resources found.
500	Internal service error. For details about the returned error code, see Error Codes.
503	Service unavailable.

Error Codes

See **Error Codes**.

6 Public Data Structures

6.1 Link Parameter Description

6.1.1 Link to a Relational Database

Description

By creating a JDBC link, you can extract data from or load data to the following relational databases:

- RDS for PostgreSQL
- RDS for SQL Server
- PostgreSQL
- Microsoft SQL Server

Sample Link

```
{
   "links": [
      {
         "link-config-values": {
            "configs": [
               {
                  "inputs": [
                     {
                        "name": "linkConfig.databaseType",
                        "value": "MYSQL"
                     },
                     {
                        "name": "linkConfig.host",
                        "value": "10.120.205.30"
                     },
                     {
                        "name": "linkConfig.port",
                        "value": "3306"
                     },
                     {
                        "name": "linkConfig.database",
                        "value": "DB_name"
                     },
                     {
```



```
                        "name": "linkConfig.username",
                        "value": "username"
                    },
                    {
                        "name": "linkConfig.password",
                        "value": "Add password here"
                    },
                    {
                        "name": "linkConfig.fetchSize",
                        "value": "100000"
                    },
                    {
                        "name": "linkConfig.commitSize",
                        "value": "10000"
                    },
                    {
                        "name": "linkConfig.usingNative",
                        "value": "false"
                    },
                    {
                        "name": "linkConfig.useSSL",
                        "value": "false"
                    }
                ],
                "name": "linkConfig"
            }
        ]
    },
    "name": "mysql_link",
    "connector-name": "generic-jdbc-connector"
  }
 ]
}
```

Link Parameters

Parameter	Mandatory	Type	Description
linkConfig.databaseType	Yes	Enumeration	Database type. The options are as follows: • ORACLE • MYSQL • SQLSERVER • DB2 • POSTGRESQL • DWS • DDM • SAP HANA
linkConfig.host	Yes	String	IP address of the database server
linkConfig.port	Yes	String	Port number of the database server

Parameter	Mandatory	Type	Description
linkConfig.databaseconfig	No	Enumeration	Oracle database link type. This parameter is available only when an Oracle link is created. The options are as follows: • **SERVICENAME**: Use **SERVICE_NAME** to connect to the Oracle database. • **SID**: Use **SID** to connect to the Oracle database.
linkConfig.sidname	No	String	Oracle instance ID, which is used to differentiate databases by instances. This parameter is available only when an Oracle link is created and the database link type **linkConfig.databaseconfig** is set to **SID**.
linkConfig.database	No	String	Database name
linkConfig.username	Yes	String	Username
linkConfig.password	Yes	String	Password
linkConfig.fetchSize	No	String	Number of data rows obtained each time
linkConfig.commitSize	No	String	Number of data rows submitted in each request

Parameter	Mandatory	Type	Description
linkConfig.usingNative	No	Boolean	Whether to use the local API acceleration function of the database When creating a MySQL link, you can use the LOAD DATA function of MySQL to accelerate data import and improve the performance of importing data to the MySQL database. **NOTE** The REPLACE and IGNORE modifiers are used to process the injected rows that have the same unique key value (PRIMARY KEY or UNIQUE index value) as an existing row. • If **Constraint Conflict Handling** is set to **replace into** or **local** and the REPLACE modifier is used, a new row with the same unique key value as an existing row will replace the existing row. • If **Constraint Conflict Handling** is set to **insert into**, the IGNORE modifier is used by default, and a new row with the same unique key value as an existing row will be discarded, but the task will not be terminated. The MySQL service cannot stop file transfer in progress. Therefore, the number of written records displayed on the CDM console is inconsistent with the actual number of updated rows. For details about the MySQL local mode, visit https://dev.mysql.com/doc/refman/8.0/en/load-data.html.
linkConfig.isRds	No	Boolean	Whether RDS is supported. The default value **true** is used for cloud databases and **false** is used for other databases.

Parameter	Mandatory	Type	Description
linkConfig.useSSL	No	Boolean	Whether to enable SSL encrypted transmission for RDS. This parameter is available only when you create a DWS connection.
linkConfig.jdbcProperties	No	Map	Link attribute, which specifies the JDBC connector attributes of the data source. For details about how to configure the link attributes, see the JDBC connector description of the corresponding database.
linkConfig.version	No	Enumeration	Oracle database version. This parameter is available only when you create an Oracle link. The options are as follows: • **HIGH_VERSION**: Select this value if the Oracle database version is later than 12.1. • **MED_VERSION**: Select this value if the Oracle database version is 12.1. • **LOW_VERSION**: Select this value if the Oracle database version is earlier than 12.1. If error message "java.sql.SQLException: Protocol violation" is displayed, select another option.
dialect.identifierEnclose	No	String	Reference identifier, which is the delimiter between the referenced table names or column names. For details, see the product documentation of the corresponding database.

6.1.2 Link to OBS

Description

By creating an OBS link, you can extract files from or load files to Object Storage Service (OBS). Files in CSV, JSON, and binary format are supported.

Sample Link

The following is the message body of a sample link. You are advised to store the AK and SK in ciphertext in the configuration file or environment variables and decrypt them when needed to ensure security.

```
{
   "links": [
      {
         "link-config-values": {
            "configs": [
               {
                  "inputs": [
                     {
                        "name": "linkConfig.storageType",
                        "value": "OBS"
                     },
                     {
                        "name": "linkConfig.obsBucketType",
                        "value": "PFS"
                     },
                     {
                        "name": "linkConfig.server",
                        "value": "10.121.16.183"
                     },
                     {
                        "name": "linkConfig.port",
                        "value": "443"
                     },
                     {
                        "name":"linkConfig.accessKey",
                        "value": "<YOUR AK>"
                     },
                     {
                        "name":"linkConfig.securityKey",
                        "value":"<YOUR SK>"
                     }
                  ],
                  "name": "linkConfig"
               }
            ]
         },
         "name": "obs_link",
         "connector-name": "obs-connector"
      }
   ]
}
```

Link Parameters

Parameter	Mandatory	Type	Description
linkConfig.storageType	Yes	String	Storage class of an object

Parameter	Mandatory	Type	Description
linkConfig.obsBucketType	No	String	OBS bucket type • **OB**: object bucket • **PFS**: parallel file system **NOTE** A parallel file system is recommended because filtering files in an object bucket is slow.
linkConfig.server	Yes	String	Endpoint of the OBS server.
linkConfig.port	Yes	String	Data transmission port. The HTTPS port number is 443 and the HTTP port number is 80.
linkConfig.accessKey	Yes	String	AK. You are advised to store it in ciphertext in the configuration file or an environment variable and decrypt it when needed to ensure security.
linkConfig.securityKey	Yes	String	SK. You are advised to store it in ciphertext in the configuration file or an environment variable and decrypt it when needed to ensure security.

6.1.3 Link to HDFS

Description

By creating an HDFS link, you can extract files from or load files to MRS, FusionInsight HD, or Apache Hadoop. Files in CSV, Parquet, and binary formats are supported.

Sample Link

```
{
   "links": [
      {
         "link-config-values": {
            "configs": [
               {
                  "inputs": [
                     {
                        "name": "linkConfig.hadoopType",
                        "value": "FusionInsight HD"
                     },
                     {
                        "name": "linkConfig.host",
```



```
                        "value": "10.120.205.143"
                    },
                    {
                        "name": "linkConfig.casPort",
                        "value": "20009"
                    },
                    {
                        "name": "linkConfig.port",
                        "value": "28443"
                    },
                    {
                        "name": "linkConfig.authType",
                        "value": "KERBEROS"
                    },
                    {
                        "name": "linkConfig.user",
                        "value": "admin"
                    },
                    {
                        "name": "linkConfig.password",
                        "value": "Add password here"
                    },
                    {
                        "name": "linkConfig.runMode",
                        "value": "STANDALONE"
                    }
                ],
                "name": "linkConfig"
            }
        ]
    },
    "name": "hdfslink",
    "connector-name": "hdfs-connector"
  }
 ]
}
```

Link Parameters

Parameter	Mandatory	Type	Description
linkConfig.hadoopType	Yes	Enumeration	Hadoop type. The options are as follows: • **MRS**: link to HDFS of MRS • **FusionInsight HD**: link to HDFS of FusionInsight HD • **Apache Hadoop**: link to HDFS of Apache Hadoop
linkConfig.uri	No	String	NameNode URI required for the link to Apache Hadoop. The format is *ip:port*.
linkConfig.host	No	String	IP address of Manager required for the link to MRS or FusionInsight HD
linkConfig.port	No	String	Port number of Manager required for the link to FusionInsight HD

Parameter	Mandatory	Type	Description
linkConfig.casPort	No	String	Port number of CAS Server that connects to FusionInsight HD required for the link to FusionInsight HD
linkConfig.user	No	String	Username for logging in to Manager. This parameter is not required if the cluster configuration is used.
linkConfig.password	No	String	Password for logging in to Manager. This parameter is not required if the cluster configuration is used.
linkConfig.authType	No	Enumeration	Authentication method. The options are as follows: • **Simple**: for non-security mode • **Kerberos**: for security mode
linkConfig.principal	No	String	Account **principal** required for Kerveros authentication. You can contact the administrator to obtain the account. Before using the cluster configuration, you must set this parameter in cluster configuration management.
linkConfig.keytab	No	FileContent	Local absolute path of the **keytab** file required for Kerveros authentication. You can also contact the administrator to obtain the file. Before using the cluster configuration, you must set this parameter in cluster configuration management.

Parameter	Mandatory	Type	Description
linkConfig.runMode	No	Enumeration	Running mode of the HDFS link. The options are as follows: • **EMBEDDED**: The link instance runs with CDM. This mode delivers better performance. • **STANDALONE**: The link instance runs in an independent process. If CDM needs to connect to multiple Hadoop data sources (MRS, Hadoop, or CloudTable) with both Kerberos and Simple authentication methods, **STANDALONE** is used. If **STANDALONE** is selected, CDM can migrate data between HDFSs of multiple MRS clusters. If a CDM cluster connects to two or more clusters with Kerberos authentication enabled and the same realm, only one cluster can be connected in **EMBEDDED** mode, and the other clusters must be connected in **STANDALONE** mode.
linkConfig.properties	No	Map	Properties. You can add configuration properties of the client. Each property must contain a name and a value.

6.1.4 Link to HBase

Description

By creating an HBase link, you can extract data from or load data to HBase of MRS, FusionInsight HD, and Apache Hadoop.

Sample Link

```
{
  "links": [
    {
```



```
    "link-config-values": {
      "configs": [
        {
          "inputs": [
            {
              "name": "linkConfig.hbaseType",
              "value": "MRS"
            },
            {
              "name": "linkConfig.host",
              "value": "192.168.0.34"
            },
            {
              "name": "linkConfig.user",
              "value": "zephyr"
            },
            {
              "name": "linkConfig.password",
              "value": "Add password here."
            },
            {
              "name": "linkConfig.authType",
              "value": "KERBEROS"
            },
            {
              "name": "linkConfig.serviceType",
              "value": "HDFS"
            },
            {
              "name": "linkConfig.hBaseVersion",
              "value": "HBASE_2_X"
            },
            {
              "name": "linkConfig.runMode",
              "value": "EMBEDDED"
            }
          ],
          "name": "linkConfig"
        }
      ],
      "extended-configs": {
        "name": "linkConfig.extendedFields",
        "value":
"eyL1c2VDbHVzdGVyQ29uZmlnIjoiZmFsc2UiJCLjbHVzdGVyQ29uZmlnUHLpbmNpcGFsIjoiemVwaHlyIn0="
      }
    },
    "name": "mrs_hbase_dlf",
    "connector-name": "hbase-connector"
  }
 ]
}
```


Link Parameters

Parameter	Mandatory	Type	Description
linkConfig.hbaseType	Yes	Enumeration	HBase type. The options are as follows: • **CloudTable**: link to CloudTable Service (CloudTable) • **MRS**: link to HBase of MRS • **FusionInsight HD**: link to HBase of FusionInsight HD • **Apache Hadoop**: link to HBase of Apache Hadoop
linkConfig.uri	No	String	NameNode URI required for the link to Apache Hadoop. The format is *ip:port*.
llinkConfig.host	No	String	IP address of Manager required for the link to MRS or FusionInsight HD
linkConfig.port	No	String	Port number of Manager required for the link to FusionInsight HD
linkConfig.casPort	No	String	Port number of CAS Server that connects to FusionInsight HD required for the link to FusionInsight HD
linkConfig.hBaseVersion	Yes	Enumeration	HBase version • HBASE_1_X • HBASE_2_X
linkConfig.user	No	String	Username for logging in to Manager. This parameter is not required if the cluster configuration is used.
linkConfig.password	No	String	Password for logging in to Manager. This parameter is not required if the cluster configuration is used.
linkConfig.authType	No	Enumeration	Authentication method. The options are as follows: • **Simple**: for non-security mode • **Kerberos**: for security mode

Parameter	Mandatory	Type	Description
linkConfig.principal	No	String	Account **principal** required for Kerveros authentication. You can contact the administrator to obtain the account.
linkConfig.keytab	No	FileContent	Local absolute path of the **keytab** file required for Kerveros authentication. You can also contact the administrator to obtain the file.
linkConfig.serviceType	No	String	Service type Currently, HDFS and HBase are supported.
linkConfig.runMode	No	Enumeration	This parameter is used only when the HBase version is **HBase_2_X**. Possible values are: • **EMBEDDED**: The link instance runs with CDM. This mode delivers better performance. • **STANDALONE**: The link instance runs in an independent process. If CDM needs to connect to multiple Hadoop data sources (MRS, Hadoop, or CloudTable) with both Kerberos and Simple authentication methods, **STANDALONE** is used. If **STANDALONE** is selected, CDM can migrate data between HDFSs of multiple MRS clusters. If a CDM cluster connects to two or more clusters with Kerberos authentication enabled and the same realm, only one cluster can be connected in **EMBEDDED** mode, and the other clusters must be connected in **STANDALONE** mode.
linkConfig.properties	No	Map	Properties. You can add configuration properties of the client. Each property must contain a name and a value.

6.1.5 Link to CloudTable

Description

By creating a CloudTable link, you can extract data from or load data to CloudTable.

Sample Link

The following is the message body of a sample link. You are advised to store the AK and SK in ciphertext in the configuration file or environment variables and decrypt them when needed to ensure security.

```
{
  "links": [
    {
      "link-config-values": {
        "configs": [
          {
            "inputs": [
              {
                "name": "linkConfig.hbaseType",
                "value": "CloudTable"
              },
              {
                "name": "linkConfig.zookeeperQuorum",
                "value": "cloudtable-pass-zk2-bae54VGN.cloudtable.com:2181,cloudtable-pass-zk1-
Fu828so2.cloudtable.com:2181"
              },
              {
                "name": "linkConfig.iamAuth",
                "value": "true"
              },
              {
                "name": "linkConfig.cloudtableUser",
                "value": "zane"
              },
              {
                "name": "linkConfig.accessKey",
                "value": "<YOUR AK>"
              },
              {
                "name": "linkConfig.securityKey",
                "value": "<YOUR SK>"
              },
              {
                "name": "linkConfig.runMode",
                "value": "EMBEDDED"
              }
            ],
            "name": "linkConfig"
          }
        ]
      },
      "name": "cloudtablelink",
      "connector-name": "hbase-connector"
    }
  ]
}
```


Link Parameters

Parameter	Mandatory	Type	Description
linkConfig.hbaseType	Yes	Enumeration	HBase type. The options are as follows: • **CloudTable**: link to CloudTable • **MRS**: link to MRS • **FusionInsight HD**: link to FusionInsight HD • **Apache Hadoop**: link to Apache Hadoop
linkConfig.zookeeperQuorum	Yes	String	ZooKeeper link of CloudTable. This parameter is mandatory for the **CloudTable** link.
linkConfig.iamAuth	Yes	Boolean	If you choose IAM for identity authentication, enter the username, AK, and SK.
linkConfig.runMode	Yes	Enumeration	Running mode of the HBase link. The options are as follows: • **EMBEDDED**: The link instance runs with CDM. This mode delivers better performance. • **STANDALONE**: The link instance runs in an independent process. If CDM needs to connect to multiple Hadoop data sources (MRS, Hadoop, or CloudTable) with both Kerberos and Simple authentication methods, **STANDALONE** is used.
linkConfig.cloudtableUser	Yes	String	Username for accessing the CloudTable cluster
linkConfig.accessKey	Yes	String	AK for accessing the CloudTable cluster. You are advised to store it in ciphertext in the configuration file or an environment variable and decrypt it when needed to ensure security.

Parameter	Mandatory	Type	Description
linkConfig.securityKey	Yes	String	SK for accessing the CloudTable cluster. You are advised to store it in ciphertext in the configuration file or an environment variable and decrypt it when needed to ensure security.

6.1.6 Link to Hive

Description

By creating a Hive link, you can extract data from or load data to Hive of MRS.

Sample Link

The following is the message body of a sample link. You are advised to store the AK and SK in ciphertext in the configuration file or environment variables and decrypt them when needed to ensure security.

```
{
  "links": [
    {
      "link-config-values": {
        "configs": [
          {
            "inputs": [
              {
                "name": "linkConfig.host",
                "value": "10.120.205.230"
              },
              {
                "name": "linkConfig.authType",
                "value": "KERBEROS"
              },
              {
                "name": "linkConfig.user",
                "value": "cdm"
              },
              {
                "name": "linkConfig.password",
                "value": "Add password here"
              }
            ],
            "name": "linkConfig"
          }
        ]
      },
      "name": "hive_link",
      "connector-name": "hive-connector"
    }
  ]
}
```


Link Parameters

Parameter	Mandatory	Type	Description
llinkConfig.host	Yes	String	IP address of MRS Manager
linkConfig.authType	Yes	Enumeration	Authentication method of MRS. The options are as follows: • **SIMPLE**: for non-security mode • **KERBEROS**: for security mode
linkConfig.principal	No	String	Account **principal** required for Kerveros authentication. You can contact the administrator to obtain the account. Before using the cluster configuration, you must set this parameter in cluster configuration management.
linkConfig.keytab	No	FileContent	Local absolute path of the **keytab** file required for Kerveros authentication. You can also contact the administrator to obtain the file. Before using the cluster configuration, you must set this parameter in cluster configuration management.
linkConfig.hiveVersion	Yes	Enumeration	Hive version • HIVE_1_X • HIVE_3_X
linkConfig.user	No	String	Username for logging in to Manager. This parameter is not required if the cluster configuration is used.
linkConfig.password	No	String	Password for logging in to Manager. This parameter is not required if the cluster configuration is used.
linkConfig.uri	No	String	NameNode URI required for the link to Apache Hadoop. The format is *ip:port*.

Parameter	Mandatory	Type	Description
linkConfig.hiveMsUris	No	String	Hive metadata address for connecting to Apache Hadoop. For details, see the **hive.metastore.uris** configuration item. Example: thrift://host-192-168-1-212:9083
linkConfig.obsSupport	Yes	Boolean	The server must support OBS storage. When creating a Hive table, you can store the table in OBS.
linkConfig.runMode	Yes	Enumeration	This parameter is used only when the Hive version is **HIVE_3**_*X*. Possible values are: • **EMBEDDED**: The link instance runs with CDM. This mode delivers better performance. • **STANDALONE**: The link instance runs in an independent process. If you want to connect CDM to multiple Hadoop data sources (MRS, Hadoop, or CloudTable), and both **KERBEROS** and **SIMPLE** authentication modes are available, you must select **STANDALONE** for this parameter. Note: The **STANDALONE** mode is used to solve the version conflict problem. If the connector versions of the source and destination ends of the same link are different, a JAR file conflict occurs. In this case, you need to place the source or destination end in the STANDALONE process to prevent the migration failure caused by the conflict.
linkConfig.accessKey	No	String	AK. This parameter is mandatory when OBS is supported. You are advised to store it in ciphertext in the configuration file or an environment variable and decrypt it when needed to ensure security.

Parameter	Mandatory	Type	Description
linkConfig.securityKey	No	String	SK. This parameter is mandatory when OBS is supported. You are advised to store it in ciphertext in the configuration file or an environment variable and decrypt it when needed to ensure security.
linkConfig.properties	No	Map	Properties. You can add configuration properties of the client. Each property must contain a name and a value.

6.1.7 Link to an FTP or SFTP Server

Description

By creating an FTP or SFTP link, you are able to extract files from or load files to the FTP or SFTP server. Files in CSV, JSON, and binary format are supported.

Sample Link

```
{
   "links": [
      {
         "link-config-values": {
            "configs": [
               {
                  "inputs": [
                     {
                        "name": "linkConfig.server",
                        "value": "10.120.85.167"
                     },
                     {
                        "name": "linkConfig.port",
                        "value": "22"
                     },
                     {
                        "name": "linkConfig.username",
                        "value": "username"
                     },
                     {
                        "name": "linkConfig.password",
                        "value": "Add password here"
                     }
                  ],
                  "name": "linkConfig"
               }
            ]
         },
         "name": "sftp_link",
         "connector-name": "sftp-connector"
      }
   ]
}
```


Link Parameters

Parameters for creating the FTP or SFTP link are the same.

Parameter	Mandatory	Type	Description
linkConfig.server	Yes	String	IP address of the FTP or SFTP server
linkConfig.port	Yes	String	Port number of the FTP or SFTP server
linkConfig.username	Yes	String	Username for logging in to the FTP or SFTP server
linkConfig.password	Yes	String	Password for logging in to the FTP or SFTP server

6.1.8 Link to MongoDB

Description

By creating a Link to MongoDB, you can extract data from or load data to MongoDB.

Sample Link

```
{
   "links": [
      {
         "link-config-values": {
            "configs": [
               {
                  "inputs": [
                     {
                        "name": "linkConfig.serverList",
                        "value": "10.120.84.149:27017"
                     },
                     {
                        "name": "linkConfig.database",
                        "value": "DB_name"
                     },
                     {
                        "name": "linkConfig.userName",
                        "value": "username"
                     },
                     {
                        "name": "linkConfig.password",
                        "value": "Add password here"
                     }
                  ],
                  "name": "linkConfig"
               }
            ]
         },
         "name": "mongo_link",
         "connector-name": "mongodb-connector"
      }
   ]
}
```

Link Parameters

Parameter	Mandatory	Type	Description
linkConfig.serverList	Yes	String	Server IP address list in *host1:port1;host2:port2* format
linkConfig.database	Yes	String	Name of the MongoDB database
linkConfig.userName	Yes	String	Username for logging in to the MongoDB server
linkConfig.password	Yes	String	Password for logging in to the MongoDB server

6.1.9 Link to Redis

Description

By creating a Redis link, you can extract data from or load data to the Redis server.

Sample Link

```
{
   "links": [
      {
         "link-config-values": {
            "configs": [
               {
                  "inputs": [
                     {
                        "name": "linkConfig.deploymentMode",
                        "value": "Cluster"
                     },
                     {
                        "name": "linkConfig.serverlist",
                        "value": "10.120.84.149:7300"
                     },
                     {
                        "name": "linkConfig.password",
                        "value": "Add password here"
                     },
                     {
                        "name": "linkConfig.dbIndex",
                        "value": "0"
                     }
                  ],
                  "name": "linkConfig"
               }
            ]
         },
         "name": "redis_link",
         "connector-name": "redis-connector"
      }
   ]
}
```

Link Parameters

Parameter	Mandatory	Type	Description
linkConfig.deploymentMode	Yes	Enumeration	Redis deployment mode. The options are as follows: • **Single**: single-node deployment • **Cluster**: cluster deployment
linkConfig.serverlist	Yes	String	Server IP address list in *host1:port1;host2:port2* format
linkConfig.password	Yes	String	Password for logging in to the Redis server
linkConfig.dbIndex	Yes	String	Redis database index

6.1.10 Link to Kafka

Description

By creating a Kafka link, you are able to access open source Kafka and migrate data from Kafka to other data sources as required. Currently, only data export from Kafka is supported.

Sample Link

```
{
"links": [
  {
    "link-config-values": {
      "configs": [
        {
          "inputs": [
            {
              "name": "linkConfig.hadoopType",
              "value": "MRS"
            },
            {
              "name": "linkConfig.host",
              "value": "192.168.1.147"
            },
            {
              "name": "linkConfig.user",
              "value": "liuhuan1"
            },
            {
              "name": "linkConfig.password",
              "value": "Add password here."
            },
            {
              "name": "linkConfig.authType",
              "value": "KERBEROS"
            }
          ],
          "name": "linkConfig"
        }
```



```
      ],
      "extended-configs": {
        "name": "linkConfig.extendedFields",
        "value": "e30="
      }
    },
    "name": "mrs_kafka_link",
    "connector-name": "kafka-connector"
  }
 ]
}
```

Link Parameters

Parameter	Mandatory	Type	Description
linkConfig.hadoopType	Yes	Enumeration	Hadoop type. The options are as follows: ● **MRS**: link to Kafka of MRS ● **Apache Kafka**: link to Kafka of Apache
linkConfig.brokerList	Yes	String	This parameter is mandatory for Apache Kafka links. Kafka broker list in *host1:port1,host2:port2* format
llinkConfig.host	Yes	String	Floating IP address of MRS Manager. Click **Select** next to the **Manager IP** text box to select an MRS cluster. CDM automatically fills in the authentication information.
linkConfig.user	Yes	String	Username used for logging in to MRS Manager.
linkConfig.password	Yes	String	Password used for logging in to MRS Manager.
linkConfig.authType	Yes	Enumeration	Authentication method. The options are as follows: ● **Simple**: for non-security mode ● **Kerberos**: for security mode
linkConfig.properties	No	Map	Properties. You can add configuration properties of the client. Each property must contain a name and a value.

6.1.11 Link to DIS

Description

By creating a DIS link, you can access DIS and migrate data from DIS to other data sources as required.

Sample Link

```
{
   "links": [
      {
         "link-config-values": {
            "configs": [
               {
                  "inputs": [
                     {
                        "name": "linkConfig.region",
                        "value": "Region"
                     },
                     {
                        "name": "linkConfig.endpoint",
                        "value": "https://dis.ap-southeast-1.myhuaweiclouds.com"
                     },
                     {
                        "name": "linkConfig.ak",
                        "value": "RSO6TTEZMJ6TTFBBAACE"
                     },
                     {
                        "name": "linkConfig.sk",
                        "value": "Add password here"
                     },
                     {
                        "name": "linkConfig.projectId",
                        "value": "11d4d5af17c84660bc90b6631327d7c7"
                     }
                  ],
                  "name": "linkConfig"
               }
            ]
         },
         "name": "dis_link",
         "connector-name": "dis-connector"
      }
   ]
}
```

Link Parameters

Parameter	Mandatory	Type	Description
linkConfig.region	Yes	String	Region where DIS resides
linkConfig.endpoint	Yes	String	URL of DIS in the format of https://*Endpoint*
linkConfig.ak	Yes	String	AK for accessing the DIS server
linkConfig.sk	Yes	String	SK for accessing the DIS server

Parameter	Mandatory	Type	Description
linkConfig.projectId	Yes	String	Project ID. For details about how to obtain the project ID, see **Project ID and Account ID**.

6.1.12 Link to Elasticsearch/Cloud Search Service

Description

By creating an Elasticsearch link, you can extract data from or load data to the Elasticsearch server or Cloud Search Service.

Sample Link

```
{
  "links": [
    {
      "link-config-values": {
        "configs": [
          {
            "inputs": [
              {
                "name": "linkConfig.host",
                "value": "192.168.0.50:9200;192.168.0.62:9200"
              },
              {
                "name": "linkConfig.safemode",
                "value": "true"
              },
              {
                "name": "linkConfig.user",
                "value": "admin"
              },
              {
                "name": "linkConfig.password",
                "value": "Add password here."
              },
              {
                "name": "linkConfig.linkType",
                "value": "CSS"
              }
            ],
            "name": "linkConfig"
          }
        ],
        "extended-configs": {
          "name": "linkConfig.extendedFields",
          "value": "eyLodHRwc0FjY2VzcyI6InRydWUifQ=="
        }
      },
      "name": "css-cdm-autotest-nodel",
      "connector-name": "elasticsearch-connector"
    }
  ]
}
```

Link Parameters

Parameter	Mandatory	Type	Description
linkConfig.host	Yes	String	List of one or more Elasticsearch servers, including the port number. The format is *ip:port*. Use semicolons (;) to separate multiple IP addresses. For example, **192.168.0.1:9200;192.168.0.2:9200**.
linkConfig.safemode	No	Boolean	If you select the security authentication mode, you must enter the username and password, and choose whether to enable HTTPS access.
linkConfig.user	No	String	For Elasticsearch that supports username and password authentication, configure the username and password when creating a link.
linkConfig.password	No	String	Password for accessing the Elasticsearch server
linkConfig.linkType	Yes	String	Link type, which is used to distinguish the Elasticsearch link from the Cloud Search Service link

6.1.13 Link to DLI

Description

By creating a DLI link, you can import data to DLI. Currently, you cannot export data from DLI using CDM.

Sample Link

```
{
   "links": [
      {
         "link-config-values": {
            "configs": [
               {
                  "inputs": [
                     {
                        "name": "linkConfig.ak",
                        "value": "GRC2WR0IDC6NGROYLWU2"
                     },
                     {
                        "name": "linkConfig.sk",
                        "value": "Add password here"
```

```
                    },
                    {
                        "name": "linkConfig.region",
                        "value": "ap-southeast-1"
                    },
                    {
                        "name": "linkConfig.projectId",
                        "value": "c48475ce8e174a7a9f775706a3d5ebe2"
                    }
                ],
                "name": "linkConfig"
            }
        ]
    },
    "name": "dli",
    "connector-name": "dli-connector"
  }
 ]
}
```

Link Parameters

Parameter	Mandatory	Type	Description
linkConfig.ak	Yes	String	AK for accessing the DLI database
linkConfig.sk	Yes	String	SK for accessing the DLI database
linkConfig.region	Yes	String	Region where DLI resides
linkConfig.projectId	Yes	String	Project ID of the DLI service

6.1.14 Link to CloudTable OpenTSDB

Description

By creating an OpenTSDB link, you can extract data from and load data to CloudTable OpenTSDB.

Sample Link

```
{
 "links": [
  {
   "link-config-values": {
    "configs": [
     {
      "inputs": [
       {
        "name": "linkConfig.openTSDBQuorum",
        "value": "opentsdb-sp8afz7bgbps5ur.cloudtable.com:4242"
       },
       {
        "name": "linkConfig.securityMode",
        "value": "UNSAFE"
       }
```



```
          ],
          "name": "linkConfig"
        }
      ]
    },
    "name": "opentsdb",
    "connector-name": "opentsdb-connector"
  }
 ]
}
```

Link Parameters

Parameter	Mandatory	Type	Description
linkConfig.openTSDBQuorum	Yes	String	ZooKeeper Link of OpenTSDB
linkConfig.securityMode	Yes	String	Security or non-security mode If you select **Security**, enter the project ID, username, and AK/SK.
linkConfig.user	No	String	Username for accessing CloudTable
linkConfig.ak	No	String	AK for accessing CloudTable
linkConfig.sk	No	String	SK for accessing CloudTable
linkConfig.projectId	No	String	Project ID of CloudTable

6.1.15 Link to DMS Kafka

Description

By creating a DMS Kafka link, you can connect to Kafka Basic or Kafka Platinum on DMS. Currently, you can only export data from DMS Kafka to Cloud Search Service.

Sample Link

```
{
 "links": [
  {
   "link-config-values": {
    "configs": [
     {
      "inputs": [
       {
        "name": "linkConfig.kafkaType",
        "value": "Platinum"
       },
       {
        "name": "linkConfig.brokerList",
        "value": "100.85.121.112:9094,100.85.220.134:9094,100.85.127.232:9094"
```



```
          },
          {
            "name": "linkConfig.isPlatinumInstance",
            "value": "false"
          }
        ],
        "name": "linkConfig"
      }
    ],
    "extended-configs": {
      "name": "linkConfig.extendedFields",
      "value": "e30="
    }
  },
  "name": "dms_kafka",
  "connector-name": "dms-kafka-connector"
 }
]
}
```

Link Parameters

Parameter	Mandatory	Type	Description
linkConfig.kafkaType	Yes	Enumeration	DMS Kafka edition. Currently, only the Platinum edition is available. • **Basic**: DMS Kafka Basic • **Platinum**: DMS Kafka Platinum
linkConfig.brokerList	Yes	String	The DMS endpoint is in **host1:port1,host2:port2** format.
linkConfig.isPlatinumInstance	Yes	Boolean	Whether to enable SSL authentication when a client connects to a Kafka premium instance. If Kafka SASL_SSL is enabled, data will be encrypted before transmission for higher security, but performance will suffer.
linkConfig.user	No	String	Username for connecting to DMS Kafka. This parameter is displayed when **Kafka SASL_SSL** is enabled.

Parameter	Mandatory	Type	Description
linkConfig.password	No	String	Password for connecting to DMS Kafka. This parameter is displayed when **Kafka SASL_SSL** is enabled.

6.2 Source Job Parameters

6.2.1 From a Relational Database

Sample JSON File

```
"from-config-values": {
      "configs": [
        {
          "inputs": [
            {
              "name": "fromJobConfig.useSql",
              "value": "false"
            },
            {
              "name": "fromJobConfig.schemaName",
              "value": "rf_database"
            },
            {
              "name": "fromJobConfig.tableName",
              "value": "rf_from"
            },
            {
              "name": "fromJobConfig.columnList",
              "value": "AA&BB"
            },
            {
              "name": "fromJobConfig.incrMigration",
              "value": "false"
            }
          ],
          "name": "fromJobConfig"
        }
      ]
    }
```

Parameter Description

Parameter	Mandatory	Type	Description
fromJobConfig.useSql	Yes	Boolean	Whether to use the customized SQL statement to export data when exporting relational database data
fromJobConfig.sql	No	String	Customized SQL statement. CDM executes the SQL statement to export data.

Parameter	Mandatory	Type	Description
fromJobConfig.schemaName	Yes	String	Database mode or tablespace. For example, **public**. **NOTE** The parameter value can contain wildcard characters (*), which is used to export all databases whose names start with a certain prefix or end with a certain suffix. The examples are as follows: • **SCHEMA*** indicates that all databases whose names starting with **SCHEMA** are exported. • ***SCHEMA** indicates that all databases whose names ending with **SCHEMA** are exported. • ***SCHEMA*** indicates that all databases whose names containing **SCHEMA** are exported.
fromJobConfig.tableName	Yes	String	Table name. For example, **TBL_EXAMPLE**. **NOTE** The table name can contain wildcard characters (*), which is used to export all tables whose names start with a certain prefix or end with a certain suffix. The number and types of fields in the tables must be the same. The examples are as follows: • **table*** indicates that all tables whose names starting with **table** are exported. • ***table** indicates that all tables whose names ending with **table** are exported. • ***table*** indicates that all tables whose names containing **table** are exported.
fromJobConfig.whereClause	No	String	WHERE clause used to specify the data to be extracted. If no WHERE clause is configured, the entire table will be extracted. For example, **age > 18 and age <= 60**.
fromJobConfig.columnList	No	String	List of fields to be extracted. Use **&** to separate field names. For example, **id&gid&name**.

Parameter	Mandatory	Type	Description
fromJobConfig.partitionColumn	No	String	Partition field to be extracted, by which a job is split in multiple sub-jobs executed concurrently. For example, **id**.
fromJobConfig.usePartition	No	Boolean	When data is exported from the Oracle database, data can be extracted from each partition in a partitioned table. When this function is enabled, you can use the **fromJobConfig.partitionList** parameter to specify the partitions in the Oracle table. This function does not support non-partitioned tables.
fromJobConfig.partitionList	No	String	Oracle table partitions to be migrated. Separate multiple partitions with ampersands (&). If you do not set this parameter, all partitions will be migrated.

6.2.2 From Object Storage

Sample JSON File

```
"from-config-values": {
      "configs": [
       {
         "inputs": [
           {
             "name": "fromJobConfig.bucketName",
             "value": "cdm-est"
           },
           {
             "name": "fromJobConfig.inputDirectory",
             "value": "/obsfrom/varchar.txt"
           },
           {
             "name": "fromJobConfig.inputFormat",
             "value": "CSV_FILE"
           },
           {
             "name": "fromJobConfig.columnList",
             "value": "1&2&3"
           },
           {
             "name": "fromJobConfig.fieldSeparator",
             "value": ","
           },
           {
             "name": "fromJobConfig.quoteChar",
             "value": "false"
           },
```



```
        {
          "name": "fromJobConfig.regexSeparator",
          "value": "false"
        },
        {
          "name": "fromJobConfig.firstRowAsHeader",
          "value": "false"
        },
        {
          "name": "fromJobConfig.encodeType",
          "value": "UTF-8"
        },
        {
          "name": "fromJobConfig.fromCompression",
          "value": "NONE"
        },
        {
          "name": "fromJobConfig.splitType",
          "value": "FILE"
        }
      ],
      "name": "fromJobConfig"
    }
  ]
}
```

Parameter Description

Parameter	Mandatory	Type	Description
fromJobConfig.bucketName	Yes	String	Bucket name
fromJobConfig.inputDirectory	Yes	String	Path for storing files to be extracted. You can enter a maximum of 50 file paths, which are separated by vertical bars (|). You can also customize the separators. For example, **FROM/example.csv|FROM/b.txt**.

Parameter	Mandatory	Type	Description
fromJobConfig.inputFormat	Yes	Enumeration	File format required for data transmission. Currently, the following file formats are supported: • **CSV_FILE**: CSV format, used to migrate files to data tables • **JSON_FILE**: JSON format, used to migrate files to data tables • **BINARY_FILE**: Files (even not in binary format) will be directly transferred without resolution. It is applicable to file copy. If you select **BINARY_FILE**, the migration destination must also be a file system.
fromJobConfig.lineSeparator	No	String	Lind feed character in a file. By default, the system automatically identifies **\\n**, **\ \r**, and **\\r\\n**. You can configure special characters. For spaces and carriage returns, encode them with URL. You can also configure them by editing the job JSON, in which case URL encoding is not required.
fromJobConfig.columnList	No	String	Numbers of columns to be extracted. Use **&** to separate column numbers in ascending order. For example, **1&3&5**.
fromJobConfig.regexSeparator	No	Boolean	Whether to use the regular expression to separate fields. This parameter is valid only when the file format is **CSV_FILE**.
fromJobConfig.regex	No	String	Regular expression. This parameter is valid only when the regular expression is used to separate fields.
fromJobConfig.fieldSeparator	No	String	Field delimiter. This parameter is valid only when the file format is **CSV_FILE**. The default value is **,**.

Parameter	Mandatory	Type	Description
fromJobConfig.quoteChar	No	Boolean	Whether to use the encircling symbol. If this parameter is set to **true**, the field delimiters in the encircling symbol are regarded as a part of the string value. Currently, the default encircling symbol of CDM is double quotation mark (").
fromJobConfig.firstRowAsHeader	No	Boolean	Whether to regard the first line as the heading line. This parameter is valid only when the file format is **CSV_FILE**. When you migrate a CSV file to a table, CDM writes all data to the table by default. If this parameter is set to **true**, CDM uses the first line of the CSV file as the heading line and does not write the line to the destination table.
fromJobConfig.fromCompression	No	Enumeration	Compression format. This parameter is valid only when the file format is **CSV_FILE** or **JSON**. The options are as follows: • **NONE**: Files in all formats are transferred. • **GZIP**: Files in gzip format are transferred. • **ZIP**: Files in Zip format are transferred.
fromJobConfig.jsonReferenceNode	No	String	Reference node. This parameter is valid when the file format is **JSON_FILE**. Resolve data on the JSON node. If the data corresponding to the node is a JSON array, the system extracts data from the array in the same mode. Nested JSON nodes are separated by periods (.). For example, **data.list**.
fromJobConfig.encodeType	No	String	Encoding type. For example, **UTF_8** or **GBK**.

Parameter	Mandatory	Type	Description
fromJobConfig.useMarkerFile	No	Boolean	Whether to start a job by a marker file. A job is started only when a marker file for starting the job exists in the source path. Otherwise, the job will be suspended for a period of time specified by **fromJobConfig.waitTime**.
fromJobConfig.markerFile	No	String	Name of the marker file for starting a job. After a marker file is specified, the task is executed only when the file exists in the source path. If the marker file is not specified, this function is disabled by default. For example, **ok.txt**.
fromJobConfig.waitTime	No	String	Period of waiting for a marker file. If you set **Start Job by Marker File** to **Yes** but no marker file exists in the source path, the job fails upon suspension timeout. If you set this parameter to **0** and no marker file exists in the source path, the job will fail immediately. Unit: second
fromJobConfig.filterType	No	Enumeration	Filter type. Possible values are as follows: • **WILDCARD**: Enter a wildcard character to filter paths or files. CDM will migrate the paths or files that meet the filter condition. • **TIME**: Specify a time filter. CDM will migrate the files modified after the specified time point.
fromJobConfig.pathFilter	No	String	Path filter, which is configured when the filter type is **WILDCARD**. It is used to filter the file directories. For example, ***input**.

Parameter	Mandatory	Type	Description
fromJobConfig.fileFilter	No	String	File filter, which is configured when the filter type is **WILDCARD**. It is used to filter files in the specified directory. Use commas (,) to separate multiple files. For example, ***.csv,*.txt**.
fromJobConfig.startTime	No	String	If you set **Filter Type** to **Time Filter**, and specify a point in time for this parameter, only the files modified at or after the specified time are transferred. The time format must be *yyyy-MM-dd HH:mm:ss*. This parameter can be set to a macro variable of date and time. For example, **${timestamp(dateformat(yyyy-MM-dd HH:mm:ss,-90,DAY))}** indicates that only files generated within the latest 90 days are migrated.
fromJobConfig.endTime	No	String	If you set **Filter Type** to **Time Filter**, and specify a point in time for this parameter, only the files modified before the specified time are transferred. The time format must be *yyyy-MM-dd HH:mm:ss*. This parameter can be set to a macro variable of date and time. For example, **${timestamp(dateformat(yyyy-MM-dd HH:mm:ss))}** indicates that only the files whose modification time is earlier than the current time are migrated.
fromJobConfig.fileSeparator	No	String	File separator. If you enter multiple file paths in **fromJobConfig.inputDirectory**, CDM uses the file separator to separate files. The default value is |.

Parameter	Mandatory	Type	Description
fromJobConfig.md5FileSuffix	No	String	Check whether the files extracted by CDM are consistent with those in the migration source.

6.2.3 From HDFS

Sample JSON File

```
"from-config-values": {
      "configs": [
       {
        "inputs": [
         {
          "name": "fromJobConfig.inputDirectory",
          "value": "/hdfsfrom/from_hdfs_est.csv"
         },
         {
          "name": "fromJobConfig.inputFormat",
          "value": "CSV_FILE"
         },
         {
          "name": "fromJobConfig.columnList",
          "value": "1"
         },
         {
          "name": "fromJobConfig.fieldSeparator",
          "value": ","
         },
         {
          "name": "fromJobConfig.quoteChar",
          "value": "false"
         },
         {
          "name": "fromJobConfig.regexSeparator",
          "value": "false"
         },
         {
          "name": "fromJobConfig.firstRowAsHeader",
          "value": "false"
         },
         {
          "name": "fromJobConfig.encodeType",
          "value": "UTF-8"
         },
         {
          "name": "fromJobConfig.fromCompression",
          "value": "NONE"
         },
         {
          "name": "fromJobConfig.compressedFileSuffix",
          "value": "*"
         },
         {
          "name": "fromJobConfig.splitType",
          "value": "FILE"
         },
         {
          "name": "fromJobConfig.useMarkerFile",
          "value": "false"
         },
         {
```



```
            "name": "fromJobConfig.fileSeparator",
            "value": "|"
          },
          {
            "name": "fromJobConfig.filterType",
            "value": "NONE"
          }
        ],
        "name": "fromJobConfig"
      }
    ]
  }
```

Parameter Description

- HDFS job parameter description

Parameter	Mandatory	Type	Description
fromJobConfig.inputDirectory	Yes	String	Path for storing data to be extracted. For example, **/data_dir**.
fromJobConfig.inputFormat	Yes	Enumeration	File format required for data transmission. Currently, the following file formats are supported: ● **CSV_FILE**: CSV format ● **PARQUET_FILE**: Parquet format ● **BINARY_FILE**: binary format If you select **BINARY_FILE**, the migration destination must also be a file system.
fromJobConfig.columnList	No	String	Numbers of columns to be extracted. Use **&** to separate column numbers in ascending order. For example, **1&3&5**.
fromJobConfig.lineSeparator	No	String	Lind feed character in a file. By default, the system automatically identifies **\\n**, **\\r**, and **\\r\\n**. You can configure special characters. For spaces and carriage returns, encode them with URL. You can also configure them by editing the job JSON, in which case URL encoding is not required.

Parameter	Mandatory	Type	Description
fromJobConfig.fieldSeparator	No	String	Field delimiter. This parameter is valid only when the file format is **CSV_FILE**. The default value is **,**.
fromJobConfig.quoteChar	No	Boolean	Whether to use the encircling symbol. If this parameter is set to **true**, the field delimiters in the encircling symbol are regarded as a part of the string value. Currently, the default encircling symbol of CDM is double quotation mark (").
fromJobConfig.regexSeparator	No	Boolean	Whether to use the regular expression to separate fields. This parameter is valid only when the file format is **CSV_FILE**.
fromJobConfig.encodeType	No	String	Encoding type. For example, **UTF_8** or **GBK**.
fromJobConfig.firstRowAsHeader	No	Boolean	Whether to regard the first line as the heading line. This parameter is valid only when the file format is **CSV_FILE**. When you migrate a CSV file to a table, CDM writes all data to the table by default. If this parameter is set to **true**, CDM uses the first line of the CSV file as the heading line and does not write the line to the destination table.
fromJobConfig.fromCompression	No	Enumeration	Compression format. Only the source files in specified compression format are transferred. **NONE** indicates files in all formats are transferred.

Parameter	Mandatory	Type	Description
fromJobConfig.compressedFileSuffix	No	String	Extension of the files to be decompressed. The decompression operation is performed only when the file name extension is used in a batch of files. Otherwise, files are transferred in the original format. If you enter ***** or leave the parameter blank, all files are decompressed.
fromJobConfig.splitType	No	Enumeration	Whether to split files by file or size. If HDFS files are split, each shard is regarded as a file. ● **FILE**: Split files by file quantity. If there are 10 files and **throttlingConfig.numExtractors** is set to **5**, each shard consists of two files. ● **SIZE**: Split files by file size. Files will not be split for balance. Suppose there are 10 files, among which nine are 10 MB and one is 200 MB in size. If **throttlingConfig.numExtractors** is set to **2**, two shards will be created, one for processing the nine 10 MB files, the other for processing the 200 MB file.
fromJobConfig.useMarkerFile	No	Boolean	Whether to start a job by a marker file. A job is started only when a marker file for starting the job exists in the source path. Otherwise, the job will be suspended for a period of time specified by **fromJobConfig.waitTime**.

Parameter	Mandatory	Type	Description
fromJobConfig.markerFile	No	String	Name of the marker file for starting a job. After a marker file is specified, the task is executed only when the file exists in the source path. If the marker file is not specified, this function is disabled by default. For example, **ok.txt**.
fromJobConfig.fileSeparator	No	String	File separator. If you enter multiple file paths in **fromJobConfig.inputDirectory**, CDM uses the file separator to separate files. The default value is |.
fromJobConfig.filterType	No	Enumeration	Filter type. Possible values are as follows: • **WILDCARD**: Enter a wildcard character to filter paths or files. CDM will migrate the paths or files that meet the filter condition. • **TIME**: Specify a time filter. CDM will migrate the files modified after the specified time point.
fromJobConfig.pathFilter	No	String	Path filter, which is configured when the filter type is **WILDCARD**. It is used to filter the file directories. For example, ***input**.
fromJobConfig.fileFilter	No	String	File filter, which is configured when the filter type is **WILDCARD**. It is used to filter files in the specified directory. Use commas (,) to separate multiple files. For example, ***.csv,*.txt**.

Parameter	Mandatory	Type	Description
fromJobConfig.startTime	No	String	If you set **Filter Type** to **Time Filter**, and specify a point in time for this parameter, only the files modified at or after the specified time are transferred. The time format must be *yyyy-MM-dd HH:mm:ss*. This parameter can be set to a macro variable of date and time. For example, **${timestamp(dateformat(yyyy-MM-dd HH:mm:ss,-90,DAY))}** indicates that only files generated within the latest 90 days are migrated.
fromJobConfig.endTime	No	String	If you set **Filter Type** to **Time Filter**, and specify a point in time for this parameter, only the files modified before the specified time are transferred. The time format must be *yyyy-MM-dd HH:mm:ss*. This parameter can be set to a macro variable of date and time. For example, **${timestamp(dateformat(yyyy-MM-dd HH:mm:ss))}** indicates that only the files whose modification time is earlier than the current time are migrated.
fromJobConfig.createSnapshot	No	Boolean	If this parameter is set to **true**, CDM creates a snapshot for the source directory to be migrated (the snapshot cannot be created for a single file) before it reads files from HDFS. Then CDM migrates the data in the snapshot. Only the HDFS administrator can create a snapshot. After the CDM job is completed, the snapshot is deleted.

Parameter	Mandatory	Type	Description
fromJobConfig.formats	No	Data structure	Time format. This parameter is mandatory only when **fromJobConfig.inputFormat** is set to **CSV_FILE** and the time field exists in the file. For details, see **Description of the fromJobConfig.formats parameter**.
fromJobConfig.decryption	No	Enumeration	This parameter is available only when **fromJobConfig.inputFormat** is set to **BINARY_FILE**. It specifies whether to decrypt the encrypted file before export, and the decryption method. The options are as follows: • **NONE**: Do not decrypt but directly export the file. • **AES-256-GCM**: Use the AES-256-GCM (NoPadding) algorithm to decrypt the file and then export the file.
fromJobConfig.dek	No	String	Data decryption key. The key is a string of 64-bit hexadecimal numbers and must be the same as the data encryption key **toJobConfig.dek** configured during encryption. If the encryption and decryption keys are inconsistent, the system does not report an exception, but the decrypted data is incorrect.

Parameter	Mandatory	Type	Description
fromJobConfig.iv	No	String	Initialization vector required for decryption. The initialization vector is a string of 32-bit hexadecimal numbers and must be the same as the initialization vector **toJobConfig.iv** configured during encryption. If the encryption and decryption keys are inconsistent, the system does not report an exception, but the decrypted data is incorrect.

- Description of the **fromJobConfig.formats** parameter

Parameter	Mandatory	Type	Description
name	Yes	String	Column number. For example, **1**.
value	Yes	String	Time format. For example, *yyyy-MM-dd*.

6.2.4 From Hive

Sample JSON File

```
"from-config-values": {
      "configs": [
        {
          "inputs": [
            {
              "name": "fromJobConfig.hive",
              "value": "hive"
            },
            {
              "name": "fromJobConfig.database",
              "value": "rf_database"
            },
            {
              "name": "fromJobConfig.table",
              "value": "rf_from"
            },
            {
              "name": "fromJobConfig.columnList",
              "value": "tiny&small&int&integer&bigint&float&double&timestamp&char&varchar&text"
            }
          ],
          "name": "fromJobConfig"
        }
      ]
    }
```

Parameter Description

Parameter	Mandatory	Type	Description
fromJobConfig.hive	No	String	Data source to be extracted. If the data source is Hive, set this parameter to **hive**.
fromJobConfig.database	No	String	Database from which data is extracted. For example, **default**.
fromJobConfig.table	Yes	String	Name of the table from which data is extracted. For example, **cdm**.
fromJobConfig.columnList	No	String	Numbers of columns to be extracted. Use **&** to separate column numbers in ascending order. For example, **1&3&5**.

6.2.5 From HBase/CloudTable

Sample JSON File

```
"from-config-values": {
      "configs": [
        {
          "inputs": [
            {
              "name": "fromJobConfig.table",
              "value": "rf_from"
            },
            {
              "name": "fromJobConfig.columnFamilies",
              "value": "rowkey&f"
            },
            {
              "name": "fromJobConfig.columns",
              "value": "rowkey:rowkey&f:_small"
            },
            {
              "name": "fromJobConfig.formats",
              "value": {
                "f:_date": "yyyy-MM-dd",
                "f:_timestamp": "yyyy-MM-dd HH:mm:ss"
              }
            }
          ],
          "name": "fromJobConfig"
        }
      ]
    }
```

Parameter Description

- HBase/CloudTable job parameter description

Parameter	Mandatory	Type	Description
fromJobConfig.table	Yes	String	Name of the table from which data is extracted. For example, **cdm**.
fromJobConfig.columnFamilies	No	String	Column family to which the data to be extracted belongs
fromJobConfig.columns	No	String	Columns to be extracted. Use **&** to separate column numbers and **:** to separate column families and columns. For example, **cf1:c1&cf2:c2**.
fromJobConfig.isSplit	No	Boolean	Whether to split the rowkey. For example, **true**.
fromJobConfig.delimiter	No	String	Delimiter used for splitting rowkeys. If this parameter is not set, row keys will not be split. For example, vertical bars (|).
fromJobConfig.startTime	No	String	Minimum timestamp of the time range (the time point is included). The format is *yyyy-MM-dd hh:mm:ss*. Only the data created at this point in time and later is extracted.
fromJobConfig.endTime	No	String	Maximum timestamp of the time range (the time point is not included). The format is *yyyy-MM-dd hh:mm:ss*. Only the data created before this point in time is extracted.
fromJobConfig.formats	No	Data structure	Time format. For details, see **Description of the fromJobConfig.formats parameter**.

- Description of the **fromJobConfig.formats** parameter

Parameter	Mandatory	Type	Description
name	Yes	String	Column number. For example, **1**.
value	Yes	String	Time format. For example, *yyyy-MM-dd*.

6.2.6 From FTP/SFTP

Sample JSON File

```
"from-config-values": {
      "configs": [
       {
        "inputs": [
         {
          "name": "fromJobConfig.inputDirectory",
          "value": "/sftpfrom/from_sftp.csv"
         },
         {
          "name": "fromJobConfig.inputFormat",
          "value": "CSV_FILE"
         },
         {
          "name": "fromJobConfig.columnList",
          "value": "1&2&3&4&5&6&7&8&9&10&11&12"
         },
         {
          "name": "fromJobConfig.fieldSeparator",
          "value": ","
         },
         {
          "name": "fromJobConfig.regexSeparator",
          "value": "false"
         },
         {
          "name": "fromJobConfig.firstRowAsHeader",
          "value": "false"
         },
         {
          "name": "fromJobConfig.encodeType",
          "value": "UTF-8"
         },
         {
          "name": "fromJobConfig.fromCompression",
          "value": "NONE"
         },
         {
          "name": "fromJobConfig.splitType",
          "value": "FILE"
         }
        ],
        "name": "fromJobConfig"
       }
      ]
    }
```

Parameter Description

Source link job parameters of FTP and SFTP are the same. **Table 6-1** describes the parameters.

Table 6-1 Source link job parameters of file systems

Parameter	Mandatory	Type	Description
fromJobConfig.inputDirectory	Yes	String	Path for storing files to be extracted. You can enter a maximum of 50 file paths, which are separated by vertical bars (|). You can also customize the separators. For example, **FROM/example.csv|FROM/b.txt**.
fromJobConfig.inputFormat	Yes	Enumeration	File format required for data transmission. Currently, the following file formats are supported: • **CSV_FILE**: CSV format, used to migrate files to data tables • **JSON_FILE**: JSON format, used to migrate files to data tables • **BINARY_FILE**: Files (even not in binary format) will be directly transferred without resolution. It is applicable to file copy. If you select **BINARY_FILE**, the migration destination must also be a file system.
fromJobConfig.lineSeparator	No	String	Lind feed character in a file. By default, the system automatically identifies **\\n**, **\\r**, and **\\r\\n**. You can configure special characters. For spaces and carriage returns, encode them with URL. You can also configure them by editing the job JSON, in which case URL encoding is not required.
fromJobConfig.columnList	No	String	Numbers of columns to be extracted. Use **&** to separate column numbers in ascending order. For example, **1&3&5**.
fromJobConfig.fieldSeparator	No	String	Field delimiter. This parameter is valid only when the file format is **CSV_FILE**. The default value is **,**.

Parameter	Mandatory	Type	Description
fromJobConfig.quoteChar	No	Boolean	Whether to use the encircling symbol. If this parameter is set to **true**, the field delimiters in the encircling symbol are regarded as a part of the string value. Currently, the default encircling symbol of CDM is double quotation mark (").
fromJobConfig.regexSeparator	No	Boolean	Whether to use the regular expression to separate fields. This parameter is valid only when the file format is **CSV_FILE**.
fromJobConfig.regex	No	String	Regular expression. This parameter is valid only when the regular expression is used to separate fields.
fromJobConfig.firstRowAsHeader	No	Boolean	Whether to regard the first line as the heading line. This parameter is valid only when the file format is **CSV_FILE**. When you migrate a CSV file to a table, CDM writes all data to the table by default. If this parameter is set to **true**, CDM uses the first line of the CSV file as the heading line and does not write the line to the destination table.
fromJobConfig.fromCompression	No	Enumeration	Compression format. This parameter is valid only when the file format is **CSV_FILE** or **JSON**. The options are as follows: • **NONE**: Files in all formats are transferred. • **GZIP**: Files in gzip format are transferred. • **ZIP**: Files in Zip format are transferred.

Parameter	Mandatory	Type	Description
fromJobConfig.splitType	No	Enumeration	Whether to split files by file or size. • **FILE**: Split files by file quantity. If there are 10 files and **throttlingConfig.numExtractors** is set to **5**, each shard consists of two files. • **SIZE**: Split files by file size. Files will not be split for balance. Suppose there are 10 files, among which nine are 10 MB and one is 200 MB in size. If **throttlingConfig.numExtractors** is set to **2**, two shards will be created, one for processing the nine 10 MB files, the other for processing the 200 MB file.
fromJobConfig.jsonReferenceNode	No	String	Reference node. This parameter is valid when the file format is **JSON_FILE**. Resolve data on the JSON node. If the data corresponding to the node is a JSON array, the system extracts data from the array in the same mode. Nested JSON nodes are separated by periods (.). For example, **data.list**.
fromJobConfig.encodeType	No	String	Encoding type. For example, **UTF_8** or **GBK**.
fromJobConfig.useMarkerFile	No	Boolean	Whether to start a job by a marker file. A job is started only when a marker file for starting the job exists in the source path. Otherwise, the job will be suspended for a period of time specified by **fromJobConfig.waitTime**.

Parameter	Mandatory	Type	Description
fromJobConfig.markerFile	No	String	Name of the marker file for starting a job. After a marker file is specified, the task is executed only when the file exists in the source path. If the marker file is not specified, this function is disabled by default. For example, **ok.txt**.
fromJobConfig.waitTime	No	String	Period of waiting for a marker file. If you set **Start Job by Marker File** to **Yes** but no marker file exists in the source path, the job fails upon suspension timeout. If you set this parameter to **0** and no marker file exists in the source path, the job will fail immediately. Unit: second
fromJobConfig.filterType	No	Enumeration	Filter type. Possible values are as follows: • **WILDCARD**: Enter a wildcard character to filter paths or files. CDM will migrate the paths or files that meet the filter condition. • **TIME**: Specify a time filter. CDM will migrate the files modified after the specified time point.
fromJobConfig.pathFilter	No	String	Path filter, which is configured when the filter type is **WILDCARD**. It is used to filter the file directories. For example, ***input**.
fromJobConfig.fileFilter	No	String	File filter, which is configured when the filter type is **WILDCARD**. It is used to filter files in the specified directory. Use commas (,) to separate multiple files. For example, ***.csv,*.txt**.

Parameter	Mandatory	Type	Description
fromJobConfig.startTime	No	String	If you set **Filter Type** to **Time Filter**, and specify a point in time for this parameter, only the files modified at or after the specified time are transferred. The time format must be *yyyy-MM-dd HH:mm:ss*. This parameter can be set to a macro variable of date and time. For example, **${timestamp(dateformat(yyyy-MM-dd HH:mm:ss,-90,DAY))}** indicates that only files generated within the latest 90 days are migrated.
fromJobConfig.endTime	No	String	If you set **Filter Type** to **Time Filter**, and specify a point in time for this parameter, only the files modified before the specified time are transferred. The time format must be *yyyy-MM-dd HH:mm:ss*. This parameter can be set to a macro variable of date and time. For example, **${timestamp(dateformat(yyyy-MM-dd HH:mm:ss))}** indicates that only the files whose modification time is earlier than the current time are migrated.
fromJobConfig.fileSeparator	No	String	File separator. If you enter multiple file paths in **fromJobConfig.inputDirectory**, CDM uses the file separator to separate files. The default value is **|**.
fromJobConfig.md5FileSuffix	No	String	Check whether the files extracted by CDM are consistent with those in the migration source.

6.2.7 From HTTP/HTTPS

Sample JSON File

```
"from-config-values": {
      "configs": [
        {
          "inputs": [
            {
              "name": "fromJobConfig.inputDirectory",
              "value": "http://10.114.196.186:8080/httpfrom/symbol.txt"
            },
            {
              "name": "fromJobConfig.inputFormat",
              "value": "BINARY_FILE"
            },
            {
              "name": "fromJobConfig.fromCompression",
              "value": "TARGZ"
            },
            {
              "name": "fromJobConfig.compressedFileSuffix",
              "value": "*"
            },
            {
              "name": "fromJobConfig.fileSeparator",
              "value": "|"
            }
          ],
          "name": "fromJobConfig"
        }
      ]
    }
```

Parameter Description

Parameter	Mandatory	Type	Description
fromJobConfig.inputDirectory	Yes	String	URL of the file to be extracted These connectors are used to read files with an HTTP/HTTPS URL, such as reading public files on the third-party object storage system and web disks.
fromJobConfig.inputFormat	Yes	Enumeration	File format required for data transmission. Currently, only the binary format is supported.

Parameter	Mandatory	Type	Description
fromJobConfig.fromCompression	No	Enumeration	Compression format of the source files. The options are as follows: • **NONE**: Files in all formats are transferred. • **GZIP**: Files in gzip format are transferred. • **ZIP**: Files in Zip format are transferred. • **TAR.GZ**: Files in TAR.GZ format are transferred.
fromJobConfig.compressedFileSuffix	No	String	Extension of the files to be decompressed. The decompression operation is performed only when the file name extension is used in a batch of files. Otherwise, files are transferred in the original format. If you enter ***** or leave the parameter blank, all files are decompressed.
fromJobConfig.fileSeparator	No	String	File separator. When multiple files are transferred, CDM uses the file separator to separate files. The default value is **|**.
fromJobConfig.useQuery	No	Boolean	• If you set this parameter to **true**, the name of the objects uploaded to OBS does not carry the **query** parameter. • If you set this parameter to **false**, the name of the objects uploaded to OBS carries the **query** parameter.
fromJobConfig.md5FileSuffix	No	String	Check whether the files extracted by CDM are consistent with those in the migration source.

6.2.8 From MongoDB/DDS

Sample JSON File

```
"from-config-values": {
      "configs": [
```



```
    {
      "inputs": [
        {
          "name": "fromJobConfig.database",
          "value": "cdm"
        },
        {
          "name": "fromJobConfig.collectionName",
          "value": "rf_from"
        },
        {
          "name": "fromJobConfig.columnList",
          "value": "TINYTEST&SMALLTEST&INTTEST&INTEGERTEST&BIGINTTEST&FLOATTEST"
        },
        {
          "name": "fromJobConfig.isBatchMigration",
          "value": "false"
        },
        {
          "name": "fromJobConfig.filters",
          "value": "{'last_name': 'Smith'}"
        }
      ],
      "name": "fromJobConfig"
    }
  ]
}
```

Parameter Description

Parameter	Mandatory	Type	Description
fromJobConfig.database	Yes	String	Name of the MongoDB/DDS database
fromJobConfig.collectionName	Yes	String	Name of the MongoDB/DDS collection
fromJobConfig.columnList	No	String	List of fields to be extracted. Use **&** to separate field names. For example, **id&gid&name**.
fromJobConfig.isBatchMigration	No	Boolean	Whether to migrate all data in the database

Parameter	Mandatory	Type	Description
fromJobConfig.filters	No	String	Filter condition for files. CDM migrates only the data that meets the filter condition. Examples: 1. Filter by expression: **{'last_name': 'Smith'}** indicates that all files whose **last_name** value is **Smith** are queried. 2. Filter by parameter: **{ x : "john" }, { z : 1 }** indicates that all **z** fields whose **x** is **john** are queried. 3. Filter by condition: **{ "field" : { $gt: 5 } }** indicates that the **field** values greater than 5 are queried.

6.2.9 From Redis

Sample JSON File

```
"from-config-values": {
    "configs": [
      {
        "inputs": [
          {
            "name": "fromJobConfig.isBatchMigration",
            "value": "false"
          },
          {
            "name": "fromJobConfig.keyPrefix",
            "value": "rf_string_from"
          },
          {
            "name": "fromJobConfig.keySeparator",
            "value": ":"
          },
          {
            "name": "fromJobConfig.valueStoreType",
            "value": "STRING"
          },
          {
            "name": "fromJobConfig.valueSeparator",
            "value": ","
          },
          {
            "name": "fromJobConfig.columnList",
            "value": "1&2&3&4&5&6&7&8&9&10&11&12"
          }
        ],
        "name": "fromJobConfig"
      }
    ]
  }
```

Parameter Description

- Redis job parameter description

Parameter	Mandatory	Type	Description
fromJobConfig.isBatchMigration	No	Boolean	Whether to migrate all data in the database
fromJobConfig.keyPrefix	Yes	String	Key prefix, which is the name of the corresponding association table. Mapping between Redis and the association table: *Name of the association table + delimiter* is a Redis key, and a row of data in the association table is a Redis value.
fromJobConfig.keySeparator	Yes	String	Key delimiter, which separates the association table and primary key
fromJobConfig.valueStoreType	Yes	String	Storage mode of rows of data in the association table on Redis. The options are **string** and **hash**. • **STRING**: indicates that a row of data is stored as a character string using delimiters to separate columns. This mode reduces storage space occupation. • **HASH**: indicates that a row of data is stored in *column name:column value* format in the hash table.
fromJobConfig.valueSeparator	No	String	Value delimiter. The default value is **\tab**. This parameter is valid when **valueStoreType** is set to **STRING**.
fromJobConfig.columnList	No	String	List of fields to be extracted. Use **&** to separate field names. For example, **id&gid&name**.

Parameter	Mandatory	Type	Description
fromJobConfig.formats	No	Data structure	Time format. For details, see **Description of the fromJobConfig.formats parameter**.

- Description of the **fromJobConfig.formats** parameter

Parameter	Mandatory	Type	Description
name	Yes	String	Column number. For example, **1**.
value	Yes	String	Time format. For example, *yyyy-MM-dd*.

6.2.10 From DIS

Sample JSON File

```
"from-config-values": {
        "configs": [
          {
            "inputs": [
              {
                "name": "fromJobConfig.streamName",
                "value": "cdm"
              },
              {
                "name": "fromJobConfig.disConsumerStrategy",
                "value": "FROM_LAST_STOP"
              },
              {
                "name": "fromJobConfig.isPermanency",
                "value": "true"
              },
              {
                "name": "fromJobConfig.maxPollRecords",
                "value": "100"
              },
              {
                "name": "fromJobConfig.shardId",
                "value": "0"
              },
              {
                "name": "fromJobConfig.dataFormat",
                "value": "BINARY"
              },
              {
                "name": "fromJobConfig.separator",
                "value": ","
              }
            ],
            "name": "fromJobConfig"
          }
        ]
    }
```


Parameter Description

Parameter	Mandatory	Type	Description
fromJobConfig.streamName	Yes	String	DIS stream name
fromJobConfig.disConsumerStrategy	Yes	Enumeration	Used to set the initial offset when data is pulled from DIS. The options are as follows: • **LATEST**: maximum offset, that is, the latest data • **FROM_LAST_STOP**: data after the last stop • **EARLIEST**: minimum offset, that is, the earliest data
fromJobConfig.isPermanency	Yes	Boolean	Whether to run permanently
fromJobConfig.maxPollRecords	No	String	Maximum number of requests that can be sent to DIS each time
fromJobConfig.shardId	Yes	String	ID of the DIS partition. You can enter multiple partition IDs, which are separated by commas (,).
fromJobConfig.dataFormat	Yes	Enumeration	Format used for parsing data. The options are as follows: • **BINARY**: Data is transferred without being parsed, which is applicable to file migration. • **CSV**: Source data will be migrated after being parsed in CSV format.
fromJobConfig.separator	No	String	Field delimiter
fromJobConfig.appName	No	String	Unique identifier of the consumer application

6.2.11 From Kafka

Sample JSON File

```
"from-config-values": {
        "configs": [
          {
```



```
            "inputs": [
               {
               "name": "fromJobConfig.topicsList",
               "value": "est1,est2"
               },
               {
               "name": "fromJobConfig.kafkaConsumerStrategy",
               "value": "EARLIEST"
               },
               {
               "name": "fromJobConfig.isPermanency",
               "value": "true"
               }
            ],
            "name": "fromJobConfig"
         }
      ]
   }
```

Parameter Description

Parameter	Mandatory	Type	Description
fromJobConfig.topicsList	Yes	String	List of Kafka topics. Separate multiple topics by commas (,).
fromJobConfig.kafkaConsumerStrategy	Yes	Enumeration	Used to set the initial offset when data is pulled from Kafka. The options are as follows: • **LATEST**: maximum offset, that is, the latest data • **EARLIEST**: minimum offset, that is, the earliest data
fromJobConfig.isPermanency	Yes	Boolean	Whether to run permanently
fromJobConfig.groupId	No	String	Consumer group ID If you export data from DMS Kafka, enter any value for Kafka Platinum but a valid consumer group ID for Kafka Basic.
fromJobConfig.dataFormat	Yes	Enumeration	Format used for parsing data. The options are as follows: • **BINARY**: Data is transferred without being parsed, which is applicable to file migration. • **CSV**: Source data will be migrated after being parsed in CSV format.

Parameter	Mandatory	Type	Description
fromJobConfig.maxPollRecords	No	String	Maximum number of requests that can be sent to Kafka each time
fromJobConfig.maxPollInterval	No	String	Maximum interval between each poll
fromJobConfig.separator	No	String	Field delimiter

6.2.12 From Elasticsearch/Cloud Search Service

Sample JSON File

```
"from-config-values": {
            "configs": [
               {
                  "inputs": [
                     {
                        "name": "fromJobConfig.index",
                        "value": "cdm"
                     },
                     {
                        "name": "fromJobConfig.type",
                        "value": "es"
                     },
                     {
                        "name": "fromJobConfig.columnList",
                        "value": "a1:numeric&s1:string"
                     },
                     {
                        "name": "fromJobConfig.splitNestedField",
                        "value": "true"
                     },
                     {
                        "name": "fromJobConfig.queryString",
                        "value": "last_name:Smith"
                     }
                  ],
                  "name": "fromJobConfig"
               }
            ]
         }
```

Parameter Description

Parameter	Mandatory	Type	Description
fromJobConfig.index	Yes	String	Index of the extracted data, which is similar to the database name in the relational database

Parameter	Mandatory	Type	Description
fromJobConfig.type	Yes	String	Type of the extracted data, which is similar to the table name in the relational database
fromJobConfig.columnList	No	String	List of fields to be extracted. Use **&** to separate field names. For example, **id&gid&name**.
fromJobConfig.splitNestedField	No	Boolean	Whether to split the JSON content of the **nested** field. For example, **a:{ b:{ c:1, d:{ e:2, f:3 } } }** can be split into **a.b.c**, **a.b.d.e**, and **a.b.d.f**.
fromJobConfig.queryString	No	String	Whether to use the Elasticsearch query string to filter the source data. CDM migrates only the data that meets the filter criteria.

6.3 Destination Job Parameters

6.3.1 To a Relational Database

Sample JSON File

```
"to-config-values": {
      "configs": [
       {
         "inputs": [
           {
             "name": "toJobConfig.schemaName",
             "value": "cdm"
           },
           {
             "name": "toJobConfig.tablePreparation",
             "value": "DROP_AND_CREATE"
           },
           {
             "name": "toJobConfig.tableName",
             "value": "rf_to"
           },
           {
             "name": "toJobConfig.columnList",
             "value": "id&gid&name"
           },
           {
             "name": "toJobConfig.isCompress",
             "value": "false"
           },
           {
             "name": "toJobConfig.orientation",
             "value": "ROW"
           },
```



```
        {
          "name": "toJobConfig.useStageTable",
          "value": "false"
        },
        {
          "name": "toJobConfig.shouldClearTable",
          "value": "false"
        },
        {
          "name": "toJobConfig.extendCharLength",
          "value": "false"
        }
      ],
      "name": "toJobConfig"
    }
  ]
}
```

Parameter Description

Parameter	Mandatory	Type	Description
toJobConfig.schemaName	Yes	String	Database mode or tablespace
toJobConfig.tablePreparation	Yes	Enumeration	This parameter is available only when both the source and destination databases are relational databases. The options for data write to tables are as follows: • **DO_NOTHING**: Do not create the table automatically. • **CREATE_WHEN_NOT_EXIST**: If the destination database does not contain the table specified by **tableName**, CDM automatically creates the table. • **DROP_AND_CREATE**: Delete the table specified by **tableName**, and then create the table again.
toJobConfig.tableName	Yes	String	Name of the table to which data is written
toJobConfig.columnList	No	String	List of fields to be loaded. Use **&** to separate field names. For example, **id&gid&name**.

Parameter	Mandatory	Type	Description
toJobConfig.beforeImportType	No	Enumeration	Whether to clear the data in the target table before data import. The options are as follows: • **none**: Do not clear the data in the target table before data import but append data to the table. • **shouldClearTable**: Clear the data in the target table before data import. • **whereClause**: To clear data in the target table based on the WHERE clause, set the **toJobConfig.whereClause** parameter. CDM deletes the data from the target table as specified.
toJobConfig.whereClause	No	String	WHERE clause used to delete data from the target table before data import
toJobConfig.orientation	No	Enumeration	Storage mode. This parameter is enabled only for the DWS database. When the DWS database table needs to be automatically created, the optional data storage modes of the table are as follows: • **ROW**: Data in the table is stored in rows. • **COLUMN**: Data in the table is stored in columns.
toJobConfig.isCompress	No	Boolean	Whether to perform compression. This parameter is enabled only for the DWS database. When the DWS database table needs to be automatically created, you can specify whether to store the data in the table after compression.

Parameter	Mandatory	Type	Description
toJobConfig.useStageTable	No	Boolean	Whether to import data to the phase table first. If this parameter is set to **true**, the data is imported to the phase table before it is imported to the destination table. After the data is successfully imported to the phase table, it is then imported from the phase table to the destination table. In this way, the data that is successfully imported to the destination table remains in case the data import fails.
toJobConfig.extendCharLength	No	Boolean	Whether to extend the length of the character string field. If this parameter is set to **true**, the length of the character string field in the destination table is three times the length of the corresponding field in the source table when the destination table needs to be automatically created.
toJobConfig.useNullable	No	Boolean	If you choose to create a target table automatically and specify the NOT NULL constraint, keep the NOT NULL constraints of the source and target tables consistent.

6.3.2 To OBS

Sample JSON File

```
"to-config-values": {
      "configs": [
        {
          "inputs": [
            {
              "name": "toJobConfig.bucketName",
              "value": "cdm"
            },
            {
              "name": "toJobConfig.outputDirectory",
              "value": "/obsfrom/advance/"
            },
            {
              "name": "toJobConfig.outputFormat",
              "value": "CSV_FILE"
            },
            {
```



```
          "name": "toJobConfig.fieldSeparator",
          "value": ","
        },
        {
          "name": "toJobConfig.writeToTempFile",
          "value": "false"
        },
        {
          "name": "toJobConfig.validateMD5",
          "value": "false"
        },
        {
          "name": "toJobConfig.recordMD5Result",
          "value": "false"
        },
        {
          "name": "toJobConfig.encodeType",
          "value": "UTF-8"
        },
        {
          "name": "toJobConfig.markerFile",
          "value": "finish.txt"
        },
        {
          "name": "toJobConfig.duplicateFileOpType",
          "value": "REPLACE"
        },
        {
          "name": "toJobConfig.columnList",
          "value": "1&2"
        },
        {
          "name": "toJobConfig.quoteChar",
          "value": "false"
        },
        {
          "name": "toJobConfig.encryption",
          "value": "NONE"
        },
        {
          "name": "toJobConfig.copyContentType",
          "value": "false"
        }
      ],
      "name": "toJobConfig"
    }
  ]
}
```

Parameter Description

Parameter	Mandatory	Type	Description
toJobConfig.bucketName	Yes	String	OBS bucket name. For example, **cdm**.
toJobConfig.outputDirectory	Yes	String	Path to which data is written. For example, **data_dir**.

Parameter	Mandatory	Type	Description
toJobConfig.outputFormat	Yes	Enumeration	File format required for data writes (except the binary format). Currently, the following file formats are supported: • **CSV_FILE**: Write data in CSV format. • **BINARY_FILE**: Files are directly transferred without resolving the content. CDM writes the file without changing the file format. If you select **BINARY_FILE**, the migration source must also be a file system.
toJobConfig.fieldSeparator	No	String	Column delimiter. This parameter is valid only when **toJobConfig.outputFormat** is **CSV_FILE**. The default value is **,**.
toJobConfig.lineSeparator	No	String	Line feed character. This parameter is valid only when **toJobConfig.outputFormat** is **CSV_FILE**. The default value is **\r\n**.
toJobConfig.writeFileSize	No	String	Whether to fragment multiple files by size so that the files are exported in proper size. The unit is MB. This parameter is valid when the migration source is a database.

Parameter	Mandatory	Type	Description
toJobConfig.duplicateFileOpType	No	Enumeration	Method for processing duplicate files. If the name and size of a file are the same as those of another file, the file is regarded as a duplicate file. Duplicate files can be processed in the following ways: • **REPLACE**: Replace duplicate files. • **SKIP**: Skip duplicate files. • **ABANDON**: Stop the job when any duplicate file is found.
toJobConfig.columnList	No	String	List of fields to be extracted. Use **&** to separate field names. For example, **id&gid&name**.
toJobConfig.encryption	No	Enumeration	Whether to encrypt the uploaded data and the encryption method. The options are as follows: • **NONE**: Directly write data without encryption. • **KMS**: Use KMS in Data Encryption Workshop (DEW) for encryption. If KMS encryption is enabled, MD5 verification for data cannot be performed.
toJobConfig.kmsID	No	String	Key used for encryption during data upload. You must create a key in KMS before data upload.
toJobConfig.projectID	No	String	ID of the project to which the KMS key belongs.
toJobConfig.writeToTempFile	No	Boolean	The binary file is written to a **.tmp** file first. After the migration is successful, run the **rename** or **move** command at the migration destination to restore the file.

Parameter	Mandatory	Type	Description
toJobConfig.validateMD5	No	Boolean	Whether to verify the MD5 value. MD5 verification cannot be used together with KMS encryption. MD5 values can be verified only when files are transferred in binary format. Calculate the MD5 value of the source file and verify it with the MD5 value returned by OBS. If an MD5 file exists on the source end, directly read the MD5 file and verify the MD5 file with the MD5 value returned by OBS.
toJobConfig.recordMD5Result	No	Boolean	Whether to record the verification result when the MD5 value is verified
toJobConfig.recordMD5Link	No	String	OBS link where the bucket to which the MD5 verification result is written resides
toJobConfig.recordMD5Bucket	No	String	OBS bucket to which the MD5 verification result is written
toJobConfig.recordMD5Directory	No	String	Directory to which the MD5 verification result is written
toJobConfig.encodeType	No	String	Encoding type. For example, **UTF_8** or **GBK**.
toJobConfig.markerFile	No	String	Whether to generate a marker file with a custom name in the destination directory after a job is executed successfully. If you do not specify a file name, this function is disabled by default.

Parameter	Mandatory	Type	Description
toJobConfig.copyContentType	No	Boolean	This parameter is displayed only when **toJobConfig.outputFormat** is **BINARY_FILE** and both the migration source and destination are object storage. If you set this parameter to **Yes**, the Content-Type attribute of the source file is copied during object file migration. This function is mainly used for static website migration The Content-Type attribute cannot be written to Archive buckets. Therefore, if you set this parameter to **Yes**, the migration destination must be a non-Archive bucket.
toJobConfig.quoteChar	No	Boolean	This parameter is available only when **toJobConfig.outputFormat** is **CSV_FILE**. It is used when database tables are migrated to file systems. If you set this parameter to **Yes** and a field in the source data table contains a field delimiter or line separator, CDM uses double quotation marks (") as the quote character to quote the field content as a whole to prevent a field delimiter from dividing a field into two fields, or a line separator from dividing a field into different lines. For example, if the **hello,world** field in the database is quoted, it will be exported to the CSV file as a whole.

Parameter	Mandatory	Type	Description
toJobConfig.firstRowAsHeader	No	Boolean	This parameter is available only when **toJobConfig.outputFormat** is **CSV_FILE**. When a table is migrated to a CSV file, CDM does not migrate the heading line of the table by default. If this parameter is set to **Yes**, CDM writes the heading line of the table to the file.
toJobConfig.filePrefix	No	String	Custom file name prefix. It can be any custom name or a combination of the table name macro, time macro, and version macro. Example: **${tableName}_${dateformat(yyyy-MM-dd HH:mm:ss, -1, DAY)}_${version}**. The file name must comply with the OBS file path naming rules.

6.3.3 To HDFS

Sample JSON File

```
"to-config-values": {
       "configs": [
        {
          "inputs": [
            {
              "name": "toJobConfig.outputDirectory",
              "value": "/hdfsto"
            },
            {
              "name": "toJobConfig.outputFormat",
              "value": "BINARY_FILE"
            },
            {
              "name": "toJobConfig.writeToTempFile",
              "value": "false"
            },
            {
              "name": "toJobConfig.duplicateFileOpType",
              "value": "REPLACE"
            },
            {
              "name": "toJobConfig.compression",
              "value": "NONE"
            },
            {
              "name": "toJobConfig.appendMode",
```



```
            "value": "true"
          }
        ],
        "name": "toJobConfig"
      }
    ]
  }
```

Parameter Description

Parameter	Mandatory	Type	Description
toJobConfig.outputDirectory	Yes	String	Path to which data is written. For example, **/data_dir**.
toJobConfig.outputFormat	Yes	Enumeration	File format required for data writes (except the binary format). Currently, the following file formats are supported: • **CSV_FILE**: Write data in CSV format. • **BINARY_FILE**: Files are directly transferred without resolving the content. CDM writes the file without changing the file format. If you select **BINARY_FILE**, the migration source must also be a file system.
toJobConfig.lineSeparator	No	String	Line feed character. This parameter is valid only when **toJobConfig.outputFormat** is **CSV_FILE**. The default value is **\r\n**.
toJobConfig.fieldSeparator	No	String	Column delimiter. This parameter is valid only when **toJobConfig.outputFormat** is **CSV_FILE**. The default value is **,**.
toJobConfig.writeToTempFile	No	Boolean	The binary file is written to a **.tmp** file first. After the migration is successful, run the **rename** or **move** command at the migration destination to restore the file.

Parameter	Mandatory	Type	Description
toJobConfig.duplicateFileOpType	No	Enumeration	Method for processing duplicate files. If the name and size of a file are the same as those of another file, the file is regarded as a duplicate file. Duplicate files can be processed in the following ways: • **REPLACE**: Replace duplicate files. • **SKIP**: Skip duplicate files. • **ABANDON**: Stop the job when any duplicate file is found.
toJobConfig.compression	No	Enumeration	After the file is written, select the compression format of the file. The following compression formats are supported: • **NONE**: Do not compress the file. • **DEFLATE**: Compress the file in DEFLATE format. • **GZIP**: Compress the file in gzip format. • **BZIP2**: Compress the file in bzip2 format. • **LZ4**: Compress the file in LZ4 format. • **SNAPPY**: Compress the file in Snappy format.
toJobConfig.appendMode	Yes	Boolean	Whether to write data when one or more files exist in the loading path. The default value is **false**.

Parameter	Mandatory	Type	Description
toJobConfig.encryption	No	Enumeration	This parameter is available only when **toJobConfig.outputFormat** is set to **BINARY_FILE**. It specifies whether to encrypt the uploaded data, and the encryption method. The options are as follows: • **NONE**: Directly write data without encryption. • **AES-256-GCM**: Use the AES 256-bit encryption algorithm to encrypt data. Currently, only the AES-256-GCM (NoPadding) encryption algorithm is supported.
toJobConfig.dek	No	String	Data encryption key. This parameter is available when **toJobConfig.encryption** is set to **AES-256-GCM**. The key is a string of 64-bit hexadecimal numbers. Remember the key configured here because the decryption key must be the same as that configured here. If the encryption and decryption keys are inconsistent, the system does not report an exception, but the decrypted data is incorrect.
toJobConfig.iv	No	String	Initialization vector. This parameter is available when **toJobConfig.encryption** is set to **AES-256-GCM**. The initialization vector is a string of 32-bit hexadecimal numbers. Remember the initialization vector configured here because the initialization vector used for decryption must be the same as that configured here. If the encryption and decryption keys are inconsistent, the system does not report an exception, but the decrypted data is incorrect.

Parameter	Mandatory	Type	Description
toJobConfig.filePrefix	No	String	Custom file name prefix, which can be a time macro, for example, **test_${dateformat(yyyyMMdd, -1, DAY)}**. The file name must comply with the HDFS file path naming rules.

6.3.4 To Hive

Sample JSON File

```
"to-config-values": {
      "configs": [
       {
        "inputs": [
         {
          "name": "toJobConfig.hive",
          "value": "hive"
         },
         {
          "name": "toJobConfig.database",
          "value": "rf_database"
         },
         {
          "name": "toJobConfig.table",
          "value": "rf_to"
         },
         {
          "name": "toJobConfig.tablePreparation",
          "value": "DO_NOTHING"
         },
         {
          "name": "toJobConfig.columnList",
          "value": "aa&bb&cc&dd"
         },
         {
          "name": "toJobConfig.shouldClearTable",
          "value": "true"
         }
        ],
        "name": "toJobConfig"
       }
      ]
    }
```

Parameter Description

Parameter	Mandatory	Type	Description
toJobConfig.hive	No	String	Data source to which data is written

Parameter	Mandatory	Type	Description
toJobConfig.database	No	String	Name of the database to which data is written. For example, **default**.
toJobConfig.table	Yes	String	Name of the table to which data is written
toJobConfig.tablePreparation	Yes	Enumeration	The options for data write to tables are as follows: • **DO_NOTHING**: Do not create the table automatically. • **CREATE_WHEN_NOT_EXIST**: If the destination database does not contain the table specified by **tableName**, CDM automatically creates the table. • **DROP_AND_CREATE**: Delete the table specified by **tableName**, and then create the table again.
toJobConfig.columnList	No	String	List of fields to be loaded. Use **&** to separate field names. For example, **id&gid&name**.
toJobConfig.shouldClearTable	No	Boolean	Whether to clear the data in the target table before data import. If this parameter is set to **true**, the data in the target table is cleared before the job is started.

6.3.5 To HBase/CloudTable

Sample JSON File

```
"to-config-values": {
      "configs": [
       {
         "inputs": [
          {
            "name": "toJobConfig.table",
            "value": "rf_to"
          },
          {
            "name": "toJobConfig.storageType",
            "value": "PUTLIST"
          },
```

```
        {
          "name": "toJobConfig.columns",
          "value": "AA:AA&BB:BB&CC:CC&DD:DD"
        },
        {
          "name": "toJobConfig.rowKeyColumn",
          "value": "AA:AA"
        },
        {
          "name": "toJobConfig.isOverride",
          "value": "false"
        },
        {
          "name": "toJobConfig.isRowkeyRedundancy",
          "value": "false"
        },
        {
          "name": "toJobConfig.algorithm",
          "value": "NONE"
        },
        {
          "name": "toJobConfig.writeToWAL",
          "value": "true"
        },
        {
          "name": "toJobConfig.transType",
          "value": "false"
        }
      ],
      "name": "toJobConfig"
    }
  ]
}
```

Parameter Description

Parameter	Mandatory	Type	Description
toJobConfig.table	Yes	String	Name of the table to which data is written. For example, **TBL_EXAMPLE**.
toJobConfig.storageType	Yes	Enumeration	Mode for writing data to an HBase table. The options are as follows: • **PUTLIST**: Data is written using the putList() function.
toJobConfig.columns	No	String	Columns to be extracted. Use **&** to separate column numbers and **:** to separate column families and columns. For example, **cf1:c1&cf2:c2**.
toJobConfig.rowKeyColumn	Yes	String	Columns serve as rowkeys. Use **&** to separate column numbers and **:** to separate column families and columns. For example, **cf1:c1&cf2:c2**.

Parameter	Mandatory	Type	Description
toJobConfig.isOverride	No	Boolean	Whether to clear data when data is imported in BULKLOAD mode. For example, **true**.
toJobConfig.delimiter	No	String	Delimiter used for separating columns when multiple columns are used as rowkeys. For example, vertical bars (|).
toJobConfig.isRowkeyRedundancy	No	Boolean	Whether to write rowkey data to the HBase column at the same time
toJobConfig.algorithm	No	Enumeration	Compression algorithm used when a new HBase table is created. The Snappy and GZ algorithms are supported. The default value is **None**.
toJobConfig.writeToWAL	No	Boolean	Whether to enable Write Ahead Log (WAL) of HBase. The options are as follows: ● **Yes**: If the HBase server breaks down after the function is enabled, you can replay the operations that have not been performed in WAL. ● **No**: If you set this parameter to **No**, the write performance is improved. However, if the HBase server breaks down, data may be lost.

Parameter	Mandatory	Type	Description
toJobConfig.transType	No	Boolean	• **true**: Data of the Short, Int, Long, Float, Double, and Decimal columns in the source database is converted into Byte[] arrays (binary) and written into HBase. Other types of data are written as character strings. If several types of data mentioned above are combined as rowkeys, they will be written as character strings. This function saves storage space. In specific scenarios, the rowkey distribution is evener. • **false**: All types of data in the source database are written into HBase as character strings.

6.3.6 To DDS

Sample JSON File

```
"to-config-values": {
      "configs": [
        {
          "inputs": [
            {
              "name": "toJobConfig.database",
              "value": "demo"
            },
            {
              "name": "toJobConfig.collectionName",
              "value": "cdmbase"
            },
            {
              "name": "toJobConfig.columnList",
              "value": "_char&_varchar"
            },
            {
              "name": "toJobConfig.isBatchMigration",
              "value": "false"
            }
          ],
          "name": "toJobConfig"
        }
      ]
    }
```


Parameter Description

Parameter	Mandatory	Type	Description
toJobConfig.database	Yes	String	Name of the MongoDB/DDS database
toJobConfig.collectionName	Yes	String	Name of the MongoDB/DDS collection
toJobConfig.columnList	No	String	List of fields to be extracted. Use **&** to separate field names. For example, **id&gid&name**.
toJobConfig.isBatchMigration	No	Boolean	Whether to migrate all data in the database

6.3.7 To Elasticsearch/Cloud Search Service

Sample JSON File

```
"to-config-values": {
            "configs": [
               {
                  "inputs": [
                     {
                       "name": "toJobConfig.index",
                       "value": "cdm"
                     },
                     {
                       "name": "toJobConfig.type",
                       "value": "type1"
                     },
                     {
                       "name": "toJobConfig.shouldClearType",
                       "value": "false"
                     },
                    {
                       "name": "toJobConfig.pipeLine",
                       "value": "es_03"
                    }
                  ],
                  "name": "toJobConfig"
               }
            ]
         }
```

Parameter Description

Parameter	Mandatory	Type	Description
toJobConfig.index	Yes	String	Index of the written data, which is similar to the database name in the relational database

Parameter	Mandatory	Type	Description
toJobConfig.type	Yes	String	Type of the written data, which is similar to the table name in the relational database
toJobConfig.shouldClearType	No	Boolean	Whether to clear data before import
toJobConfig.primaryKey	No	String	Primary key or unique index
toJobConfig.columnList	No	String	List of fields to be written. Use **&** to separate field names. For example, **id&gid&name**.
toJobConfig.pipeLine	No	String	This parameter is available only after a pipeline ID is created in Kibana. It is used to convert the data format using the data transformation pipeline of Cloud Search Service or Elasticsearch after data is transferred to Cloud Search Service or Elasticsearch.

Parameter	Mandatory	Type	Description
toJobConfig.createIndexStrategy	No	Enumeration	For streaming jobs that continuously write data to Elasticsearch, CDM periodically creates indexes and writes data to the indexes, which helps you delete expired data. The indexes can be created based on the following periods: • **EveryHour**: CDM creates indexes on the hour. The new indexes are named in the format of *Index name+Year+Month+Day+Hour*, for example, **index2018121709**. • **EveryDay**: CDM creates indexes at 00:00 every day. The new indexes are named in the format of *Index name+Year+Month+Day*, for example, **index20181217**. • **EveryWeek**: CDM creates indexes at 00:00 every Monday. The new indexes are named in the format of *Index name+Year+Week*, for example, **index201842**. • **EveryMonth**: CDM creates indexes at 00:00 on the first day of each month. The new indexes are named in the format of *Index name+Year+Month*, for example, **index201812**. When extracting data from a file, you must configure a single extractor, which means setting **Concurrent Extractors** to **1**. Otherwise, this parameter is invalid.

6.3.8 To DLI

Sample JSON File

```
"to-config-values": {
            "configs": [
              {
```



```
                "inputs": [
                  {
                    "name": "toJobConfig.queue",
                    "value": "cdm"
                  },
                  {
                    "name": "toJobConfig.database",
                    "value": "sqoop"
                  },
                  {
                    "name": "toJobConfig.table",
                    "value": "est1"
                  },
                  {
                    "name": "toJobConfig.columnList",
                    "value":
"string_&int_&date_&double_&boolean_&short_&timestamp_&long_&smallint_&bigint_"
                  },
                  {
                    "name": "toJobConfig.shouldClearTable",
                    "value": "false"
                  }
                ],
                "name": "toJobConfig"
              }
            ]
          }
```

Parameter Description

Parameter	Mandatory	Type	Description
toJobConfig.queue	Yes	String	Resource queue to which data is written
toJobConfig.database	Yes	String	DLI database to which data is written
toJobConfig.table	Yes	String	Name of the table to which data is written
toJobConfig.columnList	No	String	List of fields to be loaded. Use **&** to separate field names. For example, **id&gid&name**.
toJobConfig.shouldClearTable	No	Boolean	Whether to clear data in the resource queue before data import

6.3.9 To DIS

Sample JSON File

```
"to-config-values": {
      "configs": [
        {
          "inputs": [
            {
              "name": "toJobConfig.streamName",
              "value": "cdm"
            },
```

```
        {
          "name": "toJobConfig.separator",
          "value": ","
        },
        {
          "name": "toJobConfig.identifierEnclose",
          "value": "'"
        }
      ],
      "name": "toJobConfig"
    }
  ]
}
```

Parameter description

Parameter	Mandatory	Type	Description
toJobConfig.streamName	Yes	String	DIS stream name
toJobConfig.separator	No	String	Field separator. The default value is a space.
toJobConfig.identifierEnclose	No	String	Delimiter used to separate referenced table names or column names. By default, this parameter is left blank.

6.4 Job Parameter Description

When you perform the operations in **Creating a Job in a Specified Cluster** or **Creating and Executing a Job in a Random Cluster**, the **driver-config-values** parameter specifies the job configuration, which includes the following functions:

- **Retry upon Failure**: If a job fails to be executed, you can choose whether to automatically restart the job.
- **Job Group**: CDM allows you to group jobs. You can filter, delete, start, or export jobs by group.
- **Schedule Execution**: Specify whether to execute scheduled jobs.
- **Concurrent Extractors**: Enter the number of concurrent extractors.
- **Write Dirty Data**: Specify this parameter if data that fails to be processed or filtered out during job execution needs to be written to OBS for future viewing. Before writing dirty data, create an OBS link.
- **Delete Job After Completion**: Specify whether to delete a job after the job is executed.

Sample JSON File

```
"driver-config-values": {
    "configs": [
      {
        "inputs": [
          {
            "name": "throttlingConfig.numExtractors",
```



```
          "value": "1"
        },
        {
          "name": "throttlingConfig.numLoaders",
          "value": "1"
        },
        {
          "name": "throttlingConfig.recordDirtyData",
          "value": "false"
        }
      ],
      "name": "throttlingConfig"
    },
    {
      "inputs": [],
      "name": "jarConfig"
    },
    {
      "inputs": [
        {
          "name": "schedulerConfig.isSchedulerJob",
          "value": "false"
        },
        {
          "name": "schedulerConfig.disposableType",
          "value": "NONE"
        }
      ],
      "name": "schedulerConfig"
    },
    {
      "inputs": [],
      "name": "transformConfig"
    },
    {
      "inputs": [
        {
          "name": "retryJobConfig.retryJobType",
          "value": "NONE"
        }
      ],
      "name": "retryJobConfig"
    }
  ]
}
```

Parameter Description

Parameter	Mandatory	Type	Description
throttlingConfig.numExtractors	No	Integer	Maximum number of concurrent extraction jobs. For example, **20**.
groupJobConfig.groupName	No	Enumeration	Group to which a job belongs. The default group is **DEFAULT**.
throttlingConfig.numLoaders	No	Integer	This parameter is available only when HBase or Hive serves as the destination data source. Maximum number of loading jobs. For example, **5**.

Parameter	Mandatory	Type	Description
throttlingConfig.recordDirtyData	No	Boolean	Whether to write dirty data. For example, **true**.
throttlingConfig.writeToLink	No	String	Link to which dirty data is written. Currently, dirty data can be written only to OBS or HDFS. For example, **obslink**.
throttlingConfig.obsBucket	No	String	Name of the OBS bucket to which dirty data is written. This parameter is valid only when dirty data is written to OBS. For example, **dirtyData**.
throttlingConfig.dirtyDataDirectory	No	String	Directory to which dirty data is written • To write dirty data to HDFS, set this parameter to the specified HDFS directory. • To write dirty data to OBS, set this parameter to the directory in the OBS bucket. For example, **/data/dirtydata/**.
throttlingConfig.maxErrorRecords	No	String	Maximum number of error records in a single shard. When the number of error records of a map exceeds the upper limit, the task automatically ends. The imported data will not be rolled back.
schedulerConfig.isSchedulerJob	No	Boolean	Whether to enable a scheduled task. For example, **true**.
schedulerConfig.cycleType	No	String	Cycle type of a scheduled task. The options are as follows: • **minute**: minute • **hour**: hour • **day**: day • **week**: week • **month**: month
schedulerConfig.cycle	No	Integer	Cycle of a scheduled task. If **cycleType** is set to **minute** and **cycle** is set to **10**, the scheduled task is executed every 10 minutes.

Parameter	Mandatory	Type	Description
schedulerConfig.runAt	No	String	Time when a scheduled task is triggered in a cycle. This parameter is valid only when **cycleType** is set to **hour**, **week**, or **month**. ● If **cycleType** is set to **month**, **cycle** is set to **1**, and **runAt** is set to **15**, the scheduled task is executed on the 15th day of each month. You can set **runAt** to multiple values and separate the values with commas (,). For example, if **runAt** is set to **1,2,3,4,5**, the scheduled task is executed on the first day, second day, third day, fourth day, and fifth day of each month. ● If **cycleType** is set to **week** and **runAt** is set to **mon,tue,wed,thu,fri**, the scheduled task is executed on Monday to Friday. ● If **cycleType** is set to **hour** and **runAt** is set to **27,57**, the scheduled task is executed at the 27th and 57th minute in the cycle.
schedulerConfig.startDate	No	String	Start time of a scheduled task. For example, **2018-01-24 19:56:19**.
schedulerConfig.stopDate	No	String	End time of a scheduled task. For example, **2018-01-27 23:59:00**. If you do not set the end time, the scheduled task is always executed and will never stop.

Parameter	Mandatory	Type	Description
schedulerConfig.disposableType	No	Enumeration	Whether to delete a job after the job is executed. The options are as follows: • **NONE**: A job will not be deleted after it is executed. • **DELETE_AFTER_SUCCEED**: A job will be deleted only after it is successfully executed. It is applicable to massive one-time jobs. • **DELETE**: A job will be deleted after it is executed, regardless of the execution result.
retryJobConfig.retryJobType	No	Enumeration	Whether to automatically retry if a job fails to be executed. The options are as follows: • **NONE**: Do not retry. • **RETRY_TRIPLE**: Retry three times.

7 Permissions Policies and Supported Actions

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

By default, new IAM users do not have any permissions assigned. You need to add a user to one or more groups, and assign permissions policies or roles to these groups. The user then inherits permissions from the groups it is a member of. This process is called authorization. After authorization, the user can perform specified operations on 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 the cluster list using an API, the user must have been granted permissions that allow the **cdm:cluster:list** action.

Supported Actions

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

- Permission: A statement in a policy that allows or denies certain operations.
- APIs: REST APIs that can be called in a custom policy.

- Actions: added to a custom policy to control permissions for specific operations.
- IAM or enterprise projects: Type of projects for which an action will take effect. Policies that contain actions supporting both IAM and enterprise projects can be assigned to user groups and take effect in both IAM and Enterprise Management. Policies that only contain actions supporting IAM projects can be assigned to user groups 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.

CDM supports the actions listed in **Table 7-1** that can be defined in custom policies. All actions listed in the following table support both Project and Enterprise Project.

Table 7-1 Actions

Permission	API	Action	IAM Project	Enterprise Project
Creating a cluster	POST /v1.1/{project_id}/clusters	cdm:cluster:create	√	×
Querying the cluster list	GET /v1.1/{project_id}/clusters	cdm:cluster:list	√	×
Querying cluster details	GET /v1.1/{project_id}/clusters/{cluster_id}	cdm:cluster:get	√	×
Restarting a cluster	POST /v1.1/{project_id}/clusters/{cluster_id}/action	cdm:cluster:operate	√	×
Modifying cluster configuration	POST /v1.1/{project_id}/cluster/modify/{cluster_id}	cdm:cluster:modify	√	×
Deleting a cluster	DELETE /v1.1/{project_id}/clusters/{cluster_id}	cdm:cluster:delete	√	×
Creating a link	POST /v1.1/{project_id}/clusters/{cluster_id}/cdm/link	cdm:link:operate	√	×

Permission	API	Action	IAM Project	Enterprise Project
Querying a link	GET /v1.1/{project_id}/clusters/{cluster_id}/cdm/link/{linkName}	cdm:cluster:get	√	×
Modifying a link	PUT /v1.1/{project_id}/clusters/{cluster_id}/cdm/link/{link_name}	cdm:link:operate	√	×
Deleting a link	DELETE /v1.1/{project_id}/clusters/{cluster_id}/cdm/link/{linkName}	cdm:link:operate	√	×
Creating a job in a specified cluster	POST /v1.1/{project_id}/clusters/{cluster_id}/cdm/job	cdm:job:operate	√	×
Querying a job	GET /v1.1/{project_id}/clusters/{cluster_id}/cdm/job/{jobName}	cdm:cluster:get	√	×
Modifying a job	PUT /v1.1/{project_id}/clusters/{cluster_id}/cdm/job/{jobName}	cdm:job:operate	√	×
Starting a job	PUT /v1.1/{project_id}/clusters/{cluster_id}/cdm/job/{jobName}/start	cdm:job:operate	√	×
Stopping a job	PUT /v1.1/{project_id}/clusters/{cluster_id}/cdm/job/{jobName}/stop	cdm:job:operate	√	×

Permission	API	Action	IAM Project	Enterprise Project
Querying job status	GET /v1.1/{project_id}/clusters/{cluster_id}/cdm/job/{jobName}/status	cdm:cluster:get	√	×
Querying job execution history	GET /v1.1/{project_id}/clusters/{cluster_id}/cdm/submissions	cdm:cluster:get	√	×
Deleting a job	DELETE /v1.1/{project_id}/clusters/{cluster_id}/cdm/job/{jobName}	cdm:job:operate	√	×

8 Appendix

8.1 Status Code

Table 8-1 describes the status code.

Table 8-1 Status Code

Status Code	Code	Description
100	Continue	The client continues sending the 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	Switching protocols. The target protocol must be more advanced than the source protocol. For example, the current HTTP protocol is switched to a later version.
201	Created	The request for creating a resource has been fulfilled.
202	Accepted	The request has been accepted, but the processing has not been completed.
203	Non-Authoritative Information	The server successfully processed the request, but is returning information that may be from another source.
204	NoContent	The server has successfully processed the request, but has not returned any content. The status code is returned in response to an HTTP OPTIONS request.

Status Code	Code	Description
205	Reset Content	The server has fulfilled the request, but the requester is required to reset the content.
206	Partial Content	The server has successfully processed a part of the GET request.
300	Multiple Choices	There are multiple options for the location of the requested resource. The response contains a list of resource characteristics and addresses from which the user or user agent (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	Retrieve a location. The response to the request can be found under a different URI and should be retrieved 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 must be accessed through a proxy.
306	Unused	The HTTP status code is no longer used.
400	BadRequest	Invalid request. The client should not repeat the request without modifications.
401	Unauthorized	The 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.

Status Code	Code	Description
405	MethodNotAllowed	The method specified in the request 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 request timed out. The client may repeat the request without modifications at any later time.
409	Conflict	The request could not be completed due to a conflict with the current state of the resource. This status code indicates that the resource that the client attempts to create already exits, or the request fails to be processed because of the update of the conflict request.
410	Gone	The requested resource is no longer available. The status code indicates that the requested resource has been deleted permanently.
411	Length Required	The server refuses to process the request without a defined Content-Length.
412	Precondition Failed	The server does not meet one of the preconditions that the requester puts on the request.
413	Request Entity Too Large	The request is larger than that a server is able to process. The server may close the connection to prevent the client from continuing the request. If the server cannot process the request temporarily, the response will contain a Retry-After header field.
414	Request-URI Too Large	The URI provided was 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 fails to meet the requirements of the Expect request-header field.

Status Code	Code	Description
422	UnprocessableEntity	The request is well-formed but is unable to be processed due to semantic errors.
429	TooManyRequests	The client has sent more requests than its rate limit is allowed within a given amount of time, or the server has received more requests than it is able to process within a given amount of time. In this case, it is advisable for the client to re-initiate requests after the time specified in the Retry-After header of the response expires.
500	InternalServerError	The server is able to receive the request but it could not understand the request.
501	Not Implemented	The server does not support the requested function.
502	Bad Gateway	The server is acting as a gateway or proxy and receives an invalid request from a 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 when the **Timeout** parameter is specified in the request.
505	HTTP Version not supported	The server does not support the HTTP protocol version used in the request.

8.2 Error Code

If an error occurs in API calling, no result is returned. Identify the error cause based on the error codes of each API. If an error occurs in API calling, HTTP status code 4*xx* or 5*xx* is returned. The response body contains the specific error code and information. If you are unable to identify the cause of an error, contact customer service and provide the error code so that we can help you solve the problem as soon as possible.

- Sample exception response

```
{
  "errCode": "Cdm.0100",
  "externalMessage": "Job[jdbc2hive] doesn't exist."
}
```

- Parameter description

Parameter	Mandatory	Type	Description
errCode	No	String	Error code
externalMessage	No	String	Error message

- Error code

  In the following error message, *%s* is a variable. In the actual situation, the parameter name, table name, job name, and link name are replaced with the actual values.

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

Error Code	Status Code	Error	Description	Solution
Cdm.0000	400	System error.	A system error occurs.	Contact customer service or technical support.
Cdm.0001	400	The resource does not exist or is invalid.	The requested resource does not exist or you have no permission to access the resource.	Contact customer service or technical support.
Cdm.0004	400	Invalid parameter type.	The input parameter does not match the parameter type.	Change the parameter value based on the error message and try again.
Cdm.0009	400	*%s* is not an integer or is beyond the value range [0, 2147483647].	The input parameter is not an integer or is beyond the value range.	Change the parameter value based on the error message and try again.
Cdm.0010	400	The integer must be within the range of [*%s*].	The parameter is missing or its length is **0**.	Change the parameter value based on the error message and try again.

Error Code	Status Code	Error	Description	Solution
Cdm.0011	400	The parameter value exceeds the value range.	The parameter format is incorrect or the parameter value exceeds the value range. As a result, the parsing failed.	Check whether the parameter value is valid based on the error message. If it is not, correct it and try again.
Cdm.0012	400	JDBC driver is not found.	JDBC driver is not found.	Contact customer service or technical support.
Cdm.0014	400	Invalid parameter.	The parameter is invalid.	Change the parameter value and try again.
Cdm.0015	400	Failed to parse the file.	The file fails to be parsed.	Check whether the content or format of the uploaded file is correct. If it is not, correct it and try again.
Cdm.0016	400	The file to be uploaded cannot be empty.	The uploaded file is empty.	Ensure that the file you uploaded is not empty and try again.
Cdm.0017	400	Failed to save the input value.	The input value cannot be saved.	Contact customer service or technical support.
Cdm.0018	400	The content of jobs or links is invalid.	The content of jobs or links is invalid.	Contact customer service or technical support.
Cdm.0019	400	Failed to delete the link from the database.	The link in the database fails to be deleted.	Try again later or contact customer service or technical support.
Cdm.0020	400	The string must contain the following substring: *%s*.	The verified parameter is empty or does not contain the specified substring.	Change the parameter value based on the error message and try again.

Error Code	Status Code	Error	Description	Solution
Cdm.0021	400	Failed to connect to server *%s*.	The server fails to be connected.	Contact customer service or technical support.
Cdm.0024	400	[*%s*] must be within the range of [*%s*].	The verified parameter is not in the specified value range.	Change the parameter value based on the error message and try again.
Cdm.0031	400	Failed to create data to be submitted.	Failed to create data to be submitted.	Contact customer service or technical support.
Cdm.0032	400	You do not have required permissions. Check your permissions on IAM.	Insufficient permissions	CDM console: Obtain required permissions by referring to **Permissions Management**. DataArts Studio console: Obtain required permissions by referring to **Permission Management**.
Cdm.0036	400	Internal error.	An internal error occurred. The displayed error message may vary. Contact the customer service or technical support based on the error message (for example, "Internal error: null pointer exception").	Contact customer service or technical support.
Cdm.0037	400	Failed to submit the job.	The job fails to be submitted.	Contact customer service or technical support.

Error Code	Status Code	Error	Description	Solution
Cdm.0051	400	Invalid submission engine: *%s*.	Invalid job engine name.	Specify a correct job engine and try again.
Cdm.0052	400	Job *%s* is running.	The job is running.	The operation cannot be performed because the job is running. Try again after the job completes.
Cdm.0053	400	Job *%s* is not running.	The job is not running.	Run the job and try again.
Cdm.0054	400	Job *%s* does not exist.	The job does not exist.	Check whether the job exists.
Cdm.0055	400	Unsupported job type.	Unsupported job type.	Configure a supported job type.
Cdm.0056	400	Failed to submit the job. Cause: *%s*.	The job fails to be submitted.	Locate the cause based on the error message, rectify the fault, and try again.
Cdm.0057	400	Invalid job execution engine: *%s*.	The job engine is invalid.	Specify a correct job engine and try again.
Cdm.0058	400	Invalid combination of submission and execution engines.	The combination of submission and execution engines is invalid.	Specify a correct job engine and try again.
Cdm.0059	400	Job *%s* has been disabled and cannot be submitted.	The job has been disabled and cannot be submitted.	Create a job and try again. Alternatively, contact customer service or technical support.
Cdm.0060	400	Link *%s* for this job has been disabled. The job cannot be submitted.	The link for this job has been disabled.	Change the link and submit the job again.

Error Code	Status Code	Error	Description	Solution
Cdm.0061	400	Connector *%s* does not support the specified direction. The job cannot be submitted.	The connector cannot be used as the source or destination of a job.	The connector cannot be used as the source or destination of a job. Change the connector and submit the job again.
Cdm.0062	400	The binary file is applicable only to the SFTP, FTP, HDFS, or OBS connector.	The connector is invalid.	Specify a correct connector and try again.
Cdm.0063	400	An error occurred when creating the table. Cause: *%s*.	The table fails to be created.	Locate the cause based on the error message, rectify the fault, and try again.
Cdm.0064	400	Incorrect data format.	The data format is incorrect.	Check whether the data format is correct based on the error message. If it is not, correct it and try again.
Cdm.0065	400	Failed to start the timer. Cause: *%s*.	The timer fails to be started.	Contact customer service or technical support.
Cdm.0066	400	Failed to obtain the sample value. Cause: *%s*.	The sample value fails to be obtained.	Contact customer service or technical support.
Cdm.0067	400	Failed to obtain the schema. Cause: *%s*.	The schema field fails to be obtained.	Contact customer service or technical support.

Error Code	Status Code	Error	Description	Solution
Cdm.0085	400	*%s* exceeds the maximum value *%s*.	The parameter value exceeds the maximum value.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.0089	400	The configuration item *[%s]* does not exist.	The configuration item does not exist.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.0100	400	Job [*%s*] does not exist.	The job does not exist.	Specify a correct job and try again.
Cdm.0101	400	Link [*%s*] does not exist.	The link does not exist.	Specify a correct link and try again.
Cdm.0102	400	Connector [*%s*] does not exist.	The connector does not exist.	Specify a correct connector and try again.
Cdm.0104	400	The job name already exists.	The job name already exists.	Rename the job and try again.
Cdm.0201	400	Failed to obtain the instance.	The instance fails to be obtained.	Contact customer service or technical support.
Cdm.0202	400	Unknown status.	The job status is unknown.	Try again later or contact customer service or technical support.
Cdm.0204	400	No MRS link available.	No MRS link is created.	Go to the **Links** page to create an MRS link and try again.
Cdm.0230	400	Failed to load the specified class: *%s*.	The class fails to be loaded.	Contact customer service or technical support.

Error Code	Status Code	Error	Description	Solution
Cdm.0231	400	Failed to initialize the specified class: *%s*.	The class fails to be initialized.	Contact customer service or technical support.
Cdm.0232	400	Failed to write data. Cause: *%s*.	Data fails to be written.	Contact customer service or technical support.
Cdm.0233	400	Data extraction exception. Cause: *%s*.	An exception occurs during data extraction.	Contact customer service or technical support.
Cdm.0234	400	Data loading exception. Cause: *%s*.	An exception occurs during data loading.	Contact customer service or technical support.
Cdm.0235	400	All data has been used up. Cause: *%s*.	All data has been used up.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.0236	400	Invalid partitions have been retrieved from Partitioner.	Invalid partitions have been retrieved from Partitioner.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.0238	400	*%s* cannot be left blank.	The parameter is invalid.	Change the parameter value based on the error message and try again.
Cdm.0240	400	Failed to obtain the status of file *%s*.	The file status fails to be obtained.	Contact customer service or technical support.
Cdm.0241	400	Failed to obtain the type of file *%s*.	The file type fails to be obtained.	Contact customer service or technical support.

Error Code	Status Code	Error	Description	Solution
Cdm.0242	400	File check exception: *%s*.	An exception occurs during file check.	Contact customer service or technical support.
Cdm.0243	400	Failed to rename *%s* to *%s*.	Rename failed.	Rename the job and try again.
Cdm.0244	400	Failed to create file *%s*.	The file fails to be created.	Check whether you have the permissions or try again later. If the fault persists, contact customer service or technical support.
Cdm.0245	400	Failed to delete file *%s*.	The file fails to be deleted.	Check whether you have the permissions or try again later. If the fault persists, contact customer service or technical support.
Cdm.0246	400	Failed to create directory *%s*.	The directory fails to be created.	Check whether you have the permissions or try again later. If the fault persists, contact customer service or technical support.
Cdm.0247	400	HBase operation failed. Cause: *%s*.	HBase operation failed.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.

Error Code	Status Code	Error	Description	Solution
Cdm.0248	400	Failed to clear data *%s*. Cause: *%s*.	Data fails to be cleared.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.0249	400	Invalid file name *%s*.	The file name is invalid.	Change the file name and try again.
Cdm.0250	400	Failed to perform operations on path *%s*.	Operations on path *%s* are not allowed.	Check whether you have the permissions or try again later. If the fault persists, contact customer service or technical support.
Cdm.0251	400	Failed to load data to HBase. Cause: *%s*.	Data fails to be uploaded to HBase.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.0307	400	Failed to obtain the connection lease of the requested transaction. Cause: *%s*.	The connection lease for the requested transaction fails to be obtained.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.0315	400	Link name *%s* already exists.	The link already exists.	Specify another link name and try again.
Cdm.0316	400	Failed to update the link that does not exist.	The link that does not exist cannot be updated.	Specify a correct link and try again.
Cdm.0317	400	Invalid link *%s*.	The link is invalid.	Specify a correct link and try again.

Error Code	Status Code	Error	Description	Solution
Cdm.0318	400	The job already exists and cannot be created repeatedly.	The job already exists.	Specify another job name and try again.
Cdm.0319	400	Failed to update the job that does not exist.	The job that does not exist cannot be updated.	Check whether the job to be updated exists. If it does, change the job name and try again.
Cdm.0320	400	Invalid job *%s*.	The job is invalid.	Contact customer service or technical support.
Cdm.0321	400	Link *%s* has been used.	The link has been used.	Release the link and try again.
Cdm.0322	400	Job *%s* has been used.	The job has been used.	Contact customer service or technical support.
Cdm.0323	400	The submission already exists and cannot be created repeatedly.	The submission already exists.	Try again later.
Cdm.0327	400	Invalid link or job *%s*.	Link or job *%s* is invalid.	Specify a correct link or job and try again.
Cdm.0411	400	Failed to connect to the file server.	An error occurs when connecting to the file server.	Contact customer service or technical support.
Cdm.0413	400	Failed to transfer data to the file server.	An error occurs in data transfer to the file server.	Contact customer service or technical support.
Cdm.0415	400	Failed to download files from the server.	An error occurs when downloading files from the file server.	Contact customer service or technical support.

Error Code	Status Code	Error	Description	Solution
Cdm.0416	400	Data extraction failure.	An error occurs when extracting data.	Contact customer service or technical support.
Cdm.0420	400	Source file or source directory unavailable.	The source file or source directory does not exist.	Check whether the source file or source directory exists. If it does not, specify a correct source file or directory and try again.
Cdm.0423	400	Duplicate files exist in the destination path.	Duplicate files exist in the destination path.	Delete duplicate files from the destination path and try again.
Cdm.0467	400	The connection timed out.	The connection timed out.	Check whether the IP address, host name, and port number are correct, and whether the security group and firewall are correctly configured.
Cdm.0501	400	Invalid URI [*%s*].	The URI is invalid.	Specify a correct URI and try again.
Cdm.0518	400	Failed to connect to HDFS. Cause: *%s*.	HDFS fails to be connected.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.

Error Code	Status Code	Error	Description	Solution
Cdm.0600	400	Failed to connect to the FTP server.	The FTP server fails to be connected.	It is possible that the network is disconnected, no security group or firewall rule is configured to allow access, the FTP host name cannot be parsed, or the FTP username or password is incorrect. If the fault persists, contact customer service or technical support.
Cdm.0700	400	Failed to connect to the SFTP server.	The SFTP server fails to be connected.	It is possible that the network is disconnected, no security group or firewall rule is configured to allow access, the SFTP host name cannot be parsed, or the SFTP username or password is incorrect. If the fault persists, contact customer service or technical support.

Error Code	Status Code	Error	Description	Solution
Cdm.0800	400	Failed to connect to the OBS server.	The OBS server fails to be connected.	It is possible that the OBS endpoint is inconsistent with the current region, the AK/SK pair is incorrect, the AK/SK pair is not the one of the current user, or no security group or firewall rule is configured to allow access. If the fault persists, contact customer service or technical support.
Cdm.0801	400	OBS bucket [*%s*] unavailable.	The OBS bucket does not exist.	The OBS bucket may not exist or is not in the current region. Specify a correct OBS bucket and try again.
Cdm.0831	400	Failed to connect to the KODO server. Cause: *%s*.	The KODO server fails to be connected.	Contact customer service or technical support.
Cdm.0900	400	Table [*%s*] unavailable.	The table does not exist.	Specify a correct table name and try again.
Cdm.0901	400	Failed to connect to the database server. Cause: *%s*.	The database server fails to be connected.	Contact customer service or technical support.
Cdm.0902	400	Failed to execute the SQL statement. Cause: *%s*.	The SQL statement fails to be executed.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.

Error Code	Status Code	Error	Description	Solution
Cdm.0903	400	Failed to obtain metadata. Cause: *%s*.	Metadata fails to be obtained.	Check whether the quote character is correct or whether the database table exists when you create the link. If the fault persists, contact customer service or technical support.
Cdm.0904	400	Failed to retrieve data from the result. Cause: *%s*.	An error occurs when retrieving data from the result.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.0913	400	Schema and SQL cannot be left blank at the same time.	Either Schema or SQL must be specified.	Specify one of them and try again.
Cdm.0916	400	In incremental reading mode, the previous value must be specified.	The previous value is not specified in incremental reading.	Specify the previous value and try again.
Cdm.0917	400	Previous value cannot be obtained without field check.	The field is missing.	Contact customer service or technical support.
Cdm.0921	400	Unsupported type *%s*.	The type is invalid.	Specify a correct type and try again.
Cdm.0925	400	The partition field contains unsupported values.	The partition field contains unsupported values.	Correct the values and try again.

Error Code	Status Code	Error	Description	Solution
Cdm.0926	400	Failed to obtain the schema field. Cause: *%s*.	The schema field fails to be obtained.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.0927	400	The relay table cannot be empty.	The relay table cannot be empty.	Specify an empty relay table and try again.
Cdm.0928	400	Failed to transfer data from the relay table to the destination table.	An error occurs when transferring data from the relay table to the destination table.	Contact customer service or technical support.
Cdm.0931	400	The value of the schema field [*%s*] does not match that of the field [*%s*] in the result set.	The value of the schema field [*%s*] does not match that of the field [*%s*] in the result set.	Change the schema value to be the same as that in the result set and try again.
Cdm.0932	400	Failed to find the maximum value of the field.	The maximum value of the field cannot be found.	Contact customer service or technical support.
Cdm.0934	400	Tables with the same name exist in different schemas/ catalogs.	Tables with the same name exist in different schemas/catalogs.	Contact customer service or technical support.
Cdm.0936	400	The number of dirty data records reaches the upper limit.	The number of dirty data records reaches the upper limit.	Edit the job and increase the number of dirty data records.
Cdm.0940	400	Precise match of the table name failed.	Precise match of the table name failed.	Specify a correct table name and try again.

Error Code	Status Code	Error	Description	Solution
Cdm.0941	400	Failed to connect to the server. Cause: [*%s*].	The server fails to be connected.	Check whether the IP address, host name, and port number are correct, and whether the network security group and firewall are correctly configured. Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.0950	400	Failed to connect the authentication information to the database.	The authentication information cannot be connected to the database.	Correct the authentication information and try again.
Cdm.0962	400	The host IP address must be specified.	No host IP address is specified.	Specify the host IP address and try again.
Cdm.0963	400	The host port must be specified.	No host port is specified.	Specify the host port and try again.
Cdm.0964	400	The database must be specified.	No database is specified.	Specify a database and try again.
Cdm.1000	400	Hive table [*%s*] does not exist.	The Hive table does not exist.	Specify a correct Hive table name and try again.

Error Code	Status Code	Error	Description	Solution
Cdm.1010	400	Invalid URI *%s*. URI must be null or valid.	The URI is invalid.	Specify a correct URI and try again. Correct URI examples: • hdfs://example.com:8020/ • hdfs://example.com/ • file:/// • file:///tmp • file://localhost/tmp
Cdm.1011	400	Failed to connect to Hive. Cause: *%s*.	Hive fails to be connected.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.1100	400	Table [*%s*] unavailable.	The table does not exist.	Enter a correct table name and try again.
Cdm.1101	400	Failed to obtain the link. Cause: *%s*.	The link fails to be obtained.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.1102	400	Failed to create the table. Cause: *%s*.	The table fails to be created.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.1103	400	No rowkey is set.	No rowkey is set.	Set the rowkey and try again.

Error Code	Status Code	Error	Description	Solution
Cdm.1104	400	Failed to open the table. Cause: *%s*.	The table fails to be opened.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.1105	400	Failed to initialize the job. Cause: *%s*.	The job fails to be initialized.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.1111	400	The table name cannot be empty.	The table name is not specified.	Specify a correct table name and try again.
Cdm.1114	400	Rowkey is empty. Set it in field mapping.	Rowkey is empty.	Fix the error based on the error message.
Cdm.1115	400	**Columns** is empty. Set it in field mapping.	**Columns** is empty.	Fix the error based on the error message.
Cdm.1116	400	Duplicate column name. Reset it in the field mapping step.	The column name already exists.	Fix the error based on the error message.
Cdm.1117	400	Failed to check whether the table exists. Cause: *%s*.	An error occurs when checking whether the table exists.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.1118	400	Table *%s* does not contain the column family *%s*.	The table does not contain the specified column family.	Specify a column family and try again.

Error Code	Status Code	Error	Description	Solution
Cdm.1120	400	The table contains data. Clear the data or reset the parameter to determine whether to clear the table data before importing the table.	The table contains data. Clear the data or reset the parameter to determine whether to clear the table data before importing the table.	Fix the error based on the error message.
Cdm.1121	400	Failed to close the link. Cause: *%s*.	The link fails to be closed.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.1201	400	Failed to connect to the Redis server. Cause: *%s*.	The Redis server fails to be connected.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.1203	400	Failed to extract data from the Redis server. Cause: *%s*.	Data fails to be extracted from the Redis server.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.1205	400	The prefix of the Redis value cannot be empty.	The prefix of the Redis value cannot be empty.	Delete the whitespace before the Redis prefix and try again.
Cdm.1206	400	The storage type of the Redis value must be **string** or **hash**.	The storage type of the Redis value must be **string** or **hash**.	Fix the error based on the error message.

Error Code	Status Code	Error	Description	Solution
Cdm.1207	400	When the value storage type is **string**, **Value Delimiter** must be specified.	The value storage type is **string**, but **Value Delimiter** is not specified.	Specify a value delimiter and try again.
Cdm.1208	400	**columnList** of Redis must be specified.	**columnList** of Redis must be specified.	Specify **columnList** and try again.
Cdm.1209	400	Redis Key Delimiter cannot be empty.	Redis Key Delimiter cannot be empty.	Specify a correct delimiter and try again.
Cdm.1210	400	**primaryKeyList** of Redis must be specified.	**primaryKeyList** of Redis is not specified.	Specify **primaryKeyList** and try again.
Cdm.1211	400	**primaryKeyList** of Redis must exist in **columnList**.	**primaryKeyList** of Redis does not exist in **columnList**.	Specify **primaryKeyList** and try again.
Cdm.1213	400	**Redis Server Address** must be specified.	**Redis Server Address** is not specified.	Specify **Redis Server Address** and try again.
Cdm.1301	400	Failed to connect to the MongoDB server. Cause: *%s*.	The MongoDB server fails to be connected.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.1302	400	Failed to extract data from the MongoDB server. Cause: *%s*.	Data fails to be extracted from the MongoDB server.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.

Error Code	Status Code	Error	Description	Solution
Cdm.1304	400	The MongoDB server set must be specified.	The MongoDB server set is not specified.	Specify the MongoDB server set and try again.
Cdm.1305	400	**Server Address** of MongoDB must be specified.	**Server Address** of MongoDB is not specified.	Specify **Server Address** and try again.
Cdm.1306	400	The database name of the MongoDB service must be specified.	The database name of the MongoDB service is not specified.	Specify a database and try again.
Cdm.1307	400	**serverlist** of MongoDB must be specified.	**serverlist** of MongoDB is not specified.	Specify **serverlist** and try again.
Cdm.1501	400	Failed to connect to the Elasticsearch server. Cause: *%s*.	The Elasticsearch server fails to be connected.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.1502	400	Failed to write data to the Elasticsearch server. Cause: *%s*.	Data fails to be written to the Elasticsearch server.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.1503	400	Failed to close the Elasticsearch link. Cause: *%s*.	The Elasticsearch link fails to be closed.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.

Error Code	Status Code	Error	Description	Solution
Cdm.1504	400	Failed to obtain the Elasticsearch index. Cause: *%s*	An error occurs when obtaining the Elasticsearch index.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.1505	400	Failed to obtain the Elasticsearch type. Cause: *%s*	An error occurs when obtaining the Elasticsearch type.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.1506	400	Failed to obtain the Elasticsearch field. Cause: *%s*	An error occurs when obtaining the Elasticsearch file field.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.1508	400	The host name or IP address of the Elasticsearch server must be specified.	The host name or IP address of the Elasticsearch server is not specified.	Specify the host name or IP address and try again.
Cdm.1510	400	The Elasticsearch index must be specified.	The Elasticsearch index is not specified.	Specify an index and try again.
Cdm.1511	400	The Elasticsearch type must be specified.	The Elasticsearch type is not specified.	Specify a type and try again.
Cdm.1513	400	**columnList** must contain the field type definition.	**columnList** does not contain the field type definition.	Include the field type definition and try again.
Cdm.1514	400	**columnList** must contain **primaryKey**.	**columnList** does not contain **primaryKey**.	Specify **primaryKey** and try again.

Error Code	Status Code	Error	Description	Solution
Cdm.1516	400	Invalid column name *%s*.	The column name is invalid.	Enter a correct column name and try again.
Cdm.1517	400	Failed to obtain the number of documents.	An error occurs when obtaining the number of documents.	Contact customer service or technical support.
Cdm.1519	400	Data extraction exception.	An error occurs when extracting data.	Contact customer service or technical support.
Cdm.1601	400	Failed to connect to the server.	The server fails to be connected.	Contact customer service or technical support.
Cdm.1603	400	Failed to obtain the sample value of topic *%s*.	The sample value of topic *%s* fails to be obtained.	Contact customer service or technical support.
Cdm.1604	400	No data contained in topic*%s*.	No data exists in the topic.	Locate the cause. Alternatively, change the topic and try again.

