

ModelArts

API Reference

Issue 01
Date 2024-06-12



Copyright © Huawei Technologies Co., Ltd. 2024. All rights reserved.

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

Trademarks and Permissions



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

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

Notice

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

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

Huawei Technologies Co., Ltd.

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

Website: <https://www.huawei.com>

Email: support@huawei.com

Security Declaration

Vulnerability

Huawei's regulations on product vulnerability management are subject to the *Vul. Response Process*. For details about this process, visit the following web page:

<https://www.huawei.com/en/psirt/vul-response-process>

For vulnerability information, enterprise customers can visit the following web page:

<https://securitybulletin.huawei.com/enterprise/en/security-advisory>

Contents

1 Before You Start.....	1
1.1 Overview.....	1
1.2 API Calling.....	1
1.3 Endpoint.....	1
1.4 Constraints.....	1
1.5 Basic Concepts.....	2
2 API Overview.....	3
3 Calling APIs.....	11
3.1 Making an API Request.....	11
3.2 Authentication.....	15
3.3 Response.....	20
4 DevEnviron Management.....	22
4.1 Querying Notebook Instances.....	22
4.2 Creating a Notebook Instance.....	32
4.3 Querying Details of a Notebook Instance.....	44
4.4 Updating a Notebook Instance.....	52
4.5 Deleting a Notebook Instance.....	61
4.6 Saving a Running Instance as a Container Image.....	68
4.7 Obtaining the Available Flavors.....	72
4.8 Querying Flavors Available for a Notebook Instance.....	76
4.9 Querying the Available Duration of a Running Notebook Instance.....	81
4.10 Prolonging a Notebook Instance.....	83
4.11 Starting a Notebook Instance.....	85
4.12 Stopping a Notebook Instance.....	93
4.13 Obtaining the Notebook Instances with OBS Storage Mounted.....	100
4.14 OBS Storage Mounting.....	102
4.15 Obtaining Details About a Notebook Instance with OBS Storage Mounted.....	104
4.16 Unmounting OBS Storage from a Notebook Instance.....	106
4.17 Querying Supported Images.....	108
4.18 Registering a Custom Image.....	113
4.19 Obtaining User Image Groups.....	119
4.20 Obtaining Details of an Image.....	122

4.21 Deleting an Image.....	125
5 Training Management.....	130
5.1 Creating an Algorithm.....	130
5.2 Querying the Algorithm List.....	146
5.3 Querying Algorithm Details.....	156
5.4 Modifying an Algorithm.....	164
5.5 Deleting an Algorithm.....	180
5.6 Creating a Training Job.....	181
5.7 Querying the Details About a Training Job.....	218
5.8 Modifying the Description of a Training Job.....	236
5.9 Deleting a Training Job.....	238
5.10 Terminating a Training Job.....	239
5.11 Querying the Logs of a Specified Task in a Given Training Job (Preview).....	258
5.12 Querying the Logs of a Specified Task in a Training Job (OBS Link).....	259
5.13 Querying the Running Metrics of a Specified Task in a Training Job.....	261
5.14 Querying a Training Job List.....	263
5.15 Obtaining the General Specifications Supported by a Training Job.....	284
5.16 Obtaining the Preset AI Frameworks Supported by a Training Job.....	288
6 AI Application Management.....	292
6.1 Querying the AI Application List.....	292
6.2 Creating an AI Application.....	298
6.3 Obtaining Details About an AI Application.....	310
6.4 Deleting an AI application.....	316
7 Service Management.....	320
7.1 Obtaining Service Monitoring.....	320
7.2 Obtaining Services.....	323
7.3 Deploying Services.....	329
7.4 Obtaining Supported Service Deployment Specifications.....	345
7.5 Obtaining Service Details.....	350
7.6 Updating Service Configurations.....	358
7.7 Deleting a Service.....	368
7.8 Obtaining Dedicated Resource Pools.....	369
7.9 Obtaining Service Event Logs.....	373
7.10 Obtaining Service Update Logs.....	377
8 Resource Management.....	382
8.1 Configuration Management.....	382
8.1.1 Querying OS Configuration Parameters.....	382
8.2 Quota Management.....	384
8.2.1 Obtaining OS Quotas.....	384
8.3 Event Management.....	386
8.3.1 Obtaining the Event List.....	386

8.4 Resource Pool Job Management.....	390
8.4.1 Obtaining Jobs in a Dedicated Resource Pool.....	391
8.4.2 Obtaining Statistics About Dedicated Resource Pool Jobs.....	396
8.5 Resource Metrics.....	399
8.5.1 Obtaining the Real-Time Resource Usage.....	399
8.6 Plug-in Template Management.....	402
8.6.1 Querying a Plug-in Template.....	402
8.7 Tag Management.....	405
8.7.1 Creating a Resource Pool Tag.....	405
8.7.2 Deleting Tags of a Resource Pool.....	408
8.7.3 Querying All Tags of a Resource Pool.....	411
8.8 Network Management.....	414
8.8.1 Creating Network Resources.....	414
8.8.2 Obtaining Network Resources.....	421
8.8.3 Obtaining a Network Resource.....	427
8.8.4 Deleting a Network Resource.....	432
8.8.5 Updating a Network Resource.....	438
8.9 Node Management.....	446
8.9.1 Obtaining Nodes.....	446
8.9.2 Deleting Nodes in Batches.....	451
8.10 Resource Pool Management.....	452
8.10.1 Creating Resource Pools.....	452
8.10.2 Obtaining Resource Pools.....	470
8.10.3 Obtaining a Resource Pool.....	482
8.10.4 Deleting a Resource Pool.....	492
8.10.5 Updating a Resource Pool.....	503
8.10.6 Monitoring a Resource Pool.....	517
8.10.7 Obtaining Resource Pool Statistics.....	524
8.11 Resource Specifications Management.....	526
8.11.1 Obtaining Resource Specifications.....	527
9 Authorization Management.....	533
9.1 Viewing an Authorization List.....	533
9.2 Configuring Authorization.....	536
9.3 Deleting Authorization.....	538
9.4 Creating a ModelArts Agency.....	539
10 Use Cases.....	542
10.1 Creating a Development Environment Instance.....	542
10.2 Using PyTorch to Create a Training Job (New-Version Training).....	561
10.3 Managing ModelArts Authorization.....	576
11 Common Parameters.....	580
11.1 Status Code.....	580

11.2 Error Codes.....	584
11.3 Obtaining a Project ID and Name.....	680
11.4 Obtaining an Account Name and ID.....	680
11.5 Obtaining a Username and ID.....	681

1 Before You Start

1.1 Overview

ModelArts is a one-stop AI development platform geared toward developers and data scientists of all skill levels. It enables you to rapidly build, train, and deploy models anywhere (from the cloud to the edge), and manage full-lifecycle AI workflows. ModelArts accelerates AI development and fosters AI innovation with key capabilities, including data preprocessing and auto labeling, distributed training, automated model building, and one-click workflow execution. You can use ModelArts through open APIs.

1.2 API Calling

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

1.3 Endpoint

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

NOTICE

If an endpoint uses a domain name, configure the hosts file in the format of "{float-ip} {service_name}.{region_id}.{external_domain_name}" on the local PC. Contact the system administrator to obtain **float-ip**.

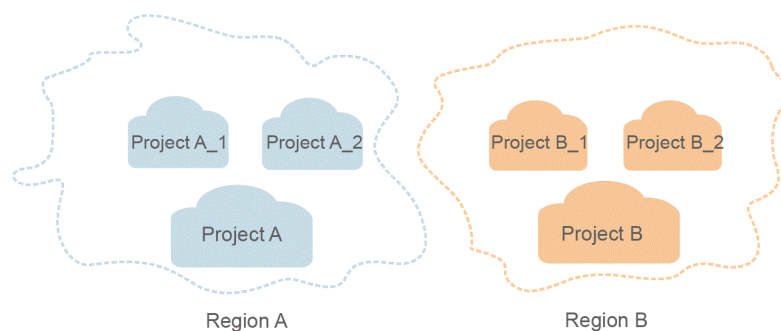
1.4 Constraints

- For more constraints, see API description.

1.5 Basic Concepts

- Account
An account is created upon successful registration with the cloud platform. 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.
- Region
A region is a physical location where a cloud service is deployed. Availability zones (AZ) in the same region can communicate with each other over an intranet but AZs in different regions cannot communicate with each other. By creating cloud resources in different regions, you can design applications to better meet customer requirements and comply with local laws and regulations.
- AZ
An AZ contains one or more physical data centers. It has independent cooling, fire extinguishing, moisture-proof, and electricity facilities. Within an AZ, computing, network, storage, and other resources are logically divided into multiple clusters. AZs within a region are interconnected using high-speed optical fibers to allow users to build cross-AZ high-availability systems.
- Project
Projects group and isolate compute, storage, and network resources across physical regions. A default project is provided for each region, and subprojects can be created under each default project. Users can be granted permissions to access all resources in a specific project. For more refined access control, create 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 isolation model



2 API Overview

All ModelArts APIs are proprietary.

You can use these APIs to manage development environments, training jobs, AI applications, and services.

DevEnviron APIs

Table 2-1 Development environment

Type	API	Description
Managing DevEnviron Instances	Obtaining Notebook Instances	Obtain the list of notebook instances meeting search criteria
	Creating a Notebook Instance	Create a notebook instance based on parameters such as the instance flavor, AI engine, and storage.
	Obtaining Details About a Notebook Instance	Obtain details about a notebook instance.
	Updating a Notebook Instance	Modify a notebook instance.
	Deleting a Notebook Instance	Delete the container and all storage resources of a notebook instance.
	Saving a Running Instance as a Container Image	Save the running instance as a container image. In the saved image, the installed dependency package (pip package) is retained. In VS Code remote development, the plug-ins installed on the server are retained.
	Obtaining Flavors Available for a Notebook Instance	Obtain available flavors.

Type	API	Description
	Obtaining Flavors Available for a Notebook Instance	Obtain the flavors available for a notebook instance.
	Obtaining the Available Duration of a Running Notebook Instance	Obtain the available duration of a running notebook instance.
	Prolonging a Notebook Instance	Prolong the available duration of a running notebook instance.
	Starting a Notebook Instance	Start a notebook instance.
	Stopping a Notebook Instance	Stop a notebook instance.
Dynamically Mounting OBS	Obtaining the Notebook Instances with OBS Storage Mounted	Obtain the notebook instances with OBS storage mounted.
	Dynamically Mounting OBS	Dynamically mount OBS to a notebook instance in running state.
	Obtaining Details About a Notebook Instance with OBS Storage Mounted	Obtain details about a notebook instance with OBS storage mounted.
	Dynamically Unmounting OBS	Dynamically unmount OBS from a notebook instance.
Image Management	Obtaining Supported Images	Obtain all images by page based on specified conditions.
	Registering a Custom Image	Register a custom image with ModelArts image management.
	Obtaining User Image Groups	Obtain the overview of user image information. Image names are used for aggregation.
	Obtaining Details About an Image	Obtain details about an image.
	Deleting an Image	Delete an image. For a private image, you can also delete its SWR image using parameters.

Training Management APIs

Table 2-2 Algorithm management APIs

API	Description
Creating an Algorithm	Create an algorithm.
Obtaining Algorithms	Obtain algorithms.
Obtaining Details About an Algorithm	Obtain a specified algorithm based on the algorithm ID.
Updating an Algorithm	Update an algorithm.
Deleting an Algorithm	Delete an algorithm.

Table 2-3 APIs for managing training jobs

API	Description
Creating a Training Job	Create a training job.
Obtaining Details About a Training Job	Obtain details about a training job.
Modifying the Description of a Training Job	Modify the description of a training job.
Deleting a Training Job	Delete a training job.
Terminate a Training Job	Terminate a training job. Only jobs in the creating, awaiting, or running state can be terminated.
Obtaining the Logs of a Specified Task in a Training Job (Preview)	Obtain the logs of a specified task in a training job (preview).
Obtaining the Logs of a Specified Task in a Training Job (OBS Link)	Obtain the logs of a specified task in a training job (OBS link). You can view all logs or download the logs.
Obtaining the Runtime Metrics of a Specified Task in a Training Job	Obtain the runtime metrics of a specified task in a training job.
Obtaining Training Jobs	Obtain the created training jobs by search criteria.

Table 2-4 APIs for resources and engine specifications

API	Description
Obtaining the General Specifications Supported by a Training Job	Obtain the general specifications supported by a training job.
Obtaining the Preset AI Frameworks Supported by a Training Job	Obtain the preset AI frameworks supported by a training job.

AI Application Management APIs

Table 2-5 APIs for managing AI applications

API	Description
Creating an AI Application	Create an AI application.
Obtaining an AI Application List	Obtain AI applications by different search criteria.
Obtaining Details About an AI Application	Obtain AI application details by ID.
Deleting an AI Application	Delete AI applications by ID. All versions of the AI application can be deleted.

Service Management APIs

Table 2-6 Service management APIs

API	Description
Deploying a Service	Deploy a service.
Obtaining Services	Obtain services.
Obtaining Service Details	Obtain service details by ID.
Updating Service Configurations	Update a model service.
Obtaining Service Monitoring	Obtain service monitoring information.
Obtaining Service Update Logs	Obtain the update logs of a real-time service.

API	Description
Obtaining Service Event Logs	Obtain service event logs, including service operation records, key actions during deployment, and deployment failure causes.
Deleting a Service	Delete a service.
Obtaining Supported Service Deployment Specifications	Obtain supported service deployment specifications.

Resource Management APIs

Table 2-7 Configuration management API

API	Description
Obtaining OS Configuration Parameters	Obtain the configuration parameters of the ModelArts OS service, such as the CIDR block and user resource quota.

Table 2-8 Configuration management API

API	Description
Obtaining OS Quotas	Obtain the quotas of some ModelArts OS resources, such as the quotas for resource pool quotas and networks.

Table 2-9 Tag management

API	Description
Adding Tags to a Resource Pool	Add tags to a dedicated resource pool.
Deleting Resource Pool Tags	Delete tags of a dedicated resource pool.
Viewing All Tags of a Resource Pool	View tags of a dedicated resource pool.

Table 2-10 Plug-in template management API

API	Description
Obtaining a Plug-in Template	Obtain details of a specified plug-in template.

Table 2-11 Node management APIs

API	Description
Obtaining Nodes	Obtain the list of nodes in a resource pool.
Deleting Nodes in a Batch	Delete nodes in a specific resource pool. At least one node must be reserved in the resource pool.

Table 2-12 Event management API

API	Description
Obtaining Events	Obtain events.

Table 2-13 Network management APIs

API	Description
Creating Network Resources	Create network resources.
Obtaining Network Resources	Obtain a list of network resources.
Obtaining a Network Resource	Obtain details about a specified network resource.
Deleting a Network Resource	Delete a specified network resource.
Updating a Network Resource	Update a specified network resource.

Table 2-14 Resource indicator management API

API	Description
Obtaining the Real-Time Resource Usage	Obtain the real-time usage of all resource pools in the current project.

Table 2-15 Resource pool management APIs

API	Description
Creating a Resource Pool	Create a resource pool.
Obtaining Resource Pools	Obtain resource pools.
Obtaining a Resource Pool	Obtain details about a specified resource pool.
Deleting a Resource Pool	Delete a specified resource pool.
Updating a Resource Pool	Update a specified resource pool.
Monitoring a Resource Pool	Obtain the monitoring information of a resource pool.
Collecting Resource Pool Statistics	Obtain the statistics of a resource pool.

Table 2-16 Resource flavor management APIs

API	Description
Obtaining Resource Flavors	Obtain resource flavors.

Table 2-17 APIs for managing resource pool jobs

API	Description
Obtaining Dedicated Resource Pool Jobs	Obtain dedicated resource pool jobs.
Obtaining Statistics for Dedicated Resource Pool Jobs	Obtain statistics for dedicated resource pool jobs.

Authorization Management APIs

Table 2-18 Authorization management APIs

API	Description
Viewing an Authorization List	View an authorization list.
Configuring Authorization	Configure ModelArts authorization. ModelArts functions such as training management, development environment, data management, and real-time services can be properly used only after required permissions are assigned.

API	Description
Deleting Authorization	Delete the authorization of a specified user or all users.
Creating a ModelArts Agency	Create a ModelArts agency for dependent services such as OBS, SWR, and IEF.

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

The format of a request URI is as follows:

{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 for the REST service endpoint. The endpoint varies depending on services in different regions. It can be obtained in Endpoint .
resource-path	Access path of an API for performing a specified operation. Obtain the path from the URI of an API. For example, the resource-path of the API used to obtain a user token is /v3/auth/tokens .
query-string	Query parameter, which is optional. Ensure that a question mark (?) is included before each query parameter that is in the format of " <i>Parameter name=Parameter value</i> ". For example, ? limit=10 indicates that a maximum of 10 data records will be displayed.

For example, to obtain an IAM token in a region, obtain the endpoint of IAM 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-endpoint}/v3/auth/tokens`

 **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-endpoint}/v3/auth/tokens`

Request Header

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

Table 3-3 describes the common request header fields to be added to the request.

Table 3-3 Common request header fields

Header	Description	Mandatory	Example
Content-type	Request body type or format. The default value is application/json .	Yes	application/json
Content-Length	Length of the request body. The unit is byte.	Mandatory for POST and PUT requests but must be left blank for GET requests	3495
X-Project-Id	Project ID. This parameter is used to obtain the token for each project.	No	e9993fc787d94b6c886cbaa340f9c0f4
X-Auth-Token	User token. It is a response to the API used to obtain a user token. This API is the only one that does not require authentication.	Mandatory for token-based authentication	None
X-Sdk-Date	Time when the request is sent. The time is in <i>YYYYMMDD'T'HHMMSS'Z'</i> format. The value is the current Greenwich Mean Time (GMT) time of the system.	Mandatory for AK/SK-based authentication, optional for PKI token-based authentication	20190307T101459Z
Authorization	Authentication information. The value is obtained from the request signature result and is required when the AK/SK are used to encrypt the signature. Type: string Default value: none	Mandatory for AK/SK-based authentication	SDK-HMAC-SHA256 Credential=ZIRRKMTWPTQFQI1WKNKB/20150907//ec2/sdk_request, SignedHeaders=content-type;host;x-sdk-date, Signature=55741b610f3c9fa3ae40b5a8021ebf7ebc2a28a603fc62d25cb3bfe6608e1994

Header	Description	Mandatory	Example
Host	<p>Information about the requested server. The value can be obtained from the URL of the service API.</p> <p>This value is <i>host name[:port number]</i>.</p> <p>If the port number is not specified, the default port is used. The default port number for https is 443.</p>	Mandatory for AK/SK-based authentication	code.test.com or code.test.com:443

 NOTE

In addition to supporting authentication using tokens, APIs support authentication using AK/SK, which uses SDK to sign a request. During the signature, the **Authorization** (signature authentication) and **X-Sdk-Date** (time when a request is sent) headers are automatically added to the request.

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

```
POST https://{iam-endpoint}/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.

If an API is used to obtain a user token, the request parameters and parameter description can be obtained from the API request. The following provides an example request with a body included. Replace *user_name*, *domain_name*, and *user_password* with the actual username, account name, and login password, respectively. **project_name** is the project name. For details, see [Obtaining a Username](#), [Obtaining an Account Name and ID](#), and [Obtaining a Project Name](#).

 NOTE

The **scope** parameter specifies where a token takes effect. In the example, the token takes effect only for the resources in a specified project. ModelArts uses a region-specific endpoint to call this API. Set **scope** to **project**. You can set **scope** to an account or a project under an account.

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

```
{
  "auth": {
    "identity": {
      "methods": ["password"],
      "password": {
        "user": {
          "name": "Username",
          "password": "User password",
          "domain": {
            "name": "Domain name"
          }
        }
      }
    }
  },
  "scope": {
    "project": {
      "name": "project_name"
    }
  }
}
```

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

3.2 Authentication

Requests for calling an API can be authenticated using either of the following methods: AK/SK-based authentication: Requests are authenticated by encrypting the request body using an AK/SK pair.

- Token-based authentication: Requests are authenticated using a token.
- AK/SK authentication: Requests are encrypted using the access key ID (AK) and secret access key (SK).

Token-based Authentication

NOTE

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

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

In [Making an API Request](#), the process of calling the API used to obtain a user token is described.

```
{
  "auth": {
    "identity": {
      "methods": [
        "password"
      ],
      "password": {
        "user": {
          "name": "user_name",
          "password": "user_password",
          "domain": {
```

```
        "name": "domain_name"
      }
    }
  },
  "scope": {
    "project": {
      "name": "project_name"
    }
  }
}
```

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

```
POST https://{endpoint}/v1/{project_id}/services
Content-Type: application/json
X-Auth-Token: ABCDEFJ....
```

AK/SK-based Authentication

An AK/SK is used to verify the identity of a request sender. In AK/SK-based authentication, a signature needs to be obtained and then added to requests.

NOTE

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.

The following uses a demo project to show how to sign a request and use an HTTP client to send an HTTPS request.

Download the demo project at <https://github.com/api-gate-way/SdkDemo>.

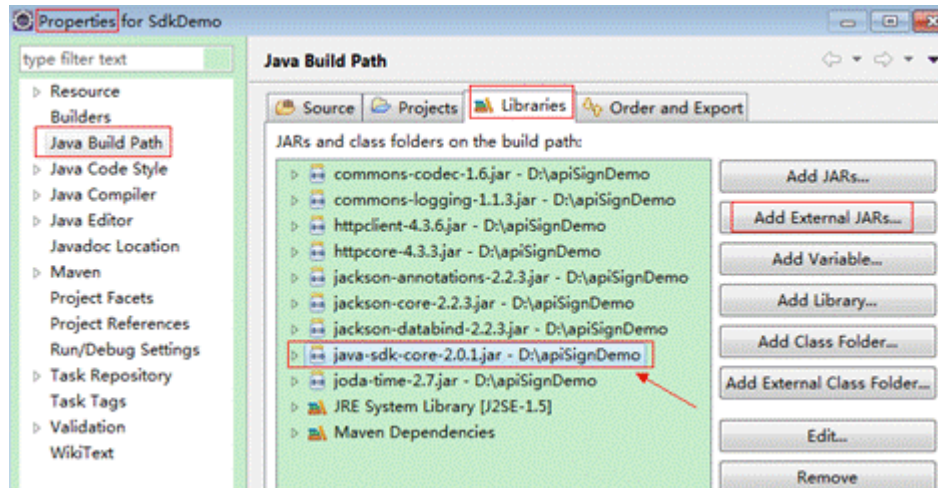
NOTE

ModelArts is a regional service. Specify the project ID when calling the API. Therefore, before running the demo project, add the project ID (**X-Project-Id**) in line 110 of `access()` in `AccessServiceImpl.java`. For details about how to obtain a project ID, see [Obtaining a Project ID and Name](#).

If you do not need the demo project, visit the following URL to download the API Gateway signing SDK:

Obtain the API Gateway signing SDK from the enterprise administrator.

Decompress the downloaded package and reference the obtained JAR files as dependencies, as highlighted in the following figure.



Step 1 Generate an AK/SK. (If an AK/SK file has already been obtained, skip this step and locate the downloaded AK/SK file. Generally, the file name will be **credentials.csv**.)

1. Log in to the console, enter the **My Credentials** page, and choose **Access Keys > Create Access Key**.
2. In the **Create Access Key** dialog box that is displayed, use the login password for verification.
3. Click **OK**, open the **credentials.csv** file, and save the key file as prompted. The access key file is saved in the default downloads folder of the browser. Then, the access key (**Access Key Id** and **Secret Access Key**) is obtained.

Step 2 Download and decompress the demo project.

Step 3 Import the demo project to Eclipse.

Figure 3-1 Selecting Existing Projects into Workspace

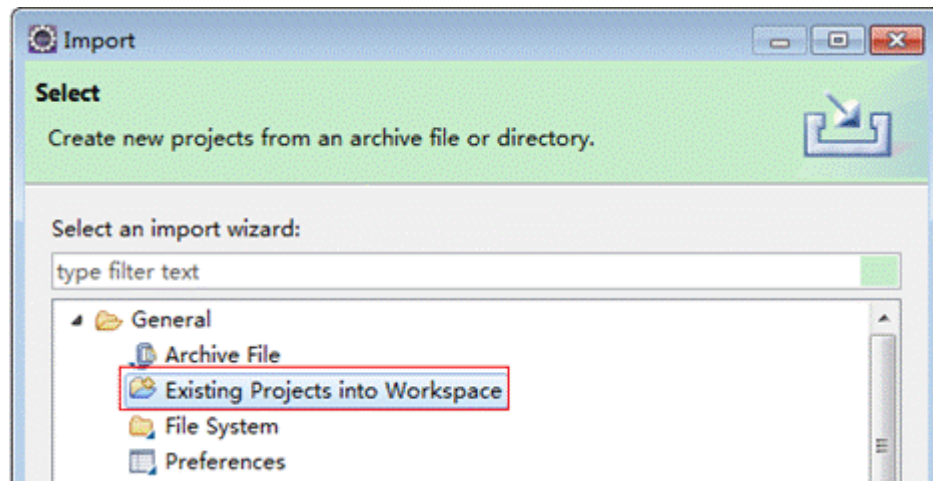


Figure 3-2 Selecting the demo project

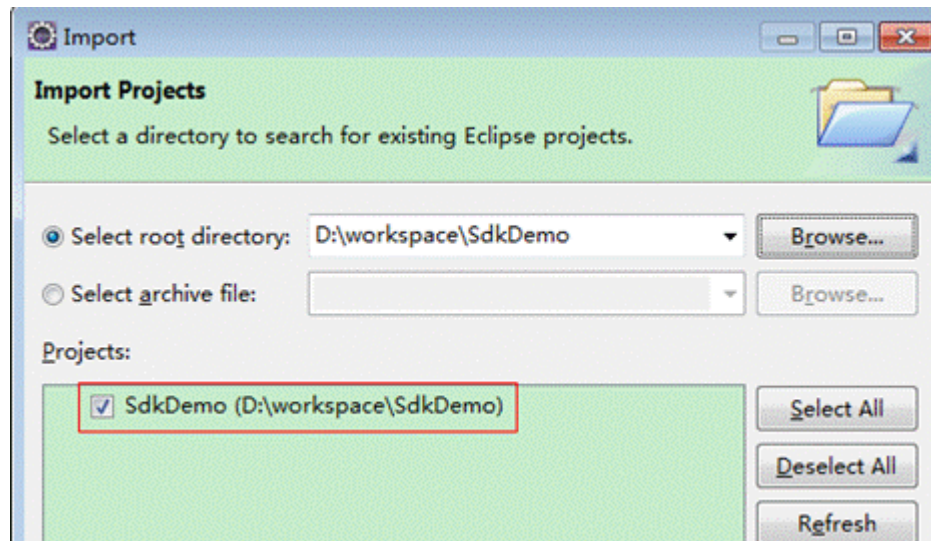
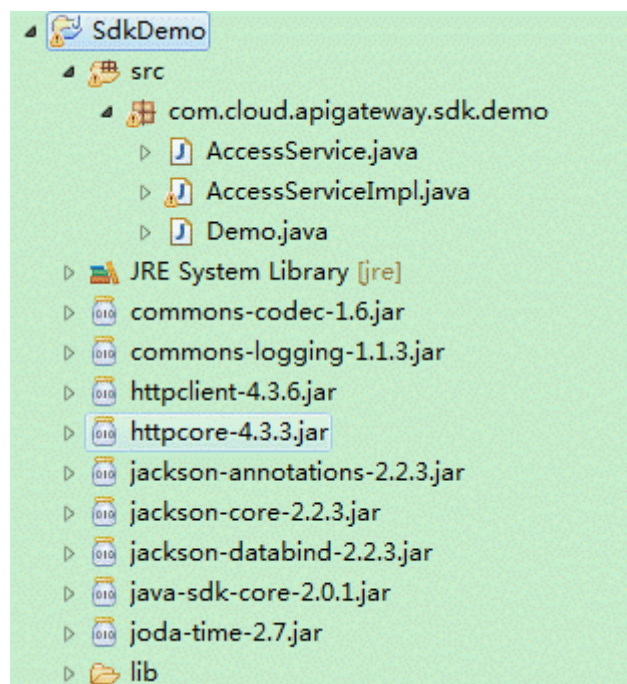


Figure 3-3 Structure of the demo project



Step 4 Sign the request.

The request signing method is integrated in the JAR files imported in [Step 3](#). The request needs to be signed before it is sent. The signature will then be added as part of the HTTP header to the request.

The demo code is classified into the following classes to demonstrate signing and sending the HTTP request:

- **AccessService**: abstract class that merges the GET, POST, PUT, and DELETE methods into the **access** method
- **Demo**: execution entry used to simulate the sending of GET, POST, PUT, and DELETE requests

- **AccessServiceImpl**: implementation of the **access** method, which contains the code required for communication with API Gateway

The following describes how to call a POST method to sign the request.

1. Add a request header.

Comment out the following second line of code in the **AccessServiceImpl.java** file, and specify the project ID.

```
//TODO: Add special headers.  
//request.addHeader("X-Project-Id", "xxxxx");
```

2. Edit the **main()** method in the Demo.java file, and replace the bold text with actual values.

As shown in the following code, if you use other methods such as POST, PUT, and DELETE, see the corresponding comment. Replace the values of **region**, **serviceName**, **ak**, **sk**, and **url**. The URL for obtaining the VPC is used in the sample project. Replace it with the actual URL. For details about how to obtain the endpoint, see [Regions and Endpoints](#).

```
//TODO: Replace the value of region with the actual region where the service to be accessed is located.
```

```
private static final String region = "";
```

```
//TODO: Replace vpc with the name of the service you want to access. For example, ecs, vpc, iam, and elb.
```

```
private static final String serviceName = "";
```

```
public static void main(String[] args) throws UnsupportedEncodingException  
{
```

```
//TODO: Replace the values of ak and sk with the AK/SK obtained on the My Credentials page.
```

```
String ak = "ZIRRKMTWP*****1WKNKB";
```

```
String sk = "Us0mdMNHk*****YrCnW0ecfzl";
```

```
//TODO: To specify a project ID (multi-project scenarios), add the X-Project-Id header.
```

```
//TODO: To access a global service, such as IAM, DNS, CDN, and TMS, add the X-Domain-Id header to specify an account ID.
```

```
//TODO: To add a header, find "Add special headers" in the AccessServiceImpl.java file.
```

```
//TODO: Test the API.
```

```
String url = "https://{Endpoint}/v1/{project_id}/vpcs";
```

```
get(ak, sk, url);
```

```
//TODO: When creating a VPC, replace {project_id} in postUrl with the actual value.
```

```
//String postUrl = "https://serviceEndpoint/v1/{project_id}/cloudservers";
```

```
//String postbody = "{\"vpc\": {\"name\": \"vpc\", \"cidr\": \"192.168.0.0/16\"}}";
```

```
//post(ak, sk, postUrl, postbody);
```

```
//TODO: When querying a VPC, replace {project_id} in url with the actual value.
```

```
//String url = "https://serviceEndpoint/v1/{project_id}/vpcs/{vpc_id}";
```

```
//get(ak, sk, url);
```

```
//TODO: When updating a VPC, replace {project_id} and {vpc_id} in putUrl with the actual values.
```

```
//String putUrl = "https://serviceEndpoint/v1/{project_id}/vpcs/{vpc_id}";
```

```
//String putbody = "{\"vpc\": {\"name\": \"vpc1\", \"cidr\": \"192.168.0.0/16\"}}";
```

```
//put(ak, sk, putUrl, putbody);
```

```
//TODO: When deleting a VPC, replace {project_id} and {vpc_id} in deleteUrl with the actual values.
```

```
//String deleteUrl = "https://serviceEndpoint/v1/{project_id}/vpcs/{vpc_id}";
```

```
//delete(ak, sk, deleteUrl);
}
```

3. Compile the code and call the API.

In the **Package Explorer** area on the left, right-click **Demo.java** and choose **Run AS > Java Application** from the shortcut menu to run the demo code.

You can view the API call logs on the console.

----End

3.3 Response

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

Status Code

A status code is a group of digits, ranging from 1xx to 5xx. 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**.

Figure 1 shows the response header fields for 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.

Figure 3-4 Header fields of the response to the request for obtaining a user token

```
connection → keep-alive
content-type → application/json
date → Tue, 12 Feb 2019 06:52:13 GMT
server → Web Server
strict-transport-security → max-age=31536000; includeSubdomains;
transfer-encoding → chunked
via → proxy A
x-content-type-options → nosniff
x-download-options → noopen
x-frame-options → SAMEORIGIN
x-iam-trace-id → 218d45ab-d674-4995-af3a-2d0255ba41b5
x-subject-token → MIIVXQVJKoZlIhvcNAQcCoIYTJCCGEoCAQEExDTALBglghkgBZQMEAgEwgharBgkqhkiG9w0BBwGgghacBIIWmHsidG9rZW4iOnsiZXhwaXJlc19hdCI6ijlwMTktMDItMTNUMC
fj3KJs6YgKnpVNRbW2eZ5eb78SZOkqjACgkqO1wi4JlGzrpdl8LGXK5tdffq4lqHCYb8P4NaY0NyejcAgzJVeFYtLWT1GSO0zxKZmlQHqJ82HBqHdglZO9fuEbL5dMhdavj+33wEl
xHRCe9I87o+k9-
j+CMZSEB7bUGd5Uj6eRASXI1jipPEGA270g1FruooL6jgglFkNPQuFSOU8+uSsttVwRtNfsC+qTp22Rkd5MCqFGQ8LcuUxC3a+9CMBnOintWW7oeRUVhVpxk8pxiX1wTEboX-
RzT6MUbpvGw-oPNFYxJECKnoH3HRozv0vN--n5d6Nbxg==
x-xss-protection → 1; mode=block;
```

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": "aaa",
            .....

```

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 error response body, **error_code** is an error code, and **error_msg** provides information about the error. For more details, see [Error Codes](#).

4 DevEnviron Management

4.1 Querying Notebook Instances

Function

This API is used to query notebook instances based on specified search criteria.

Constraints

None

URI

GET /v1/{project_id}/notebooks

Table 4-1 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .

Table 4-2 Query Parameters

Parameter	Mandatory	Type	Description
feature	No	String	Instance type. The default value is NOTEBOOK . Options: <ul style="list-style-type: none"> • DEFAULT: free CodeLab instance. You can create only one. • NOTEBOOK: billed instance.
limit	No	Integer	Number of records on each page. The default value is 10 .
name	No	String	Instance name, which contains a maximum of 128 characters and can consist of uppercase and lowercase letters, digits, hyphens (-), and underscores (_). Fuzzy match is supported.
pool_id	No	String	ID of a dedicated resource pool
offset	No	Integer	Start offset of the records on each page. The default value is 0 .
owner	No	String	User ID of the instance, which contains 32 characters, including lowercase letters and digits. This parameter is valid only when the account is a big account or has the admin permission. Generally, the value is the ID of the current login user.
sort_dir	No	String	Sorting mode. The options are ASC (ascending order) and DESC (descending order). The default value is DESC .
sort_key	No	String	Sorting fields. Separate multiple fields with commas (,).

Parameter	Mandatory	Type	Description
status	No	String	Instance status. Options: <ul style="list-style-type: none"> • INIT: The instance is being initialized. • CREATING: The instance is being created. • STARTING: The instance is starting. • STOPPING: The instance is being stopped. • DELETING: The instance is being deleted. • RUNNING: The instance is running. • STOPPED: The instance has been stopped. • SNAPSHOTTING: The image of the instance is being saved. • CREATE_FAILED: Creating the instance failed. • START_FAILED: Starting the instance failed. • DELETE_FAILED: Deleting the instance failed. • ERROR: An error occurred. • DELETED: The instance has been deleted. • FROZEN: The instance is frozen.
workspaceId	No	String	Workspace ID. If no workspaces are available, the default value is 0 .

Request Parameters

None

Response Parameters

Status code: 200

Table 4-3 Response body parameters

Parameter	Type	Description
current	Integer	Current page
data	Array of NotebookResp objects	Data
pages	Integer	Total pages
size	Integer	Number of records on each page
total	Long	Total records

Table 4-4 NotebookResp

Parameter	Type	Description
action_progress	Array of JobProgress objects	Instance initialization progress.
description	String	Instance description
endpoints	Array of EndpointsRes objects	This section describes how to configure the local IDE (such as PyCharm and VS Code) or SSH client to remotely access a notebook instance through SSH.
fail_reason	String	Instance failure cause
flavor	String	Instance flavor
id	String	Instance ID.
image	Image object	Instance image
lease	Lease object	Countdown to automatic instance stop.
name	String	Instance name
pool	Pool object	Basic information about the dedicated pool, which is returned by the instance created in the dedicated pool.

Parameter	Type	Description
status	String	Instance status. Options: <ul style="list-style-type: none"> • INIT: The instance is being initialized. • CREATING: The instance is being created. • STARTING: The instance is starting. • STOPPING: The instance is being stopped. • DELETING: The instance is being deleted. • RUNNING: The instance is running. • STOPPED: The instance has been stopped. • SNAPSHOTTING: The image of the instance is being saved. • CREATE_FAILED: Creating the instance failed. • START_FAILED: Starting the instance failed. • DELETE_FAILED: Deleting the instance failed. • ERROR: An error occurred. • DELETED: The instance has been deleted. • FROZEN: The instance is frozen.
token	String	Token information used for notebook authentication.
url	String	URL for accessing the notebook instance
volume	VolumeRes object	Storage volume
workspace_id	String	Workspace ID. If no workspaces are available, the default value is 0 .
feature	String	Instance type. Options: <ul style="list-style-type: none"> • DEFAULT: free CodeLab instance. You can create only one. • NOTEBOOK: billed instance.

Table 4-5 JobProgress

Parameter	Type	Description
notebook_id	String	Instance ID.

Parameter	Type	Description
status	String	Job status in a specified step. Options: <ul style="list-style-type: none"> ● WAITING: The job is awaiting. ● PROCESSING: The job is being processed. ● FAILED: The job failed. ● COMPLETED: The job is complete.
step	Integer	Job step. Options: <ul style="list-style-type: none"> ● 1: Prepare storage. ● 2: Prepare compute resources. ● 3: Configure the network. ● 4: Initialize the instance.
step_description	String	Description of a step in a job.

Table 4-6 EndpointsRes

Parameter	Type	Description
allowed_access_ips	Array of strings	Whitelist of public IP addresses that are allowed to access the notebook instance through SSH. By default, all public IP addresses can access the notebook instance. If this parameter is specified, only the clients with the specified IP addresses can access the notebook instance.
dev_service	String	Supported service. Options: <ul style="list-style-type: none"> ● NOTEBOOK: You can access notebook instances using HTTPS. ● SSH: You can remotely access the notebook instance through SSH.
ssh_keys	Array of strings	List of SSH key pairs. You can set multiple key pairs to access an SSH instance at the same time.

Table 4-7 Image

Parameter	Type	Description
arch	String	Processor architecture supported by the image. Options: <ul style="list-style-type: none"> ● X86_64: x86 architecture ● AARCH64: Arm architecture

Parameter	Type	Description
create_at	Long	Specifies the time (UTC ms) when the image is created.
description	String	Image description with a maximum of 512 characters
dev_services	Array of strings	Services supported by the image. Options: <ul style="list-style-type: none"> • NOTEBOOK: You can access the notebook instance using HTTPS. • SSH: You can remotely access the notebook instance from a local IDE through SSH.
id	String	ID of the image used for creating notebook instances. The ID is in Universally Unique Identifier (UUID) format. For details, see Querying Supported Images .
name	String	Image name, which contains a maximum of 512 characters, including lowercase letters, digits, hyphens (-), underscores (_), and periods (.)
namespace	String	Organization to which the image belongs. You can create and view the organization on the Organization Management page of the SWR console.
origin	String	Image source, which defaults to CUSTOMIZE . Options: <ul style="list-style-type: none"> • CUSTOMIZE: user-defined image • IMAGE_SAVE: image saved using a development environment instance
resource_categories	Array of strings	Flavors supported by the image. Options: <ul style="list-style-type: none"> • CPU • GPU
service_type	String	Supported image types. Options: <ul style="list-style-type: none"> • COMMON: common image • INFERENCE: image used for inference • TRAIN: image used for training • DEV: image used for development and debugging • UNKNOWN: image whose supported services are not specified
size	Long	Specifies the image size, in KB.

Parameter	Type	Description
status	String	Image status. Options: <ul style="list-style-type: none"> • INIT: The image is being initialized. • CREATING: The image is being saved. In this case, the notebook instance is unavailable. • CREATE_FAILED: Saving the image failed. • ERROR: An error occurs. • DELETED: The image has been deleted. • ACTIVE: The image has been saved, which you can view on the SWR console and use to create notebook instances.
status_message	String	Build information during image saving
support_res_categories	Array of strings	Flavors supported by the image. Options: <ul style="list-style-type: none"> • CPU • GPU
swr_path	String	SWR image address
tag	String	Image tag
type	String	Image type. Options: <ul style="list-style-type: none"> • BUILD_IN: built-in system image • DEDICATED: private image
update_at	Long	Specifies the time (UTC ms) when the image was last updated.
visibility	String	Image visibility. Options: <ul style="list-style-type: none"> • PRIVATE: private image • PUBLIC: All users can perform read-only operations based on the image ID.
workspace_id	String	Workspace ID. If no workspaces are available, the default value is 0 .

Table 4-8 Lease

Parameter	Type	Description
create_at	Long	Time (UTC) when the instance is created, accurate to millisecond.

Parameter	Type	Description
duration	Long	Instance running duration, which is calculated based on the instance creation time. If the instance creation time plus the duration is greater than the current time, the system automatically stops the instance.
enable	Boolean	Whether to enable auto stop of the instance.
type	String	Indicates the automatic stop type.
update_at	Long	Time (UTC) when the instance is last updated (excluding the keepalive heartbeat time), accurate to millisecond.

Table 4-9 Pool

Parameter	Type	Description
id	String	ID of a dedicated resource pool
name	String	Name of a dedicated resource pool

Table 4-10 VolumeRes

Parameter	Type	Description
capacity	Integer	Storage capacity. The default value is 5 GB for EVS and 50 GB for EFS. The maximum value is 4096 GB.
category	String	Supported storage types. For details about the differences between the storage types, see Selecting Storage in DevEnviron . Options: <ul style="list-style-type: none"> • SFS: Scalable File Service • EVS
mount_path	String	Directory of the notebook instance to which OBS storage is mounted. Currently, the directory is <code>/home/ma-user/work/</code> .
ownership	String	Owner to which the resource belongs. Options: <ul style="list-style-type: none"> • MANAGED: Resources are managed by service. • DEDICATED: Resources are managed by user account. This mode is supported only when the instance category is EFS.

Parameter	Type	Description
status	String	EVS disk capacity expansion status, which is RESIZING during capacity expansion and does not affect the instance.

Example Requests

```
GET https://{endpoint}/v1/{project_id}/notebooks
```

Example Responses

Status code: 200

OK

```
{
  "current" : 0,
  "data" : [ {
    "description" : "api-test",
    "feature" : "DEFAULT",
    "flavor" : "modelarts.vm.cpu.free",
    "id" : "f9937afa-ca78-45b6-bc12-7ecf42553c48",
    "image" : {
      "id" : "e1a07296-22a8-4f05-8bc8-e936c8e54090",
      "name" : "notebook2.0-mul-kernel-cpu-cp36",
      "swr_path" : "swr.xxxx.com/atelier/notebook2.0-mul-kernel-cpu-cp36:3.3.2-release_v1",
      "type" : "BUILD_IN"
    },
    "lease" : {
      "create_at" : 1638841805439,
      "duration" : 3600000,
      "enable" : true,
      "update_at" : 1638841805439
    },
    "name" : "notebook_5ee4bf0e",
    "status" : "CREATING",
    "token" : "58ba50c6-e8ff-245c-4840-49e51aa70737",
    "volume" : {
      "category" : "EVS",
      "ownership" : "MANAGED",
      "mount_path" : "/home/ma-user/work/",
      "capacity" : 50
    },
    "workspace_id" : "0"
  }, {
    "description" : "api-test",
    "feature" : "NOTEBOOK",
    "flavor" : "modelarts.vm.cpu.2u",
    "id" : "f9937afa-4451-42db-a76b-72d624749f66",
    "image" : {
      "id" : "e1a07296-22a8-4f05-8bc8-e936c8e54090",
      "name" : "notebook2.0-mul-kernel-cpu-cp36",
      "swr_path" : "swr.xxxx.com/atelier/notebook2.0-mul-kernel-cpu-cp36:3.3.2-release_v1",
      "type" : "BUILD_IN"
    },
    "lease" : {
      "create_at" : 1638841744515,
      "duration" : 3600000,
      "enable" : true,
      "update_at" : 1638841744515
    },
    "name" : "notebooks_test",
    "status" : "CREATING",
```

```
"token" : "3eff13f2-3d70-5456-6dc7-e3f99f562022",  
"volume" : {  
  "category" : "EVS",  
  "ownership" : "MANAGED",  
  "mount_path" : "/home/ma-user/work/",  
  "capacity" : 50  
},  
"workspace_id" : "0"  
}],  
"pages" : 1,  
"size" : 10,  
"total" : 2  
}
```

Status Codes

Status Code	Description
200	OK
401	Unauthorized
403	Forbidden
404	Not Found

Error Codes

See [Error Codes](#).

4.2 Creating a Notebook Instance

Function

This API is used to create a notebook instance based on the specified flavor, AI engine image, and storage. You can access the instance through a web page or SSH client. Calling this API is an asynchronous operation. The job status can be obtained by calling the API for obtaining details about a development environment instance.

Constraints

None

URI

POST /v1/{project_id}/notebooks

Table 4-11 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .

Request Parameters

Table 4-12 Request body parameters

Parameter	Mandatory	Type	Description
description	No	String	Instance description. The value can contain a maximum of 512 characters and cannot contain the following special characters: &<>"/. By default, this parameter is left blank.
duration	No	Long	Running duration starting from the current time. It is recommended that this attribute be configured in leaseReq. If this attribute has a value, the value of leaseReq is ignored and the automatic instance stop type is scheduled stop. Unit: ms.
endpoints	No	Array of EndpointsReq objects	This section describes how to configure the local IDE (such as PyCharm and VS Code) or SSH client to remotely access a notebook instance through SSH.
feature	No	String	Instance type. The default value is NOTEBOOK . Options: <ul style="list-style-type: none"> • DEFAULT: free CodeLab instance. You can create only one. • NOTEBOOK: billed instance.

Parameter	Mandatory	Type	Description
flavor	Yes	String	<p>Instance flavor. The following flavors are for reference only. Obtain the flavors displayed on the console in the target region.</p> <ul style="list-style-type: none"> modelarts.vm.cpu.2u: General-purpose Intel CPU flavor, ideal for rapid data exploration and experiments modelarts.vm.cpu.8u: General computing-plus Intel CPU flavors, ideal for compute-intensive applications modelarts.bm.gpu.v100NV32: One NVIDIA V100 GPU with 32 GB of memory, ideal for training and debugging deep learning algorithms modelarts.bm.snt9.xlarge.1: One Arm-powered Ascend Snt9 processor with 32 GB of GPU memory, ideal for training and debugging deep learning models modelarts.bm.snt9.xlarge.2: Two Arm-powered Ascend Snt9 processors with 32 GB of GPU memory, ideal for training and debugging deep learning models modelarts.bm.snt9.xlarge.8: Eight Arm-powered Ascend Snt9 processors with 32 GB of GPU memory, ideal for training and debugging deep learning models
image_id	Yes	String	<p>ID of the image used for creating notebook instances. The ID is in Universally Unique Identifier (UUID) format. For details, see Querying Supported Images.</p>

Parameter	Mandatory	Type	Description
name	Yes	String	Specifies the instance name. The value contains a maximum of 128 characters, The name can contain uppercase letters, lowercase letters, digits, hyphens (-), and underscores (_). The name can be duplicate.
pool_id	No	String	ID of a dedicated resource pool, for example, pool41664192 . This parameter is mandatory if a dedicated resource pool needs to be specified for creating an instance.
volume	Yes	VolumeReq object	Storage volume.
workspace_id	No	String	Workspace ID. If no workspaces are available, the default value is 0 .
hooks	No	CustomHooks object	Customized startup script hook configuration.
lease	No	LeaseReq object	Instance lease configuration.

Table 4-13 EndpointsReq

Parameter	Mandatory	Type	Description
allowed_access_ips	No	Array of strings	Public IP addresses that can remotely access the notebook instance. A maximum of five public IP addresses are supported.
dev_service	No	String	Supported service. Options: <ul style="list-style-type: none"> • NOTEBOOK: You can access notebook instances using HTTPS. • SSH: You can remotely access the notebook instance through SSH.

Parameter	Mandatory	Type	Description
ssh_keys	No	Array of strings	Name of the SSH key pair, which can be created and viewed on the Key Pair page of the Elastic Cloud Server (ECS) console.

Table 4-14 VolumeReq

Parameter	Mandatory	Type	Description
capacity	No	Integer	Storage capacity. The default value is 5 GB for EVS. The maximum value is 4096 GB.
category	Yes	String	Supported storage types. For details about the differences between the storage types, see Selecting Storage in DevEnviron . Options: <ul style="list-style-type: none"> • SFS: Scalable File Service • EVS
ownership	Yes	String	Owner to which the resource belongs. Options: <ul style="list-style-type: none"> • MANAGED: Resources are managed by service. • DEDICATED: Resources are managed by user account. This mode is supported only when the instance category is EFS.
uri	No	String	URI of the EFS. This parameter is mandatory only when the instance category is EFS and ownership is DEDICATED . An example URI is 192.168.0.1:/user-9sfdsgdfgh5ea4d56871e75d6966aa274/mount/ .

Table 4-15 CustomHooks

Parameter	Mandatory	Type	Description
container_hooks	No	ContainerHooks object	Container-related hook configuration.

Table 4-16 ContainerHooks

Parameter	Mandatory	Type	Description
post_start	No	Config object	Configure the script after the service is started.
pre_start	No	Config object	Configure the script before starting the service.

Table 4-17 Config

Parameter	Mandatory	Type	Description
script	No	String	User-defined script content (Base64 encoding) or absolute script path.
type	No	String	Script type. The options are as follows: <ul style="list-style-type: none"> The script content (Base64 encoding) must be specified in COMMAND script. Default value of SCRIPT. The script path must be specified in the script.

Table 4-18 LeaseReq

Parameter	Mandatory	Type	Description
duration	No	Long	Indicates the running duration starting from the current time. After the running duration expires, the system automatically stops. Unit: ms.
type	No	String	Automatic stop type. The default value is timing.

Response Parameters

Status code: 200

Table 4-19 Response body parameters

Parameter	Type	Description
action_progress	Array of JobProgress objects	Instance initialization progress.
description	String	Instance description
endpoints	Array of EndpointsRes objects	This section describes how to configure the local IDE (such as PyCharm and VS Code) or SSH client to remotely access a notebook instance through SSH.
fail_reason	String	Instance failure cause
flavor	String	Instance flavor
id	String	Instance ID.
image	Image object	Instance image
lease	Lease object	Countdown to automatic instance stop.
name	String	Instance name
pool	Pool object	Basic information about the dedicated pool, which is returned by the instance created in the dedicated pool.
status	String	Instance status. Options: <ul style="list-style-type: none"> • INIT: The instance is being initialized. • CREATING: The instance is being created. • STARTING: The instance is starting. • STOPPING: The instance is being stopped. • DELETING: The instance is being deleted. • RUNNING: The instance is running. • STOPPED: The instance has been stopped. • SNAPSHOTTING: The image of the instance is being saved. • CREATE_FAILED: Creating the instance failed. • START_FAILED: Starting the instance failed. • DELETE_FAILED: Deleting the instance failed. • ERROR: An error occurred. • DELETED: The instance has been deleted. • FROZEN: The instance is frozen.
token	String	Token information used for notebook authentication.

Parameter	Type	Description
url	String	URL for accessing the notebook instance
volume	VolumeRes object	Storage volume
workspace_id	String	Workspace ID. If no workspaces are available, the default value is 0 .
feature	String	Instance type. Options: <ul style="list-style-type: none"> • DEFAULT: free CodeLab instance. You can create only one. • NOTEBOOK: billed instance.

Table 4-20 JobProgress

Parameter	Type	Description
notebook_id	String	Instance ID.
status	String	Job status in a specified step. Options: <ul style="list-style-type: none"> • WAITING: The job is awaiting. • PROCESSING: The job is being processed. • FAILED: The job failed. • COMPLETED: The job is complete.
step	Integer	Job step. Options: <ul style="list-style-type: none"> • 1: Prepare storage. • 2: Prepare compute resources. • 3: Configure the network. • 4: Initialize the instance.
step_description	String	Description of a step in a job.

Table 4-21 EndpointsRes

Parameter	Type	Description
allowed_access_ips	Array of strings	Whitelist of public IP addresses that are allowed to access the notebook instance through SSH. By default, all public IP addresses can access the notebook instance. If this parameter is specified, only the clients with the specified IP addresses can access the notebook instance.

Parameter	Type	Description
dev_service	String	Supported service. Options: <ul style="list-style-type: none"> • NOTEBOOK: You can access notebook instances using HTTPS. • SSH: You can remotely access the notebook instance through SSH.
ssh_keys	Array of strings	List of SSH key pairs. You can set multiple key pairs to access an SSH instance at the same time.

Table 4-22 Image

Parameter	Type	Description
arch	String	Processor architecture supported by the image. Options: <ul style="list-style-type: none"> • X86_64: x86 architecture • AARCH64: Arm architecture
create_at	Long	Specifies the time (UTC ms) when the image is created.
description	String	Image description with a maximum of 512 characters
dev_services	Array of strings	Services supported by the image. Options: <ul style="list-style-type: none"> • NOTEBOOK: You can access the notebook instance using HTTPS. • SSH: You can remotely access the notebook instance from a local IDE through SSH.
id	String	ID of the image used for creating notebook instances. The ID is in Universally Unique Identifier (UUID) format. For details, see Querying Supported Images .
name	String	Image name, which contains a maximum of 512 characters, including lowercase letters, digits, hyphens (-), underscores (_), and periods (.)
namespace	String	Organization to which the image belongs. You can create and view the organization on the Organization Management page of the SWR console.

Parameter	Type	Description
origin	String	Image source, which defaults to CUSTOMIZE . Options: <ul style="list-style-type: none"> ● CUSTOMIZE: user-defined image ● IMAGE_SAVE: image saved using a development environment instance
resource_categories	Array of strings	Flavors supported by the image. Options: <ul style="list-style-type: none"> ● CPU ● GPU
service_type	String	Supported image types. Options: <ul style="list-style-type: none"> ● COMMON: common image ● INFERENCE: image used for inference ● TRAIN: image used for training ● DEV: image used for development and debugging ● UNKNOWN: image whose supported services are not specified
size	Long	Specifies the image size, in KB.
status	String	Image status. Options: <ul style="list-style-type: none"> ● INIT: The image is being initialized. ● CREATING: The image is being saved. In this case, the notebook instance is unavailable. ● CREATE_FAILED: Saving the image failed. ● ERROR: An error occurs. ● DELETED: The image has been deleted. ● ACTIVE: The image has been saved, which you can view on the SWR console and use to create notebook instances.
status_message	String	Build information during image saving
support_resource_categories	Array of strings	Flavors supported by the image. Options: <ul style="list-style-type: none"> ● CPU ● GPU
swr_path	String	SWR image address
tag	String	Image tag
type	String	Image type. Options: <ul style="list-style-type: none"> ● BUILD_IN: built-in system image ● DEDICATED: private image

Parameter	Type	Description
update_at	Long	Specifies the time (UTC ms) when the image was last updated.
visibility	String	Image visibility. Options: <ul style="list-style-type: none"> • PRIVATE: private image • PUBLIC: All users can perform read-only operations based on the image ID.
workspace_id	String	Workspace ID. If no workspaces are available, the default value is 0 .

Table 4-23 Lease

Parameter	Type	Description
create_at	Long	Time (UTC) when the instance is created, accurate to millisecond.
duration	Long	Instance running duration, which is calculated based on the instance creation time. If the instance creation time plus the duration is greater than the current time, the system automatically stops the instance.
enable	Boolean	Whether to enable auto stop of the instance.
type	String	Indicates the automatic stop type.
update_at	Long	Time (UTC) when the instance is last updated (excluding the keepalive heartbeat time), accurate to millisecond.

Table 4-24 Pool

Parameter	Type	Description
id	String	ID of a dedicated resource pool
name	String	Name of a dedicated resource pool

Table 4-25 VolumeRes

Parameter	Type	Description
capacity	Integer	Storage capacity. The default value is 5 GB for EVS and 50 GB for EFS. The maximum value is 4096 GB.

Parameter	Type	Description
category	String	Supported storage types. For details about the differences between the storage types, see Selecting Storage in DevEnviron . Options: <ul style="list-style-type: none"> • SFS: Scalable File Service • EVS
mount_path	String	Directory of the notebook instance to which OBS storage is mounted. Currently, the directory is /home/ma-user/work/ .
ownership	String	Owner to which the resource belongs. Options: <ul style="list-style-type: none"> • MANAGED: Resources are managed by service. • DEDICATED: Resources are managed by user account. This mode is supported only when the instance category is EFS.
status	String	EVS disk capacity expansion status, which is RESIZING during capacity expansion and does not affect the instance.

Example Requests

The following is an example of how to create a notebook instance whose flavor is **modelarts.vm.cpu.2u**, storage type is **evs**, ownership is **MANAGED**, and storage capacity is 50 GB.

```
{
  "name" : "notebooks_test",
  "feature" : "NOTEBOOK",
  "workspace_id" : "0",
  "description" : "api-test",
  "flavor" : "modelarts.vm.cpu.2u",
  "image_id" : "e1a07296-22a8-4f05-8bc8-e936c8e54090",
  "volume" : {
    "category" : "evs",
    "ownership" : "managed",
    "capacity" : 50
  }
}
```

Example Responses

Status code: 200

OK

```
{
  "description" : "api-test",
  "feature" : "NOTEBOOK",
  "flavor" : "modelarts.vm.cpu.2u",
  "id" : "f9937afa-4451-42db-a76b-72d624749f66",
  "image" : {
```

```

"description" : "description",
"id" : "e1a07296-22a8-4f05-8bc8-e936c8e54090",
"name" : "notebook2.0-mul-kernel-cpu-cp36",
"swr_path" : "swr.xxxx.com/atelier/notebook2.0-mul-kernel-cpu-cp36:3.3.2-release_v1",
>tag" : "3.3.2-release_v1",
"type" : "BUILD_IN"
},
"lease" : {
"create_at" : 1638841744515,
"duration" : 3600000,
"enable" : true,
"type" : "TIMING",
"update_at" : 1638841744515
},
"name" : "notebooks_test",
"status" : "CREATING",
"token" : "3eff13f2-3d70-5456-6dc7-e3f99f562022",
"workspace_id" : "0"
}

```

Status Codes

Status Code	Description
200	OK
201	Created
401	Unauthorized
403	Forbidden
404	Not Found

Error Codes

See [Error Codes](#).

4.3 Querying Details of a Notebook Instance

Function

This API is used to query details about a notebook instance, including its ID, name, flavor, image, status, and accessible URLs.

Constraints

None

URI

GET /v1/{project_id}/notebooks/{id}

Table 4-26 Path Parameters

Parameter	Mandatory	Type	Description
id	Yes	String	Notebook instance ID.
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .

Request Parameters

None

Response Parameters

Status code: 200

Table 4-27 Response body parameters

Parameter	Type	Description
action_progress	Array of JobProgress objects	Instance initialization progress.
description	String	Instance description
endpoints	Array of EndpointsRes objects	This section describes how to configure the local IDE (such as PyCharm and VS Code) or SSH client to remotely access a notebook instance through SSH.
fail_reason	String	Instance failure cause
flavor	String	Instance flavor
id	String	Instance ID.
image	Image object	Instance image
lease	Lease object	Countdown to automatic instance stop.
name	String	Instance name
pool	Pool object	Basic information about the dedicated pool, which is returned by the instance created in the dedicated pool.

Parameter	Type	Description
status	String	Instance status. Options: <ul style="list-style-type: none"> • INIT: The instance is being initialized. • CREATING: The instance is being created. • STARTING: The instance is starting. • STOPPING: The instance is being stopped. • DELETING: The instance is being deleted. • RUNNING: The instance is running. • STOPPED: The instance has been stopped. • SNAPSHOTTING: The image of the instance is being saved. • CREATE_FAILED: Creating the instance failed. • START_FAILED: Starting the instance failed. • DELETE_FAILED: Deleting the instance failed. • ERROR: An error occurred. • DELETED: The instance has been deleted. • FROZEN: The instance is frozen.
token	String	Token information used for notebook authentication.
url	String	URL for accessing the notebook instance
volume	VolumeRes object	Storage volume
workspace_id	String	Workspace ID. If no workspaces are available, the default value is 0 .
feature	String	Instance type. Options: <ul style="list-style-type: none"> • DEFAULT: free CodeLab instance. You can create only one. • NOTEBOOK: billed instance.

Table 4-28 JobProgress

Parameter	Type	Description
notebook_id	String	Instance ID.

Parameter	Type	Description
status	String	Job status in a specified step. Options: <ul style="list-style-type: none"> ● WAITING: The job is awaiting. ● PROCESSING: The job is being processed. ● FAILED: The job failed. ● COMPLETED: The job is complete.
step	Integer	Job step. Options: <ul style="list-style-type: none"> ● 1: Prepare storage. ● 2: Prepare compute resources. ● 3: Configure the network. ● 4: Initialize the instance.
step_description	String	Description of a step in a job.

Table 4-29 EndpointsRes

Parameter	Type	Description
allowed_access_ips	Array of strings	Whitelist of public IP addresses that are allowed to access the notebook instance through SSH. By default, all public IP addresses can access the notebook instance. If this parameter is specified, only the clients with the specified IP addresses can access the notebook instance.
dev_service	String	Supported service. Options: <ul style="list-style-type: none"> ● NOTEBOOK: You can access notebook instances using HTTPS. ● SSH: You can remotely access the notebook instance through SSH.
ssh_keys	Array of strings	List of SSH key pairs. You can set multiple key pairs to access an SSH instance at the same time.

Table 4-30 Image

Parameter	Type	Description
arch	String	Processor architecture supported by the image. Options: <ul style="list-style-type: none"> ● X86_64: x86 architecture ● AARCH64: Arm architecture

Parameter	Type	Description
create_at	Long	Specifies the time (UTC ms) when the image is created.
description	String	Image description with a maximum of 512 characters
dev_services	Array of strings	Services supported by the image. Options: <ul style="list-style-type: none"> • NOTEBOOK: You can access the notebook instance using HTTPS. • SSH: You can remotely access the notebook instance from a local IDE through SSH.
id	String	ID of the image used for creating notebook instances. The ID is in Universally Unique Identifier (UUID) format. For details, see Querying Supported Images .
name	String	Image name, which contains a maximum of 512 characters, including lowercase letters, digits, hyphens (-), underscores (_), and periods (.)
namespace	String	Organization to which the image belongs. You can create and view the organization on the Organization Management page of the SWR console.
origin	String	Image source, which defaults to CUSTOMIZE . Options: <ul style="list-style-type: none"> • CUSTOMIZE: user-defined image • IMAGE_SAVE: image saved using a development environment instance
resource_categories	Array of strings	Flavors supported by the image. Options: <ul style="list-style-type: none"> • CPU • GPU
service_type	String	Supported image types. Options: <ul style="list-style-type: none"> • COMMON: common image • INFERENCE: image used for inference • TRAIN: image used for training • DEV: image used for development and debugging • UNKNOWN: image whose supported services are not specified
size	Long	Specifies the image size, in KB.

Parameter	Type	Description
status	String	Image status. Options: <ul style="list-style-type: none"> • INIT: The image is being initialized. • CREATING: The image is being saved. In this case, the notebook instance is unavailable. • CREATE_FAILED: Saving the image failed. • ERROR: An error occurs. • DELETED: The image has been deleted. • ACTIVE: The image has been saved, which you can view on the SWR console and use to create notebook instances.
status_message	String	Build information during image saving
support_res_categories	Array of strings	Flavors supported by the image. Options: <ul style="list-style-type: none"> • CPU • GPU
swr_path	String	SWR image address
tag	String	Image tag
type	String	Image type. Options: <ul style="list-style-type: none"> • BUILD_IN: built-in system image • DEDICATED: private image
update_at	Long	Specifies the time (UTC ms) when the image was last updated.
visibility	String	Image visibility. Options: <ul style="list-style-type: none"> • PRIVATE: private image • PUBLIC: All users can perform read-only operations based on the image ID.
workspace_id	String	Workspace ID. If no workspaces are available, the default value is 0 .

Table 4-31 Lease

Parameter	Type	Description
create_at	Long	Time (UTC) when the instance is created, accurate to millisecond.

Parameter	Type	Description
duration	Long	Instance running duration, which is calculated based on the instance creation time. If the instance creation time plus the duration is greater than the current time, the system automatically stops the instance.
enable	Boolean	Whether to enable auto stop of the instance.
type	String	Indicates the automatic stop type.
update_at	Long	Time (UTC) when the instance is last updated (excluding the keepalive heartbeat time), accurate to millisecond.

Table 4-32 Pool

Parameter	Type	Description
id	String	ID of a dedicated resource pool
name	String	Name of a dedicated resource pool

Table 4-33 VolumeRes

Parameter	Type	Description
capacity	Integer	Storage capacity. The default value is 5 GB for EVS and 50 GB for EFS. The maximum value is 4096 GB.
category	String	Supported storage types. For details about the differences between the storage types, see Selecting Storage in DevEnviron . Options: <ul style="list-style-type: none"> • SFS: Scalable File Service • EVS
mount_path	String	Directory of the notebook instance to which OBS storage is mounted. Currently, the directory is <code>/home/ma-user/work/</code> .
ownership	String	Owner to which the resource belongs. Options: <ul style="list-style-type: none"> • MANAGED: Resources are managed by service. • DEDICATED: Resources are managed by user account. This mode is supported only when the instance category is EFS.

Parameter	Type	Description
status	String	EVS disk capacity expansion status, which is RESIZING during capacity expansion and does not affect the instance.

Example Requests

GET https://{endpoint}/v1/{project_id}/notebooks/{id}

Example Responses

Status code: 200

OK

```
{
  "description": "api-test",
  "feature": "NOTEBOOK",
  "flavor": "modelarts.vm.cpu.2u",
  "id": "f9937afa-631e-4a8f-a8f7-3b6c800585f0",
  "image": {
    "id": "e1a07296-22a8-4f05-8bc8-e936c8e54090",
    "name": "notebook2.0-mul-kernel-cpu-cp36",
    "swr_path": "swr.xxxx.com/atelier/notebook2.0-mul-kernel-cpu-cp36:3.3.1.B007_V2",
    "type": "BUILD_IN"
  },
  "lease": {
    "create_at": 1638778344300,
    "duration": 3600000,
    "enable": true,
    "update_at": 1638778344300
  },
  "name": "notebooks_test",
  "status": "CREATING",
  "token": "7b22482a-a4cb-4b46-e3b3-6a793a47967e",
  "url": "https://authoring-modelarts-xxxx.xxxx.com/f9937afa-631e-4a8f-a8f7-3b6c800585f0/lab",
  "workspace_id": "0"
}
```

Status Codes

Status Code	Description
200	OK
401	Unauthorized
403	Forbidden
404	Not Found

Error Codes

See [Error Codes](#).

4.4 Updating a Notebook Instance

Function

This API is used to update a development environment instance, including the name, description, specifications, and image ID. This API can be used only when the notebook instance is stopped.

Constraints

None

URI

PUT /v1/{project_id}/notebooks/{id}

Table 4-34 Path Parameters

Parameter	Mandatory	Type	Description
id	Yes	String	Notebook instance ID.
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .

Request Parameters

Table 4-35 Request body parameters

Parameter	Mandatory	Type	Description
description	No	String	Instance description. The value can contain a maximum of 512 characters and cannot contain the following special characters: &<>"/.
endpoints	No	Array of EndpointsReq objects	This section describes how to configure the local IDE (such as PyCharm and VS Code) or SSH client to remotely access a notebook instance through SSH.
flavor	No	String	Instance flavor. For details, see Querying Flavors Available for a Notebook Instance .

Parameter	Mandatory	Type	Description
image_id	No	String	Image ID. For details, see Querying Supported Images .
name	No	String	The instance name can be updated. The name contains a maximum of 128 characters, The value can contain letters, digits, hyphens (-), and underscores (_).
storage_new_size	No	Integer	EVS instance capacity that can be expanded, in GB. The maximum value is 4,096 GB.
hooks	No	CustomHooks object	The instance startup script can be updated.

Table 4-36 EndpointsReq

Parameter	Mandatory	Type	Description
allowed_access_ips	No	Array of strings	Public IP addresses that can remotely access the notebook instance. A maximum of five public IP addresses are supported.
dev_service	No	String	Supported service. Options: <ul style="list-style-type: none"> • NOTEBOOK: You can access notebook instances using HTTPS. • SSH: You can remotely access the notebook instance through SSH.
ssh_keys	No	Array of strings	Name of the SSH key pair, which can be created and viewed on the Key Pair page of the Elastic Cloud Server (ECS) console.

Table 4-37 CustomHooks

Parameter	Mandatory	Type	Description
container_hooks	No	ContainerHooks object	Container-related hook configuration.

Table 4-38 ContainerHooks

Parameter	Mandatory	Type	Description
post_start	No	Config object	Configure the script after the service is started.
pre_start	No	Config object	Configure the script before starting the service.

Table 4-39 Config

Parameter	Mandatory	Type	Description
script	No	String	User-defined script content (Base64 encoding) or absolute script path.
type	No	String	Script type. The options are as follows: <ul style="list-style-type: none"> The script content (Base64 encoding) must be specified in COMMAND script. Default value of SCRIPT. The script path must be specified in the script.

Response Parameters

Status code: 200

Table 4-40 Response body parameters

Parameter	Type	Description
action_progress	Array of JobProgress objects	Instance initialization progress.
description	String	Instance description
endpoints	Array of EndpointsRes objects	This section describes how to configure the local IDE (such as PyCharm and VS Code) or SSH client to remotely access a notebook instance through SSH.
fail_reason	String	Instance failure cause
flavor	String	Instance flavor

Parameter	Type	Description
id	String	Instance ID.
image	Image object	Instance image
lease	Lease object	Countdown to automatic instance stop.
name	String	Instance name
pool	Pool object	Basic information about the dedicated pool, which is returned by the instance created in the dedicated pool.
status	String	Instance status. Options: <ul style="list-style-type: none"> • INIT: The instance is being initialized. • CREATING: The instance is being created. • STARTING: The instance is starting. • STOPPING: The instance is being stopped. • DELETING: The instance is being deleted. • RUNNING: The instance is running. • STOPPED: The instance has been stopped. • SNAPSHOTTING: The image of the instance is being saved. • CREATE_FAILED: Creating the instance failed. • START_FAILED: Starting the instance failed. • DELETE_FAILED: Deleting the instance failed. • ERROR: An error occurred. • DELETED: The instance has been deleted. • FROZEN: The instance is frozen.
token	String	Token information used for notebook authentication.
url	String	URL for accessing the notebook instance
volume	VolumeRes object	Storage volume
workspace_id	String	Workspace ID. If no workspaces are available, the default value is 0 .
feature	String	Instance type. Options: <ul style="list-style-type: none"> • DEFAULT: free CodeLab instance. You can create only one. • NOTEBOOK: billed instance.

Table 4-41 JobProgress

Parameter	Type	Description
notebook_id	String	Instance ID.
status	String	Job status in a specified step. Options: <ul style="list-style-type: none"> ● WAITING: The job is awaiting. ● PROCESSING: The job is being processed. ● FAILED: The job failed. ● COMPLETED: The job is complete.
step	Integer	Job step. Options: <ul style="list-style-type: none"> ● 1: Prepare storage. ● 2: Prepare compute resources. ● 3: Configure the network. ● 4: Initialize the instance.
step_description	String	Description of a step in a job.

Table 4-42 EndpointsRes

Parameter	Type	Description
allowed_access_ips	Array of strings	Whitelist of public IP addresses that are allowed to access the notebook instance through SSH. By default, all public IP addresses can access the notebook instance. If this parameter is specified, only the clients with the specified IP addresses can access the notebook instance.
dev_service	String	Supported service. Options: <ul style="list-style-type: none"> ● NOTEBOOK: You can access notebook instances using HTTPS. ● SSH: You can remotely access the notebook instance through SSH.
ssh_keys	Array of strings	List of SSH key pairs. You can set multiple key pairs to access an SSH instance at the same time.

Table 4-43 Image

Parameter	Type	Description
arch	String	Processor architecture supported by the image. Options: <ul style="list-style-type: none"> ● X86_64: x86 architecture ● AARCH64: Arm architecture
create_at	Long	Specifies the time (UTC ms) when the image is created.
description	String	Image description with a maximum of 512 characters
dev_services	Array of strings	Services supported by the image. Options: <ul style="list-style-type: none"> ● NOTEBOOK: You can access the notebook instance using HTTPS. ● SSH: You can remotely access the notebook instance from a local IDE through SSH.
id	String	ID of the image used for creating notebook instances. The ID is in Universally Unique Identifier (UUID) format. For details, see Querying Supported Images .
name	String	Image name, which contains a maximum of 512 characters, including lowercase letters, digits, hyphens (-), underscores (_), and periods (.)
namespace	String	Organization to which the image belongs. You can create and view the organization on the Organization Management page of the SWR console.
origin	String	Image source, which defaults to CUSTOMIZE . Options: <ul style="list-style-type: none"> ● CUSTOMIZE: user-defined image ● IMAGE_SAVE: image saved using a development environment instance
resource_categories	Array of strings	Flavors supported by the image. Options: <ul style="list-style-type: none"> ● CPU ● GPU

Parameter	Type	Description
service_type	String	Supported image types. Options: <ul style="list-style-type: none"> ● COMMON: common image ● INFERENCE: image used for inference ● TRAIN: image used for training ● DEV: image used for development and debugging ● UNKNOWN: image whose supported services are not specified
size	Long	Specifies the image size, in KB.
status	String	Image status. Options: <ul style="list-style-type: none"> ● INIT: The image is being initialized. ● CREATING: The image is being saved. In this case, the notebook instance is unavailable. ● CREATE_FAILED: Saving the image failed. ● ERROR: An error occurs. ● DELETED: The image has been deleted. ● ACTIVE: The image has been saved, which you can view on the SWR console and use to create notebook instances.
status_message	String	Build information during image saving
support_resource_categories	Array of strings	Flavors supported by the image. Options: <ul style="list-style-type: none"> ● CPU ● GPU
swr_path	String	SWR image address
tag	String	Image tag
type	String	Image type. Options: <ul style="list-style-type: none"> ● BUILD_IN: built-in system image ● DEDICATED: private image
update_at	Long	Specifies the time (UTC ms) when the image was last updated.
visibility	String	Image visibility. Options: <ul style="list-style-type: none"> ● PRIVATE: private image ● PUBLIC: All users can perform read-only operations based on the image ID.
workspace_id	String	Workspace ID. If no workspaces are available, the default value is 0 .

Table 4-44 Lease

Parameter	Type	Description
create_at	Long	Time (UTC) when the instance is created, accurate to millisecond.
duration	Long	Instance running duration, which is calculated based on the instance creation time. If the instance creation time plus the duration is greater than the current time, the system automatically stops the instance.
enable	Boolean	Whether to enable auto stop of the instance.
type	String	Indicates the automatic stop type.
update_at	Long	Time (UTC) when the instance is last updated (excluding the keepalive heartbeat time), accurate to millisecond.

Table 4-45 Pool

Parameter	Type	Description
id	String	ID of a dedicated resource pool
name	String	Name of a dedicated resource pool

Table 4-46 VolumeRes

Parameter	Type	Description
capacity	Integer	Storage capacity. The default value is 5 GB for EVS and 50 GB for EFS. The maximum value is 4096 GB.
category	String	Supported storage types. For details about the differences between the storage types, see Selecting Storage in DevEnviron . Options: <ul style="list-style-type: none"> • SFS: Scalable File Service • EVS
mount_path	String	Directory of the notebook instance to which OBS storage is mounted. Currently, the directory is <code>/home/ma-user/work/</code> .

Parameter	Type	Description
ownership	String	Owner to which the resource belongs. Options: <ul style="list-style-type: none"> • MANAGED: Resources are managed by service. • DEDICATED: Resources are managed by user account. This mode is supported only when the instance category is EFS.
status	String	EVS disk capacity expansion status, which is RESIZING during capacity expansion and does not affect the instance.

Example Requests

```
{
  "description" : "update"
}
```

Example Responses

Status code: 200

OK

```
{
  "description" : "test",
  "flavor" : "modelarts.vm.cpu.8u",
  "name" : "notebook-1111",
  "endpoints" : [ {
    "allowed_access_ips" : [ "1.1.1.1" ]
  } ]
}
```

Status Codes

Status Code	Description
200	OK
201	Created
401	Unauthorized
403	Forbidden
404	Not Found

Error Codes

See [Error Codes](#).

4.5 Deleting a Notebook Instance

Function

This API is used to delete the container and all storage resources of a notebook instance.

Constraints

None

URI

DELETE /v1/{project_id}/notebooks/{id}

Table 4-47 Path Parameters

Parameter	Mandatory	Type	Description
id	Yes	String	Notebook instance ID.
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .

Request Parameters

None

Response Parameters

Status code: 200

Table 4-48 Response body parameters

Parameter	Type	Description
action_progress	Array of JobProgress objects	Instance initialization progress.
description	String	Instance description
endpoints	Array of EndpointsRes objects	This section describes how to configure the local IDE (such as PyCharm and VS Code) or SSH client to remotely access a notebook instance through SSH.
fail_reason	String	Instance failure cause

Parameter	Type	Description
flavor	String	Instance flavor
id	String	Instance ID.
image	Image object	Instance image
lease	Lease object	Countdown to automatic instance stop.
name	String	Instance name
pool	Pool object	Basic information about the dedicated pool, which is returned by the instance created in the dedicated pool.
status	String	Instance status. Options: <ul style="list-style-type: none"> • INIT: The instance is being initialized. • CREATING: The instance is being created. • STARTING: The instance is starting. • STOPPING: The instance is being stopped. • DELETING: The instance is being deleted. • RUNNING: The instance is running. • STOPPED: The instance has been stopped. • SNAPSHOTTING: The image of the instance is being saved. • CREATE_FAILED: Creating the instance failed. • START_FAILED: Starting the instance failed. • DELETE_FAILED: Deleting the instance failed. • ERROR: An error occurred. • DELETED: The instance has been deleted. • FROZEN: The instance is frozen.
token	String	Token information used for notebook authentication.
url	String	URL for accessing the notebook instance
volume	VolumeRes object	Storage volume
workspace_id	String	Workspace ID. If no workspaces are available, the default value is 0 .
feature	String	Instance type. Options: <ul style="list-style-type: none"> • DEFAULT: free CodeLab instance. You can create only one. • NOTEBOOK: billed instance.

Table 4-49 JobProgress

Parameter	Type	Description
notebook_id	String	Instance ID.
status	String	Job status in a specified step. Options: <ul style="list-style-type: none"> ● WAITING: The job is awaiting. ● PROCESSING: The job is being processed. ● FAILED: The job failed. ● COMPLETED: The job is complete.
step	Integer	Job step. Options: <ul style="list-style-type: none"> ● 1: Prepare storage. ● 2: Prepare compute resources. ● 3: Configure the network. ● 4: Initialize the instance.
step_description	String	Description of a step in a job.

Table 4-50 EndpointsRes

Parameter	Type	Description
allowed_access_ips	Array of strings	Whitelist of public IP addresses that are allowed to access the notebook instance through SSH. By default, all public IP addresses can access the notebook instance. If this parameter is specified, only the clients with the specified IP addresses can access the notebook instance.
dev_service	String	Supported service. Options: <ul style="list-style-type: none"> ● NOTEBOOK: You can access notebook instances using HTTPS. ● SSH: You can remotely access the notebook instance through SSH.
ssh_keys	Array of strings	List of SSH key pairs. You can set multiple key pairs to access an SSH instance at the same time.

Table 4-51 Image

Parameter	Type	Description
arch	String	Processor architecture supported by the image. Options: <ul style="list-style-type: none"> ● X86_64: x86 architecture ● AARCH64: Arm architecture
create_at	Long	Specifies the time (UTC ms) when the image is created.
description	String	Image description with a maximum of 512 characters
dev_services	Array of strings	Services supported by the image. Options: <ul style="list-style-type: none"> ● NOTEBOOK: You can access the notebook instance using HTTPS. ● SSH: You can remotely access the notebook instance from a local IDE through SSH.
id	String	ID of the image used for creating notebook instances. The ID is in Universally Unique Identifier (UUID) format. For details, see Querying Supported Images .
name	String	Image name, which contains a maximum of 512 characters, including lowercase letters, digits, hyphens (-), underscores (_), and periods (.)
namespace	String	Organization to which the image belongs. You can create and view the organization on the Organization Management page of the SWR console.
origin	String	Image source, which defaults to CUSTOMIZE . Options: <ul style="list-style-type: none"> ● CUSTOMIZE: user-defined image ● IMAGE_SAVE: image saved using a development environment instance
resource_categories	Array of strings	Flavors supported by the image. Options: <ul style="list-style-type: none"> ● CPU ● GPU

Parameter	Type	Description
service_type	String	Supported image types. Options: <ul style="list-style-type: none"> ● COMMON: common image ● INFERENCE: image used for inference ● TRAIN: image used for training ● DEV: image used for development and debugging ● UNKNOWN: image whose supported services are not specified
size	Long	Specifies the image size, in KB.
status	String	Image status. Options: <ul style="list-style-type: none"> ● INIT: The image is being initialized. ● CREATING: The image is being saved. In this case, the notebook instance is unavailable. ● CREATE_FAILED: Saving the image failed. ● ERROR: An error occurs. ● DELETED: The image has been deleted. ● ACTIVE: The image has been saved, which you can view on the SWR console and use to create notebook instances.
status_message	String	Build information during image saving
support_resource_categories	Array of strings	Flavors supported by the image. Options: <ul style="list-style-type: none"> ● CPU ● GPU
swr_path	String	SWR image address
tag	String	Image tag
type	String	Image type. Options: <ul style="list-style-type: none"> ● BUILD_IN: built-in system image ● DEDICATED: private image
update_at	Long	Specifies the time (UTC ms) when the image was last updated.
visibility	String	Image visibility. Options: <ul style="list-style-type: none"> ● PRIVATE: private image ● PUBLIC: All users can perform read-only operations based on the image ID.
workspace_id	String	Workspace ID. If no workspaces are available, the default value is 0 .

Table 4-52 Lease

Parameter	Type	Description
create_at	Long	Time (UTC) when the instance is created, accurate to millisecond.
duration	Long	Instance running duration, which is calculated based on the instance creation time. If the instance creation time plus the duration is greater than the current time, the system automatically stops the instance.
enable	Boolean	Whether to enable auto stop of the instance.
type	String	Indicates the automatic stop type.
update_at	Long	Time (UTC) when the instance is last updated (excluding the keepalive heartbeat time), accurate to millisecond.

Table 4-53 Pool

Parameter	Type	Description
id	String	ID of a dedicated resource pool
name	String	Name of a dedicated resource pool

Table 4-54 VolumeRes

Parameter	Type	Description
capacity	Integer	Storage capacity. The default value is 5 GB for EVS and 50 GB for EFS. The maximum value is 4096 GB.
category	String	Supported storage types. For details about the differences between the storage types, see Selecting Storage in DevEnviron . Options: <ul style="list-style-type: none"> • SFS: Scalable File Service • EVS
mount_path	String	Directory of the notebook instance to which OBS storage is mounted. Currently, the directory is <code>/home/ma-user/work/</code> .

Parameter	Type	Description
ownership	String	Owner to which the resource belongs. Options: <ul style="list-style-type: none"> • MANAGED: Resources are managed by service. • DEDICATED: Resources are managed by user account. This mode is supported only when the instance category is EFS.
status	String	EVS disk capacity expansion status, which is RESIZING during capacity expansion and does not affect the instance.

Example Requests

DELETE https://{endpoint}/v1/{project_id}/notebooks/{id}

Example Responses

Status code: 200

OK

```
{
  "create_at" : 1638841805440,
  "description" : "update",
  "feature" : "DEFAULT",
  "flavor" : "modelarts.vm.cpu.free",
  "id" : "f9937afa-ca78-45b6-bc12-7ecf42553c48",
  "image" : {
    "description" : "description",
    "id" : "e1a07296-22a8-4f05-8bc8-e936c8e54090",
    "name" : "notebook2.0-mul-kernel-cpu-cp36",
    "swr_path" : "swr.xxxx.com/atelier/notebook2.0-mul-kernel-cpu-cp36:3.3.2-release_v1",
    "tag" : "3.3.2-release_v1",
    "type" : "BUILD_IN"
  },
  "lease" : {
    "create_at" : 1638841805439,
    "duration" : 3600000,
    "enable" : true,
    "update_at" : 1638841805439
  },
  "name" : "notebook_5ee4bf0e",
  "status" : "DELETING",
  "token" : "58ba50c6-e8ff-245c-4840-49e51aa70737",
  "update_at" : 1638842504178,
  "workspace_id" : "0"
}
```

Status Codes

Status Code	Description
200	OK
204	No Content

Status Code	Description
401	Unauthorized
403	Forbidden

Error Codes

See [Error Codes](#).

4.6 Saving a Running Instance as a Container Image

Function

A running instance can be saved as a container image. In the saved image, the installed dependency package (pip package) is not lost. In the VS Code remote development scenario, the plug-ins installed on the server are not lost.

Constraints

None

URI

POST /v1/{project_id}/notebooks/{id}/create-image

Table 4-55 Path Parameters

Parameter	Mandatory	Type	Description
id	Yes	String	Notebook instance ID.
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .

Request Parameters

Table 4-56 Request body parameters

Parameter	Mandatory	Type	Description
description	No	String	Image description with a maximum of 512 characters

Parameter	Mandatory	Type	Description
name	No	String	Image name, which contains a maximum of 512 characters, including lowercase letters, digits, hyphens (-), underscores (_), and periods (.)
namespace	No	String	Organization to which the image belongs. You can create and view the organization on the Organization Management page of the SWR console.
tag	No	String	Image tag, which contains a maximum of 64 characters, including letters, digits, hyphens (-), underscores (_), and period (.)
workspace_id	No	String	Workspace ID. If no workspaces are available, the default value is 0 .

Response Parameters

Status code: 200

Table 4-57 Response body parameters

Parameter	Type	Description
arch	String	Processor architecture supported by the image. Options: <ul style="list-style-type: none"> ● X86_64: x86 architecture ● AARCH64: Arm architecture
create_at	Long	Specifies the time (UTC ms) when the image is created.
description	String	Image description with a maximum of 512 characters
dev_services	Array of strings	Services supported by the image. Options: <ul style="list-style-type: none"> ● NOTEBOOK: You can access the notebook instance using HTTPS. ● SSH: You can remotely access the notebook instance from a local IDE through SSH.

Parameter	Type	Description
id	String	ID of the image used for creating notebook instances. The ID is in Universally Unique Identifier (UUID) format. For details, see Querying Supported Images .
name	String	Image name, which contains a maximum of 512 characters, including lowercase letters, digits, hyphens (-), underscores (_), and periods (.)
namespace	String	Organization to which the image belongs. You can create and view the organization on the Organization Management page of the SWR console.
origin	String	Image source, which defaults to CUSTOMIZE . Options: <ul style="list-style-type: none"> • CUSTOMIZE: user-defined image • IMAGE_SAVE: image saved using a development environment instance
resource_categories	Array of strings	Flavors supported by the image. Options: <ul style="list-style-type: none"> • CPU • GPU
service_type	String	Supported image types. Options: <ul style="list-style-type: none"> • COMMON: common image • INFERENCE: image used for inference • TRAIN: image used for training • DEV: image used for development and debugging • UNKNOWN: image whose supported services are not specified
size	Long	Specifies the image size, in KB.
status	String	Image status. Options: <ul style="list-style-type: none"> • INIT: The image is being initialized. • CREATING: The image is being saved. In this case, the notebook instance is unavailable. • CREATE_FAILED: Saving the image failed. • ERROR: An error occurs. • DELETED: The image has been deleted. • ACTIVE: The image has been saved, which you can view on the SWR console and use to create notebook instances.

Parameter	Type	Description
status_message	String	Build information during image saving
support_resource_categories	Array of strings	Flavors supported by the image. Options: <ul style="list-style-type: none"> • CPU • GPU
swr_path	String	SWR image address
tag	String	Image tag
type	String	Image type. Options: <ul style="list-style-type: none"> • BUILD_IN: built-in system image • DEDICATED: private image
update_at	Long	Specifies the time (UTC ms) when the image was last updated.
visibility	String	Image visibility. Options: <ul style="list-style-type: none"> • PRIVATE: private image • PUBLIC: All users can perform read-only operations based on the image ID.
workspace_id	String	Workspace ID. If no workspaces are available, the default value is 0 .

Example Requests

The following is an example of how to save a running instance as a container image whose image name is **pytorch1_4** and organization is **atelier-auto**.

```
{
  "name": "pytorch1_4",
  "namespace": "atelier-auto",
  "tag": "20221223",
  "description": "save from notebook-x21d",
  "workspace_id": "0"
}
```

Example Responses

Status code: 200

OK

```
{
  "arch": "x86_64",
  "create_at": 1671786468811,
  "description": "notebook2.0 20200816",
  "dev_services": [ "SSH", "NOTEBOOK" ],
  "id": "4e0d1854-63e5-4517-b683-a0ee97a692a1",
  "name": "pytorch1_4",
  "namespace": "atelier-auto",
  "origin": "IMAGE_SAVE",
  "resource_categories": [ "CPU", "GPU" ],
```

```

"service_type" : "TRAIN",
"status" : "INIT",
"swr_path" : "swr.xxxxx.com/atelier-auto/pytorch1_4:20221223",
"tag" : "20221223",
"type" : "DEDICATED",
"update_at" : 1671786468811,
"visibility" : "PRIVATE",
"workspace_id" : "0"
}

```

Status Codes

Status Code	Description
200	OK
201	Created
401	Unauthorized
403	Forbidden
404	Not Found

Error Codes

See [Error Codes](#).

4.7 Obtaining the Available Flavors

Function

Obtain the available flavors.

Constraints

None

URI

GET /v1/{project_id}/notebooks/flavors

Table 4-58 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .

Table 4-59 Query Parameters

Parameter	Mandatory	Type	Description
category	No	String	Processor type. Options: <ul style="list-style-type: none"> • CPU • GPU
limit	No	Integer	Number of records on each page. (No limit by default)
offset	No	Integer	Start offset of the records on each page. The default value is 0 .
type	No	String	Cluster type. <ul style="list-style-type: none"> • MANAGED: public cluster • DEDICATED: dedicated cluster
sort_dir	No	String	Sorting mode. The options are ASC (ascending order) and DESC (descending order). The default value is DESC .
sort_key	No	String	Sorting fields. Separate multiple fields with commas (,).

Request Parameters

None

Response Parameters

Status code: 200

Table 4-60 Response body parameters

Parameter	Type	Description
current	Integer	Current page
data	Array of NotebookFlavor objects	Pagination data
flavors	Array of NotebookFlavor objects	List of specifications that can be switched.
pages	Integer	Total number of pages

Parameter	Type	Description
size	Integer	Number of records on each page
total	Long	Total number of records

Table 4-61 NotebookFlavor

Parameter	Type	Description
arch	String	Architecture type. Options: <ul style="list-style-type: none"> • X86_64 • AARCH64
ascend	AscendInfo object	NPU information
billing	BillingInfo object	CDR information
category	String	Processor type. Options: <ul style="list-style-type: none"> • CPU • GPU
description	String	Specification description
feature	String	Specification type. The options are as follows: <ul style="list-style-type: none"> • DEFAULT: CodeLab specification. • NOTEBOOK: Notebook specifications.
free	Boolean	Specifies whether the flavor is free of charge.
gpu	GPUInfo object	GPU information
id	String	Flavor ID
memory	Long	Memory size
name	String	Flavor name
sold_out	Boolean	Whether resources are sufficient. <ul style="list-style-type: none"> • true: Resources are insufficient. • false: Resources are sufficient.
storages	Array of strings	Storage type. Options: <ul style="list-style-type: none"> • EFS • EVS
vcpus	Integer	Number of vCPUs

Table 4-62 AscendInfo

Parameter	Type	Description
npu	Integer	Number of NPUs
npu_memory	String	NPU memory
type	String	NPU type

Table 4-63 BillingInfo

Parameter	Type	Description
code	String	Billing code
unit_num	Integer	Billing unit

Table 4-64 GPUInfo

Parameter	Type	Description
gpu	Integer	Number of GPUs
gpu_memory	String	GPU memory
type	String	GPU type

Example Requests

GET https://{endpoint}/v1/{project_id}/notebooks/flavors

Example Responses

Status code: 200

OK

```
{
  "current" : 2,
  "data" : [ {
    "arch" : "x86_64",
    "billing" : {
      "code" : "modelarts.bm.gpu.v100NV32",
      "unit_num" : 4
    },
    "category" : "GPU",
    "description" : "Four NVIDIA V100 GPUs, each with 32GB of memory, ideal for deep learning algorithm training and debugging.",
    "feature" : "NOTEBOOK",
    "free" : false,
    "gpu" : {
      "gpu" : 4,
      "gpu_memory" : "128",
      "type" : "nvidia-v100-pcie32"
    }
  }
]
```

```

    "id" : "modelarts.bm.gpu.4v100NV32.nosdi",
    "memory" : 134217728,
    "name" : "GPU: 4*V100(128GB)|CPU: 32vCPUs 256GB",
    "sold_out" : true,
    "storages" : [ "EFS", "OBSFS", "OBS", "EVS" ],
    "vcpus" : 32
  }, {
    "arch" : "x86_64",
    "billing" : {
      "code" : "modelarts.bm.gpu.v100NV32.eco",
      "unit_num" : 2
    },
    "category" : "GPU",
    "description" : "Two Shared NVIDIA V100 GPU with 64GB of memory, ideal for ML development",
    "feature" : "NOTEBOOK",
    "free" : false,
    "gpu" : {
      "gpu" : 0.5,
      "gpu_memory" : "64",
      "type" : "nvidia-v100-nv32"
    },
    "id" : "modelarts.bm.gpu.2v100NV32.share",
    "memory" : 67108864,
    "name" : "vGPU: 2*V100(32GB)|CPU: 8vCPUs 64GB",
    "sold_out" : false,
    "storages" : [ "EFS", "OBSFS", "OBS", "EVS" ],
    "vcpus" : 8
  } ],
  "pages" : 26,
  "size" : 2,
  "total" : 52
}

```

Status Codes

Status Code	Description
200	OK
401	Unauthorized
403	Forbidden
404	Not Found

Error Codes

See [Error Codes](#).

4.8 Querying Flavors Available for a Notebook Instance

Function

This API is used to query the flavors available for a notebook instance.

Constraints

None

URI

GET /v1/{project_id}/notebooks/{id}/flavors

Table 4-65 Path Parameters

Parameter	Mandatory	Type	Description
id	Yes	String	Notebook instance ID.
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .

Table 4-66 Query Parameters

Parameter	Mandatory	Type	Description
limit	No	Integer	Number of records on each page. (No limit by default)
offset	No	Integer	Start offset of the records on each page

Request Parameters

None

Response Parameters

Status code: 200

Table 4-67 Response body parameters

Parameter	Type	Description
current	Integer	Current page
data	Array of NotebookFlavor objects	Pagination data
flavors	Array of NotebookFlavor objects	List of specifications that can be switched.
pages	Integer	Total number of pages
size	Integer	Number of records on each page
total	Long	Total number of records

Table 4-68 NotebookFlavor

Parameter	Type	Description
arch	String	Architecture type. Options: <ul style="list-style-type: none"> • X86_64 • AARCH64
ascend	AscendInfo object	NPU information
billing	BillingInfo object	CDR information
category	String	Processor type. Options: <ul style="list-style-type: none"> • CPU • GPU
description	String	Specification description
feature	String	Specification type. The options are as follows: <ul style="list-style-type: none"> • DEFAULT: CodeLab specification. • NOTEBOOK: Notebook specifications.
free	Boolean	Specifies whether the flavor is free of charge.
gpu	GPUInfo object	GPU information
id	String	Flavor ID
memory	Long	Memory size
name	String	Flavor name
sold_out	Boolean	Whether resources are sufficient. <ul style="list-style-type: none"> • true: Resources are insufficient. • false: Resources are sufficient.
storages	Array of strings	Storage type. Options: <ul style="list-style-type: none"> • EFS • EVS
vcpus	Integer	Number of vCPUs

Table 4-69 AscendInfo

Parameter	Type	Description
npu	Integer	Number of NPUs
npu_memory	String	NPU memory

Parameter	Type	Description
type	String	NPU type

Table 4-70 BillingInfo

Parameter	Type	Description
code	String	Billing code
unit_num	Integer	Billing unit

Table 4-71 GPUInfo

Parameter	Type	Description
gpu	Integer	Number of GPUs
gpu_memory	String	GPU memory
type	String	GPU type

Example Requests

GET https://{endpoint}/v1/{project_id}/notebooks/{id}/flavors

Example Responses

Status code: 200

OK

```
{
  "current" : 1,
  "data" : [ {
    "arch" : "aarch64",
    "ascend" : {
      "npu_memory" : "32",
      "npu" : 2,
      "type" : "ascend-snt9"
    },
    "billing" : {
      "code" : "modelarts.kat1.xlarge",
      "unit_num" : 2
    },
    "category" : "ASCEND",
    "description" : "The Ascend specification is suitable for deep learning code running and debugging",
    "feature" : "NOTEBOOK",
    "free" : false,
    "id" : "modelarts.bm.snt9.xlarge.2",
    "memory" : 201326592,
    "name" : "Ascend: 2*Ascend snt9|CPU: 48vCPUs 192GB",
    "sold_out" : false,
    "storages" : [ "EFS" ],
    "vcpus" : 48
  }, {
```

```

"arch" : "aarch64",
"ascend" : {
  "npu_memory" : "128",
  "npu" : 8,
  "type" : "ascend-snt9"
},
"billing" : {
  "code" : "modelarts.kat1.8xlarge",
  "unit_num" : 1
},
"category" : "ASCEND",
"description" : "The Ascend specification is suitable for deep learning code running and debugging",
"feature" : "NOTEBOOK",
"free" : false,
"id" : "modelarts.bm.snt9.xlarge.8",
"memory" : 805306368,
"name" : "Ascend: 8*Ascend snt9|CPU: 192vCPUs 768GB",
"sold_out" : false,
"storages" : [ "EFS" ],
"vcpus" : 192
}],
"flavors" : [ {
  "arch" : "aarch64",
  "ascend" : {
    "npu_memory" : "32",
    "npu" : 2,
    "type" : "ascend-snt9"
  },
  "billing" : {
    "code" : "modelarts.kat1.xlarge",
    "unit_num" : 2
  },
  "category" : "ASCEND",
  "description" : "The Ascend specification is suitable for deep learning code running and debugging",
  "feature" : "NOTEBOOK",
  "free" : false,
  "id" : "modelarts.bm.snt9.xlarge.2",
  "memory" : 201326592,
  "name" : "Ascend: 2*Ascend snt9|CPU: 48vCPUs 192GB",
  "sold_out" : false,
  "storages" : [ "EFS" ],
  "vcpus" : 48
}, {
  "arch" : "aarch64",
  "ascend" : {
    "npu_memory" : "128",
    "npu" : 8,
    "type" : "ascend-snt9"
  },
  "billing" : {
    "code" : "modelarts.kat1.8xlarge",
    "unit_num" : 1
  },
  "category" : "ASCEND",
  "description" : "The Ascend specification is suitable for deep learning code running and debugging",
  "feature" : "NOTEBOOK",
  "free" : false,
  "id" : "modelarts.bm.snt9.xlarge.8",
  "memory" : 805306368,
  "name" : "Ascend: 8*Ascend snt9|CPU: 192vCPUs 768GB",
  "sold_out" : false,
  "storages" : [ "EFS" ],
  "vcpus" : 192
}],
"pages" : 1,
"size" : 2,
"total" : 2
}

```

Status Codes

Status Code	Description
200	OK
401	Unauthorized
403	Forbidden
404	Not Found

Error Codes

See [Error Codes](#).

4.9 Querying the Available Duration of a Running Notebook Instance

Function

This API is used to query the available duration of a running notebook instance.

Constraints

None

URI

GET /v1/{project_id}/notebooks/{id}/lease

Table 4-72 Path Parameters

Parameter	Mandatory	Type	Description
id	Yes	String	Notebook instance ID.
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .

Request Parameters

None

Response Parameters

Status code: 200

Table 4-73 Response body parameters

Parameter	Type	Description
create_at	Long	Time (UTC) when the instance is created, accurate to millisecond.
duration	Long	Instance running duration, which is calculated based on the instance creation time. If the instance creation time plus the duration is greater than the current time, the system automatically stops the instance.
enable	Boolean	Whether to enable auto stop of the instance.
type	String	Indicates the automatic stop type.
update_at	Long	Time (UTC) when the instance is last updated (excluding the keepalive heartbeat time), accurate to millisecond.

Example Requests

GET https://{endpoint}/v1/{project_id}/notebooks/{id}/lease

Example Responses

Status code: 200

OK

```
{
  "create_at" : 1638841744515,
  "duration" : 3600000,
  "enable" : true,
  "type" : "TIMING",
  "update_at" : 1638842905925
}
```

Status Codes

Status Code	Description
200	OK
401	Unauthorized
403	Forbidden
404	Not Found

Error Codes

See [Error Codes](#).

4.10 Prolonging a Notebook Instance

Function

This API is used to prolong a notebook instance.

Constraints

None

URI

PATCH /v1/{project_id}/notebooks/{id}/lease

Table 4-74 Path Parameters

Parameter	Mandatory	Type	Description
id	Yes	String	Notebook instance ID.
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .

Table 4-75 Query Parameters

Parameter	Mandatory	Type	Description
duration	No	Long	Renewal duration. You are advised to set this parameter in leaseReq. If the request parameter contains duration, the value of leaseReq is ignored and the automatic instance stop type is scheduled stop. (Unit: ms)

Request Parameters

Table 4-76 Request body parameters

Parameter	Mandatory	Type	Description
duration	No	Long	Indicates the running duration starting from the current time. After the running duration expires, the system automatically stops. Unit: ms.
type	No	String	Automatic stop type. The default value is timing.

Response Parameters

Status code: 200

Table 4-77 Response body parameters

Parameter	Type	Description
create_at	Long	Time (UTC) when the instance is created, accurate to millisecond.
duration	Long	Instance running duration, which is calculated based on the instance creation time. If the instance creation time plus the duration is greater than the current time, the system automatically stops the instance.
enable	Boolean	Whether to enable auto stop of the instance.
type	String	Indicates the automatic stop type.
update_at	Long	Time (UTC) when the instance is last updated (excluding the keepalive heartbeat time), accurate to millisecond.

Example Requests

Prolong the available duration of a running notebook instance.

```
https://{endpoint}/v1/{project_id}/notebooks/{id}/lease
```

```
{
  "duration" : 3600000,
  "type" : "timing"
}
```

Example Responses

Status code: 200

OK

```
{  
  "create_at" : 1638841744515,  
  "duration" : 3600000,  
  "enable" : true,  
  "type" : "TIMING",  
  "update_at" : 1638843018759  
}
```

Status Codes

Status Code	Description
200	OK
204	No Content
401	Unauthorized
403	Forbidden

Error Codes

See [Error Codes](#).

4.11 Starting a Notebook Instance

Function

This API is used to start a notebook instance.

Constraints

None

URI

POST /v1/{project_id}/notebooks/{id}/start

Table 4-78 Path Parameters

Parameter	Mandatory	Type	Description
id	Yes	String	Notebook instance ID.

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .

Table 4-79 Query Parameters

Parameter	Mandatory	Type	Description
duration	No	Long	Running duration after startup, in milliseconds
type	No	String	Automatic stop type. The default value is timing.

Request Parameters

Table 4-80 Request body parameters

Parameter	Mandatory	Type	Description
-	No	Object	notebookStartRequest

Response Parameters

Status code: 200

Table 4-81 Response body parameters

Parameter	Type	Description
action_progress	Array of JobProgress objects	Instance initialization progress.
description	String	Instance description
endpoints	Array of EndpointsRes objects	This section describes how to configure the local IDE (such as PyCharm and VS Code) or SSH client to remotely access a notebook instance through SSH.
fail_reason	String	Instance failure cause
flavor	String	Instance flavor
id	String	Instance ID.

Parameter	Type	Description
image	Image object	Instance image
lease	Lease object	Countdown to automatic instance stop.
name	String	Instance name
pool	Pool object	Basic information about the dedicated pool, which is returned by the instance created in the dedicated pool.
status	String	Instance status. Options: <ul style="list-style-type: none"> • INIT: The instance is being initialized. • CREATING: The instance is being created. • STARTING: The instance is starting. • STOPPING: The instance is being stopped. • DELETING: The instance is being deleted. • RUNNING: The instance is running. • STOPPED: The instance has been stopped. • SNAPSHOTTING: The image of the instance is being saved. • CREATE_FAILED: Creating the instance failed. • START_FAILED: Starting the instance failed. • DELETE_FAILED: Deleting the instance failed. • ERROR: An error occurred. • DELETED: The instance has been deleted. • FROZEN: The instance is frozen.
token	String	Token information used for notebook authentication.
url	String	URL for accessing the notebook instance
volume	VolumeRes object	Storage volume
workspace_id	String	Workspace ID. If no workspaces are available, the default value is 0 .
feature	String	Instance type. Options: <ul style="list-style-type: none"> • DEFAULT: free CodeLab instance. You can create only one. • NOTEBOOK: billed instance.

Table 4-82 JobProgress

Parameter	Type	Description
notebook_id	String	Instance ID.
status	String	Job status in a specified step. Options: <ul style="list-style-type: none"> ● WAITING: The job is awaiting. ● PROCESSING: The job is being processed. ● FAILED: The job failed. ● COMPLETED: The job is complete.
step	Integer	Job step. Options: <ul style="list-style-type: none"> ● 1: Prepare storage. ● 2: Prepare compute resources. ● 3: Configure the network. ● 4: Initialize the instance.
step_description	String	Description of a step in a job.

Table 4-83 EndpointsRes

Parameter	Type	Description
allowed_access_ips	Array of strings	Whitelist of public IP addresses that are allowed to access the notebook instance through SSH. By default, all public IP addresses can access the notebook instance. If this parameter is specified, only the clients with the specified IP addresses can access the notebook instance.
dev_service	String	Supported service. Options: <ul style="list-style-type: none"> ● NOTEBOOK: You can access notebook instances using HTTPS. ● SSH: You can remotely access the notebook instance through SSH.
ssh_keys	Array of strings	List of SSH key pairs. You can set multiple key pairs to access an SSH instance at the same time.

Table 4-84 Image

Parameter	Type	Description
arch	String	Processor architecture supported by the image. Options: <ul style="list-style-type: none"> ● X86_64: x86 architecture ● AARCH64: Arm architecture
create_at	Long	Specifies the time (UTC ms) when the image is created.
description	String	Image description with a maximum of 512 characters
dev_services	Array of strings	Services supported by the image. Options: <ul style="list-style-type: none"> ● NOTEBOOK: You can access the notebook instance using HTTPS. ● SSH: You can remotely access the notebook instance from a local IDE through SSH.
id	String	ID of the image used for creating notebook instances. The ID is in Universally Unique Identifier (UUID) format. For details, see Querying Supported Images .
name	String	Image name, which contains a maximum of 512 characters, including lowercase letters, digits, hyphens (-), underscores (_), and periods (.)
namespace	String	Organization to which the image belongs. You can create and view the organization on the Organization Management page of the SWR console.
origin	String	Image source, which defaults to CUSTOMIZE . Options: <ul style="list-style-type: none"> ● CUSTOMIZE: user-defined image ● IMAGE_SAVE: image saved using a development environment instance
resource_categories	Array of strings	Flavors supported by the image. Options: <ul style="list-style-type: none"> ● CPU ● GPU

Parameter	Type	Description
service_type	String	Supported image types. Options: <ul style="list-style-type: none"> ● COMMON: common image ● INFERENCE: image used for inference ● TRAIN: image used for training ● DEV: image used for development and debugging ● UNKNOWN: image whose supported services are not specified
size	Long	Specifies the image size, in KB.
status	String	Image status. Options: <ul style="list-style-type: none"> ● INIT: The image is being initialized. ● CREATING: The image is being saved. In this case, the notebook instance is unavailable. ● CREATE_FAILED: Saving the image failed. ● ERROR: An error occurs. ● DELETED: The image has been deleted. ● ACTIVE: The image has been saved, which you can view on the SWR console and use to create notebook instances.
status_message	String	Build information during image saving
support_resource_categories	Array of strings	Flavors supported by the image. Options: <ul style="list-style-type: none"> ● CPU ● GPU
swr_path	String	SWR image address
tag	String	Image tag
type	String	Image type. Options: <ul style="list-style-type: none"> ● BUILD_IN: built-in system image ● DEDICATED: private image
update_at	Long	Specifies the time (UTC ms) when the image was last updated.
visibility	String	Image visibility. Options: <ul style="list-style-type: none"> ● PRIVATE: private image ● PUBLIC: All users can perform read-only operations based on the image ID.
workspace_id	String	Workspace ID. If no workspaces are available, the default value is 0 .

Table 4-85 Lease

Parameter	Type	Description
create_at	Long	Time (UTC) when the instance is created, accurate to millisecond.
duration	Long	Instance running duration, which is calculated based on the instance creation time. If the instance creation time plus the duration is greater than the current time, the system automatically stops the instance.
enable	Boolean	Whether to enable auto stop of the instance.
type	String	Indicates the automatic stop type.
update_at	Long	Time (UTC) when the instance is last updated (excluding the keepalive heartbeat time), accurate to millisecond.

Table 4-86 Pool

Parameter	Type	Description
id	String	ID of a dedicated resource pool
name	String	Name of a dedicated resource pool

Table 4-87 VolumeRes

Parameter	Type	Description
capacity	Integer	Storage capacity. The default value is 5 GB for EVS and 50 GB for EFS. The maximum value is 4096 GB.
category	String	Supported storage types. For details about the differences between the storage types, see Selecting Storage in DevEnviron . Options: <ul style="list-style-type: none"> • SFS: Scalable File Service • EVS
mount_path	String	Directory of the notebook instance to which OBS storage is mounted. Currently, the directory is <code>/home/ma-user/work/</code> .

Parameter	Type	Description
ownership	String	Owner to which the resource belongs. Options: <ul style="list-style-type: none"> ● MANAGED: Resources are managed by service. ● DEDICATED: Resources are managed by user account. This mode is supported only when the instance category is EFS.
status	String	EVS disk capacity expansion status, which is RESIZING during capacity expansion and does not affect the instance.

Example Requests

```
{}
```

Example Responses

Status code: 200

OK

```
{
  "description": "api-test",
  "feature": "NOTEBOOK",
  "flavor": "modelarts.vm.cpu.2u",
  "id": "f9937afa-4451-42db-a76b-72d624749f66",
  "image": {
    "description": "description",
    "id": "e1a07296-22a8-4f05-8bc8-e936c8e54090",
    "name": "notebook2.0-mul-kernel-cpu-cp36",
    "swr_path": "swr.xxxx.com/atelier/notebook2.0-mul-kernel-cpu-cp36:3.3.2-release_v1",
    "tag": "3.3.2-release_v1",
    "type": "BUILD_IN"
  },
  "lease": {
    "create_at": 1638841744515,
    "duration": 6327212,
    "enable": true,
    "type": "TIMING",
    "update_at": 1638844471727
  },
  "name": "notebooks_test",
  "status": "STARTING",
  "token": "5cc60e8b-8772-7690-efd6-a5874ca387c0",
  "url": "https://authoring-modelarts-xxxx.xxxx.com/f9937afa-4451-42db-a76b-72d624749f66/lab",
  "volume": {
    "category": "EFS",
    "ownership": "MANAGED",
    "mount_path": "/home/ma-user/work/",
    "capacity": 50
  },
  "workspace_id": "0"
}
```

Status Codes

Status Code	Description
200	OK
201	Created
401	Unauthorized
403	Forbidden
404	Not Found

Error Codes

See [Error Codes](#).

4.12 Stopping a Notebook Instance

Function

This API is used to stop a notebook instance.

Constraints

None

URI

POST /v1/{project_id}/notebooks/{id}/stop

Table 4-88 Path Parameters

Parameter	Mandatory	Type	Description
id	Yes	String	Notebook instance ID.
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .

Request Parameters

None

Response Parameters

Status code: 200

Table 4-89 Response body parameters

Parameter	Type	Description
action_progress	Array of JobProgress objects	Instance initialization progress.
description	String	Instance description
endpoints	Array of EndpointsRes objects	This section describes how to configure the local IDE (such as PyCharm and VS Code) or SSH client to remotely access a notebook instance through SSH.
fail_reason	String	Instance failure cause
flavor	String	Instance flavor
id	String	Instance ID.
image	Image object	Instance image
lease	Lease object	Countdown to automatic instance stop.
name	String	Instance name
pool	Pool object	Basic information about the dedicated pool, which is returned by the instance created in the dedicated pool.
status	String	Instance status. Options: <ul style="list-style-type: none"> • INIT: The instance is being initialized. • CREATING: The instance is being created. • STARTING: The instance is starting. • STOPPING: The instance is being stopped. • DELETING: The instance is being deleted. • RUNNING: The instance is running. • STOPPED: The instance has been stopped. • SNAPSHOTTING: The image of the instance is being saved. • CREATE_FAILED: Creating the instance failed. • START_FAILED: Starting the instance failed. • DELETE_FAILED: Deleting the instance failed. • ERROR: An error occurred. • DELETED: The instance has been deleted. • FROZEN: The instance is frozen.
token	String	Token information used for notebook authentication.

Parameter	Type	Description
url	String	URL for accessing the notebook instance
volume	VolumeRes object	Storage volume
workspace_id	String	Workspace ID. If no workspaces are available, the default value is 0 .
feature	String	Instance type. Options: <ul style="list-style-type: none"> ● DEFAULT: free CodeLab instance. You can create only one. ● NOTEBOOK: billed instance.

Table 4-90 JobProgress

Parameter	Type	Description
notebook_id	String	Instance ID.
status	String	Job status in a specified step. Options: <ul style="list-style-type: none"> ● WAITING: The job is awaiting. ● PROCESSING: The job is being processed. ● FAILED: The job failed. ● COMPLETED: The job is complete.
step	Integer	Job step. Options: <ul style="list-style-type: none"> ● 1: Prepare storage. ● 2: Prepare compute resources. ● 3: Configure the network. ● 4: Initialize the instance.
step_description	String	Description of a step in a job.

Table 4-91 EndpointsRes

Parameter	Type	Description
allowed_access_ips	Array of strings	Whitelist of public IP addresses that are allowed to access the notebook instance through SSH. By default, all public IP addresses can access the notebook instance. If this parameter is specified, only the clients with the specified IP addresses can access the notebook instance.

Parameter	Type	Description
dev_service	String	Supported service. Options: <ul style="list-style-type: none"> • NOTEBOOK: You can access notebook instances using HTTPS. • SSH: You can remotely access the notebook instance through SSH.
ssh_keys	Array of strings	List of SSH key pairs. You can set multiple key pairs to access an SSH instance at the same time.

Table 4-92 Image

Parameter	Type	Description
arch	String	Processor architecture supported by the image. Options: <ul style="list-style-type: none"> • X86_64: x86 architecture • AARCH64: Arm architecture
create_at	Long	Specifies the time (UTC ms) when the image is created.
description	String	Image description with a maximum of 512 characters
dev_services	Array of strings	Services supported by the image. Options: <ul style="list-style-type: none"> • NOTEBOOK: You can access the notebook instance using HTTPS. • SSH: You can remotely access the notebook instance from a local IDE through SSH.
id	String	ID of the image used for creating notebook instances. The ID is in Universally Unique Identifier (UUID) format. For details, see Querying Supported Images .
name	String	Image name, which contains a maximum of 512 characters, including lowercase letters, digits, hyphens (-), underscores (_), and periods (.)
namespace	String	Organization to which the image belongs. You can create and view the organization on the Organization Management page of the SWR console.

Parameter	Type	Description
origin	String	Image source, which defaults to CUSTOMIZE . Options: <ul style="list-style-type: none"> ● CUSTOMIZE: user-defined image ● IMAGE_SAVE: image saved using a development environment instance
resource_categories	Array of strings	Flavors supported by the image. Options: <ul style="list-style-type: none"> ● CPU ● GPU
service_type	String	Supported image types. Options: <ul style="list-style-type: none"> ● COMMON: common image ● INFERENCE: image used for inference ● TRAIN: image used for training ● DEV: image used for development and debugging ● UNKNOWN: image whose supported services are not specified
size	Long	Specifies the image size, in KB.
status	String	Image status. Options: <ul style="list-style-type: none"> ● INIT: The image is being initialized. ● CREATING: The image is being saved. In this case, the notebook instance is unavailable. ● CREATE_FAILED: Saving the image failed. ● ERROR: An error occurs. ● DELETED: The image has been deleted. ● ACTIVE: The image has been saved, which you can view on the SWR console and use to create notebook instances.
status_message	String	Build information during image saving
support_resource_categories	Array of strings	Flavors supported by the image. Options: <ul style="list-style-type: none"> ● CPU ● GPU
swr_path	String	SWR image address
tag	String	Image tag
type	String	Image type. Options: <ul style="list-style-type: none"> ● BUILD_IN: built-in system image ● DEDICATED: private image

Parameter	Type	Description
update_at	Long	Specifies the time (UTC ms) when the image was last updated.
visibility	String	Image visibility. Options: <ul style="list-style-type: none"> ● PRIVATE: private image ● PUBLIC: All users can perform read-only operations based on the image ID.
workspace_id	String	Workspace ID. If no workspaces are available, the default value is 0 .

Table 4-93 Lease

Parameter	Type	Description
create_at	Long	Time (UTC) when the instance is created, accurate to millisecond.
duration	Long	Instance running duration, which is calculated based on the instance creation time. If the instance creation time plus the duration is greater than the current time, the system automatically stops the instance.
enable	Boolean	Whether to enable auto stop of the instance.
type	String	Indicates the automatic stop type.
update_at	Long	Time (UTC) when the instance is last updated (excluding the keepalive heartbeat time), accurate to millisecond.

Table 4-94 Pool

Parameter	Type	Description
id	String	ID of a dedicated resource pool
name	String	Name of a dedicated resource pool

Table 4-95 VolumeRes

Parameter	Type	Description
capacity	Integer	Storage capacity. The default value is 5 GB for EVS and 50 GB for EFS. The maximum value is 4096 GB.

Parameter	Type	Description
category	String	Supported storage types. For details about the differences between the storage types, see Selecting Storage in DevEnviron . Options: <ul style="list-style-type: none"> • SFS: Scalable File Service • EVS
mount_path	String	Directory of the notebook instance to which OBS storage is mounted. Currently, the directory is <code>/home/ma-user/work/</code> .
ownership	String	Owner to which the resource belongs. Options: <ul style="list-style-type: none"> • MANAGED: Resources are managed by service. • DEDICATED: Resources are managed by user account. This mode is supported only when the instance category is EFS.
status	String	EVS disk capacity expansion status, which is RESIZING during capacity expansion and does not affect the instance.

Example Requests

```
POST https://{endpoint}/v1/{project_id}/notebooks/{id}/stop
```

Example Responses

Status code: 200

OK

```
{
  "description": "api-test",
  "feature": "NOTEBOOK",
  "flavor": "modelarts.vm.cpu.2u",
  "id": "f9937afa-4451-42db-a76b-72d624749f66",
  "image": {
    "description": "description",
    "id": "e1a07296-22a8-4f05-8bc8-e936c8e54090",
    "name": "notebook2.0-mul-kernel-cpu-cp36",
    "swr_path": "swr.xxxx.com/atelier/notebook2.0-mul-kernel-cpu-cp36:3.3.2-release_v1",
    "tag": "3.3.2-release_v1",
    "type": "BUILD_IN"
  },
  "lease": {
    "create_at": 1638841744515,
    "duration": 5313106,
    "enable": true,
    "update_at": 1638843457621
  },
  "name": "notebooks_test",
  "status": "STOPPING",
  "token": "7bddd8ff-8a0d-e063-3107-c70a862832de",
  "url": "https://authoring-modelarts-xxxxx.xxxx.com/f9937afa-4451-42db-a76b-72d624749f66/lab",
  "volume": {
    "category": "EFS",
```

```

"ownership" : "MANAGED",
"mount_path" : "/home/ma-user/work/",
"capacity" : 50
},
"workspace_id" : "0"
}

```

Status Codes

Status Code	Description
200	OK
201	Created
401	Unauthorized
403	Forbidden
404	Not Found

Error Codes

See [Error Codes](#).

4.13 Obtaining the Notebook Instances with OBS Storage Mounted

Function

This API is used to obtain the notebook instances with OBS storage mounted.

Constraints

None

URI

GET /v1/{project_id}/notebooks/{instance_id}/storage

Table 4-96 Path Parameters

Parameter	Mandatory	Type	Description
instance_id	Yes	String	Notebook instance ID.
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .

Request Parameters

None

Response Parameters

Status code: 200

Table 4-97 Response body parameters

Parameter	Type	Description
current	Integer	Current page
data	Array of DataVolumesRes objects	Data
pages	Integer	Total pages
size	Integer	Number of records on each page
total	Long	Total records

Table 4-98 DataVolumesRes

Parameter	Type	Description
category	String	Storage type The value can be OBS.
id	String	ID of the instance with OBS storage mounted.
mount_path	String	Path where OBS storage is mounted to a notebook instance.
status	String	Status of OBS storage to be mounted. Options: <ul style="list-style-type: none"> ● MOUNTING: being mounted ● MOUNT_FAILED: mounting failed ● MOUNTED: mounted ● UNMOUNTING: being unmounted ● UNMOUNT_FAILED: unmounting failed ● UNMOUNTED: unmounted
uri	String	OBS object path

Example Requests

GET https://{endpoint}/v1/{project_id}/notebooks/{instance_id}/storage

Example Responses

Status code: 200

OK

```
{
  "current" : 1,
  "data" : [ {
    "id" : "91dd2d3f-2d92-475f-a375-04636af26cc9",
    "category" : "OBSFS",
    "mount_path" : "/data/wang/",
    "uri" : "obs://authoring-test/wang/",
    "status" : "MOUNTED"
  } ],
  "pages" : 1,
  "size" : 1,
  "total" : 1
}
```

Status Codes

Status Code	Description
200	OK
401	Unauthorized
403	Forbidden
404	Not Found

Error Codes

See [Error Codes](#).

4.14 OBS Storage Mounting

Function

An OBS parallel file system can be mounted to a specified file directory of a running notebook instance. After the mounting, objects in the OBS parallel file system can be read and written in the container as a file system.

Constraints

None

URI

POST /v1/{project_id}/notebooks/{instance_id}/storage

Table 4-99 Path Parameters

Parameter	Mandatory	Type	Description
instance_id	Yes	String	Notebook instance ID.

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .

Request Parameters

Table 4-100 Request body parameters

Parameter	Mandatory	Type	Description
category	No	String	Storage type The value can be OBS.
mount_path	No	String	Path mounted to the notebook instance. The path must be in the /data/ subdirectory of the notebook instance.
uri	No	String	OBS object path, for example, obs://modelarts/notebook/

Response Parameters

Status code: 200

Table 4-101 Response body parameters

Parameter	Type	Description
category	String	Storage type The value can be OBS.
id	String	ID of the instance with OBS storage mounted.
mount_path	String	Path where OBS storage is mounted to a notebook instance.
status	String	Status of OBS storage to be mounted. Options: <ul style="list-style-type: none"> ● MOUNTING: being mounted ● MOUNT_FAILED: mounting failed ● MOUNTED: mounted ● UNMOUNTING: being unmounted ● UNMOUNT_FAILED: unmounting failed ● UNMOUNTED: unmounted
uri	String	OBS object path

Example Requests

The following is an example of how to dynamically mount an OBS parallel file system to the `/data/wang/` directory in the instance.

```
{
  "category": "OBS",
  "mount_path": "/data/wang/",
  "uri": "obs://authoring-test/wang/"
}
```

Example Responses

Status code: 200

OK

```
{
  "id": "91dd2d3f-2d92-475f-a375-04636af26cc9",
  "category": "OBSFS",
  "mount_path": "/data/wang/",
  "uri": "obs://authoring-test/wang/",
  "status": "MOUNTING"
}
```

Status Codes

Status Code	Description
200	OK
201	Created
401	Unauthorized
403	Forbidden
404	Not Found

Error Codes

See [Error Codes](#).

4.15 Obtaining Details About a Notebook Instance with OBS Storage Mounted

Function

This API is used to obtain details about a notebook instance with OBS storage mounted.

Constraints

None

URI

GET /v1/{project_id}/notebooks/{instance_id}/storage/{storage_id}

Table 4-102 Path Parameters

Parameter	Mandatory	Type	Description
instance_id	Yes	String	Notebook instance ID.
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .
storage_id	Yes	String	OBS storage ID.

Request Parameters

None

Response Parameters

Status code: 200

Table 4-103 Response body parameters

Parameter	Type	Description
category	String	Storage type The value can be OBS.
id	String	ID of the instance with OBS storage mounted.
mount_path	String	Path where OBS storage is mounted to a notebook instance.
status	String	Status of OBS storage to be mounted. Options: <ul style="list-style-type: none"> ● MOUNTING: being mounted ● MOUNT_FAILED: mounting failed ● MOUNTED: mounted ● UNMOUNTING: being unmounted ● UNMOUNT_FAILED: unmounting failed ● UNMOUNTED: unmounted
uri	String	OBS object path

Example Requests

GET https://{endpoint}/v1/{project_id}/notebooks/{instance_id}/storage/{storage_id}

Example Responses

Status code: 200

OK

```
{
  "id" : "91dd2d3f-2d92-475f-a375-04636af26cc9",
  "category" : "OBSFS",
  "mount_path" : "/data/wang/",
  "uri" : "obs://authoring-test/wang/",
  "status" : "MOUNTED"
}
```

Status Codes

Status Code	Description
200	OK
401	Unauthorized
403	Forbidden
404	Not Found

Error Codes

See [Error Codes](#).

4.16 Unmounting OBS Storage from a Notebook Instance

Function

This API is used to unmount OBS storage from a notebook instance. After OBS storage is unmounted, OBS objects remain unchanged but cannot be operated in the notebook container.

Constraints

None

URI

DELETE /v1/{project_id}/notebooks/{instance_id}/storage/{storage_id}

Table 4-104 Path Parameters

Parameter	Mandatory	Type	Description
instance_id	Yes	String	Notebook instance ID.
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .
storage_id	Yes	String	OBS storage ID.

Request Parameters

None

Response Parameters

Status code: 200

Table 4-105 Response body parameters

Parameter	Type	Description
category	String	Storage type The value can be OBS.
id	String	ID of the instance with OBS storage mounted.
mount_path	String	Path where OBS storage is mounted to a notebook instance.
status	String	Status of OBS storage to be mounted. Options: <ul style="list-style-type: none"> ● MOUNTING: being mounted ● MOUNT_FAILED: mounting failed ● MOUNTED: mounted ● UNMOUNTING: being unmounted ● UNMOUNT_FAILED: unmounting failed ● UNMOUNTED: unmounted
uri	String	OBS object path

Example Requests

```
DELETE https://{endpoint}/v1/{project_id}/notebooks/{instance_id}/storage/{storage_id}
```

Example Responses

Status code: 200

OK

```
{
  "category" : "OBSFS",
```

```

{id" : "91dd2d3f-2d92-475f-a375-04636af26cc9",
"mount_path" : "/data/wang/",
"status" : "UNMOUNTING",
"uri" : "obs://authoring-test/wang/"
}

```

Status Codes

Status Code	Description
200	OK
204	No Content
401	Unauthorized
403	Forbidden

Error Codes

See [Error Codes](#).

4.17 Querying Supported Images

Function

This API is used to query all images by page based on specified conditions.

Constraints

None

URI

GET /v1/{project_id}/images

Table 4-106 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .

Table 4-107 Query Parameters

Parameter	Mandatory	Type	Description
limit	No	Integer	Number of records on each page. The default value is 200 .
name	No	String	Image name, which contains a maximum of 512 characters, including lowercase letters, digits, hyphens (-), underscores (_), and periods (.)
name_fuzzy_match	No	Boolean	Whether the image name is used for fuzzy match. The default value is true .
namespace	No	String	Organization to which the image belongs. You can create and view the organization on the Organization Management page of the SWR console.
offset	No	Integer	Start offset of the records on each page. The default value is 0 .
service_type	No	String	Supported image types. Options: <ul style="list-style-type: none"> ● COMMON: common image ● INFERENCE: image used for inference ● TRAIN: image used for training ● DEV: image used for development and debugging ● UNKNOWN: image whose supported services are not specified
sort_dir	No	String	Sorting mode. The options are ASC (ascending order) and DESC (descending order). The default value is DESC .
sort_key	No	String	Sorting fields. Separate multiple fields with commas (,).

Parameter	Mandatory	Type	Description
type	No	String	Image type. Options: <ul style="list-style-type: none"> • BUILD_IN: built-in system image • DEDICATED: private image
workspace_id	No	String	Workspace ID. If no workspaces are available, the default value is 0 .

Request Parameters

None

Response Parameters

Status code: 200

Table 4-108 Response body parameters

Parameter	Type	Description
current	Integer	Current page
data	Array of ImageResponse objects	Data
pages	Integer	Total pages
size	Integer	Number of records on each page
total	Long	Total records

Table 4-109 ImageResponse

Parameter	Type	Description
arch	String	Processor architecture supported by the image. Options: <ul style="list-style-type: none"> • X86_64: x86 architecture • AARCH64: Arm architecture
create_at	Long	Time (UTC) when the image is created, accurate to millisecond
description	String	Image description with a maximum of 512 characters

Parameter	Type	Description
dev_services	Array of strings	Services supported by the image. Options: <ul style="list-style-type: none"> • NOTEBOOK: You can access a notebook instance using HTTPS. • SSH: You can remotely access the notebook instance from a local IDE through SSH.
id	String	ID of the image used for creating notebook instances. The ID is in Universally Unique Identifier (UUID) format. For details, see Querying Supported Images .
name	String	Image name, which contains a maximum of 512 characters, including lowercase letters, digits, hyphens (-), underscores (_), and periods (.)
namespace	String	Organization to which the image belongs. You can create and view the organization on the Organization Management page of the SWR console.
origin	String	Image source, which defaults to CUSTOMIZE . Options: <ul style="list-style-type: none"> • CUSTOMIZE: user-defined image • IMAGE_SAVE: image saved using a development environment instance
resource_categories	Array of strings	Flavors supported by the image. Options: <ul style="list-style-type: none"> • CPU • GPU
service_type	String	Supported image types. Options: <ul style="list-style-type: none"> • COMMON: common image • INFERENCE: image used for inference • TRAIN: image used for training • DEV: image used for development and debugging • UNKNOWN: image whose supported services are not specified
size	Long	Specifies the image size, in KB.

Parameter	Type	Description
status	String	Image status. Options: <ul style="list-style-type: none"> • INIT: The image is being initialized. • CREATING: The image is being saved. In this case, the notebook instance is unavailable. • CREATE_FAILED: Saving the image failed. • ERROR: An error occurs. • DELETED: The image has been deleted. • ACTIVE: The image has been saved, which you can view on the SWR console and use to create notebook instances.
status_message	String	Build information during image saving
swr_path	String	SWR image address
tag	String	Image tag
type	String	Image type. Options: <ul style="list-style-type: none"> • BUILD_IN: built-in system image • DEDICATED: image you have saved
update_at	Long	Time (UTC) when the image was last updated, accurate to millisecond
visibility	String	Image visibility. Options: <ul style="list-style-type: none"> • PRIVATE: private image • PUBLIC: All users can perform read-only operations based on the image ID.
workspace_id	String	Workspace ID. If no workspaces are available, the default value is 0 .

Example Requests

```
GET https://{endpoint}/v1/{project_id}/images
```

Example Responses

Status code: 200

OK

```
{
  "current" : 4,
  "data" : [ {
    "arch" : "x86_64",
    "create_at" : 1638841805439,
    "description" : "AI inference application development, preconfigured ModelBox and AI engine LibTorch, only SSH connection supported.",
```

```

"dev_services" : [ "AI_FLOW", "SSH" ],
"id" : "e1a07296-22a8-4f05-8bc8-e936c8e54103",
"name" : "modelbox1.3.0-libtorch1.9.1-cuda10.2-cudnn8-euler2.9.6",
"resource_categories" : [ "GPU" ],
"service_type" : "TRAIN",
"status" : "ACTIVE",
"swr_path" : "swr.xxx.com/atelier/modelarts-modelbox-libtorch-gpu-x86:1.3.0-20221027202714-160a680",
"tag" : "1.3.0-20221027202714-160a680",
"type" : "BUILD_IN",
"update_at" : 1638234504492,
"workspace_id" : "0"
}, {
  "arch" : "x86_64",
  "create_at" : 1638841805439,
  "description" : "CPU and GPU general algorithm development and training, preconfigured with AI engine
PyTorch1.8",
  "dev_services" : [ "NOTEBOOK", "SSH" ],
  "id" : "278e88d1-5b71-4766-8502-b3ba72e824d9",
  "name" : "pytorch1.8-cuda10.2-cudnn7-ubuntu18.04",
  "resource_categories" : [ "GPU", "CPU" ],
  "service_type" : "COMMON",
  "status" : "ACTIVE",
  "swr_path" : "swr.xxx.com/atelier/pytorch_1_8:pytorch_1.8.0-cuda_10.2-py_3.7-ubuntu_18.04-
x86_64-20220926104358-041ba2e",
  "tag" : "pytorch_1.8.0-cuda_10.2-py_3.7-ubuntu_18.04-x86_64-20220926104358-041ba2e",
  "type" : "BUILD_IN",
  "update_at" : 1638234504492,
  "workspace_id" : "0"
}],
"pages" : 54,
"size" : 2,
"total" : 107
}

```

Status Codes

Status Code	Description
200	OK
401	Unauthorized
403	Forbidden
404	Not Found

Error Codes

See [Error Codes](#).

4.18 Registering a Custom Image

Function

Register a custom image with ModelArts Image Management.

Constraints

None

URI

POST /v1/{project_id}/images

Table 4-110 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .

Request Parameters

Table 4-111 Request body parameters

Parameter	Mandatory	Type	Description
arch	No	String	Processor architecture supported by the image. Options: <ul style="list-style-type: none"> • X86_64: x86 architecture • AARCH64: Arm architecture
description	No	String	Provides supplementary information about the image. The value contains a maximum of 512 characters.
origin	No	String	Image source, which defaults to CUSTOMIZE . Options: <ul style="list-style-type: none"> • CUSTOMIZE: user-defined image • IMAGE_SAVE: image saved using a development environment instance
resource_category	No	Array of strings	Flavors supported by the image. Options: <ul style="list-style-type: none"> • CPU • GPU

Parameter	Mandatory	Type	Description
service_type	No	String	Supported image types. Options: <ul style="list-style-type: none"> ● COMMON: common image ● INFERENCE: image used for inference ● TRAIN: image used for training ● DEV: image used for development and debugging ● UNKNOWN: image whose supported services are not specified
services	No	Array of strings	Services supported by the image. Options: <ul style="list-style-type: none"> ● NOTEBOOK: You can access a notebook instance using HTTPS. ● SSH: You can remotely access the notebook instance from a local IDE through SSH.
swr_path	No	String	SWR image address
visibility	No	String	Image visibility. Options: <ul style="list-style-type: none"> ● PRIVATE: private image ● PUBLIC: All users can perform read-only operations based on the image ID.
workspace_id	No	String	Workspace ID. If no workspaces are available, the default value is 0 .

Response Parameters

Status code: 200

Table 4-112 Response body parameters

Parameter	Type	Description
arch	String	Processor architecture supported by the image. Options: <ul style="list-style-type: none"> ● X86_64: x86 architecture ● AARCH64: Arm architecture
create_at	Long	Specifies the time (UTC ms) when the image is created.
description	String	Image description with a maximum of 512 characters
dev_services	Array of strings	Services supported by the image. Options: <ul style="list-style-type: none"> ● NOTEBOOK: You can access the notebook instance using HTTPS. ● SSH: You can remotely access the notebook instance from a local IDE through SSH.
id	String	ID of the image used for creating notebook instances. The ID is in Universally Unique Identifier (UUID) format. For details, see Querying Supported Images .
name	String	Image name, which contains a maximum of 512 characters, including lowercase letters, digits, hyphens (-), underscores (_), and periods (.)
namespace	String	Organization to which the image belongs. You can create and view the organization on the Organization Management page of the SWR console.
origin	String	Image source, which defaults to CUSTOMIZE . Options: <ul style="list-style-type: none"> ● CUSTOMIZE: user-defined image ● IMAGE_SAVE: image saved using a development environment instance
resource_categories	Array of strings	Flavors supported by the image. Options: <ul style="list-style-type: none"> ● CPU ● GPU

Parameter	Type	Description
service_type	String	Supported image types. Options: <ul style="list-style-type: none"> ● COMMON: common image ● INFERENCE: image used for inference ● TRAIN: image used for training ● DEV: image used for development and debugging ● UNKNOWN: image whose supported services are not specified
size	Long	Specifies the image size, in KB.
status	String	Image status. Options: <ul style="list-style-type: none"> ● INIT: The image is being initialized. ● CREATING: The image is being saved. In this case, the notebook instance is unavailable. ● CREATE_FAILED: Saving the image failed. ● ERROR: An error occurs. ● DELETED: The image has been deleted. ● ACTIVE: The image has been saved, which you can view on the SWR console and use to create notebook instances.
status_message	String	Build information during image saving
support_resource_categories	Array of strings	Flavors supported by the image. Options: <ul style="list-style-type: none"> ● CPU ● GPU
swr_path	String	SWR image address
tag	String	Image tag
type	String	Image type. Options: <ul style="list-style-type: none"> ● BUILD_IN: built-in system image ● DEDICATED: private image
update_at	Long	Specifies the time (UTC ms) when the image was last updated.
visibility	String	Image visibility. Options: <ul style="list-style-type: none"> ● PRIVATE: private image ● PUBLIC: All users can perform read-only operations based on the image ID.
workspace_id	String	Workspace ID. If no workspaces are available, the default value is 0 .

Example Requests

The following is an example of how to register a custom image whose resource type is **CPU** and architecture is **X86_64**.

```
{
  "description": "",
  "resource_category": [ "CPU" ],
  "arch": "X86_64",
  "swr_path": "swr.xxx.com/op_svc_modelarts_container2/pytorch_1_8:train-pytorch_1.8.0-cuda_10.2-py_3.7"
}
```

Example Responses

Status code: 200

OK

```
{
  "arch": "x86_64",
  "create_at": 1671708630448,
  "description": "",
  "dev_services": [ "NOTEBOOK", "SSH" ],
  "id": "708ca95d-c601-4dc7-86b9-670adfd5e818",
  "name": "pytorch_1_8",
  "namespace": "op_svc_modelarts_container2",
  "origin": "CUSTOMIZE",
  "resource_categories": [ "CPU" ],
  "service_type": "UNKNOWN",
  "size": 3376133259,
  "status": "ACTIVE",
  "swr_path": "swr.xxx.com/op_svc_modelarts_container2/pytorch_1_8:train-pytorch_1.8.0-cuda_10.2-py_3.7",
  "tag": "train-pytorch_1.8.0-cuda_10.2-py_3.7",
  "type": "DEDICATED",
  "update_at": 1671708630448,
  "visibility": "PRIVATE",
  "workspace_id": "0"
}
```

Status Codes

Status Code	Description
200	OK
201	Created
401	Unauthorized
403	Forbidden
404	Not Found

Error Codes

See [Error Codes](#).

4.19 Obtaining User Image Groups

Function

Obtain the overview of user image information. The image name is used as the aggregated information.

Constraints

None

URI

GET /v1/{project_id}/images/group

Table 4-113 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .

Table 4-114 Query Parameters

Parameter	Mandatory	Type	Description
limit	No	Integer	Number of records on each page. The default value is 200 .
name	No	String	Image name, which contains a maximum of 512 characters, including lowercase letters, digits, hyphens (-), underscores (_), and periods (.)
name_fuzzy_match	No	Boolean	Whether the image name is used for fuzzy match. The default value is true .
namespace	No	String	Organization to which the image belongs. You can create and view the organization on the Organization Management page of the SWR console.

Parameter	Mandatory	Type	Description
offset	No	Integer	Start offset of the records on each page. The default value is 0 .
service_type	No	String	Supported image types. Options: <ul style="list-style-type: none"> ● COMMON: common image ● INFERENCE: image used for inference ● TRAIN: image used for training ● DEV: image used for development and debugging ● UNKNOWN: image whose supported services are not specified
sort_dir	No	String	Sorting mode. The options are ASC (ascending order) and DESC (descending order). The default value is DESC .
sort_key	No	String	Sorting fields. Separate multiple fields with commas (,).
type	No	String	Image type. Options: <ul style="list-style-type: none"> ● BUILD_IN: built-in system image ● DEDICATED: private image
workspace_id	No	String	Workspace ID. If no workspaces are available, the default value is 0 .

Request Parameters

None

Response Parameters

Status code: 200

Table 4-115 Response body parameters

Parameter	Type	Description
current	Integer	Current page number.
data	Array of ImageGroup objects	Indicates the data.
pages	Integer	Total number of pages
size	Integer	Number of records on each page
total	Long	Total number of records

Table 4-116 ImageGroup

Parameter	Type	Description
name	String	The image name.
create_at	Long	Specifies the time (UTC ms) when the image is created.
namespace	String	Mirror the SWR organization described.
update_at	Long	Specifies the time (UTC ms) when the image was last updated.
version_count	Integer	Specifies the number of image versions.

Example Requests

```
GET https://{endpoint}/v1/{project_id}/images/group
```

Example Responses

Status code: 200

OK

```
{
  "current" : 1,
  "data" : [ {
    "create_at" : 1652878011643,
    "name" : "123",
    "namespace" : "xxxxxx-infer-model-dev",
    "update_at" : 1652878531791,
    "version_count" : 1
  }, {
    "create_at" : 1671708630448,
    "name" : "pytorch_1_8",
    "namespace" : "op_svc_modelarts_container2",
    "update_at" : 1671708630448,
    "version_count" : 1
  }, {
    "create_at" : 1671093486722,
```



```

    "name" : "mock-service-python",
    "namespace" : "mock-service1",
    "update_at" : 1671093486722,
    "version_count" : 1
  } ],
  "pages" : 1,
  "size" : 3,
  "total" : 3
}

```

Status Codes

Status Code	Description
200	OK
401	Unauthorized
403	Forbidden
404	Not Found

Error Codes

See [Error Codes](#).

4.20 Obtaining Details of an Image

Function

Obtain the details of an image.

Constraints

None

URI

GET /v1/{project_id}/images/{id}

Table 4-117 Path Parameters

Parameter	Mandatory	Type	Description
id	Yes	String	Image ID
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .

Request Parameters

None

Response Parameters

Status code: 200

Table 4-118 Response body parameters

Parameter	Type	Description
arch	String	Processor architecture supported by the image. Options: <ul style="list-style-type: none"> ● X86_64: x86 architecture ● AARCH64: Arm architecture
create_at	Long	Specifies the time (UTC ms) when the image is created.
description	String	Image description with a maximum of 512 characters
dev_services	Array of strings	Services supported by the image. Options: <ul style="list-style-type: none"> ● NOTEBOOK: You can access the notebook instance using HTTPS. ● SSH: You can remotely access the notebook instance from a local IDE through SSH.
id	String	ID of the image used for creating notebook instances. The ID is in Universally Unique Identifier (UUID) format. For details, see Querying Supported Images .
name	String	Image name, which contains a maximum of 512 characters, including lowercase letters, digits, hyphens (-), underscores (_), and periods (.)
namespace	String	Organization to which the image belongs. You can create and view the organization on the Organization Management page of the SWR console.
origin	String	Image source, which defaults to CUSTOMIZE . Options: <ul style="list-style-type: none"> ● CUSTOMIZE: user-defined image ● IMAGE_SAVE: image saved using a development environment instance
resource_categories	Array of strings	Flavors supported by the image. Options: <ul style="list-style-type: none"> ● CPU ● GPU

Parameter	Type	Description
service_type	String	Supported image types. Options: <ul style="list-style-type: none"> ● COMMON: common image ● INFERENCE: image used for inference ● TRAIN: image used for training ● DEV: image used for development and debugging ● UNKNOWN: image whose supported services are not specified
size	Long	Specifies the image size, in KB.
status	String	Image status. Options: <ul style="list-style-type: none"> ● INIT: The image is being initialized. ● CREATING: The image is being saved. In this case, the notebook instance is unavailable. ● CREATE_FAILED: Saving the image failed. ● ERROR: An error occurs. ● DELETED: The image has been deleted. ● ACTIVE: The image has been saved, which you can view on the SWR console and use to create notebook instances.
status_message	String	Build information during image saving
support_resource_categories	Array of strings	Flavors supported by the image. Options: <ul style="list-style-type: none"> ● CPU ● GPU
swr_path	String	SWR image address
tag	String	Image tag
type	String	Image type. Options: <ul style="list-style-type: none"> ● BUILD_IN: built-in system image ● DEDICATED: private image
update_at	Long	Specifies the time (UTC ms) when the image was last updated.
visibility	String	Image visibility. Options: <ul style="list-style-type: none"> ● PRIVATE: private image ● PUBLIC: All users can perform read-only operations based on the image ID.
workspace_id	String	Workspace ID. If no workspaces are available, the default value is 0 .

Example Requests

```
GET https://{endpoint}/v1/{project_id}/images/{id}
```

Example Responses

Status code: 200

OK

```
{
  "arch": "x86_64",
  "create_at": 1638234504492,
  "description": "CPU and GPU general algorithm development and training, preconfigured with AI engine PyTorch1.8",
  "dev_services": [ "NOTEBOOK", "SSH" ],
  "id": "278e88d1-5b71-4766-8502-b3ba72e824d9",
  "name": "pytorch1.8-cuda10.2-cudnn7-ubuntu18.04",
  "resource_categories": [ "GPU", "CPU" ],
  "service_type": "COMMON",
  "status": "ACTIVE",
  "swr_path": "swr.xxx.com/atelier/pytorch_1_8:pytorch_1.8.0-cuda_10.2-py_3.7-ubuntu_18.04-x86_64-20220926104358-041ba2e",
  "tag": "pytorch_1.8.0-cuda_10.2-py_3.7-ubuntu_18.04-x86_64-20220926104358-041ba2e",
  "type": "BUILD_IN",
  "update_at": 1638234504492,
  "workspace_id": "0"
}
```

Status Codes

Status Code	Description
200	OK
401	Unauthorized
403	Forbidden
404	Not Found

Error Codes

See [Error Codes](#).

4.21 Deleting an Image

Function

Delete an image object. For a private image, you can also delete the image content from SWR using parameters.

Constraints

None

URI

DELETE /v1/{project_id}/images/{id}

Table 4-119 Path Parameters

Parameter	Mandatory	Type	Description
id	Yes	String	Image ID
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .

Table 4-120 Query Parameters

Parameter	Mandatory	Type	Description
force	No	Boolean	Delete the image content from SWR for private images.

Request Parameters

None

Response Parameters

Status code: 200

Table 4-121 Response body parameters

Parameter	Type	Description
arch	String	Processor architecture supported by the image. Options: <ul style="list-style-type: none"> • X86_64: x86 architecture • AARCH64: Arm architecture
create_at	Long	Specifies the time (UTC ms) when the image is created.
description	String	Image description with a maximum of 512 characters

Parameter	Type	Description
dev_services	Array of strings	Services supported by the image. Options: <ul style="list-style-type: none"> • NOTEBOOK: You can access the notebook instance using HTTPS. • SSH: You can remotely access the notebook instance from a local IDE through SSH.
id	String	ID of the image used for creating notebook instances. The ID is in Universally Unique Identifier (UUID) format. For details, see Querying Supported Images .
name	String	Image name, which contains a maximum of 512 characters, including lowercase letters, digits, hyphens (-), underscores (_), and periods (.)
namespace	String	Organization to which the image belongs. You can create and view the organization on the Organization Management page of the SWR console.
origin	String	Image source, which defaults to CUSTOMIZE . Options: <ul style="list-style-type: none"> • CUSTOMIZE: user-defined image • IMAGE_SAVE: image saved using a development environment instance
resource_categories	Array of strings	Flavors supported by the image. Options: <ul style="list-style-type: none"> • CPU • GPU
service_type	String	Supported image types. Options: <ul style="list-style-type: none"> • COMMON: common image • INFERENCE: image used for inference • TRAIN: image used for training • DEV: image used for development and debugging • UNKNOWN: image whose supported services are not specified
size	Long	Specifies the image size, in KB.

Parameter	Type	Description
status	String	Image status. Options: <ul style="list-style-type: none"> • INIT: The image is being initialized. • CREATING: The image is being saved. In this case, the notebook instance is unavailable. • CREATE_FAILED: Saving the image failed. • ERROR: An error occurs. • DELETED: The image has been deleted. • ACTIVE: The image has been saved, which you can view on the SWR console and use to create notebook instances.
status_message	String	Build information during image saving
support_res_categories	Array of strings	Flavors supported by the image. Options: <ul style="list-style-type: none"> • CPU • GPU
swr_path	String	SWR image address
tag	String	Image tag
type	String	Image type. Options: <ul style="list-style-type: none"> • BUILD_IN: built-in system image • DEDICATED: private image
update_at	Long	Specifies the time (UTC ms) when the image was last updated.
visibility	String	Image visibility. Options: <ul style="list-style-type: none"> • PRIVATE: private image • PUBLIC: All users can perform read-only operations based on the image ID.
workspace_id	String	Workspace ID. If no workspaces are available, the default value is 0 .

Example Requests

```
DELETE https://{endpoint}/v1/{project_id}/images/{id}
```

Example Responses

Status code: 200

OK

```
{
  "arch": "x86_64",
```

```
"create_at" : 1671093486722,  
"description" : "",  
"dev_services" : [ "NOTEBOOK", "SSH" ],  
"id" : "c9ab2b2f-edda-4556-bdbb-494e868d043c",  
"name" : "mock-service-python",  
"namespace" : "mock-service1",  
"origin" : "CUSTOMIZE",  
"resource_categories" : [ "CPU" ],  
"service_type" : "UNKNOWN",  
"size" : 387840321,  
"status" : "DELETED",  
"swr_path" : "swr.xxx.com/mock-service1/mock-service-python:0.0.17",  
"tag" : "0.0.17",  
"type" : "DEDICATED",  
"update_at" : 1671093486722,  
"visibility" : "PRIVATE",  
"workspace_id" : "0"  
}
```

Status Codes

Status Code	Description
200	OK
204	No Content
401	Unauthorized
403	Forbidden

Error Codes

See [Error Codes](#).

5 Training Management

5.1 Creating an Algorithm

Function

This API is used to create an algorithm.

URI

POST /v2/{project_id}/algorithms

Table 5-1 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .

Request Parameters

Table 5-2 Request body parameters

Parameter	Mandatory	Type	Description
metadata	No	metadata object	Algorithm metadata, which describes basic algorithm information.
job_config	No	job_config object	Algorithm configuration, such as the boot file.

Parameter	Mandatory	Type	Description
resource_requirements	No	Array of resource_requirements objects	Algorithm resource constraints. This parameter is optional. After this parameter is set, the console filters available public resource pools when the algorithm is used in training jobs.
advanced_config	No	advanced_config object	Advanced algorithm policy: <ul style="list-style-type: none"> • auto_search

Table 5-3 metadata

Parameter	Mandatory	Type	Description
id	No	Integer	Algorithm UUID. You do not need to set this parameter when creating an algorithm.
name	Yes	String	Algorithm name. The value must contain 1 to 64 characters consisting of only digits, letters, underscores (_), and hyphens (-).
description	No	String	Description of the algorithm. The value is a string of [0, 256] characters. The default value is NULL.
workspace_id	No	String	Workspace where a specified algorithm is located. The default value is 0 . The value 0 indicates the default workspace.
ai_project	No	String	AI project to which a specified algorithm belongs. The default value is default-ai-project . The AI project has been brought offline. Ignore it.

Table 5-4 job_config

Parameter	Mandatory	Type	Description
code_dir	No	String	Algorithm code directory, for example, <code>/usr/app/</code> . This parameter must be used together with boot_file .
boot_file	No	String	Code boot file of the algorithm, which needs to be stored in the code directory, for example, <code>/usr/app/boot.py</code> . This parameter must be used together with code_dir .
command	No	String	Container startup command of a custom image algorithm.
parameters	No	Array of Parameter objects	Running parameter of an algorithm.
inputs	No	Array of inputs objects	Data input of an algorithm.
outputs	No	Array of outputs objects	Data output of an algorithm.
engine	No	engine object	Algorithm engine.
parameters_customization	No	Boolean	Whether the algorithm allows hyperparameter customization during training job creation.

Table 5-5 Parameter

Parameter	Mandatory	Type	Description
name	No	String	Parameter name.
value	No	String	Parameter value.
description	No	String	Parameter description.
constraint	No	constraint object	Parameter constraint.
internationalization	No	internationalization object	Internationalization description.

Table 5-6 constraint

Parameter	Mandatory	Type	Description
type	No	String	Parameter type.
editable	No	Boolean	Whether the parameter is editable.
required	No	Boolean	Whether the parameter is mandatory.
sensitive	No	Boolean	Whether the parameter is sensitive This function is not implemented currently.
valid_type	No	String	Valid type.
valid_range	No	Array of strings	Valid range.

Table 5-7 i18n_description

Parameter	Mandatory	Type	Description
language	No	String	Language.
description	No	String	Description.

Table 5-8 inputs

Parameter	Mandatory	Type	Description
name	Yes	String	Name of the data input channel.
description	No	String	Description of the data input channel.
remote_constraints	No	Array of remote_constraints objects	Data input constraint.

Table 5-9 remote_constraints

Parameter	Mandatory	Type	Description
data_type	No	String	Data input type. The data storage location (OBS) and ModelArts dataset are supported.

Parameter	Mandatory	Type	Description
attributes	No	Array of Map<String,String> objects	Attributes if a dataset is used as the data input. Options: <ul style="list-style-type: none"> • data_format: Data format • data_segmentation: Data segmentation • dataset_type: Labeling type

Table 5-10 outputs

Parameter	Mandatory	Type	Description
name	Yes	String	Name of the data output channel.
description	No	String	Description of the data output channel.

Table 5-11 engine

Parameter	Mandatory	Type	Description
engine_id	No	String	Engine ID selected for an algorithm.
engine_name	No	String	Engine version name selected for an algorithm. If engine_id is specified, leave this parameter blank.
engine_version	No	String	Engine version name selected for an algorithm. If engine_id is specified, leave this parameter blank.
image_url	No	String	Custom image URL selected by an algorithm.

Table 5-12 resource_requirements

Parameter	Mandatory	Type	Description
key	No	String	Resource constraint. The options are as follows: <ul style="list-style-type: none"> flavor_type indicates the resource type. The value can be CPU, GPU, or Ascend (tag: hc, hk,fcs_super). device_distributed_mode: indicates whether to support multi-card training. The value can be multiple (supported) or singular (not supported). host_distributed_mode: indicates whether distributed training is supported. The value can be multiple (supported) or singular (not supported).
values	No	Array of strings	Value of the resource constraint key.
operator	No	String	Relationship between keys and values. Currently, only in is supported. For example: flavor_type in [CPU,GPU].

Table 5-13 advanced_config

Parameter	Mandatory	Type	Description
auto_search	No	auto_search object	Hyperparameter search policy.

Table 5-14 auto_search

Parameter	Mandatory	Type	Description
skip_search_parameters	No	String	Hyperparameter parameters that need to be skipped.
reward_attrs	No	Array of reward_attrs objects	List of search metrics.

Parameter	Mandatory	Type	Description
search_params	No	Array of search_params objects	Search parameters.
algo_configs	No	Array of algo_configs objects	Search algorithm configurations.

Table 5-15 reward_attrs

Parameter	Mandatory	Type	Description
name	No	String	Metric name.
mode	No	String	Search direction. <ul style="list-style-type: none"> • max: A larger metric value indicates better performance. • min: A smaller metric value indicates better performance.
regex	No	String	Regular expression of a metric.

Table 5-16 search_params

Parameter	Mandatory	Type	Description
name	No	String	Hyperparameter name.
param_type	No	String	Parameter type <ul style="list-style-type: none"> • If continuous is specified, the hyperparameter is of the continuous type. When an algorithm is used in a training job, continuous hyperparameters are displayed as text boxes on the console. - discrete: The hyperparameter is of the discrete type. When an algorithm is used for training jobs, discrete hyperparameters are displayed as a drop-down list box on the console.

Parameter	Mandatory	Type	Description
lower_bound	No	String	Lower bound of the hyperparameter.
upper_bound	No	String	Upper bound of the hyperparameter.
discrete_points_num	No	String	Number of discrete points of a continuous hyperparameter.
discrete_values	No	String	List of discrete hyperparameter values.

Table 5-17 algo_configs

Parameter	Mandatory	Type	Description
name	No	String	Name of the search algorithm.
params	No	Array of AutoSearchAlgoConfigParameter objects	Search algorithm parameters.

Table 5-18 AutoSearchAlgoConfigParameter

Parameter	Mandatory	Type	Description
key	No	String	Parameter key.
value	No	String	Parameter value.
type	No	String	Parameter type.

Response Parameters

Status code: 201

Table 5-19 Response body parameters

Parameter	Type	Description
metadata	metadata object	Algorithm metadata, which describes basic algorithm information.
job_config	job_config object	Algorithm configuration, such as the boot file.

Parameter	Type	Description
resource_requirements	Array of resource_requirements objects	Algorithm resource constraint. This parameter is optional. After this parameter is set, the console filters available public resource pools when the algorithm is used in training jobs.
advanced_config	advanced_config object	Advanced algorithm policy: <ul style="list-style-type: none"> • auto_search

Table 5-20 metadata

Parameter	Type	Description
id	Integer	Algorithm UUID. You do not need to set this parameter when creating an algorithm.
name	String	Algorithm name. The value must contain 1 to 64 characters consisting of only digits, letters, underscores (_), and hyphens (-).
description	String	Description of the algorithm. The value is a string of [0, 256] characters. The default value is NULL.
workspace_id	String	Workspace where a specified algorithm is located. The default value is 0 . The value 0 indicates the default workspace.
ai_project	String	AI project to which a specified algorithm belongs. The default value is default-ai-project . The AI project has been brought offline. Ignore it.
user_name	String	Username.
domain_id	String	Domain ID of a user.
source	String	Algorithm source.
api_version	String	Algorithm API version, which identifies the old and new ones.
is_valid	String	Algorithm availability.
state	String	Algorithm state.
tags	Array of Map<String,String> objects	Algorithm tags.
attr_list	Array of strings	Algorithm attribute list.

Parameter	Type	Description
version_num	Integer	Number of algorithm versions. The default value is 0 .
size	Integer	Algorithm size.
create_time	Long	Timestamp when the algorithm is created.
update_time	Long	Timestamp when the algorithm is updated.

Table 5-21 job_config

Parameter	Type	Description
code_dir	String	Algorithm code directory, for example, <code>/usr/app/</code> . This parameter must be used together with boot_file .
boot_file	String	Code boot file of the algorithm, which must be stored in the code directory, for example, <code>/usr/app/boot.py</code> . This parameter must be used with code_dir .
command	String	Container startup command of a custom image algorithm.
parameters	Array of Parameter objects	Running parameter of an algorithm.
inputs	Array of inputs objects	Data input of an algorithm.
outputs	Array of outputs objects	Data output of an algorithm.
engine	engine object	Algorithm engine.
code_tree	Array of code_tree objects	Algorithm directory tree
parameters_customization	Boolean	Whether the algorithm allows hyperparameter customization during training job creation.

Table 5-22 Parameter

Parameter	Type	Description
name	String	Parameter name.

Parameter	Type	Description
value	String	Parameter value.
description	String	Parameter description.
constraint	constraint object	Parameter constraint.
i18n_description	i18n_description object	Internationalization description.

Table 5-23 constraint

Parameter	Type	Description
type	String	Parameter type.
editable	Boolean	Whether the parameter is editable.
required	Boolean	Whether the parameter is mandatory.
sensitive	Boolean	Whether the parameter is sensitive This function is not implemented currently.
valid_type	String	Valid type.
valid_range	Array of strings	Valid range.

Table 5-24 i18n_description

Parameter	Type	Description
language	String	Language.
description	String	Description.

Table 5-25 inputs

Parameter	Type	Description
name	String	Name of the data input channel.
description	String	Description of the data input channel.
remote_constraints	Array of remote_constraints objects	Data input constraint.

Table 5-26 remote_constraints

Parameter	Type	Description
data_type	String	Data input type, including the data storage location and dataset.
attributes	Array of Map<String,String> objects	Attributes if a dataset is used as the data input. Options: <ul style="list-style-type: none"> • data_format: Data format • data_segmentation: Data segmentation • dataset_type: Labeling type

Table 5-27 outputs

Parameter	Type	Description
name	String	Name of the data output channel.
description	String	Description of the data output channel.

Table 5-28 engine

Parameter	Type	Description
engine_id	String	Engine ID selected for an algorithm.
engine_name	String	Engine version name selected for an algorithm. If engine_id is specified, leave this parameter blank.
engine_version	String	Engine version name selected for an algorithm. If engine_id is specified, leave this parameter blank.
image_url	String	Custom image URL selected by an algorithm.

Table 5-29 code_tree

Parameter	Type	Description
name	String	Name of the current directory in the algorithm directory tree.
children	Object	Subfiles and subdirectories in the current directory of the algorithm directory tree.

Table 5-30 resource_requirements

Parameter	Type	Description
key	String	Resource constraint. The options are as follows: <ul style="list-style-type: none"> Resource type (flavor_type). The value can be CPU, GPU, or Ascend (tag: hc, hk,fcs_super). Whether to support multi-card training (device_distributed_mode). The value can be multiple or singular. Indicates whether distributed training is supported (host_distributed_mode). The value can be multiple or singular.
value	Array of strings	Value of the resource constraint key.
operator	String	Relationship between keys and values. Currently, only in is supported. For example: flavor_type in [CPU,GPU].

Table 5-31 advanced_config

Parameter	Type	Description
auto_search	auto_search object	Hyperparameter search policy.

Table 5-32 auto_search

Parameter	Type	Description
skip_search_params	String	Hyperparameter parameters that need to be skipped.
reward_attrs	Array of reward_attrs objects	List of search metrics.
search_params	Array of search_params objects	Search parameters.
algo_configs	Array of algo_configs objects	Search algorithm configurations.

Table 5-33 reward_attrs

Parameter	Type	Description
name	String	Metric name.
mode	String	Search direction. <ul style="list-style-type: none"> • max: A larger metric value indicates better performance. • min: A smaller metric value indicates better performance.
regex	String	Regular expression of a metric.

Table 5-34 search_params

Parameter	Type	Description
name	String	Hyperparameter name.
param_type	String	Parameter type <ul style="list-style-type: none"> • If continuous is specified, the hyperparameter is of the continuous type. When an algorithm is used in a training job, continuous hyperparameters are displayed as text boxes on the console. - discrete: The hyperparameter is of the discrete type. When an algorithm is used for training jobs, discrete hyperparameters are displayed as a drop-down list box on the console.
lower_bound	String	Lower bound of the hyperparameter.
upper_bound	String	Upper bound of the hyperparameter.
discrete_points_num	String	Number of discrete points of a continuous hyperparameter.
discrete_values	String	List of discrete hyperparameter values.

Table 5-35 algo_configs

Parameter	Type	Description
name	String	Name of the search algorithm.
params	Array of AutoSearchAlgorithmConfigParameter objects	Search algorithm parameters.

Table 5-36 AutoSearchAlgoConfigParameter

Parameter	Type	Description
key	String	Parameter key.
value	String	Parameter value.
type	String	Parameter type.

Example Requests

The following is an example of how to create an algorithm whose name is **TestModelArtsalgorithm** and description is **This is a ModelArts algorithm**.

POST https://endpoint/v2/{project_id}/algorithms

```
{
  "metadata": {
    "name": "TestModelArtsalgorithm",
    "description": "This is a ModelArts algorithm"
  },
  "job_config": {
    "code_dir": "/algo-test/pytorch/work1/code/",
    "boot_file": "/algo-test/pytorch/work1/code/test-pytorch.py",
    "parameters": [ {
      "name": "test-parameter",
      "value": "10",
      "constraint": {
        "type": "String",
        "editable": true,
        "required": false,
        "sensitive": false,
        "valid_type": "None",
        "valid_range": [ ]
      }
    }
  ],
  "parameters_customization": true,
  "inputs": [ {
    "name": "data_url",
    "description": "data source."
  } ],
  "outputs": [ {
    "name": "train_url",
    "description": "model output."
  } ],
  "engine": {
    "engine_name": "PyTorch",
    "engine_version": "PyTorch-1.3.0-python3.6"
  }
}
```

Example Responses

Status code: 201

ok

```
{
  "metadata": {
    "id": "2e5451fe-913f-4492-821a-2981031382f7",
    "name": "TestModelArtsalgorithm",
    "description": "This is a ModelArts algorithm",
```

```

"create_time": 1636600721742,
"workspace_id": "0",
"ai_project": "default-ai-project",
"user_name": "",
"domain_id": "xxxxxxxxxxxxxxxxxxxxxxxxxxxx",
"source": "custom",
"api_version": "",
"is_valid": true,
"state": "",
"size": 4791,
"tags": null,
"attr_list": null,
"version_num": 0,
"update_time": 0
},
"job_config": {
"code_dir": "/algo-test/pytorch/work1/code/",
"boot_file": "/algo-test/pytorch/work1/code/test-pytorch.py",
"command": "",
"parameters": [ {
"name": "test-parameter",
"description": "",
"i18n_description": null,
"value": "10",
"constraint": {
"type": "String",
"editable": true,
"required": false,
"sensitive": false,
"valid_type": "None",
"valid_range": [ ]
}
}
],
"parameters_customization": true,
"inputs": [ {
"name": "data_url",
"description": "name to translate"
}
],
"outputs": [ {
"name": "train_url",
"description": "name to translate"
}
],
"engine": {
"engine_id": "pytorch-cp36-1.3.0",
"engine_name": "PyTorch",
"engine_version": "PyTorch-1.3.0-python3.6"
},
"code_tree": {
"name": "code/",
"children": [ {
"name": "test-pytorch.py"
}
]
}
},
"resource_requirements": null,
"advanced_config": { }
}

```

Status Codes

Status Code	Description
201	ok

Error Codes

See [Error Codes](#).

5.2 Querying the Algorithm List

Function

This API is used to query the algorithm list.

URI

GET /v2/{project_id}/algorithms

Table 5-37 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .

Table 5-38 Query Parameters

Parameter	Mandatory	Type	Description
offset	No	Integer	Offset for querying algorithms. The minimum value is 0 . For example, if this parameter is set to 1 , the query starts from the second one.
limit	No	Integer	Limit of algorithms to be queried. The value ranges from 1 to 50.
sort_by	No	String	Metric for sorting algorithms to be queried. create_time is used by default.
order	No	String	Order of queried algorithms. The default value is desc , indicating that queried algorithms are sorted in descending order. You can also select asc to sort the records in ascending order.
group_by	No	String	Condition for grouping the algorithms to be queried.

Parameter	Mandatory	Type	Description
searches	No	String	Filter criteria for algorithms to be queried, for example, fuzzy match by the algorithm name.
workspace_id	No	String	Workspace ID

Request Parameters

None

Response Parameters

Status code: 200

Table 5-39 Response body parameters

Parameter	Type	Description
total	Integer	Total number of queried algorithms of the current user.
count	Integer	Total number of algorithms that meet the search criteria of the current user.
limit	Integer	Maximum number of queried algorithms of the current user.
offset	Integer	Offset of all algorithm queries of the current user.
sort_by	String	Fields used to sort queried algorithms of the current user.
order	String	Sorting mode of queried algorithms of the current user. The default value is desc , indicating the descending order. You can also set this parameter to asc , indicating the ascending order.
group_by	String	Grouping mode of queried algorithms of the current user.
items	Array of AlgorithmResponse objects	Details about all algorithms that meet the search criteria of the current user.

Table 5-40 AlgorithmResponse

Parameter	Type	Description
metadata	metadata object	Algorithm metadata, which describes basic algorithm information.
job_config	job_config object	Algorithm configuration, such as the boot file.
resource_requirements	Array of resource_requirements objects	Algorithm resource constraint. This parameter is optional. After this parameter is set, the console filters available public resource pools when the algorithm is used in training jobs.
advanced_config	advanced_config object	Advanced algorithm policy: <ul style="list-style-type: none"> • auto_search

Table 5-41 metadata

Parameter	Type	Description
id	Integer	Algorithm UUID. You do not need to set this parameter when creating an algorithm.
name	String	Algorithm name. The value must contain 1 to 64 characters consisting of only digits, letters, underscores (_), and hyphens (-).
description	String	Description of the algorithm. The value is a string of [0, 256] characters. The default value is NULL.
workspace_id	String	Workspace where a specified algorithm is located. The default value is 0 . The value 0 indicates the default workspace.
ai_project	String	AI project to which a specified algorithm belongs. The default value is default-ai-project . The AI project has been brought offline. Ignore it.
user_name	String	Username.
domain_id	String	Domain ID of a user.
source	String	Algorithm source.
api_version	String	Algorithm API version, which identifies the old and new ones.
is_valid	String	Algorithm availability.
state	String	Algorithm state.

Parameter	Type	Description
tags	Array of Map<String,String> objects	Algorithm tags.
attr_list	Array of strings	Algorithm attribute list.
version_num	Integer	Number of algorithm versions. The default value is 0 .
size	Integer	Algorithm size.
create_time	Long	Timestamp when the algorithm is created.
update_time	Long	Timestamp when the algorithm is updated.

Table 5-42 job_config

Parameter	Type	Description
code_dir	String	Algorithm code directory, for example, /usr/app/ . This parameter must be used together with boot_file .
boot_file	String	Code boot file of the algorithm, which must be stored in the code directory, for example, /usr/app/boot.py . This parameter must be used with code_dir .
command	String	Container startup command of a custom image algorithm.
parameters	Array of Parameter objects	Running parameter of an algorithm.
inputs	Array of inputs objects	Data input of an algorithm.
outputs	Array of outputs objects	Data output of an algorithm.
engine	engine object	Algorithm engine.
code_tree	Array of code_tree objects	Algorithm directory tree
parameters_customization	Boolean	Whether the algorithm allows hyperparameter customization during training job creation.

Table 5-43 Parameter

Parameter	Type	Description
name	String	Parameter name.
value	String	Parameter value.
description	String	Parameter description.
constraint	constraint object	Parameter constraint.
i18n_description	i18n_description object	Internationalization description.

Table 5-44 constraint

Parameter	Type	Description
type	String	Parameter type.
editable	Boolean	Whether the parameter is editable.
required	Boolean	Whether the parameter is mandatory.
sensitive	Boolean	Whether the parameter is sensitive This function is not implemented currently.
valid_type	String	Valid type.
valid_range	Array of strings	Valid range.

Table 5-45 i18n_description

Parameter	Type	Description
language	String	Language.
description	String	Description.

Table 5-46 inputs

Parameter	Type	Description
name	String	Name of the data input channel.
description	String	Description of the data input channel.

Parameter	Type	Description
remote_constraints	Array of remote_constraints objects	Data input constraint.

Table 5-47 remote_constraints

Parameter	Type	Description
data_type	String	Data input type, including the data storage location and dataset.
attributes	Array of Map<String,String> objects	Attributes if a dataset is used as the data input. Options: <ul style="list-style-type: none"> • data_format: Data format • data_segmentation: Data segmentation • dataset_type: Labeling type

Table 5-48 outputs

Parameter	Type	Description
name	String	Name of the data output channel.
description	String	Description of the data output channel.

Table 5-49 engine

Parameter	Type	Description
engine_id	String	Engine ID selected for an algorithm.
engine_name	String	Engine version name selected for an algorithm. If engine_id is specified, leave this parameter blank.
engine_version	String	Engine version name selected for an algorithm. If engine_id is specified, leave this parameter blank.
image_url	String	Custom image URL selected by an algorithm.

Table 5-50 code_tree

Parameter	Type	Description
name	String	Name of the current directory in the algorithm directory tree.
children	Object	Subfiles and subdirectories in the current directory of the algorithm directory tree.

Table 5-51 resource_requirements

Parameter	Type	Description
key	String	Resource constraint. The options are as follows: <ul style="list-style-type: none"> Resource type (flavor_type). The value can be CPU, GPU, or Ascend (tag: hc, hk,fcs_super). Whether to support multi-card training (device_distributed_mode). The value can be multiple or singular. Indicates whether distributed training is supported (host_distributed_mode). The value can be multiple or singular.
value	Array of strings	Value of the resource constraint key.
operator	String	Relationship between keys and values. Currently, only in is supported. For example: flavor_type in [CPU,GPU].

Table 5-52 advanced_config

Parameter	Type	Description
auto_search	auto_search object	Hyperparameter search policy.

Table 5-53 auto_search

Parameter	Type	Description
skip_search_params	String	Hyperparameter parameters that need to be skipped.
reward_attrs	Array of reward_attrs objects	List of search metrics.

Parameter	Type	Description
search_params	Array of search_params objects	Search parameters.
algo_configs	Array of algo_configs objects	Search algorithm configurations.

Table 5-54 reward_attrs

Parameter	Type	Description
name	String	Metric name.
mode	String	Search direction. <ul style="list-style-type: none"> • max: A larger metric value indicates better performance. • min: A smaller metric value indicates better performance.
regex	String	Regular expression of a metric.

Table 5-55 search_params

Parameter	Type	Description
name	String	Hyperparameter name.
param_type	String	Parameter type <ul style="list-style-type: none"> • If continuous is specified, the hyperparameter is of the continuous type. When an algorithm is used in a training job, continuous hyperparameters are displayed as text boxes on the console. - discrete: The hyperparameter is of the discrete type. When an algorithm is used for training jobs, discrete hyperparameters are displayed as a drop-down list box on the console.
lower_bound	String	Lower bound of the hyperparameter.
upper_bound	String	Upper bound of the hyperparameter.
discrete_points_num	String	Number of discrete points of a continuous hyperparameter.
discrete_values	String	List of discrete hyperparameter values.

Table 5-56 algo_configs

Parameter	Type	Description
name	String	Name of the search algorithm.
params	Array of AutoSearchAlgoConfigParameter objects	Search algorithm parameters.

Table 5-57 AutoSearchAlgoConfigParameter

Parameter	Type	Description
key	String	Parameter key.
value	String	Parameter value.
type	String	Parameter type.

Example Requests

The following shows how to query all algorithms in jobs whose names contain **TestModelArtsAlgorithm** and maximum number of algorithms is set to **1**.

```
GET https://endpoint/v2/{project_id}/algorithms?limit=1&searches=name%3ATestModelArtsAlgorithm
```

Example Responses

Status code: 200

ok

```
{
  "total": 1,
  "count": 1,
  "limit": 1,
  "offset": 0,
  "sort_by": "create_time",
  "order": "desc",
  "group_by": "",
  "items": [ {
    "metadata": {
      "id": "2e5451fe-913f-4492-821a-2981031382f7",
      "name": "TestModelArtsAlgorithm",
      "description": "This is a ModelArts algorithm",
      "create_time": 1636600721742,
      "workspace_id": "0",
      "ai_project": "default-ai-project",
      "user_name": "",
      "domain_id": "xxxxxxxxxxxxxxxxxxxxxxxxxxxx",
      "source": "custom",
      "api_version": "",
      "is_valid": true,
      "state": "",
      "size": 4791,
      "tags": null,

```

```

"attr_list" : null,
"version_num" : 0,
"update_time" : 0
},
"job_config" : {
"code_dir" : "/algo-test/pytorch/work1/code/",
"boot_file" : "/algo-test/pytorch/work1/code/test-pytorch.py",
"command" : "",
"parameters" : [ {
"name" : "test-parameter",
"description" : "",
"i18n_description" : null,
"value" : "10",
"constraint" : {
"type" : "String",
"editable" : true,
"required" : false,
"sensitive" : false,
"valid_type" : "None",
"valid_range" : [ ]
}
}
],
"parameters_customization" : true,
"inputs" : [ {
"name" : "data_url",
"description" : "name to translate"
}
],
"outputs" : [ {
"name" : "train_url",
"description" : "name to translate"
}
],
"engine" : {
"engine_id" : "pytorch-cp36-1.3.0",
"engine_name" : "PyTorch",
"engine_version" : "PyTorch-1.3.0-python3.6"
},
"code_tree" : {
"name" : "code/",
"children" : [ {
"name" : "test-pytorch.py"
}
]
}
},
"resource_requirements" : null,
"advanced_config" : { }
}
}

```

Status Codes

Status Code	Description
200	ok

Error Codes

See [Error Codes](#).

5.3 Querying Algorithm Details

Function

This API is used to query a specified algorithm based on the algorithm ID.

URI

GET /v2/{project_id}/algorithms/{algorithm_id}

Table 5-58 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .
algorithm_id	Yes	String	Algorithm ID.

Request Parameters

None

Response Parameters

Status code: 200

Table 5-59 Response body parameters

Parameter	Type	Description
metadata	metadata object	Algorithm metadata, which describes basic algorithm information.
job_config	job_config object	Algorithm configuration, such as the boot file.
resource_requirements	Array of resource_requirements objects	Algorithm resource constraint. This parameter is optional. After this parameter is set, the console filters available public resource pools when the algorithm is used in training jobs.
advanced_config	advanced_config object	Advanced algorithm policy: <ul style="list-style-type: none"> • auto_search

Table 5-60 metadata

Parameter	Type	Description
id	Integer	Algorithm UUID. You do not need to set this parameter when creating an algorithm.
name	String	Algorithm name. The value must contain 1 to 64 characters consisting of only digits, letters, underscores (_), and hyphens (-).
description	String	Description of the algorithm. The value is a string of [0, 256] characters. The default value is NULL.
workspace_id	String	Workspace where a specified algorithm is located. The default value is 0 . The value 0 indicates the default workspace.
ai_project	String	AI project to which a specified algorithm belongs. The default value is default-ai-project . The AI project has been brought offline. Ignore it.
user_name	String	Username.
domain_id	String	Domain ID of a user.
source	String	Algorithm source.
api_version	String	Algorithm API version, which identifies the old and new ones.
is_valid	String	Algorithm availability.
state	String	Algorithm state.
tags	Array of Map<String,String> objects	Algorithm tags.
attr_list	Array of strings	Algorithm attribute list.
version_num	Integer	Number of algorithm versions. The default value is 0 .
size	Integer	Algorithm size.
create_time	Long	Timestamp when the algorithm is created.
update_time	Long	Timestamp when the algorithm is updated.

Table 5-61 job_config

Parameter	Type	Description
code_dir	String	Algorithm code directory, for example, <code>/usr/app/</code> . This parameter must be used together with <code>boot_file</code> .
boot_file	String	Code boot file of the algorithm, which must be stored in the code directory, for example, <code>/usr/app/boot.py</code> . This parameter must be used with <code>code_dir</code> .
command	String	Container startup command of a custom image algorithm.
parameters	Array of Parameter objects	Running parameter of an algorithm.
inputs	Array of inputs objects	Data input of an algorithm.
outputs	Array of outputs objects	Data output of an algorithm.
engine	engine object	Algorithm engine.
code_tree	Array of code_tree objects	Algorithm directory tree
parameters_customization	Boolean	Whether the algorithm allows hyperparameter customization during training job creation.

Table 5-62 Parameter

Parameter	Type	Description
name	String	Parameter name.
value	String	Parameter value.
description	String	Parameter description.
constraint	constraint object	Parameter constraint.
internationalization	internationalization object	Internationalization description.

Table 5-63 constraint

Parameter	Type	Description
type	String	Parameter type.
editable	Boolean	Whether the parameter is editable.
required	Boolean	Whether the parameter is mandatory.
sensitive	Boolean	Whether the parameter is sensitive This function is not implemented currently.
valid_type	String	Valid type.
valid_range	Array of strings	Valid range.

Table 5-64 i18n_description

Parameter	Type	Description
language	String	Language.
description	String	Description.

Table 5-65 inputs

Parameter	Type	Description
name	String	Name of the data input channel.
description	String	Description of the data input channel.
remote_constraints	Array of remote_constraints objects	Data input constraint.

Table 5-66 remote_constraints

Parameter	Type	Description
data_type	String	Data input type, including the data storage location and dataset.
attributes	Array of Map<String,String> objects	Attributes if a dataset is used as the data input. Options: <ul style="list-style-type: none"> • data_format: Data format • data_segmentation: Data segmentation • dataset_type: Labeling type

Table 5-67 outputs

Parameter	Type	Description
name	String	Name of the data output channel.
description	String	Description of the data output channel.

Table 5-68 engine

Parameter	Type	Description
engine_id	String	Engine ID selected for an algorithm.
engine_name	String	Engine version name selected for an algorithm. If engine_id is specified, leave this parameter blank.
engine_version	String	Engine version name selected for an algorithm. If engine_id is specified, leave this parameter blank.
image_url	String	Custom image URL selected by an algorithm.

Table 5-69 code_tree

Parameter	Type	Description
name	String	Name of the current directory in the algorithm directory tree.
children	Object	Subfiles and subdirectories in the current directory of the algorithm directory tree.

Table 5-70 resource_requirements

Parameter	Type	Description
key	String	Resource constraint. The options are as follows: <ul style="list-style-type: none"> Resource type (flavor_type). The value can be CPU, GPU, or Ascend (tag: hc, hk,fcs_super). Whether to support multi-card training (device_distributed_mode). The value can be multiple or singular. Indicates whether distributed training is supported (host_distributed_mode). The value can be multiple or singular.

Parameter	Type	Description
value	Array of strings	Value of the resource constraint key.
operator	String	Relationship between keys and values. Currently, only in is supported. For example: flavor_type in [CPU,GPU].

Table 5-71 advanced_config

Parameter	Type	Description
auto_search	auto_search object	Hyperparameter search policy.

Table 5-72 auto_search

Parameter	Type	Description
skip_search_params	String	Hyperparameter parameters that need to be skipped.
reward_attrs	Array of reward_attrs objects	List of search metrics.
search_params	Array of search_params objects	Search parameters.
algo_configs	Array of algo_configs objects	Search algorithm configurations.

Table 5-73 reward_attrs

Parameter	Type	Description
name	String	Metric name.
mode	String	Search direction. <ul style="list-style-type: none"> • max: A larger metric value indicates better performance. • min: A smaller metric value indicates better performance.
regex	String	Regular expression of a metric.

Table 5-74 search_params

Parameter	Type	Description
name	String	Hyperparameter name.
param_type	String	Parameter type <ul style="list-style-type: none"> If continuous is specified, the hyperparameter is of the continuous type. When an algorithm is used in a training job, continuous hyperparameters are displayed as text boxes on the console. - discrete: The hyperparameter is of the discrete type. When an algorithm is used for training jobs, discrete hyperparameters are displayed as a drop-down list box on the console.
lower_bound	String	Lower bound of the hyperparameter.
upper_bound	String	Upper bound of the hyperparameter.
discrete_points_num	String	Number of discrete points of a continuous hyperparameter.
discrete_values	String	List of discrete hyperparameter values.

Table 5-75 algo_configs

Parameter	Type	Description
name	String	Name of the search algorithm.
params	Array of AutoSearchAlgorithmConfigParameter objects	Search algorithm parameters.

Table 5-76 AutoSearchAlgorithmConfigParameter

Parameter	Type	Description
key	String	Parameter key.
value	String	Parameter value.
type	String	Parameter type.

Example Requests

The following shows how to query the algorithm whose UUID is **2e5451fe-913f-4492-821a-2981031382f7**.

```
GET https://endpoint/v2/{project_id}/algorithms/2e5451fe-913f-4492-821a-2981031382f7
```

Example Responses

Status code: 200

ok

```
{
  "metadata": {
    "id": "2e5451fe-913f-4492-821a-2981031382f7",
    "name": "TestModelArtsalgorithm",
    "description": "This is a ModelArts algorithm",
    "create_time": 1636600721742,
    "workspace_id": "0",
    "ai_project": "default-ai-project",
    "user_name": "",
    "domain_id": "xxxxxxxxxxxxxxxxxxxxxxxxxxxx",
    "source": "custom",
    "api_version": "",
    "is_valid": true,
    "state": "",
    "size": 4791,
    "tags": null,
    "attr_list": null,
    "version_num": 0,
    "update_time": 0
  },
  "job_config": {
    "code_dir": "/algo-test/pytorch/work1/code/",
    "boot_file": "/algo-test/pytorch/work1/code/test-pytorch.py",
    "command": "",
    "parameters": [ {
      "name": "test-parameter",
      "description": "",
      "i18n_description": null,
      "value": "10",
      "constraint": {
        "type": "String",
        "editable": true,
        "required": false,
        "sensitive": false,
        "valid_type": "None",
        "valid_range": [ ]
      }
    }
  ],
  "parameters_customization": true,
  "inputs": [ {
    "name": "data_url",
    "description": "name to translate"
  } ],
  "outputs": [ {
    "name": "train_url",
    "description": "name to translate"
  } ],
  "engine": {
    "engine_id": "pytorch-cp36-1.3.0",
    "engine_name": "PyTorch",
    "engine_version": "PyTorch-1.3.0-python3.6"
  },
  "code_tree": {
    "name": "code/",
    "children": [ {
```

```

    "name" : "test-pytorch.py"
  } ]
}
},
"resource_requirements" : null,
"advanced_config" : { }
}

```

Status Codes

Status Code	Description
200	ok

Error Codes

See [Error Codes](#).

5.4 Modifying an Algorithm

Function

This API is used to modify an algorithm.

URI

PUT /v2/{project_id}/algorithms/{algorithm_id}

Table 5-77 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .
algorithm_id	Yes	String	Algorithm ID.

Request Parameters

Table 5-78 Request body parameters

Parameter	Mandatory	Type	Description
metadata	No	metadata object	Algorithm metadata, which describes basic algorithm information.

Parameter	Mandatory	Type	Description
job_config	No	job_config object	Algorithm configuration, such as the boot file.
resource_requirements	No	Array of resource_requirements objects	Algorithm resource constraints. This parameter is optional. After this parameter is set, the console filters available public resource pools when the algorithm is used in training jobs.
advanced_config	No	advanced_config object	Advanced algorithm policy: <ul style="list-style-type: none"> • auto_search

Table 5-79 metadata

Parameter	Mandatory	Type	Description
id	No	Integer	Algorithm UUID. You do not need to set this parameter when creating an algorithm.
name	Yes	String	Algorithm name. The value must contain 1 to 64 characters consisting of only digits, letters, underscores (_), and hyphens (-).
description	No	String	Description of the algorithm. The value is a string of [0, 256] characters. The default value is NULL.
workspace_id	No	String	Workspace where a specified algorithm is located. The default value is 0 . The value 0 indicates the default workspace.
ai_project	No	String	AI project to which a specified algorithm belongs. The default value is default-ai-project . The AI project has been brought offline. Ignore it.

Table 5-80 job_config

Parameter	Mandatory	Type	Description
code_dir	No	String	Algorithm code directory, for example, <code>/usr/app/</code> . This parameter must be used together with boot_file .
boot_file	No	String	Code boot file of the algorithm, which needs to be stored in the code directory, for example, <code>/usr/app/boot.py</code> . This parameter must be used together with code_dir .
command	No	String	Container startup command of a custom image algorithm.
parameters	No	Array of Parameter objects	Running parameter of an algorithm.
inputs	No	Array of inputs objects	Data input of an algorithm.
outputs	No	Array of outputs objects	Data output of an algorithm.
engine	No	engine object	Algorithm engine.
parameters_customization	No	Boolean	Whether the algorithm allows hyperparameter customization during training job creation.

Table 5-81 Parameter

Parameter	Mandatory	Type	Description
name	No	String	Parameter name.
value	No	String	Parameter value.
description	No	String	Parameter description.
constraint	No	constraint object	Parameter constraint.
internationalization	No	internationalization object	Internationalization description.

Table 5-82 constraint

Parameter	Mandatory	Type	Description
type	No	String	Parameter type.
editable	No	Boolean	Whether the parameter is editable.
required	No	Boolean	Whether the parameter is mandatory.
sensitive	No	Boolean	Whether the parameter is sensitive This function is not implemented currently.
valid_type	No	String	Valid type.
valid_range	No	Array of strings	Valid range.

Table 5-83 i18n_description

Parameter	Mandatory	Type	Description
language	No	String	Language.
description	No	String	Description.

Table 5-84 inputs

Parameter	Mandatory	Type	Description
name	Yes	String	Name of the data input channel.
description	No	String	Description of the data input channel.
remote_constraints	No	Array of remote_constraints objects	Data input constraint.

Table 5-85 remote_constraints

Parameter	Mandatory	Type	Description
data_type	No	String	Data input type. The data storage location (OBS) and ModelArts dataset are supported.

Parameter	Mandatory	Type	Description
attributes	No	Array of Map<String,String> objects	Attributes if a dataset is used as the data input. Options: <ul style="list-style-type: none"> • data_format: Data format • data_segmentation: Data segmentation • dataset_type: Labeling type

Table 5-86 outputs

Parameter	Mandatory	Type	Description
name	Yes	String	Name of the data output channel.
description	No	String	Description of the data output channel.

Table 5-87 engine

Parameter	Mandatory	Type	Description
engine_id	No	String	Engine ID selected for an algorithm.
engine_name	No	String	Engine version name selected for an algorithm. If engine_id is specified, leave this parameter blank.
engine_version	No	String	Engine version name selected for an algorithm. If engine_id is specified, leave this parameter blank.
image_url	No	String	Custom image URL selected by an algorithm.

Table 5-88 resource_requirements

Parameter	Mandatory	Type	Description
key	No	String	Resource constraint. The options are as follows: <ul style="list-style-type: none"> flavor_type indicates the resource type. The value can be CPU, GPU, or Ascend (tag: hc, hk,fcs_super). device_distributed_mode: indicates whether to support multi-card training. The value can be multiple (supported) or singular (not supported). host_distributed_mode: indicates whether distributed training is supported. The value can be multiple (supported) or singular (not supported).
values	No	Array of strings	Value of the resource constraint key.
operator	No	String	Relationship between keys and values. Currently, only in is supported. For example: flavor_type in [CPU,GPU].

Table 5-89 advanced_config

Parameter	Mandatory	Type	Description
auto_search	No	auto_search object	Hyperparameter search policy.

Table 5-90 auto_search

Parameter	Mandatory	Type	Description
skip_search_parameters	No	String	Hyperparameter parameters that need to be skipped.
reward_attrs	No	Array of reward_attrs objects	List of search metrics.

Parameter	Mandatory	Type	Description
search_params	No	Array of search_params objects	Search parameters.
algo_configs	No	Array of algo_configs objects	Search algorithm configurations.

Table 5-91 reward_attrs

Parameter	Mandatory	Type	Description
name	No	String	Metric name.
mode	No	String	Search direction. <ul style="list-style-type: none"> • max: A larger metric value indicates better performance. • min: A smaller metric value indicates better performance.
regex	No	String	Regular expression of a metric.

Table 5-92 search_params

Parameter	Mandatory	Type	Description
name	No	String	Hyperparameter name.
param_type	No	String	Parameter type <ul style="list-style-type: none"> • If continuous is specified, the hyperparameter is of the continuous type. When an algorithm is used in a training job, continuous hyperparameters are displayed as text boxes on the console. - discrete: The hyperparameter is of the discrete type. When an algorithm is used for training jobs, discrete hyperparameters are displayed as a drop-down list box on the console.

Parameter	Mandatory	Type	Description
lower_bound	No	String	Lower bound of the hyperparameter.
upper_bound	No	String	Upper bound of the hyperparameter.
discrete_points_num	No	String	Number of discrete points of a continuous hyperparameter.
discrete_values	No	String	List of discrete hyperparameter values.

Table 5-93 algo_configs

Parameter	Mandatory	Type	Description
name	No	String	Name of the search algorithm.
params	No	Array of AutoSearchAlgoConfigParameter objects	Search algorithm parameters.

Table 5-94 AutoSearchAlgoConfigParameter

Parameter	Mandatory	Type	Description
key	No	String	Parameter key.
value	No	String	Parameter value.
type	No	String	Parameter type.

Response Parameters

Status code: 201

Table 5-95 Response body parameters

Parameter	Type	Description
metadata	metadata object	Algorithm metadata, which describes basic algorithm information.
job_config	job_config object	Algorithm configuration, such as the boot file.

Parameter	Type	Description
resource_requirements	Array of resource_requirements objects	Algorithm resource constraint. This parameter is optional. After this parameter is set, the console filters available public resource pools when the algorithm is used in training jobs.
advanced_config	advanced_config object	Advanced algorithm policy: <ul style="list-style-type: none"> • auto_search

Table 5-96 metadata

Parameter	Type	Description
id	Integer	Algorithm UUID. You do not need to set this parameter when creating an algorithm.
name	String	Algorithm name. The value must contain 1 to 64 characters consisting of only digits, letters, underscores (_), and hyphens (-).
description	String	Description of the algorithm. The value is a string of [0, 256] characters. The default value is NULL.
workspace_id	String	Workspace where a specified algorithm is located. The default value is 0 . The value 0 indicates the default workspace.
ai_project	String	AI project to which a specified algorithm belongs. The default value is default-ai-project . The AI project has been brought offline. Ignore it.
user_name	String	Username.
domain_id	String	Domain ID of a user.
source	String	Algorithm source.
api_version	String	Algorithm API version, which identifies the old and new ones.
is_valid	String	Algorithm availability.
state	String	Algorithm state.
tags	Array of Map<String,String> objects	Algorithm tags.
attr_list	Array of strings	Algorithm attribute list.

Parameter	Type	Description
version_num	Integer	Number of algorithm versions. The default value is 0 .
size	Integer	Algorithm size.
create_time	Long	Timestamp when the algorithm is created.
update_time	Long	Timestamp when the algorithm is updated.

Table 5-97 job_config

Parameter	Type	Description
code_dir	String	Algorithm code directory, for example, <code>/usr/app/</code> . This parameter must be used together with boot_file .
boot_file	String	Code boot file of the algorithm, which must be stored in the code directory, for example, <code>/usr/app/boot.py</code> . This parameter must be used with code_dir .
command	String	Container startup command of a custom image algorithm.
parameters	Array of Parameter objects	Running parameter of an algorithm.
inputs	Array of inputs objects	Data input of an algorithm.
outputs	Array of outputs objects	Data output of an algorithm.
engine	engine object	Algorithm engine.
code_tree	Array of code_tree objects	Algorithm directory tree
parameters_customization	Boolean	Whether the algorithm allows hyperparameter customization during training job creation.

Table 5-98 Parameter

Parameter	Type	Description
name	String	Parameter name.

Parameter	Type	Description
value	String	Parameter value.
description	String	Parameter description.
constraint	constraint object	Parameter constraint.
i18n_description	i18n_description object	Internationalization description.

Table 5-99 constraint

Parameter	Type	Description
type	String	Parameter type.
editable	Boolean	Whether the parameter is editable.
required	Boolean	Whether the parameter is mandatory.
sensitive	Boolean	Whether the parameter is sensitive This function is not implemented currently.
valid_type	String	Valid type.
valid_range	Array of strings	Valid range.

Table 5-100 i18n_description

Parameter	Type	Description
language	String	Language.
description	String	Description.

Table 5-101 inputs

Parameter	Type	Description
name	String	Name of the data input channel.
description	String	Description of the data input channel.
remote_constraints	Array of remote_constraints objects	Data input constraint.

Table 5-102 remote_constraints

Parameter	Type	Description
data_type	String	Data input type, including the data storage location and dataset.
attributes	Array of Map<String,String> objects	Attributes if a dataset is used as the data input. Options: <ul style="list-style-type: none"> • data_format: Data format • data_segmentation: Data segmentation • dataset_type: Labeling type

Table 5-103 outputs

Parameter	Type	Description
name	String	Name of the data output channel.
description	String	Description of the data output channel.

Table 5-104 engine

Parameter	Type	Description
engine_id	String	Engine ID selected for an algorithm.
engine_name	String	Engine version name selected for an algorithm. If engine_id is specified, leave this parameter blank.
engine_version	String	Engine version name selected for an algorithm. If engine_id is specified, leave this parameter blank.
image_url	String	Custom image URL selected by an algorithm.

Table 5-105 code_tree

Parameter	Type	Description
name	String	Name of the current directory in the algorithm directory tree.
children	Object	Subfiles and subdirectories in the current directory of the algorithm directory tree.

Table 5-106 resource_requirements

Parameter	Type	Description
key	String	Resource constraint. The options are as follows: <ul style="list-style-type: none"> Resource type (flavor_type). The value can be CPU, GPU, or Ascend (tag: hc, hk,fcs_super). Whether to support multi-card training (device_distributed_mode). The value can be multiple or singular. Indicates whether distributed training is supported (host_distributed_mode). The value can be multiple or singular.
value	Array of strings	Value of the resource constraint key.
operator	String	Relationship between keys and values. Currently, only in is supported. For example: flavor_type in [CPU,GPU].

Table 5-107 advanced_config

Parameter	Type	Description
auto_search	auto_search object	Hyperparameter search policy.

Table 5-108 auto_search

Parameter	Type	Description
skip_search_params	String	Hyperparameter parameters that need to be skipped.
reward_attrs	Array of reward_attrs objects	List of search metrics.
search_params	Array of search_params objects	Search parameters.
algo_configs	Array of algo_configs objects	Search algorithm configurations.

Table 5-109 reward_attrs

Parameter	Type	Description
name	String	Metric name.
mode	String	Search direction. <ul style="list-style-type: none"> • max: A larger metric value indicates better performance. • min: A smaller metric value indicates better performance.
regex	String	Regular expression of a metric.

Table 5-110 search_params

Parameter	Type	Description
name	String	Hyperparameter name.
param_type	String	Parameter type <ul style="list-style-type: none"> • If continuous is specified, the hyperparameter is of the continuous type. When an algorithm is used in a training job, continuous hyperparameters are displayed as text boxes on the console. - discrete: The hyperparameter is of the discrete type. When an algorithm is used for training jobs, discrete hyperparameters are displayed as a drop-down list box on the console.
lower_bound	String	Lower bound of the hyperparameter.
upper_bound	String	Upper bound of the hyperparameter.
discrete_points_num	String	Number of discrete points of a continuous hyperparameter.
discrete_values	String	List of discrete hyperparameter values.

Table 5-111 algo_configs

Parameter	Type	Description
name	String	Name of the search algorithm.
params	Array of AutoSearchAlgorithmConfigParameter objects	Search algorithm parameters.

Table 5-112 AutoSearchAlgoConfigParameter

Parameter	Type	Description
key	String	Parameter key.
value	String	Parameter value.
type	String	Parameter type.

Example Requests

The following shows how to modify the algorithm whose UUID is **2e5451fe-913f-4492-821a-2981031382f7**.

PUT https://endpoint/v2/{project_id}/algorithms/2e5451fe-913f-4492-821a-2981031382f7

```
{
  "metadata": {
    "name": "TestModelArtsAlgorithm",
    "description": "This is a ModelArts algorithm modified"
  },
  "job_config": {
    "code_dir": "/algo-test/pytorch/work1/code/",
    "boot_file": "/algo-test/pytorch/work1/code/test-pytorch.py",
    "parameters": [ {
      "name": "test-parameter",
      "value": "10",
      "constraint": {
        "type": "String",
        "editable": true,
        "required": false,
        "sensitive": false,
        "valid_type": "None",
        "valid_range": [ ]
      }
    }
  ],
  "parameters_customization": true,
  "inputs": [ {
    "name": "data_url",
    "description": "data source."
  } ],
  "outputs": [ {
    "name": "train_url",
    "description": "model output."
  } ],
  "engine": {
    "engine_name": "PyTorch",
    "engine_version": "PyTorch-1.3.0-python3.6"
  }
}
```

Example Responses

Status code: 201

ok

```
{
  "metadata": {
    "id": "2e5451fe-913f-4492-821a-2981031382f7",
    "name": "TestModelArtsAlgorithm",
    "description": "This is a ModelArts algorithm modified",
```

```

"create_time": 1636600721742,
"workspace_id": "0",
"ai_project": "default-ai-project",
"user_name": "",
"domain_id": "xxxxxxxxxxxxxxxxxxxxxxxxxxxx",
"source": "custom",
"api_version": "",
"is_valid": true,
"state": "",
"size": 4791,
"tags": null,
"attr_list": null,
"version_num": 0,
"update_time": 0
},
"job_config": {
"code_dir": "/algo-test/pytorch/work1/code/",
"boot_file": "/algo-test/pytorch/work1/code/test-pytorch.py",
"command": "",
"parameters": [ {
"name": "test-parameter",
"description": "",
"i18n_description": null,
"value": "10",
"constraint": {
"type": "String",
"editable": true,
"required": false,
"sensitive": false,
"valid_type": "None",
"valid_range": [ ]
}
}
],
"parameters_customization": true,
"inputs": [ {
"name": "data_url",
"description": "name to translate"
}
],
"outputs": [ {
"name": "train_url",
"description": "name to translate"
}
],
"engine": {
"engine_id": "pytorch-cp36-1.3.0",
"engine_name": "PyTorch",
"engine_version": "PyTorch-1.3.0-python3.6"
},
"code_tree": {
"name": "code/",
"children": [ {
"name": "test-pytorch.py"
}
]
}
},
"resource_requirements": null,
"advanced_config": { }
}

```

Status Codes

Status Code	Description
201	ok

Error Codes

See [Error Codes](#).

5.5 Deleting an Algorithm

Function

This API is used to delete an algorithm.

URI

DELETE /v2/{project_id}/algorithms/{algorithm_id}

Table 5-113 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .
algorithm_id	Yes	String	Algorithm ID.

Request Parameters

None

Response Parameters

None

Example Requests

The following shows how to modify the algorithm whose UUID is **2e5451fe-913f-4492-821a-2981031382f7**.

```
DELETE https://endpoint/v2/{project_id}/algorithms/2e5451fe-913f-4492-821a-2981031382f7
```

Example Responses

Status code: 202

No Content

```
null
```

Status Codes

Status Code	Description
202	No Content

Error Codes

See [Error Codes](#).

5.6 Creating a Training Job

Function

This API is used to create a training job.

URI

POST /v2/{project_id}/training-jobs

Table 5-114 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .

Request Parameters

Table 5-115 Request body parameters

Parameter	Mandatory	Type	Description
kind	Yes	String	Training job type, which is job by default. Options: <ul style="list-style-type: none"> job: training job
metadata	Yes	JobMetadata object	Metadata of a training job.

Parameter	Mandatory	Type	Description
algorithm	No	JobAlgorithm object	Algorithm used by a training job. Options: <ul style="list-style-type: none"> • id: Only the algorithm ID is used. • subscription_id +item_version_id: The subscription ID and version ID of the algorithm are used. • code_dir+boot_file: The code directory and boot file of the training job are used.
tasks	No	Array of Task objects	Task list. This function is not implemented currently.
spec	No	spec object	Specifications of a training job. If this parameter is specified, leave the tasks parameter blank.

Table 5-116 JobMetadata

Parameter	Mandatory	Type	Description
id	No	String	Training job ID, which is generated and returned by ModelArts after the training job is created.
name	Yes	String	Name of a training job. The value must contain 1 to 64 characters consisting of only digits, letters, underscores (_), and hyphens (-).
workspace_id	No	String	Workspace where a job is located. The default value is 0 .
description	No	String	Training job description. The value must contain 0 to 256 characters. The default value is NULL .
create_time	No	Long	Time when a training job was created, in milliseconds. The value is generated and returned by ModelArts after a training job is created.

Parameter	Mandatory	Type	Description
user_name	No	String	Username for creating a training job. The username is generated and returned by ModelArts after a training job is created.
annotations	No	Map<String,String>	Advanced configuration of a training job. Options: <ul style="list-style-type: none"> • job_template: Template RL (heterogeneous job) • fault-tolerance/job-retry-num: 3 (number of retries upon a fault)

Table 5-117 JobAlgorithm

Parameter	Mandatory	Type	Description
id	No	String	Algorithm ID.
name	No	String	Algorithm name. Leave it blank.
subscription_id	No	String	Subscription ID of a subscribed algorithm, which must be used with item_version_id
item_version_id	No	String	Version ID of the subscribed algorithm, which must be used with subscription_id
code_dir	No	String	Code directory of a training job, for example, /usr/app/ . This parameter must be used together with boot_file . If id or subscription_id +item_version_id is set, leave it blank.
boot_file	No	String	Boot file of a training job, which must be stored in the code directory, for example, /usr/app/boot.py . This parameter must be used with code_dir . Leave this parameter blank if id , or subscription_id and item_version_id are specified.

Parameter	Mandatory	Type	Description
autosearch_config_path	No	String	YAML configuration path of auto search jobs. An OBS URL is required.
autosearch_framework_path	No	String	Framework code directory of auto search jobs. An OBS URL is required.
command	No	String	Command for starting the container of the custom image of a training job in the custom image scenario.
parameters	No	Array of parameters objects	Running parameters of a training job.
policies	No	policies object	Policies supported by jobs, which are used for hyperparameter search.
inputs	No	Array of Input objects	Input of a training job.
outputs	No	Array of Output objects	Output of a training job.
engine	No	engine object	Engine of a training job. Leave this parameter blank if the job is created using id of the algorithm in algorithm management, or subscription_id+item_version_id of the subscribed algorithm.
local_code_dir	No	String	Local directory to the training container to which the algorithm code directory is downloaded Rules: <ul style="list-style-type: none"> • The value must be a directory in /home. • In v1 compatibility mode, the current field does not take effect. • When code_dir is prefixed with file://, the current field does not take effect.

Parameter	Mandatory	Type	Description
working_dir	No	String	Work directory where an algorithm is executed. Note that this parameter does not take effect in v1 compatibility mode.
environments	No	Array of Map<String,String> objects	Environment variables of a training job. The format is key: value . Leave this parameter blank.

Table 5-118 parameters

Parameter	Mandatory	Type	Description
name	No	String	Parameter name.
value	No	String	Parameter value.
description	No	String	Parameter description.
constraint	No	constraint object	Parameter constraint.
i18n_description	No	i18n_description object	Internationalization description.

Table 5-119 constraint

Parameter	Mandatory	Type	Description
type	No	String	Parameter type.
editable	No	Boolean	Whether the parameter is editable.
required	No	Boolean	Whether the parameter is mandatory.
sensitive	No	Boolean	Whether the parameter is sensitive This function is not implemented currently.
valid_type	No	String	Valid type.
valid_range	No	Array of strings	Valid range.

Table 5-120 i18n_description

Parameter	Mandatory	Type	Description
language	No	String	Internationalization language.
description	No	String	Description.

Table 5-121 policies

Parameter	Mandatory	Type	Description
auto_search	No	auto_search object	Hyperparameter search configuration.

Table 5-122 auto_search

Parameter	Mandatory	Type	Description
skip_search_params	No	String	Hyperparameter parameters that need to be skipped.
reward_attrs	No	Array of reward_attrs objects	List of search metrics.
search_params	No	Array of search_params objects	Search parameters.
algo_configs	No	Array of algo_configs objects	Search algorithm configurations.

Table 5-123 reward_attrs

Parameter	Mandatory	Type	Description
name	No	String	Metric name.
mode	No	String	Search direction. <ul style="list-style-type: none"> • max: A larger metric value indicates better performance. • min: A smaller metric value indicates better performance.
regex	No	String	Regular expression of a metric.

Table 5-124 search_params

Parameter	Mandatory	Type	Description
name	No	String	Hyperparameter name.
param_type	No	String	Parameter type <ul style="list-style-type: none"> If continuous is specified, the hyperparameter is of the continuous type. When an algorithm is used in a training job, continuous hyperparameters are displayed as text boxes on the console. - discrete: The hyperparameter is of the discrete type. When an algorithm is used for training jobs, discrete hyperparameters are displayed as a drop-down list box on the console.
lower_bound	No	String	Lower bound of the hyperparameter.
upper_bound	No	String	Upper bound of the hyperparameter.
discrete_point_s_num	No	String	Number of discrete points of a continuous hyperparameter.
discrete_values	No	Array of strings	List of discrete hyperparameter values.

Table 5-125 algo_configs

Parameter	Mandatory	Type	Description
name	No	String	Name of the search algorithm.
params	No	Array of AutoSearchAlgoConfigParameter objects	Search algorithm parameters.

Table 5-126 AutoSearchAlgoConfigParameter

Parameter	Mandatory	Type	Description
key	No	String	Parameter key.

Parameter	Mandatory	Type	Description
value	No	String	Parameter value.
type	No	String	Parameter type.

Table 5-127 engine

Parameter	Mandatory	Type	Description
engine_id	No	String	Engine ID selected for a training job. You can set this parameter to engine_id , engine_name + engine_version , or image_url .
engine_name	No	String	Name of the engine selected for a training job. If engine_id is set, leave this parameter blank.
engine_version	No	String	Name of the engine version selected for a training job. If engine_id is set, leave this parameter blank.
image_url	No	String	Custom image URL selected for a training job.

Table 5-128 Task

Parameter	Mandatory	Type	Description
role	No	String	Task role. This function is not supported currently.
algorithm	No	algorithm object	Algorithm management and configuration.
task_resource	No	task_resource object	Resource flavors of a training job.

Table 5-129 algorithm

Parameter	Mandatory	Type	Description
job_config	No	job_config object	Algorithm configuration, such as the boot file.

Parameter	Mandatory	Type	Description
code_dir	No	String	Algorithm code directory, for example, /usr/app/ . This parameter must be used together with boot_file .
boot_file	No	String	Code boot file of the algorithm, which needs to be stored in the code directory, for example, /usr/app/boot.py . This parameter must be used together with code_dir .
engine	No	engine object	Engine of a heterogeneous job algorithm.
inputs	No	Array of inputs objects	Data input of an algorithm.
outputs	No	Array of outputs objects	Data output of an algorithm.
local_code_dir	No	String	Local directory to the training container to which the algorithm code directory is downloaded. Ensure that the following rules are complied with: - The directory must be in the /home directory. - In v1 compatibility mode, the current field does not take effect. - When code_dir is prefixed with file:// , the current field does not take effect.
working_dir	No	String	Work directory where an algorithm is executed. Note that this parameter does not take effect in v1 compatibility mode.

Table 5-130 job_config

Parameter	Mandatory	Type	Description
parameters	No	Array of Parameter objects	Running parameter of an algorithm.

Parameter	Mandatory	Type	Description
inputs	No	Array of Input objects	Data input of an algorithm.
outputs	No	Array of Output objects	Data output of an algorithm.
engine	No	engine object	Algorithm engine.

Table 5-131 Parameter

Parameter	Mandatory	Type	Description
name	No	String	Parameter name.
value	No	String	Parameter value.
description	No	String	Parameter description.
constraint	No	constraint object	Parameter constraint.
i18n_description	No	i18n_description object	Internationalization description.

Table 5-132 constraint

Parameter	Mandatory	Type	Description
type	No	String	Parameter type.
editable	No	Boolean	Whether the parameter is editable.
required	No	Boolean	Whether the parameter is mandatory.
sensitive	No	Boolean	Whether the parameter is sensitive This function is not implemented currently.
valid_type	No	String	Valid type.
valid_range	No	Array of strings	Valid range.

Table 5-133 i18n_description

Parameter	Mandatory	Type	Description
language	No	String	Language.
description	No	String	Description.

Table 5-134 Input

Parameter	Mandatory	Type	Description
name	Yes	String	Name of the data input channel.
description	No	String	Description of the data input channel.
local_dir	No	String	Local directory of the container to which the data input channel is mapped.
remote	Yes	InputDataInfo object	Data input. Options: <ul style="list-style-type: none"> • dataset: Dataset as the data input • obs: OBS path as the data input
remote_constraint	No	Array of remote_constraint objects	Data input constraint

Table 5-135 InputDataInfo

Parameter	Mandatory	Type	Description
dataset	No	dataset object	Dataset as the data input.
obs	No	obs object	OBS in which data input and output stored.

Table 5-136 dataset

Parameter	Mandatory	Type	Description
id	Yes	String	Dataset ID of a training job.
version_id	Yes	String	Dataset version ID of a training job.

Parameter	Mandatory	Type	Description
obs_url	No	String	OBS URL of the dataset required by a training job. ModelArts automatically parses and generates the URL based on the dataset and dataset version IDs. For example, /usr/data/ .

Table 5-137 obs

Parameter	Mandatory	Type	Description
obs_url	Yes	String	OBS URL of the dataset required by a training job. For example, /usr/data/ .

Table 5-138 remote_constraint

Parameter	Mandatory	Type	Description
data_type	No	String	Data input type, including the data storage location and dataset.
attributes	No	String	Attributes if a dataset is used as the data input. Options: <ul style="list-style-type: none"> • data_format: Data format • data_segmentation: Data segmentation • dataset_type: Labeling type

Table 5-139 Output

Parameter	Mandatory	Type	Description
name	Yes	String	Name of the data output channel.
description	No	String	Description of the data output channel.
local_dir	No	String	Local directory of the container to which the data output channel is mapped.

Parameter	Mandatory	Type	Description
remote	Yes	remote object	Description of the actual data output.

Table 5-140 remote

Parameter	Mandatory	Type	Description
obs	Yes	obs object	OBS to which data is actually exported.

Table 5-141 obs

Parameter	Mandatory	Type	Description
obs_url	Yes	String	OBS URL to which data is actually exported.

Table 5-142 engine

Parameter	Mandatory	Type	Description
engine_id	No	String	Engine ID selected for an algorithm.
engine_name	No	String	Engine version name selected for an algorithm. If engine_id is specified, leave this parameter blank.
engine_version	No	String	Engine version name selected for an algorithm. If engine_id is specified, leave this parameter blank.
image_url	No	String	Custom image URL selected by an algorithm.

Table 5-143 engine

Parameter	Mandatory	Type	Description
engine_id	No	String	Engine ID of a heterogeneous job, for example, caffe-1.0.0-python2.7 .

Parameter	Mandatory	Type	Description
engine_name	No	String	Engine name of a heterogeneous job, for example, Caffe .
engine_version	No	String	Engine version of a heterogeneous job.
image_url	No	String	Custom image URL selected by an algorithm.

Table 5-144 inputs

Parameter	Mandatory	Type	Description
name	Yes	String	Name of the data input channel.
description	No	String	Description of the data input channel.
local_dir	No	String	Local directory of the container to which the data input channel is mapped.
remote	Yes	remote object	Data input. Options: <ul style="list-style-type: none"> • dataset: Dataset as the data input • obs: OBS path as the data input

Table 5-145 remote

Parameter	Mandatory	Type	Description
obs	No	obs object	OBS in which data input and output stored.

Table 5-146 obs

Parameter	Mandatory	Type	Description
obs_url	Yes	String	OBS URL of the dataset required by a training job. For example, /usr/data/ .

Table 5-147 outputs

Parameter	Mandatory	Type	Description
name	Yes	String	Name of the data output channel.
description	No	String	Description of the data output channel.
local_dir	No	String	Local directory of the container to which the data output channel is mapped.
remote	Yes	remote object	Description of the actual data output.

Table 5-148 remote

Parameter	Mandatory	Type	Description
obs	Yes	obs object	OBS to which data is actually exported.

Table 5-149 obs

Parameter	Mandatory	Type	Description
obs_url	Yes	String	OBS URL to which data is actually exported.

Table 5-150 task_resource

Parameter	Mandatory	Type	Description
flavor_id	No	String	Resource flavor ID of a training job.
node_count	Yes	Integer	Number of resource replicas selected for a training job.

Table 5-151 spec

Parameter	Mandatory	Type	Description
resource	No	resource object	Resource flavors of a training job. Select either flavor_id or pool_id+[flavor_id] .

Parameter	Mandatory	Type	Description
volumes	No	Array of volumes objects	Volumes attached to a training job.
log_export_path	No	log_export_path object	Export path of training job logs.
auto_stop	No	auto_stop object	Auto stop configuration of a training job

Table 5-152 resource

Parameter	Mandatory	Type	Description
flavor_id	No	String	ID of the resource flavor selected for a training job. flavor_id cannot be specified for dedicated resource pools with CPU specifications. The options for dedicated resource pools with GPU/Ascend specifications are as follows: <ul style="list-style-type: none"> • modelarts.pool.visual.xlarge (1 card) • modelarts.pool.visual.2xlarge (2 cards) • modelarts.pool.visual.4xlarge (4 cards) • modelarts.pool.visual.8xlarge (8 cards)
node_count	No	Integer	Number of nodes used for creating a training job in a pool. By default, a single node is used.
pool_id	No	String	Dedicated resource pool ID.

Table 5-153 volumes

Parameter	Mandatory	Type	Description
nfs	No	nfs object	Volumes attached in NFS mode.

Table 5-154 nfs

Parameter	Mandatory	Type	Description
nfs_server_path	No	String	NFS server path.
local_path	No	String	Path for attaching volumes to the training container.
read_only	No	Boolean	Whether the volumes attached to the container in NFS mode are read-only.

Table 5-155 log_export_path

Parameter	Mandatory	Type	Description
obs_url	No	String	OBS URL for storing training job logs.
host_path	No	String	Path of the host where training job logs are stored.

Table 5-156 auto_stop

Parameter	Mandatory	Type	Description
time_unit	Yes	String	Time unit. Options: <ul style="list-style-type: none"> HOURS
duration	Yes	Integer	Running time. The minimum value is 1 .

Response Parameters

Status code: 201

Table 5-157 Response body parameters

Parameter	Type	Description
kind	String	Training job type, which is job by default. Options: <ul style="list-style-type: none"> job: training job
metadata	JobMetadata object	Metadata of a training job.

Parameter	Type	Description
status	Status object	Status of a training job. You do not need to set this parameter when creating a job.
algorithm	JobAlgorithmResponse object	Algorithm used by a training job. Options: <ul style="list-style-type: none"> • id: Only the algorithm ID is used. • subscription_id+item_version_id: The subscription ID and version ID of the algorithm are used. • code_dir+boot_file: The code directory and boot file of the training job are used.
tasks	Array of TaskResponse objects	List of tasks in heterogeneous training jobs.
spec	spec object	Specifications of a training job.

Table 5-158 JobMetadata

Parameter	Type	Description
id	String	Training job ID, which is generated and returned by ModelArts after the training job is created.
name	String	Name of a training job. The value must contain 1 to 64 characters consisting of only digits, letters, underscores (_), and hyphens (-).
workspace_id	String	Workspace where a job is located. The default value is 0 .
description	String	Training job description. The value must contain 0 to 256 characters. The default value is NULL .
create_time	Long	Time when a training job was created, in milliseconds. The value is generated and returned by ModelArts after a training job is created.
user_name	String	Username for creating a training job. The username is generated and returned by ModelArts after a training job is created.

Parameter	Type	Description
annotations	Map<String,String>	Advanced configuration of a training job. Options: <ul style="list-style-type: none"> • job_template: Template RL (heterogeneous job) • fault-tolerance/job-retry-num: 3 (number of retries upon a fault)

Table 5-159 Status

Parameter	Type	Description
phase	String	Level-1 status of a training job. The options are as follows: Creating Pending Running Failed Completed, Terminating Terminated Abnormal
secondary_phase	String	The level-2 status of a training job is an internal detailed status, which may be added, modified, or deleted. Dependency is not recommended. The options are as follows: Creating Queuing Running Failed Completed, Terminating Terminated CreateFailed TerminatedFailed Unknown Lost
duration	Long	Running duration of a training job, in milliseconds
node_count_metrics	Array<Array<Integer>>	Node count changes during the training job running period.
tasks	Array of strings	Tasks of a training job.
start_time	String	Start time of a training job. The value is in timestamp format.
task_statuses	Array of task_statuses objects	Status of a training job task.

Table 5-160 task_statuses

Parameter	Type	Description
task	String	Name of a training job task.
exit_code	Integer	Exit code of a training job task.

Parameter	Type	Description
message	String	Error message of a training job task.

Table 5-161 JobAlgorithmResponse

Parameter	Type	Description
id	String	Algorithm used by a training job. Options: <ul style="list-style-type: none"> • id: Only the algorithm ID is used. • subscription_id+item_version_id: The subscription ID and version ID of the algorithm are used. • code_dir+boot_file: The code directory and boot file of the training job are used.
name	String	Algorithm name.
subscription_id	String	Subscription ID of a subscribed algorithm, which must be used with item_version_id
item_version_id	String	Version ID of the subscribed algorithm, which must be used with subscription_id
code_dir	String	Code directory of a training job, for example, /usr/app/ . This parameter must be used together with boot_file . If id or subscription_id+item_version_id is set, leave it blank.
boot_file	String	Boot file of a training job, which must be stored in the code directory, for example, /usr/app/boot.py . This parameter must be used with code_dir . Leave this parameter blank if id , or subscription_id and item_version_id are specified.
autosearch_config_path	String	YAML configuration path of auto search jobs. An OBS URL is required.
autosearch_framework_path	String	Framework code directory of auto search jobs. An OBS URL is required.
command	String	Boot command used to start the container of a custom image of a training job. For example, python train.py .
parameters	Array of Parameter objects	Running parameters of a training job.

Parameter	Type	Description
policies	policies object	Policies supported by jobs.
inputs	Array of Input objects	Input of a training job.
outputs	Array of Output objects	Output of a training job.
engine	engine object	Engine of a training job. Leave this parameter blank if the job is created using id of the algorithm in algorithm management, or subscription_id+item_version_id of the subscribed algorithm.
local_code_dir	String	Local directory to the training container to which the algorithm code directory is downloaded. Ensure that the following rules are complied with: - The directory must be in the /home directory. - In v1 compatibility mode, the current field does not take effect. - When code_dir is prefixed with file:// , the current field does not take effect.
working_dir	String	Work directory where an algorithm is executed. Note that this parameter does not take effect in v1 compatibility mode.
environments	Array of Map<String,String> objects	Environment variables of a training job. The format is key: value . Leave this parameter blank.

Table 5-162 Parameter

Parameter	Type	Description
name	String	Parameter name.
value	String	Parameter value.
description	String	Parameter description.
constraint	constraint object	Parameter constraint.
i18n_description	i18n_description object	Internationalization description.

Table 5-163 constraint

Parameter	Type	Description
type	String	Parameter type.
editable	Boolean	Whether the parameter is editable.
required	Boolean	Whether the parameter is mandatory.
sensitive	Boolean	Whether the parameter is sensitive This function is not implemented currently.
valid_type	String	Valid type.
valid_range	Array of strings	Valid range.

Table 5-164 i18n_description

Parameter	Type	Description
language	String	Language.
description	String	Description.

Table 5-165 policies

Parameter	Type	Description
auto_search	auto_search object	Hyperparameter search configuration.

Table 5-166 auto_search

Parameter	Type	Description
skip_search_parameters	String	Hyperparameter parameters that need to be skipped.
reward_attrs	Array of reward_attrs objects	List of search metrics.
search_parameters	Array of search_parameters objects	Search parameters.
algo_configs	Array of algo_configs objects	Search algorithm configurations.

Table 5-167 reward_attrs

Parameter	Type	Description
name	String	Metric name.
mode	String	Search direction. <ul style="list-style-type: none"> • max: A larger metric value indicates better performance. • min: A smaller metric value indicates better performance.
regex	String	Regular expression of a metric.

Table 5-168 search_params

Parameter	Type	Description
name	String	Hyperparameter name.
param_type	String	Parameter type <ul style="list-style-type: none"> • If continuous is specified, the hyperparameter is of the continuous type. When an algorithm is used in a training job, continuous hyperparameters are displayed as text boxes on the console. - discrete: The hyperparameter is of the discrete type. When an algorithm is used for training jobs, discrete hyperparameters are displayed as a drop-down list box on the console.
lower_bound	String	Lower bound of the hyperparameter.
upper_bound	String	Upper bound of the hyperparameter.
discrete_points_num	String	Number of discrete points of a continuous hyperparameter.
discrete_values	Array of strings	List of discrete hyperparameter values.

Table 5-169 algo_configs

Parameter	Type	Description
name	String	Name of the search algorithm.
params	Array of AutoSearchAlgorithmConfigParameter objects	Search algorithm parameters.

Table 5-170 AutoSearchAlgoConfigParameter

Parameter	Type	Description
key	String	Parameter key.
value	String	Parameter value.
type	String	Parameter type.

Table 5-171 Input

Parameter	Type	Description
name	String	Name of the data input channel.
description	String	Description of the data input channel.
local_dir	String	Local directory of the container to which the data input channel is mapped.
remote	InputDataInfo object	Data input. Options: <ul style="list-style-type: none"> • dataset: Dataset as the data input • obs: OBS path as the data input
remote_constraint	Array of remote_constraint objects	Data input constraint

Table 5-172 InputDataInfo

Parameter	Type	Description
dataset	dataset object	Dataset as the data input.
obs	obs object	OBS in which data input and output stored.

Table 5-173 dataset

Parameter	Type	Description
id	String	Dataset ID of a training job.
version_id	String	Dataset version ID of a training job.
obs_url	String	OBS URL of the dataset required by a training job. ModelArts automatically parses and generates the URL based on the dataset and dataset version IDs. For example, /usr/data/ .

Table 5-174 obs

Parameter	Type	Description
obs_url	String	OBS URL of the dataset required by a training job. For example, /usr/data/ .

Table 5-175 remote_constraint

Parameter	Type	Description
data_type	String	Data input type, including the data storage location and dataset.
attributes	String	Attributes if a dataset is used as the data input. Options: <ul style="list-style-type: none"> • data_format: Data format • data_segmentation: Data segmentation • dataset_type: Labeling type

Table 5-176 Output

Parameter	Type	Description
name	String	Name of the data output channel.
description	String	Description of the data output channel.
local_dir	String	Local directory of the container to which the data output channel is mapped.
remote	remote object	Description of the actual data output.

Table 5-177 remote

Parameter	Type	Description
obs	obs object	OBS to which data is actually exported.

Table 5-178 obs

Parameter	Type	Description
obs_url	String	OBS URL to which data is actually exported.

Table 5-179 engine

Parameter	Type	Description
engine_id	String	Engine ID selected for a training job. You can set this parameter to engine_id , engine_name + engine_version , or image_url .
engine_name	String	Name of the engine selected for a training job. If engine_id is set, leave this parameter blank.
engine_version	String	Name of the engine version selected for a training job. If engine_id is set, leave this parameter blank.
image_url	String	Custom image URL selected for a training job.

Table 5-180 TaskResponse

Parameter	Type	Description
role	String	Task role. This function is not supported currently.
algorithm	algorithm object	Algorithm management and configuration.
task_resource	FlavorResponse object	Flavors of a training job or an algorithm.

Table 5-181 algorithm

Parameter	Type	Description
code_dir	String	Absolute path of the directory where the algorithm boot file is stored.
boot_file	String	Absolute path of the algorithm boot file.
inputs	inputs object	Algorithm input channel.
outputs	outputs object	Algorithm output channel.
engine	engine object	Engine on which a heterogeneous job depends.

Parameter	Type	Description
local_code_dir	String	Local directory to the training container to which the algorithm code directory is downloaded. Ensure that the following rules are complied with: - The directory must be in the /home directory. - In v1 compatibility mode, the current field does not take effect. - When code_dir is prefixed with file:// , the current field does not take effect.
working_dir	String	Work directory where an algorithm is executed. Note that this parameter does not take effect in v1 compatibility mode.

Table 5-182 inputs

Parameter	Type	Description
name	String	Name of the data input channel.
local_dir	String	Local path of the container to which the data input and output channels are mapped.
remote	remote object	Actual data input. Heterogeneous jobs support only OBS.

Table 5-183 remote

Parameter	Type	Description
obs	obs object	OBS in which data input and output stored.

Table 5-184 obs

Parameter	Type	Description
obs_url	String	OBS URL of the dataset required by a training job. For example, /usr/data/ .

Table 5-185 outputs

Parameter	Type	Description
name	String	Name of the data output channel.

Parameter	Type	Description
local_dir	String	Local directory of the container to which the data output channel is mapped.
remote	remote object	Description of the actual data output.
mode	String	Data transmission mode. The default value is upload_periodically .
period	String	Data transmission period. The default value is 30s .

Table 5-186 remote

Parameter	Type	Description
obs	obs object	OBS to which data is actually exported.

Table 5-187 obs

Parameter	Type	Description
obs_url	String	OBS URL to which data is actually exported.

Table 5-188 engine

Parameter	Type	Description
engine_id	String	Engine ID of a heterogeneous job, for example, caffe-1.0.0-python2.7 .
engine_name	String	Engine name of a heterogeneous job, for example, Caffe .
engine_version	String	Engine version of a heterogeneous job.
v1_compatible	Boolean	Whether the v1 compatibility mode is used.
run_user	String	User UID started by default by the engine.
image_url	String	Custom image URL selected by an algorithm.

Table 5-189 FlavorResponse

Parameter	Type	Description
flavor_id	String	ID of the resource flavor.
flavor_name	String	Name of the resource flavor.
max_num	Integer	Maximum number of nodes in a resource flavor.
flavor_type	String	Resource flavor type. Options: <ul style="list-style-type: none"> • CPU • GPU
billing	billing object	Billing information of a resource flavor.
flavor_info	flavor_info object	Resource flavor details.
attributes	Map<String,String>	Other specification attributes.

Table 5-190 billing

Parameter	Type	Description
code	String	Billing code.
unit_num	Integer	Number of billing units.

Table 5-191 flavor_info

Parameter	Type	Description
max_num	Integer	Maximum number of nodes that can be selected. The value 1 indicates that the distributed mode is not supported.
cpu	cpu object	CPU specifications.
gpu	gpu object	GPU specifications.
npu	npu object	Ascend specifications
memory	memory object	Memory information.
disk	disk object	Disk information.

Table 5-192 cpu

Parameter	Type	Description
arch	String	CPU architecture.
core_num	Integer	Number of cores.

Table 5-193 gpu

Parameter	Type	Description
unit_num	Integer	Number of GPUs.
product_name	String	Product name.
memory	String	Memory.

Table 5-194 npu

Parameter	Type	Description
unit_num	String	Number of NPUs.
product_name	String	Product name.
memory	String	Memory.

Table 5-195 memory

Parameter	Type	Description
size	Integer	Memory size.
unit	String	Memory size

Table 5-196 disk

Parameter	Type	Description
size	Integer	Disk size.
unit	String	Unit of the disk size.

Table 5-197 spec

Parameter	Type	Description
resource	Resource object	Resource flavors of a training job. Select either flavor_id or pool_id+[flavor_id] .
volumes	Array of volumes objects	Volumes attached to a training job.
log_export_path	log_export_path object	Export path of training job logs.

Table 5-198 Resource

Parameter	Type	Description
policy	String	Resource flavor of a training job. Options: regular
flavor_id	String	ID of the resource flavor selected for a training job. flavor_id cannot be specified for dedicated resource pools with CPU specifications. The options for dedicated resource pools with GPU/Ascend specifications are as follows: <ul style="list-style-type: none"> • modelarts.pool.visual.xlarge (1 card) • modelarts.pool.visual.2xlarge (2 cards) • modelarts.pool.visual.4xlarge (4 cards) • modelarts.pool.visual.8xlarge (8 cards)
flavor_name	String	Read-only flavor name returned by ModelArts when flavor_id is used.
node_count	Integer	Number of resource replicas selected for a training job.
pool_id	String	Resource pool ID selected for a training job.
flavor_detail	flavor_detail object	Flavors of a training job or an algorithm.

Table 5-199 flavor_detail

Parameter	Type	Description
flavor_type	String	Resource flavor type. Options: <ul style="list-style-type: none"> • CPU • GPU

Parameter	Type	Description
billing	billing object	Billing information of a resource flavor.
flavor_info	flavor_info object	Resource flavor details.

Table 5-200 billing

Parameter	Type	Description
code	String	Billing code.
unit_num	Integer	Number of billing units.

Table 5-201 flavor_info

Parameter	Type	Description
max_num	Integer	Maximum number of nodes that can be selected. The value 1 indicates that the distributed mode is not supported.
cpu	cpu object	CPU specifications.
gpu	gpu object	GPU specifications.
npu	npu object	Ascend specifications
memory	memory object	Memory information.
disk	disk object	Disk information.

Table 5-202 cpu

Parameter	Type	Description
arch	String	CPU architecture.
core_num	Integer	Number of cores.

Table 5-203 gpu

Parameter	Type	Description
unit_num	Integer	Number of GPUs.
product_name	String	Product name.

Parameter	Type	Description
memory	String	Memory.

Table 5-204 npu

Parameter	Type	Description
unit_num	String	Number of NPUs.
product_name	String	Product name.
memory	String	Memory.

Table 5-205 memory

Parameter	Type	Description
size	Integer	Memory size.
unit	String	Number of memory units.

Table 5-206 disk

Parameter	Type	Description
size	String	Disk size.
unit	String	Unit of the disk size. Generally, the value is GB.

Table 5-207 volumes

Parameter	Type	Description
nfs	nfs object	Volumes attached in NFS mode.

Table 5-208 nfs

Parameter	Type	Description
nfs_server_path	String	NFS server path.
local_path	String	Path for attaching volumes to the training container.

Parameter	Type	Description
read_only	Boolean	Whether the volumes attached to the container in NFS mode are read-only.

Table 5-209 log_export_path

Parameter	Type	Description
obs_url	String	OBS URL for storing training job logs.
host_path	String	Path of the host where training job logs are stored.

Status code: 400

Table 5-210 Response body parameters

Parameter	Type	Description
error_msg	String	Error message
error_code	String	Error code
error_solution	String	Solution

Example Requests

- The following is an example of how to create a training job with free specifications. The job name has been set to **TestModelArtsJob** and the description has been set to **This is a ModelArts job**. The required algorithm's ID is **3f5d6706-7b67-408d-8ba0-ec08048c45ed**. The inputs and outputs have not been defined for the algorithm.

POST https://endpoint/v2/{project_id}/training-jobs

```
{
  "kind": "job",
  "metadata": {
    "name": "TestModelArtsJob",
    "description": "This is a ModelArts job"
  },
  "algorithm": {
    "id": "3f5d6706-7b67-408d-8ba0-ec08048c45ed",
    "parameters": [ {
      "name": "input_dir",
      "value": "obs://xxxxxx-rse/test/moxingtest-dir/"
    }, {
      "name": "input_file",
      "value": "obs://xxxxxx-rse/test/moxingtest/"
    }, {
      "name": "large_file_method",
      "value": "1"
    }
  ],
  "policies": {
```

```

    "auto_search" : null
  },
  "environments" : { }
},
"spec" : {
  "resource" : {
    "flavor_id" : "modelarts.p3.large.public.free",
    "node_count" : 1
  },
  "log_export_path" : {
    "obs_url" : ""
  }
}
}
}

```

- The following is an example of how to use a custom image to create a training job whose name is **TestModelArtsJob2** and description is **This is a ModelArts job2**. A dedicated resource pool and NFS mounting are used.

POST https://endpoint/v2/{project_id}/training-jobs

```

{
  "kind" : "job",
  "metadata" : {
    "name" : "TestModelArtsJob2",
    "description" : "This is a ModelArts job2"
  },
  "algorithm" : {
    "engine" : {
      "image_url" : "hwstaff_z00424192/fastseq:1.2"
    },
    "command" : "cd /home/ma-user/ddp_demo && sh run_ddp.sh",
    "parameters" : [ ],
    "policies" : {
      "auto_search" : null
    },
    "environments" : {
      "NCCL_DEBUG" : "INFO",
      "NCCL_IB_DISABLE" : "0"
    }
  },
  "spec" : {
    "resource" : {
      "flavor_id" : "modelarts.pool.visual.xlarge",
      "node_count" : 1,
      "pool_id" : "poolfaf38d76"
    },
    "log_export_path" : {
      "obs_url" : "/xxxxxx-training-test/limou/ddp-demo-log/"
    },
    "volumes" : [ {
      "nfs" : {
        "nfs_server_path" : "192.168.0.82:/",
        "local_path" : "/home/ma-user/nfs/",
        "read_only" : false
      }
    }
  ]
}
}
}

```

Example Responses

Status code: 201

ok

```

{
  "kind" : "job",
  "metadata" : {

```

```

    "id": "425b7087-83de-49ed-9e40-5bb642be956f",
    "name": "TestModelArtsJob",
    "description": "This is a ModelArts job",
    "create_time": 1637045545982,
    "workspace_id": "0",
    "user_name": ""
  },
  "status": {
    "phase": "Creating",
    "secondary_phase": "Creating",
    "duration": 0,
    "start_time": 0,
    "node_count_metrics": null,
    "tasks": [ "worker-0", "server-0" ]
  },
  "algorithm": {
    "id": "3f5d6706-7b67-408d-8ba0-ec08048c45ed",
    "name": "tnt-obs-gpu",
    "code_dir": "/xxxxxx-rse/test/moxingtest-code/",
    "boot_file": "/xxxxxx-rse/test/moxingtest-code/test_obs_gpu.py",
    "parameters": [ {
      "name": "input_dir",
      "description": "",
      "i18n_description": null,
      "value": "s://xxxxxx-rse/test/moxingtest-dir/",
      "constraint": {
        "type": "String",
        "editable": true,
        "required": true,
        "sensitive": false,
        "valid_type": "None",
        "valid_range": [ ]
      }
    }, {
      "name": "input_file",
      "description": "",
      "i18n_description": null,
      "value": "obs://xxxxxx-rse/test/moxingtest/",
      "constraint": {
        "type": "String",
        "editable": true,
        "required": true,
        "sensitive": false,
        "valid_type": "None",
        "valid_range": [ ]
      }
    }
  ], {
    "name": "large_file_method",
    "description": "",
    "i18n_description": null,
    "value": "1",
    "constraint": {
      "type": "Integer",
      "editable": true,
      "required": true,
      "sensitive": false,
      "valid_type": "None",
      "valid_range": [ ]
    }
  }
}, {
  "engine": {
    "engine_id": "horovod-cp36-tf-1.16.2",
    "engine_name": "Horovod",
    "engine_version": "0.16.2-TF-1.13.1-python3.6"
  },
  "policies": { }
},
"spec": {
  "resource": {

```

```

"policy": "regular",
"flavor_id": "modelarts.p3.large.public.free",
"flavor_name": "Computing GPU(V100) instance",
"node_count": 1,
"flavor_detail": {
  "flavor_type": "GPU",
  "billing": {
    "code": "modelarts.vm.gpu.free",
    "unit_num": 1
  },
  "flavor_info": {
    "cpu": {
      "arch": "x86",
      "core_num": 8
    },
    "gpu": {
      "unit_num": 1,
      "product_name": "NVIDIA-V100",
      "memory": "32GB"
    },
    "memory": {
      "size": 64,
      "unit": "GB"
    }
  }
},
"log_export_path": {}
}

```

Status code: 400

Format of the body for a common error response. The following shows the returned information when an algorithm with ID 3f5d6706-7b67-408d-8ba0-ec08048c45ee is not found.

```

{
  "error_msg": "algorithm not found.",
  "error_code": "ModelArts.2755",
  "error_solution": "Check whether the training project information in the request is valid."
}

```

Status Codes

Status Code	Description
201	ok
400	Format of the body for a common error response. The following shows the returned information when an algorithm with ID 3f5d6706-7b67-408d-8ba0-ec08048c45ee is not found.

Error Codes

See [Error Codes](#).

5.7 Querying the Details About a Training Job

Function

This API is used to query the details about a training job.

URI

GET /v2/{project_id}/training-jobs/{training_job_id}

Table 5-211 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .
training_job_id	Yes	String	ID of a training job.

Request Parameters

None

Response Parameters

Status code: 200

Table 5-212 Response body parameters

Parameter	Type	Description
kind	String	Training job type, which is job by default. Options: <ul style="list-style-type: none"> job: training job
metadata	JobMetadata object	Metadata of a training job.
status	Status object	Status of a training job. You do not need to set this parameter when creating a job.

Parameter	Type	Description
algorithm	JobAlgorithm Response object	Algorithm used by a training job. Options: <ul style="list-style-type: none"> • id: Only the algorithm ID is used. • subscription_id+item_version_id: The subscription ID and version ID of the algorithm are used. • code_dir+boot_file: The code directory and boot file of the training job are used.
tasks	Array of TaskResponse objects	List of tasks in heterogeneous training jobs.
spec	spec object	Specifications of a training job.

Table 5-213 JobMetadata

Parameter	Type	Description
id	String	Training job ID, which is generated and returned by ModelArts after the training job is created.
name	String	Name of a training job. The value must contain 1 to 64 characters consisting of only digits, letters, underscores (_), and hyphens (-).
workspace_id	String	Workspace where a job is located. The default value is 0 .
description	String	Training job description. The value must contain 0 to 256 characters. The default value is NULL .
create_time	Long	Time when a training job was created, in milliseconds. The value is generated and returned by ModelArts after a training job is created.
user_name	String	Username for creating a training job. The username is generated and returned by ModelArts after a training job is created.
annotations	Map<String,String>	Advanced configuration of a training job. Options: <ul style="list-style-type: none"> • job_template: Template RL (heterogeneous job) • fault-tolerance/job-retry-num: 3 (number of retries upon a fault)

Table 5-214 Status

Parameter	Type	Description
phase	String	Level-1 status of a training job. The options are as follows: Creating Pending Running Failed Completed, Terminating Terminated Abnormal
secondary_phase	String	The level-2 status of a training job is an internal detailed status, which may be added, modified, or deleted. Dependency is not recommended. The options are as follows: Creating Queuing Running Failed Completed, Terminating Terminated CreateFailed TerminatedFailed Unknown Lost
duration	Long	Running duration of a training job, in milliseconds
node_count_metrics	Array<Array<Integer>>	Node count changes during the training job running period.
tasks	Array of strings	Tasks of a training job.
start_time	String	Start time of a training job. The value is in timestamp format.
task_statuses	Array of task_statuses objects	Status of a training job task.

Table 5-215 task_statuses

Parameter	Type	Description
task	String	Name of a training job task.
exit_code	Integer	Exit code of a training job task.
message	String	Error message of a training job task.

Table 5-216 JobAlgorithmResponse

Parameter	Type	Description
id	String	Algorithm used by a training job. Options: <ul style="list-style-type: none"> • id: Only the algorithm ID is used. • subscription_id+item_version_id: The subscription ID and version ID of the algorithm are used. • code_dir+boot_file: The code directory and boot file of the training job are used.
name	String	Algorithm name.
subscription_id	String	Subscription ID of a subscribed algorithm, which must be used with item_version_id
item_version_id	String	Version ID of the subscribed algorithm, which must be used with subscription_id
code_dir	String	Code directory of a training job, for example, <code>/usr/app/</code> . This parameter must be used together with boot_file . If id or subscription_id+item_version_id is set, leave it blank.
boot_file	String	Boot file of a training job, which must be stored in the code directory, for example, <code>/usr/app/boot.py</code> . This parameter must be used with code_dir . Leave this parameter blank if id , or subscription_id and item_version_id are specified.
autosearch_config_path	String	YAML configuration path of auto search jobs. An OBS URL is required.
autosearch_framework_path	String	Framework code directory of auto search jobs. An OBS URL is required.
command	String	Boot command used to start the container of a custom image of a training job. For example, <code>python train.py</code> .
parameters	Array of Parameter objects	Running parameters of a training job.
policies	policies object	Policies supported by jobs.
inputs	Array of Input objects	Input of a training job.

Parameter	Type	Description
outputs	Array of Output objects	Output of a training job.
engine	engine object	Engine of a training job. Leave this parameter blank if the job is created using id of the algorithm in algorithm management, or subscription_id+item_version_id of the subscribed algorithm.
local_code_dir	String	Local directory to the training container to which the algorithm code directory is downloaded. Ensure that the following rules are complied with: - The directory must be in the /home directory. - In v1 compatibility mode, the current field does not take effect. - When code_dir is prefixed with file:// , the current field does not take effect.
working_dir	String	Work directory where an algorithm is executed. Note that this parameter does not take effect in v1 compatibility mode.
environments	Array of Map<String,String> objects	Environment variables of a training job. The format is key: value . Leave this parameter blank.

Table 5-217 Parameter

Parameter	Type	Description
name	String	Parameter name.
value	String	Parameter value.
description	String	Parameter description.
constraint	constraint object	Parameter constraint.
i18n_description	i18n_description object	Internationalization description.

Table 5-218 constraint

Parameter	Type	Description
type	String	Parameter type.
editable	Boolean	Whether the parameter is editable.

Parameter	Type	Description
required	Boolean	Whether the parameter is mandatory.
sensitive	Boolean	Whether the parameter is sensitive This function is not implemented currently.
valid_type	String	Valid type.
valid_range	Array of strings	Valid range.

Table 5-219 i18n_description

Parameter	Type	Description
language	String	Language.
description	String	Description.

Table 5-220 policies

Parameter	Type	Description
auto_search	auto_search object	Hyperparameter search configuration.

Table 5-221 auto_search

Parameter	Type	Description
skip_search_parameters	String	Hyperparameter parameters that need to be skipped.
reward_attrs	Array of reward_attrs objects	List of search metrics.
search_parameters	Array of search_parameters objects	Search parameters.
algo_configs	Array of algo_configs objects	Search algorithm configurations.

Table 5-222 reward_attrs

Parameter	Type	Description
name	String	Metric name.
mode	String	Search direction. <ul style="list-style-type: none"> • max: A larger metric value indicates better performance. • min: A smaller metric value indicates better performance.
regex	String	Regular expression of a metric.

Table 5-223 search_params

Parameter	Type	Description
name	String	Hyperparameter name.
param_type	String	Parameter type <ul style="list-style-type: none"> • If continuous is specified, the hyperparameter is of the continuous type. When an algorithm is used in a training job, continuous hyperparameters are displayed as text boxes on the console. - discrete: The hyperparameter is of the discrete type. When an algorithm is used for training jobs, discrete hyperparameters are displayed as a drop-down list box on the console.
lower_bound	String	Lower bound of the hyperparameter.
upper_bound	String	Upper bound of the hyperparameter.
discrete_points_num	String	Number of discrete points of a continuous hyperparameter.
discrete_values	Array of strings	List of discrete hyperparameter values.

Table 5-224 algo_configs

Parameter	Type	Description
name	String	Name of the search algorithm.
params	Array of AutoSearchAlgorithmConfigParameter objects	Search algorithm parameters.

Table 5-225 AutoSearchAlgoConfigParameter

Parameter	Type	Description
key	String	Parameter key.
value	String	Parameter value.
type	String	Parameter type.

Table 5-226 Input

Parameter	Type	Description
name	String	Name of the data input channel.
description	String	Description of the data input channel.
local_dir	String	Local directory of the container to which the data input channel is mapped.
remote	InputDataInfo object	Data input. Options: <ul style="list-style-type: none"> • dataset: Dataset as the data input • obs: OBS path as the data input
remote_constraint	Array of remote_constraint objects	Data input constraint

Table 5-227 InputDataInfo

Parameter	Type	Description
dataset	dataset object	Dataset as the data input.
obs	obs object	OBS in which data input and output stored.

Table 5-228 dataset

Parameter	Type	Description
id	String	Dataset ID of a training job.
version_id	String	Dataset version ID of a training job.
obs_url	String	OBS URL of the dataset required by a training job. ModelArts automatically parses and generates the URL based on the dataset and dataset version IDs. For example, <code>/usr/data/</code> .

Table 5-229 obs

Parameter	Type	Description
obs_url	String	OBS URL of the dataset required by a training job. For example, /usr/data/ .

Table 5-230 remote_constraint

Parameter	Type	Description
data_type	String	Data input type, including the data storage location and dataset.
attributes	String	Attributes if a dataset is used as the data input. Options: <ul style="list-style-type: none"> • data_format: Data format • data_segmentation: Data segmentation • dataset_type: Labeling type

Table 5-231 Output

Parameter	Type	Description
name	String	Name of the data output channel.
description	String	Description of the data output channel.
local_dir	String	Local directory of the container to which the data output channel is mapped.
remote	remote object	Description of the actual data output.

Table 5-232 remote

Parameter	Type	Description
obs	obs object	OBS to which data is actually exported.

Table 5-233 obs

Parameter	Type	Description
obs_url	String	OBS URL to which data is actually exported.

Table 5-234 engine

Parameter	Type	Description
engine_id	String	Engine ID selected for a training job. You can set this parameter to engine_id , engine_name + engine_version , or image_url .
engine_name	String	Name of the engine selected for a training job. If engine_id is set, leave this parameter blank.
engine_version	String	Name of the engine version selected for a training job. If engine_id is set, leave this parameter blank.
image_url	String	Custom image URL selected for a training job.

Table 5-235 TaskResponse

Parameter	Type	Description
role	String	Task role. This function is not supported currently.
algorithm	algorithm object	Algorithm management and configuration.
task_resource	FlavorResponse object	Flavors of a training job or an algorithm.

Table 5-236 algorithm

Parameter	Type	Description
code_dir	String	Absolute path of the directory where the algorithm boot file is stored.
boot_file	String	Absolute path of the algorithm boot file.
inputs	inputs object	Algorithm input channel.
outputs	outputs object	Algorithm output channel.
engine	engine object	Engine on which a heterogeneous job depends.

Parameter	Type	Description
local_code_dir	String	Local directory to the training container to which the algorithm code directory is downloaded. Ensure that the following rules are complied with: - The directory must be in the /home directory. - In v1 compatibility mode, the current field does not take effect. - When code_dir is prefixed with file:// , the current field does not take effect.
working_dir	String	Work directory where an algorithm is executed. Note that this parameter does not take effect in v1 compatibility mode.

Table 5-237 inputs

Parameter	Type	Description
name	String	Name of the data input channel.
local_dir	String	Local path of the container to which the data input and output channels are mapped.
remote	remote object	Actual data input. Heterogeneous jobs support only OBS.

Table 5-238 remote

Parameter	Type	Description
obs	obs object	OBS in which data input and output stored.

Table 5-239 obs

Parameter	Type	Description
obs_url	String	OBS URL of the dataset required by a training job. For example, /usr/data/ .

Table 5-240 outputs

Parameter	Type	Description
name	String	Name of the data output channel.

Parameter	Type	Description
local_dir	String	Local directory of the container to which the data output channel is mapped.
remote	remote object	Description of the actual data output.
mode	String	Data transmission mode. The default value is upload_periodically .
period	String	Data transmission period. The default value is 30s .

Table 5-241 remote

Parameter	Type	Description
obs	obs object	OBS to which data is actually exported.

Table 5-242 obs

Parameter	Type	Description
obs_url	String	OBS URL to which data is actually exported.

Table 5-243 engine

Parameter	Type	Description
engine_id	String	Engine ID of a heterogeneous job, for example, caffe-1.0.0-python2.7 .
engine_name	String	Engine name of a heterogeneous job, for example, Caffe .
engine_version	String	Engine version of a heterogeneous job.
v1_compatible	Boolean	Whether the v1 compatibility mode is used.
run_user	String	User UID started by default by the engine.
image_url	String	Custom image URL selected by an algorithm.

Table 5-244 FlavorResponse

Parameter	Type	Description
flavor_id	String	ID of the resource flavor.
flavor_name	String	Name of the resource flavor.
max_num	Integer	Maximum number of nodes in a resource flavor.
flavor_type	String	Resource flavor type. Options: <ul style="list-style-type: none"> • CPU • GPU
billing	billing object	Billing information of a resource flavor.
flavor_info	flavor_info object	Resource flavor details.
attributes	Map<String,String>	Other specification attributes.

Table 5-245 billing

Parameter	Type	Description
code	String	Billing code.
unit_num	Integer	Number of billing units.

Table 5-246 flavor_info

Parameter	Type	Description
max_num	Integer	Maximum number of nodes that can be selected. The value 1 indicates that the distributed mode is not supported.
cpu	cpu object	CPU specifications.
gpu	gpu object	GPU specifications.
npu	npu object	Ascend specifications
memory	memory object	Memory information.
disk	disk object	Disk information.

Table 5-247 cpu

Parameter	Type	Description
arch	String	CPU architecture.
core_num	Integer	Number of cores.

Table 5-248 gpu

Parameter	Type	Description
unit_num	Integer	Number of GPUs.
product_name	String	Product name.
memory	String	Memory.

Table 5-249 npu

Parameter	Type	Description
unit_num	String	Number of NPUs.
product_name	String	Product name.
memory	String	Memory.

Table 5-250 memory

Parameter	Type	Description
size	Integer	Memory size.
unit	String	Memory size

Table 5-251 disk

Parameter	Type	Description
size	Integer	Disk size.
unit	String	Unit of the disk size.

Table 5-252 spec

Parameter	Type	Description
resource	Resource object	Resource flavors of a training job. Select either flavor_id or pool_id+[flavor_id] .
volumes	Array of volumes objects	Volumes attached to a training job.
log_export_path	log_export_path object	Export path of training job logs.

Table 5-253 Resource

Parameter	Type	Description
policy	String	Resource flavor of a training job. Options: regular
flavor_id	String	ID of the resource flavor selected for a training job. flavor_id cannot be specified for dedicated resource pools with CPU specifications. The options for dedicated resource pools with GPU/Ascend specifications are as follows: <ul style="list-style-type: none"> • modelarts.pool.visual.xlarge (1 card) • modelarts.pool.visual.2xlarge (2 cards) • modelarts.pool.visual.4xlarge (4 cards) • modelarts.pool.visual.8xlarge (8 cards)
flavor_name	String	Read-only flavor name returned by ModelArts when flavor_id is used.
node_count	Integer	Number of resource replicas selected for a training job.
pool_id	String	Resource pool ID selected for a training job.
flavor_detail	flavor_detail object	Flavors of a training job or an algorithm.

Table 5-254 flavor_detail

Parameter	Type	Description
flavor_type	String	Resource flavor type. Options: <ul style="list-style-type: none"> • CPU • GPU

Parameter	Type	Description
billing	billing object	Billing information of a resource flavor.
flavor_info	flavor_info object	Resource flavor details.

Table 5-255 billing

Parameter	Type	Description
code	String	Billing code.
unit_num	Integer	Number of billing units.

Table 5-256 flavor_info

Parameter	Type	Description
max_num	Integer	Maximum number of nodes that can be selected. The value 1 indicates that the distributed mode is not supported.
cpu	cpu object	CPU specifications.
gpu	gpu object	GPU specifications.
npu	npu object	Ascend specifications
memory	memory object	Memory information.
disk	disk object	Disk information.

Table 5-257 cpu

Parameter	Type	Description
arch	String	CPU architecture.
core_num	Integer	Number of cores.

Table 5-258 gpu

Parameter	Type	Description
unit_num	Integer	Number of GPUs.
product_name	String	Product name.

Parameter	Type	Description
memory	String	Memory.

Table 5-259 npu

Parameter	Type	Description
unit_num	String	Number of NPUs.
product_name	String	Product name.
memory	String	Memory.

Table 5-260 memory

Parameter	Type	Description
size	Integer	Memory size.
unit	String	Number of memory units.

Table 5-261 disk

Parameter	Type	Description
size	String	Disk size.
unit	String	Unit of the disk size. Generally, the value is GB.

Table 5-262 volumes

Parameter	Type	Description
nfs	nfs object	Volumes attached in NFS mode.

Table 5-263 nfs

Parameter	Type	Description
nfs_server_path	String	NFS server path.
local_path	String	Path for attaching volumes to the training container.

Parameter	Type	Description
read_only	Boolean	Whether the volumes attached to the container in NFS mode are read-only.

Table 5-264 log_export_path

Parameter	Type	Description
obs_url	String	OBS URL for storing training job logs.
host_path	String	Path of the host where training job logs are stored.

Example Requests

The following shows how to query a training job whose UUID is **3faf5c03-aaa1-4cbe-879d-24b05d997347**.

```
GET https://endpoint/v2/{project_id}/training-jobs/3faf5c03-aaa1-4cbe-879d-24b05d997347
```

Example Responses

Status code: 200

ok

```
{
  "kind": "job",
  "metadata": {
    "id": "3faf5c03-aaa1-4cbe-879d-24b05d997347",
    "name": "trainjob--py14_mem06-108",
    "description": "",
    "create_time": 1636447346315,
    "workspace_id": "0",
    "user_name": ""
  },
  "status": {
    "phase": "Abnormal",
    "secondary_phase": "CreateFailed",
    "duration": 0,
    "start_time": 0,
    "node_count_metrics": [ [ 1636447746000, 0 ], [ 1636447755000, 0 ], [ 1636447756000, 0 ] ],
    "tasks": [ "worker-0" ]
  },
  "algorithm": {
    "code_dir": "obs://test/economic_test/py_minist/",
    "boot_file": "obs://test/economic_test/py_minist/minist_common.py",
    "inputs": [ {
      "name": "data_url",
      "local_dir": "/home/ma-user/modelarts/inputs/data_url_0",
      "remote": {
        "obs": {
          "obs_url": "/test/data/py_minist/"
        }
      }
    } ],
    "outputs": [ {
      "name": "train_url",
```

```

"local_dir" : "/home/ma-user/modelarts/outputs/train_url_0",
"remote" : {
  "obs" : {
    "obs_url" : "/test/train_output/"
  }
},
"engine" : {
  "engine_id" : "pytorch-cp36-1.4.0-v2",
  "engine_name" : "PyTorch",
  "engine_version" : "PyTorch-1.4.0-python3.6-v2"
},
"spec" : {
  "resource" : {
    "flavor_id" : "modelarts.vm.p100.large.eco",
    "node_count" : 1,
    "flavor_detail" : {
      "flavor_type" : "GPU",
      "billing" : {
        "code" : "modelarts.vm.gpu.p100.eco",
        "unit_num" : 1
      },
      "flavor_info" : {
        "cpu" : {
          "arch" : "x86",
          "core_num" : 8
        },
        "gpu" : {
          "unit_num" : 1,
          "memory" : "8GB"
        },
        "memory" : {
          "size" : 64,
          "unit" : "GB"
        }
      }
    }
  }
}

```

Status Codes

Status Code	Description
200	ok

Error Codes

See [Error Codes](#).

5.8 Modifying the Description of a Training Job

Function

This API is used to modify the description of a training job.

URI

PUT /v2/{project_id}/training-jobs/{training_job_id}

Table 5-265 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .
training_job_id	Yes	String	ID of a training job.

Request Parameters

Table 5-266 Request body parameters

Parameter	Mandatory	Type	Description
description	No	String	Training job description, which consists of 0 to 256 characters. The default value is NULL .

Response Parameters

None

Example Requests

The following shows how to modify a training job with UUID **3faf5c03-aaa1-4cbe-879d-24b05d997347**. After the modification, call the API for obtaining a training job(ListTrainingJobs.xml) to view the modified description.

```
PUT https://endpoint/v2/{project_id}/training-jobs/3faf5c03-aaa1-4cbe-879d-24b05d997347
{
  "description" : "hahaha"
}
```

Example Responses

Status code: 200

No Content

```
null
```

Status Codes

Status Code	Description
200	No Content

Error Codes

See [Error Codes](#).

5.9 Deleting a Training Job

Function

This API is used to delete a training job.

URI

DELETE /v2/{project_id}/training-jobs/{training_job_id}

Table 5-267 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .
training_job_id	Yes	String	ID of a training job.

Request Parameters

None

Response Parameters

None

Example Requests

The following shows how to delete a training job whose UUID is **3faf5c03-aaa1-4cbe-879d-24b05d997347**.

```
DELETE https://endpoint/v2/{project_id}/training-jobs/3faf5c03-aaa1-4cbe-879d-24b05d997347
```

Example Responses

Status code: 202

No Content

""

Status Codes

Status Code	Description
202	No Content

Error Codes

See [Error Codes](#).

5.10 Terminating a Training Job

Function

This API is used to terminate a training job. Only jobs in the **Creating**, **Waiting**, or **Running** state can be terminated.

URI

POST /v2/{project_id}/training-jobs/{training_job_id}/actions

Table 5-268 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .
training_job_id	Yes	String	ID of a training job.

Request Parameters

Table 5-269 Request body parameters

Parameter	Mandatory	Type	Description
action_type	No	String	Operation request for a training job. If this parameter is set to terminate, the training job is terminated.

Response Parameters

Status code: 202

Table 5-270 Response body parameters

Parameter	Type	Description
kind	String	Training job type, which is job by default. Options: <ul style="list-style-type: none"> • job: training job
metadata	JobMetadata object	Metadata of a training job.
status	Status object	Status of a training job. You do not need to set this parameter when creating a job.
algorithm	JobAlgorithm Response object	Algorithm used by a training job. Options: <ul style="list-style-type: none"> • id: Only the algorithm ID is used. • subscription_id+item_version_id: The subscription ID and version ID of the algorithm are used. • code_dir+boot_file: The code directory and boot file of the training job are used.
tasks	Array of TaskResponse objects	List of tasks in heterogeneous training jobs.
spec	spec object	Specifications of a training job.

Table 5-271 JobMetadata

Parameter	Type	Description
id	String	Training job ID, which is generated and returned by ModelArts after the training job is created.
name	String	Name of a training job. The value must contain 1 to 64 characters consisting of only digits, letters, underscores (_), and hyphens (-).
workspace_id	String	Workspace where a job is located. The default value is 0 .
description	String	Training job description. The value must contain 0 to 256 characters. The default value is NULL .

Parameter	Type	Description
create_time	Long	Time when a training job was created, in milliseconds. The value is generated and returned by ModelArts after a training job is created.
user_name	String	Username for creating a training job. The username is generated and returned by ModelArts after a training job is created.
annotations	Map<String,String>	Advanced configuration of a training job. Options: <ul style="list-style-type: none"> ● job_template: Template RL (heterogeneous job) ● fault-tolerance/job-retry-num: 3 (number of retries upon a fault)

Table 5-272 Status

Parameter	Type	Description
phase	String	Level-1 status of a training job. The options are as follows: Creating Pending Running Failed Completed, Terminating Terminated Abnormal
secondary_phase	String	The level-2 status of a training job is an internal detailed status, which may be added, modified, or deleted. Dependency is not recommended. The options are as follows: Creating Queuing Running Failed Completed, Terminating Terminated CreateFailed TerminatedFailed Unknown Lost
duration	Long	Running duration of a training job, in milliseconds
node_count_metrics	Array<Array<Integer>>	Node count changes during the training job running period.
tasks	Array of strings	Tasks of a training job.
start_time	String	Start time of a training job. The value is in timestamp format.
task_statuses	Array of task_statuses objects	Status of a training job task.

Table 5-273 task_statuses

Parameter	Type	Description
task	String	Name of a training job task.
exit_code	Integer	Exit code of a training job task.
message	String	Error message of a training job task.

Table 5-274 JobAlgorithmResponse

Parameter	Type	Description
id	String	Algorithm used by a training job. Options: <ul style="list-style-type: none"> • id: Only the algorithm ID is used. • subscription_id+item_version_id: The subscription ID and version ID of the algorithm are used. • code_dir+boot_file: The code directory and boot file of the training job are used.
name	String	Algorithm name.
subscription_id	String	Subscription ID of a subscribed algorithm, which must be used with item_version_id
item_version_id	String	Version ID of the subscribed algorithm, which must be used with subscription_id
code_dir	String	Code directory of a training job, for example, /usr/app/ . This parameter must be used together with boot_file . If id or subscription_id+item_version_id is set, leave it blank.
boot_file	String	Boot file of a training job, which must be stored in the code directory, for example, /usr/app/boot.py . This parameter must be used with code_dir . Leave this parameter blank if id , or subscription_id and item_version_id are specified.
autosearch_config_path	String	YAML configuration path of auto search jobs. An OBS URL is required.
autosearch_framework_path	String	Framework code directory of auto search jobs. An OBS URL is required.
command	String	Boot command used to start the container of a custom image of a training job. For example, python train.py .

Parameter	Type	Description
parameters	Array of Parameter objects	Running parameters of a training job.
policies	policies object	Policies supported by jobs.
inputs	Array of Input objects	Input of a training job.
outputs	Array of Output objects	Output of a training job.
engine	engine object	Engine of a training job. Leave this parameter blank if the job is created using id of the algorithm in algorithm management, or subscription_id+item_version_id of the subscribed algorithm.
local_code_dir	String	Local directory to the training container to which the algorithm code directory is downloaded. Ensure that the following rules are complied with: - The directory must be in the /home directory. - In v1 compatibility mode, the current field does not take effect. - When code_dir is prefixed with file:// , the current field does not take effect.
working_dir	String	Work directory where an algorithm is executed. Note that this parameter does not take effect in v1 compatibility mode.
environments	Array of Map<String,String> objects	Environment variables of a training job. The format is key: value . Leave this parameter blank.

Table 5-275 Parameter

Parameter	Type	Description
name	String	Parameter name.
value	String	Parameter value.
description	String	Parameter description.
constraint	constraint object	Parameter constraint.
i18n_description	i18n_description object	Internationalization description.

Table 5-276 constraint

Parameter	Type	Description
type	String	Parameter type.
editable	Boolean	Whether the parameter is editable.
required	Boolean	Whether the parameter is mandatory.
sensitive	Boolean	Whether the parameter is sensitive This function is not implemented currently.
valid_type	String	Valid type.
valid_range	Array of strings	Valid range.

Table 5-277 i18n_description

Parameter	Type	Description
language	String	Language.
description	String	Description.

Table 5-278 policies

Parameter	Type	Description
auto_search	auto_search object	Hyperparameter search configuration.

Table 5-279 auto_search

Parameter	Type	Description
skip_search_parameters	String	Hyperparameter parameters that need to be skipped.
reward_attrs	Array of reward_attrs objects	List of search metrics.
search_parameters	Array of search_parameters objects	Search parameters.
algo_configs	Array of algo_configs objects	Search algorithm configurations.

Table 5-280 reward_attrs

Parameter	Type	Description
name	String	Metric name.
mode	String	Search direction. <ul style="list-style-type: none"> • max: A larger metric value indicates better performance. • min: A smaller metric value indicates better performance.
regex	String	Regular expression of a metric.

Table 5-281 search_params

Parameter	Type	Description
name	String	Hyperparameter name.
param_type	String	Parameter type <ul style="list-style-type: none"> • If continuous is specified, the hyperparameter is of the continuous type. When an algorithm is used in a training job, continuous hyperparameters are displayed as text boxes on the console. - discrete: The hyperparameter is of the discrete type. When an algorithm is used for training jobs, discrete hyperparameters are displayed as a drop-down list box on the console.
lower_bound	String	Lower bound of the hyperparameter.
upper_bound	String	Upper bound of the hyperparameter.
discrete_points_num	String	Number of discrete points of a continuous hyperparameter.
discrete_values	Array of strings	List of discrete hyperparameter values.

Table 5-282 algo_configs

Parameter	Type	Description
name	String	Name of the search algorithm.
params	Array of AutoSearchAlgorithmConfigParameter objects	Search algorithm parameters.

Table 5-283 AutoSearchAlgoConfigParameter

Parameter	Type	Description
key	String	Parameter key.
value	String	Parameter value.
type	String	Parameter type.

Table 5-284 Input

Parameter	Type	Description
name	String	Name of the data input channel.
description	String	Description of the data input channel.
local_dir	String	Local directory of the container to which the data input channel is mapped.
remote	InputDataInfo object	Data input. Options: <ul style="list-style-type: none"> • dataset: Dataset as the data input • obs: OBS path as the data input
remote_constraint	Array of remote_constraint objects	Data input constraint

Table 5-285 InputDataInfo

Parameter	Type	Description
dataset	dataset object	Dataset as the data input.
obs	obs object	OBS in which data input and output stored.

Table 5-286 dataset

Parameter	Type	Description
id	String	Dataset ID of a training job.
version_id	String	Dataset version ID of a training job.
obs_url	String	OBS URL of the dataset required by a training job. ModelArts automatically parses and generates the URL based on the dataset and dataset version IDs. For example, /usr/data/ .

Table 5-287 obs

Parameter	Type	Description
obs_url	String	OBS URL of the dataset required by a training job. For example, /usr/data/ .

Table 5-288 remote_constraint

Parameter	Type	Description
data_type	String	Data input type, including the data storage location and dataset.
attributes	String	Attributes if a dataset is used as the data input. Options: <ul style="list-style-type: none"> • data_format: Data format • data_segmentation: Data segmentation • dataset_type: Labeling type

Table 5-289 Output

Parameter	Type	Description
name	String	Name of the data output channel.
description	String	Description of the data output channel.
local_dir	String	Local directory of the container to which the data output channel is mapped.
remote	remote object	Description of the actual data output.

Table 5-290 remote

Parameter	Type	Description
obs	obs object	OBS to which data is actually exported.

Table 5-291 obs

Parameter	Type	Description
obs_url	String	OBS URL to which data is actually exported.

Table 5-292 engine

Parameter	Type	Description
engine_id	String	Engine ID selected for a training job. You can set this parameter to engine_id , engine_name + engine_version , or image_url .
engine_name	String	Name of the engine selected for a training job. If engine_id is set, leave this parameter blank.
engine_version	String	Name of the engine version selected for a training job. If engine_id is set, leave this parameter blank.
image_url	String	Custom image URL selected for a training job.

Table 5-293 TaskResponse

Parameter	Type	Description
role	String	Task role. This function is not supported currently.
algorithm	algorithm object	Algorithm management and configuration.
task_resource	FlavorResponse object	Flavors of a training job or an algorithm.

Table 5-294 algorithm

Parameter	Type	Description
code_dir	String	Absolute path of the directory where the algorithm boot file is stored.
boot_file	String	Absolute path of the algorithm boot file.
inputs	inputs object	Algorithm input channel.
outputs	outputs object	Algorithm output channel.
engine	engine object	Engine on which a heterogeneous job depends.

Parameter	Type	Description
local_code_dir	String	Local directory to the training container to which the algorithm code directory is downloaded. Ensure that the following rules are complied with: - The directory must be in the /home directory. - In v1 compatibility mode, the current field does not take effect. - When code_dir is prefixed with file:// , the current field does not take effect.
working_dir	String	Work directory where an algorithm is executed. Note that this parameter does not take effect in v1 compatibility mode.

Table 5-295 inputs

Parameter	Type	Description
name	String	Name of the data input channel.
local_dir	String	Local path of the container to which the data input and output channels are mapped.
remote	remote object	Actual data input. Heterogeneous jobs support only OBS.

Table 5-296 remote

Parameter	Type	Description
obs	obs object	OBS in which data input and output stored.

Table 5-297 obs

Parameter	Type	Description
obs_url	String	OBS URL of the dataset required by a training job. For example, /usr/data/ .

Table 5-298 outputs

Parameter	Type	Description
name	String	Name of the data output channel.

Parameter	Type	Description
local_dir	String	Local directory of the container to which the data output channel is mapped.
remote	remote object	Description of the actual data output.
mode	String	Data transmission mode. The default value is upload_periodically .
period	String	Data transmission period. The default value is 30s .

Table 5-299 remote

Parameter	Type	Description
obs	obs object	OBS to which data is actually exported.

Table 5-300 obs

Parameter	Type	Description
obs_url	String	OBS URL to which data is actually exported.

Table 5-301 engine

Parameter	Type	Description
engine_id	String	Engine ID of a heterogeneous job, for example, caffe-1.0.0-python2.7 .
engine_name	String	Engine name of a heterogeneous job, for example, Caffe .
engine_version	String	Engine version of a heterogeneous job.
v1_compatible	Boolean	Whether the v1 compatibility mode is used.
run_user	String	User UID started by default by the engine.
image_url	String	Custom image URL selected by an algorithm.

Table 5-302 FlavorResponse

Parameter	Type	Description
flavor_id	String	ID of the resource flavor.
flavor_name	String	Name of the resource flavor.
max_num	Integer	Maximum number of nodes in a resource flavor.
flavor_type	String	Resource flavor type. Options: <ul style="list-style-type: none"> • CPU • GPU
billing	billing object	Billing information of a resource flavor.
flavor_info	flavor_info object	Resource flavor details.
attributes	Map<String,String>	Other specification attributes.

Table 5-303 billing

Parameter	Type	Description
code	String	Billing code.
unit_num	Integer	Number of billing units.

Table 5-304 flavor_info

Parameter	Type	Description
max_num	Integer	Maximum number of nodes that can be selected. The value 1 indicates that the distributed mode is not supported.
cpu	cpu object	CPU specifications.
gpu	gpu object	GPU specifications.
npu	npu object	Ascend specifications
memory	memory object	Memory information.
disk	disk object	Disk information.

Table 5-305 cpu

Parameter	Type	Description
arch	String	CPU architecture.
core_num	Integer	Number of cores.

Table 5-306 gpu

Parameter	Type	Description
unit_num	Integer	Number of GPUs.
product_name	String	Product name.
memory	String	Memory.

Table 5-307 npu

Parameter	Type	Description
unit_num	String	Number of NPUs.
product_name	String	Product name.
memory	String	Memory.

Table 5-308 memory

Parameter	Type	Description
size	Integer	Memory size.
unit	String	Memory size

Table 5-309 disk

Parameter	Type	Description
size	Integer	Disk size.
unit	String	Unit of the disk size.

Table 5-310 spec

Parameter	Type	Description
resource	Resource object	Resource flavors of a training job. Select either flavor_id or pool_id+[flavor_id] .
volumes	Array of volumes objects	Volumes attached to a training job.
log_export_path	log_export_path object	Export path of training job logs.

Table 5-311 Resource

Parameter	Type	Description
policy	String	Resource flavor of a training job. Options: regular
flavor_id	String	ID of the resource flavor selected for a training job. flavor_id cannot be specified for dedicated resource pools with CPU specifications. The options for dedicated resource pools with GPU/Ascend specifications are as follows: <ul style="list-style-type: none"> • modelarts.pool.visual.xlarge (1 card) • modelarts.pool.visual.2xlarge (2 cards) • modelarts.pool.visual.4xlarge (4 cards) • modelarts.pool.visual.8xlarge (8 cards)
flavor_name	String	Read-only flavor name returned by ModelArts when flavor_id is used.
node_count	Integer	Number of resource replicas selected for a training job.
pool_id	String	Resource pool ID selected for a training job.
flavor_detail	flavor_detail object	Flavors of a training job or an algorithm.

Table 5-312 flavor_detail

Parameter	Type	Description
flavor_type	String	Resource flavor type. Options: <ul style="list-style-type: none"> • CPU • GPU

Parameter	Type	Description
billing	billing object	Billing information of a resource flavor.
flavor_info	flavor_info object	Resource flavor details.

Table 5-313 billing

Parameter	Type	Description
code	String	Billing code.
unit_num	Integer	Number of billing units.

Table 5-314 flavor_info

Parameter	Type	Description
max_num	Integer	Maximum number of nodes that can be selected. The value 1 indicates that the distributed mode is not supported.
cpu	cpu object	CPU specifications.
gpu	gpu object	GPU specifications.
npu	npu object	Ascend specifications
memory	memory object	Memory information.
disk	disk object	Disk information.

Table 5-315 cpu

Parameter	Type	Description
arch	String	CPU architecture.
core_num	Integer	Number of cores.

Table 5-316 gpu

Parameter	Type	Description
unit_num	Integer	Number of GPUs.
product_name	String	Product name.

Parameter	Type	Description
memory	String	Memory.

Table 5-317 npu

Parameter	Type	Description
unit_num	String	Number of NPUs.
product_name	String	Product name.
memory	String	Memory.

Table 5-318 memory

Parameter	Type	Description
size	Integer	Memory size.
unit	String	Number of memory units.

Table 5-319 disk

Parameter	Type	Description
size	String	Disk size.
unit	String	Unit of the disk size. Generally, the value is GB.

Table 5-320 volumes

Parameter	Type	Description
nfs	nfs object	Volumes attached in NFS mode.

Table 5-321 nfs

Parameter	Type	Description
nfs_server_path	String	NFS server path.
local_path	String	Path for attaching volumes to the training container.

Parameter	Type	Description
read_only	Boolean	Whether the volumes attached to the container in NFS mode are read-only.

Table 5-322 log_export_path

Parameter	Type	Description
obs_url	String	OBS URL for storing training job logs.
host_path	String	Path of the host where training job logs are stored.

Example Requests

The following is an example of how to stop the training job whose UUID is **3faf5c03-aaa1-4cbe-879d-24b05d997347**.

```
POST https://endpoint/v2/{project_id}/training-jobs/cf63aba9-63b1-4219-b717-708a2665100b/actions
{
  "action_type": "terminate"
}
```

Example Responses

Status code: 202

ok

```
{
  "kind": "job",
  "metadata": {
    "id": "cf63aba9-63b1-4219-b717-708a2665100b",
    "name": "trainjob--py14_mem06-110",
    "description": "",
    "create_time": 163651522282,
    "workspace_id": "0",
    "user_name": "ei_modelarts_z00424192_01"
  },
  "status": {
    "phase": "Terminating",
    "secondary_phase": "Terminating",
    "duration": 0,
    "start_time": 0,
    "node_count_metrics": null,
    "tasks": [ "worker-0" ]
  },
  "algorithm": {
    "code_dir": "obs://test/economic_test/py_minist/",
    "boot_file": "obs://test/economic_test/py_minist/minist_common.py",
    "inputs": [ {
      "name": "data_url",
      "local_dir": "/home/ma-user/modelarts/inputs/data_url_0",
      "remote": {
        "obs": {
          "obs_url": "test/data/py_minist/"
        }
      }
    }
  ]
}
```

```

    }
  },
  "outputs": [ {
    "name": "train_url",
    "local_dir": "/home/ma-user/modelarts/outputs/train_url_0",
    "remote": {
      "obs": {
        "obs_url": "/test/train_output/"
      }
    }
  } ],
  "engine": {
    "engine_id": "pytorch-cp36-1.4.0-v2",
    "engine_name": "PyTorch",
    "engine_version": "PyTorch-1.4.0-python3.6-v2"
  },
  "spec": {
    "resource": {
      "policy": "economic",
      "flavor_id": "modelarts.vm.p100.large.eco",
      "flavor_name": "Computing GPU(P100) instance",
      "node_count": 1,
      "flavor_detail": {
        "flavor_type": "GPU",
        "billing": {
          "code": "modelarts.vm.gpu.p100.eco",
          "unit_num": 1
        },
        "flavor_info": {
          "cpu": {
            "arch": "x86",
            "core_num": 8
          },
          "gpu": {
            "unit_num": 1,
            "product_name": "NVIDIA-P100",
            "memory": "8GB"
          },
          "memory": {
            "size": 64,
            "unit": "GB"
          }
        }
      }
    }
  }
}

```

Status Codes

Status Code	Description
202	ok

Error Codes

See [Error Codes](#).

5.11 Querying the Logs of a Specified Task in a Given Training Job (Preview)

Function

This API is used to query the logs of a specified task in a given training job (preview).

URI

GET /v2/{project_id}/training-jobs/{training_job_id}/tasks/{task_id}/logs/preview

Table 5-323 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .
training_job_id	Yes	String	ID of a training job.
task_id	Yes	String	Name of a training job. You can obtain the value from the status.tasks field in the training job details.

Request Parameters

None

Response Parameters

Status code: 200

Table 5-324 Response body parameters

Parameter	Type	Description
content	String	Log content. If the size of a log file does not exceed the upper limit (n MB), all the log files are returned. Otherwise, the latest log file (n MB) is returned. After 2022/03/01 00:00:00 (GMT+08:00), the parameter name is changed from context to content.
current_size	Integer	Size of the returned log content, in bytes. The maximum size is 5 MB.

Parameter	Type	Description
full_size	Integer	Size of complete log content, in bytes

Example Requests

The following shows how to obtain the worker-0 logs of the training job whose UUID is **2cd88daa-31a4-40a8-a58f-d186b0e93e4f**.

```
GET https://endpoint/v2/{project_id}/training-jobs/2cd88daa-31a4-40a8-a58f-d186b0e93e4f/tasks/worker-0/logs/preview
```

Example Responses

Status code: 200

ok

```
{
  "content": "[Modelarts Service Log]collect and upload ascend logs end at 2021-05-18-14:28:13\n[Modelarts Service Log]exiting... \n [Modelarts Service Log]exiting...\n[Modelarts Service Log]exiting...: \n [Modelarts Service Log]exiting...\n[Modelarts Service Log]exit with : \n [Modelarts Service Log]exit with 0\n[Modelarts Service Log]exit with : \n [Modelarts Service Log]exit with 0\n[ModelArts Service Log][INFO][2021/05/18 14:28:14,207]:\n output-handler finalizing due to: [training finished]\n[ModelArts Service Log][INFO][2021/05/18 14:28:14,207]:\n output-handler finalized\n[Modelarts Service Log][sidecar] exiting at 2021-05-18-14:28:14\n[Modelarts Service Log][sidecar] wait python processes exit... \n [Modelarts Service Log][sidecar] wait python processes exit...\n[Modelarts Service Log][sidecar] exit with : \n [Modelarts Service Log][sidecar] exit with 0",
  "current_size": 126548,
  "full_size": 5242880
}
```

Status Codes

Status Code	Description
200	ok

Error Codes

See [Error Codes](#).

5.12 Querying the Logs of a Specified Task in a Training Job (OBS Link)

Function

This API is used to obtain the logs of a specified task of a training job (temporary OBS link, which is valid for 5 minutes). You can view all logs or download the logs.

URI

GET /v2/{project_id}/training-jobs/{training_job_id}/tasks/{task_id}/logs/url

Table 5-325 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .
training_job_id	Yes	String	ID of a training job.
task_id	Yes	String	Name of a training job. You can obtain the value from the status.tasks field in the training job details.

Request Parameters

Table 5-326 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	No	String	Message body type. If this parameter is set to text/plain, the temporary preview link is returned. Set this parameter to application/octet-stream. A temporary download link is returned.

Response Parameters

Status code: 200

Table 5-327 Response body parameters

Parameter	Type	Description
obs_url	String	Temporary OBS URL of logs. You can copy the URL to the browser to view the current complete logs.

Example Requests

The following shows how to query the temporary OBS URL for the **work-0** tasks of the training job whose UUID is **2cd88daa-31a4-40a8-a58f-d186b0e93e4f**.

```
GET https://endpoint/v2/{project_id}/training-jobs/2cd88daa-31a4-40a8-a58f-d186b0e93e4f/tasks/worker-0/logs/url?Content-Type=text/plain
```

Example Responses

Status code: 200

ok

```
{
  "obs_url" : "http://10.155.101.248:20000/xxxxxx-test/xk/00chess_test/test11/logs/modelarts-job-0f2ccdbb-4f34-4d53-afb9-d526f3be8c68-ma-platform-init-worker-0-172.16.24.51-01909681.log?AWSAccessKeyId=xxxxx"
}
```

Status Codes

Status Code	Description
200	ok

Error Codes

See [Error Codes](#).

5.13 Querying the Running Metrics of a Specified Task in a Training Job

Function

This API is used to query the running metrics of a specified task in a training job.

URI

```
GET /v2/{project_id}/training-jobs/{training_job_id}/metrics/{task_id}
```

Table 5-328 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .
training_job_id	Yes	String	ID of a training job.

Parameter	Mandatory	Type	Description
task_id	Yes	String	Name of a training job. You can obtain the value from the status.tasks field in the training job details.

Request Parameters

None

Response Parameters

Status code: 200

Table 5-329 Response body parameters

Parameter	Type	Description
metrics	Array of metrics objects	Running metrics.

Table 5-330 metrics

Parameter	Type	Description
metric	String	Running metric. The options are as follows: <ul style="list-style-type: none"> cpuUsage: CPU usage memUsage: indicates the physical memory usage. gpuUtil: GPU usage gpuMemUsage: GPU memory usage npuUtil: NPU usage npuMemUsage: NPU GPU memory usage
value	Array of numbers	Value of a running metric. An average value is collected every minute.

Example Requests

The following shows how to query the running metrics of the **work-0** task of the training job whose UUID is **2cd88daa-31a4-40a8-a58f-d186b0e93e4f**.

```
GET https://endpoint/v2/{project_id}/training-jobs/2cd88daa-31a4-40a8-a58f-d186b0e93e4f/metrics/worker-0
```

Example Responses

Status code: 200

ok

```
{
  "metrics": [ {
    "metric": "cpuUsage",
    "value": [ -1, -1, 2.43, 4.524, 6.714, 12.422, 9.214, 5.36, 7.5, 10.088, 8.975, 11.423, 11.548, 14.563,
16.833 ]
  }, {
    "metric": "memUsage",
    "value": [ -1, -1, 0.04, 0.521, 1.652, 4.252, 6.433, 7.384, 7.982, 8.718, 9.365, 9.881, 10.192, 9.994, 9.005 ]
  }, {
    "metric": "gpuUtil",
    "value": [ -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1 ]
  }, {
    "metric": "gpuMemUsage",
    "value": [ -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1 ]
  }, {
    "metric": "npuUtil",
    "value": [ -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1 ]
  }, {
    "metric": "npuMemUsage",
    "value": [ -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1 ]
  }
]
```

Status Codes

Status Code	Description
200	ok

Error Codes

See [Error Codes](#).

5.14 Querying a Training Job List

Function

This API is used to query the the created training jobs that meet the search criteria.

URI

POST /v2/{project_id}/training-job-searches

Table 5-331 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .

Request Parameters

Table 5-332 Request body parameters

Parameter	Mandatory	Type	Description
offset	No	Integer	Offset for querying jobs. The minimum value is 0 . For example, if this parameter is set to 1 , the query starts from the second one.
limit	No	Integer	Maximum number of jobs to be queried. The value ranges from 1 to 50.
sort_by	No	String	Metric for sorting jobs to be queried. create_time is used by default for sorting.
order	No	String	Order of queried jobs. The default value is desc , indicating the descending order. You can also set this parameter to asc , indicating the ascending order.
group_by	No	String	Condition for grouping the jobs to be queried.
filters	No	Array of filters objects	Filters for querying jobs.

Table 5-333 filters

Parameter	Mandatory	Type	Description
key	No	String	Grouping condition key.

Parameter	Mandatory	Type	Description
operator	No	String	Grouping condition key-value relationship. The options are between (range), like (similar), in (included), and not (not).
value	No	Array of strings	Value of the grouping condition key.

Response Parameters

Status code: 200

Table 5-334 Response body parameters

Parameter	Type	Description
total	Integer	Total number of queried jobs of the current user.
count	Integer	Total number of jobs that meet the search criteria of the current user.
limit	Integer	Maximum number of jobs to be queried. The value ranges from 1 to 50.
offset	Integer	Offset for querying jobs. The minimum value is 0 . For example, if this parameter is set to 1 , the query starts from the second one.
sort_by	String	Metric for sorting jobs to be queried. create_time is used by default for sorting.
order	String	Order of queried jobs. The default value is desc , indicating the descending order. You can also set this parameter to asc , indicating the ascending order.
group_by	String	Condition for grouping the jobs to be queried.
workspace_id	String	Workspace where a job is located. The default value is 0 .
ai_project	String	AI project to which a job belongs. The default value is default-ai-project .
items	Array of JobResponse objects	Jobs that meet the search criteria of the current user.

Table 5-335 JobResponse

Parameter	Type	Description
kind	String	Training job type, which is job by default. Options: <ul style="list-style-type: none"> • job: training job
metadata	JobMetadata object	Metadata of a training job.
status	Status object	Status of a training job. You do not need to set this parameter when creating a job.
algorithm	JobAlgorithm Response object	Algorithm used by a training job. Options: <ul style="list-style-type: none"> • id: Only the algorithm ID is used. • subscription_id+item_version_id: The subscription ID and version ID of the algorithm are used. • code_dir+boot_file: The code directory and boot file of the training job are used.
tasks	Array of TaskResponse objects	List of tasks in heterogeneous training jobs.
spec	spec object	Specifications of a training job.

Table 5-336 JobMetadata

Parameter	Type	Description
id	String	Training job ID, which is generated and returned by ModelArts after the training job is created.
name	String	Name of a training job. The value must contain 1 to 64 characters consisting of only digits, letters, underscores (_), and hyphens (-).
workspace_id	String	Workspace where a job is located. The default value is 0 .
description	String	Training job description. The value must contain 0 to 256 characters. The default value is NULL .
create_time	Long	Time when a training job was created, in milliseconds. The value is generated and returned by ModelArts after a training job is created.

Parameter	Type	Description
user_name	String	Username for creating a training job. The username is generated and returned by ModelArts after a training job is created.
annotations	Map<String,String>	Advanced configuration of a training job. Options: <ul style="list-style-type: none"> • job_template: Template RL (heterogeneous job) • fault-tolerance/job-retry-num: 3 (number of retries upon a fault)

Table 5-337 Status

Parameter	Type	Description
phase	String	Level-1 status of a training job. The options are as follows: Creating Pending Running Failed Completed, Terminating Terminated Abnormal
secondary_phase	String	The level-2 status of a training job is an internal detailed status, which may be added, modified, or deleted. Dependency is not recommended. The options are as follows: Creating Queuing Running Failed Completed, Terminating Terminated CreateFailed TerminatedFailed Unknown Lost
duration	Long	Running duration of a training job, in milliseconds
node_count_metrics	Array<Array<Integer>>	Node count changes during the training job running period.
tasks	Array of strings	Tasks of a training job.
start_time	String	Start time of a training job. The value is in timestamp format.
task_statuses	Array of task_statuses objects	Status of a training job task.

Table 5-338 task_statuses

Parameter	Type	Description
task	String	Name of a training job task.
exit_code	Integer	Exit code of a training job task.
message	String	Error message of a training job task.

Table 5-339 JobAlgorithmResponse

Parameter	Type	Description
id	String	Algorithm used by a training job. Options: <ul style="list-style-type: none"> • id: Only the algorithm ID is used. • subscription_id+item_version_id: The subscription ID and version ID of the algorithm are used. • code_dir+boot_file: The code directory and boot file of the training job are used.
name	String	Algorithm name.
subscription_id	String	Subscription ID of a subscribed algorithm, which must be used with item_version_id
item_version_id	String	Version ID of the subscribed algorithm, which must be used with subscription_id
code_dir	String	Code directory of a training job, for example, /usr/app/ . This parameter must be used together with boot_file . If id or subscription_id+item_version_id is set, leave it blank.
boot_file	String	Boot file of a training job, which must be stored in the code directory, for example, /usr/app/boot.py . This parameter must be used with code_dir . Leave this parameter blank if id , or subscription_id and item_version_id are specified.
autosearch_config_path	String	YAML configuration path of auto search jobs. An OBS URL is required.
autosearch_framework_path	String	Framework code directory of auto search jobs. An OBS URL is required.
command	String	Boot command used to start the container of a custom image of a training job. For example, python train.py .

Parameter	Type	Description
parameters	Array of Parameter objects	Running parameters of a training job.
policies	policies object	Policies supported by jobs.
inputs	Array of Input objects	Input of a training job.
outputs	Array of Output objects	Output of a training job.
engine	engine object	Engine of a training job. Leave this parameter blank if the job is created using id of the algorithm in algorithm management, or subscription_id+item_version_id of the subscribed algorithm.
local_code_dir	String	Local directory to the training container to which the algorithm code directory is downloaded. Ensure that the following rules are complied with: - The directory must be in the /home directory. - In v1 compatibility mode, the current field does not take effect. - When code_dir is prefixed with file:// , the current field does not take effect.
working_dir	String	Work directory where an algorithm is executed. Note that this parameter does not take effect in v1 compatibility mode.
environments	Array of Map<String,String> objects	Environment variables of a training job. The format is key: value . Leave this parameter blank.

Table 5-340 Parameter

Parameter	Type	Description
name	String	Parameter name.
value	String	Parameter value.
description	String	Parameter description.
constraint	constraint object	Parameter constraint.
i18n_description	i18n_description object	Internationalization description.

Table 5-341 constraint

Parameter	Type	Description
type	String	Parameter type.
editable	Boolean	Whether the parameter is editable.
required	Boolean	Whether the parameter is mandatory.
sensitive	Boolean	Whether the parameter is sensitive This function is not implemented currently.
valid_type	String	Valid type.
valid_range	Array of strings	Valid range.

Table 5-342 i18n_description

Parameter	Type	Description
language	String	Language.
description	String	Description.

Table 5-343 policies

Parameter	Type	Description
auto_search	auto_search object	Hyperparameter search configuration.

Table 5-344 auto_search

Parameter	Type	Description
skip_search_parameters	String	Hyperparameter parameters that need to be skipped.
reward_attrs	Array of reward_attrs objects	List of search metrics.
search_parameters	Array of search_parameters objects	Search parameters.
algo_configs	Array of algo_configs objects	Search algorithm configurations.

Table 5-345 reward_attrs

Parameter	Type	Description
name	String	Metric name.
mode	String	Search direction. <ul style="list-style-type: none"> • max: A larger metric value indicates better performance. • min: A smaller metric value indicates better performance.
regex	String	Regular expression of a metric.

Table 5-346 search_params

Parameter	Type	Description
name	String	Hyperparameter name.
param_type	String	Parameter type <ul style="list-style-type: none"> • If continuous is specified, the hyperparameter is of the continuous type. When an algorithm is used in a training job, continuous hyperparameters are displayed as text boxes on the console. - discrete: The hyperparameter is of the discrete type. When an algorithm is used for training jobs, discrete hyperparameters are displayed as a drop-down list box on the console.
lower_bound	String	Lower bound of the hyperparameter.
upper_bound	String	Upper bound of the hyperparameter.
discrete_points_num	String	Number of discrete points of a continuous hyperparameter.
discrete_values	Array of strings	List of discrete hyperparameter values.

Table 5-347 algo_configs

Parameter	Type	Description
name	String	Name of the search algorithm.
params	Array of AutoSearchAlgorithmConfigParameter objects	Search algorithm parameters.

Table 5-348 AutoSearchAlgoConfigParameter

Parameter	Type	Description
key	String	Parameter key.
value	String	Parameter value.
type	String	Parameter type.

Table 5-349 Input

Parameter	Type	Description
name	String	Name of the data input channel.
description	String	Description of the data input channel.
local_dir	String	Local directory of the container to which the data input channel is mapped.
remote	InputDataInfo object	Data input. Options: <ul style="list-style-type: none"> • dataset: Dataset as the data input • obs: OBS path as the data input
remote_constraint	Array of remote_constraint objects	Data input constraint

Table 5-350 InputDataInfo

Parameter	Type	Description
dataset	dataset object	Dataset as the data input.
obs	obs object	OBS in which data input and output stored.

Table 5-351 dataset

Parameter	Type	Description
id	String	Dataset ID of a training job.
version_id	String	Dataset version ID of a training job.
obs_url	String	OBS URL of the dataset required by a training job. ModelArts automatically parses and generates the URL based on the dataset and dataset version IDs. For example, /usr/data/ .

Table 5-352 obs

Parameter	Type	Description
obs_url	String	OBS URL of the dataset required by a training job. For example, /usr/data/ .

Table 5-353 remote_constraint

Parameter	Type	Description
data_type	String	Data input type, including the data storage location and dataset.
attributes	String	Attributes if a dataset is used as the data input. Options: <ul style="list-style-type: none"> • data_format: Data format • data_segmentation: Data segmentation • dataset_type: Labeling type

Table 5-354 Output

Parameter	Type	Description
name	String	Name of the data output channel.
description	String	Description of the data output channel.
local_dir	String	Local directory of the container to which the data output channel is mapped.
remote	remote object	Description of the actual data output.

Table 5-355 remote

Parameter	Type	Description
obs	obs object	OBS to which data is actually exported.

Table 5-356 obs

Parameter	Type	Description
obs_url	String	OBS URL to which data is actually exported.

Table 5-357 engine

Parameter	Type	Description
engine_id	String	Engine ID selected for a training job. You can set this parameter to engine_id , engine_name + engine_version , or image_url .
engine_name	String	Name of the engine selected for a training job. If engine_id is set, leave this parameter blank.
engine_version	String	Name of the engine version selected for a training job. If engine_id is set, leave this parameter blank.
image_url	String	Custom image URL selected for a training job.

Table 5-358 TaskResponse

Parameter	Type	Description
role	String	Task role. This function is not supported currently.
algorithm	algorithm object	Algorithm management and configuration.
task_resource	FlavorResponse object	Flavors of a training job or an algorithm.

Table 5-359 algorithm

Parameter	Type	Description
code_dir	String	Absolute path of the directory where the algorithm boot file is stored.
boot_file	String	Absolute path of the algorithm boot file.
inputs	inputs object	Algorithm input channel.
outputs	outputs object	Algorithm output channel.
engine	engine object	Engine on which a heterogeneous job depends.

Parameter	Type	Description
local_code_dir	String	Local directory to the training container to which the algorithm code directory is downloaded. Ensure that the following rules are complied with: - The directory must be in the /home directory. - In v1 compatibility mode, the current field does not take effect. - When code_dir is prefixed with file:// , the current field does not take effect.
working_dir	String	Work directory where an algorithm is executed. Note that this parameter does not take effect in v1 compatibility mode.

Table 5-360 inputs

Parameter	Type	Description
name	String	Name of the data input channel.
local_dir	String	Local path of the container to which the data input and output channels are mapped.
remote	remote object	Actual data input. Heterogeneous jobs support only OBS.

Table 5-361 remote

Parameter	Type	Description
obs	obs object	OBS in which data input and output stored.

Table 5-362 obs

Parameter	Type	Description
obs_url	String	OBS URL of the dataset required by a training job. For example, /usr/data/ .

Table 5-363 outputs

Parameter	Type	Description
name	String	Name of the data output channel.

Parameter	Type	Description
local_dir	String	Local directory of the container to which the data output channel is mapped.
remote	remote object	Description of the actual data output.
mode	String	Data transmission mode. The default value is upload_periodically .
period	String	Data transmission period. The default value is 30s .

Table 5-364 remote

Parameter	Type	Description
obs	obs object	OBS to which data is actually exported.

Table 5-365 obs

Parameter	Type	Description
obs_url	String	OBS URL to which data is actually exported.

Table 5-366 engine

Parameter	Type	Description
engine_id	String	Engine ID of a heterogeneous job, for example, caffe-1.0.0-python2.7 .
engine_name	String	Engine name of a heterogeneous job, for example, Caffe .
engine_version	String	Engine version of a heterogeneous job.
v1_compatible	Boolean	Whether the v1 compatibility mode is used.
run_user	String	User UID started by default by the engine.
image_url	String	Custom image URL selected by an algorithm.

Table 5-367 FlavorResponse

Parameter	Type	Description
flavor_id	String	ID of the resource flavor.
flavor_name	String	Name of the resource flavor.
max_num	Integer	Maximum number of nodes in a resource flavor.
flavor_type	String	Resource flavor type. Options: <ul style="list-style-type: none"> • CPU • GPU
billing	billing object	Billing information of a resource flavor.
flavor_info	flavor_info object	Resource flavor details.
attributes	Map<String,String>	Other specification attributes.

Table 5-368 billing

Parameter	Type	Description
code	String	Billing code.
unit_num	Integer	Number of billing units.

Table 5-369 flavor_info

Parameter	Type	Description
max_num	Integer	Maximum number of nodes that can be selected. The value 1 indicates that the distributed mode is not supported.
cpu	cpu object	CPU specifications.
gpu	gpu object	GPU specifications.
npu	npu object	Ascend specifications
memory	memory object	Memory information.
disk	disk object	Disk information.

Table 5-370 cpu

Parameter	Type	Description
arch	String	CPU architecture.
core_num	Integer	Number of cores.

Table 5-371 gpu

Parameter	Type	Description
unit_num	Integer	Number of GPUs.
product_name	String	Product name.
memory	String	Memory.

Table 5-372 npu

Parameter	Type	Description
unit_num	String	Number of NPUs.
product_name	String	Product name.
memory	String	Memory.

Table 5-373 memory

Parameter	Type	Description
size	Integer	Memory size.
unit	String	Memory size

Table 5-374 disk

Parameter	Type	Description
size	Integer	Disk size.
unit	String	Unit of the disk size.

Table 5-375 spec

Parameter	Type	Description
resource	Resource object	Resource flavors of a training job. Select either flavor_id or pool_id+[flavor_id] .
volumes	Array of volumes objects	Volumes attached to a training job.
log_export_path	log_export_path object	Export path of training job logs.

Table 5-376 Resource

Parameter	Type	Description
policy	String	Resource flavor of a training job. Options: regular
flavor_id	String	ID of the resource flavor selected for a training job. flavor_id cannot be specified for dedicated resource pools with CPU specifications. The options for dedicated resource pools with GPU/Ascend specifications are as follows: <ul style="list-style-type: none"> • modelarts.pool.visual.xlarge (1 card) • modelarts.pool.visual.2xlarge (2 cards) • modelarts.pool.visual.4xlarge (4 cards) • modelarts.pool.visual.8xlarge (8 cards)
flavor_name	String	Read-only flavor name returned by ModelArts when flavor_id is used.
node_count	Integer	Number of resource replicas selected for a training job.
pool_id	String	Resource pool ID selected for a training job.
flavor_detail	flavor_detail object	Flavors of a training job or an algorithm.

Table 5-377 flavor_detail

Parameter	Type	Description
flavor_type	String	Resource flavor type. Options: <ul style="list-style-type: none"> • CPU • GPU

Parameter	Type	Description
billing	billing object	Billing information of a resource flavor.
flavor_info	flavor_info object	Resource flavor details.

Table 5-378 billing

Parameter	Type	Description
code	String	Billing code.
unit_num	Integer	Number of billing units.

Table 5-379 flavor_info

Parameter	Type	Description
max_num	Integer	Maximum number of nodes that can be selected. The value 1 indicates that the distributed mode is not supported.
cpu	cpu object	CPU specifications.
gpu	gpu object	GPU specifications.
npu	npu object	Ascend specifications
memory	memory object	Memory information.
disk	disk object	Disk information.

Table 5-380 cpu

Parameter	Type	Description
arch	String	CPU architecture.
core_num	Integer	Number of cores.

Table 5-381 gpu

Parameter	Type	Description
unit_num	Integer	Number of GPUs.
product_name	String	Product name.

Parameter	Type	Description
memory	String	Memory.

Table 5-382 npu

Parameter	Type	Description
unit_num	String	Number of NPUs.
product_name	String	Product name.
memory	String	Memory.

Table 5-383 memory

Parameter	Type	Description
size	Integer	Memory size.
unit	String	Number of memory units.

Table 5-384 disk

Parameter	Type	Description
size	String	Disk size.
unit	String	Unit of the disk size. Generally, the value is GB.

Table 5-385 volumes

Parameter	Type	Description
nfs	nfs object	Volumes attached in NFS mode.

Table 5-386 nfs

Parameter	Type	Description
nfs_server_path	String	NFS server path.
local_path	String	Path for attaching volumes to the training container.

Parameter	Type	Description
read_only	Boolean	Whether the volumes attached to the container in NFS mode are read-only.

Table 5-387 log_export_path

Parameter	Type	Description
obs_url	String	OBS URL for storing training job logs.
host_path	String	Path of the host where training job logs are stored.

Example Requests

The following is an example of how to obtain training jobs. The number of obtained training jobs has been limited to **1**, and the system will only query data for training jobs with names containing **trainjob**.

POST https://endpoint/v2/{project_id}/training-job-searches?limit=1

```
{
  "offset" : 0,
  "limit" : 1,
  "filters" : [ {
    "key" : "name",
    "operator" : "like",
    "value" : [ "trainjob" ]
  }, {
    "key" : "create_time",
    "operator" : "between",
    "value" : [ "", "" ]
  }, {
    "key" : "phase",
    "operator" : "in",
    "value" : [ "" ]
  }, {
    "key" : "algorithm_name",
    "operator" : "like",
    "value" : [ "" ]
  }, {
    "key" : "kind",
    "operator" : "in",
    "value" : [ ]
  }, {
    "key" : "user_id",
    "operator" : "in",
    "value" : [ "" ]
  }
]
```

Example Responses

Status code: 200

ok

```
{
  "total" : 5059,
```

```

"count" : 1,
"limit" : 1,
"offset" : 0,
"sort_by" : "create_time",
"order" : "desc",
"group_by" : "",
"workspace_id" : "0",
"ai_project" : "default-ai-project",
"items" : [ {
  "kind" : "job",
  "metadata" : {
    "id" : "3faf5c03-aaa1-4cbe-879d-24b05d997347",
    "name" : "trainjob--py14_mem06-byd-108",
    "description" : "",
    "create_time" : 1636447346315,
    "workspace_id" : "0",
    "user_name" : "ei_modelarts_q00357245_01"
  },
  "status" : {
    "phase" : "Abnormal",
    "secondary_phase" : "CreateFailed",
    "duration" : 0,
    "start_time" : 0,
    "node_count_metrics" : [ [ 1636447746000, 0 ], [ 1636447755000, 0 ], [ 1636447756000, 0 ] ],
    "tasks" : [ "worker-0" ]
  },
  "algorithm" : {
    "code_dir" : "obs://test-crq/economic_test/py_minist/",
    "boot_file" : "obs://test-crq/economic_test/py_minist/minist_common.py",
    "inputs" : [ {
      "name" : "data_url",
      "local_dir" : "/home/ma-user/modelarts/inputs/data_url_0",
      "remote" : {
        "obs" : {
          "obs_url" : "/test-crq/data/py_minist/"
        }
      }
    } ],
    "outputs" : [ {
      "name" : "train_url",
      "local_dir" : "/home/ma-user/modelarts/outputs/train_url_0",
      "remote" : {
        "obs" : {
          "obs_url" : "/test-crq/train_output/"
        }
      }
    } ],
    "engine" : {
      "engine_id" : "pytorch-cp36-1.4.0-v2",
      "engine_name" : "PyTorch",
      "engine_version" : "PyTorch-1.4.0-python3.6-v2"
    }
  },
  "spec" : {
    "resource" : {
      "policy" : "economic",
      "flavor_id" : "modelarts.vm.p100.large.eco",
      "flavor_name" : "Computing GPU(P100) instance",
      "node_count" : 1,
      "flavor_detail" : {
        "flavor_type" : "GPU",
        "billing" : {
          "code" : "modelarts.vm.gpu.p100.eco",
          "unit_num" : 1
        }
      },
      "flavor_info" : {
        "cpu" : {
          "arch" : "x86",
          "core_num" : 8
        }
      }
    }
  }
} ]

```

```
  },
  "gpu" : {
    "unit_num" : 1,
    "product_name" : "NVIDIA-P100",
    "memory" : "8GB"
  },
  "memory" : {
    "size" : 64,
    "unit" : "GB"
  }
}
}
```

Status Codes

Status Code	Description
200	ok

Error Codes

See [Error Codes](#).

5.15 Obtaining the General Specifications Supported by a Training Job

Function

This API is used to obtain the public flavors supported by a training job.

URI

GET /v2/{project_id}/training-job-flavors

Table 5-388 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .

Table 5-389 Query Parameters

Parameter	Mandatory	Type	Description
flavor_type	No	String	This API is used to obtain training job flavors. If this parameter is not specified, all flavors will be obtained. Options: <ul style="list-style-type: none"> • CPU • GPU

Request Parameters

None

Response Parameters

Status code: 200

Table 5-390 Response body parameters

Parameter	Type	Description
total_count	Integer	Total number of resource flavors of a training job.
flavors	Array of FlavorResponse objects	List of resource flavors of a training job.

Table 5-391 FlavorResponse

Parameter	Type	Description
flavor_id	String	ID of the resource flavor.
flavor_name	String	Name of the resource flavor.
max_num	Integer	Maximum number of nodes in a resource flavor.
flavor_type	String	Resource flavor type. Options: <ul style="list-style-type: none"> • CPU • GPU
billing	billing object	Billing information of a resource flavor.
flavor_info	flavor_info object	Resource flavor details.

Parameter	Type	Description
attributes	Map<String,String>	Other specification attributes.

Table 5-392 billing

Parameter	Type	Description
code	String	Billing code.
unit_num	Integer	Number of billing units.

Table 5-393 flavor_info

Parameter	Type	Description
max_num	Integer	Maximum number of nodes that can be selected. The value 1 indicates that the distributed mode is not supported.
cpu	cpu object	CPU specifications.
gpu	gpu object	GPU specifications.
npu	npu object	Ascend specifications
memory	memory object	Memory information.
disk	disk object	Disk information.

Table 5-394 cpu

Parameter	Type	Description
arch	String	CPU architecture.
core_num	Integer	Number of cores.

Table 5-395 gpu

Parameter	Type	Description
unit_num	Integer	Number of GPUs.
product_name	String	Product name.
memory	String	Memory.

Table 5-396 npu

Parameter	Type	Description
unit_num	String	Number of NPUs.
product_name	String	Product name.
memory	String	Memory.

Table 5-397 memory

Parameter	Type	Description
size	Integer	Memory size.
unit	String	Memory size

Table 5-398 disk

Parameter	Type	Description
size	Integer	Disk size.
unit	String	Unit of the disk size.

Example Requests

The following shows how to query the public CPU resource flavors of training jobs.

```
GET https://endpoint/v2/{project_id}/training-job-flavors?flavor_type=CPU
```

Example Responses

Status code: 200

ok

```
{
  "total_count": 2,
  "flavors": [ {
    "flavor_id": "modelarts.vm.cpu.2u",
    "flavor_name": "Computing CPU(2U) instance",
    "flavor_type": "CPU",
    "billing": {
      "code": "modelarts.vm.cpu.2u",
      "unit_num": 1
    },
    "flavor_info": {
      "max_num": 1,
      "cpu": {
        "arch": "x86",
```

```

    "core_num" : 2
  },
  "memory" : {
    "size" : 8,
    "unit" : "GB"
  },
  "disk" : {
    "size" : 50,
    "unit" : "GB"
  }
}, {
  "flavor_id" : "modelarts.vm.cpu.8u",
  "flavor_name" : "Computing CPU(8U) instance",
  "flavor_type" : "CPU",
  "billing" : {
    "code" : "modelarts.vm.cpu.8u",
    "unit_num" : 1
  },
  "flavor_info" : {
    "max_num" : 16,
    "cpu" : {
      "arch" : "x86",
      "core_num" : 8
    },
    "memory" : {
      "size" : 32,
      "unit" : "GB"
    },
    "disk" : {
      "size" : 50,
      "unit" : "GB"
    }
  }
}
}
]
}

```

Status Codes

Status Code	Description
200	ok

Error Codes

See [Error Codes](#).

5.16 Obtaining the Preset AI Frameworks Supported by a Training Job

Function

This API is used to obtain the preset AI frameworks supported by a training job.

URI

GET /v2/{project_id}/training-job-engines

Table 5-399 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .

Request Parameters

None

Response Parameters

Status code: 200

Table 5-400 Response body parameters

Parameter	Type	Description
total	Integer	Total number of training job engines.
items	Array of items objects	List of engine specifications.

Table 5-401 items

Parameter	Type	Description
engine_id	String	Engine ID, for example, caffe-1.0.0-python2.7 .
engine_name	String	Engine name, for example, Caffe .
engine_version	String	Engine version. Engines with the same name have multiple versions, for example, Caffe-1.0.0-python2.7 of Python 2.7.
v1_compatible	Boolean	Whether the v1 compatibility mode is used.
run_user	String	User UID started by default by the engine.
image_info	image_info object	Engine information.

Table 5-402 image_info

Parameter	Type	Description
cpu_image_url	String	Image with the matched CPU specifications.
gpu_image_url	String	Image with the matched GPU flavors
image_version	String	Image version.

Example Requests

The following shows how to query all public engine specifications of a training job(only part of the specifications are displayed because there are too many engines).

GET https://endpoint/v2/{project_id}/training-job-engines

Example Responses

Status code: 200

ok

```
{
  "total": 20,
  "items": [ {
    "engine_id": "caffe-1.0.0-python2.7",
    "engine_name": "Caffe",
    "engine_version": "caffe-1.0.0-python2.7",
    "v1_compatible": true,
    "run_user": "",
    "image_info": {
      "cpu_image_url": "modelarts-job-dev-image/caffe1-cpu-cp27:1.0.0",
      "gpu_image_url": "modelarts-job-dev-image/caffe1-gpu-cuda8-cp27:1.0.0",
      "image_version": "3.1.0"
    }
  }, {
    "engine_id": "horovod-cp36-tf-1.16.2",
    "engine_name": "Horovod",
    "engine_version": "0.16.2-TF-1.13.1-python3.6",
    "v1_compatible": true,
    "run_user": "",
    "image_info": {
      "cpu_image_url": "modelarts-job-dev-image/tensorflow-gpu-cuda10-cp36-horovod0162:1.13.1",
      "gpu_image_url": "modelarts-job-dev-image/tensorflow-gpu-cuda10-cp36-horovod0162:1.13.1",
      "image_version": "3.2.1"
    }
  }, {
    "engine_id": "horovod_0.20.0-tensorflow_2.1.0-cuda_10.1-py_3.7-ubuntu_18.04-x86_64",
    "engine_name": "Horovod",
    "engine_version": "horovod_0.20.0-tensorflow_2.1.0-cuda_10.1-py_3.7-ubuntu_18.04-x86_64",
    "v1_compatible": false,
    "run_user": "1102",
    "image_info": {
      "cpu_image_url": "aip/horovod_tensorflow:train",
      "gpu_image_url": "aip/horovod_tensorflow:train",
      "image_version": "horovod_0.20.0-tensorflow_2.1.0-cuda_10.1-py_3.7-ubuntu_18.04-x86_64-20210912152543-1e0838d"
    }
  }, ".....", {
```

```
"engine_id" : "tensorflow_2.1.0-cuda_10.1-py_3.7-ubuntu_18.04-x86_64",
"engine_name" : "TensorFlow",
"engine_version" : "tensorflow_2.1.0-cuda_10.1-py_3.7-ubuntu_18.04-x86_64",
"v1_compatible" : false,
"run_user" : "1102",
"image_info" : {
  "cpu_image_url" : "aip/tensorflow_2_1:train",
  "gpu_image_url" : "aip/tensorflow_2_1:train",
  "image_version" : "tensorflow_2.1.0-cuda_10.1-py_3.7-ubuntu_18.04-x86_64-20210912152543-1e0838d"
}
}, {
"engine_id" : "xgboost-sklearn-python3.6",
"engine_name" : "XGBoost-Sklearn",
"engine_version" : "XGBoost-0.80-Sklearn-0.18.1-python3.6",
"v1_compatible" : true,
"run_user" : "",
"image_info" : {
  "cpu_image_url" : "modelarts-job-dev-image/python-train-py36:secure",
  "gpu_image_url" : "",
  "image_version" : "2.0.10-20211101113705"
}
}]
}
```

Status Codes

Status Code	Description
200	ok

Error Codes

See [Error Codes](#).

6 AI Application Management

6.1 Querying the AI Application List

Function

This API is used to query the AI application list based on different search parameters.

URI

GET /v1/{project_id}/models

Table 6-1 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .

Table 6-2 Query Parameters

Parameter	Mandatory	Type	Description
model_name	No	String	Model name. Fuzzy match is supported. If a model name contains an underscore (_), add the exact_match parameter to the request and set the parameter value to true because the underscore needs to be escaped. This ensures that the query operation can be performed properly.
exact_match	No	String	Whether to escape underscores (.). <i>If a model name contains underscores (),</i> set this parameter to true to ensure that the query operation can be performed properly. By default, this parameter is left blank.
model_version	No	String	Model version The format is Digit:Digit:Digit, where Digit is a one-digit or two-digit positive integer. Note that the version number cannot start with 0, for example, 01.01.01.
model_status	No	String	Model status. You can obtain models based on model statuses. Options: <ul style="list-style-type: none"> • publishing: The model is being published. • published: The model has been published. • failed: Publishing the model failed. • building: The image is being created. • building_failed: Creating an image failed.
description	No	String	Description. Fuzzy match is supported.
offset	No	Integer	Index of the query page, which defaults to 0

Parameter	Mandatory	Type	Description
limit	No	Integer	Maximum number of records returned on each page. Default value: 1000
sort_by	No	String	Sorting field. Options: <ul style="list-style-type: none"> • create_at: time when an AI application is created (default value) • model_version: AI application version • model_size: AI application size
order	No	String	Sorting mode. Options: <ul style="list-style-type: none"> • asc: ascending order • desc: descending order (default value)
workspace_id	No	String	Workspace ID, which defaults to 0
model_type	No	String	Model type. The models of this type are queried. <code>model_type</code> and <code>not_model_type</code> are mutually exclusive and cannot co-exist. The value can be TensorFlow, PyTorch, MindSpore, Image, Custom or Template.
not_model_type	No	String	Model type, which is used for obtaining models of types except for this type. The value can be TensorFlow, PyTorch, MindSpore, Image, Custom or Template.

Request Parameters

Table 6-3 Request header parameters

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

Response Parameters

Status code: 200

Table 6-4 Response body parameters

Parameter	Type	Description
models	Array of ModelListItem objects	Model metadata
total_count	Integer	Total number of models that meet the search criteria when no paging is performed
count	Integer	Total number of models that meet the search criteria

Table 6-5 ModelListItem

Parameter	Type	Description
owner	String	User ID of the tenant to which a model belongs
model_version	String	Model version
model_type	String	Model type
description	String	Model description
project	String	Project ID of the tenant to which a model belongs
source_type	String	Model source type. This parameter is valid and its value is auto only if the model is deployed using ExeML.

Parameter	Type	Description
model_id	String	Model ID
model_source	String	Model source. Options: <ul style="list-style-type: none"> • auto: ExeML • algos: built-in algorithm • custom: custom model
install_type	Array of strings	Deployment types supported by a model
model_size	Integer	Model size, in bytes
workspace_id	String	Workspace ID. Value 0 indicates the default workspace.
model_status	String	Model status
market_flag	Boolean	Whether a model is subscribed from AI Gallery
tunable	Boolean	Whether a model can be tuned. true indicates that the model can be tuned, and false indicates not.
model_name	String	Model name
create_at	Long	Time when a model is created, in milliseconds calculated from 1970.1.1 0:0:0 UTC.
publishable_flag	Boolean	Whether a model can be published to AI Gallery
source_copy	String	Whether to enable image replication. This parameter is valid only when model_type is set to Image . <ul style="list-style-type: none"> • true: Image replication is enabled. After this function is enabled, AI applications cannot be rapidly created, and modifying or deleting an image in the SWR source directory will not affect service deployment. • false: Image replication is not enabled. After this function is disabled, AI applications can be rapidly created, but modifying or deleting an image in the SWR source directory will affect service deployment. If this parameter is not configured, image replication is enabled by default.
tenant	String	Account ID of the tenant to which a model belongs
subscription_id	String	Model subscription ID

Parameter	Type	Description
extra	String	Extended parameter
specification	ModelSpecification object	Minimum specifications for model deployment

Table 6-6 ModelSpecification

Parameter	Type	Description
min_cpu	String	Minimal CPU specifications
min_gpu	String	Minimal GPU specifications
min_memory	String	Minimum memory
min_ascend	String	Minimal Ascend specifications

Example Requests

GET https://{endpoint}/v1/{project_id}/models

Example Responses

Status code: 200

Models

```
{
  "total_count": 1,
  "count": 1,
  "models": [ {
    "model_name": "mnist",
    "model_version": "1.0.0",
    "model_id": "10eb0091-887f-4839-9929-cbc884f1e20e",
    "model_type": "tensorflow",
    "model_size": 5012312,
    "tenant": "6d28e85aa78b4e1a9b4bd83501bcd4a1",
    "project": "d04c10db1f264cfef1966deff1a3527c",
    "owner": "6d28e85aa78b4e1a9b4bd83501bcd4a1",
    "create_at": 1533041553000,
    "description": "mnist model",
    "workspace_id": "0",
    "specification": { }
  } ]
}
```

Status Codes

Status Code	Description
200	Models

Error Codes

See [Error Codes](#).

6.2 Creating an AI Application

Function

Import a meta model to create an AI application. The execution code and model must be uploaded to OBS first. By default, the model generated by a training job is stored in OBS.

Constraints

The body parameter requirements for importing a model using a template are different from those for importing a model without using a template. In the following body parameters, **template parameters** indicate the parameters that can be configured when a model is imported using a template, **non-template parameters** indicate the parameters that can be configured when a model is imported without using a template, and **public parameters** indicate the parameters that are irrelevant to the model import mode.

- When a model is imported using a template (**model_type** is set to **Template**), the **template** field is mandatory and the **source_location** field does not need to be configured.
- If a model is imported without using a template (**model_type** is not set to **Template**), **source_location** is mandatory and **template** does not need to be configured.

URI

POST /v1/{project_id}/models

Table 6-7 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .

Request Parameters

Table 6-8 Request header parameters

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

Table 6-9 Request body parameters

Parameter	Mandatory	Type	Description
model_docs	No	Array of GuideDoc objects	List of model description documents. A maximum of three documents are supported. Common parameter If model_type is set to Custom , this parameter is invalid.
template	No	template object	Configuration items in a template. This parameter is mandatory when model_type is set to Template . Template parameter If model_type is set to Custom , this parameter is invalid.
model_version	Yes	String	Model version in the format of <i>Digit.Digit.Digit</i> . Each digit is a one-digit or two-digit positive integer, but cannot start with 0. For example, 01.01.01 is not allowed. Common parameter

Parameter	Mandatory	Type	Description
source_job_version	No	String	Version of the source training job. If the model is generated from a training job, input this parameter for source tracing. If the model is imported from a third-party meta model, leave this parameter blank. This parameter is left blank by default. Non-template parameter If model_type is set to Custom , this parameter is invalid.
source_location	Yes	String	OBS path where the model is located or the SWR image location. If the model_type value is Custom , set source_location to the OBS path to the model.
source_copy	No	String	Whether to enable image replication. This parameter is valid only when model_type is set to Image . Options: <ul style="list-style-type: none"> • true: Default value, indicating that image replication is enabled. After this function is enabled, AI applications cannot be rapidly created, and modifying or deleting an image in the SWR source directory will not affect service deployment. • false: Image replication is not enabled. After this function is disabled, AI applications can be rapidly created, but modifying or deleting an image in the SWR source directory will affect service deployment.

Parameter	Mandatory	Type	Description
initial_config	No	String	Character string converted from the model configuration file. Obtain fields such as apis , dependencies , input_params , output_params , and health in the initial_config configuration file. Non-template parameter If model_type is set to Custom , this parameter is invalid.
execution_code	No	String	OBS path for storing the execution code. By default, this parameter is left blank. The name of the execution code file is consistently to be customize_service.py . The inference code file must be stored in the model directory. This parameter can be left blank. Then, the system will automatically identify the inference code in the model directory. Common parameter If model_type is set to Custom , this parameter is invalid.
source_job_id	No	String	ID of the source training job. If the model is generated from a training job, input this parameter for source tracing. If the model is imported from a third-party meta model, leave this parameter blank. This parameter is left blank by default. Non-template parameter If model_type is set to Custom , this parameter is invalid.
model_type	Yes	String	Model type. The value is TensorFlow/Image/PyTorch/Template/MindSpore/Custom, which is read from the configuration file. Common Parameters

Parameter	Mandatory	Type	Description
output_parameters	No	Array of CreateModelRequestInferParams objects	Collection of output parameters of a model. By default, this parameter is left blank. If the parameters are read from apis in the configuration file, provide only the initial_config field, and this field can be left blank. Non-template parameter If model_type is set to Custom , this parameter is invalid.
description	No	String	Model description that consists of 1 to 100 characters. The following special characters cannot be contained: &!"'<>= Common parameter
runtime	No	String	Model runtime environment. Its possible values are determined based on model_type . For details about runtime options, see Managing AI Applications > Creating an AI Application > Importing a Meta Model from OBS in <i>ModelArts User Guide</i> .
model_metrics	No	String	Model precision. If the value is read from the configuration file, this parameter can be left blank. Non-template parameter

Parameter	Mandatory	Type	Description
source_type	No	String	Model source type. Currently, the value can only be auto, which is used to distinguish models deployed through ExeML (the model download function is not provided). This parameter is not required for models deployed through training jobs or other methods. It is left blank by default. Non-template parameter If model_type is set to Custom , this parameter is invalid.
dependencies	No	Array of ModelDependencies objects	Package required for inference code and model. By default, this parameter is left blank. If the package is read from the configuration file, this parameter can be left blank. Non-template parameter If model_type is set to Custom , this parameter is invalid.
workspace_id	No	String	Workspace ID, which defaults to 0 . Common parameter
model_algorithm	No	String	Model algorithm. If the algorithm is read from the configuration file, this parameter can be left blank. The value can be predict_analysis , object_detection , or image_classification . Non-template parameter
apis	No	Array of CreateModelRequestModelApis objects	All API input and output parameters of the model. If the parameters are parsed from the configuration file, this parameter can be left blank. Non-template parameter If model_type is set to Custom in an asynchronous request, this parameter is invalid.

Parameter	Mandatory	Type	Description
model_name	Yes	String	Model name, which consists of 1 to 64 characters. Common parameter
install_type	No	Array of strings	Deployment type. Only lowercase letters are supported. The value can be real-time or batch . Default value: [real-time , batch]
input_params	No	Array of CreateModelRequestInferParams objects	Collection of input parameters of a model. By default, this parameter is left blank. If the parameters are read from apis in the configuration file, provide only the initial_config field, and this field can be left blank. Non-template parameter
cmd	No	String	Image startup command.
dynamic_load_mode	No	String	Dynamic loading mode. The default value is None , indicating that this mode is not used. The value Single indicating that this mode is used.
deployment_constraints	No	deployment_constraints objects	Model deployment constraints.

Table 6-10 GuideDoc

Parameter	Mandatory	Type	Description
doc_url	Yes	String	HTTP(S) link of the document
doc_name	Yes	String	Document name, which must start with a letter.

Table 6-11 template

Parameter	Mandatory	Type	Description
infer_format	No	String	ID of the input and output mode. When this parameter is used, the input and output mode built in the template does not take effect.
template_inputs	Yes	Array of CreateModelRequestTemplateInput objects	Template input configuration, specifying the source path for configuring a model.
template_id	Yes	String	ID of the used template. The template has a built-in input and output mode.

Table 6-12 CreateModelRequestTemplateInput

Parameter	Mandatory	Type	Description
input	Yes	String	Template input path, which can be a path to an OBS file or directory. When you use a template with multiple input items to create a model, if the target paths input_properties specified in the template are the same, the OBS directory or OBS file name entered here must be unique to prevent files from being overwritten.
input_id	Yes	String	Input item ID, which is obtained from template details

Table 6-13 ModelDependencies

Parameter	Mandatory	Type	Description
installer	Yes	String	Installation mode. Only pip is supported.
packages	Yes	Array of Packages objects	Collection of dependency packages

Table 6-14 Packages

Parameter	Mandatory	Type	Description
package_version	No	String	Version of a dependency package. If this parameter is left blank, the latest version is installed by default.
package_name	Yes	String	Name of a dependency package. Ensure that the package name is correct and available.
restraint	No	String	Version restriction, which can be EXACT , ATLEAST , or ATMOST . This parameter is mandatory only when package_version is available.

Table 6-15 CreateModelRequestModelApis

Parameter	Mandatory	Type	Description
protocol	No	String	Request protocol. The options are HTTP and HTTPS.
method	No	String	Request method, which can be post or get
input_params	No	ModelInOutParams object	API input and output parameters, described in JSON Schema format
output_params	No	ModelInOutParams object	API input and output parameters, described in JSON Schema format
url	No	String	Inference request URL

Table 6-16 ModelInOutParams

Parameter	Mandatory	Type	Description
type	No	String	Type in JSON Schema, which can be object
properties	No	Object	Properties of an object element in JSON Schema. You can configure parameters, including the parameter name and type.

Table 6-17 CreateModelRequestInferParams

Parameter	Mandatory	Type	Description
protocol	Yes	String	Request protocol. The options are HTTP and HTTPS.
min	No	Number	Minimum value of the parameter. This parameter is optional when param_type is set to int or float. By default, this parameter is left blank.
method	Yes	String	Request method, which can be post or get .
max	No	Number	Maximum value of the parameter. This parameter is optional when param_type is set to int or float. By default, this parameter is left blank.
param_desc	No	String	Parameter description. It is recommended that the parameter description contain a maximum of 100 characters. By default, this parameter is left blank.
param_name	Yes	String	Parameter name. It is recommended that the parameter name contain a maximum of 64 characters.
url	Yes	String	API URL
param_type	Yes	String	Parameter type, which can be int , string , float , timestamp , date , or file

Table 6-18 health parameters

Parameter	Mandatory	Type	Description
check_method	Yes	String	Health check method.
protocol	No	String	Protocol of a health check API. The default value is https .
url	Yes	String	Health check URL.
period_seconds	Yes	String	Health check period.

Parameter	Mandatory	Type	Description
initial_delay_seconds	No	String	Delay for initializing the health check.
failure_threshold	Yes	String	Maximum number of health check failures.
timeout_seconds	No	String	Health check timeout.
command	No	String	Commands, which are strings separated by spaces.

Table 6-19 deployment_constraints

Parameter	Mandatory	Type	Description
request_mode	Yes	String	Request mode. The value can be async or sync.
cpu_type	Yes	String	CPU type. The value can be aarch64 or x86_64.
accelerators	Yes	Array of accelerators object	Accelerator card type. Example: [{"type": "GPU", "name": "a800"}]

Response Parameters

Status code: 200

Table 6-20 Response body parameters

Parameter	Type	Description
model_id	String	Model ID

Example Requests

The following is an example of how to create an AI application whose name is **mnist**, version is **1.0.0**, type is **TensorFlow**, and model file source is an OBS bucket.

```
POST https://{endpoint}/v1/{project_id}/models
{
  "model_name": "mnist",
  "model_version": "1.0.0",
  "source_location": "https://models.obs.xxxx.com/mnist",
```

```

"source_job_id" : "55",
"source_job_version" : "V100",
"model_type" : "TensorFlow",
"runtime" : "python2.7",
"description" : "mnist model",
"execution_code" : "https://testmodel.obs.xxx.com/customize_service.py",
"input_params" : [ {
  "url" : "/v1/xxx/image",
  "protocol" : "http",
  "method" : "post",
  "param_name" : "image_url",
  "param_type" : "string",
  "min" : 0,
  "max" : 9,
  "param_desc" : "http://test/test.jpeg"
} ],
"output_params" : [ {
  "url" : "/v1/xxx/image",
  "protocol" : "http",
  "method" : "post",
  "param_name" : "face_location",
  "param_type" : "box",
  "param_desc" : "face_location param value description"
} ],
"dependencies" : [ {
  "installer" : "pip",
  "packages" : [ {
    "package_name" : "numpy",
    "package_version" : "1.5.0",
    "restraint" : "ATLEAST"
  } ]
} ],
"model_algorithm" : "object_detection",
"model_metrics" : "{\"f1\":0.52381,\"recall\":0.666667,\"precision\":0.466667,\"accuracy\":0.625}",
"apis" : [ {
  "url" : "/v1/xxx/image",
  "protocol" : "http",
  "method" : "post",
  "input_params" : {
    "type" : "object",
    "properties" : {
      "image_url" : {
        "type" : "string"
      }
    }
  },
  "output_params" : {
    "type" : "object",
    "properties" : {
      "face_location" : {
        "type" : "box"
      }
    }
  }
} ]
} ]
}

```

Example Responses

Status code: 200

The model is created.

```

{
  "model_id" : "7feb7235-ed9c-48ae-9833-2876b2458445"
}

```

Status Codes

Status Code	Description
200	The model is created.

Error Codes

See [Error Codes](#).

6.3 Obtaining Details About an AI Application

Function

This API is used to query details about an AI application based on the AI application ID.

URI

GET /v1/{project_id}/models/{model_id}

Table 6-21 Path Parameters

Parameter	Mandatory	Type	Description
model_id	Yes	String	Model ID
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .

Request Parameters

Table 6-22 Request header parameters

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

Response Parameters

Status code: 200

Table 6-23 Response body parameters

Parameter	Type	Description
model_version	String	Model version
source_job_version	String	Version of the source training job
source_location	String	OBS path where the model is located or the template address of the SWR image
source_job_id	String	ID of the source training job
source_copy	String	Whether to enable image replication. This parameter is valid only when model_type is set to Image . <ul style="list-style-type: none"> • true: Default value, indicating that image replication is enabled. After this function is enabled, AI applications cannot be rapidly created, and modifying or deleting an image in the SWR source directory will not affect service deployment. • false: Image replication is not enabled. After this function is disabled, AI applications can be rapidly created, but modifying or deleting an image in the SWR source directory will affect service deployment.
description	String	Model description
project	String	Project ID of the tenant to which a model belongs
workspace_id	String	Workspace ID. Value 0 indicates the default workspace.
model_algorithm	String	Model algorithm type, which can be predict_analysis , object_detection , or image_classification
model_name	String	Model name
tenant	String	Account ID of the tenant to which a model belongs
model_docs	Array of GuideDoc objects	List of template documents

Parameter	Type	Description
owner	String	ID of the tenant to which a model belongs
execution_code	String	OBS path for storing the execution code. The name of the execution code file is consistently to be customize_service.py .
schema_doc	String	Download address of the model schema file
image_addresses	String	Image path generated after model packaging
output_parameters	Array of ModelParamsInfo objects	Collection of output parameters of a model
health	ModelHealth object	Model health check information
runtime	String	Model runtime environment
model_metrics	String	Model precision
source_type	String	Model source type. This parameter is valid and its value is auto only if the model is deployed using ExeML.
model_type	String	Model type. The value is TensorFlow/Image/PyTorch/Template/MindSpore.
model_id	String	Model ID
dependencies	Array of ModelDependencies objects	Package required for running the code and model
model_size	Long	Model size, in bytes
model_status	String	Model status
apis	String	All API input and output parameter information of a model, which is obtained from the model preview
model_source	String	Model source. Options: <ul style="list-style-type: none"> • auto: ExeML • algos: preset algorithm • custom: customized
tunable	Boolean	Whether a model can be tuned. Options: <ul style="list-style-type: none"> • true: Supported. • false: Not supported.

Parameter	Type	Description
market_flag	Boolean	Whether a model is subscribed from the marketplace. Options: <ul style="list-style-type: none"> • true: from the market • false: no
publishable_flag	Boolean	Whether a model can be published to the marketplace. Options: <ul style="list-style-type: none"> • true: The product can be released to the market. • false: The product cannot be released to the marketplace.
model_labels	Array of strings	Model label array
labels_map	Map<String,String>	Model label map. The key is consistently to be labels , and the value is the model label array.
install_type	Array of strings	Supported service type for deployment
config	String	Model configurations
specification	ModelSpecification object	Minimum specifications for model deployment
input_params	Array of ModelParamInfo objects	Collection of input parameters of a model
create_at	Long	Time when a model is created, in milliseconds calculated from 1970.1.1 0:0:0 UTC.

Table 6-24 GuideDoc

Parameter	Type	Description
doc_url	String	HTTP(S) link of the document
doc_name	String	Document name, which must start with a letter.

Table 6-25 ModelHealth

Parameter	Type	Description
protocol	String	Request protocol for health check, which can only be HTTP

Parameter	Type	Description
initial_delay_seconds	String	After an instance is started, a health check starts after the time configured in initial_delay_seconds expires.
timeout_seconds	String	Health check timeout
url	String	Path to the health check API

Table 6-26 ModelDependencies

Parameter	Type	Description
installer	String	Installation mode. Only pip is supported.
packages	Array of Packages objects	Collection of dependency packages

Table 6-27 Packages

Parameter	Type	Description
package_version	String	Version of a dependency package. If this parameter is left blank, the latest version is installed by default.
package_name	String	Name of a dependency package. Ensure that the package name is correct and available.
restraint	String	Version restriction, which can be EXACT , ATLEAST , or ATMOST . This parameter is mandatory only when package_version is available.

Table 6-28 ModelSpecification

Parameter	Type	Description
min_cpu	String	Minimal CPU specifications
min_gpu	String	Minimal GPU specifications
min_memory	String	Minimum memory
min_ascend	String	Minimal Ascend specifications

Table 6-29 ModelParamsInfo

Parameter	Type	Description
protocol	String	Request protocol, for example, http
url	String	API URL
min	Number	Minimum value of a numeric parameter
method	String	Request method, for example, post
max	Number	Maximum value of a numeric parameter
param_desc	String	Parameters
param_name	String	Parameter
param_type	String	Type

Example Requests

GET https://{endpoint}/v1/{project_id}/models/{model_id}

Example Responses

Status code: 200

Model details

```
{
  "model_id": "10eb0091-887f-4839-9929-cbc884f1e20e",
  "model_name": "mnist",
  "model_version": "1.0.0",
  "runtime": "python2.7",
  "tenant": "6d28e85aa78b4e1a9b4bd83501bcd4a1",
  "project": "d04c10db1f264cfef1966deff1a3527c",
  "owner": "6d28e85aa78b4e1a9b4bd83501bcd4a1",
  "source_location": "https://models.obs.xxxx.com/mnist",
  "model_type": "TensorFlow",
  "model_size": 5633481,
  "model_status": "published",
  "execution_code": "https://testmodel.obs.xxxx.com/customize_service.py",
  "image_address": "100.125.5.235:20202/models/10eb0091-887f-4839-9929-cbc884f1e20e:1.0.0",
  "input_params": [ {
    "url": "/",
    "method": "post",
    "protocol": "http",
    "param_name": "data",
    "param_type": "object",
    "param_desc": "{\n  \"type\": \"object\", \"properties\": {\n    \"req_data\": {\n      \"items\": [\n        {\n          \"type\": \"object\", \"properties\": {\n            \"type\": \"array\"
        }
      ]
    }
  }
}]"
  },
  "output_params": [ {
    "url": "/",
    "method": "post",
    "protocol": "http",
    "param_name": "data",
    "param_type": "object",
    "param_desc": "{\n  \"type\": \"object\", \"properties\": {\n    \"resp_data\": {\n      \"type\": \"array\", \"items\": [\n        {\n          \"type\": \"object\", \"properties\": {\n            \"type\": \"array\"
        }
      ]
    }
  }
}]"
  },
  "dependencies": [ {
```

```

"installer" : "pip",
"packages" : [ {
  "package_name" : "pkg1",
  "package_version" : "1.0.1",
  "restraint" : "ATLEAST"
} ]
},
"model_metrics" : "{\f1\":0.52381,\recall\":0.666667,\precision\":0.466667,\accuracy\":0.625}",
"apis" : "[{\protocol\":\http\",method\":\post\",url\":\\",input_params\":{\type\":\object\",properties\":{\data\":{\type\":\object\",properties\":{\req_data\":{\items\":{\type\":\object\",properties\":{\}}},\type\":\array\"}}},\output_params\":{\type\":\object\",properties\":{\data\":{\type\":\object\",properties\":{\resp_data\":{\type\":\array\",items\":{\type\":\object\",properties\":{\}}}}}}}],
"model_labels" : [ ],
"labels_map" : {
  "labels" : [ ]
},
"workspace_id" : "0",
"install_type" : [ "realtime", "batch", "edge" ],
"specification" : { },
"config" : "{\model_algorithm\":\image_classification\",model_source\":\auto\",tunable\":false,\downloadable_flag\":true,\algorithm\":\resnet_v2_50,mobilenet_v1\",metrics\":{\f1\":0.263250724969475,\model_type\":\TensorFlow\",runtime\":\tf1.13-python3.6-cpu\",apis\":[{\protocol\":\https\",url\":\\",method\":\post\",request\":{\data\":{\type\":\object\",properties\":{\images\":{\type\":\file\"}},Content-type\":\multipart/form-data\",response\":{\data\":{\type\":\object\",required\":[\predicted_label\",scores\"],properties\":{\predicted_label\":{\type\":\string\"},scores\":{\type\":\array\",items\":{\type\":\array\",minItems\":2,maxItems\":2,items\":{\type\":\string\"},\type\":\number\"}}}],Content-type\":\multipart/form-data\"}},dependencies\":[{\installer\":\pip\",packages\":[{\package_name\":\numpy\",package_version\":\1.17.0\",restraint\":\EXACT\"},{\package_name\":\h5py\",package_version\":\2.8.0\",restraint\":\EXACT\"},{\package_name\":\Pillow\",package_version\":\5.2.0\",restraint\":\EXACT\"},{\package_name\":\scipy\",package_version\":\1.2.1\",restraint\":\EXACT\"},{\package_name\":\resampy\",package_version\":\0.2.1\",restraint\":\EXACT\"},{\package_name\":\scikit-learn\",package_version\":\0.19.1\",restraint\":\EXACT\"}]}]
}

```

Status Codes

Status Code	Description
200	Model details

Error Codes

See [Error Codes](#).

6.4 Deleting an AI application

Function

This interface is used to delete an AI application based on the AI application ID. When cascade is set to true, the AI application specified by the AI application ID and other AI applications with the same name but different versions as the specified AI application are deleted. By default, only the AI application corresponding to the current AI application ID is deleted.

URI

DELETE /v1/{project_id}/models/{model_id}

Table 6-30 Path Parameters

Parameter	Mandatory	Type	Description
model_id	Yes	String	ID of the AI application to be deleted.
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .

Table 6-31 Query Parameters

Parameter	Mandatory	Type	Description
cascade	No	Boolean	Indicates whether to perform cascading deletion. The default value is false, indicating that only the model with the specified model ID is deleted. If this parameter is set to true, the model specified by the model ID is deleted, and all models with the same name but different versions as the specified model are deleted. A maximum of 20 models can be deleted at a time. Excess models are not deleted.

Request Parameters

Table 6-32 Request header parameters

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

Response Parameters

Status code: 200

Table 6-33 Response body parameters

Parameter	Type	Description
delete_success_list	Array of strings	ID list of models successfully deleted
delete_failed_list	Array of DeleteModelResponseFailedList objects	ID of the model that fails to be deleted and the failure cause list

Table 6-34 DeleteModelResponseFailedList

Parameter	Type	Description
error_msg	String	Error message of the model deletion failure
error_code	String	Error code of the model deletion failure
model_id	String	ID of a model that fails to be deleted

Example Requests

```
DELETE https://{endpoint}/v1/{project_id}/models/{model_id}
```

Example Responses

Status code: 200

Message indicating a successful deletion or a deletion failure

```
{
  "delete_success_list": [ "10eb0091-887f-4839-9929-cbc884f1e20e" ],
  "delete_failed_list": [ {
    "error_msg": "Failed to delete model because the model (759645d9-3672-4db1-bb6d-49ed58b84e10)
has been used to deploy a service.",
    "error_code": "ModelArts.3009",
    "model_id": "e527d311-a947-46da-a6e0-66c49945dfaa"
  } ]
}
```

Status Codes

Status Code	Description
200	Message indicating a successful deletion or a deletion failure

Error Codes

See [Error Codes](#).

7 Service Management

7.1 Obtaining Service Monitoring

Function

This API is used to obtain service monitoring information.

URI

GET /v1/{project_id}/services/{service_id}/monitor

Table 7-1 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .
service_id	Yes	String	Service ID

Table 7-2 Query Parameters

Parameter	Mandatory	Type	Description
node_id	No	String	ID of the edge node to be obtained. This parameter is available only when infer_type is set to edge . By default, all nodes are obtained.

Request Parameters

Table 7-3 Request header parameters

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

Response Parameters

Status code: 200

Table 7-4 Response body parameters

Parameter	Type	Description
service_name	String	Service name
service_id	String	Service ID
monitors	Array of Monitor objects	Monitoring details
service_running_instance_count	Integer	Number of running service instances
service_instance_count	Integer	Number of service instances
req_count_per_min	Long	Total number of service calls in the last minute

Table 7-5 Monitor

Parameter	Type	Description
failed_times	Integer	Number of times that a model instance fails to be called. This parameter is available for real-time services.
model_version	String	Model version, which is available for real-time services

Parameter	Type	Description
cpu_memory_total	Integer	Total memory, in MB
gpu_usage	Float	Number of used GPUs
node_name	String	Node name, which is available for edge services
gpu_total	Float	Total number of GPUs
model_id	String	Model ID, which is available for real-time services
invocation_times	Integer	Number of times that a model instance is called. This parameter is available for real-time services.
cpu_core_usage	Float	Number of used CPU cores
cpu_core_total	Float	Total number of CPU cores
model_name	String	Model name, which is available for real-time services
cpu_memory_usage	Integer	Used memory, in MB
node_id	String	Edge node ID, which is available for edge services
model_running_instance_count	Integer	Number of running model instances
model_instance_count	Integer	Number of model instances

Example Requests

```
GET https://{endpoint}/v1/{project_id}/services/{service_id}/monitor
```

Example Responses

Status code: 200

Monitoring information

```
{
  "service_name": "mnist",
  "service_id": "195c1f2d-136d-40af-a0f3-db5717d2634a",
  "monitors": [ {
    "failed_times": 1,
    "model_version": "1.0.0",
    "cpu_core_total": 4,
```

```

"cpu_memory_total" : 8192,
"model_name" : "minst",
"gpu_usage" : 0.6,
"cpu_memory_usage" : 2011,
"gpu_total" : 1,
"model_id" : "0e07b41b-173e-42db-8c16-8e1b44cc0d44",
"invocation_times" : 50,
"cpu_core_usage" : 2.4
} ]
}

```

Status Codes

Status Code	Description
200	Monitoring information

Error Codes

See [Error Codes](#).

7.2 Obtaining Services

Function

This API is used to obtain model services.

URI

GET /v1/{project_id}/services

Table 7-6 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .

Table 7-7 Query Parameters

Parameter	Mandatory	Type	Description
service_id	No	String	Service ID. By default, the service ID is not filtered.
service_name	No	String	Service name. By default, the service name is not filtered.

Parameter	Mandatory	Type	Description
model_id	No	String	Model ID. By default, the model ID is not filtered.
workspace_id	No	String	Workspace ID. If this parameter is not configured, only the default workspace is obtained.
infer_type	No	String	Inference mode. By default, the inference mode is not filtered. Options: <ul style="list-style-type: none"> • real-time • batch

Parameter	Mandatory	Type	Description
status	No	String	<p>Service status. By default, the service status is not filtered. Options:</p> <ul style="list-style-type: none"> ● running: The service is running properly. ● deploying: The service is being deployed, including image creation and resource scheduling deployment. ● concerning: An alarm has been generated, indicating that some backend instances malfunction. ● failed: Deploying the service failed. For details about the failure cause, see the event and log tab pages. ● stopped: The service has been stopped. ● finished: Service running is completed. This status is available only for batch services. ● stopping: The VM is being stopped. ● deleting: The VM is being deleted. ● pending: to be started. This state is available only when the system is online. ● waiting: The resource is queuing. This state is available only for online services.
offset	No	Integer	Start page for pagination display. The default value is 0 .
limit	No	Integer	Maximum number of records returned on each page. Default value: 1000

Parameter	Mandatory	Type	Description
sort_by	No	String	Specifies the sorting field. The options are as follows: <ul style="list-style-type: none"> publish_at: service publishing time, which is the default value. service_name: service name transition_at: update time.
order	No	String	Sorting mode. Options: <ul style="list-style-type: none"> asc: ascending order desc: descending order (default value)
tags	No	String	Tag. By default, tags are not filtered.

Request Parameters

Table 7-8 Request header parameters

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

Response Parameters

Status code: 200

Table 7-9 Response body parameters

Parameter	Type	Description
total_count	Integer	Total number of services that meet the search criteria when no paging is performed
count	Integer	Number of services in the query result. If offset and limit are not configured, the values of count and total are the same.

Parameter	Type	Description
services	Array of ListServices objects	Collection of obtained services

Table 7-10 ListServices

Parameter	Type	Description
failed_times	Long	Number of failed service calls
owner	String	User to which a service belongs
due_time	Number	Time when an online service automatically stops, in milliseconds calculated from 1970.1.1 0:0:0 UTC. This parameter is not returned if automatic stop is not configured.
finished_time	Number	Batch service completion time, in milliseconds calculated from 1970.1.1 0:0:0 UTC. This parameter is returned only when the service is a batch service.
infer_type	String	Inference mode. Options: <ul style="list-style-type: none"> • real-time • batch
service_name	String	Service name
description	String	Service description
project	String	Project to which a service belongs
invocation_times	Long	Total number of service calls
publish_at	Long	Latest service release time, in milliseconds calculated from 1970.1.1 0:0:0 UTC.
workspace_id	String	Workspace ID
schedule	Array of Schedule objects	Service scheduling configuration. If this parameter is not configured, no value will be returned.
start_time	Number	Batch service start time, in milliseconds calculated from 1970.1.1 0:0:0 UTC. This parameter is returned only when the service is a batch service.
operation_time	Number	Operation time of a request

Parameter	Type	Description
is_shared	Boolean	Whether a service is subscribed True indicates that the service is a subscription service.
service_id	String	Service ID
progress	Integer	Deployment progress. This parameter is available when the status is deploying .
shared_count	Number	Number of subscribed services
tenant	String	Tenant to which a service belongs
status	String	Service status. Options: <ul style="list-style-type: none"> • running: The service is running properly. • deploying: The service is being deployed, including image creation and resource scheduling deployment. • concerning: An alarm has been generated, indicating that some backend instances malfunction. • failed: Deploying the service failed. For details about the failure cause, see the event and log tab pages. • stopped: The service has been stopped. • finished: Service running is completed. This status is available only for batch services.
is_opened_sample_collection	String	Whether to enable data collection, which defaults to false
transition_at	Number	Time when the service status changes
is_free	Boolean	Whether a free-of-charge flavor is used
additional_properties	Map<String,String>	Additional service attribute. If this parameter is not configured, no value will be returned.

Table 7-11 Schedule

Parameter	Type	Description
duration	Integer	Value mapping a time unit. For example, if the task stops after two hours, set time_unit to HOURS and duration to 2 .
time_unit	String	Scheduling time unit. Possible values are DAYS , HOURS , and MINUTES .

Parameter	Type	Description
type	String	Scheduling type. Only the value stop is supported.

Example Requests

GET https://{endpoint}/v1/{project_id}/services

Example Responses

Status code: 200

Service list

```
{
  "total_count": 1,
  "count": 1,
  "services": [ {
    "failed_times": 1,
    "owner": "b575785bcece44beb23597770fb819f9",
    "infer_type": "real-time",
    "service_name": "mnist",
    "description": "",
    "project": "b575785bcece44beb23597770fb819f9",
    "invocation_times": 50,
    "publish_at": 1243143243,
    "workspace_id": 0,
    "is_shared": false,
    "service_id": "195c1f2d-136d-40af-a0f3-db5717d2634a",
    "shared_count": 0,
    "tenant": "b575785bcece44beb23597770fb819f9",
    "status": "running"
  } ]
}
```

Status Codes

Status Code	Description
200	Service list

Error Codes

See [Error Codes](#).

7.3 Deploying Services

Function

This API is used to deploy a model as a service.

URI

POST /v1/{project_id}/services

Table 7-12 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .

Request Parameters

Table 7-13 Request header parameters

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

Table 7-14 Request body parameters

Parameter	Mandatory	Type	Description
workspace_id	No	String	ID of the workspace to which a service belongs. The default value is 0 , indicating the default workspace.
schedule	No	Array of Schedule objects	Service scheduling configuration, which can be configured only for real-time services. By default, this parameter is not used. Services run for a long time.

Parameter	Mandatory	Type	Description
cluster_id	No	String	Dedicated resource pool ID. By default, this parameter is left blank, indicating that dedicated resource pools are not used. When using a dedicated resource pool to deploy services, ensure that the cluster is running properly. After this parameter is configured, the network configuration of the cluster is used, and the vpc_id parameter does not take effect. If both this parameter and cluster_id in real-time config are configured, cluster_id in real-time config is preferentially used. If a dedicated resource pool is used, either cluster_id or pool_name must be specified.
pool_name	No	String	ID of a new-version dedicated resource pool. This parameter is left blank by default, indicating that no dedicated resource pool is used. When using a dedicated resource pool to deploy services, ensure that the cluster is running properly. If both this parameter and pool_name in real-time config are configured, pool_name in real-time config is preferentially used. If a dedicated resource pool is used, either cluster_id or pool_name must be specified.

Parameter	Mandatory	Type	Description
infer_type	Yes	String	<p>Inference mode. The value is real-time/batch.</p> <ul style="list-style-type: none"> • real-time indicates a real-time service. A model is deployed as a web service. Service, which provides online test UI and monitoring capabilities. The service keeps running. • Batch is a batch service. A batch service can perform inference on batch data and automatically stops after data processing is complete. • must be created on Intelligent EdgeFabric (IEF) in advance.
vpc_id	No	String	<p>ID of the VPC to which a real-time service instance is deployed. By default, this parameter is left blank. In this case, ModelArts allocates a dedicated VPC to each user, and users are isolated from each other. To access other service components in the VPC of the service instance, set this parameter to the ID of the corresponding VPC. Once a VPC is configured, it cannot be modified. If both vpc_id and cluster_id are configured, only the dedicated resource pool takes effect.</p>
service_name	Yes	String	<p>Service name, which consists of 1 to 64 characters.</p>
description	No	String	<p>Service remarks. By default, this parameter is left blank. The value contains a maximum of 100 characters and cannot contain! . < > + &""</p>

Parameter	Mandatory	Type	Description
security_group_id	No	String	Security group. By default, this parameter is left blank. This parameter is mandatory if vpc_id is configured. A security group is a virtual firewall that provides secure network access control policies for service instances. A security group must contain at least one inbound rule to permit the requests whose protocol is TCP, source address is 0.0.0.0/0, and port number is 8080.
subnet_network_id	No	String	ID of a subnet. By default, this parameter is left blank. This parameter is mandatory if vpc_id is configured. Enter the network ID displayed in the subnet details on the VPC management console. A subnet provides dedicated network resources that are isolated from other networks.
config	Yes	Array of ServiceConfig objects	Model running configurations. If infer_type is batch , you can configure only one model. If infer_type is real-time , you can configure multiple models and assign weights based on service requirements. However, the versions of multiple models must be unique.
additional_properties	No	Map<String, ServiceAdditionalProperties >	Additional service attribute, which facilitates service management

Table 7-15 Schedule

Parameter	Mandatory	Type	Description
duration	Yes	Integer	Value mapping a time unit. For example, if the task stops after two hours, set time_unit to HOURS and duration to 2 .

Parameter	Mandatory	Type	Description
time_unit	Yes	String	Scheduling time unit. Possible values are DAYS , HOURS , and MINUTES .
type	Yes	String	Scheduling type. Only the value stop is supported.

Table 7-16 ServiceConfig

Parameter	Mandatory	Type	Description
custom_spec	No	CustomSpec object	Custom resource specifications object
envs	No	Map<String,String>	Common parameter. (Optional) Environment variable key-value pair required for running a model. By default, this parameter is left blank.
specification	Yes	String	Common parameters Resource specifications. You can obtain the specification list by querying the supported service deployment specifications. In the current version, modelarts.vm.cpu.2u/ modelarts.vm.gpu.p4 (needs to be applied for), modelsarts.vm.ai1.a310 (needs to be applied for), and custom (supported only when deployed in a dedicated resource pool) are available. [Submit a service ticket.] (tag: hc, hk) ModelArts O&M engineers add permissions. If this parameter is set to custom, the custom_spec parameter must be specified.

Parameter	Mandatory	Type	Description
weight	No	Integer	This parameter is mandatory for real-time . Weight of traffic allocated to a model. This parameter is mandatory only when infer_type is set to real-time . The sum of all weights must be equal to 100. If multiple model versions are configured with different traffic weights in a real-time service, ModelArts will continuously access the prediction API of the service and forward prediction requests to the model instances of the corresponding versions based on the weights.
deploy_timeout_in_seconds	No	Integer	Timeout interval for deploying a single model instance
model_id	Yes	String	Common parameters Model ID. You can obtain the value by calling the API for querying the AI application list.
src_path	No	String	Mandatory for batch services. OBS path to the input data of a batch job
req_uri	No	String	Mandatory for batch services. Inference API called in a batch task, which is the RESTful API exposed in the model image. You must select an API URL from the config.json file of the model for inference. If a built-in inference image of ModelArts is used, the API is displayed as <code>/</code> .

Parameter	Mandatory	Type	Description
mapping_type	No	String	<p>The batch service type is mandatory. Mapping type of the input data. The value can be file or csv.</p> <ul style="list-style-type: none"> • If file is selected, each inference request corresponds to a file in the input data directory. When this mode is used, req_uri corresponding to the model can have only one input parameter and the parameter type is file. • If csv is selected, each inference request corresponds to a row of data in the CSV file. If this mode is used, the file name extension in the input data directory must be .csv, and the mapping_rule parameter must be configured to indicate the CSV index corresponding to each parameter in the inference request body.
cluster_id	No	String	<p>Optional for real-time services. ID of a dedicated resource pool. This parameter is left blank by default, indicating that no dedicated resource pool is used. When using a dedicated resource pool to deploy services, ensure that the resource pool is running properly. After this parameter is configured, the network configuration of the cluster is used, and the vpc_id parameter does not take effect.</p>

Parameter	Mandatory	Type	Description
pool_name	No	String	Specifies the ID of the new dedicated resource pool. By default, this parameter is left blank, indicating that the dedicated resource pool is not used. This parameter corresponds to the ID of the new resource pool. When using dedicated resource pool to deploy services, ensure that the cluster status is normal. If pool_name in real-time config and pool_name in real-time config are configured at the same time, pool_name in real-time config is preferred.
nodes	No	Array of strings	Mandatory for edge services. Edge node ID array. The node ID is the edge node ID on IEF, which can be obtained after the edge node is created on IEF.
mapping_rule	No	Object	Optional for batch services. Mapping between input parameters and CSV data. This parameter is mandatory only when mapping_type is set to csv . The mapping rule is similar to the definition of the input parameters in the config.json file. You only need to configure the index parameters under each parameter of the string, number, integer, or boolean type, and specify the value of this parameter to the values of the index parameters in the CSV file to send an inference request. Use commas (,) to separate multiple pieces of CSV data. The values of the index parameters start from 0. If the value of the index parameter is -1, ignore this parameter. For details, see the sample of creating a batch service.

Parameter	Mandatory	Type	Description
src_type	No	String	Mandatory for batch services. Data source type, which can be ManifestFile . By default, this parameter is left blank, indicating that only files in the src_path directory are read. If this parameter is set to ManifestFile , src_path must be set to a specific manifest path. Multiple data paths can be specified in the manifest file. For details, see the manifest inference specifications.
dest_path	No	String	Mandatory for batch services. OBS path to the output data of a batch job
instance_count	Yes	Integer	Common parameter. Number of instances deployed for a model. The maximum number of instances is 5. To use more instances, submit a service ticket.
additional_properties	No	Map<String,ModelAdditionalProperties>	Additional attributes for model deployment, facilitating service instance management

Table 7-17 CustomSpec

Parameter	Mandatory	Type	Description
gpu_p4	No	Float	Number of GPUs, which can be a decimal. The value cannot be smaller than 0, with the third decimal place is rounded off. This parameter is optional and is not used by default.
memory	Yes	Integer	Memory in MB, which must be an integer
cpu	Yes	Float	Number of CPU cores, which can be a decimal. The value cannot be smaller than 0.01, with the third decimal place is rounded off.

Parameter	Mandatory	Type	Description
ascend_a310	No	Integer	Number of Ascend chips. This parameter is optional and is not used by default. Either this parameter or gpu_p4 is configured.

Table 7-18 ModelAdditionalProperties

Parameter	Mandatory	Type	Description
log_volume	No	Array of log_volume objects	Host directory mounting. This parameter takes effect only if a dedicated resource pool is used. If a public resource pool is used to deploy services, this parameter cannot be configured. Otherwise, an error will occur.
max_surge	No	Float	The value must be greater than 0. If this parameter is not set, the default value 1 is used. If the value is less than 1, it indicates the percentage of instances to be added during the rolling upgrade. If the value is greater than 1, it indicates the maximum number of instances to be added during the rolling upgrade.
max_unavailable	No	Float	The value must be greater than 0. If this parameter is not set, the default value 0 is used. If the value is less than 1, it indicates the percentage of instances that can be scaled in during the rolling upgrade. If the value is greater than 1, it indicates the number of instances that can be scaled in during the rolling upgrade.
termination_grace_period_seconds	No	Integer	Graceful stop time of a container

Parameter	Mandatory	Type	Description
persistent_volumes	No	Array of persistent_volumes objects	Persistent Storage Mounting Configuration

Table 7-19 log_volume

Parameter	Mandatory	Type	Description
host_path	Yes	String	Log path to be mapped on the host
mount_path	Yes	String	Path to the logs in the container

Table 7-20 persistent_volumes

Parameter	Mandatory	Type	Description
name	No	String	Specifies the name of the storage volume.
mount_path	Yes	String	Mount path of the storage volume in the container.

Table 7-21 ServiceAdditionalProperties

Parameter	Mandatory	Type	Description
smn_notification	Yes	Map<String, SMNNotification >	SMN message notification structure, which is used to notify the user of the service status change
log_report_channels	No	Array of LogReportPipeline objects	Log channel group. If this parameter is not specified or the array length is 0, LTS log interconnection is disabled. This function cannot be modified after being enabled.

Parameter	Mandatory	Type	Description
websocket_upgrade	No	Boolean	<p>Whether the service interface is upgraded to WebSocket. During service deployment, the default value is false. During service configuration update, the default value is the value set last time.</p> <ul style="list-style-type: none"> • false: Do not upgrade to WebSocket. • true: Upgrade to WebSocket. This parameter cannot be modified after WebSocket is enabled. WebSocket cannot be enabled together with Traffic Limit.

Table 7-22 SmnNotification

Parameter	Mandatory	Type	Description
topic_urn	Yes	String	URN of an SMN topic
events	Yes	Array of integers	Event ID. Options: 1 : failed 3 : running 7 : concerning 11 : pending

Table 7-23 LogReportPipeline

Parameter	Mandatory	Type	Description
type	Yes	String	Log channel type. Currently, LTS is supported.
configuration	No	LtsConfiguration object	LTS Log Configuration

Table 7-24 LtsConfiguration

Parameter	Mandatory	Type	Description
log_group_id	Yes	String	Specifies the LTS log group ID. The value contains 64 characters.

Parameter	Mandatory	Type	Description
log_stream_id	Yes	String	LTS log stream ID. The value contains 64 characters.

Response Parameters

Status code: 200

Table 7-25 Response body parameters

Parameter	Type	Description
service_id	String	Service ID
resource_ids	Array of strings	Resource ID array for the resource IDs generated by the target model

Example Requests

- Sample request of creating a real-time service

POST https://{endpoint}/v1/{project_id}/services

```
{
  "infer_type": "real-time",
  "service_name": "mnist",
  "description": "mnist service",
  "config": [ {
    "specification": "modelarts.vm.cpu.2u",
    "weight": 100,
    "model_id": "0e07b41b-173e-42db-8c16-8e1b44cc0d44",
    "instance_count": 1
  } ]
}
```

- Sample request of creating a real-time service and configuring multiple versions for traffic distribution

POST https://{endpoint}/v1/{project_id}/services

```
{
  "service_name": "mnist",
  "description": "mnist service",
  "infer_type": "real-time",
  "config": [ {
    "model_id": "xxxmodel-idxxx",
    "weight": "70",
    "specification": "modelarts.vm.cpu.2u",
    "instance_count": 1,
    "envs": {
      "model_name": "mxnet-model-1",
      "load_epoch": "0"
    }
  }, {
    "model_id": "xxxxxx",
    "weight": "30",
    "specification": "modelarts.vm.cpu.2u",
    "instance_count": 1
  } ]
}
```

- Sample request of creating a real-time service in a dedicated resource pool with custom specifications

POST https://{endpoint}/v1/{project_id}/services

```
{
  "service_name": "realtime-demo",
  "description": "",
  "infer_type": "real-time",
  "cluster_id": "8abf68a969c3cb3a0169c4acb24b0000",
  "config": [ {
    "model_id": "eb6a4a8c-5713-4a27-b8ed-c7e694499af5",
    "weight": "100",
    "cluster_id": "8abf68a969c3cb3a0169c4acb24b0000",
    "specification": "custom",
    "custom_spec": {
      "cpu": 1.5,
      "memory": 7500,
      "gpu_p4": 0,
      "ascend_a310": 0
    },
  },
  "instance_count": 1
}]
}
```

- Sample request of creating a real-time service and configuring it to automatically stop

POST https://{endpoint}/v1/{project_id}/services

```
{
  "service_name": "service-demo",
  "description": "demo",
  "infer_type": "real-time",
  "config": [ {
    "model_id": "xxxmodel-idxxx",
    "weight": "100",
    "specification": "modelarts.vm.cpu.2u",
    "instance_count": 1
  } ],
  "schedule": [ {
    "type": "stop",
    "time_unit": "HOURS",
    "duration": 1
  } ]
}
```

- Sample request of creating a batch service and setting **mapping_type** to **file**

POST https://{endpoint}/v1/{project_id}/services

```
{
  "service_name": "batchservicetest",
  "description": "",
  "infer_type": "batch",
  "cluster_id": "8abf68a969c3cb3a0169c4acb24b****",
  "config": [ {
    "model_id": "598b913a-af3e-41ba-a1b5-bf065320f1e2",
    "specification": "modelarts.vm.cpu.2u",
    "instance_count": 1,
    "src_path": "https://infern-data.obs.xxxx.com/xgboosterdata/",
    "dest_path": "https://infern-data.obs.xxxx.com/output/",
    "req_uri": "/",
    "mapping_type": "file"
  } ]
}
```

- Sample request of creating a batch service and setting **mapping_type** to **csv**

POST https://{endpoint}/v1/{project_id}/services

```
{
```

```
"service_name" : "batchservicetest",
"description" : "",
"infer_type" : "batch",
"config" : [ {
  "model_id" : "598b913a-af3e-41ba-a1b5-bf065320f1e2",
  "specification" : "modelarts.vm.cpu.2u",
  "instance_count" : 1,
  "src_path" : "https://infers-data.obs.xxxx.com/xgboosterdata/",
  "dest_path" : "https://infers-data.obs.xxxx.com/output/",
  "req_uri" : "/",
  "mapping_type" : "csv",
  "mapping_rule" : {
    "type" : "object",
    "properties" : {
      "data" : {
        "type" : "object",
        "properties" : {
          "req_data" : {
            "type" : "array",
            "items" : [ {
              "type" : "object",
              "properties" : {
                "input5" : {
                  "type" : "number",
                  "index" : 0
                },
                "input4" : {
                  "type" : "number",
                  "index" : 1
                },
                "input3" : {
                  "type" : "number",
                  "index" : 2
                },
                "input2" : {
                  "type" : "number",
                  "index" : 3
                },
                "input1" : {
                  "type" : "number",
                  "index" : 4
                }
              }
            }
          ]
        }
      }
    }
  }
} ]
}
```

- Sample request for creating an edge service

POST https://{endpoint}/v1/{project_id}/services

```
{
  "service_name" : "service-edge-demo",
  "description" : "",
  "infer_type" : "edge",
  "config" : [ {
    "model_id" : "eb6a4a8c-5713-4a27-b8ed-c7e694499af5",
    "specification" : "custom",
    "custom_spec" : {
      "cpu" : 1.5,
      "memory" : 7500,
      "gpu_p4" : 0,
      "ascend_a310" : 0
    },
    "envs" : { },
    "nodes" : [ "2r8c4fb9-t497-40u3-89yf-skui77db0472" ]
  }
]
```

```
    }]  
  }
```

Example Responses

Status code: 200

Service deployed

```
{  
  "service_id" : "10eb0091-887f-4839-9929-cbc884f1e20e",  
  "resource_ids" : [ "INF-f878991839647358@1598319442708" ]  
}
```

Status Codes

Status Code	Description
200	Service deployed

Error Codes

See [Error Codes](#).

7.4 Obtaining Supported Service Deployment Specifications

Function

This API is used to obtain supported service deployment specifications.

URI

GET /v1/{project_id}/services/specifications

Table 7-26 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .

Table 7-27 Query Parameters

Parameter	Mandatory	Type	Description
is_personal_cluster	No	Boolean	Whether to obtain the service deployment flavors supported by dedicated resource pool. The default value is false .
infer_type	No	String	Inference mode. Options: <ul style="list-style-type: none"> • real-time, which is the default value • batch
limit	No	String	Maximum number of records returned on each page. The default value is 1000.
offset	No	String	Start offset of the returned data. Default value: 0

Request Parameters

Table 7-28 Request header parameters

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

Response Parameters

Status code: 200

Table 7-29 Response body parameters

Parameter	Type	Description
total_count	Integer	Total number of deployed service specifications.
count	Integer	Number of specifications in the current list.
specifications	Array of Specification objects	Supported service deployment flavors

Table 7-30 Specification

Parameter	Type	Description
is_open	Boolean	Whether to enable the flavor. The default value is true . If this parameter is set to false , submit a service ticket to apply for the flavor.
spec_status	String	Flavor status.
specification	String	Unique flavor ID
billing_spec	String	Unique ID of the billing specifications
category	String	The flavor type.
cpu_info	CpuInfo object	Specifies the CPU information of the flavor.
memory_info	MemoryInfo object	Specifies the memory information of the flavor.
gpu_info	GpuInfo object	Specifies the GPU information of the flavor.
npu_info	NpuInfo object	NPU information of the flavor.
source_type	String	Model type, which can be empty or auto . The default value is empty , indicating that the model is generated by the user. If the value is auto , the model is trained using ExeML. The billing mode varies depending on the model type.
is_free	Boolean	Whether the flavor is free of charge. The value true indicates that the flavor is free of charge.
over_quota	Boolean	Whether the quota exceeds the upper limit. The value true indicates that the quota exceeds the upper limit.
extend_params	Integer	Billing item
display_en	String	Specifications description in English
display_cn	String	Specification description

Table 7-31 CpuInfo

Parameter	Type	Description
arch	String	Architecture type.
cpu	Number	Number of CPUs

Table 7-32 MemoryInfo

Parameter	Type	Description
unit	String	Unit.
memory	Integer	Memory size.

Table 7-33 GpuInfo

Parameter	Type	Description
brand	String	Brand.
version	String	Card type.
unit	String	Unit
memory	Integer	Memory size of the card.
gpu	Number	Number of GPUs

Table 7-34 NpuInfo

Parameter	Type	Description
brand	String	Brand.
version	String	Card type.
unit	String	Unit
memory	Integer	Memory size of the card.
npu	Integer	Number of NPUs.

Example Requests

GET https://{endpoint}/v1/{project_id}/services/specifications

Example Responses

Status code: 200

Service deployment flavors

```
{
  "total_count": 3,
  "count": 3,
  "specifications": [ {
    "specification": "modelarts.vm.cpu.2u",
    "billing_spec": "modelarts.vm.cpu.2u",
    "category": "CPU",
    "cpu_info": {
      "arch": "x86",
```

```

"cpu" : 2.0
},
"memory_info" : {
  "unit" : "GB",
  "memory" : 8
},
},
"gpu_info" : {
  "unit" : "MB",
  "gpu" : 0.0,
  "memory" : 0
},
},
"npu_info" : {
  "unit" : "MB",
  "npu" : 0,
  "memory" : 0
},
},
"display_en" : "CPU: 2 vCPUs | 8 GiB",
"display_cn" : "CPU: 2 Core 8 GiB",
"is_open" : true,
"spec_status" : "normal",
"is_free" : false,
"over_quota" : false,
"extend_params" : 1
}, {
  "specification" : "modelarts.vm.gpu.p4",
  "billing_spec" : "modelarts.vm.gpu.p4",
  "category" : "GPU",
  "cpu_info" : {
    "arch" : "x86",
    "cpu" : 8.0
  },
},
"memory_info" : {
  "unit" : "GB",
  "memory" : 32
},
},
"gpu_info" : {
  "unit" : "GB",
  "brand" : "NVIDIA",
  "version" : "P4",
  "gpu" : 1.0,
  "memory" : 16
},
},
"npu_info" : {
  "unit" : "MB",
  "npu" : 0,
  "memory" : 0
},
},
"display_en" : "CPU: 8 vCPUs | 32 GiB GPU: P4",
"display_cn" : "CPU: 8 Core 32 GiB GPU: 1 * P4",
"is_open" : true,
"spec_status" : "normal",
"is_free" : false,
"over_quota" : false,
"extend_params" : 1
}, {
  "specification" : "modelarts.vm.ai1.a310",
  "billing_spec" : "modelarts.vm.ai1.a310",
  "category" : "NPU",
  "cpu_info" : {
    "arch" : "x86",
    "cpu" : 2.0
  },
},
"memory_info" : {
  "unit" : "GB",
  "memory" : 8
},
},
"gpu_info" : {
  "unit" : "MB",
  "gpu" : 0.0,

```

```

"memory" : 0
},
"npu_info" : {
  "unit" : "GB",
  "brand" : "Ascend",
  "version" : "D310",
  "npu" : 1,
  "memory" : 8
},
"display_en" : "CPU: 2 vCPUs | 8 GiB Ascend: 1 x Ascend 310",
"display_cn" : "CPU: 2 Core 8 GiB Ascend: 1 * Ascend 310",
"is_open" : false,
"spec_status" : "normal",
"is_free" : false,
"over_quota" : false,
"extend_params" : 1
}]
}

```

Status Codes

Status Code	Description
200	Service deployment flavors

Error Codes

See [Error Codes](#).

7.5 Obtaining Service Details

Function

This API is used to obtain the details about a model service based on the service ID.

URI

GET /v1/{project_id}/services/{service_id}

Table 7-35 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .
service_id	Yes	String	Service ID

Request Parameters

Table 7-36 Request header parameters

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

Response Parameters

Status code: 200

Table 7-37 Response body parameters

Parameter	Type	Description
service_id	String	Service ID
service_name	String	Service name
description	String	Service description
tenant	String	Tenant to which a service belongs
project	String	Project to which a service belongs
owner	String	User to which a service belongs
publish_at	Number	Latest service release time, in milliseconds calculated from 1970.1.1 0:0:0 UTC.
infer_type	String	Inference mode. Options: <ul style="list-style-type: none"> • real-time • batch
workspace_id	String	Workspace ID
cluster_id	String	ID of the dedicated resource pool used by the real-time or batch service. This parameter is available only when a dedicated resource pool is used.
vpc_id	String	ID of the VPC to which the real-time service instance belongs. This parameter is available when the network configuration is customized.
subnet_network_id	String	ID of the subnet to which the real-time service instance belongs. This parameter is available when the network configuration is customized.

Parameter	Type	Description
security_group_id	String	Security group to which the real-time service instance belongs. This parameter is available when the network configuration is customized.
status	String	Service status. The options are as follows: <ul style="list-style-type: none"> ● running: The service is running properly. ● deploying: The service is being deployed, including image creation and resource scheduling deployment. ● concerning: An alarm has been generated, indicating that some backend instances malfunction. ● failed: Deploying the service failed. For details about the failure cause, see the event and log tab pages. ● stopped: The service has been stopped. ● finished: Service running is completed. This status is available only for batch services. ● stopping: The VM is being stopped. ● deleting: The VM is being deleted. ● pending: to be started. This state is available only when the system is online. ● waiting: The resource is queuing. This state is available only for online services.
progress	Integer	Deployment progress. This parameter is available when the status is deploying .
error_msg	String	Error message. When status is failed , an error message carrying the failure cause is returned.
config	Array of QueryService Config objects	Service configuration (If a service is shared, only model_id , model_name , and model_version are returned.)
access_addresses	String	Access address of an inference request. This parameter is available when infer_type is set to real-time .
bind_access_address	String	Request address of a custom domain name. This parameter is available after a domain name is bound.
invocation_times	Number	Total number of service calls
failed_times	Number	Number of failed service calls
is_shared	Boolean	Whether a service is subscribed

Parameter	Type	Description
shared_count	Number	Number of subscribed services
schedule	Array of Schedule objects	Service scheduling configuration. If this parameter is not configured, no value will be returned.
update_time	Number	Time when the configuration used by the current service is updated, in milliseconds calculated from 1970.1.1 0:0:0 UTC.
debug_url	String	Online debugging address of a real-time service. This parameter is available only when the model supports online debugging and there is only one instance.
due_time	Number	Time when an online service automatically stops, in milliseconds calculated from 1970.1.1 0:0:0 UTC. If automatic stop is not configured, this parameter is not returned.
operation_time	Number	Operation time of a request
transition_at	Number	Time when the service status changes
is_free	Boolean	Whether a free-of-charge flavor is used
additional_properties	Map<String,String>	Additional service attribute
pool_name	String	Specifies the ID of the new dedicated resource pool. By default, this parameter is left blank, indicating that the dedicated resource pool is not used. This parameter corresponds to the ID of the new resource pool.

Table 7-38 QueryServiceConfig

Parameter	Type	Description
model_version	String	Model version
finished_time	Number	Task completion time, in milliseconds calculated from 1970.1.1 0:0:0 UTC. This parameter is not returned before the task is complete.
custom_spec	CustomSpec object	Custom resource specifications

Parameter	Type	Description
envs	Map<String,String>	Environment variable key-value pair required for running a model
specification	String	Resource flavor, which can be modelarts.vm.cpu.2u , modelarts.vm.gpu.p4 , or modelarts.vm.ai1.a310/custom
weight	Integer	Traffic weight allocated to a model
source_type	String	Model source. This parameter is returned when a model is created using ExeML. The value is auto .
model_id	String	Model ID
src_path	String	OBS path to the input data of a batch job
req_uri	String	Inference path of a batch job
mapping_type	String	Mapping type of the input data, which can be file or csv
start_time	Number	Task start time, in milliseconds calculated from 1970.1.1 0:0:0 UTC. This parameter is not returned before the task starts.
cluster_id	String	ID of a dedicated resource pool used by a service instance. This parameter is returned only when a dedicated resource pool is configured.
nodes	Array of Nodes objects	Node information
mapping_rule	Object	Mapping between input parameters and CSV data. This parameter is mandatory only when mapping_type is set to csv .
model_name	String	Model name
src_type	String	Data source type. This parameter is returned only when ManifestFile is used.
dest_path	String	OBS path to the output data of a batch job
instance_count	Integer	Number of instances deployed for a model

Parameter	Type	Description
status	String	Service status. Options: <ul style="list-style-type: none"> ● running: The service is running properly. ● deploying: The service is being deployed, including image creation and resource scheduling deployment. ● concerning: An alarm has been generated, indicating that some backend instances malfunction. ● failed: Deploying the service failed. For details about the failure cause, see the event and log tab pages. ● stopped: The service has been stopped. ● finished: Service running is completed. This status is available only for batch services.
scaling	Boolean	Whether auto scaling is enabled
support_debug	Boolean	Whether a model supports online debugging
additional_properties	Map<String,String>	Additional model deployment attribute
pool_name	String	Specifies the ID of the new dedicated resource pool. By default, this parameter is left blank, indicating that the dedicated resource pool is not used. This parameter corresponds to the ID of the new resource pool.

Table 7-39 CustomSpec

Parameter	Type	Description
gpu_p4	Float	Number of GPUs, which can be a decimal. The value cannot be smaller than 0, with the third decimal place is rounded off. This parameter is optional and is not used by default.
memory	Integer	Memory in MB, which must be an integer
cpu	Float	Number of CPU cores, which can be a decimal. The value cannot be smaller than 0.01, with the third decimal place is rounded off.
ascend_a310	Integer	Number of Ascend chips. This parameter is optional and is not used by default. Either this parameter or gpu_p4 is configured.

Table 7-40 Nodes

Parameter	Type	Description
memory	Integer	Memory size, in MB
os_version	String	OS version of a node
cpu	Integer	Number of CPU cores
created_at	String	Creation time, in the format of YYYY-MM-DDThh:mm:ss (UTC)
description	String	Description
message	String	Cause if instance_status is failed or notReady
predict_url	String	Inference URL of a node
enable_gpu	Boolean	Whether to enable GPUs
gpu_num	Integer	Number of GPUs
host_ips	Array of strings	Host IP address of a node
updated_at	String	Update time, in the format of YYYY-MM-DDThh:mm:ss (UTC)
node_label	String	Node label
os_type	String	OS type of a node
name	String	Name of an edge node
os_name	String	OS name of a node
arch	String	Node architecture
id	String	Edge node ID
instance_statuses	String	Status of a model instance on a node, which can be running , stopped , notReady , or failed
state	String	Host status, which can be RUNNING , FAIL , or UNCONNECTED
deployment_number	Integer	Number of application instances deployed on a node
host_name	String	Host name of a node

Table 7-41 Schedule

Parameter	Type	Description
duration	Integer	Value mapping a time unit. For example, if the task stops after two hours, set time_unit to HOURS and duration to 2 .
time_unit	String	Scheduling time unit. Possible values are DAYS , HOURS , and MINUTES .
type	String	Scheduling type. Only the value stop is supported.

Example Requests

```
GET https://{endpoint}/v1/{project_id}/services/{service_id}
```

Example Responses

Status code: 200

Service Details

```
{
  "service_id": "f76f20ba-78f5-44e8-893a-37c8c600c02f",
  "service_name": "service-demo",
  "tenant": "xxxxx",
  "project": "xxxxx",
  "owner": "xxxxx",
  "publish_at": 1585809231902,
  "update_time": 1585809358259,
  "infer_type": "real-time",
  "status": "running",
  "progress": 100,
  "access_address": "https://xxxxx.apigw.xxxx.com/v1/infers/088458d9-5755-4110-97d8-1d21065ea10b/f76f20ba-78f5-44e8-893a-37c8c600c02f",
  "cluster_id": "088458d9-5755-4110-97d8-1d21065ea10b",
  "workspace_id": "0",
  "additional_properties": { },
  "is_shared": false,
  "invocation_times": 0,
  "failed_times": 0,
  "shared_count": 0,
  "operation_time": 1586249085447,
  "config": [ {
    "model_id": "044ebf3d-8bf4-48df-bf40-bad0e664c1e2",
    "model_name": "jar-model",
    "model_version": "1.0.1",
    "specification": "custom",
    "custom_spec": { },
    "status": "notReady",
    "weight": 100,
    "instance_count": 1,
    "scaling": false,
    "envs": { },
    "additional_properties": { },
    "support_debug": false
  } ],
  "transition_at": 1585809231902,
  "is_free": false
}
```

Status Codes

Status Code	Description
200	Service Details

Error Codes

See [Error Codes](#).

7.6 Updating Service Configurations

Function

This API is used to update configurations of a model service. It can also be used to start or stop a service.

URI

PUT /v1/{project_id}/services/{service_id}

Table 7-42 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .
service_id	Yes	String	Service ID

Request Parameters

Table 7-43 Request header parameters

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

Table 7-44 Request body parameters

Parameter	Mandatory	Type	Description
schedule	No	Array of Schedule objects	Service scheduling configuration, which can be configured only for real-time services. By default, this parameter is not used. Services run for a long time.
description	No	String	Service description, which contains a maximum of 100 characters and cannot contain the following characters: <>=&"". If this parameter is not set, no update is performed.
config	No	Array of ServiceConfig objects	Service configuration. If this parameter is not configured, the service is not updated.
status	No	String	Service status, which can be running or stopped . If this parameter is not configured, the service status is not changed. Either status or config can be modified. If both of them are used, modify status only.
additional_properties	No	Map<String, ServiceAdditionalProperties >	Additional service attribute, which facilitates service management

Table 7-45 Schedule

Parameter	Mandatory	Type	Description
duration	Yes	Integer	Value mapping a time unit. For example, if the task stops after two hours, set time_unit to HOURS and duration to 2 .
time_unit	Yes	String	Scheduling time unit. Possible values are DAYS , HOURS , and MINUTES .
type	Yes	String	Scheduling type. Only the value stop is supported.

Table 7-46 ServiceConfig

Parameter	Mandatory	Type	Description
custom_spec	No	CustomSpec object	Custom resource specifications
envs	No	Map<String,String>	Common parameter. (Optional) Environment variable key-value pair required for running a model. By default, this parameter is left blank.
specification	Yes	String	Common parameters Resource specifications. You can obtain the specification list by querying the supported service deployment specifications. In the current version, modelarts.vm.cpu.2u/modelarts.vm.gpu.p4 (needs to be applied for), modelsarts.vm.ai1.a310 (needs to be applied for), and custom (supported only when deployed in a dedicated resource pool) are available. [Submit a service ticket.] (tag: hc, hk) ModelArts O&M engineers add permissions. If this parameter is set to custom, the custom_spec parameter must be specified.
weight	No	Integer	This parameter is mandatory for real-time . Weight of traffic allocated to a model. This parameter is mandatory only when infer_type is set to real-time . The sum of all weights must be equal to 100. If multiple model versions are configured with different traffic weights in a real-time service, ModelArts will continuously access the prediction API of the service and forward prediction requests to the model instances of the corresponding versions based on the weights.

Parameter	Mandatory	Type	Description
deploy_timeout_in_seconds	No	Integer	Timeout interval for deploying a single model instance
model_id	Yes	String	Common parameters Model ID. You can obtain the value by calling the API for querying the AI application list.
src_path	No	String	Mandatory for batch services. OBS path to the input data of a batch job
req_uri	No	String	Mandatory for batch services. Inference API called in a batch task, which is the RESTful API exposed in the model image. You must select an API URL from the config.json file of the model for inference. If a built-in inference image of ModelArts is used, the API is displayed as <code>/</code> .
mapping_type	No	String	<p>The batch service type is mandatory. Mapping type of the input data. The value can be file or csv.</p> <ul style="list-style-type: none"> • If file is selected, each inference request corresponds to a file in the input data directory. When this mode is used, req_uri corresponding to the model can have only one input parameter and the parameter type is file. • If csv is selected, each inference request corresponds to a row of data in the CSV file. If this mode is used, the file name extension in the input data directory must be .csv, and the mapping_rule parameter must be configured to indicate the CSV index corresponding to each parameter in the inference request body.

Parameter	Mandatory	Type	Description
cluster_id	No	String	Optional for real-time services. ID of a dedicated resource pool. This parameter is left blank by default, indicating that no dedicated resource pool is used. When using a dedicated resource pool to deploy services, ensure that the resource pool is running properly. After this parameter is configured, the network configuration of the cluster is used, and the vpc_id parameter does not take effect.
pool_name	No	String	Specifies the ID of the new dedicated resource pool. By default, this parameter is left blank, indicating that the dedicated resource pool is not used. This parameter corresponds to the ID of the new resource pool. When using dedicated resource pool to deploy services, ensure that the cluster status is normal. If pool_name in real-time config and pool_name in real-time config are configured at the same time, pool_name in real-time config is preferred.
nodes	No	Array of strings	Mandatory for edge services. Edge node ID array. The node ID is the edge node ID on IEF, which can be obtained after the edge node is created on IEF.

Parameter	Mandatory	Type	Description
mapping_rule	No	Object	Optional for batch services. Mapping between input parameters and CSV data. This parameter is mandatory only when mapping_type is set to csv . The mapping rule is similar to the definition of the input parameters in the config.json file. You only need to configure the index parameters under each parameter of the string, number, integer, or boolean type, and specify the value of this parameter to the values of the index parameters in the CSV file to send an inference request. Use commas (,) to separate multiple pieces of CSV data. The values of the index parameters start from 0 . If the value of the index parameter is -1 , ignore this parameter. For details, see the sample of creating a batch service.
src_type	No	String	Mandatory for batch services. Data source type, which can be ManifestFile . By default, this parameter is left blank, indicating that only files in the src_path directory are read. If this parameter is set to ManifestFile , src_path must be set to a specific manifest path. Multiple data paths can be specified in the manifest file. For details, see the manifest inference specifications.
dest_path	No	String	Mandatory for batch services. OBS path to the output data of a batch job

Parameter	Mandatory	Type	Description
instance_count	Yes	Integer	Common parameter. Number of instances deployed for a model. The maximum number of instances is 5. To use more instances, submit a service ticket.
additional_properties	No	Map<String,ModelAdditionalProperties>	Additional attributes for model deployment, facilitating service instance management

Table 7-47 CustomSpec

Parameter	Mandatory	Type	Description
gpu_p4	No	Float	Number of GPUs, which can be a decimal. The value cannot be smaller than 0, with the third decimal place is rounded off. This parameter is optional and is not used by default.
memory	Yes	Integer	Memory in MB, which must be an integer
cpu	Yes	Float	Number of CPU cores, which can be a decimal. The value cannot be smaller than 0.01, with the third decimal place is rounded off.
ascend_a310	No	Integer	Number of Ascend chips. This parameter is optional and is not used by default. Either this parameter or gpu_p4 is configured.

Table 7-48 ModelAdditionalProperties

Parameter	Mandatory	Type	Description
log_volume	No	Array of log_volume objects	Host directory mounting. This parameter takes effect only if a dedicated resource pool is used. If a public resource pool is used to deploy services, this parameter cannot be configured. Otherwise, an error will occur.
max_surge	No	Float	The value must be greater than 0. If this parameter is not set, the default value 1 is used. If the value is less than 1, it indicates the percentage of instances to be added during the rolling upgrade. If the value is greater than 1, it indicates the maximum number of instances to be added during the rolling upgrade.
max_unavailable	No	Float	The value must be greater than 0. If this parameter is not set, the default value 0 is used. If the value is less than 1, it indicates the percentage of instances that can be scaled in during the rolling upgrade. If the value is greater than 1, it indicates the number of instances that can be scaled in during the rolling upgrade.
termination_grace_period_seconds	No	Integer	Graceful stop time of a container
persistent_volumes	No	Array of persistent_volumes objects	Persistent Storage Mounting Configuration

Table 7-49 log_volume

Parameter	Mandatory	Type	Description
host_path	Yes	String	Log path to be mapped on the host

Parameter	Mandatory	Type	Description
mount_path	Yes	String	Path to the logs in the container

Table 7-50 persistent_volumes

Parameter	Mandatory	Type	Description
name	No	String	Specifies the name of the storage volume.
mount_path	Yes	String	Mount path of the storage volume in the container.

Table 7-51 ServiceAdditionalProperties

Parameter	Mandatory	Type	Description
smn_notification	Yes	Map<String, SMNNotification >	SMN message notification structure, which is used to notify the user of the service status change
log_report_channels	No	Array of LogReportPipeline objects	Log channel group. If this parameter is not specified or the array length is 0, LTS log interconnection is disabled. This function cannot be modified after being enabled.
websocket_upgrade	No	Boolean	Whether the service interface is upgraded to WebSocket. During service deployment, the default value is false . During service configuration update, the default value is the value set last time. <ul style="list-style-type: none"> false: Do not upgrade to WebSocket. true: Upgrade to WebSocket. This parameter cannot be modified after WebSocket is enabled. WebSocket cannot be enabled together with Traffic Limit.

Table 7-52 SmnNotification

Parameter	Mandatory	Type	Description
topic_urn	Yes	String	URN of an SMN topic
events	Yes	Array of integers	Event ID. Options: 1 : failed 3 : running 7 : concerning 11 : pending

Table 7-53 LogReportPipeline

Parameter	Mandatory	Type	Description
type	Yes	String	Log channel type. Currently, LTS is supported.
configuration	No	LtsConfigurat ion object	LTS Log Configuration

Table 7-54 LtsConfiguration

Parameter	Mandatory	Type	Description
log_group_id	Yes	String	Specifies the LTS log group ID. The value contains 64 characters.
log_stream_id	Yes	String	LTS log stream ID. The value contains 64 characters.

Response Parameters

None

Example Requests

The following shows how to update a real-time service.

```
PUT https://{endpoint}/v1/{project_id}/services/{service_id}
```

```
{
  "description": "",
  "status": "running",
  "config": [ {
    "model_id": "xxx",
    "weight": "100",
    "specification": "modelarts.vm.cpu.2u",
    "instance_count": 1
  } ]
}
```

Example Responses

Status code: 200

Service updated.

```
{}
```

Status Codes

Status Code	Description
200	Service updated.

Error Codes

See [Error Codes](#).

7.7 Deleting a Service

Function

This API is used to delete a model service. You can delete your own services only.

URI

DELETE /v1/{project_id}/services/{service_id}

Table 7-55 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .
service_id	Yes	String	Service ID. To delete multiple services in a batch, use commas (,) to separate multiple service_id values.

Request Parameters

Table 7-56 Request header parameters

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

Response Parameters

None

Example Requests

The following shows how to delete the model service with ID **xxxxxx**.

```
DELETE https://endpoint/v1/{project_id}/services/xxxxxx
```

Example Responses

Status code: 200

The service is deleted.

```
{}
```

Status Codes

Status Code	Description
200	The service is deleted.

Error Codes

See [Error Codes](#).

7.8 Obtaining Dedicated Resource Pools

Function

This API is used to obtain dedicated resource pools.

URI

GET /v1/{project_id}/clusters

Table 7-57 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .

Table 7-58 Query Parameters

Parameter	Mandatory	Type	Description
cluster_name	No	String	Cluster name. By default, the cluster name is not filtered.
status	No	String	Cluster status. By default, the status is not filtered.
offset	No	Integer	Start page for pagination display. The default value is 0 .
limit	No	Integer	Maximum number of records returned on each page. Default value: 1000
sort_by	No	String	Sorting field. Options: <ul style="list-style-type: none"> • created_at: default value • cluster_name
order	No	String	Sorting mode. Options: <ul style="list-style-type: none"> • asc: ascending order • desc: descending order (default value)

Request Parameters

Table 7-59 Request header parameters

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

Response Parameters

Status code: 200

Table 7-60 Response body parameters

Parameter	Type	Description
total_count	Integer	Total number of clusters that meet the search criteria when no paging is performed
count	Integer	Number of clusters in the query result. If offset and limit are not configured, the values of count and total are the same.
clusters	Array of Cluster objects	Obtained clusters

Table 7-61 Cluster

Parameter	Type	Description
owner	String	User to which a cluster belongs
cluster_name	String	Cluster name
period_num	Integer	Number of subscription periods. This parameter is returned only when the cluster is billed on a yearly/monthly basis.
created_at	Integer	Time when a cluster is created, in milliseconds calculated from 1970.1.1 0:0:0 UTC.
description	String	Cluster description
project	String	Project to which a cluster belongs

Parameter	Type	Description
allocatable_memory	Integer	Number of available memory resources
cluster_id	String	Cluster ID
nodes	ClusterNode object	Cluster node configurations
allocatable_cpu_cores	Float	Number of available CPU cores
product_id	String	Product ID. This parameter is returned only when the cluster is billed on a yearly/monthly basis.
allocatable_gpus	Float	Number of available GPUs
order_id	String	Order ID. This parameter is returned only when the cluster is billed on a yearly/monthly basis.
period_type	String	Subscription period, which can be year or month . This parameter is returned only when the cluster is billed on a yearly/monthly basis.
tenant	String	Tenant to which a cluster belongs
status	String	Cluster status. Options: <ul style="list-style-type: none"> • deploying: The cluster is being deployed. • running: The cluster is running. • concerning: An alarm has been reported for the cluster. • abnormal: The cluster malfunctions.

Table 7-62 ClusterNode

Parameter	Type	Description
available_count	Integer	Number of available nodes
count	Integer	Number of nodes
specification	String	Node specifications

Example Requests

GET https://{endpoint}/v1/{project_id}/clusters

Example Responses

Status code: 200

Service deployment flavors

```
{
  "total_count" : 1,
  "count" : 1,
  "clusters" : [ {
    "cluster_id" : "ff808081673fbb3d01673fbb824d0000",
    "cluster_name" : "my-cluster",
    "description" : "",
    "tenant" : "tenant_id",
    "project" : "project_id",
    "owner" : "owner_id",
    "created_at" : 1574923077927,
    "status" : "running",
    "nodes" : {
      "specification" : "modelarts.vm.cpu.8ud",
      "count" : 1,
      "available_count" : 1
    },
    "allocatable_cpu_cores" : 5.5,
    "allocatable_memory" : 4096,
    "allocatable_gpu" : 0.5
  } ]
}
```

Status Codes

Status Code	Description
200	Service deployment flavors

Error Codes

See [Error Codes](#).

7.9 Obtaining Service Event Logs

Function

This API is used to obtain service event logs, including service operation records, key actions during deployment, and deployment failure causes.

URI

GET /v1/{project_id}/services/{service_id}/events

Table 7-63 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .
service_id	Yes	String	Service ID

Table 7-64 Query Parameters

Parameter	Mandatory	Type	Description
event_type	No	String	Event type. Options: <ul style="list-style-type: none"> • normal: normal events • abnormal: abnormal events
start_time	No	Number	Start time for filtering events. By default, events are not filtered.
end_time	No	Number	End time for filtering events. By default, events are not filtered.
offset	No	Integer	Start page for pagination display. The default value is 0 .
limit	No	Integer	Maximum number of records returned on each page. Default value: 1000
sort_by	No	String	Specifies the sorting field. The default value is occur_time (event generation time).
order	No	String	Sorting mode. Options: <ul style="list-style-type: none"> • asc: ascending order • desc: descending order, which is the default value

Request Parameters

Table 7-65 Request header parameters

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

Response Parameters

Status code: 200

Table 7-66 Response body parameters

Parameter	Type	Description
service_name	String	Service name
total_count	Integer	Total number of events that meet the search criteria when no paging is performed
service_id	String	Service ID
count	Integer	Number of events in the query result
events	Array of Events objects	Event logs of a service

Table 7-67 Events

Parameter	Type	Description
event_type	String	Event type. Options: <ul style="list-style-type: none"> normal: normal events abnormal: abnormal events
event_info	String	Event information, which mainly describes the the five phases of the deployment process. More information can be supplemented later. The five phases are image building, environment preparation, resource scheduling, image pulling, and model startup.

Parameter	Type	Description
occur_time	Number	Time when an event occurs, in milliseconds calculated from 1970.1.1 0:0:0 UTC.

Example Requests

GET https://{endpoint}/v1/{project_id}/services/{service_id}/events

Example Responses

Status code: 200

Event logs of a service

```
{
  "service_name": "service-07085",
  "total_count": 9,
  "service_id": "35de3ca9-1bca-4ae7-9cb0-914f30fa7d3e",
  "count": 9,
  "events": [ {
    "event_type": "normal",
    "event_info": "start to deploy service",
    "occur_time": 1562597251764
  }, {
    "event_type": "normal",
    "event_info": "building image for model [TF 3.0.0]",
    "occur_time": 1562597251788
  }, {
    "event_type": "normal",
    "event_info": "model (TF 3.0.0) build image success",
    "occur_time": 1562597251805
  }, {
    "event_type": "normal",
    "event_info": "preparing environment",
    "occur_time": 1562597255744
  }, {
    "event_type": "normal",
    "event_info": "[TF 3.0.0] prepare environment success",
    "occur_time": 1562597275915
  }, {
    "event_type": "normal",
    "event_info": "[TF 3.0.0] schedule resource success",
    "occur_time": 1562597275921
  }, {
    "event_type": "normal",
    "event_info": "[TF 3.0.0] pulling model image",
    "occur_time": 1562597275928
  }, {
    "event_type": "normal",
    "event_info": "[TF 3.0.0] pull image success",
    "occur_time": 1562597332570
  }, {
    "event_type": "normal",
    "event_info": "[TF 3.0.0] starting model",
    "occur_time": 1562597332582
  }
  ]
}
```

Status Codes

Status Code	Description
200	Event logs of a service

Error Codes

See [Error Codes](#).

7.10 Obtaining Service Update Logs

Function

This API is used to obtain the update logs of a real-time service.

URI

GET /v1/{project_id}/services/{service_id}/logs

Table 7-68 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .
service_id	Yes	String	Service ID

Table 7-69 Query Parameters

Parameter	Mandatory	Type	Description
update_time	No	Number	Update time for filtering. This parameter can be used to obtain the update logs of a real-time service. By default, update logs are not filtered.

Request Parameters

Table 7-70 Request header parameters

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

Response Parameters

Status code: 200

Table 7-71 Response body parameters

Parameter	Type	Description
service_name	String	Service name
service_id	String	Service ID
logs	Array of Log objects	Service update records

Table 7-72 Log

Parameter	Type	Description
update_time	Number	Update time, in milliseconds calculated from 1970.1.1 0:0:0 UTC.
result	String	Update result. Options: <ul style="list-style-type: none"> • SUCCESS • FAIL • RUNNING
config	Array of config objects	Model running configurations
success_num	Number	Number of nodes that are successfully operated. This parameter is returned when infer_type is set to edge .
failed_num	Number	Number of nodes that fail to be operated. This parameter is returned when infer_type is set to edge .

Parameter	Type	Description
result_detail	Array of UpdateResult objects	Operation result details. This parameter is returned when infer_type is set to edge .
cluster_id	String	ID of a dedicated resource pool
extend_config	Array of strings	Personalized configuration

Table 7-73 config

Parameter	Type	Description
model_id	String	Common parameters Model ID You can obtain the value by calling the API for querying the AI application list.
model_name	String	Model name
model_version	String	Model version
weight	Integer	Weight of traffic allocated to a model. This parameter is mandatory only when infer_type is set to real-time . The sum of all weights must be equal to 100. If multiple model versions are configured with different traffic weights in a real-time service, ModelArts will continuously access the prediction API of the service and forward prediction requests to the model instances of the corresponding versions based on the weights.
specification	String	Resource flavor. This parameter is returned when infer_type is set to real-time .
custom_spec	CustomSpec object	Custom resource specifications
envs	Map<String,String>	Environment variable key-value pair required for running a model
cluster_id	String	ID of a dedicated resource pool
instance_count	Integer	Number of instances where a model is deployed. This parameter is returned when infer_type is set to real-time .

Table 7-74 CustomSpec

Parameter	Type	Description
gpu_p4	Float	Number of GPUs, which can be a decimal. The value cannot be smaller than 0, with the third decimal place is rounded off. This parameter is optional and is not used by default.
memory	Integer	Memory in MB, which must be an integer
cpu	Float	Number of CPU cores, which can be a decimal. The value cannot be smaller than 0.01, with the third decimal place is rounded off.
ascend_a310	Integer	Number of Ascend chips. This parameter is optional and is not used by default. Either this parameter or gpu_p4 is configured.

Table 7-75 UpdateResult

Parameter	Type	Description
result	Boolean	Operation result. true indicates that the operation is successful, and false indicates that the operation failed.
node_name	String	Name of an edge node
operation	String	Operation. Options: <ul style="list-style-type: none"> • deploy • delete
node_id	String	Edge node ID You can obtain the value after creating an edge node on IEF.

Example Requests

```
GET https://{endpoint}/v1/{project_id}/services/{service_id}/logs
```

Example Responses

Status code: 200

Service update logs

```
{
  "service_name": "mnist",
  "service_id": "195c1f2d-136d-40af-a0f3-db5717d2634a",
  "logs": [ {
    "result": "SUCCESS",
    "update_time": 1574770228888,
    "config": [ {
      "model_version": "0.0.1",
      "specification": "modelarts.vm.cpu.2u",

```

```
"model_name" : "minst",  
"weight" : 100,  
"model_id" : "0e07b41b-173e-42db-8c16-8e1b44cc0d44",  
"instance_count" : 1  
}]  
}]  
}
```

Status Codes

Status Code	Description
200	Service update logs

Error Codes

See [Error Codes](#).

8 Resource Management

8.1 Configuration Management

8.1.1 Querying OS Configuration Parameters

Function

This API is used to obtain the configuration parameters of the ModelArts OS service, such as the CIDR block and user resource quota.

URI

GET /v1/{project_id}/os-user-config

Table 8-1 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .

Request Parameters

None

Response Parameters

Status code: 200

Table 8-2 Response body parameters

Parameter	Type	Description
networkCidrs	Array of strings	Network parameters
networkQuota	Integer	Network quota
poolQuota	Integer	Resource pool quota
poolHighAvailable	Boolean	Whether resource pools with high availability can be created in the current environment or region

Status code: 404

Table 8-3 Response body parameters

Parameter	Type	Description
error_code	String	Error code Minimum: 8 Maximum: 36
error_msg	String	Error message Minimum: 2 Maximum: 512

Example Requests

None

Example Responses

Status code: 200

OK

```
{
  "networkCidrs": [ "192.168.0.0/24", "172.16.31.0/16" ],
  "networkQuota": 15,
  "poolQuota": 15,
  "poolHighAvailable": true
}
```

Status Codes

Status Code	Description
200	OK
404	Not found

Error Codes

See [Error Codes](#).

8.2 Quota Management

8.2.1 Obtaining OS Quotas

Function

This API is used to obtain the quotas of some ModelArts OS resources, such as the quotas for resource pools and networks.

URI

GET /v1/{project_id}/quotas

Table 8-4 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .

Request Parameters

None

Response Parameters

Status code: 200

Table 8-5 Response body parameters

Parameter	Type	Description
quotas	quotas object	Resource quotas

Table 8-6 quotas

Parameter	Type	Description
resources	Array of resources objects	Resource quota information

Table 8-7 resources

Parameter	Type	Description
type	String	Resource type
quota	Integer	Upper limit of the resource quota
used	Integer	Used quota

Status code: 404

Table 8-8 Response body parameters

Parameter	Type	Description
error_code	String	Error code Minimum: 8 Maximum: 36
error_msg	String	Error message Minimum: 2 Maximum: 512

Example Requests

None

Example Responses

Status code: 200

OK

```
{
  "quotas": {
    "resources": [ {
      "type": "pool",
      "quota": 15,
      "used": 10
    }, {
      "type": "network",
      "quota": 15,
```



```
"used" : 10
  } ]
}
}
```

Status Codes

Status Code	Description
200	OK
404	Not found

Error Codes

See [Error Codes](#).

8.3 Event Management

8.3.1 Obtaining the Event List

Function

This API is used to obtain the event list.

URI

GET /v1/{project_id}/events

Table 8-9 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name . Minimum: 32 Maximum: 36

Table 8-10 Query Parameters

Parameter	Mandatory	Type	Description
resource	Yes	String	Type of the resource for which an event occurs. Options: <ul style="list-style-type: none"> pools: resource pool

Parameter	Mandatory	Type	Description
name	Yes	String	Name of the resource for which an event occurs
limit	No	Integer	Maximum number of records on each page. If this parameter is left blank or set to 0 , 500 records are returned by default. A maximum of 500 records are allowed on each page.
offset	No	Integer	Offset of the pagination query. This parameter is left blank when the first page is queried.
type	No	String	Event type. Options: <ul style="list-style-type: none"> • Normal: queries normal events. • Warning: queries alarm events. If this parameter is left blank, all types of events are returned.
since	No	Integer	Event start timestamp
until	No	Integer	Event end timestamp
ascend	No	Boolean	Whether the results are sorted in ascending order

Request Parameters

None

Response Parameters

Status code: 200

Table 8-11 Response body parameters

Parameter	Type	Description
apiVersion	String	API version. Options: <ul style="list-style-type: none"> • v1
kind	String	Resource type. Options: <ul style="list-style-type: none"> • EventList: event list
items	Array of Event objects	Events

Parameter	Type	Description
total	Integer	Total number of events

Table 8-12 Event

Parameter	Type	Description
apiVersion	String	API version. Options: <ul style="list-style-type: none"> • v1
kind	String	Resource type. Options: <ul style="list-style-type: none"> • Event: event
type	String	Event type. Options: <ul style="list-style-type: none"> • Normal: normal event • Warning: abnormal event
firstTimestamp	String	Time when an event occurred for the first time
lastTimestamp	String	Time when an event occurred for the last time
count	Integer	Consecutive occurrences of an event
reason	String	Event cause
message	String	Event details

Status code: 400

Table 8-13 Response body parameters

Parameter	Type	Description
error_code	String	Error code Minimum: 8 Maximum: 36
error_msg	String	Error message Minimum: 2 Maximum: 512

Status code: 404

Table 8-14 Response body parameters

Parameter	Type	Description
error_code	String	Error code Minimum: 8 Maximum: 36
error_msg	String	Error message Minimum: 2 Maximum: 512

Example Requests

Querying events of resource pool **pool-6f5da086876d4cd084d36f8bd3346036** by page

```
/v1/{project_id}/events?
resource=pools&name=pool-6f5da086876d4cd084d36f8bd3346036&limit=5&continue=cde36780-1120-4753
-bf75-0edb9ebd5a9e
{ }
```

Example Responses

Status code: 200

OK

```
{
  "kind" : "EventList",
  "apiVersion" : "v1",
  "metadata" : {
    "continue" : "52eddc13-cfad-42d3-ae4-92fea5813e7f"
  },
  "items" : [ {
    "kind" : "Event",
    "apiVersion" : "v1",
    "type" : "Warning",
    "firstTimestamp" : "2022-12-30T02:16:19Z",
    "lastTimestamp" : "2022-12-30T02:16:19Z",
    "count" : 1,
    "reason" : "PoolResourcesStatusChange",
    "message" : "Pool resources status changed, available/abnormal/creating/deleting count from 1/0/0/0 to 0/1/0/0, timestamp: 1672366579."
  }, {
    "kind" : "Event",
    "apiVersion" : "v1",
    "type" : "Normal",
    "firstTimestamp" : "2023-01-02T09:02:45Z",
    "lastTimestamp" : "2023-01-02T09:02:45Z",
    "count" : 1,
    "reason" : "PoolResourcesStatusChange",
    "message" : "Pool resources status changed, available/abnormal/creating/deleting count from 0/1/0/0 to 1/0/0/0, timestamp: 1672650165."
  }, {
    "kind" : "Event",
    "apiVersion" : "v1",
    "type" : "Warning",
    "firstTimestamp" : "2023-01-16T06:55:35Z",
    "lastTimestamp" : "2023-01-16T06:55:35Z",
```

```

"count" : 1,
"reason" : "PoolStatusChange",
"message" : "Pool status changed, from Running to Abnormal, details: ."
}, {
"kind" : "Event",
"apiVersion" : "v1",
"type" : "Warning",
"firstTimestamp" : "2023-01-16T06:57:51Z",
"lastTimestamp" : "2023-01-16T06:57:51Z",
"count" : 1,
"reason" : "PoolResourcesStatusChange",
"message" : "Pool resources status changed, available/abnormal/creating/deleting count from 1/0/0/0 to 0/1/0/0, timestamp: 1673852271."
}, {
"kind" : "Event",
"apiVersion" : "v1",
"type" : "Normal",
"firstTimestamp" : "2023-01-29T02:29:04Z",
"lastTimestamp" : "2023-01-29T02:29:04Z",
"count" : 1,
"reason" : "PoolStatusChange",
"message" : "Pool status changed, from Abnormal to Running."
}
}

```

Status code: 400

Bad request

```

{
"error_code" : "ModelArts.50004000",
"error_msg" : "Bad request."
}

```

Status code: 404

Not found

```

{
"error_code" : "ModelArts.50015001",
"error_msg" : "Pool {name} not found."
}

```

Status Codes

Status Code	Description
200	OK
400	Bad request
404	Not found

Error Codes

See [Error Codes](#).

8.4 Resource Pool Job Management

8.4.1 Obtaining Jobs in a Dedicated Resource Pool

Function

This API is used to obtain jobs in a dedicated resource pool.

URI

GET /v2/{project_id}/pools/{pool_name}/workloads

Table 8-15 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name . Minimum: 32 Maximum: 36
pool_name	Yes	String	Namespace to which a job belongs

Table 8-16 Query Parameters

Parameter	Mandatory	Type	Description
type	No	String	Service type of a job. Options: <ul style="list-style-type: none"> • train: training job • infer: inference job • notebook: notebook job

Parameter	Mandatory	Type	Description
status	No	String	Job status. Options: <ul style="list-style-type: none"> • Queue: The job is queuing. • Pending: The job is pending. • Abnormal: The job is abnormal. • Terminating: The job is being terminated. • Creating: The job is being created. • Running: The job is running. • Completed: The job has been completed. • Terminated: The job has been terminated. • Failed: The job failed to be run.
sort	No	String	Sorting field. Options: <ul style="list-style-type: none"> • create_time: Sort jobs by creation time.
ascend	No	Boolean	Whether the results are sorted in ascending order
offset	No	String	Start position for pagination query
limit	No	Integer	Maximum number of records on each page. If this parameter is left blank or set to 0 , 500 records are returned by default. A maximum of 500 records are allowed on each page.

Request Parameters

None

Response Parameters

Status code: 200

Table 8-17 Response body parameters

Parameter	Type	Description
apiVersion	String	API version. Options: <ul style="list-style-type: none"> • v1
kind	String	Resource type. Options: <ul style="list-style-type: none"> • WorkloadList: job list
items	Array of Workload objects	Job list

Table 8-18 Workload

Parameter	Type	Description
apiVersion	String	Resource version. Options: <ul style="list-style-type: none"> • v1
kind	String	Resource type. Options: <ul style="list-style-type: none"> • Workload
type	String	Service type of a job. Options: <ul style="list-style-type: none"> • train: training job
namespace	String	Name of the resource pool to which a job belongs
name	String	Job name
jobName	String	Upper-layer service job name
uid	String	Job UID
jobUUID	String	Upper-layer service job ID
flavor	String	Job specifications
status	String	Job status
resourceRequirement	resourceRequirement object	Number of resources requested for running a job
priority	String	Job priority
runningDuration	Integer	Running duration of a job
pendingDuration	Integer	Queuing duration of a job

Parameter	Type	Description
pendingPosition	Integer	Queuing position of a job
createTime	Integer	Job creation time
gvk	String	GVK of a job
hostIps	String	IP addresses of the nodes where a job is running. The values are separated with commas (,).

Table 8-19 resourceRequirement

Parameter	Type	Description
cpu	String	CPU usage
memory	String	Memory usage
nvidia.com/gpu	String	GPU usage
huawei.com/ascend-310	String	Ascend D310 usage
huawei.com/ascend-1980	String	Ascend snt9 usage

Status code: 400

Table 8-20 Response body parameters

Parameter	Type	Description
error_code	String	Error code Minimum: 8 Maximum: 36
error_msg	String	Error message Minimum: 2 Maximum: 512

Status code: 404

Table 8-21 Response body parameters

Parameter	Type	Description
error_code	String	Error code Minimum: 8 Maximum: 36
error_msg	String	Error message Minimum: 2 Maximum: 512

Example Requests

None

Example Responses

Status code: 400

Bad request

```
{
  "error_code" : "ModelArts.50004000",
  "error_msg" : "Bad request."
}
```

Status code: 404

Not found

```
{
  "error_code" : "ModelArts.50015001",
  "error_msg" : "Pool {name} not found."
}
```

Status Codes

Status Code	Description
200	OK
400	Bad request
404	Not found

Error Codes

See [Error Codes](#).

8.4.2 Obtaining Statistics About Dedicated Resource Pool Jobs

Function

This API is used to obtain statistics about dedicated resource pool jobs.

URI

GET /v2/{project_id}/statistics/pools/{pool_name}/workloads

Table 8-22 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name . Minimum: 32 Maximum: 36
pool_name	Yes	String	Resource pool to which a job belongs

Request Parameters

None

Response Parameters

Status code: 200

Table 8-23 Response body parameters

Parameter	Type	Description
statistics	statistics object	Job statistics

Table 8-24 statistics

Parameter	Type	Description
total	Integer	Number of statistics lists
items	Array of WorkloadStatistics objects	Statistics of jobs with a specific type

Table 8-25 WorkloadStatistics

Parameter	Type	Description
type	String	Job type. Options: <ul style="list-style-type: none"> ● train: training job ● infer: inference job ● notebook: notebook job
total	Integer	Number of jobs
status	status object	Number of jobs in different statuses

Table 8-26 status

Parameter	Type	Description
Queue	Integer	Number of queuing jobs
Pending	Integer	Number of pending jobs
Abnormal	Integer	Number of abnormal jobs
Terminating	Integer	Number of jobs that are being terminated
Creating	Integer	Number of jobs that are being created
Running	Integer	Number of running jobs
Completed	Integer	Number of completed jobs
Terminated	Integer	Number of terminated jobs
Failed	Integer	Number of jobs that fail to be executed

Status code: 400

Table 8-27 Response body parameters

Parameter	Type	Description
error_code	String	Error code Minimum: 8 Maximum: 36
error_msg	String	Error message Minimum: 2 Maximum: 512

Status code: 404

Table 8-28 Response body parameters

Parameter	Type	Description
error_code	String	Error code Minimum: 8 Maximum: 36
error_msg	String	Error message Minimum: 2 Maximum: 512

Example Requests

None

Example Responses

Status code: 400

Bad request

```
{
  "error_code" : "ModelArts.50004000",
  "error_msg" : "Bad request."
}
```

Status code: 404

Not found

```
{
  "error_code" : "ModelArts.50015001",
  "error_msg" : "Pool {name} not found."
}
```

Status Codes

Status Code	Description
200	OK
400	Bad request
404	Not found

Error Codes

See [Error Codes](#).

8.5 Resource Metrics

8.5.1 Obtaining the Real-Time Resource Usage

Function

This API is used to obtain the real-time usage of all resource pools in the current project.

URI

GET /v2/{project_id}/metrics/runtime/pools

Table 8-29 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .

Request Parameters

None

Response Parameters

Status code: 200

Table 8-30 Response body parameters

Parameter	Type	Description
apiVersion	String	Resource version. Options: <ul style="list-style-type: none"> v2
kind	String	Resource type. Options: <ul style="list-style-type: none"> PoolMetricsList
items	Array of MetricsItem objects	Metric list

Table 8-31 MetricsItem

Parameter	Type	Description
table	table object	Resource list
metadata	ResourceMetricsMetadata object	Metadata of resource metrics

Table 8-32 table

Parameter	Type	Description
allocated	Allocated object	Allocated resources
capacity	Capacity object	Total resource capacity

Table 8-33 Allocated

Parameter	Type	Description
value	Value object	Resource amount
timestamp	String	UTC time, in the format of yyyy-MM-dd'T'HH:mm:ss'Z'
window	String	Statistics interval. 1s indicates 1 second, 1m indicates 1 minute, and 1h indicates 1 hour.

Table 8-34 Capacity

Parameter	Type	Description
value	Value object	Resource amount
maxValue	Value object	Maximum number of elastic resources
timestamp	String	UTC time, in the format of yyyy-MM-dd'T'HH:mm:ss'Z'
window	String	Statistics interval. 1s indicates 1 second, 1m indicates 1 minute, and 1h indicates 1 hour.

Table 8-35 Value

Parameter	Type	Description
cpu	String	Number of CPUs
memory	String	Memory
nvidia.com/t4	String	Number of T4 CPUs

Table 8-36 ResourceMetricsMetadata

Parameter	Type	Description
name	String	Name of a resource metric
labels	Object	Labels of a resource metric

Example Requests

None

Example Responses

Status code: 200

OK

```
{
  "apiVersion": "v2",
  "kind": "PoolMetricsList",
  "items": [ {
    "table": {
      "allocated": {
        "value": {
          "cpu": 5,
          "memory": "15548Mi",
          "nvidia.com/t4": "1073m"
        }
      },
      "timestamp": "2022-03-30T07:09:10Z",
      "window": "1m"
    },
    "capacity": {
      "value": {
        "cpu": 16,
        "memory": "64Gi",
        "nvidia.com/t4": 2
      },
      "maxValue": {
        "cpu": 16,
        "memory": "64Gi",
        "nvidia.com/t4": 2
      },
      "timestamp": "2022-03-30T07:09:10Z",
      "window": "1m"
    }
  }
],
  "metadata": {
    "name": "hougang-rse-pool"
  }
}
```



```
}]
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

8.6 Plug-in Template Management

8.6.1 Querying a Plug-in Template

Function

This API is used to obtain details of a specified plug-in template.

URI

GET /v1/{project_id}/plugintemplates/{plugintemplate_name}

Table 8-37 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .
plugintemplate_name	Yes	String	Plug-in template name. Options: <ul style="list-style-type: none"> gpu-driver: GPU driver plug-in template npu-driver: NPU driver plug-in template

Request Parameters

None

Response Parameters

Status code: 200

Table 8-38 Response body parameters

Parameter	Type	Description
apiVersion	String	API version. Options: <ul style="list-style-type: none"> • v1
kind	String	Resource type. Options: <ul style="list-style-type: none"> • PluginTemplate: plug-in template
metadata	PluginTemplateMetadata object	Plug-in template metadata
spec	PluginTemplateSpec object	Plug-in template specifications

Table 8-39 PluginTemplateMetadata

Parameter	Type	Description
name	String	Plug-in template name

Table 8-40 PluginTemplateSpec

Parameter	Type	Description
type	String	Plug-in template type. Options: <ul style="list-style-type: none"> • npuDriver: NPU driver • gpuDriver: GPU driver • ccePlugin: CCE plug-in • helm: Helm template • icAgent: ICAgent
description	String	Plug-in template description
versions	Map<String, PluginTemplateVersion >	Plug-in template versions

Table 8-41 PluginTemplateVersion

Parameter	Type	Description
detail	String	Version description

Status code: 404

Table 8-42 Response body parameters

Parameter	Type	Description
error_code	String	Error code Minimum: 8 Maximum: 36
error_msg	String	Error message Minimum: 2 Maximum: 512

Example Requests

None

Example Responses

Status code: 200

OK

```
{
  "apiVersion": "v1",
  "kind": "PluginTemplate",
  "metadata": {
    "name": "npu-driver"
  },
  "spec": {
    "type": "npuDriver",
    "description": "npu driver"
  },
  "versions": {
    "78-21.0.2": {
      "detail": "c78driver&firmware"
    },
    "77-21.0.cr1": {
      "detail": "c77driver&firmware"
    }
  }
}
```

Status code: 404

Not found

```
{
  "error_code": "ModelArts.50005101",
  "error_msg": "PluginTemplate {name} not found."
}
```

Status Codes

Status Code	Description
200	OK
404	Not found

Error Codes

See [Error Codes](#).

8.7 Tag Management

8.7.1 Creating a Resource Pool Tag

Function

This API is used to add tags to a specified resource pool. Tags can be added in batches. If a tag to be added has the same key as an existing tag, the tag will overwrite the existing one.

URI

POST /v1/{project_id}/pools/{resource_id}/tags/create

Table 8-43 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .
resource_id	Yes	String	Resource ID, which is the name of the resource pool

Request Parameters

Table 8-44 Request body parameters

Parameter	Mandatory	Type	Description
tags	Yes	Array of Tag objects	List of tags to be added

Table 8-45 Tag

Parameter	Mandatory	Type	Description
key	Yes	String	Tag key Minimum: 1 Maximum: 128
value	Yes	String	Tag value Minimum: 0 Maximum: 255

Response Parameters

Status code: 204

Table 8-46 Response body parameters

Parameter	Type	Description
tags	Tag object	Resource tag list

Table 8-47 Tag

Parameter	Type	Description
key	String	Tag key Minimum: 1 Maximum: 128
value	String	Tag value Minimum: 0 Maximum: 255

Status code: 400

Table 8-48 Response body parameters

Parameter	Type	Description
error_code	String	Error codes of ModelArts
error_msg	String	Error message

Status code: 401

Table 8-49 Response body parameters

Parameter	Type	Description
error_code	String	Error codes of ModelArts
error_msg	String	Error message

Status code: 403

Table 8-50 Response body parameters

Parameter	Type	Description
error_code	String	Error codes of ModelArts
error_msg	String	Error message

Status code: 404

Table 8-51 Response body parameters

Parameter	Type	Description
error_code	String	Error codes of ModelArts
error_msg	String	Error message

Example Requests

```
https://{endpoint}/v1/{project_id}/pools/a55eba18-1ebf-4e9a-8229-d2d3b593a3dc/tags/create
{
  "tags": [ {
    "key": "test",
    "value": "service-gpu"
  }, {
    "key": "model_version",
    "value": "0.1"
  } ]
}
```

Example Responses

Status code: 204

Tags added successfully.

```
{
  "tags": [ {
    "key": "test",
    "value": "service-gpu"
  }, {
    "key": "model_version",
```

```
"value": "0.1"
}]
}
```

Status Codes

Status Code	Description
204	Tags added successfully.
400	Invalid parameter.
401	Authentication failed.
403	Insufficient permission.
404	Resource not found.

Error Codes

See [Error Codes](#).

8.7.2 Deleting Tags of a Resource Pool

Function

This API is used to delete tags of a specified resource. Batch deletion is supported.

URI

DELETE /v1/{project_id}/pools/{resource_id}/tags/delete

Table 8-52 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .
resource_id	Yes	String	Resource ID, which is the name of the resource pool

Request Parameters

Table 8-53 Request body parameters

Parameter	Mandatory	Type	Description
tags	Yes	Array of tags objects	Tags to be deleted

Table 8-54 tags

Parameter	Mandatory	Type	Description
key	Yes	String	Tag key Minimum: 1 Maximum: 128
value	No	String	Tag value. This parameter is optional. If this parameter is left blank, the tag that matches the key is deleted. Minimum: 0 Maximum: 255

Response Parameters

Status code: 204

Table 8-55 Response body parameters

Parameter	Type	Description
tags	Tag object	Resource tag list

Table 8-56 Tag

Parameter	Type	Description
key	String	Tag key Minimum: 1 Maximum: 128
value	String	Tag value Minimum: 0 Maximum: 255

Status code: 400

Table 8-57 Response body parameters

Parameter	Type	Description
error_code	String	Error codes of ModelArts
error_msg	String	Error message

Status code: 401

Table 8-58 Response body parameters

Parameter	Type	Description
error_code	String	Error codes of ModelArts
error_msg	String	Error message

Status code: 403

Table 8-59 Response body parameters

Parameter	Type	Description
error_code	String	Error codes of ModelArts
error_msg	String	Error message

Status code: 404

Table 8-60 Response body parameters

Parameter	Type	Description
error_code	String	Error codes of ModelArts
error_msg	String	Error message

Example Requests

```
https://{endpoint}/v1/{project_id}/pools/a55eba18-1ebf-4e9a-8229-d2d3b593a3dc/tags/delete
{
  "tags": [ {
    "key": "dev",
    "value": "dev1"
  } ]
}
```

Example Responses

Status code: 204

Tags deleted successfully.

```
{
  "tags": [ {
    "key": "dev",
    "value": "dev1"
  } ]
}
```

Status Codes

Status Code	Description
204	Tags deleted successfully.
400	Invalid parameter.
401	Authentication failed.
403	Insufficient permission.
404	The resource does not exist.

Error Codes

See [Error Codes](#).

8.7.3 Querying All Tags of a Resource Pool

Function

This API is used to query all tags of a resource pool in the current project. By default, all workspaces are queried. Tag data is not returned for workspaces on which the user does not have permission.

URI

GET /v1/{project_id}/pools/tags

Table 8-61 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .

Table 8-62 Query Parameters

Parameter	Mandatory	Type	Description
limit	No	Integer	Maximum number of records returned on each page. The default value is 200 . Minimum: 1 Default: 200
offset	No	Integer	Start page for pagination display. The default value is 0 . Minimum: 0 Default: 0

Request Parameters

None

Response Parameters

Status code: 200

Table 8-63 Response body parameters

Parameter	Type	Description
tags	Array of CombineTag objects	Data structure of merged tags with the same key Array Length: 0 - 200

Table 8-64 CombineTag

Parameter	Type	Description
key	String	Tag key Minimum: 1 Maximum: 128
values	Array of strings	Merged tag values with the same key Array Length: 0 - 200

Status code: 400

Table 8-65 Response body parameters

Parameter	Type	Description
error_code	String	Error codes of ModelArts
error_msg	String	Error message

Status code: 401

Table 8-66 Response body parameters

Parameter	Type	Description
error_code	String	Error codes of ModelArts
error_msg	String	Error message

Status code: 403

Table 8-67 Response body parameters

Parameter	Type	Description
error_code	String	Error codes of ModelArts
error_msg	String	Error message

Example Requests

`https://{endpoint}/v1/{project_id}/pools/tags`

Example Responses

Status code: 200

All tags of the specified resource type in the project are returned.

```
{
  "tags": [ {
    "key": "model_version",
    "values": [ "0.1", "0.2" ]
  }, {
    "key": "conda_version",
    "values": [ "10.2", "11.0" ]
  } ]
}
```

Status Codes

Status Code	Description
200	All tags of the specified resource type in the project are returned.
400	Invalid parameter.
401	Authentication failed.
403	Insufficient permission.

Error Codes

See [Error Codes](#).

8.8 Network Management

8.8.1 Creating Network Resources

Function

This API is used to create network resources.

URI

POST /v1/{project_id}/networks

Table 8-68 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.

Request Parameters

Table 8-69 Request body parameters

Parameter	Mandatory	Type	Description
apiVersion	Yes	String	API version. Options: <ul style="list-style-type: none"> v1
kind	Yes	String	Resource type. Options: <ul style="list-style-type: none"> Network

Parameter	Mandatory	Type	Description
metadata	Yes	NetworkMetadataCreation object	Metadata of network resources.
spec	Yes	NetworkSpec object	Description of network resources.

Table 8-70 NetworkMetadataCreation

Parameter	Mandatory	Type	Description
labels	Yes	NetworkMetadataLabels object	Metadata labels of network resources.

Table 8-71 NetworkMetadataLabels

Parameter	Mandatory	Type	Description
os.modelarts/ name	Yes	String	Specified network name. Minimum: 4 Maximum: 32

Table 8-72 NetworkSpec

Parameter	Mandatory	Type	Description
ipv6enable	No	Boolean	Whether to enable IPv6. Once IPv6 is enabled, it cannot be disabled.
cidr	Yes	String	Network CIDR. Value range: <ul style="list-style-type: none"> 172.16.0.0/12-172.16.0.0/24 192.168.0.0/16-192.168.0.0/24
connection	No	NetworkConnection object	Automatically interconnected endpoint.

Table 8-73 NetworkConnection

Parameter	Mandatory	Type	Description
peerConnectionList	No	Array of peerConnectionList objects	Peering connections
sfsTurboConnectionList	No	Array of sfsTurboConnectionList objects	SFS Turbo connections through attached NICs

Table 8-74 peerConnectionList

Parameter	Mandatory	Type	Description
peerVpcId	Yes	String	VPC ID of the peer end
peerSubnetId	Yes	String	Subnet ID of the peer end

Table 8-75 sfsTurboConnectionList

Parameter	Mandatory	Type	Description
sfsId	Yes	String	ID of an SFS Turbo instance
name	Yes	String	Name of an SFS Turbo instance

Response Parameters

Status code: 200

Table 8-76 Response body parameters

Parameter	Type	Description
apiVersion	String	API version. Options: <ul style="list-style-type: none"> v1
kind	String	Resource type. Options: <ul style="list-style-type: none"> Network
metadata	NetworkMetadata object	Metadata of network resources.
spec	NetworkSpec object	Description of network resources.

Parameter	Type	Description
status	NetworkStatus object	Status of network resources.

Table 8-77 NetworkMetadata

Parameter	Type	Description
name	String	Automatically generated network name, which is equivalent to networkId .
creationTimestamp	String	Timestamp, for example, 2021-11-01T03:49:41Z.
labels	NetworkMetadataLabels object	Labels of network resources.
annotations	NetworkMetadataAnnotations object	Annotations of network resources.

Table 8-78 NetworkMetadataLabels

Parameter	Type	Description
os.modelarts/name	String	Specified network name. Minimum: 4 Maximum: 32

Table 8-79 NetworkMetadataAnnotations

Parameter	Type	Description
os.modelarts/description	String	Network resource description, which is used to describe a scenario. The following special characters are not allowed: !<>=&" Minimum: 0 Maximum: 100

Table 8-80 NetworkSpec

Parameter	Type	Description
ipv6enable	Boolean	Whether to enable IPv6. Once IPv6 is enabled, it cannot be disabled.
cidr	String	Network CIDR. Value range: <ul style="list-style-type: none"> 172.16.0.0/12-172.16.0.0/24 192.168.0.0/16-192.168.0.0/24
connection	NetworkConnection object	Automatically interconnected endpoint.

Table 8-81 NetworkConnection

Parameter	Type	Description
peerConnectionList	Array of peerConnectionList objects	Peering connections
sfsTurboConnectionList	Array of sfsTurboConnectionList objects	SFS Turbo connections through attached NICs

Table 8-82 peerConnectionList

Parameter	Type	Description
peerVpId	String	VPC ID of the peer end
peerSubnetId	String	Subnet ID of the peer end

Table 8-83 sfsTurboConnectionList

Parameter	Type	Description
sfsId	String	ID of an SFS Turbo instance
name	String	Name of an SFS Turbo instance

Table 8-84 NetworkStatus

Parameter	Type	Description
phase	String	Current network status. Options: <ul style="list-style-type: none"> • Creating: The network is being created. • Active: The network is functional. • Abnormal: The network malfunctions.
connectionStatus	NetworkConnectionStatus object	Network connection status.

Table 8-85 NetworkConnectionStatus

Parameter	Type	Description
peerConnectionStatus	Array of peerConnectionStatus objects	Peering connection status
sfsTurboStatus	Array of sfsTurboStatus objects	Status of SFS Turbo accessible to the network

Table 8-86 peerConnectionStatus

Parameter	Type	Description
peerVpId	String	VPC ID of the peer end
peerSubnetId	String	Subnet ID of the peer end
phase	String	Network connection status. Options: <ul style="list-style-type: none"> • Connecting: The network is being connected. • Active: The network is connected properly. • Abnormal: The network connection is abnormal.

Table 8-87 sfsTurboStatus

Parameter	Type	Description
sfsId	String	SFS Turbo ID
name	String	SFS Turbo name

Parameter	Type	Description
status	String	Status of the connection to SFS Turbo. Options: <ul style="list-style-type: none"> ● Active: The SFS connection is normal. ● Abnormal: The SFS connection is abnormal. ● Creating: The SFS connection is being set up. ● Deleting: The SFS connection is being deleted.
ipAddr	String	SFS Turbo access address
connectionType	String	Connection type. Options: <ul style="list-style-type: none"> ● VpcPort: passthrough through attached NICs ● Peering: connection through VPC peering

Status code: 400

Table 8-88 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Example Requests

Create a network.

```
POST {endpoint}/v2/{project_id}/pools
{
  "apiVersion": "v1",
  "kind": "Network",
  "metadata": {
    "labels": {
      "os.modelarts/name": "network-7a03",
      "os.modelarts/workspace.id": "0"
    }
  },
  "spec": {
    "cidr": "192.168.128.0/17"
  }
}
```

Example Responses

Status code: 200

OK

```
{
  "kind": "Network",
  "apiVersion": "v1",
  "metadata": {
    "name": "network-7a03-86c13962597848eeb29c5861153a391f",
    "creationTimestamp": "2022-09-16T09:44:59Z",
    "labels": {
      "os.modelarts/name": "network-7a03",
      "os.modelarts/workspace.id": "0"
    },
    "annotations": { }
  },
  "spec": {
    "cidr": "192.168.128.0/17",
    "connection": { }
  },
  "status": {
    "phase": ""
  }
}
```

Status code: 400

Bad request.

```
{
  "error_code": "ModelArts.50004000",
  "error_msg": "Bad request."
}
```

Status Codes

Status Code	Description
200	OK
400	Bad request.

Error Codes

See [Error Codes](#).

8.8.2 Obtaining Network Resources

Function

This API is used to obtain network resources.

URI

GET /v1/{project_id}/networks

Table 8-89 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.

Table 8-90 Query Parameters

Parameter	Mandatory	Type	Description
labelSelector	No	String	Filter by label.
limit	No	Integer	Number of records returned for a single pagination query.
continue	No	String	Previous query location in pagination query.

Request Parameters

None

Response Parameters

Status code: 200

Table 8-91 Response body parameters

Parameter	Type	Description
apiVersion	String	API version. Options: <ul style="list-style-type: none"> • v1
kind	String	Resource type. Options: <ul style="list-style-type: none"> • NetworkList: networks
metadata	metadata object	Metadata of network resources.
items	Array of Network objects	Network resources.

Table 8-92 metadata

Parameter	Type	Description
continue	String	Next query location in pagination query.
remainingItemCount	Integer	Remaining resources.

Table 8-93 Network

Parameter	Type	Description
apiVersion	String	API version. Options: <ul style="list-style-type: none"> • v1
kind	String	Resource type. Options: <ul style="list-style-type: none"> • Network
metadata	NetworkMetadata object	Metadata of network resources.
spec	NetworkSpec object	Description of network resources.
status	NetworkStatus object	Status of network resources.

Table 8-94 NetworkMetadata

Parameter	Type	Description
name	String	Automatically generated network name, which is equivalent to networkId .
creationTimestamp	String	Timestamp, for example, 2021-11-01T03:49:41Z.
labels	NetworkMetadataLabels object	Labels of network resources.
annotations	NetworkMetadataAnnotations object	Annotations of network resources.

Table 8-95 NetworkMetadataLabels

Parameter	Type	Description
os.modelarts/name	String	Specified network name. Minimum: 4 Maximum: 32

Table 8-96 NetworkMetadataAnnotations

Parameter	Type	Description
os.modelarts/ description	String	Network resource description, which is used to describe a scenario. The following special characters are not allowed: !<>=&" Minimum: 0 Maximum: 100

Table 8-97 NetworkSpec

Parameter	Type	Description
ipv6enable	Boolean	Whether to enable IPv6. Once IPv6 is enabled, it cannot be disabled.
cidr	String	Network CIDR. Value range: <ul style="list-style-type: none"> 172.16.0.0/12-172.16.0.0/24 192.168.0.0/16-192.168.0.0/24
connection	NetworkConnection object	Automatically interconnected endpoint.

Table 8-98 NetworkConnection

Parameter	Type	Description
peerConnectionList	Array of peerConnectionList objects	Peering connections
sfsTurboConnectionList	Array of sfsTurboConnectionList objects	SFS Turbo connections through attached NICs

Table 8-99 peerConnectionList

Parameter	Type	Description
peerVpId	String	VPC ID of the peer end
peerSubnetId	String	Subnet ID of the peer end

Table 8-100 sfsTurboConnectionList

Parameter	Type	Description
sfsId	String	ID of an SFS Turbo instance
name	String	Name of an SFS Turbo instance

Table 8-101 NetworkStatus

Parameter	Type	Description
phase	String	Current network status. Options: <ul style="list-style-type: none"> • Creating: The network is being created. • Active: The network is functional. • Abnormal: The network malfunctions.
connectionStatus	NetworkConnectionStatus object	Network connection status.

Table 8-102 NetworkConnectionStatus

Parameter	Type	Description
peerConnectionStatus	Array of peerConnectionStatus objects	Peering connection status
sfsTurboStatus	Array of sfsTurboStatus objects	Status of SFS Turbo accessible to the network

Table 8-103 peerConnectionStatus

Parameter	Type	Description
peerVpcId	String	VPC ID of the peer end
peerSubnetId	String	Subnet ID of the peer end
phase	String	Network connection status. Options: <ul style="list-style-type: none"> • Connecting: The network is being connected. • Active: The network is connected properly. • Abnormal: The network connection is abnormal.

Table 8-104 sfsTurboStatus

Parameter	Type	Description
sfsId	String	SFS Turbo ID
name	String	SFS Turbo name
status	String	Status of the connection to SFS Turbo. Options: <ul style="list-style-type: none"> ● Active: The SFS connection is normal. ● Abnormal: The SFS connection is abnormal. ● Creating: The SFS connection is being set up. ● Deleting: The SFS connection is being deleted.
ipAddr	String	SFS Turbo access address
connectionType	String	Connection type. Options: <ul style="list-style-type: none"> ● VpcPort: passthrough through attached NICs ● Peering: connection through VPC peering

Example Requests

Obtain resource pools.

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

```
{ }
```

Example Responses

Status code: 200

OK

```
{
  "kind": "NetworkList",
  "apiVersion": "v1",
  "metadata": { },
  "items": [ {
    "kind": "Network",
    "apiVersion": "v1",
    "metadata": {
      "name": "network-7a03-86c13962597848eeb29c5861153a391f",
      "creationTimestamp": "2022-09-16T09:44:59Z",
      "labels": {
        "os.modelarts/name": "network-7a03",
        "os.modelarts/workspace.id": "0"
      },
      "annotations": { }
    },
    "spec": {
      "cidr": "192.168.128.0/17",
      "connection": { }
    },
    "status": {
```

```
"phase" : "Active",
"connectionStatus" : { }
}
}]
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

8.8.3 Obtaining a Network Resource

Function

This API is used to obtain details about a specified network resource.

URI

GET /v1/{project_id}/networks/{network_name}

Table 8-105 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
network_name	Yes	String	Automatically generated network name.

Request Parameters

None

Response Parameters

Status code: 200

Table 8-106 Response body parameters

Parameter	Type	Description
apiVersion	String	API version. Options: <ul style="list-style-type: none"> v1

Parameter	Type	Description
kind	String	Resource type. Options: <ul style="list-style-type: none"> • Network
metadata	NetworkMetadata object	Metadata of network resources.
spec	NetworkSpec object	Description of network resources.
status	NetworkStatus object	Status of network resources.

Table 8-107 NetworkMetadata

Parameter	Type	Description
name	String	Automatically generated network name, which is equivalent to networkId .
creationTimestamp	String	Timestamp, for example, 2021-11-01T03:49:41Z.
labels	NetworkMetadataLabels object	Labels of network resources.
annotations	NetworkMetadataAnnotations object	Annotations of network resources.

Table 8-108 NetworkMetadataLabels

Parameter	Type	Description
os.modelarts/name	String	Specified network name. Minimum: 4 Maximum: 32

Table 8-109 NetworkMetadataAnnotations

Parameter	Type	Description
os.modelarts/ description	String	Network resource description, which is used to describe a scenario. The following special characters are not allowed: !<>=&" Minimum: 0 Maximum: 100

Table 8-110 NetworkSpec

Parameter	Type	Description
ipv6enable	Boolean	Whether to enable IPv6. Once IPv6 is enabled, it cannot be disabled.
cidr	String	Network CIDR. Value range: <ul style="list-style-type: none"> • 172.16.0.0/12-172.16.0.0/24 • 192.168.0.0/16-192.168.0.0/24
connection	NetworkConnection object	Automatically interconnected endpoint.

Table 8-111 NetworkConnection

Parameter	Type	Description
peerConnectionList	Array of peerConnectionList objects	Peering connections
sfsTurboConnectionList	Array of sfsTurboConnectionList objects	SFS Turbo connections through attached NICs

Table 8-112 peerConnectionList

Parameter	Type	Description
peerVpId	String	VPC ID of the peer end
peerSubnetId	String	Subnet ID of the peer end

Table 8-113 sfsTurboConnectionList

Parameter	Type	Description
sfsId	String	ID of an SFS Turbo instance
name	String	Name of an SFS Turbo instance

Table 8-114 NetworkStatus

Parameter	Type	Description
phase	String	Current network status. Options: <ul style="list-style-type: none"> • Creating: The network is being created. • Active: The network is functional. • Abnormal: The network malfunctions.
connectionStatus	NetworkConnectionStatus object	Network connection status.

Table 8-115 NetworkConnectionStatus

Parameter	Type	Description
peerConnectionStatus	Array of peerConnectionStatus objects	Peering connection status
sfsTurboStatus	Array of sfsTurboStatus objects	Status of SFS Turbo accessible to the network

Table 8-116 peerConnectionStatus

Parameter	Type	Description
peerVpcId	String	VPC ID of the peer end
peerSubnetId	String	Subnet ID of the peer end
phase	String	Network connection status. Options: <ul style="list-style-type: none"> • Connecting: The network is being connected. • Active: The network is connected properly. • Abnormal: The network connection is abnormal.

Table 8-117 sfsTurboStatus

Parameter	Type	Description
sfsId	String	SFS Turbo ID
name	String	SFS Turbo name
status	String	Status of the connection to SFS Turbo. Options: <ul style="list-style-type: none"> ● Active: The SFS connection is normal. ● Abnormal: The SFS connection is abnormal. ● Creating: The SFS connection is being set up. ● Deleting: The SFS connection is being deleted.
ipAddr	String	SFS Turbo access address
connectionType	String	Connection type. Options: <ul style="list-style-type: none"> ● VpcPort: passthrough through attached NICs ● Peering: connection through VPC peering

Status code: 404

Table 8-118 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Example Requests

Obtain details about a network.

```
GET https://{endpoint}/v1/{project_id}/networks/{network_name}
{ }
```

Example Responses

Status code: 200

OK

```
{
  "kind": "Network",
  "apiVersion": "v1",
  "metadata": {
    "name": "network-7a03-86c13962597848eeb29c5861153a391f",
    "creationTimestamp": "2022-09-16T09:44:59Z",
```

```

"labels" : {
  "os.modelarts/name" : "network-7a03",
  "os.modelarts/workspace.id" : "0"
},
"annotations" : { }
},
"spec" : {
  "cidr" : "192.168.128.0/17",
  "connection" : { }
},
"status" : {
  "phase" : "Active",
  "connectionStatus" : { }
}
}

```

Status code: 404

Not found.

```

{
  "error_code" : "ModelArts.50025001",
  "error_msg" : "Network not exist."
}

```

Status Codes

Status Code	Description
200	OK
404	Not found.

Error Codes

See [Error Codes](#).

8.8.4 Deleting a Network Resource

Function

This API is used to delete a specified network resource.

URI

DELETE /v1/{project_id}/networks/{network_name}

Table 8-119 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
network_name	Yes	String	Network resource name.

Request Parameters

None

Response Parameters

Status code: 200

Table 8-120 Response body parameters

Parameter	Type	Description
apiVersion	String	API version. Options: <ul style="list-style-type: none"> • v1
kind	String	Resource type. Options: <ul style="list-style-type: none"> • Network
metadata	NetworkMetadata object	Metadata of network resources.
spec	NetworkSpec object	Description of network resources.
status	NetworkStatus object	Status of network resources.

Table 8-121 NetworkMetadata

Parameter	Type	Description
name	String	Automatically generated network name, which is equivalent to networkId .
creationTimestamp	String	Timestamp, for example, 2021-11-01T03:49:41Z.
labels	NetworkMetadataLabels object	Labels of network resources.
annotations	NetworkMetadataAnnotations object	Annotations of network resources.

Table 8-122 NetworkMetadataLabels

Parameter	Type	Description
os.modelarts/ name	String	Specified network name. Minimum: 4 Maximum: 32

Table 8-123 NetworkMetadataAnnotations

Parameter	Type	Description
os.modelarts/ description	String	Network resource description, which is used to describe a scenario. The following special characters are not allowed: !<>=&"" Minimum: 0 Maximum: 100

Table 8-124 NetworkSpec

Parameter	Type	Description
ipv6enable	Boolean	Whether to enable IPv6. Once IPv6 is enabled, it cannot be disabled.
cidr	String	Network CIDR. Value range: <ul style="list-style-type: none"> 172.16.0.0/12-172.16.0.0/24 192.168.0.0/16-192.168.0.0/24
connection	NetworkConnection object	Automatically interconnected endpoint.

Table 8-125 NetworkConnection

Parameter	Type	Description
peerConnectionList	Array of peerConnectionList objects	Peering connections
sfsTurboConnectionList	Array of sfsTurboConnectionList objects	SFS Turbo connections through attached NICs

Table 8-126 peerConnectionList

Parameter	Type	Description
peerVpclId	String	VPC ID of the peer end
peerSubnetId	String	Subnet ID of the peer end

Table 8-127 sfsTurboConnectionList

Parameter	Type	Description
sfsId	String	ID of an SFS Turbo instance
name	String	Name of an SFS Turbo instance

Table 8-128 NetworkStatus

Parameter	Type	Description
phase	String	Current network status. Options: <ul style="list-style-type: none"> • Creating: The network is being created. • Active: The network is functional. • Abnormal: The network malfunctions.
connectionStatus	NetworkConnectionStatus object	Network connection status.

Table 8-129 NetworkConnectionStatus

Parameter	Type	Description
peerConnectionStatus	Array of peerConnectionStatus objects	Peering connection status
sfsTurboStatus	Array of sfsTurboStatus objects	Status of SFS Turbo accessible to the network

Table 8-130 peerConnectionStatus

Parameter	Type	Description
peerVpclId	String	VPC ID of the peer end

Parameter	Type	Description
peerSubnetId	String	Subnet ID of the peer end
phase	String	Network connection status. Options: <ul style="list-style-type: none"> ● Connecting: The network is being connected. ● Active: The network is connected properly. ● Abnormal: The network connection is abnormal.

Table 8-131 sfsTurboStatus

Parameter	Type	Description
sfsId	String	SFS Turbo ID
name	String	SFS Turbo name
status	String	Status of the connection to SFS Turbo. Options: <ul style="list-style-type: none"> ● Active: The SFS connection is normal. ● Abnormal: The SFS connection is abnormal. ● Creating: The SFS connection is being set up. ● Deleting: The SFS connection is being deleted.
ipAddr	String	SFS Turbo access address
connectionType	String	Connection type. Options: <ul style="list-style-type: none"> ● VpcPort: passthrough through attached NICs ● Peering: connection through VPC peering

Status code: 404

Table 8-132 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Example Requests

Delete a network.

```
DELETE https://{endpoint}/v1/{project_id}/networks/{network_name}
{ }
```

Example Responses

Status code: 200

OK

```
{
  "kind": "Network",
  "apiVersion": "v1",
  "metadata": {
    "name": "network-7a03-86c13962597848eeb29c5861153a391f",
    "creationTimestamp": "2022-09-16T09:44:59Z",
    "deletionTimestamp": "2022-09-16T10:06:27Z",
    "labels": {
      "os.modelarts/name": "network-7a03",
      "os.modelarts/workspace.id": "0"
    },
    "annotations": { }
  },
  "spec": {
    "cidr": "192.168.128.0/17",
    "connection": {
      "peerConnectionList": [ {
        "peerVpclid": "03e4f4d7-fc62-409b-9c52-df885525e30b",
        "peerSubnetid": "42aeebc3-f7c7-45aa-b884-e6e9ac2f841d"
      } ]
    }
  },
  "status": {
    "phase": "Active",
    "connectionStatus": {
      "peerConnectionStatus": [ {
        "peerVpclid": "03e4f4d7-fc62-409b-9c52-df885525e30b",
        "peerSubnetid": "42aeebc3-f7c7-45aa-b884-e6e9ac2f841d",
        "phase": "Active"
      } ]
    }
  }
}
```

Status code: 404

Not found.

```
{
  "error_code": "ModelArts.50025001",
  "error_msg": "Network not exist."
}
```

Status Codes

Status Code	Description
200	OK
404	Not found.

Error Codes

See [Error Codes](#).

8.8.5 Updating a Network Resource

Function

This API is used to update a specified network resource.

URI

PATCH /v1/{project_id}/networks/{network_name}

Table 8-133 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
network_name	Yes	String	Network resource name.

Request Parameters

Table 8-134 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Application/merge-patch + JSON.

Table 8-135 Request body parameters

Parameter	Mandatory	Type	Description
metadata	No	NetworkMetadataUpdate object	Updated network resource metadata.
spec	No	NetworkSpecUpdate object	Updated network resource description.

Table 8-136 NetworkMetadataUpdate

Parameter	Mandatory	Type	Description
annotations	No	NetworkMetadataAnnotations object	Resource annotations.
labels	No	NetworkUpdateLabels object	Labels of network resources

Table 8-137 NetworkMetadataAnnotations

Parameter	Mandatory	Type	Description
os.modelarts/ description	No	String	Network resource description, which is used to describe a scenario. The following special characters are not allowed: ! <>=&" Minimum: 0 Maximum: 100

Table 8-138 NetworkUpdateLabels

Parameter	Mandatory	Type	Description
os.modelarts/ workspace.id	No	String	ID of the workspace to which network belongs

Table 8-139 NetworkSpecUpdate

Parameter	Mandatory	Type	Description
ipv6enable	No	Boolean	Whether to enable IPv6. Once IPv6 is enabled, it cannot be disabled.
connection	No	NetworkConnection object	Updated network connection.

Table 8-140 NetworkConnection

Parameter	Mandatory	Type	Description
peerConnectionList	No	Array of peerConnectionList objects	Peering connections
sfsTurboConnectionList	No	Array of sfsTurboConnectionList objects	SFS Turbo connections through attached NICs

Table 8-141 peerConnectionList

Parameter	Mandatory	Type	Description
peerVpcId	Yes	String	VPC ID of the peer end
peerSubnetId	Yes	String	Subnet ID of the peer end

Table 8-142 sfsTurboConnectionList

Parameter	Mandatory	Type	Description
sfsId	Yes	String	ID of an SFS Turbo instance
name	Yes	String	Name of an SFS Turbo instance

Response Parameters

Status code: 200

Table 8-143 Response body parameters

Parameter	Type	Description
apiVersion	String	API version. Options: <ul style="list-style-type: none"> v1
kind	String	Resource type. Options: <ul style="list-style-type: none"> Network
metadata	NetworkMetadata object	Metadata of network resources.
spec	NetworkSpec object	Description of network resources.

Parameter	Type	Description
status	NetworkStatus object	Status of network resources.

Table 8-144 NetworkMetadata

Parameter	Type	Description
name	String	Automatically generated network name, which is equivalent to networkId .
creationTimestamp	String	Timestamp, for example, 2021-11-01T03:49:41Z.
labels	NetworkMetadataLabels object	Labels of network resources.
annotations	NetworkMetadataAnnotations object	Annotations of network resources.

Table 8-145 NetworkMetadataLabels

Parameter	Type	Description
os.modelarts/name	String	Specified network name. Minimum: 4 Maximum: 32

Table 8-146 NetworkMetadataAnnotations

Parameter	Type	Description
os.modelarts/description	String	Network resource description, which is used to describe a scenario. The following special characters are not allowed: !<>=&" Minimum: 0 Maximum: 100

Table 8-147 NetworkSpec

Parameter	Type	Description
ipv6enable	Boolean	Whether to enable IPv6. Once IPv6 is enabled, it cannot be disabled.
cidr	String	Network CIDR. Value range: <ul style="list-style-type: none"> 172.16.0.0/12-172.16.0.0/24 192.168.0.0/16-192.168.0.0/24
connection	NetworkConnection object	Automatically interconnected endpoint.

Table 8-148 NetworkConnection

Parameter	Type	Description
peerConnectionList	Array of peerConnectionList objects	Peering connections
sfsTurboConnectionList	Array of sfsTurboConnectionList objects	SFS Turbo connections through attached NICs

Table 8-149 peerConnectionList

Parameter	Type	Description
peerVpcId	String	VPC ID of the peer end
peerSubnetId	String	Subnet ID of the peer end

Table 8-150 sfsTurboConnectionList

Parameter	Type	Description
sfsId	String	ID of an SFS Turbo instance
name	String	Name of an SFS Turbo instance

Table 8-151 NetworkStatus

Parameter	Type	Description
phase	String	Current network status. Options: <ul style="list-style-type: none"> • Creating: The network is being created. • Active: The network is functional. • Abnormal: The network malfunctions.
connectionStatus	NetworkConnectionStatus object	Network connection status.

Table 8-152 NetworkConnectionStatus

Parameter	Type	Description
peerConnectionStatus	Array of peerConnectionStatus objects	Peering connection status
sfsTurboStatus	Array of sfsTurboStatus objects	Status of SFS Turbo accessible to the network

Table 8-153 peerConnectionStatus

Parameter	Type	Description
peerVpcId	String	VPC ID of the peer end
peerSubnetId	String	Subnet ID of the peer end
phase	String	Network connection status. Options: <ul style="list-style-type: none"> • Connecting: The network is being connected. • Active: The network is connected properly. • Abnormal: The network connection is abnormal.

Table 8-154 sfsTurboStatus

Parameter	Type	Description
sfsId	String	SFS Turbo ID
name	String	SFS Turbo name

Parameter	Type	Description
status	String	Status of the connection to SFS Turbo. Options: <ul style="list-style-type: none"> • Active: The SFS connection is normal. • Abnormal: The SFS connection is abnormal. • Creating: The SFS connection is being set up. • Deleting: The SFS connection is being deleted.
ipAddr	String	SFS Turbo access address
connectionType	String	Connection type. Options: <ul style="list-style-type: none"> • VpcPort: passthrough through attached NICs • Peering: connection through VPC peering

Status code: 400

Table 8-155 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Status code: 404

Table 8-156 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Example Requests

Interconnect with a VPC.

PATCH https://{endpoint}/v1/{project_id}/networks/{network_name}

```
{
  "spec": {
    "connection": {
      "peerConnectionList": [ {
        "peerVpclid": "03e4f4d7-fc62-409b-9c52-df885525e30b",
        "peerSubnetId": "42aebc3-f7c7-45aa-b884-e6e9ac2f841d"
      }
    ]
  }
}
```

```
    }],  
    "sfsTurboConnectionList" : [ {  
      "sfsId" : "97beb2bb-1a5b-41dd-b7fb-65a9c7954517",  
      "name" : "mulVpc-02"  
    }]  
  }  
}
```

Example Responses

Status code: 200

OK

```
{  
  "kind" : "Network",  
  "apiVersion" : "v1",  
  "metadata" : {  
    "name" : "network-7a03-86c13962597848eeb29c5861153a391f",  
    "creationTimestamp" : "2022-09-16T09:44:59Z",  
    "labels" : {  
      "os.modelarts/name" : "network-7a03",  
      "os.modelarts/workspace.id" : "0"  
    },  
    "annotations" : { }  
  },  
  "spec" : {  
    "cidr" : "192.168.128.0/17",  
    "connection" : {  
      "peerConnectionList" : [ {  
        "peerVpId" : "03e4f4d7-fc62-409b-9c52-df885525e30b",  
        "peerSubnetId" : "42aeebc3-f7c7-45aa-b884-e6e9ac2f841d"  
      } ],  
      "sfsTurboConnectionList" : [ {  
        "sfsId" : "97beb2bb-1a5b-41dd-b7fb-65a9c7954517",  
        "name" : "mulVpc-02"  
      } ]  
    }  
  },  
  "status" : {  
    "phase" : "Active",  
    "connectionStatus" : { }  
  }  
}
```

Status code: 400

Bad request

```
{  
  "error_code" : "ModelArts.50004000",  
  "error_msg" : "Bad request."  
}
```

Status code: 404

Not found.

```
{  
  "error_code" : "ModelArts.50025001",  
  "error_msg" : "Network not exist."  
}
```

Status Codes

Status Code	Description
200	OK
400	Bad request
404	Not found.

Error Codes

See [Error Codes](#).

8.9 Node Management

8.9.1 Obtaining Nodes

Function

This API is used to obtain nodes in a resource pool.

URI

GET /v2/{project_id}/pools/{pool_name}/nodes

Table 8-157 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
pool_name	Yes	String	Name of a resource pool.

Table 8-158 Query Parameters

Parameter	Mandatory	Type	Description
continue	No	String	Previous query location in pagination query.
limit	No	Integer	Number of records returned for a single pagination query.

Request Parameters

None

Response Parameters

Status code: 200

Table 8-159 Response body parameters

Parameter	Type	Description
apiVersion	String	API version. Options: <ul style="list-style-type: none"> • v2
kind	String	Resource type. Options: <ul style="list-style-type: none"> • NodeList: nodes
metadata	NodeListMetadata object	Metadata of resources.
items	Array of Node objects	Nodes.

Table 8-160 NodeListMetadata

Parameter	Type	Description
continue	String	Next query position in pagination query
remainingItemCount	Long	Remaining resources

Table 8-161 Node

Parameter	Type	Description
apiVersion	String	API version. Options: <ul style="list-style-type: none"> • v2
kind	String	Resource type. Options: <ul style="list-style-type: none"> • Node: node
metadata	metadata object	Basic node information.
spec	NodeSpec object	Node description.
status	NodeStatus object	Node status.

Table 8-162 metadata

Parameter	Type	Description
name	String	Node name.
creationTimes tamp	String	Creation time.

Table 8-163 NodeSpec

Parameter	Type	Description
flavor	String	Node specifications

Table 8-164 NodeStatus

Parameter	Type	Description
phase	String	Node status. Options: <ul style="list-style-type: none"> • Available: The node is available. • Creating: The node is being created. • Deleting: The node is being deleted. • Abnormal: The node is not running properly.
az	String	AZ to which the node belongs
privatelp	String	IP address of a node
resources	NodeResource object	Node resources
availableResources	NodeResource object	Available node resources

Table 8-165 NodeResource

Parameter	Type	Description
cpu	String	CPUs.
memory	String	Memory.
nvidia.com/gpu	String	GPUs.
huawei.com/ascend-310	String	Ascend 310.

Parameter	Type	Description
huawei.com/ascend-1980	String	Ascend snt9.

Status code: 404

Table 8-166 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Example Requests

Obtain nodes in a resource pool.

```
GET https://{endpoint}/v2/{project_id}/pools/{pool_name}/nodes
{ }
```

Example Responses

Status code: 200

OK

```
{
  "kind": "NodeList",
  "apiVersion": "v2",
  "metadata": { },
  "items": [ {
    "kind": "Node",
    "apiVersion": "v2",
    "metadata": {
      "name": "os-node-created-zlncn",
      "creationTimestamp": "2022-09-16T05:32:44Z"
    },
    "spec": {
      "flavor": "modelarts.vm.cpu.4ud"
    },
    "status": {
      "phase": "Available",
      "az": "xx-xxx-xx",
      "privateIp": "192.168.0.1",
      "resources": {
        "cpu": "3920m",
        "memory": "6270Mi"
      },
      "availableResources": {
```



```

      "cpu" : "2970m",
      "memory" : "4558Mi"
    }
  }, {
    "kind" : "Node",
    "apiVersion" : "v2",
    "metadata" : {
      "name" : "os-node-created-4s522",
      "creationTimestamp" : "2022-09-16T03:20:53Z"
    },
    "spec" : {
      "flavor" : "modelarts.vm.cpu.4ud"
    },
    "status" : {
      "phase" : "Available",
      "az" : "xx-xxxx-xx",
      "privateIp" : "192.168.0.2",
      "resources" : {
        "cpu" : "3920m",
        "memory" : "6270Mi"
      },
      "availableResources" : {
        "cpu" : "2970m",
        "memory" : "4558Mi"
      }
    }
  }, {
    "kind" : "Node",
    "apiVersion" : "v2",
    "metadata" : {
      "name" : "os-node-created-v7hfj",
      "creationTimestamp" : "2022-09-16T09:16:37Z"
    },
    "spec" : {
      "flavor" : "modelarts.vm.cpu.4ud"
    },
    "status" : {
      "phase" : "Available",
      "az" : "xx-xxxx-xx",
      "privateIp" : "192.168.0.3",
      "resources" : {
        "cpu" : "3920m",
        "memory" : "6270Mi"
      },
      "availableResources" : {
        "cpu" : "3720m",
        "memory" : "5670Mi"
      }
    }
  }
}

```

Status code: 404

Not found.

```

{
  "error_code" : "ModelArts.50015001",
  "error_msg" : "pool not found"
}

```

Status Codes

Status Code	Description
200	OK
404	Not found.

Error Codes

See [Error Codes](#).

8.9.2 Deleting Nodes in Batches

Function

This API is used to delete nodes from a specified resource pool in batches. At least one node must be reserved in the resource pool.

URI

POST /v2/{project_id}/pools/{pool_name}/nodes/batch-delete

Table 8-167 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .
pool_name	Yes	String	Name of a resource pool

Request Parameters

Table 8-168 Request body parameters

Parameter	Mandatory	Type	Description
deleteNodeNames	Yes	Array of strings	Names of the nodes to be deleted

Response Parameters

Status code: 404

Table 8-169 Response body parameters

Parameter	Type	Description
error_code	String	Error code. Minimum: 8 Maximum: 36
error_msg	String	Error message. Minimum: 2 Maximum: 512

Example Requests

```
POST /v2/{project_id}/pools/{pool_name}/nodes/batch-delete
{
  "deleteNodeNames" : [ "os-node-created-mnmc" ]
}
```

Example Responses

Status code: 404

Not found

```
{
  "error_code" : "ModelArts.50015001",
  "error_msg" : "pool not found"
}
```

Status Codes

Status Code	Description
200	OK
404	Not found

Error Codes

See [Error Codes](#).

8.10 Resource Pool Management

8.10.1 Creating Resource Pools

Function

This API is used to create resource pools.

URI

POST /v2/{project_id}/pools

Table 8-170 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.

Request Parameters

Table 8-171 Request header parameters

Parameter	Mandatory	Type	Description
request-type	No	String	Request type.

Table 8-172 Request body parameters

Parameter	Mandatory	Type	Description
apiVersion	Yes	String	API version. Options: <ul style="list-style-type: none"> • v2
kind	Yes	String	Resource type. Options: <ul style="list-style-type: none"> • Pool: resource pool
metadata	Yes	PoolMetadataCreation object	Metadata of a resource pool.
spec	No	PoolSpecCreation object	Description of a resource pool.

Table 8-173 PoolMetadataCreation

Parameter	Mandatory	Type	Description
labels	Yes	PoolLabelsCreation object	Resource pool labels.
annotations	No	PoolAnnotationsCreation object	Resource pool annotations.

Table 8-174 PoolLabelsCreation

Parameter	Mandatory	Type	Description
os.modelarts/ workspace.id	No	String	Workspace ID. The default value is 0 . Default: 0
os.modelarts/ name	Yes	String	Specified resource pool name Minimum: 4 Maximum: 64
os.modelarts/ sharing	No	String	Whether the resource pool can be shared. The value can be unshared or shared .
os.modelarts/ node.prefix	No	String	Customized node prefix. This parameter is optional.

Table 8-175 PoolAnnotationsCreation

Parameter	Mandatory	Type	Description
os.modelarts/ description	No	String	Description of a resource pool
os.modelarts/ billing.mode	No	String	Billing mode. Options: <ul style="list-style-type: none"> • 0: pay-per-use
os.modelarts/ period.num	No	String	Subscription period on a yearly/monthly basis, for example, 2 . This parameter is mandatory when Billing Mode is set to Yearly/ Monthly .
os.modelarts/ period.type	No	String	Yearly/Monthly subscription type. Options: <ul style="list-style-type: none"> • 2: month • 3: year • 4: hour This parameter is mandatory when Billing Mode is set to Yearly/Monthly .

Parameter	Mandatory	Type	Description
os.modelarts/ auto.renew	No	String	Whether to automatically renew the subscription. Options: <ul style="list-style-type: none"> • 0: auto-renewal disabled (default value) • 1: auto-renewal enabled
os.modelarts/ promotion.info	No	String	Discount selected on the operations platform
os.modelarts/ service.console.url	No	String	URL of the page displayed after the subscription order is paid
os.modelarts/ order.id	No	String	Order ID, which is mandatory for creating a yearly/monthly resource pool or changing the billing mode
os.modelarts/ scheduler.extension	No	String	Priority that can be set for the jobs delivered by other users to the shared resource pool

Table 8-176 PoolSpecCreation

Parameter	Mandatory	Type	Description
type	Yes	object	Resource pool type. <ul style="list-style-type: none"> • Dedicate: physical resource pool, which uses a separate network and supports network connection, custom drivers, and custom job types. • Logical: logical resource pool, which can be used after the administrator's approval.
scope	No	Array of strings	Job type enabled, which cannot be specified for logical pools. Options: <ul style="list-style-type: none"> • Train: training jobs • Infer: inference jobs [-Notebook: notebook jobs] (tags: hc)

Parameter	Mandatory	Type	Description
resources	Yes	Array of PoolResource FlavorCount objects	Resource specifications in a resource pool, including resource specifications and the number of resources for each specification
parent	No	String	Parent resource pool ID. This parameter is mandatory when a logical sub-pool is created based on a specified resource pool.
network	No	PoolNetwork object	Network settings for a resource pool. This parameter is mandatory for creating physical resource pools and is unavailable for creating logical resource pools.
containerNetwork	No	containerNetwork object	Kubernetes container network segment. Only one container network segment is supported. This parameter is optional. If this parameter is not set, the default value is specified by CCE.
kubernetesSvcIpRange	No	String	CIDR of the Kubernetes service network segment
masters	No	Array of PoolClusterMaster objects	Master node parameters in a resource pool. This parameter is optional for physical resource pools and is unavailable for logical resource pools.
driver	No	PoolDriver object	Resource pool driver
userLogin	No	PoolUserLogin object	Node login information of a privilege pool
clusters	No	Array of PoolClusterInfo objects	Privilege pool cluster information
ipv6enable	No	Boolean	Whether to enable IPv6

Table 8-177 PoolResourceFlavorCount

Parameter	Mandatory	Type	Description
flavor	Yes	String	Resource specifications name, for example, modelarts.vm.gpu.t4u8
count	Yes	Integer	Minimum count for the specifications in a resource pool
maxCount	Yes	Integer	Elastic usage of the resource specifications. This parameter value is the same the count value in a physical pool; It is greater than or equal to the count value in a logical pool.
azs	No	Array of PoolNodeAz objects	AZ where resource pool nodes are deployed.

Table 8-178 PoolNodeAz

Parameter	Mandatory	Type	Description
az	Yes	String	AZ name.
count	Yes	Integer	Number of nodes for expanding the capacity of a specified AZ. Minimum: 1 Maximum: 2000

Table 8-179 PoolNetwork

Parameter	Mandatory	Type	Description
name	No	String	Network name. When you create a network with a specified name, the system will automatically create subnets for you. By default, the first subnet will be used. Minimum: 4 Maximum: 128

Parameter	Mandatory	Type	Description
vpclId	No	String	VPC ID, which must be specified when a privileged pool is created and is unavailable for a non-privileged pool
subnetId	No	String	Subnet ID, which must be specified when a privileged pool is created and is unavailable for a non-privileged pool

Table 8-180 containerNetwork

Parameter	Mandatory	Type	Description
cidr	No	String	CIDR block of the container network segment. Recommended: 10.0.0.0/12-19, 172.16.0.0/16-19, or 192.168.0.0/16-19. If the selected CIDR block conflicts with existing ones, an error will be reported. This parameter cannot be modified after the cluster is created.

Table 8-181 PoolClusterMaster

Parameter	Mandatory	Type	Description
az	No	String	AZ where the master node is located

Table 8-182 PoolDriver

Parameter	Mandatory	Type	Description
gpuVersion	No	String	GPU driver version. This parameter is available when GPUs are used in a physical resource pool. For example, the GPU driver version is 440.33 .

Parameter	Mandatory	Type	Description
npuVersion	No	String	NPU driver version. This parameter is available when Ascend chips are used in a physical resource pool. For example, the Ascend driver version is C78 .
updateStrategy	No	String	Driver upgrade policy. Options: <ul style="list-style-type: none"> • force: forcible upgrade. The node drivers are upgraded immediately, which may affect jobs running on the node. • idle: secure upgrade. The drivers are upgraded when no job is running on the node.

Table 8-183 PoolUserLogin

Parameter	Mandatory	Type	Description
keyPairName	No	String	Key pair name
password	No	String	Password, which must be salted, encrypted, and encoded using Base64. The default username is root .

Table 8-184 PoolClusterInfo

Parameter	Mandatory	Type	Description
name	No	String	Cluster name
providerId	No	String	Cluster ID

Response Parameters

Status code: 200

Table 8-185 Response body parameters

Parameter	Type	Description
apiVersion	String	API version. Options: <ul style="list-style-type: none"> • v2
kind	String	Resource type. Options: <ul style="list-style-type: none"> • Pool: resource pool
metadata	PoolMetadata object	Metadata of a resource pool.
spec	PoolSpec object	Description of a resource pool.
status	PoolStatus object	Status of a resource pool.

Table 8-186 PoolMetadata

Parameter	Type	Description
name	String	Automatically generated resource pool name, which is equivalent to pool ID . Minimum: 4 Maximum: 128
creationTimestamp	String	Timestamp, for example, 2021-11-01T03:49:41Z.
labels	PoolMetadataLabels object	Resource pool labels.
annotations	PoolMetadataAnnotations object	Resource pool annotations.

Table 8-187 PoolMetadataLabels

Parameter	Type	Description
os.modelarts/workspace.id	String	Workspace ID. Default: 0
os.modelarts/name	String	Specified resource pool name. Minimum: 4 Maximum: 64

Parameter	Type	Description
os.modelarts/resource.id	String	Resource ID, which is typically provided for CBC.
os.modelarts/tenant.domain.id	String	ID of the tenant corresponding to the resource pool, which records the tenant account where the resource pool is created
os.modelarts/create-from	String	Source of a resource pool, for example, admin-console , indicating that the resource pool is created by the administrator on the ModelArts console
os.modelarts.pool/biz	String	Business type of a resource pool. The value can be public or private .
os.modelarts/privileged	String	Whether a resource pool is a privileged pool. If this parameter is specified, the resource pool is a privileged one.
os.modelarts/sharing	String	Whether the resource pool can be shared. Options: <ul style="list-style-type: none"> ● unshared: It cannot be shared. ● shared: It can be shared.

Table 8-188 PoolMetaAnnotations

Parameter	Type	Description
os.modelarts/description	String	Description of a resource pool.
os.modelarts/billing.mode	String	Billing mode. Options: <ul style="list-style-type: none"> ● 0: pay-per-use ● 1: yearly/monthly
os.modelarts/period.num	String	Subscription period on a yearly/monthly basis, for example, 2 .
os.modelarts/period.type	String	Yearly/monthly subscription. Options: <ul style="list-style-type: none"> ● 2: month ● 3: year ● 4: hour
os.modelarts/auto.renew	String	Whether to automatically renew the subscription. Options: <ul style="list-style-type: none"> ● 0: auto-renewal disabled (default value) ● 1: auto-renewal enabled

Parameter	Type	Description
os.modelarts/promotion.info	String	Discount selected in CBC.
os.modelarts/service.console.url	String	URL of the page displayed after the subscription order is paid.
os.modelarts/order.id	String	Order ID, which is mandatory when creating a yearly/monthly resource pool or changing its billing mode.
os.modelarts/flavor.resource.ids	String	Resource ID corresponding to each specification, which is used for interaction with the operations platform
os.modelarts/tms.tags	String	Resource tags specified by the user during creation
os.modelarts/scheduler.extension	String	Priority that can be set for the jobs delivered by other users to the shared resource pool
os.modelarts.pool/subpools.count	String	Number of logical sub-pools in a resource pool

Table 8-189 PoolSpec

Parameter	Type	Description
type	object	Resource pool type. <ul style="list-style-type: none"> • Dedicate: physical resource pool, which uses a separate network and supports network connection, custom drivers, and custom job types. • Logical: logical resource pool, which can be used after the administrator's approval.
scope	Array of strings	Job type enabled, which cannot be specified for logical pools. Options: <ul style="list-style-type: none"> • Train: training jobs • Infer: inference jobs [- Notebook: notebook jobs](tags: hc)
resources	Array of PoolResource FlavorCount objects	Resource specifications in a resource pool, including resource specifications and the number of resources for each specification.

Parameter	Type	Description
network	PoolNetwork object	Network settings for a resource pool. This parameter is mandatory for physical resource pools and is unavailable for logical resource pools.
containerNetwork	PoolClusterContainerNetwork object	Kubernetes container network
kubernetesSvcIpRange	String	CIDR of the Kubernetes service network segment
masters	Array of PoolClusterMaster objects	Master node parameters in a resource pool. This parameter is optional for physical resource pools and is unavailable for logical resource pools.
driver	PoolDriver object	Resource pool driver.
userLogin	PoolUserLogin object	Node login information of a privilege pool
clusters	Array of PoolClusterInfo objects	Privilege pool cluster information
ipv6enable	Boolean	Whether to enable IPv6
controlMode	Integer	Restriction status of a resource pool. Options: <ul style="list-style-type: none"> ● 0: The resource pool is not restricted. ● 1: Changing to yearly/monthly billing is restricted. ● 2: Modifying specifications is restricted. ● 4: The service is restricted. ● 8: The resource pool is frozen. ● 16: The resource pool is frozen by the public security department (cannot be unsubscribed). In addition, multiple statuses are allowed. For example, a resource pool is in state 9, indicating that it is frozen and changing its billing mode to yearly/monthly is restricted.

Table 8-190 PoolResourceFlavorCount

Parameter	Type	Description
flavor	String	Resource specifications name, for example, modelarts.vm.gpu.t4u8
count	Integer	Minimum count for the specifications in a resource pool
maxCount	Integer	Elastic usage of the resource specifications. This parameter value is the same the count value in a physical pool; It is greater than or equal to the count value in a logical pool.
azs	Array of PoolNodeAz objects	AZ where resource pool nodes are deployed.

Table 8-191 PoolNetwork

Parameter	Type	Description
name	String	Network name. When you create a network with a specified name, the system will automatically create subnets for you. By default, the first subnet will be used. Minimum: 4 Maximum: 128
vpcId	String	VPC ID, which must be specified when a privileged pool is created and is unavailable for a non-privileged pool
subnetId	String	Subnet ID, which must be specified when a privileged pool is created and is unavailable for a non-privileged pool

Table 8-192 PoolClusterContainerNetwork

Parameter	Type	Description
mode	String	Container network model
cidr	String	Container network segment. This parameter is available only when the container network model is overlay_l2 or vpc-router .

Table 8-193 PoolClusterMaster

Parameter	Type	Description
az	String	AZ where the master node is located

Table 8-194 PoolDriver

Parameter	Type	Description
gpuVersion	String	GPU driver version. This parameter is available when GPUs are used in a physical resource pool. For example, the GPU driver version is 440.33 .
npuVersion	String	NPU driver version. This parameter is available when Ascend chips are used in a physical resource pool. For example, the Ascend driver version is C78 .
updateStrategy	String	Driver upgrade policy. Options: <ul style="list-style-type: none"> • force: forcible upgrade. The node drivers are upgraded immediately, which may affect jobs running on the node. • idle: secure upgrade. The drivers are upgraded when no job is running on the node.

Table 8-195 PoolUserLogin

Parameter	Type	Description
keyPairName	String	Key pair name
password	String	Password, which must be salted, encrypted, and encoded using Base64. The default username is root .

Table 8-196 PoolStatus

Parameter	Type	Description
phase	String	Resource pool status. Options: <ul style="list-style-type: none"> ● Creating: The resource pool is being created. ● Running: The resource pool is running. ● Abnormal: The resource pool malfunctions. ● Deleting: The resource pool is being deleted. ● Error: An error occurred in the resource pool. ● CreationFailed: Creating the resource pool failed. ● ScalingFailed: Expanding the capacity of the resource pool failed. ● Waiting: The resource pool is awaiting creation, which is typically caused by an unpaid order or unapproved request.
message	String	Message indicating that the resource pool is in the current state.
resources	resources object	Left blank for logical pools, which do not need to be created.
scope	Array of scope objects	Service status of a resource pool.
driver	driver object	Resource pool driver.
parent	String	Name of the parent node of a resource pool. This parameter is left blank for physical pools.
root	String	Name of the root node in a resource pool. For a physical pool, the value is its name.
clusters	Array of PoolClusterInfo objects	Resource pool cluster information. This parameter is available only for privileged pools.

Table 8-197 resources

Parameter	Type	Description
creating	PoolResourceFlavorCount object	Number of resources that are being created.

Parameter	Type	Description
available	PoolResourceFlavorCount object	Number of available resources.
abnormal	PoolResourceFlavorCount object	Number of unavailable resources.
deleting	PoolResourceFlavorCount object	Number of resources that are being deleted.

Table 8-198 PoolNodeAz

Parameter	Type	Description
az	String	AZ name.
count	Integer	Number of nodes for expanding the capacity of a specified AZ. Minimum: 1 Maximum: 2000

Table 8-199 scope

Parameter	Type	Description
scopeType	String	Job type enabled, which cannot be specified for logical pools. Options: <ul style="list-style-type: none"> ● Train: training jobs ● Infer: inference jobs
state	String	Service status. Options: <ul style="list-style-type: none"> ● Enabling: The service is being enabled. ● Enabled: The service is enabled. ● Disabling: The service is being disabled. ● Disabled: The service is disabled.

Table 8-200 driver

Parameter	Type	Description
gpu	PoolDriverStatus object	GPU driver.

Parameter	Type	Description
npu	PoolDriverStatus object	NPU driver.

Table 8-201 PoolDriverStatus

Parameter	Type	Description
version	String	Driver version
state	String	Driver status. Options: <ul style="list-style-type: none"> • Creating: The driver is being created. • Upgrading: The driver is being upgraded. • Running: The driver is running. • Abnormal: The driver malfunctions.

Table 8-202 PoolClusterInfo

Parameter	Type	Description
name	String	Cluster name
providerId	String	Cluster ID

Status code: 400

Table 8-203 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Status code: 409

Table 8-204 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Example Requests

- Create a physical pool.

POST https://{endpoint}/v2/{project_id}/pools

```
{
  "kind": "Pool",
  "apiVersion": "v1",
  "metadata": {
    "labels": {
      "os.modelarts/name": "pool-001",
      "os.modelarts/workspace.id": "xxxxxx"
    },
    "annotations": {
      "os.modelarts/description": "",
      "os.modelarts/billing.mode": "0"
    }
  },
  "spec": {
    "type": "Dedicate",
    "scope": [ "Train" ],
    "network": {
      "name": "net-01"
    },
    "masters": [ {
      "az": "xxxxxx-7a"
    } ],
    "resources": [ {
      "flavor": "modelarts.vm.gpu.t4u8",
      "count": 2
    } ],
    "driver": {
      "gpuVersion": "440.31"
    }
  }
}
```

- Create a logical pool.

POST https://{endpoint}/v2/{project_id}/pools

```
{
  "kind": "Pool",
  "apiVersion": "v1",
  "metadata": {
    "labels": {
      "os.modelarts/name": "pool-logic-01"
    },
    "annotations": {
      "os.modelarts/description": "",
      "os.modelarts/billing.mode": 0,
      "os.modelarts/product.id": "xxx"
    }
  },
  "spec": {
    "type": "Logical",
    "scope": [ "Train" ],
    "resources": [ {
      "flavor": "modelarts.vm.gpu.t4u8",
      "count": 2
    } ]
  }
}
```

Example Responses

Status code: 400

Bad request

```
{
  "error_code": "ModelArts.50004000",
  "error_msg": "Bad request"
}
```

Status code: 409

Already exists

```
{
  "error_code": "ModelArts.50015000",
  "error_msg": "Pool already exists."
}
```

Status Codes

Status Code	Description
200	OK
400	Bad request
409	Already exists

Error Codes

See [Error Codes](#).

8.10.2 Obtaining Resource Pools

Function

This API is used to obtain resource pools.

URI

GET /v2/{project_id}/pools

Table 8-205 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.

Table 8-206 Query Parameters

Parameter	Mandatory	Type	Description
workspaceId	No	String	Workspace ID. If this parameter is left blank, the default workspace is used.

Parameter	Mandatory	Type	Description
labelSelector	No	String	Filter by label.
status	No	String	Status of the selected resource pool. Value created indicates that the resource pool has been created, failed indicates that creating the resource pool failed, and creating indicates that the resource pool is being created. This parameter is left blank by default, indicating that all resource pools will be returned, regardless of their statuses.
parent	No	String	Name of the parent resource pool. It is used to query the sub-pools of a specified resource pool.

Request Parameters

None

Response Parameters

Status code: 200

Table 8-207 Response body parameters

Parameter	Type	Description
apiVersion	String	API version. Options: <ul style="list-style-type: none"> • v2
kind	String	Resource type. Options: <ul style="list-style-type: none"> • PoolList: resource pools
metadata	metadata object	Metadata of resource pools.
items	Array of Pool objects	Resource pools.

Table 8-208 metadata

Parameter	Type	Description
continue	String	Next query location in pagination query.
remainingItemCount	Integer	Number of remaining resources.

Table 8-209 Pool

Parameter	Type	Description
apiVersion	String	API version. Options: <ul style="list-style-type: none"> • v2
kind	String	Resource type. Options: <ul style="list-style-type: none"> • Pool: resource pool
metadata	PoolMetadata object	Metadata of a resource pool.
spec	PoolSpec object	Description of a resource pool.
status	PoolStatus object	Status of a resource pool.

Table 8-210 PoolMetadata

Parameter	Type	Description
name	String	Automatically generated resource pool name, which is equivalent to pool ID . Minimum: 4 Maximum: 128
creationTimestamp	String	Timestamp, for example, 2021-11-01T03:49:41Z.
labels	PoolMetadataLabels object	Resource pool labels.
annotations	PoolMetadataAnnotations object	Resource pool annotations.

Table 8-211 PoolMetaLabels

Parameter	Type	Description
os.modelarts/workspace.id	String	Workspace ID. Default: 0
os.modelarts/name	String	Specified resource pool name. Minimum: 4 Maximum: 64
os.modelarts/resource.id	String	Resource ID, which is typically provided for CBC.
os.modelarts/tenant.domain.id	String	ID of the tenant corresponding to the resource pool, which records the tenant account where the resource pool is created
os.modelarts/create-from	String	Source of a resource pool, for example, admin-console , indicating that the resource pool is created by the administrator on the ModelArts console
os.modelarts.pool/biz	String	Business type of a resource pool. The value can be public or private .
os.modelarts/privileged	String	Whether a resource pool is a privileged pool. If this parameter is specified, the resource pool is a privileged one.
os.modelarts/sharing	String	Whether the resource pool can be shared. Options: <ul style="list-style-type: none"> ● unshared: It cannot be shared. ● shared: It can be shared.

Table 8-212 PoolMetaAnnotations

Parameter	Type	Description
os.modelarts/description	String	Description of a resource pool.
os.modelarts/billing.mode	String	Billing mode. Options: <ul style="list-style-type: none"> ● 0: pay-per-use ● 1: yearly/monthly
os.modelarts/period.num	String	Subscription period on a yearly/monthly basis, for example, 2 .

Parameter	Type	Description
os.modelarts/ period.type	String	Yearly/monthly subscription. Options: <ul style="list-style-type: none"> ● 2: month ● 3: year ● 4: hour
os.modelarts/ auto.renew	String	Whether to automatically renew the subscription. Options: <ul style="list-style-type: none"> ● 0: auto-renewal disabled (default value) ● 1: auto-renewal enabled
os.modelarts/ promotion.info	String	Discount selected in CBC.
os.modelarts/ service.console.url	String	URL of the page displayed after the subscription order is paid.
os.modelarts/ order.id	String	Order ID, which is mandatory when creating a yearly/monthly resource pool or changing its billing mode.
os.modelarts/ flavor.resource.ids	String	Resource ID corresponding to each specification, which is used for interaction with the operations platform
os.modelarts/ tms.tags	String	Resource tags specified by the user during creation
os.modelarts/ scheduler.extension	String	Priority that can be set for the jobs delivered by other users to the shared resource pool
os.modelarts. pool/ subpools.count	String	Number of logical sub-pools in a resource pool

Table 8-213 PoolSpec

Parameter	Type	Description
type	object	Resource pool type. <ul style="list-style-type: none"> ● Dedicate: physical resource pool, which uses a separate network and supports network connection, custom drivers, and custom job types. ● Logical: logical resource pool, which can be used after the administrator's approval.

Parameter	Type	Description
scope	Array of strings	Job type enabled, which cannot be specified for logical pools. Options: <ul style="list-style-type: none"> ● Train: training jobs ● Infer: inference jobs
resources	Array of PoolResourceFlavorCount objects	Resource specifications in a resource pool, including resource specifications and the number of resources for each specification.
network	PoolNetwork object	Network settings for a resource pool. This parameter is mandatory for physical resource pools and is unavailable for logical resource pools.
containerNetwork	PoolClusterContainerNetwork object	Kubernetes container network
kubernetesSvcIpRange	String	CIDR of the Kubernetes service network segment
masters	Array of PoolClusterMaster objects	Master node parameters in a resource pool. This parameter is optional for physical resource pools and is unavailable for logical resource pools.
driver	PoolDriver object	Resource pool driver.
userLogin	PoolUserLogin object	Node login information of a privilege pool
clusters	Array of PoolClusterInfo objects	Privilege pool cluster information
ipv6enable	Boolean	Whether to enable IPv6

Parameter	Type	Description
controlMode	Integer	Restriction status of a resource pool. Options: <ul style="list-style-type: none"> ● 0: The resource pool is not restricted. ● 1: Changing to yearly/monthly billing is restricted. ● 2: Modifying specifications is restricted. ● 4: The service is restricted. ● 8: The resource pool is frozen. ● 16: The resource pool is frozen by the public security department (cannot be unsubscribed). In addition, multiple statuses are allowed. For example, a resource pool is in state 9, indicating that it is frozen and changing its billing mode to yearly/monthly is restricted.

Table 8-214 PoolResourceFlavorCount

Parameter	Type	Description
flavor	String	Resource specifications name, for example, modelarts.vm.gpu.t4u8
count	Integer	Minimum count for the specifications in a resource pool
maxCount	Integer	Elastic usage of the resource specifications. This parameter value is the same the count value in a physical pool; It is greater than or equal to the count value in a logical pool.
azs	Array of PoolNodeAz objects	AZ where resource pool nodes are deployed.

Table 8-215 PoolNetwork

Parameter	Type	Description
name	String	Network name. When you create a network with a specified name, the system will automatically create subnets for you. By default, the first subnet will be used. Minimum: 4 Maximum: 128

Parameter	Type	Description
vpcId	String	VPC ID, which must be specified when a privileged pool is created and is unavailable for a non-privileged pool
subnetId	String	Subnet ID, which must be specified when a privileged pool is created and is unavailable for a non-privileged pool

Table 8-216 PoolClusterContainerNetwork

Parameter	Type	Description
mode	String	Container network model
cidr	String	Container network segment. This parameter is available only when the container network model is overlay_l2 or vpc-router .

Table 8-217 PoolClusterMaster

Parameter	Type	Description
az	String	AZ where the master node is located

Table 8-218 PoolDriver

Parameter	Type	Description
gpuVersion	String	GPU driver version. This parameter is available when GPUs are used in a physical resource pool. For example, the GPU driver version is 440.33 .
npuVersion	String	NPU driver version. This parameter is available when Ascend chips are used in a physical resource pool. For example, the Ascend driver version is C78 .
updateStrategy	String	Driver upgrade policy. Options: <ul style="list-style-type: none"> force: forcible upgrade. The node drivers are upgraded immediately, which may affect jobs running on the node. idle: secure upgrade. The drivers are upgraded when no job is running on the node.

Table 8-219 PoolUserLogin

Parameter	Type	Description
keyPairName	String	Key pair name
password	String	Password, which must be salted, encrypted, and encoded using Base64. The default username is root .

Table 8-220 PoolStatus

Parameter	Type	Description
phase	String	Resource pool status. Options: <ul style="list-style-type: none"> ● Creating: The resource pool is being created. ● Running: The resource pool is running. ● Abnormal: The resource pool malfunctions. ● Deleting: The resource pool is being deleted. ● Error: An error occurred in the resource pool. ● CreationFailed: Creating the resource pool failed. ● ScalingFailed: Expanding the capacity of the resource pool failed. ● Waiting: The resource pool is awaiting creation, which is typically caused by an unpaid order or unapproved request.
message	String	Message indicating that the resource pool is in the current state.
resources	resources object	Left blank for logical pools, which do not need to be created.
scope	Array of scope objects	Service status of a resource pool.
driver	driver object	Resource pool driver.
parent	String	Name of the parent node of a resource pool. This parameter is left blank for physical pools.
root	String	Name of the root node in a resource pool. For a physical pool, the value is its name.
clusters	Array of PoolClusterInfo objects	Resource pool cluster information. This parameter is available only for privileged pools.

Table 8-221 resources

Parameter	Type	Description
creating	PoolResource FlavorCount object	Number of resources that are being created.
available	PoolResource FlavorCount object	Number of available resources.
abnormal	PoolResource FlavorCount object	Number of unavailable resources.
deleting	PoolResource FlavorCount object	Number of resources that are being deleted.

Table 8-222 PoolNodeAz

Parameter	Type	Description
az	String	AZ name.
count	Integer	Number of nodes for expanding the capacity of a specified AZ. Minimum: 1 Maximum: 2000

Table 8-223 scope

Parameter	Type	Description
scopeType	String	Job type enabled, which cannot be specified for logical pools. Options: <ul style="list-style-type: none"> • Train: training jobs • Infer: inference jobs
state	String	Service status. Options: <ul style="list-style-type: none"> • Enabling: The service is being enabled. • Enabled: The service is enabled. • Disabling: The service is being disabled. • Disabled: The service is disabled.

Table 8-224 driver

Parameter	Type	Description
gpu	PoolDriverStatus object	GPU driver.
npu	PoolDriverStatus object	NPU driver.

Table 8-225 PoolDriverStatus

Parameter	Type	Description
version	String	Driver version
state	String	Driver status. Options: <ul style="list-style-type: none"> • Creating: The driver is being created. • Upgrading: The driver is being upgraded. • Running: The driver is running. • Abnormal: The driver malfunctions.

Table 8-226 PoolClusterInfo

Parameter	Type	Description
name	String	Cluster name
providerId	String	Cluster ID

Example Requests

Obtain resource pools.

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

Example Responses

Status code: 200

OK

```
{
  "kind": "PoolList",
  "apiVersion": "v2",
  "items": [ {
    "kind": "Pool",
    "apiVersion": "v2",
    "metadata": {
      "name": "auto-pool-os-86c13962597848eeb29c5861153a391f",
```

```
"creationTimestamp" : "2022-09-16T03:10:40Z",
"labels" : {
  "os.modelarts/name" : "auto-pool-os",
  "os.modelarts/workspace.id" : "0",
  "os.modelarts/resource.id" : "maos-auto-pool-os-72w8d"
},
"annotations" : {
  "os.modelarts/description" : "",
  "os.modelarts/billing.mode" : "0",
  "os.modelarts/external-access" : "elb"
}
},
"spec" : {
  "type" : "Dedicate",
  "scope" : [ "Train", "Infer" ],
  "resources" : [ {
    "flavor" : "modelarts.vm.cpu.4ud",
    "count" : 2
  } ],
  "network" : {
    "name" : "network-maos-86c13962597848eeb29c5861153a391f"
  }
},
"status" : {
  "phase" : "Running",
  "root" : "auto-pool-os-86c13962597848eeb29c5861153a391f",
  "scope" : [ {
    "scopeType" : "Train",
    "state" : "Enabled"
  }, {
    "scopeType" : "Infer",
    "state" : "Enabled"
  } ],
  "resources" : {
    "available" : [ {
      "flavor" : "modelarts.vm.cpu.4ud",
      "count" : 2,
      "azs" : [ {
        "az" : "xxxxxx-7c",
        "count" : 2
      } ]
    } ]
  }
}
}
}
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

8.10.3 Obtaining a Resource Pool

Function

This API is used to obtain details about a specified resource pool.

URI

GET /v2/{project_id}/pools/{pool_name}

Table 8-227 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
pool_name	Yes	String	Name of a resource pool.

Request Parameters

None

Response Parameters

Status code: 200

Table 8-228 Response body parameters

Parameter	Type	Description
apiVersion	String	API version. Options: <ul style="list-style-type: none"> • v2
kind	String	Resource type. Options: <ul style="list-style-type: none"> • Pool: resource pool
metadata	PoolMetadata object	Metadata of a resource pool.
spec	PoolSpec object	Description of a resource pool.
status	PoolStatus object	Status of a resource pool.

Table 8-229 PoolMetadata

Parameter	Type	Description
name	String	Automatically generated resource pool name, which is equivalent to pool ID . Minimum: 4 Maximum: 128
creationTimestamp	String	Timestamp, for example, 2021-11-01T03:49:41Z.
labels	PoolMetaLabels object	Resource pool labels.
annotations	PoolMetaAnnotations object	Resource pool annotations.

Table 8-230 PoolMetaLabels

Parameter	Type	Description
os.modelarts/workspace.id	String	Workspace ID. Default: 0
os.modelarts/name	String	Specified resource pool name. Minimum: 4 Maximum: 64
os.modelarts/resource.id	String	Resource ID, which is typically provided for CBC.
os.modelarts/tenant.domain.id	String	ID of the tenant corresponding to the resource pool, which records the tenant account where the resource pool is created
os.modelarts/create-from	String	Source of a resource pool, for example, admin-console , indicating that the resource pool is created by the administrator on the ModelArts console
os.modelarts.pool/biz	String	Business type of a resource pool. The value can be public or private .
os.modelarts/privileged	String	Whether a resource pool is a privileged pool. If this parameter is specified, the resource pool is a privileged one.

Parameter	Type	Description
os.modelarts/sharing	String	Whether the resource pool can be shared. Options: <ul style="list-style-type: none"> ● unshared: It cannot be shared. ● shared: It can be shared.

Table 8-231 PoolMetaAnnotations

Parameter	Type	Description
os.modelarts/description	String	Description of a resource pool.
os.modelarts/billing.mode	String	Billing mode. Options: <ul style="list-style-type: none"> ● 0: pay-per-use ● 1: yearly/monthly
os.modelarts/period.num	String	Subscription period on a yearly/monthly basis, for example, 2 .
os.modelarts/period.type	String	Yearly/monthly subscription. Options: <ul style="list-style-type: none"> ● 2: month ● 3: year ● 4: hour
os.modelarts/auto.renew	String	Whether to automatically renew the subscription. Options: <ul style="list-style-type: none"> ● 0: auto-renewal disabled (default value) ● 1: auto-renewal enabled
os.modelarts/promotion.info	String	Discount selected in CBC.
os.modelarts/service.console.url	String	URL of the page displayed after the subscription order is paid.
os.modelarts/order.id	String	Order ID, which is mandatory when creating a yearly/monthly resource pool or changing its billing mode.
os.modelarts/flavor.resource.ids	String	Resource ID corresponding to each specification, which is used for interaction with the operations platform
os.modelarts/tms.tags	String	Resource tags specified by the user during creation

Parameter	Type	Description
os.modelarts/scheduler.extension	String	Priority that can be set for the jobs delivered by other users to the shared resource pool
os.modelarts.pool/subpools.count	String	Number of logical sub-pools in a resource pool

Table 8-232 PoolSpec

Parameter	Type	Description
type	object	Resource pool type. <ul style="list-style-type: none"> • Dedicate: physical resource pool, which uses a separate network and supports network connection, custom drivers, and custom job types. • Logical: logical resource pool, which can be used after the administrator's approval.
scope	Array of strings	Job type enabled, which cannot be specified for logical pools. Options: <ul style="list-style-type: none"> • Train: training jobs • Infer: inference jobs
resources	Array of PoolResourceFlavorCount objects	Resource specifications in a resource pool, including resource specifications and the number of resources for each specification.
network	PoolNetwork object	Network settings for a resource pool. This parameter is mandatory for physical resource pools and is unavailable for logical resource pools.
containerNetwork	PoolClusterContainerNetwork object	Kubernetes container network
kubernetesServiceIpRange	String	CIDR of the Kubernetes service network segment
masters	Array of PoolClusterMaster objects	Master node parameters in a resource pool. This parameter is optional for physical resource pools and is unavailable for logical resource pools.
driver	PoolDriver object	Resource pool driver.

Parameter	Type	Description
userLogin	PoolUserLogin object	Node login information of a privilege pool
clusters	Array of PoolClusterInfo objects	Privilege pool cluster information
ipv6enable	Boolean	Whether to enable IPv6
controlMode	Integer	Restriction status of a resource pool. Options: <ul style="list-style-type: none"> ● 0: The resource pool is not restricted. ● 1: Changing to yearly/monthly billing is restricted. ● 2: Modifying specifications is restricted. ● 4: The service is restricted. ● 8: The resource pool is frozen. ● 16: The resource pool is frozen by the public security department (cannot be unsubscribed). In addition, multiple statuses are allowed. For example, a resource pool is in state 9, indicating that it is frozen and changing its billing mode to yearly/monthly is restricted.

Table 8-233 PoolResourceFlavorCount

Parameter	Type	Description
flavor	String	Resource specifications name, for example, modelarts.vm.gpu.t4u8
count	Integer	Minimum count for the specifications in a resource pool
maxCount	Integer	Elastic usage of the resource specifications. This parameter value is the same the count value in a physical pool; It is greater than or equal to the count value in a logical pool.
azs	Array of PoolNodeAz objects	AZ where resource pool nodes are deployed.

Table 8-234 PoolNetwork

Parameter	Type	Description
name	String	Network name. When you create a network with a specified name, the system will automatically create subnets for you. By default, the first subnet will be used. Minimum: 4 Maximum: 128
vpcId	String	VPC ID, which must be specified when a privileged pool is created and is unavailable for a non-privileged pool
subnetId	String	Subnet ID, which must be specified when a privileged pool is created and is unavailable for a non-privileged pool

Table 8-235 PoolClusterContainerNetwork

Parameter	Type	Description
mode	String	Container network model
cidr	String	Container network segment. This parameter is available only when the container network model is overlay_l2 or vpc-router .

Table 8-236 PoolClusterMaster

Parameter	Type	Description
az	String	AZ where the master node is located

Table 8-237 PoolDriver

Parameter	Type	Description
gpuVersion	String	GPU driver version. This parameter is available when GPUs are used in a physical resource pool. For example, the GPU driver version is 440.33 .
npuVersion	String	NPU driver version. This parameter is available when Ascend chips are used in a physical resource pool. For example, the Ascend driver version is C78 .

Parameter	Type	Description
updateStrategy	String	Driver upgrade policy. Options: <ul style="list-style-type: none"> • force: forcible upgrade. The node drivers are upgraded immediately, which may affect jobs running on the node. • idle: secure upgrade. The drivers are upgraded when no job is running on the node.

Table 8-238 PoolUserLogin

Parameter	Type	Description
keyPairName	String	Key pair name
password	String	Password, which must be salted, encrypted, and encoded using Base64. The default username is root .

Table 8-239 PoolStatus

Parameter	Type	Description
phase	String	Resource pool status. Options: <ul style="list-style-type: none"> • Creating: The resource pool is being created. • Running: The resource pool is running. • Abnormal: The resource pool malfunctions. • Deleting: The resource pool is being deleted. • Error: An error occurred in the resource pool. • CreationFailed: Creating the resource pool failed. • ScalingFailed: Expanding the capacity of the resource pool failed. • Waiting: The resource pool is awaiting creation, which is typically caused by an unpaid order or unapproved request.
message	String	Message indicating that the resource pool is in the current state.
resources	resources object	Left blank for logical pools, which do not need to be created.

Parameter	Type	Description
scope	Array of scope objects	Service status of a resource pool.
driver	driver object	Resource pool driver.
parent	String	Name of the parent node of a resource pool. This parameter is left blank for physical pools.
root	String	Name of the root node in a resource pool. For a physical pool, the value is its name.
clusters	Array of PoolClusterInfo objects	Resource pool cluster information. This parameter is available only for privileged pools.

Table 8-240 resources

Parameter	Type	Description
creating	PoolResourceFlavorCount object	Number of resources that are being created.
available	PoolResourceFlavorCount object	Number of available resources.
abnormal	PoolResourceFlavorCount object	Number of unavailable resources.
deleting	PoolResourceFlavorCount object	Number of resources that are being deleted.

Table 8-241 PoolNodeAz

Parameter	Type	Description
az	String	AZ name.
count	Integer	Number of nodes for expanding the capacity of a specified AZ. Minimum: 1 Maximum: 2000

Table 8-242 scope

Parameter	Type	Description
scopeType	String	Job type enabled, which cannot be specified for logical pools. Options: <ul style="list-style-type: none"> ● Train: training jobs ● Infer: inference jobs
state	String	Service status. Options: <ul style="list-style-type: none"> ● Enabling: The service is being enabled. ● Enabled: The service is enabled. ● Disabling: The service is being disabled. ● Disabled: The service is disabled.

Table 8-243 driver

Parameter	Type	Description
gpu	PoolDriverStatus object	GPU driver.
npu	PoolDriverStatus object	NPU driver.

Table 8-244 PoolDriverStatus

Parameter	Type	Description
version	String	Driver version
state	String	Driver status. Options: <ul style="list-style-type: none"> ● Creating: The driver is being created. ● Upgrading: The driver is being upgraded. ● Running: The driver is running. ● Abnormal: The driver malfunctions.

Table 8-245 PoolClusterInfo

Parameter	Type	Description
name	String	Cluster name
providerId	String	Cluster ID

Status code: 404

Table 8-246 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Example Requests

Obtain details about a resource pool.

```
GET https://{endpoint}/v2/{project_id}/pools/{pool_name}
{ }
```

Example Responses

Status code: 200

OK

```
{
  "kind": "Pool",
  "apiVersion": "v2",
  "metadata": {
    "name": "auto-pool-os-86c13962597848eeb29c5861153a391f",
    "creationTimestamp": "2022-09-16T03:10:40Z",
    "labels": {
      "os.modelarts/name": "auto-pool-os",
      "os.modelarts/workspace.id": "0",
      "os.modelarts/resource.id": "maos-auto-pool-os-72w8d"
    },
    "annotations": {
      "os.modelarts/description": "",
      "os.modelarts/billing.mode": "0",
      "os.modelarts/external-access": "elb"
    }
  },
  "spec": {
    "type": "Dedicate",
    "scope": [ "Train", "Infer" ],
    "resources": [ {
      "flavor": "modelarts.vm.cpu.4ud",
      "count": 2
    } ],
    "network": {
      "name": "network-maos-86c13962597848eeb29c5861153a391f"
    }
  },
  "status": {
    "phase": "Running",
    "root": "auto-pool-os-86c13962597848eeb29c5861153a391f",
    "scope": [ {
      "scopeType": "Train",
      "state": "Enabled"
    }, {
      "scopeType": "Infer",
      "state": "Enabled"
    } ],
    "resources": {
```

```

"available" : [ {
  "flavor" : "modelarts.vm.cpu.4ud",
  "count" : 2,
  "azs" : [ {
    "az" : "xxxxxx-7c",
    "count" : 2
  } ]
} ]
}
}
}

```

Status code: 404

Not found.

```

{
  "error_code" : "ModelArts.50015001",
  "error_msg" : "pool not found"
}

```

Status Codes

Status Code	Description
200	OK
404	Not found.

Error Codes

See [Error Codes](#).

8.10.4 Deleting a Resource Pool

Function

This API is used to delete a specified resource pool.

URI

DELETE /v2/{project_id}/pools/{pool_name}

Table 8-247 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
pool_name	Yes	String	Automatically generated resource pool name.

Request Parameters

Table 8-248 Request header parameters

Parameter	Mandatory	Type	Description
request-type	No	String	Request type.

Response Parameters

Status code: 200

Table 8-249 Response body parameters

Parameter	Type	Description
apiVersion	String	API version. Options: <ul style="list-style-type: none"> • v2
kind	String	Resource type. Options: <ul style="list-style-type: none"> • Pool: resource pool
metadata	PoolMetadataDeletion object	Metadata of a resource pool.
spec	PoolSpec object	Description of a resource pool.
status	PoolStatus object	Status of a resource pool.

Table 8-250 PoolMetadataDeletion

Parameter	Type	Description
name	String	Automatically generated resource pool name, which is equivalent to pool ID .
creationTimestamp	String	Timestamp, for example, 2021-11-01T03:49:41Z.
deletionTimestamp	String	Timestamp, for example, 2021-11-01T03:49:41Z.
labels	PoolMetadataLabels object	Resource pool labels.
annotations	PoolMetadataAnnotations object	Resource pool annotations.

Table 8-251 PoolMetaLabels

Parameter	Type	Description
os.modelarts/workspace.id	String	Workspace ID. Default: 0
os.modelarts/name	String	Specified resource pool name. Minimum: 4 Maximum: 64
os.modelarts/resource.id	String	Resource ID, which is typically provided for CBC.
os.modelarts/tenant.domain.id	String	ID of the tenant corresponding to the resource pool, which records the tenant account where the resource pool is created
os.modelarts/create-from	String	Source of a resource pool, for example, admin-console , indicating that the resource pool is created by the administrator on the ModelArts console
os.modelarts.pool/biz	String	Business type of a resource pool. The value can be public or private .
os.modelarts/privileged	String	Whether a resource pool is a privileged pool. If this parameter is specified, the resource pool is a privileged one.
os.modelarts/sharing	String	Whether the resource pool can be shared. Options: <ul style="list-style-type: none"> ● unshared: It cannot be shared. ● shared: It can be shared.

Table 8-252 PoolMetaAnnotations

Parameter	Type	Description
os.modelarts/description	String	Description of a resource pool.
os.modelarts/billing.mode	String	Billing mode. Options: <ul style="list-style-type: none"> ● 0: pay-per-use ● 1: yearly/monthly
os.modelarts/period.num	String	Subscription period on a yearly/monthly basis, for example, 2 .

Parameter	Type	Description
os.modelarts/ period.type	String	Yearly/monthly subscription. Options: <ul style="list-style-type: none"> • 2: month • 3: year • 4: hour
os.modelarts/ auto.renew	String	Whether to automatically renew the subscription. Options: <ul style="list-style-type: none"> • 0: auto-renewal disabled (default value) • 1: auto-renewal enabled
os.modelarts/ promotion.info	String	Discount selected in CBC.
os.modelarts/ service.console.url	String	URL of the page displayed after the subscription order is paid.
os.modelarts/ order.id	String	Order ID, which is mandatory when creating a yearly/monthly resource pool or changing its billing mode.
os.modelarts/ flavor.resource.ids	String	Resource ID corresponding to each specification, which is used for interaction with the operations platform
os.modelarts/ tms.tags	String	Resource tags specified by the user during creation
os.modelarts/ scheduler.extension	String	Priority that can be set for the jobs delivered by other users to the shared resource pool
os.modelarts. pool/ subpools.count	String	Number of logical sub-pools in a resource pool

Table 8-253 PoolSpec

Parameter	Type	Description
type	object	Resource pool type. <ul style="list-style-type: none"> • Dedicate: physical resource pool, which uses a separate network and supports network connection, custom drivers, and custom job types. • Logical: logical resource pool, which can be used after the administrator's approval.

Parameter	Type	Description
scope	Array of strings	Job type enabled, which cannot be specified for logical pools. Options: <ul style="list-style-type: none"> • Train: training jobs • Infer: inference jobs
resources	Array of PoolResourceFlavorCount objects	Resource specifications in a resource pool, including resource specifications and the number of resources for each specification.
network	PoolNetwork object	Network settings for a resource pool. This parameter is mandatory for physical resource pools and is unavailable for logical resource pools.
containerNetwork	PoolClusterContainerNetwork object	Kubernetes container network
kubernetesSvcIpRange	String	CIDR of the Kubernetes service network segment
masters	Array of PoolClusterMaster objects	Master node parameters in a resource pool. This parameter is optional for physical resource pools and is unavailable for logical resource pools.
driver	PoolDriver object	Resource pool driver.
userLogin	PoolUserLogin object	Node login information of a privilege pool
clusters	Array of PoolClusterInfo objects	Privilege pool cluster information
ipv6enable	Boolean	Whether to enable IPv6

Parameter	Type	Description
controlMode	Integer	Restriction status of a resource pool. Options: <ul style="list-style-type: none"> ● 0: The resource pool is not restricted. ● 1: Changing to yearly/monthly billing is restricted. ● 2: Modifying specifications is restricted. ● 4: The service is restricted. ● 8: The resource pool is frozen. ● 16: The resource pool is frozen by the public security department (cannot be unsubscribed). In addition, multiple statuses are allowed. For example, a resource pool is in state 9, indicating that it is frozen and changing its billing mode to yearly/monthly is restricted.

Table 8-254 PoolResourceFlavorCount

Parameter	Type	Description
flavor	String	Resource specifications name, for example, modelarts.vm.gpu.t4u8
count	Integer	Minimum count for the specifications in a resource pool
maxCount	Integer	Elastic usage of the resource specifications. This parameter value is the same the count value in a physical pool; It is greater than or equal to the count value in a logical pool.
azs	Array of PoolNodeAz objects	AZ where resource pool nodes are deployed.

Table 8-255 PoolNetwork

Parameter	Type	Description
name	String	Network name. When you create a network with a specified name, the system will automatically create subnets for you. By default, the first subnet will be used. Minimum: 4 Maximum: 128

Parameter	Type	Description
vpcId	String	VPC ID, which must be specified when a privileged pool is created and is unavailable for a non-privileged pool
subnetId	String	Subnet ID, which must be specified when a privileged pool is created and is unavailable for a non-privileged pool

Table 8-256 PoolClusterContainerNetwork

Parameter	Type	Description
mode	String	Container network model
cidr	String	Container network segment. This parameter is available only when the container network model is overlay_l2 or vpc-router .

Table 8-257 PoolClusterMaster

Parameter	Type	Description
az	String	AZ where the master node is located

Table 8-258 PoolDriver

Parameter	Type	Description
gpuVersion	String	GPU driver version. This parameter is available when GPUs are used in a physical resource pool. For example, the GPU driver version is 440.33 .
npuVersion	String	NPU driver version. This parameter is available when Ascend chips are used in a physical resource pool. For example, the Ascend driver version is C78 .
updateStrategy	String	Driver upgrade policy. Options: <ul style="list-style-type: none"> force: forcible upgrade. The node drivers are upgraded immediately, which may affect jobs running on the node. idle: secure upgrade. The drivers are upgraded when no job is running on the node.

Table 8-259 PoolUserLogin

Parameter	Type	Description
keyPairName	String	Key pair name
password	String	Password, which must be salted, encrypted, and encoded using Base64. The default username is root .

Table 8-260 PoolStatus

Parameter	Type	Description
phase	String	Resource pool status. Options: <ul style="list-style-type: none"> ● Creating: The resource pool is being created. ● Running: The resource pool is running. ● Abnormal: The resource pool malfunctions. ● Deleting: The resource pool is being deleted. ● Error: An error occurred in the resource pool. ● CreationFailed: Creating the resource pool failed. ● ScalingFailed: Expanding the capacity of the resource pool failed. ● Waiting: The resource pool is awaiting creation, which is typically caused by an unpaid order or unapproved request.
message	String	Message indicating that the resource pool is in the current state.
resources	resources object	Left blank for logical pools, which do not need to be created.
scope	Array of scope objects	Service status of a resource pool.
driver	driver object	Resource pool driver.
parent	String	Name of the parent node of a resource pool. This parameter is left blank for physical pools.
root	String	Name of the root node in a resource pool. For a physical pool, the value is its name.
clusters	Array of PoolClusterInfo objects	Resource pool cluster information. This parameter is available only for privileged pools.

Table 8-261 resources

Parameter	Type	Description
creating	PoolResource FlavorCount object	Number of resources that are being created.
available	PoolResource FlavorCount object	Number of available resources.
abnormal	PoolResource FlavorCount object	Number of unavailable resources.
deleting	PoolResource FlavorCount object	Number of resources that are being deleted.

Table 8-262 PoolNodeAz

Parameter	Type	Description
az	String	AZ name.
count	Integer	Number of nodes for expanding the capacity of a specified AZ. Minimum: 1 Maximum: 2000

Table 8-263 scope

Parameter	Type	Description
scopeType	String	Job type enabled, which cannot be specified for logical pools. Options: <ul style="list-style-type: none"> ● Train: training jobs ● Infer: inference jobs
state	String	Service status. Options: <ul style="list-style-type: none"> ● Enabling: The service is being enabled. ● Enabled: The service is enabled. ● Disabling: The service is being disabled. ● Disabled: The service is disabled.

Table 8-264 driver

Parameter	Type	Description
gpu	PoolDriverStatus object	GPU driver.
npu	PoolDriverStatus object	NPU driver.

Table 8-265 PoolDriverStatus

Parameter	Type	Description
version	String	Driver version
state	String	Driver status. Options: <ul style="list-style-type: none"> • Creating: The driver is being created. • Upgrading: The driver is being upgraded. • Running: The driver is running. • Abnormal: The driver malfunctions.

Table 8-266 PoolClusterInfo

Parameter	Type	Description
name	String	Cluster name
providerId	String	Cluster ID

Status code: 404

Table 8-267 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Example Requests

Delete a resource pool.

```
DELETE https://{endpoint}/v2/{project_id}/pools/{pool_name}
{ }
```

Example Responses

Status code: 200

OK

```
{
  "kind": "Pool",
  "apiVersion": "v2",
  "metadata": {
    "name": "auto-pool-os-86c13962597848eeb29c5861153a391f",
    "creationTimestamp": "2022-09-16T03:10:40Z",
    "labels": {
      "os.modelarts/name": "auto-pool-os",
      "os.modelarts/workspace.id": "0",
      "os.modelarts/resource.id": "maos-auto-pool-os-72w8d"
    },
    "annotations": {
      "os.modelarts/description": "",
      "os.modelarts/billing.mode": "0",
      "os.modelarts/external-access": "elb"
    }
  },
  "spec": {
    "type": "Dedicate",
    "scope": [ "Train", "Infer" ],
    "resources": [ {
      "flavor": "modelarts.vm.cpu.4ud",
      "count": 2
    } ],
    "network": {
      "name": "network-maos-86c13962597848eeb29c5861153a391f"
    }
  },
  "status": {
    "phase": "Running",
    "root": "auto-pool-os-86c13962597848eeb29c5861153a391f",
    "scope": [ {
      "scopeType": "Train",
      "state": "Enabled"
    }, {
      "scopeType": "Infer",
      "state": "Enabled"
    } ],
    "resources": {
      "available": [ {
        "flavor": "modelarts.vm.cpu.4ud",
        "count": 2,
        "azs": [ {
          "az": "xxxxxx-7c",
          "count": 2
        } ]
      } ]
    }
  }
}
```

Status code: 404

Not found.

```
{
  "error_code": "ModelArts.50015001",
  "error_msg": "pool not found"
}
```

Status Codes

Status Code	Description
200	OK
404	Not found.

Error Codes

See [Error Codes](#).

8.10.5 Updating a Resource Pool

Function

This API is used to update a specified resource pool.

URI

PATCH /v2/{project_id}/pools/{pool_name}

Table 8-268 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
pool_name	Yes	String	Automatically generated resource pool name.

Request Parameters

Table 8-269 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Application/merge-patch + JSON.
request-type	No	String	Request type.

Table 8-270 Request body parameters

Parameter	Mandatory	Type	Description
metadata	No	PoolMetadataUpdate object	Metadata of a resource pool.
spec	No	PoolSpecUpdate object	Description of a resource pool.

Table 8-271 PoolMetadataUpdate

Parameter	Mandatory	Type	Description
annotations	No	annotations object	Resource pool annotations.
labels	No	labels object	Resource pool labels

Table 8-272 annotations

Parameter	Mandatory	Type	Description
os.modelarts/description	No	String	Description of a resource pool, specifying the application scenarios of the resource pool. The following special characters are not allowed: ! <>=&"" Minimum: 0 Maximum: 100
os.modelarts/order.id	No	String	Order ID, which is required when creating or modifying a yearly/monthly order.
os.modelarts/license/exclude.scope	No	String	Whether a resource pool is controlled by license. If the label is not contained or the value is invalid, the resource pool must be controlled by license.
os.modelarts/scheduler.extension	No	String	Priority that can be set for the jobs delivered by other users to the shared resource pool

Table 8-273 labels

Parameter	Mandatory	Type	Description
os.modelarts/sharing	No	String	Whether the resource pool can be shared. Options: <ul style="list-style-type: none"> • unshared: It cannot be shared. • shared: It can be shared.
os.modelarts/workspace.id	No	String	Workspace ID. Note: The workspace ID cannot be changed while other resource pool operations (such as scaling) are being performed.

Table 8-274 PoolSpecUpdate

Parameter	Mandatory	Type	Description
scope	No	Array of strings	Job type enabled, which cannot be specified for logical pools. Options: <ul style="list-style-type: none"> • Train: training jobs • Infer: inference jobs
resources	No	Array of PoolResourceFlavorCount objects	Data model for the number of resources of the specified specifications
driver	No	PoolDriver object	Resource pool driver, which cannot be specified for logical pools.

Table 8-275 PoolResourceFlavorCount

Parameter	Mandatory	Type	Description
flavor	Yes	String	Resource specifications name, for example, modelarts.vm.gpu.t4u8
count	Yes	Integer	Minimum count for the specifications in a resource pool

Parameter	Mandatory	Type	Description
maxCount	Yes	Integer	Elastic usage of the resource specifications. This parameter value is the same the count value in a physical pool; It is greater than or equal to the count value in a logical pool.
azs	No	Array of PoolNodeAz objects	AZ where resource pool nodes are deployed.

Table 8-276 PoolNodeAz

Parameter	Mandatory	Type	Description
az	Yes	String	AZ name.
count	Yes	Integer	Number of nodes for expanding the capacity of a specified AZ. Minimum: 1 Maximum: 2000

Table 8-277 PoolDriver

Parameter	Mandatory	Type	Description
gpuVersion	No	String	GPU driver version. This parameter is available when GPUs are used in a physical resource pool. For example, the GPU driver version is 440.33 .
npuVersion	No	String	NPU driver version. This parameter is available when Ascend chips are used in a physical resource pool. For example, the Ascend driver version is C78 .

Parameter	Mandatory	Type	Description
updateStrategy	No	String	Driver upgrade policy. Options: <ul style="list-style-type: none"> • force: forcible upgrade. The node drivers are upgraded immediately, which may affect jobs running on the node. • idle: secure upgrade. The drivers are upgraded when no job is running on the node.

Response Parameters

Status code: 200

Table 8-278 Response body parameters

Parameter	Type	Description
apiVersion	String	API version. Options: <ul style="list-style-type: none"> • v2
kind	String	Resource type. Options: <ul style="list-style-type: none"> • Pool: resource pool
metadata	PoolMetadata object	Metadata of a resource pool.
spec	PoolSpec object	Description of a resource pool.
status	PoolStatus object	Status of a resource pool.

Table 8-279 PoolMetadata

Parameter	Type	Description
name	String	Automatically generated resource pool name, which is equivalent to pool ID . Minimum: 4 Maximum: 128
creationTimestamp	String	Timestamp, for example, 2021-11-01T03:49:41Z.

Parameter	Type	Description
labels	PoolMetaLabels object	Resource pool labels.
annotations	PoolMetaAnnotations object	Resource pool annotations.

Table 8-280 PoolMetaLabels

Parameter	Type	Description
os.modelarts/workspace.id	String	Workspace ID. Default: 0
os.modelarts/name	String	Specified resource pool name. Minimum: 4 Maximum: 64
os.modelarts/resource.id	String	Resource ID, which is typically provided for CBC.
os.modelarts/tenant.domain.id	String	ID of the tenant corresponding to the resource pool, which records the tenant account where the resource pool is created
os.modelarts/create-from	String	Source of a resource pool, for example, admin-console , indicating that the resource pool is created by the administrator on the ModelArts console
os.modelarts.pool/biz	String	Business type of a resource pool. The value can be public or private .
os.modelarts/privileged	String	Whether a resource pool is a privileged pool. If this parameter is specified, the resource pool is a privileged one.
os.modelarts/sharing	String	Whether the resource pool can be shared. Options: <ul style="list-style-type: none"> • unshared: It cannot be shared. • shared: It can be shared.

Table 8-281 PoolMetaAnnotations

Parameter	Type	Description
os.modelarts/description	String	Description of a resource pool.

Parameter	Type	Description
os.modelarts/ billing.mode	String	Billing mode. Options: <ul style="list-style-type: none"> ● 0: pay-per-use ● 1: yearly/monthly
os.modelarts/ period.num	String	Subscription period on a yearly/monthly basis, for example, 2.
os.modelarts/ period.type	String	Yearly/monthly subscription. Options: <ul style="list-style-type: none"> ● 2: month ● 3: year ● 4: hour
os.modelarts/ auto.renew	String	Whether to automatically renew the subscription. Options: <ul style="list-style-type: none"> ● 0: auto-renewal disabled (default value) ● 1: auto-renewal enabled
os.modelarts/ promotion.info	String	Discount selected in CBC.
os.modelarts/ service.console.url	String	URL of the page displayed after the subscription order is paid.
os.modelarts/ order.id	String	Order ID, which is mandatory when creating a yearly/monthly resource pool or changing its billing mode.
os.modelarts/ flavor.resource.ids	String	Resource ID corresponding to each specification, which is used for interaction with the operations platform
os.modelarts/ tms.tags	String	Resource tags specified by the user during creation
os.modelarts/ scheduler.extension	String	Priority that can be set for the jobs delivered by other users to the shared resource pool
os.modelarts. pool/ subpools.count	String	Number of logical sub-pools in a resource pool

Table 8-282 PoolSpec

Parameter	Type	Description
type	object	Resource pool type. <ul style="list-style-type: none"> • Dedicate: physical resource pool, which uses a separate network and supports network connection, custom drivers, and custom job types. • Logical: logical resource pool, which can be used after the administrator's approval.
scope	Array of strings	Job type enabled, which cannot be specified for logical pools. Options: <ul style="list-style-type: none"> • Train: training jobs • Infer: inference jobs
resources	Array of PoolResourceFlavorCount objects	Resource specifications in a resource pool, including resource specifications and the number of resources for each specification.
network	PoolNetwork object	Network settings for a resource pool. This parameter is mandatory for physical resource pools and is unavailable for logical resource pools.
containerNetwork	PoolClusterContainerNetwork object	Kubernetes container network
kubernetesSvcIpRange	String	CIDR of the Kubernetes service network segment
masters	Array of PoolClusterMaster objects	Master node parameters in a resource pool. This parameter is optional for physical resource pools and is unavailable for logical resource pools.
driver	PoolDriver object	Resource pool driver.
userLogin	PoolUserLogin object	Node login information of a privilege pool
clusters	Array of PoolClusterInfo objects	Privilege pool cluster information
ipv6enable	Boolean	Whether to enable IPv6

Parameter	Type	Description
controlMode	Integer	Restriction status of a resource pool. Options: <ul style="list-style-type: none"> ● 0: The resource pool is not restricted. ● 1: Changing to yearly/monthly billing is restricted. ● 2: Modifying specifications is restricted. ● 4: The service is restricted. ● 8: The resource pool is frozen. ● 16: The resource pool is frozen by the public security department (cannot be unsubscribed). In addition, multiple statuses are allowed. For example, a resource pool is in state 9, indicating that it is frozen and changing its billing mode to yearly/monthly is restricted.

Table 8-283 PoolResourceFlavorCount

Parameter	Type	Description
flavor	String	Resource specifications name, for example, modelarts.vm.gpu.t4u8
count	Integer	Minimum count for the specifications in a resource pool
maxCount	Integer	Elastic usage of the resource specifications. This parameter value is the same the count value in a physical pool; It is greater than or equal to the count value in a logical pool.
azs	Array of PoolNodeAz objects	AZ where resource pool nodes are deployed.

Table 8-284 PoolNetwork

Parameter	Type	Description
name	String	Network name. When you create a network with a specified name, the system will automatically create subnets for you. By default, the first subnet will be used. Minimum: 4 Maximum: 128

Parameter	Type	Description
vpcId	String	VPC ID, which must be specified when a privileged pool is created and is unavailable for a non-privileged pool
subnetId	String	Subnet ID, which must be specified when a privileged pool is created and is unavailable for a non-privileged pool

Table 8-285 PoolClusterContainerNetwork

Parameter	Type	Description
mode	String	Container network model
cidr	String	Container network segment. This parameter is available only when the container network model is overlay_l2 or vpc-router .

Table 8-286 PoolClusterMaster

Parameter	Type	Description
az	String	AZ where the master node is located

Table 8-287 PoolDriver

Parameter	Type	Description
gpuVersion	String	GPU driver version. This parameter is available when GPUs are used in a physical resource pool. For example, the GPU driver version is 440.33 .
npuVersion	String	NPU driver version. This parameter is available when Ascend chips are used in a physical resource pool. For example, the Ascend driver version is C78 .
updateStrategy	String	Driver upgrade policy. Options: <ul style="list-style-type: none"> force: forcible upgrade. The node drivers are upgraded immediately, which may affect jobs running on the node. idle: secure upgrade. The drivers are upgraded when no job is running on the node.

Table 8-288 PoolUserLogin

Parameter	Type	Description
keyPairName	String	Key pair name
password	String	Password, which must be salted, encrypted, and encoded using Base64. The default username is root .

Table 8-289 PoolStatus

Parameter	Type	Description
phase	String	Resource pool status. Options: <ul style="list-style-type: none"> ● Creating: The resource pool is being created. ● Running: The resource pool is running. ● Abnormal: The resource pool malfunctions. ● Deleting: The resource pool is being deleted. ● Error: An error occurred in the resource pool. ● CreationFailed: Creating the resource pool failed. ● ScalingFailed: Expanding the capacity of the resource pool failed. ● Waiting: The resource pool is awaiting creation, which is typically caused by an unpaid order or unapproved request.
message	String	Message indicating that the resource pool is in the current state.
resources	resources object	Left blank for logical pools, which do not need to be created.
scope	Array of scope objects	Service status of a resource pool.
driver	driver object	Resource pool driver.
parent	String	Name of the parent node of a resource pool. This parameter is left blank for physical pools.
root	String	Name of the root node in a resource pool. For a physical pool, the value is its name.
clusters	Array of PoolClusterInfo objects	Resource pool cluster information. This parameter is available only for privileged pools.

Table 8-290 resources

Parameter	Type	Description
creating	PoolResource FlavorCount object	Number of resources that are being created.
available	PoolResource FlavorCount object	Number of available resources.
abnormal	PoolResource FlavorCount object	Number of unavailable resources.
deleting	PoolResource FlavorCount object	Number of resources that are being deleted.

Table 8-291 PoolNodeAz

Parameter	Type	Description
az	String	AZ name.
count	Integer	Number of nodes for expanding the capacity of a specified AZ. Minimum: 1 Maximum: 2000

Table 8-292 scope

Parameter	Type	Description
scopeType	String	Job type enabled, which cannot be specified for logical pools. Options: <ul style="list-style-type: none"> • Train: training jobs • Infer: inference jobs
state	String	Service status. Options: <ul style="list-style-type: none"> • Enabling: The service is being enabled. • Enabled: The service is enabled. • Disabling: The service is being disabled. • Disabled: The service is disabled.

Table 8-293 driver

Parameter	Type	Description
gpu	PoolDriverStatus object	GPU driver.
npu	PoolDriverStatus object	NPU driver.

Table 8-294 PoolDriverStatus

Parameter	Type	Description
version	String	Driver version
state	String	Driver status. Options: <ul style="list-style-type: none"> • Creating: The driver is being created. • Upgrading: The driver is being upgraded. • Running: The driver is running. • Abnormal: The driver malfunctions.

Table 8-295 PoolClusterInfo

Parameter	Type	Description
name	String	Cluster name
providerId	String	Cluster ID

Status code: 400

Table 8-296 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Status code: 404

Table 8-297 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Example Requests

Expand the capacity of a resource pool.

```
PATCH https://{endpoint}/v2/{project_id}/pools/{pool_name}
{
  "spec": {
    "resources": [ {
      "flavor": "modelarts.vm.cpu.4ud",
      "count": 3
    } ]
  }
}
```

Example Responses

Status code: 200

OK

```
{
  "kind": "Pool",
  "apiVersion": "v2",
  "metadata": {
    "name": "auto-pool-os-86c13962597848eeb29c5861153a391f",
    "creationTimestamp": "2022-09-16T03:10:40Z",
    "labels": {
      "os.modelarts/name": "auto-pool-os",
      "os.modelarts/workspace.id": "0",
      "os.modelarts/resource.id": "maos-auto-pool-os-72w8d"
    },
    "annotations": {
      "os.modelarts/description": "",
      "os.modelarts/billing.mode": "0"
    }
  },
  "spec": {
    "type": "Dedicate",
    "scope": [ "Train", "Infer" ],
    "resources": [ {
      "flavor": "modelarts.vm.cpu.4ud",
      "count": 3
    } ],
    "network": {
      "name": "network-maos-86c13962597848eeb29c5861153a391f"
    }
  },
  "status": {
    "phase": "Running",
    "root": "auto-pool-os-86c13962597848eeb29c5861153a391f",
    "scope": [ {
      "scopeType": "Train",
      "state": "Enabled"
    } ],
    {
      "scopeType": "Infer",

```

```

"state" : "Enabled"
}],
"resources" : {
  "creating" : [{
    "flavor" : "modelarts.vm.cpu.4ud",
    "count" : 1,
    "azs" : [{
      "az" : "xxxxxx-7c",
      "count" : 1
    }]
  }],
  "available" : [{
    "flavor" : "modelarts.vm.cpu.4ud",
    "count" : 2,
    "azs" : [{
      "az" : "xxxxxx-7c",
      "count" : 2
    }]
  }]
}
}
}

```

Status code: 400

Bad request

```

{
  "error_code" : "ModelArts.50004000",
  "error_msg" : "Bad request"
}

```

Status code: 404

Not found.

```

{
  "error_code" : "ModelArts.50015001",
  "error_msg" : "pool not found"
}

```

Status Codes

Status Code	Description
200	OK
400	Bad request
404	Not found.

Error Codes

See [Error Codes](#).

8.10.6 Monitoring a Resource Pool

Function

This API is used to obtain the monitored resource pool information.

URI

GET /v2/{project_id}/pools/{pool_name}/monitor

Table 8-298 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
pool_name	Yes	String	Automatically generated resource pool name.

Table 8-299 Query Parameters

Parameter	Mandatory	Type	Description
time_range	No	String	Default value: -1.-1.60
statistics	No	String	Default value: maximum
period	No	String	Default value: 60

Request Parameters

None

Response Parameters

Status code: 200

Table 8-300 Response body parameters

Parameter	Type	Description
metrics	Array of metrics objects	Metric list. The value is a JSON array that contains a maximum of 20 objects.

Table 8-301 metrics

Parameter	Type	Description
metric	metric object	Metrics
dataPoints	Array of dataPoints objects	Key metrics

Table 8-302 metric

Parameter	Type	Description
dimensions	Array of dimensions objects	Dimensions
metricName	String	Metric name. Options: <ul style="list-style-type: none"> • cpuUsage: CPU usage • memUsedRate: memory usage • gpuUtil: GPU usage • gpuMemUsage: used GPU memory • npuUtil: NPU usage • npuMemUsage: used NPU memory • diskCapacity: disk capacity • diskAvailableCapacity: available disk capacity • diskUsedRate: disk usage
namespace	String	Metric namespace. Options: <ul style="list-style-type: none"> • PAAS.CONTAINER: namespace of component, instance, process, and container metrics • PAAS.NODE: namespace of host, network, disk, and file system metrics • PAAS.SLA: namespace of SLA metrics • PAAS.AGGR: namespace of cluster metrics • CUSTOMMETRICS: default namespace of custom metrics

Table 8-303 dimensions

Parameter	Type	Description
name	String	Metric dimension name
value	String	Metric dimension value

Table 8-304 dataPoints

Parameter	Type	Description
timestamp	Integer	Timestamp
unit	String	Time series unit

Parameter	Type	Description
statistics	Array of statistics objects	List of statistical values

Table 8-305 statistics

Parameter	Type	Description
statistic	String	Statistical mode. Options: <ul style="list-style-type: none"> • maximum: maximum value statistics • average: average value statistics
value	Float	Statistical result. The value -1 indicates invalid data.

Status code: 404

Table 8-306 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Example Requests

Obtain the monitored resource pool information.

```
GET https://{endpoint}/v2/{project_id}/pools/{pool_name}/monitor
{ }
```

Example Responses

Status code: 200

OK

```
{
  "metrics" : [ {
    "metric" : {
      "dimensions" : [ {
        "name" : "clusterId",
        "value" : "83df17f1-d74c-11ec-a070-0255ac1000c3"
      } ],
      "metricName" : "cpuUsage",
      "namespace" : "PAAS.AGGR"
    },
    "dataPoints" : [ {
```

```

"timestamp" : 1655193600000,
"unit" : "Percent",
"statistics" : [ {
  "statistic" : "average",
  "value" : 7.944
} ]
}, {
"timestamp" : 1655197200000,
"unit" : "Percent",
"statistics" : [ {
  "statistic" : "average",
  "value" : 7.88
} ]
} ]
}, {
"metric" : {
  "dimensions" : [ {
    "name" : "clusterId",
    "value" : "83df17f1-d74c-11ec-a070-0255ac1000c3"
  } ],
  "metricName" : "memUsedRate",
  "namespace" : "PAAS.AGGR"
},
"dataPoints" : [ {
  "timestamp" : 1655193600000,
  "unit" : "Percent",
  "statistics" : [ {
    "statistic" : "average",
    "value" : 13.83
  } ]
} ]
}, {
"timestamp" : 1655197200000,
"unit" : "Percent",
"statistics" : [ {
  "statistic" : "average",
  "value" : 13.836
} ]
} ]
}, {
"metric" : {
  "dimensions" : [ {
    "name" : "clusterId",
    "value" : "83df17f1-d74c-11ec-a070-0255ac1000c3"
  } ],
  "metricName" : "gpuUtil",
  "namespace" : "PAAS.AGGR"
},
"dataPoints" : [ {
  "timestamp" : 1655193600000,
  "unit" : "Percent",
  "statistics" : [ {
    "statistic" : "average",
    "value" : -1
  } ]
} ]
}, {
"timestamp" : 1655197200000,
"unit" : "Percent",
"statistics" : [ {
  "statistic" : "average",
  "value" : -1
} ]
} ]
}, {
"metric" : {
  "dimensions" : [ {
    "name" : "clusterId",
    "value" : "83df17f1-d74c-11ec-a070-0255ac1000c3"
  } ],
  "metricName" : "gpuMemUsage",

```



```
"namespace" : "PAAS.AGGR"
},
"dataPoints" : [ {
  "timestamp" : 1655193600000,
  "unit" : "Percent",
  "statistics" : [ {
    "statistic" : "average",
    "value" : -1
  } ]
}, {
  "timestamp" : 1655197200000,
  "unit" : "Percent",
  "statistics" : [ {
    "statistic" : "average",
    "value" : -1
  } ]
} ]
}, {
  "metric" : {
    "dimensions" : [ {
      "name" : "clusterId",
      "value" : "83df17f1-d74c-11ec-a070-0255ac1000c3"
    } ],
    "metricName" : "npuUtil",
    "namespace" : "PAAS.AGGR"
  },
  "dataPoints" : [ {
    "timestamp" : 1655193600000,
    "unit" : "",
    "statistics" : [ {
      "statistic" : "average",
      "value" : -1
    } ]
  }, {
    "timestamp" : 1655197200000,
    "unit" : "",
    "statistics" : [ {
      "statistic" : "average",
      "value" : -1
    } ]
  } ]
}, {
  "metric" : {
    "dimensions" : [ {
      "name" : "clusterId",
      "value" : "83df17f1-d74c-11ec-a070-0255ac1000c3"
    } ],
    "metricName" : "npuMemUsage",
    "namespace" : "PAAS.AGGR"
  },
  "dataPoints" : [ {
    "timestamp" : 1655193600000,
    "unit" : "",
    "statistics" : [ {
      "statistic" : "average",
      "value" : -1
    } ]
  }, {
    "timestamp" : 1655197200000,
    "unit" : "",
    "statistics" : [ {
      "statistic" : "average",
      "value" : -1
    } ]
  } ]
}, {
  "metric" : {
    "dimensions" : [ {
      "name" : "clusterId",
```

```

    "value" : "83df17f1-d74c-11ec-a070-0255ac1000c3"
  },
  "metricName" : "diskAvailableCapacity",
  "namespace" : "PAAS.AGGR"
},
"dataPoints" : [ {
  "timestamp" : 1655193600000,
  "unit" : "Megabytes",
  "statistics" : [ {
    "statistic" : "average",
    "value" : 834383.4
  } ]
}, {
  "timestamp" : 1655197200000,
  "unit" : "Megabytes",
  "statistics" : [ {
    "statistic" : "average",
    "value" : 834379.2
  } ]
} ]
}, {
  "metric" : {
    "dimensions" : [ {
      "name" : "clusterId",
      "value" : "83df17f1-d74c-11ec-a070-0255ac1000c3"
    } ],
    "metricName" : "diskCapacity",
    "namespace" : "PAAS.AGGR"
  },
  "dataPoints" : [ {
    "timestamp" : 1655193600000,
    "unit" : "Megabytes",
    "statistics" : [ {
      "statistic" : "average",
      "value" : 1105920
    } ]
  }, {
    "timestamp" : 1655197200000,
    "unit" : "Megabytes",
    "statistics" : [ {
      "statistic" : "average",
      "value" : 1105920
    } ]
  } ]
}, {
  "metric" : {
    "dimensions" : [ {
      "name" : "clusterId",
      "value" : "83df17f1-d74c-11ec-a070-0255ac1000c3"
    } ],
    "metricName" : "diskUsedRate",
    "namespace" : "PAAS.AGGR"
  },
  "dataPoints" : [ {
    "timestamp" : 1655193600000,
    "unit" : "Percent",
    "statistics" : [ {
      "statistic" : "average",
      "value" : 24.553
    } ]
  }, {
    "timestamp" : 1655197200000,
    "unit" : "Percent",
    "statistics" : [ {
      "statistic" : "average",
      "value" : 24.553
    } ]
  } ]
} ]
} ]

```

```
}]
}
```

Status code: 404

Not found.

```
{
  "error_code" : "ModelArts.50015001",
  "error_msg" : "pool not found"
}
```

Status Codes

Status Code	Description
200	OK
404	Not found.

Error Codes

See [Error Codes](#).

8.10.7 Obtaining Resource Pool Statistics

Function

This API is used to obtain resource pool statistics.

URI

GET /v2/{project_id}/statistics/pools

Table 8-307 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .

Table 8-308 Query Parameters

Parameter	Mandatory	Type	Description
workspaceId	No	String	Workspace. The default value is 0 .

Request Parameters

None

Response Parameters

Status code: 200

Table 8-309 Response body parameters

Parameter	Type	Description
statistics	statistics object	Resource pool statistics
operationTime	String	Statistics time

Table 8-310 statistics

Parameter	Type	Description
status	status object	Statistics about resource pools in different statuses

Table 8-311 status

Parameter	Type	Description
creating	Integer	Number of resource pools that are being created
created	Integer	Number of created resource pools
failed	Integer	Number of resource pools that failed to be created in the last three days. The maximum value is 500 .
pending	Integer	Number of resource pools in the waiting status, usually yearly/monthly pools that have not been paid

Status code: 500

Table 8-312 Response body parameters

Parameter	Type	Description
error_code	String	Error code.

Parameter	Type	Description
error_msg	String	Error message.

Example Requests

Obtaining the monitored information of a resource pool

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

Example Responses

Status code: 200

OK

```
{
  "statistics" : {
    "status" : {
      "created" : 3,
      "creating" : 0,
      "failed" : 1,
      "pending" : 0
    }
  },
  "operationTime" : "2022-12-05 11:15:59.329633162 +0000 UTC"
}
```

Status code: 500

Internal error

```
{
  "error_code" : "ModelArts.50005000",
  "error_msg" : "internal error"
}
```

Status Codes

Status Code	Description
200	OK
500	Internal error

Error Codes

See [Error Codes](#).

8.11 Resource Specifications Management

8.11.1 Obtaining Resource Specifications

Function

Obtain resource specifications.

URI

GET /v1/{project_id}/resourceflavors

Table 8-313 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.

Table 8-314 Query Parameters

Parameter	Mandatory	Type	Description
continue	No	String	Previous query location in pagination query.
labelSelector	No	String	Filter by label.
limit	No	Integer	Number of records on each page.

Request Parameters

None

Response Parameters

Status code: 200

Table 8-315 Response body parameters

Parameter	Type	Description
apiVersion	String	API version. Options: <ul style="list-style-type: none"> • v1
kind	String	Resource type. Options: <ul style="list-style-type: none"> • ResourceFlavorList: resource specifications
metadata	metadata object	Metadata of resource specifications.

Parameter	Type	Description
items	Array of ResourceFlavor objects	Resource specifications.

Table 8-316 metadata

Parameter	Type	Description
continue	String	Next query location in pagination query.
remainingItemCount	Integer	Remaining resources.

Table 8-317 ResourceFlavor

Parameter	Type	Description
apiVersion	String	API version. Options: <ul style="list-style-type: none"> • v1
kind	String	Resource type. Options: <ul style="list-style-type: none"> • ResourceFlavor: resource specification
metadata	metadata object	Metadata of a resource specification.
spec	ResourceFlavorSpec object	Description of a resource specification.
status	ResourceFlavorStatus object	Status of a resource specification.

Table 8-318 metadata

Parameter	Type	Description
name	String	Resource specification name.
labels	ResourceFlavorLabel object	Labels of a resource specification

Table 8-319 ResourceFlavorLabel

Parameter	Type	Description
os.modelarts/scope	String	Job types supported by a resource specification

Table 8-320 ResourceFlavorSpec

Parameter	Type	Description
type	String	Resource specification type. Options: <ul style="list-style-type: none"> • Dedicate: physical resources • Logical: logical resources
cpuArch	String	Computer architecture. Options: <ul style="list-style-type: none"> • x86 • arm64
cpu	String	Number of CPU cores.
memory	String	Memory size in GiB.
gpu	gpu object	GPU information.
npu	npu object	NPU information.
dataVolumes	Array of dataVolumes objects	Data disks
billingModes	Array of integers	Billing mode supported by the flavor. Options: <ul style="list-style-type: none"> • 0: pay-per-use • 1: yearly/monthly
jobFlavors	Array of strings	Training job types supported by resource specifications.

Table 8-321 gpu

Parameter	Type	Description
type	String	GPU type.
size	String	Number of GPUs
memory	String	GPU memory

Table 8-322 npu

Parameter	Type	Description
type	String	NPU type.
size	String	Number of NPUs.
memory	String	NPU memory

Table 8-323 dataVolumes

Parameter	Type	Description
volumeType	String	Disk type. Options: <ul style="list-style-type: none"> • SSD: ultra-high I/O disk • GPSSD: general-purpose SSD • SAS: high I/O disk • SATA: common disk
size	String	Disk size, in GiB

Table 8-324 ResourceFlavorStatus

Parameter	Type	Description
phase	Map<String,String>	Sales status of a resource specification in each AZ. The value is (AZ, Status).

Status code: 401

Table 8-325 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Status code: 404

Table 8-326 Response body parameters

Parameter	Type	Description
error_code	String	Error code.

Parameter	Type	Description
error_msg	String	Error message.

Example Requests

This API is used to obtain resource specifications.

```
GET https://{endpoint}/v1/{project_id}/resourceflavors
{ }
```

Example Responses

Status code: 200

OK

```
{
  "kind" : "ResourceFlavorList",
  "apiVersion" : "v1",
  "metadata" : { },
  "items" : [ {
    "kind" : "ResourceFlavor",
    "apiVersion" : "v1",
    "metadata" : {
      "name" : "modelarts.vm.cpu8u32g",
      "labels" : { }
    },
    "spec" : {
      "cpuArch" : "x86",
      "cpu" : "8",
      "memory" : "32Gi",
      "type" : "Dedicate",
      "billingModes" : [ 0 ],
      "dataVolumes" : [ {
        "volumeType" : "SSD",
        "size" : "500Gi"
      } ]
    },
    "status" : {
      "phase" : {
        "xxxxxx-7a" : "soldout",
        "xxxxxx-7b" : "soldout",
        "xxxxxx-7c" : "normal"
      }
    }
  } ]
}
```

Status code: 401

Authorization failed.

```
{
  "error_code" : "ModelArts.50001000",
  "error_msg" : "token is invalid"
}
```

Status code: 404

Not found.

```
{
  "error_code" : "ModelArts.50005101",
```

```
"error_msg" : "Resourceflavor not found."  
}
```

Status Codes

Status Code	Description
200	OK
401	Authorization failed.
404	Not found.

Error Codes

See [Error Codes](#).

9 Authorization Management

9.1 Viewing an Authorization List

Function

This API is used to view an authorization list.

URI

GET /v2/{project_id}/authorizations

Table 9-1 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .

Table 9-2 Query Parameters

Parameter	Mandatory	Type	Description
sort_by	No	String	Sorting field. Options: <ul style="list-style-type: none">• user_name: IAM user• create_time: created at
order	No	String	Sorting method. Options: <ul style="list-style-type: none">• asc: ascending order• desc: descending order

Parameter	Mandatory	Type	Description
limit	No	Integer	Maximum number of records returned on each page. Default value: 1000 The value range ranges from 1 to 1000.
offset	No	Integer	Start page for pagination display. The default value is 0 .

Request Parameters

None

Response Parameters

Status code: 200

Table 9-3 Response body parameters

Parameter	Type	Description
total_count	Number	Authorization information.
auth	Array of AuthorizationResponse objects	Authorization information list.

Table 9-4 AuthorizationResponse

Parameter	Type	Description
user_id	String	User ID. For details about how to obtain a user ID, see Obtaining a User ID . If user_id is set to all , all IAM users are authorized. If some IAM users have been authorized, the authorization setting will be updated. This parameter is mandatory only if the authorization method is set to Agency .
type	String	Authorization type. Agency is recommended. Options: <ul style="list-style-type: none"> • agency: authorization through an agency • credential: authorization through an access Key (AK/SK)

Parameter	Type	Description
content	String	Authorization content. <ul style="list-style-type: none"> If Authorization Type is set to Agency, this field indicates the agency name. If Authorization Type is set to AK/SK, this field indicates the access key ID (AK).
secret_key	String	Secret Access Key (SK). This field is required only when Authorization Method is set to AK/SK .
user_name	String	Username. If user_id is set to all-users , all users will be displayed.
create_time	Long	Timestamp when a training job was created

Example Requests

View an authorization list.

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

Example Responses

Status code: 200

OK

```
{
  "total_count": 1,
  "auth": [{
    "user_id": "****d80fb058844ae8b82aa66d9fe****",
    "user_name": "iam-user01",
    "type": "agency",
    "content": "modelarts_agency",
    "create_time": 15657747821288
  }]
}
```

Status Codes

Status Code	Description
200	OK
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found

Error Codes

See [Error Codes](#).

9.2 Configuring Authorization

Function

This API is used to configure ModelArts authorization. ModelArts functions such as training management, development environment, data management, and real-time services can be properly used only after required permissions are assigned. The administrator can use this API to configure an agency for IAM users and configure the access key of the current user.

URI

POST /v2/{project_id}/authorizations

Table 9-5 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .

Request Parameters

Table 9-6 Request body parameters

Parameter	Mandatory	Type	Description
user_id	No	String	User ID. For details about how to obtain a user ID, see Obtaining a User ID . If user_id is set to all , all IAM users are authorized. If some IAM users have been authorized, the authorization setting will be updated. This parameter is mandatory only if the authorization method is set to Agency .

Parameter	Mandatory	Type	Description
type	No	String	Authorization type. Agency is recommended. Options: <ul style="list-style-type: none"> • agency: authorization through an agency • credential: authorization through an access Key (AK/SK)
content	Yes	String	Authorization content. <ul style="list-style-type: none"> • If Authorization Type is set to Agency, this field indicates the agency name. • If Authorization Type is set to AK/SK, this field indicates the access key ID (AK).
secret_key	No	String	Secret Access Key (SK). This field is required only when Authorization Method is set to AK/SK .
user_name	No	String	Username. If user_id is set to all-users , all users will be displayed.

Response Parameters

None

Example Requests

The following is an example of how to upload authorization whose authorization type is **agency** and authorization content is **modelarts_agency**.

```
POST https://{endpoint}/v2/{project_id}/authorizations
{
  "user_id" : "****d80fb058844ae8b82aa66d9fe****",
  "type" : "agency",
  "content" : "modelarts_agency"
}
```

Example Responses

Status code: 200

OK


```
{
  "result" : "true"
}
```

Status Codes

Status Code	Description
200	OK
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found

Error Codes

See [Error Codes](#).

9.3 Deleting Authorization

Function

This API is used to delete the authorization of a specified user or all users.

URI

DELETE /v2/{project_id}/authorizations

Table 9-7 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .

Table 9-8 Query Parameters

Parameter	Mandatory	Type	Description
user_id	No	String	User ID. If this parameter is set to all , the authorization of all IAM users will be deleted.

Request Parameters

None

Response Parameters

None

Example Requests

Delete the authorization of a specified user.

```
DELETE https://{endpoint}/v2/{project_id}/authorizations?user_id=****d80fb058844ae8b82aa66d9fe****
```

Example Responses

Status code: 200

OK

```
{  
  "result" : "true"  
}
```

Status Codes

Status Code	Description
200	OK
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found

Error Codes

See [Error Codes](#).

9.4 Creating a ModelArts Agency

Function

This API is used to create an agency so that ModelArts can access dependent services such as OBS, SWR, and IEF.

URI

POST /v2/{project_id}/agency

Table 9-9 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. For details, see Obtaining a Project ID and Name .

Request Parameters

Table 9-10 Request body parameters

Parameter	Mandatory	Type	Description
agency_name _suffix	No	String	<p>Agency name suffix.</p> <p>The parameter contains a maximum of 50 characters.</p> <p>The agency name prefix is consistently to be ma_agency.</p> <p>For example, if the value of this parameter is iam-user01, the name of the created agency is ma_agency_iam-user01.</p> <p>The value of this parameter is left blank by default, indicating that an agency named modelarts_agency is created.</p>

Response Parameters

None

Example Requests

The following is an example of how to create a ModelArts agency whose name suffix is **iam-user01**.

```
POST https://{endpoint}/v2/{project_id}/agency
{
  "agency_name_suffix" : "iam-user01"
}
```

Example Responses

Status code: 200

OK

```
{  
  "agency_name" : "ma_agency_iam-user01"  
}
```

Status Codes

Status Code	Description
200	OK
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found

Error Codes

See [Error Codes](#).

10 Use Cases

10.1 Creating a Development Environment Instance

This section describes how to create a development environment instance by calling ModelArts APIs.

Overview

The process for creating a development environment instance is as follows:

1. Call the API for [authentication](#) to obtain a user token, which will be added in a request header for authentication.
2. Call the API for [querying supported images](#) to view the image type and version in the development environment.
3. Call the API for [creating a notebook instance](#) to create an instance.
4. Call the API for [querying details of a notebook instance](#) to query the instance creation details based on the instance ID.
5. Call the API for [prolonging a notebook instance](#) to reset the usage duration of the instance.
6. Call the API for [stopping a notebook instance](#) to stop the instance that is running.
7. Call the API for [starting a notebook instance](#) to restart the instance.
8. Call the API for [deleting a notebook instance](#) to delete the instance that is no longer needed.

Prerequisites

- You have obtained the endpoints of [ModelArts](#).
- The following information is available: region where ModelArts is deployed, project ID and name, account name and ID, and username and user ID.

Procedure

1. Call the API for [querying supported images](#) to view the image type and version in the development environment.

a. Request body:

URI: GET `https://{ma_endpoint}/v1/{project_id}/images`

Request header:

- X-auth-Token
→ *MIIZmgYJKoZlhvcNAQcCollizCCGYcCAQExDTALBglghkgBZQMEAgEwgXXXXXX...*
- Content-Type → application/json

Set the following parameters based on site requirements:

- *ma_endpoint*: ModelArts endpoint
- *project_id*: user's project ID
- **X-auth-Token**: token obtained in the previous step

b. Status code **200** is returned. The response body is as follows:

```
{
  "current": 0,
  "data": [
    {
      "arch": "x86_64",
      "description": "CPU and GPU general algorithm development and training, preconfigured with
AI engine PyTorch1.8",
      "dev_services": [
        "NOTEBOOK",
        "SSH"
      ],
      "id": "278e88d1-5b71-4766-8502-b3ba72e824d9",
      "name": "pytorch1.8-cuda10.2-cudnn7-ubuntu18.04",
      "resource_categories": [
        "CPU",
        "GPU"
      ],
      "service_type": "COMMON",
      "status": "ACTIVE",
      "swr_path": "swr.com/atelier/pytorch_1_8:pytorch_1.8.0-cuda_10.2-py_3.7-ubuntu_18.04-
x86_64-20221118143845-d65d817",
      "tag": "pytorch_1.8.0-cuda_10.2-py_3.7-ubuntu_18.04-x86_64-20221118143845-d65d817",
      "tags": [],
      "type": "BUILD_IN",
      "update_at": 1648866992843,
      "workspace_id": "0"
    },
    {
      "arch": "x86_64",
      "description": "CPU and GPU general algorithm development and training, preconfigured with
AI engine MindSpore1.7.0 and cuda10.1",
      "dev_services": [
        "NOTEBOOK",
        "SSH"
      ],
      "id": "e1a07296-22a8-4f05-8bc8-e936c8e54202",
      "name": "mindspore1.7.0-cuda10.1-py3.7-ubuntu18.04",
      "resource_categories": [
        "GPU"
      ],
      "service_type": "TRAIN",
      "status": "ACTIVE",
      "swr_path": "swr.com/atelier/mindspore_1_7_0:mindspore_1.7.0-cuda_10.1-py_3.7-
ubuntu_18.04-x86_64-20221118143809-d65d817",
      "tag": "mindspore_1.7.0-cuda_10.1-py_3.7-ubuntu_18.04-x86_64-20221118143809-d65d817",
      "tags": [],
      "type": "BUILD_IN",

```

```

"workspace_id": "0"
},
{
  "arch": "x86_64",
  "description": "CPU general algorithm development and training, preconfigured with AI
engine MindSpore1.7.0",
  "dev_services": [
    "NOTEBOOK",
    "SSH"
  ],
  "id": "c0b31f09-1490-4555-9b8b-ab0b2de35b20",
  "name": "mindspore1.7.0-py3.7-ubuntu18.04",
  "resource_categories": [
    "CPU"
  ],
  "service_type": "TRAIN",
  "status": "ACTIVE",
  "swr_path": "swr.com/atelier/mindspore_1_7_0:mindspore_1.7.0-cpu-py_3.7-ubuntu_18.04-
x86_64-20221118143809-d65d817",
  "tag": "mindspore_1.7.0-cpu-py_3.7-ubuntu_18.04-x86_64-20221118143809-d65d817",
  "tags": [],
  "type": "BUILD_IN",
  "workspace_id": "0"
},
{
  "arch": "x86_64",
  "description": "CPU and GPU general algorithm development and training, preconfigured with
AI engine TensorFlow2.1",
  "dev_services": [
    "NOTEBOOK",
    "SSH"
  ],
  "id": "e1a07296-22a8-4f05-8bc8-e936c8e54100",
  "name": "tensorflow2.1-cuda10.1-cudnn7-ubuntu18.04",
  "resource_categories": [
    "CPU",
    "GPU"
  ],
  "service_type": "COMMON",
  "status": "ACTIVE",
  "swr_path": "swr.com/atelier/tensorflow_2_1:tensorflow_2.1.0-cuda_10.1-py_3.7-ubuntu_18.04-
x86_64-20221121111529-d65d817",
  "tag": "tensorflow_2.1.0-cuda_10.1-py_3.7-ubuntu_18.04-x86_64-20221121111529-d65d817",
  "tags": [],
  "type": "BUILD_IN",
  "update_at": 1643166780367,
  "workspace_id": "0"
},
{
  "arch": "x86_64",
  "description": "CPU and GPU general algorithm development and training, preconfigured with
AI engine PyTorch1.10 and cuda10.2",
  "dev_services": [
    "NOTEBOOK",
    "SSH"
  ],
  "id": "d996b661-e127-48c4-a90a-fca29535f201",
  "name": "pytorch1.10-cuda10.2-cudnn7-ubuntu18.04",
  "resource_categories": [
    "CPU",
    "GPU"
  ],
  "service_type": "UNKNOWN",
  "status": "ACTIVE",
  "swr_path": "swr.com/atelier/pytorch_1_10:pytorch_1.10.2-cuda_10.2-py_3.7-ubuntu_18.04-
x86_64-20221118143845-d65d817",
  "tag": "pytorch_1.10.2-cuda_10.2-py_3.7-ubuntu_18.04-x86_64-20221118143845-d65d817",
  "tags": [],
  "type": "BUILD_IN",

```

```

"workspace_id": "0"
},
{
  "arch": "x86_64",
  "description": "Clean user customized base image include cuda10.2, conda",
  "dev_services": [
    "NOTEBOOK",
    "SSH"
  ],
  "id": "d937149a-785c-4d2d-a568-8dde7c06cca0",
  "name": "conda3-cuda10.2-cudnn7-ubuntu18.04",
  "resource_categories": [
    "GPU"
  ],
  "service_type": "UNKNOWN",
  "status": "ACTIVE",
  "swr_path": "swr.com/atelier/user_defined_base:cuda_10.2-ubuntu_18.04-x86_64-20230404095316-7fcd503",
  "tag": "cuda_10.2-ubuntu_18.04-x86_64-20230404095316-7fcd503",
  "tags": [],
  "type": "BUILD_IN",
  "workspace_id": "0"
},
{
  "arch": "x86_64",
  "description": "Clean user customized base image only include conda",
  "dev_services": [
    "NOTEBOOK",
    "SSH"
  ],
  "id": "27542a4a-3b37-404d-add9-a7d2d2ce6893",
  "name": "conda3-ubuntu18.04",
  "resource_categories": [
    "CPU"
  ],
  "service_type": "UNKNOWN",
  "status": "ACTIVE",
  "swr_path": "swr.com/atelier/user_defined_base:ubuntu_18.04-x86_64-20230404095316-7fcd503",
  "tag": "ubuntu_18.04-x86_64-20230404095316-7fcd503",
  "tags": [],
  "type": "BUILD_IN",
  "workspace_id": "0"
},
{
  "arch": "x86_64",
  "description": "CPU and GPU general algorithm development and training, preconfigured with AI engine PyTorch1.4",
  "dev_services": [
    "NOTEBOOK",
    "SSH"
  ],
  "id": "e1a07296-22a8-4f05-8bc8-e936c8e54099",
  "name": "pytorch1.4-cuda10.1-cudnn7-ubuntu18.04",
  "resource_categories": [
    "CPU",
    "GPU"
  ],
  "service_type": "TRAIN",
  "status": "ACTIVE",
  "swr_path": "swr.com/atelier/pytorch_1_4:pytorch_1.4-cuda_10.1-py37-ubuntu_18.04-x86_64-20221118143845-d65d817",
  "tag": "pytorch_1.4-cuda_10.1-py37-ubuntu_18.04-x86_64-20221118143845-d65d817",
  "tags": [],
  "type": "BUILD_IN",
  "update_at": 1648866992868,
  "workspace_id": "0"
},
{

```



```

"arch": "x86_64",
"description": "GPU algorithm development and training, preconfigured with AI engine TensorFlow1.13.1",
"dev_services": [
  "NOTEBOOK",
  "SSH"
],
"id": "b80bbf3d-a7af-42f6-ad12-33ff9116ab0d",
"name": "tensorflow1.13-cuda10.0-cudnn7-ubuntu18.04",
"resource_categories": [
  "GPU"
],
"service_type": "TRAIN",
"status": "ACTIVE",
"swr_path": "swr.com/atelier/tensorflow_1_13:tensorflow_1.13-cuda_10.0-py_3.7-ubuntu_18.04-x86_64-20221118143845-d65d817",
"tag": "tensorflow_1.13-cuda_10.0-py_3.7-ubuntu_18.04-x86_64-20221118143845-d65d817",
"tags": [],
"type": "BUILD_IN",
"update_at": 1648866992960,
"workspace_id": "0"
},
{
  "arch": "aarch64",
  "create_at": 1608937196685,
  "description": "Ascend+ARM algorithm development and training. TensorFlow and MindSpore are preset in the AI engine.",
  "dev_services": [
    "NOTEBOOK",
    "SSH"
  ],
  "id": "59a6e9f5-93c0-44dd-85b0-82f390c5d53a",
  "name": "tensorflow1.15-mindspore1.7.0-cann5.1.0-euler2.8-aarch64",
  "resource_categories": [
    "ASCEND"
  ],
  "service_type": "TRAIN",
  "status": "ACTIVE",
  "swr_path": "swr.com/atelier/notebook2.0-mul-kernel-arm-ascend-cp37.5.0.1-c81-20220726",
  "tag": "5.0.1-c81-20220726",
  "tags": [],
  "type": "BUILD_IN",
  "update_at": 1648866992983,
  "workspace_id": "0"
},
{
  "arch": "x86_64",
  "description": "AI inference application development, preconfigured ModelBox and AI engine LibTorch, only SSH connection supported.",
  "dev_services": [
    "AI_FLOW",
    "SSH"
  ],
  "id": "e1a07296-22a8-4f05-8bc8-e936c8e54103",
  "name": "modelbox1.3.0-libtorch1.9.1-cuda10.2-cudnn8-euler2.9.6",
  "resource_categories": [
    "GPU"
  ],
  "service_type": "TRAIN",
  "status": "ACTIVE",
  "swr_path": "swr.com/atelier/modelarts-modelbox-libtorch-gpu-x86:1.3.0-20221223142251-b3da6d6",
  "tag": "1.3.0-20221223142251-b3da6d6",
  "tags": [],
  "type": "BUILD_IN",
  "update_at": 1648866993005,
  "workspace_id": "0"
},
{

```

```

"arch": "x86_64",
"description": "AI inference application development, preconfigured ModelBox and AI engine TensorRT, only SSH connection supported.",
"dev_services": [
  "AI_FLOW",
  "SSH"
],
"id": "e1a07296-22a8-4f05-8bc8-e936c8e54101",
"name": "modelbox1.3.0-tensorrt7.1.3-cuda10.2-cudnn8-euler2.9.6",
"resource_categories": [
  "GPU"
],
"service_type": "TRAIN",
"status": "ACTIVE",
"swr_path": "swr.com/atelier/modelarts-modelbox-tensorrt-gpu-x86:1.3.0-20221223142251-b3da6d6",
"tag": "1.3.0-20221223142251-b3da6d6",
"tags": [],
"type": "BUILD_IN",
"update_at": 1648866993030,
"workspace_id": "0"
},
{
  "arch": "aarch64",
  "description": "Ascend operator development. The professional operator development tool MindStudio is preconfigured, only SSH connection supported.",
  "dev_services": [
    "SSH"
  ],
  "id": "e1a07296-22a8-4f05-8bc8-e936c8e54088",
  "name": "mindstudio5.0.rc1-ascendsnt9-cann5.1.0-euler2.8.3-aarch64",
  "resource_categories": [
    "ASCEND"
  ],
  "service_type": "TRAIN",
  "status": "ACTIVE",
  "swr_path": "swr.com/atelier/mindstudio-modelarts-image:5.0.rc1-20230322101430-75f458a",
  "tag": "5.0.rc1-20230322101430-75f458a",
  "tags": [],
  "type": "BUILD_IN",
  "update_at": 1648866993052,
  "workspace_id": "0"
},
{
  "arch": "x86_64",
  "description": "CPU algorithm development and training, preconfigured PySpark 2.4.5 and scala 2.11.12 for code development in local notebook and remote spark cluster including MRS and DLI",
  "dev_services": [
    "NOTEBOOK"
  ],
  "id": "0b2d0728-4c01-11ec-994f-001a7dda7112",
  "name": "spark2.4.5-ubuntu18.04",
  "resource_categories": [
    "CPU"
  ],
  "service_type": "TRAIN",
  "status": "ACTIVE",
  "swr_path": "swr.com/atelier/pyspark_2_4_5:develop-remote-pyspark_2.4.5-py_3.7-cpu-ubuntu_18.04-x86_64-uid1000-20221222203856-fcc979e",
  "tag": "develop-remote-pyspark_2.4.5-py_3.7-cpu-ubuntu_18.04-x86_64-uid1000-20221222203856-fcc979e",
  "tags": [],
  "type": "BUILD_IN",
  "update_at": 1648867218663,
  "workspace_id": "0"
},
{
  "arch": "x86_64",

```

```

"create_at": 1605759392357,
"description": "CPU algorithm development and training, preconfigured with the AI engine
MindSpore-CPU",
"dev_services": [
  "NOTEBOOK",
  "SSH"
],
"id": "65f636a0-56cf-49df-b941-7d2a07ba8c8c",
"name": "mindspore1.2.0-openmpi2.1.1-ubuntu18.04",
"resource_categories": [
  "CPU"
],
"service_type": "TRAIN",
"status": "ACTIVE",
"swr_path": "swr.com/atelier/mindspore_1_2_0:mindspore_1.2.0-py_3.7-ubuntu_18.04-
x86_64-20221118143809-d65d817",
"tag": "mindspore_1.2.0-py_3.7-ubuntu_18.04-x86_64-20221118143809-d65d817",
"tags": [],
"type": "BUILD_IN",
"update_at": 1643166780389,
"workspace_id": "0"
},
{
  "arch": "x86_64",
  "create_at": 1664501979865,
  "description": "",
  "dev_services": [
    "NOTEBOOK",
    "SSH"
  ],
  "id": "df78b3f7-98a4-4616-aef0-71cff4195c9",
  "name": "spark",
  "namespace": "testdli002",
  "origin": "CUSTOMIZE",
  "resource_categories": [
    "CPU"
  ],
  "service_type": "UNKNOWN",
  "size": 1133670676,
  "status": "ACTIVE",
  "swr_path": "swr.com/testdli002/spark:2.4.5.tensorflow",
  "tag": "2.4.5.tensorflow",
  "tags": [],
  "type": "DEDICATED",
  "update_at": 1664501979865,
  "visibility": "PRIVATE",
  "workspace_id": "0"
},
{
  "arch": "x86_64",
  "create_at": 1664513619044,
  "description": "",
  "dev_services": [
    "NOTEBOOK",
    "SSH"
  ],
  "id": "836ab55d-4a02-4dbb-b04f-ece555d642a8",
  "name": "tensorflow2_1_1",
  "namespace": "hwstaff_pub_cbuinfo_ei",
  "origin": "IMAGE_SAVE",
  "resource_categories": [
    "CPU",
    "GPU"
  ],
  "service_type": "COMMON",
  "size": 5094544544,
  "status": "ERROR",
  "status_message": "",
  "swr_path": "swr.com/hwstaff_pub_cbuinfo_ei/tensorflow2_1_1:1.0.0",

```

```

"tag": "1.0.0",
"tags": [],
"type": "DEDICATED",
"update_at": 1664513676950,
"visibility": "PRIVATE",
"workspace_id": "0"
},
{
  "arch": "x86_64",
  "create_at": 1668482562290,
  "description": "test",
  "dev_services": [
    "NOTEBOOK",
    "SSH"
  ],
  "id": "689c81b3-15dd-4500-b63e-1871e24eb391",
  "name": "pytorch_1_8",
  "namespace": "atelier",
  "origin": "CUSTOMIZE",
  "resource_categories": [
    "GPU"
  ],
  "service_type": "UNKNOWN",
  "size": 8285974481,
  "status": "ACTIVE",
  "swr_path": "swr.com/atelier/pytorch_1_8:pytorch_1.8.2-cuda_11.1-py_3.7-ubuntu_18.04-x86_64-20220926104358-041ba2e",
  "tag": "pytorch_1.8.2-cuda_11.1-py_3.7-ubuntu_18.04-x86_64-20220926104358-041ba2e",
  "tags": [],
  "type": "DEDICATED",
  "update_at": 1668482562290,
  "visibility": "PRIVATE",
  "workspace_id": "0"
},
{
  "arch": "aarch64",
  "description": "Ascend+ARM algorithm development and training. MindSpore is preset in the AI engine.",
  "dev_services": [
    "NOTEBOOK",
    "SSH"
  ],
  "id": "f6d0908e-9596-41f9-9843-83089cbdd0de",
  "name": "mindspore1.7.0-cann5.1.0-py3.7-euler2.8.3",
  "namespace": "atelier",
  "resource_categories": [
    "ASCEND"
  ],
  "service_type": "UNKNOWN",
  "status": "ACTIVE",
  "swr_path": "swr.com/atelier/mindspore_1_7_0:mindspore_1.7.0-cann_5.1.0-py_3.7-euler_2.8.3-aarch64-snt9-20220906",
  "tag": "mindspore_1.7.0-cann_5.1.0-py_3.7-euler_2.8.3-aarch64-snt9-20220906",
  "tags": [],
  "type": "BUILD_IN",
  "workspace_id": "0"
},
{
  "arch": "aarch64",
  "description": "Ascend+ARM algorithm development and training. TensorFlow is preset in the AI engine.",
  "dev_services": [
    "NOTEBOOK",
    "SSH"
  ],
  "id": "c5b7507b-ca8d-48d5-a373-fe4b42c66ed8",
  "name": "tensorflow1.15-cann5.1.0-py3.7-euler2.8.3",
  "namespace": "atelier",
  "resource_categories": [

```

```

"ASCEND"
],
"service_type": "UNKNOWN",
"status": "ACTIVE",
"swr_path": "swr.com/atelier/tensorflow_1_15_ascend:tensorflow_1.15-cann_5.1.0-py_3.7-
euler_2.8.3-aarch64-snt9-20220906",
"tag": "tensorflow_1.15-cann_5.1.0-py_3.7-euler_2.8.3-aarch64-snt9-20220906",
"tags": [],
"type": "BUILD_IN",
"workspace_id": "0"
},
{
"arch": "x86_64",
"create_at": 1678261148079,
"description": "",
"dev_services": [
"NOTEBOOK",
"SSH"
],
"id": "e1ab81ef-f452-46b5-9663-6fc1f982f9e9",
"name": "grafana",
"namespace": "hwstaff_pub_cbuinfo_ei",
"origin": "IMAGE_SAVE",
"resource_categories": [
"CPU",
"GPU"
],
"service_type": "COMMON",
"size": 5247805223,
"status": "ACTIVE",
"status_message": "",
"swr_path": "swr.com/hwstaff_pub_cbuinfo_ei/grafana:v1.0",
"tag": "v1.0",
"tags": [],
"type": "DEDICATED",
"update_at": 1678261330238,
"visibility": "PRIVATE",
"workspace_id": "0"
},
{
"arch": "x86_64",
"create_at": 1681973786157,
"dev_services": [
"NOTEBOOK",
"SSH"
],
"id": "a5a43175-30a6-43d2-9596-38bee562f8c0",
"name": "pytorch_1_8",
"namespace": "sdk-test2",
"origin": "CUSTOMIZE",
"resource_categories": [
"CPU",
"GPU"
],
"service_type": "UNKNOWN",
"size": 2308736380,
"status": "ACTIVE",
"swr_path": "swr.com/sdk-test2/pytorch_1_8:v2",
"tag": "v2",
"tags": [],
"type": "DEDICATED",
"update_at": 1681973786157,
"visibility": "PRIVATE",
"workspace_id": "0"
},
{
"arch": "aarch64",
"create_at": 1682670088194,
"description": "Ascend+ARM algorithm development and training. MindSpore is preset in the

```

```

AI engine.",
  "dev_services": [
    "NOTEBOOK",
    "SSH"
  ],
  "id": "75cbf0f2-0a3e-48c9-b2c4-7e78af18d86e",
  "name": "mindspore_1.9.0-cann_6.0.0-py_3.7-euler_2.8.3",
  "namespace": "atelier",
  "resource_categories": [
    "ASCEND"
  ],
  "service_type": "TRAIN",
  "size": 4011027643,
  "status": "ACTIVE",
  "swr_path": "swr.com/atelier/mindspore_1_9_ascend:mindspore_1.9.0-cann_6.0.0-py_3.7-euler_2.8.3-aarch64-snt9-20221116111529",
  "tag": "mindspore_1.9.0-cann_6.0.0-py_3.7-euler_2.8.3-aarch64-snt9-20221116111529",
  "tags": [],
  "type": "BUILD_IN",
  "update_at": 1682670088197,
  "visibility": "PUBLIC",
  "workspace_id": "0"
},
{
  "arch": "x86_64",
  "description": "notebook2.0 gpu",
  "dev_services": [
    "NOTEBOOK",
    "SSH"
  ],
  "id": "e1a07296-22a8-4f05-8bc8-e936c8e54092",
  "name": "notebook2.0-mul-kernel-cpu-cp36",
  "resource_categories": [
    "GPU"
  ],
  "service_type": "TRAIN",
  "status": "ACTIVE",
  "swr_path": "swr.com/atelier/notebook2.0-mul-kernel-gpu-cp36:5.0.1-release-v2-20220505",
  "tag": "5.0.1-release-v2-20220505",
  "tags": [],
  "type": "BUILD_IN",
  "update_at": 1628221753209,
  "workspace_id": "0"
},
{
  "arch": "aarch64",
  "create_at": 1683537880541,
  "description": "Ascend+ARM algorithm development and training. MindSpore is preset in the
AI engine.",
  "dev_services": [
    "NOTEBOOK",
    "SSH"
  ],
  "id": "31ae7ba4-63e6-4fa6-8aeb-cb382953e414",
  "name": "mindspore_1.10.0-cann_6.0.1-py_3.7-euler_2.8.3",
  "namespace": "atelier",
  "resource_categories": [
    "ASCEND"
  ],
  "service_type": "COMMON",
  "size": 4057170552,
  "status": "ACTIVE",
  "swr_path": "swr.com/atelier/mindspore_1_10_ascend:mindspore_1.10.0-cann_6.0.1-py_3.7-euler_2.8.3-aarch64-snt9-20230303173945-815d627",
  "tag": "mindspore_1.10.0-cann_6.0.1-py_3.7-euler_2.8.3-aarch64-snt9-20230303173945-815d627",
  "tags": [],
  "type": "BUILD_IN",
  "update_at": 1683537880548,

```

```

"visibility": "PUBLIC",
"workspace_id": "0"
},
{
  "arch": "x86_64",
  "description": "CPU algorithm development and training, including the MLStudio tool for graphical ML algorithm development, and preconfigured PySpark 2.3.2",
  "dev_services": [
    "NOTEBOOK"
  ],
  "id": "0e5f9a41-c9c2-4d9a-a190-4e1b17a7782f",
  "name": "mlstudio-pyspark2.3.2-ubuntu16.04",
  "resource_categories": [
    "CPU"
  ],
  "service_type": "TRAIN",
  "status": "ACTIVE",
  "swr_path": "swr.com/atelier/notebook2.0-mlstudio-cp36:3.3.1.9",
  "tag": "3.3.1.9",
  "tags": [],
  "type": "BUILD_IN",
  "update_at": 1648867218685,
  "workspace_id": "0"
},
{
  "arch": "x86_64",
  "description": "notebook2.0 cpu base image",
  "dev_services": [
    "NOTEBOOK",
    "SSH"
  ],
  "id": "e1a07296-22a8-4f05-8bc8-e936c8e54090",
  "name": "notebook2.0-mul-kernel-cpu-cp36",
  "resource_categories": [
    "CPU"
  ],
  "service_type": "TRAIN",
  "status": "ACTIVE",
  "swr_path": "swr.com/atelier/notebook2.0-mul-kernel-cpu-cp36:5.0.1-release-v2-20220505",
  "tag": "5.0.1-release-v2-20220505",
  "tags": [],
  "type": "BUILD_IN",
  "update_at": 1628221753345,
  "workspace_id": "0"
},
{
  "arch": "x86_64",
  "description": "GPU algorithm development and training, preconfigured with the AI engine MindSpore-GPU",
  "dev_services": [
    "NOTEBOOK",
    "SSH"
  ],
  "id": "d7fb5355-9045-4deb-94c6-4033e1e62728",
  "name": "mindspore1.2.0-openmpi2.1.1-ubuntu18.04",
  "resource_categories": [
    "GPU"
  ],
  "service_type": "TRAIN",
  "status": "ACTIVE",
  "swr_path": "swr.com/atelier/mindspore_1_2_0:mindspore_1.2.0-py_3.7-ubuntu_18.04-x86_64-20221118143809-d65d817",
  "tag": "mindspore_1.2.0-py_3.7-ubuntu_18.04-x86_64-20221118143809-d65d817",
  "tags": [],
  "type": "BUILD_IN",
  "update_at": 1636963735672,
  "workspace_id": "0"
},
{

```

```

"arch": "x86_64",
"create_at": 1628757809703,
"description": "CPU operations research development, preconfigured with cylv, cbcpy, ortools,
cplex(community).",
"dev_services": [
  "NOTEBOOK",
  "SSH"
],
"id": "b9933af0-3119-4045-a427-5e668327dafd",
"name": "cylv0.91.4-cbcpy2.10-ortools9.0-cplex20.1.0-ubuntu18.04",
"namespace": "atelier",
"resource_categories": [
  "CPU"
],
"service_type": "TRAIN",
"size": 2550402546,
"status": "ACTIVE",
"swr_path": "swr.com/atelier/or_1_0_0:or_1.0.0-py_3.7-ubuntu_18.04-x86_64-
roma-20220812093355-e50493d",
>tag": "or_1.0.0-py_3.7-ubuntu_18.04-x86_64-roma-20220812093355-e50493d",
"tags": [],
"type": "BUILD_IN",
"update_at": 1642836699554,
"workspace_id": "0"
},
{
  "arch": "x86_64",
"description": "CPU algorithm development and training, including the MLStudio tool for
graphical ML algorithm development, and preconfigured PySpark 2.4.5",
"dev_services": [
  "NOTEBOOK"
],
"id": "0b2d0728-4c01-11ec-994f-001a7dda7111",
"name": "mlstudio-pyspark2.4.5-ubuntu18.04",
"resource_categories": [
  "CPU"
],
"service_type": "TRAIN",
"status": "ACTIVE",
"swr_path": "swr.com/atelier/notebook2.0-mlstudio-cp37:5.0.1-mls-20230118153946",
>tag": "5.0.1-mls-20230118153946",
"tags": [],
"type": "BUILD_IN",
"update_at": 1648867218708,
"workspace_id": "0"
},
{
  "arch": "x86_64",
"create_at": 1605759392404,
"description": "GPU algorithm development and training, preconfigured with the AI engine
MindSpore-GPU",
"dev_services": [
  "NOTEBOOK",
  "SSH"
],
"id": "89de30ec-6871-4f22-84af-be37ef28335d",
"name": "mindspore1.2.0-cuda10.1-cudnn7-ubuntu18.04",
"resource_categories": [
  "GPU"
],
"service_type": "TRAIN",
"status": "ACTIVE",
"swr_path": "swr.com/atelier/mindspore_1_2_0:mindspore_1.2.0-py_3.7-cuda_10.1-
ubuntu_18.04-x86_64-20221118143809-d65d817",
>tag": "mindspore_1.2.0-py_3.7-cuda_10.1-ubuntu_18.04-x86_64-20221118143809-d65d817",
"tags": [],
"type": "BUILD_IN",
"update_at": 1648867218639,
"workspace_id": "0"

```



```

},
{
  "arch": "x86_64",
  "description": "description",
  "dev_services": [
    "NOTEBOOK"
  ],
  "id": "88bd7bcd-0c91-45b2-ad0e-ef65553d19c5",
  "name": "dls-feature-engineering",
  "resource_categories": [
    "CPU"
  ],
  "service_type": "TRAIN",
  "status": "ACTIVE",
  "swr_path": "swr.com/atelier/notebook2.0-mul-kernel-dls-feature-engineering-cpu-
py37:3.2.0109",
  "tag": "3.2.0109",
  "tags": [],
  "type": "BUILD_IN",
  "update_at": 1623899358020,
  "workspace_id": "0"
},
{
  "arch": "x86_64",
  "description": "description",
  "dev_services": [
    "NOTEBOOK"
  ],
  "id": "1d1b1327-b243-425b-ad81-2689584c1acc",
  "name": "mls-feature-engineering",
  "resource_categories": [
    "CPU"
  ],
  "service_type": "TRAIN",
  "status": "ACTIVE",
  "swr_path": "swr.com/atelier/notebook2.0-mul-kernel-mls-feature-engineering-cpu-
py37:3.2.0109",
  "tag": "3.2.0109",
  "tags": [],
  "type": "BUILD_IN",
  "update_at": 1623899357995,
  "workspace_id": "0"
},
{
  "arch": "x86_64",
  "description": "MindSpore1.7.0 and MindQuantum0.6.0",
  "dev_services": [
    "NOTEBOOK",
    "SSH"
  ],
  "id": "6592fa02-a40a-4054-a05f-f22215e45ec1",
  "name": "mindquantum0.6.0-mindspore1.7.0-ubuntu18.04",
  "resource_categories": [
    "CPU"
  ],
  "service_type": "TRAIN",
  "status": "ACTIVE",
  "swr_path": "swr.com/atelier/mindspore_1_7_0:mindspore_1.7.0-cpu-py_3.7-ubuntu_18.04-
x86_64-20220727174747-6a4cdd5",
  "tag": "mindspore_1.7.0-cpu-py_3.7-ubuntu_18.04-x86_64-20220727174747-6a4cdd5",
  "tags": [],
  "type": "BUILD_IN",
  "workspace_id": "0"
},
{
  "arch": "x86_64",
  "create_at": 1628757853111,
  "description": "CPU and GPU algorithm development and training, preconfigured with AI
engine ray for reinforcement learning.",

```

```

"dev_services": [
  "NOTEBOOK",
  "SSH"
],
"id": "4233d6f9-c3b5-4cf2-9ee6-2ef565935d6d",
"name": "rlstudio1.0.0-ray1.3.0-cuda10.1-ubuntu18.04",
"namespace": "rl-dev",
"resource_categories": [
  "CPU",
  "GPU"
],
"service_type": "TRAIN",
"size": 4857883146,
"status": "ACTIVE",
"swr_path": "swr.com/atelier/notebook2.0-rl-1.0.0-kernel-cp37:rl-v1220211203",
"tag": "rl-v1220211203",
"tags": [],
"type": "BUILD_IN",
"update_at": 1642836699527,
"workspace_id": "0"
},
{
  "arch": "aarch64",
  "description": "Ascend+ARM algorithm development and training. TensorFlow and MindSpore are preset in the AI engine.",
  "dev_services": [
    "NOTEBOOK",
    "SSH"
  ],
  "id": "59a6e9f5-93c0-44dd-85b0-82f390c5d53b",
  "name": "tensorflow1.15-mindspore1.7.0-cann5.1.0-euler2.8-aarch64",
  "resource_categories": [
    "CPU",
    "ASCEND"
  ],
  "service_type": "TRAIN",
  "status": "ACTIVE",
  "swr_path": "swr.com/atelier/notebook2.0-mul-kernel-arm-ascend-cp37:5.0.1-c81-20220726",
  "tag": "5.0.1-c81-20220726",
  "tags": [],
  "type": "BUILD_IN",
  "update_at": 1640398185602,
  "workspace_id": "0"
},
{
  "arch": "x86_64",
  "description": "CPU general algorithm development and training, preconfigured with AI engine MindSpore1.7.0",
  "dev_services": [
    "NOTEBOOK",
    "SSH"
  ],
  "id": "9d63f4d1-dc09-4873-b669-3483cea777c0",
  "name": "mindspore1.7.0-ubuntu18.04-default",
  "resource_categories": [
    "CPU"
  ],
  "service_type": "TRAIN",
  "status": "ACTIVE",
  "swr_path": "swr.com/atelier/mindspore_1_7_0:mindspore_1.7.0-cpu-py_3.7-ubuntu_18.04-x86_64-20220625205423-5a13f29",
  "tag": "mindspore_1.7.0-cpu-py_3.7-ubuntu_18.04-x86_64-20220625205423-5a13f29",
  "tags": [],
  "type": "BUILD_IN",
  "workspace_id": "0"
},
{
  "arch": "x86_64",
  "description": "CPU and GPU general algorithm development and training, preconfigured with

```

```
AI engine MindSpore1.7.0 and cuda10.1",
  "dev_services": [
    "NOTEBOOK",
    "SSH"
  ],
  "id": "e1a07296-22a8-4f05-8bc8-e936c8e54203",
  "name": "mindspore1.7.0-ubuntu18.04-default",
  "resource_categories": [
    "GPU"
  ],
  "service_type": "TRAIN",
  "status": "ACTIVE",
  "swr_path": "swr.com/atelier/mindspore_1_7_0:mindspore_1.7.0-cuda_10.1-py_3.7-ubuntu_18.04-x86_64-20220625205423-5a13f29",
  "tag": "mindspore_1.7.0-cuda_10.1-py_3.7-ubuntu_18.04-x86_64-20220625205423-5a13f29",
  "tags": [],
  "type": "BUILD_IN",
  "workspace_id": "0"
}
],
"pages": 1,
"size": 200,
"total": 39
}
```

Select the image required for creating a notebook instance based on the **description** and **name** parameters and record its ID. This section provides an example of using TensorFlow to create a notebook instance with an **id** of **e1a07296-22a8-4f05-8bc8-e936c8e54100**.

2. Call the API for **creating a notebook instance** to create an instance.

a. Request body:

URI: POST `https://{ma_endpoint}/v1/{project_id}/notebooks`

Request header:

- X-auth-Token
→ **MIIZmgYJKoZlhvcNAQcCollZizCCGYcCAQExDTALBglghkgBZQMEAgEwgXXXXXX...**
- Content-Type → application/json

Request body:

```
{
  "name": "notebooks_test",
  "feature": "NOTEBOOK",
  "workspace_id": "0",
  "description": "api-test",
  "flavor": "modelarts.vm.cpu.2u",
  "image_id": "e1a07296-22a8-4f05-8bc8-e936c8e54090",
  "volume": {
    "category": "efs",
    "ownership": "managed",
    "capacity": 50
  }
}
```

Set the following parameters based on site requirements:

- *ma_endpoint*: ModelArts endpoint
- *project_id*: user's project ID
- **X-auth-Token**: token obtained in the previous step

- **flavor**: flavor of the notebook instance
 - **image_id**: image ID of the notebook instance
- b. Status code **200** is returned. The response body is as follows:

```
{
  "action_progress": [
    {
      "step": 4,
      "status": "WAITING",
      "description": "Initialize the notebook instance."
    },
    {
      "step": 3,
      "status": "WAITING",
      "description": "Configuring the network."
    },
    {
      "step": 2,
      "status": "WAITING",
      "description": "Prepare the compute resource."
    },
    {
      "step": 1,
      "status": "WAITING",
      "description": "Prepare the storage."
    }
  ],
  "create_at": 1687656452472,
  "description": "api-test",
  "endpoints": [],
  "feature": "NOTEBOOK",
  "flavor": "modelarts.vm.cpu.2u",
  "id": "936bea3e-d3df-435e-8b58-d817283284ae",
  "image": {
    "description": "",
    "id": "e1a07296-22a8-4f05-8bc8-e936c8e54090",
    "name": "notebook2.0-mul-kernel-cpu-cp36",
    "swr_path": "swr.com/atelier/notebook2.0-mul-kernel-cpu-cp36:5.0.1-release-v2-20220505",
    "tag": "5.0.1-release-v2-20220505",
    "type": "BUILD_IN"
  },
  "lease": {
    "create_at": 1687656452470,
    "duration": 3600000,
    "enable": true,
    "type": "TIMING",
    "update_at": 1687656452470
  },
  "name": "notebooks_test",
  "status": "RUNNING",
  "tags": [],
  "token": "3452e0d5-15fe-a20d-18a2-010a574aeaf",
  "update_at": 1687656452588,
  "user_id": "99250e439b33431081xxxxxxxxxa885",
  "workspace_id": "0",
  "billing_items": []
}
```

You can view the notebook instance details in the response. If **status** is **RUNNING**, the notebook instance is successfully created.

3. Call the API for [querying details of a notebook instance](#) to query the instance creation details based on the instance ID.
- a. Request body:
- URI: GET `https://{ma_endpoint}/v1/{project_id}/notebooks/{id}`

Request header: X-auth-Token

→ ***MIIZmgYJKoZlhvcNAQcCollizCCGYcCAQExDTALBgIghkgBZQMEAGFwgXXXXXX...***

Set the bold parameters based on site requirements.

- b. Status code **200** is returned. The response body is as follows:

```
{
  "create_at": 1687656452472,
  "data_volumes": [],
  "description": "api-test",
  "endpoints": [
    {
      "service": "NOTEBOOK",
      "uri": "https://authoring-modelarts.com/936bea3e-d3df-435e-8b58-d817283284ae/lab"
    }
  ],
  "feature": "NOTEBOOK",
  "flavor": "modelarts.vm.cpu.2u",
  "id": "936bea3e-d3df-435e-8b58-d817283284ae",
  "image": {
    "description": "",
    "id": "e1a07296-22a8-4f05-8bc8-e936c8e54090",
    "name": "notebook2.0-mul-kernel-cpu-cp36",
    "swr_path": "swr.com/atelier/notebook2.0-mul-kernel-cpu-cp36:5.0.1-release-v2-20220505",
    "tag": "5.0.1-release-v2-20220505",
    "type": "BUILD_IN"
  },
  "lease": {
    "create_at": 1687656452470,
    "duration": 3627372,
    "enable": true,
    "type": "TIMING",
    "update_at": 1687656479842
  },
  "name": "notebooks_test",
  "status": "RUNNING",
  "tags": [],
  "token": "3452e0d5-15fe-a20d-18a2-010a574aeaaf",
  "update_at": 1687656479880,
  "url": "https://authoring-modelarts.com/936bea3e-d3df-435e-8b58-d817283284ae/lab",
  "user": {
    "domain": {
      "id": "878991804cdc4ba597xxxxxxxx9dd9",
      "name": "hwstaff_pub_CBUInfo_EI"
    },
    "id": "99250e439b33431081xxxxxxxxxa885",
    "name": "xwx1128222"
  },
  "user_id": "99250e439b33431081xxxxxxxxxa885",
  "volume": {
    "category": "EFS",
    "ownership": "MANAGED",
    "mount_path": "/home/ma-user/work/",
    "capacity": 50,
    "read_only": false
  },
  "workspace_id": "0",
  "billing_items": [
    "COMPUTE"
  ]
}
```

4. Call the API for **prolonging a notebook instance** to reset the usage duration of the instance.

- a. Request body:

URI: PATCH `https://{ma_endpoint}/v1/{project_id}/notebooks/{id}/lease`

Request header:

- X-auth-Token
→ ***MIIZmgYJKoZlhvcNAQcCollIZizCCGYcCAQExDTALBglghkgBZQMEAgEwgXXXXXX...***
- Content-Type → application/json

Request body:

```
{
  "duration": 3600000,
  "type": "timing"
}
```

Set the following parameters based on site requirements:

- **duration**: instance running duration, which is calculated based on the instance creation time. If the instance creation time plus the duration is greater than the current time, the system automatically stops the instance.
 - **type**: auto stop type. The default value is **timing**.
- b. Status code **200** is returned, indicating that labeling is successful. The response body is as follows:

```
{
  "create_at": 1687656452470,
  "duration": 4657544,
  "enable": true,
  "type": "TIMING",
  "update_at": 1687657510014
}
```

5. Call the API for **stopping a notebook instance** to stop the instance that is running.

- a. Request body.

URI: POSThttps://{ma_endpoint}/v1/{project_id}/notebooks/{id}/stop

Request header: X-auth-Token

→ ***MIIZmgYJKoZlhvcNAQcCollIZizCCGYcCAQExDTALBglghkgBZQMEAgEwgXXXXXX...***

Set the bold parameters based on site requirements.

- b. Status code **200** is returned. The response body is as follows:

```
{
  "create_at": 1687656452472,
  "data_volumes": [],
  "description": "api-test",
  "endpoints": [
    {
      "service": "NOTEBOOK",
      "uri": "https://authoring-modelarts.com/936bea3e-d3df-435e-8b58-d817283284ae/lab"
    }
  ],
  "feature": "NOTEBOOK",
  "flavor": "modelarts.vm.cpu.2u",
  "id": "936bea3e-d3df-435e-8b58-d817283284ae",
  "image": {
    "description": "",
    "id": "e1a07296-22a8-4f05-8bc8-e936c8e54090",
    "name": "notebook2.0-mul-kernel-cpu-cp36",
    "swr_path": "swr.com/atelier/notebook2.0-mul-kernel-cpu-cp36:5.0.1-release-v2-20220505",
    "tag": "5.0.1-release-v2-20220505",
    "type": "BUILD_IN"
  }
}
```

```

},
"lease": {
  "create_at": 1687656452470,
  "duration": 6199814,
  "enable": true,
  "type": "TIMING",
  "update_at": 1687659052284
},
"name": "notebooks_test",
"status": "STOPPING",
"tags": [],
"token": "3452e0d5-15fe-a20d-18a2-010a574aeaaf",
"update_at": 1687656479880,
"url": "https://authoring-modelarts.com/936bea3e-d3df-435e-8b58-d817283284ae/lab",
"user": {
  "domain": {
    "id": "878991804cdc4ba597xxxxxxxxx9dd9",
    "name": "hwstaff_test"
  },
  "id": "99250e439b33431081xxxxxxxxxa885",
  "name": "test"
},
"user_id": "99250e439b33431081xxxxxxxxxa885",
"volume": {
  "category": "EFS",
  "ownership": "MANAGED",
  "mount_path": "/home/ma-user/work/",
  "capacity": 50,
  "read_only": false
},
"workspace_id": "0",
"billing_items": []
}

```

6. Call the API for **starting a notebook instance** to restart the instance.

a. Request body.

URI: GET `https://{ma_endpoint}/v1/{project_id}/notebooks/{id}/start`

Request header: X-auth-Token

→ ***MIIZmgYJKoZlHvcNAQcCollIZizCCGYcCAQExDTALBgIghkgBZQMEEAgEwgXXXXXX...***

Set the bold parameters based on site requirements.

b. Status code **200** is returned. The response body is as follows:

```

{
  "create_at": 1687656452472,
  "data_volumes": [],
  "description": "api-test",
  "endpoints": [
    {
      "service": "NOTEBOOK",
      "uri": "https://authoring-modelarts.com/936bea3e-d3df-435e-8b58-d817283284ae/lab"
    }
  ],
  "feature": "NOTEBOOK",
  "flavor": "modelarts.vm.cpu.2u",
  "id": "936bea3e-d3df-435e-8b58-d817283284ae",
  "image": {
    "description": "",
    "id": "e1a07296-22a8-4f05-8bc8-e936c8e54090",
    "name": "notebook2.0-mul-kernel-cpu-cp36",
    "swr_path": "swr.com/atelier/notebook2.0-mul-kernel-cpu-cp36:5.0.1-release-v2-20220505",
    "tag": "5.0.1-release-v2-20220505",
    "type": "BUILD_IN"
  },
  "lease": {
    "create_at": 1687656452470,
    "duration": 6540099,

```

```

"enable": true,
"type": "TIMING",
"update_at": 1687659392569
},
"name": "notebooks_test",
"status": "STARTING",
"tags": [],
"token": "6f773860-21d4-9fe8-75c8-a38ea13ebf08",
"update_at": 1687659203630,
"url": "https://authoring-modelarts.com/936bea3e-d3df-435e-8b58-d817283284ae/lab",
"user": {
  "domain": {
    "id": "878991804cdc4ba597xxxxxxxxx9dd9",
    "name": "hwstaff_test"
  },
  "id": "99250e439b33431081xxxxxxxxxa885",
  "name": "test"
},
"user_id": "99250e439b33431081xxxxxxxxxa885",
"volume": {
  "category": "EFS",
  "ownership": "MANAGED",
  "mount_path": "/home/ma-user/work/",
  "capacity": 50,
  "read_only": false
},
"workspace_id": "0",
"billing_items": []
}

```

7. Call the API for **deleting a notebook instance** to delete the instance that is no longer needed.

- a. Request body:

URI: DELETE `https://{ma_endpoint}/v1/{project_id}/notebooks/{id}`

Request header:

- X-auth-Token
→ ***MIIZmgYJKoZlhvcNAQcCollIzCCGYcCAQExDTALBglghkgBZQMEAgEwgXXXXXX...***
- Content-Type → application/json

Set the bold parameters based on site requirements.

- b. Status code **200** is returned, indicating that the instance is successfully deleted.

10.2 Using PyTorch to Create a Training Job (New-Version Training)

This section describes how to train a model by calling ModelArts APIs.

Overview

The process for creating a training job using PyTorch is as follows:

1. , which will be added in a request header for authentication.
2. Call the API for **obtaining general flavors supported by a training job** to obtain the required flavors.

3. Call the API for **obtaining the preset AI frameworks supported by a training job** to view the engines and their versions supported by a training job.
4. Call the API for **creating an algorithm** to create an algorithm and record the algorithm ID.
5. Call the API for **creating a training job** to create a training job using the UUID returned by the created algorithm and record the job ID.
6. Call the API for **querying details about a training job** to query the job status using the job ID.
7. Call the API for **querying the logs of a specified task in a training job (OBS link)** to obtain the OBS path of the training job logs.
8. Call the API for **querying the running metrics of a specified task in a training job** to view detailed metrics of the job.
9. Call the API for **deleting a training job** to delete the job if it is no longer needed.

Prerequisites

- You have obtained the endpoints of ModelArts.
- The following information is available: region where ModelArts is deployed, project ID and name, account name and ID, and username and user ID.
- The training code of PyTorch is available. For example, the startup file **test-pytorch.py** has been stored in the **obs://xxxxxx-job-test-v2/pytorch/fast_example/code/cpu** directory of OBS.
- A data file for the training job is available. For example, a training dataset has been stored in the **obs://xxxxxx-job-test-v2/pytorch/fast_example/data** directory of OBS.
- A path for outputting the training job model has been created, for example, **obs://xxxxxx-job-test-v2/pytorch/fast_example/outputs**.
- A path for outputting the training job logs has been created, for example, **obs://xxxxxx-job-test-v2/pytorch/fast_example/log**.

Procedure

1. Call the API for **obtaining general flavors supported by a training job** to obtain the required flavors.
 - a. Request body:
URI: GET `https://{ma_endpoint}/v2/{project_id}/training-job-flavors?flavor_type=CPU`
Request header: X-Auth-Token
→ ***MIIZmgYJKoZlhvcNAQcCoIIzizCCGYcCAQExDTALBglghkgBZQMEAgEwgXXXXXX...***
Set the following parameters based on site requirements:
 - *ma_endpoint*: ModelArts endpoint
 - *project_id*: user's project ID
 - **X-auth-Token**: token obtained in the previous step

- b. Status code **200** is returned. The response body is as follows:

```

{
  "total_count": 2,
  "flavors": [
    {
      "flavor_id": "modelarts.vm.cpu.2u",
      "flavor_name": "Computing CPU(2U) instance",
      "flavor_type": "CPU",
      "billing": {
        "code": "modelarts.vm.cpu.2u",
        "unit_num": 1
      },
      "flavor_info": {
        "max_num": 1,
        "cpu": {
          "arch": "x86",
          "core_num": 2
        },
        "memory": {
          "size": 8,
          "unit": "GB"
        },
        "disk": {
          "size": 50,
          "unit": "GB"
        }
      }
    },
    {
      "flavor_id": "modelarts.vm.cpu.8u",
      "flavor_name": "Computing CPU(8U) instance",
      "flavor_type": "CPU",
      "billing": {
        "code": "modelarts.vm.cpu.8u",
        "unit_num": 1
      },
      "flavor_info": {
        "max_num": 16,
        "cpu": {
          "arch": "x86",
          "core_num": 8
        },
        "memory": {
          "size": 32,
          "unit": "GB"
        },
        "disk": {
          "size": 50,
          "unit": "GB"
        }
      }
    }
  ]
}

```

- Select and record the flavor required for creating the training job based on the **flavor_id** value. This section uses flavor **modelarts.vm.cpu.8u** with its **max_num** set to **16** as an example.
2. Call the API for **obtaining the preset AI frameworks supported by a training job** to view the engines and their versions supported by a training job.

- a. Request body:

URI: GET https://{ma_endpoint}/v2/{project_id}/job/training-job-engines

Request header:

X-Auth-Token → **MIIZmgYJKoZIHvcNAQcCoIIzizCCGYcCAQExDTALBglghkgBZQM EAgEwgXXXXXX...**

Content-Type → application/json

Set the bold parameters based on site requirements.

- b. Status code **200** is returned. The response body is as follows (only part of the response body is displayed because there are many engines):

```
{
  "total": 28,
  "items": [
    .....
    {
      "engine_id": "mindspore_1.6.0-cann_5.0.3.6-py_3.7-euler_2.8.3-aarch64",
      "engine_name": "Ascend-Powered-Engine",
      "engine_version": "mindspore_1.6.0-cann_5.0.3.6-py_3.7-euler_2.8.3-aarch64",
      "v1_compatible": false,
      "run_user": "1000",
      "image_info": {
        "cpu_image_url": "",
        "gpu_image_url": "atelier/mindspore_1_6_0:train",
        "image_version": "mindspore_1.6.0-cann_5.0.3.6-py_3.7-euler_2.8.3-aarch64-snt9-roma-20211231193205-33131ee"
      }
    },
    .....
    {
      "engine_id": "pytorch_1.8.0-cuda_10.2-py_3.7-ubuntu_18.04-x86_64",
      "engine_name": "PyTorch",
      "engine_version": "pytorch_1.8.0-cuda_10.2-py_3.7-ubuntu_18.04-x86_64",
      "tags": [
        {
          "key": "auto_search",
          "value": "True"
        }
      ],
      "v1_compatible": false,
      "run_user": "1102",
      "image_info": {
        "cpu_image_url": "aip/pytorch_1_8:train",
        "gpu_image_url": "aip/pytorch_1_8:train",
        "image_version": "pytorch_1.8.0-cuda_10.2-py_3.7-ubuntu_18.04-x86_64-20210912152543-1e0838d"
      }
    },
    .....
    {
      "engine_id": "tensorflow_2.1.0-cuda_10.1-py_3.7-ubuntu_18.04-x86_64",
      "engine_name": "TensorFlow",
      "engine_version": "tensorflow_2.1.0-cuda_10.1-py_3.7-ubuntu_18.04-x86_64",
      "tags": [
        {
          "key": "auto_search",
          "value": "True"
        }
      ],
      "v1_compatible": false,
      "run_user": "1102",
      "image_info": {
        "cpu_image_url": "aip/tensorflow_2_1:train",
        "gpu_image_url": "aip/tensorflow_2_1:train",
        "image_version": "tensorflow_2.1.0-cuda_10.1-py_3.7-ubuntu_18.04-x86_64-20210912152543-1e0838d"
      }
    },
    .....
  ]
}
```

```
]
}
```

Select the engine flavor required for creating a training job based on the **engine_name** and **engine_version** fields, and record the field values. This section uses the PyTorch engine as an example to describe how to create a job. In this example, the **engine_name** value is **PyTorch**, and the **engine_version** value is **pytorch_1.8.0-cuda_10.2-py_3.7-ubuntu_18.04-x86_64**.

3. Call the API for **creating an algorithm** to create an algorithm and record the algorithm ID.

- a. Request body:

URI: POST `https://{ma_endpoint}/v2/{project_id}/algorithms`

Request header:

X-Auth-

Token → **MIIZmgYJKoZIHvcNAQcCoIIzizCCGYcCAQExDTALBglghkgBZQM
EAgEwgXXXXXX...**

Content-Type → `application/json`

Set the bold parameters based on site requirements.

Request body:

```
{
  "metadata": {
    "name": "test-pytorch-cpu",
    "description": "test pytorch job in cpu in mode gloo"
  },
  "job_config": {
    "boot_file": "/xxxxxx-job-test-v2/pytorch/fast_example/code/cpu/test-pytorch.py",
    "code_dir": "/xxxxxx-job-test-v2/pytorch/fast_example/code/cpu/",
    "engine": {
      "engine_name": "PyTorch",
      "engine_version": "pytorch_1.8.0-cuda_10.2-py_3.7-ubuntu_18.04-x86_64"
    },
    "inputs": [{
      "name": "data_url",
      "description": "Data source 1"
    }],
    "outputs": [{
      "name": "train_url",
      "description": "Output data 1"
    }],
    "parameters": [{
      "name": "dist",
      "description": "",
      "value": "False",
      "constraint": {
        "editable": true,
        "required": false,
        "sensitive": false,
        "type": "Boolean",
        "valid_range": [],
        "valid_type": "None"
      }
    }],
    {
      "name": "world_size",
      "description": "",
      "value": "1",
      "constraint": {
        "editable": true,
        "required": false,
        "sensitive": false,

```

```

        "type": "Integer",
        "valid_range": [],
        "valid_type": "None"
    }
}
],
"parameters_customization": true
},
"resource_requirements": []
}

```

Set the following parameters based on site requirements:

- **name** and **description** in the **metadata** field indicate the algorithm name and description, respectively.
 - **code_dir** and **boot_file** in the **job_config** field indicate the code directory and code startup file of the algorithm, respectively. The code directory is the level-1 directory of the code startup file.
 - **inputs** and **outputs** in the **job_config** field indicate the input and output of the algorithm, respectively. You can specify **data_url** and **train_url** based on the instance, and parse hyperparameters in the code to specify the local path of the data file required for training and the local output path of the model generated during training.
 - **parameters_customization** in the **job_config** field indicates whether to support custom hyperparameters. Set this parameter to **true**.
 - **parameters** in the **job_config** field indicates the hyperparameters of the algorithm. Set **name** to the hyperparameter name (a maximum of 64 characters, including uppercase letters, lowercase letters, digits, underscores (_), and hyphens (-)). Set **value** to the default value of the hyperparameter. Set **constraint** to the constraints of the hyperparameter. For example, set **type** to **String** (**String**, **Integer**, **Float**, and **Boolean** are supported), set **editable** to **true**, and set **required** to **false**.
 - **engine** in the **job_config** field indicates the engine on which the algorithm depends. Use the **engine_name** and **engine_version** values recorded in [2](#).
- b. Status code **200 OK** is returned, indicating that the algorithm is successfully created. The response body is as follows:

```

{
  "metadata": {
    "id": "01c399ae-8593-4ef5-9e4d-085950aacde1",
    "name": "test-pytorch-cpu",
    "description": "test pytorch job in cpu in mode gloo",
    "create_time": 1641890623262,
    "workspace_id": "0",
    "ai_project": "default-ai-project",
    "user_name": "",
    "domain_id": "0659fbf6de00109b0ff1c01fc037d240",
    "source": "custom",
    "api_version": "",
    "is_valid": true,
    "state": "",
    "size": 4790,
    "tags": null,
    "attr_list": null,
    "version_num": 0,
    "update_time": 0
  }
}

```

```

},
"share_info": {},
"job_config": {
  "code_dir": "/xxxxxx-job-test-v2/pytorch/fast_example/code/cpu/",
  "boot_file": "/xxxxxx-job-test-v2/pytorch/fast_example/code/cpu/test-pytorch.py",
  "parameters": [
    {
      "name": "dist",
      "description": "",
      "i18n_description": null,
      "value": "False",
      "constraint": {
        "type": "Boolean",
        "editable": true,
        "required": false,
        "sensitive": false,
        "valid_type": "None",
        "valid_range": []
      }
    },
    {
      "name": "world_size",
      "description": "",
      "i18n_description": null,
      "value": "1",
      "constraint": {
        "type": "Integer",
        "editable": true,
        "required": false,
        "sensitive": false,
        "valid_type": "None",
        "valid_range": []
      }
    }
  ],
  "parameters_customization": true,
  "inputs": [
    {
      "name": "data_url",
      "description": "Data source 1"
    }
  ],
  "outputs": [
    {
      "name": "train_url",
      "description": "Output data 1"
    }
  ],
  "engine": {
    "engine_id": "pytorch_1.8.0-cuda_10.2-py_3.7-ubuntu_18.04-x86_64",
    "engine_name": "PyTorch",
    "engine_version": "pytorch_1.8.0-cuda_10.2-py_3.7-ubuntu_18.04-x86_64",
    "tags": [
      {
        "key": "auto_search",
        "value": "True"
      }
    ],
    "v1_compatible": false,
    "run_user": "1102",
    "image_info": {
      "cpu_image_url": "aip/pytorch_1_8:train",
      "gpu_image_url": "aip/pytorch_1_8:train",
      "image_version": "pytorch_1.8.0-cuda_10.2-py_3.7-ubuntu_18.04-x86_64-20210912152543-1e0838d"
    }
  },
  "code_tree": {
    "name": "cpu/"
  }
}

```

```

    "children": [
      {
        "name": "test-pytorch.py"
      }
    ]
  },
  "resource_requirements": [],
  "advanced_config": {}
}

```

Record the value of **id** (algorithm ID, 32-bit UUID) in the **metadata** field for subsequent steps.

4. Call the API for **creating a training job** to create a training job using the UUID returned by the created algorithm and record the job ID.

- a. Request body:

URI: POST `https://{ma_endpoint}/v2/{project_id}/training-jobs`

Request header:

- X-Auth-Token
→ ***MIIZmgYJKoZlhvcNAQcCollizCCGYcCAQExDTALBglghkgBZQMEAgEwgXXXXXX...***
- Content-Type → application/json

Set the bold parameters based on site requirements.

Request body:

```

{
  "kind": "job",
  "metadata": {
    "name": "test-pytorch-cpu01",
    "description": "test pytorch work cpu in mode gloo"
  },
  "algorithm": {
    "id": "01c399ae-8593-4ef5-9e4d-085950aacde1",
    "parameters": [{
      "name": "dist",
      "value": "False"
    },
    {
      "name": "world_size",
      "value": "1"
    }
  ],
  "inputs": [{
    "name": "data_url",
    "remote": {
      "obs": {
        "obs_url": "/xxxxxx-job-test-v2/pytorch/fast_example/data/"
      }
    }
  }
  ],
  "outputs": [{
    "name": "train_url",
    "remote": {
      "obs": {
        "obs_url": "/xxxxxx-job-test-v2/pytorch/fast_example/outputs/"
      }
    }
  }
  ],
  "spec": {
    "resource": {
      "flavor_id": "modelarts.vm.cpu.8u",

```

```

        "node_count": 1
    },
    "log_export_path": {
        "obs_url": "/xxxxxx-job-test-v2/pytorch/fast_example/log/"
    }
}
}

```

Set the following parameters based on site requirements:

- Set **kind** to the type of the training job. The default value is **job**.
 - Set **name** and **description** in the **metadata** field to the name and description of the training job.
 - Set **id** in the **algorithm** field to the algorithm ID obtained in [4](#).
 - Set **inputs** and **outputs** in the **algorithm** field to the information about the input and output URLs of the training job. In this example, **obs_url** in **remote** of the **inputs** parameter indicates the OBS path for selecting the training data from the OBS bucket. **obs_url** in **remote** of the **outputs** parameter indicates the OBS path for storing the training output.
 - Set **flavor_id** in the **spec** field to the flavor on which the training job depends. Use the **flavor_id** recorded in [1](#). **node_count** indicates whether to use multi-node training (distributed training). Set it to **1** for a single-node training by default. **log_export_path** specifies the OBS path to which logs are uploaded.
- b. Status code **201 Created** is returned, indicating that the training job has been created. The response body is as follows:

```

{
  "kind": "job",
  "metadata": {
    "id": "66ff6991-fd66-40b6-8101-0829a46d3731",
    "name": "test-pytorch-cpu01",
    "description": "test pytorch work cpu in mode gloo",
    "create_time": 1641892642625,
    "workspace_id": "0",
    "ai_project": "default-ai-project",
    "user_name": "",
    "annotations": {
      "job_template": "Template DL",
      "key_task": "worker"
    }
  },
  "status": {
    "phase": "Creating",
    "secondary_phase": "Creating",
    "duration": 0,
    "start_time": 0,
    "node_count_metrics": null,
    "tasks": [
      "worker-0"
    ]
  },
  "algorithm": {
    "id": "01c399ae-8593-4ef5-9e4d-085950aacde1",
    "name": "test-pytorch-cpu",
    "code_dir": "/xxxxxx-job-test-v2/pytorch/fast_example/code/cpu/",
    "boot_file": "/xxxxxx-job-test-v2/pytorch/fast_example/code/cpu/test-pytorch.py",
    "parameters": [
      {
        "name": "dist",

```



```

        "description": "",
        "i18n_description": null,
        "value": "False",
        "constraint": {
            "type": "Boolean",
            "editable": true,
            "required": false,
            "sensitive": false,
            "valid_type": "None",
            "valid_range": []
        }
    },
    {
        "name": "world_size",
        "description": "",
        "i18n_description": null,
        "value": "1",
        "constraint": {
            "type": "Integer",
            "editable": true,
            "required": false,
            "sensitive": false,
            "valid_type": "None",
            "valid_range": []
        }
    }
],
"parameters_customization": true,
"inputs": [
    {
        "name": "data_url",
        "description": "Data source 1",
        "local_dir": "/home/ma-user/modelarts/inputs/data_url_0",
        "remote": {
            "obs": {
                "obs_url": "/xxxxxx-job-test-v2/pytorch/fast_example/data/"
            }
        }
    }
],
"outputs": [
    {
        "name": "train_url",
        "description": "Output data 1",
        "local_dir": "/home/ma-user/modelarts/outputs/train_url_0",
        "remote": {
            "obs": {
                "obs_url": "/xxxxxx-job-test-v2/pytorch/fast_example/outputs/"
            }
        },
        "mode": "upload_periodically",
        "period": 30
    }
],
"engine": {
    "engine_id": "pytorch_1.8.0-cuda_10.2-py_3.7-ubuntu_18.04-x86_64",
    "engine_name": "PyTorch",
    "engine_version": "pytorch_1.8.0-cuda_10.2-py_3.7-ubuntu_18.04-x86_64",
    "usage": "training",
    "support_groups": "public",
    "tags": [
        {
            "key": "auto_search",
            "value": "True"
        }
    ]
},
"v1_compatible": false,
"run_user": "1102"
}

```

```

    },
    "spec": {
      "resource": {
        "flavor_id": "modelarts.vm.cpu.8u",
        "flavor_name": "Computing CPU(8U) instance",
        "node_count": 1,
        "flavor_detail": {
          "flavor_type": "CPU",
          "billing": {
            "code": "modelarts.vm.cpu.8u",
            "unit_num": 1
          },
          "flavor_info": {
            "cpu": {
              "arch": "x86",
              "core_num": 8
            },
            "memory": {
              "size": 32,
              "unit": "GB"
            },
            "disk": {
              "size": 50,
              "unit": "GB"
            }
          }
        }
      },
      "log_export_path": {
        "obs_url": "/xxxxxx-job-test-v2/pytorch/fast_example/log/"
      },
      "is_hosted_log": true
    }
  }
}

```

- Record the **id** value (training job ID) in the **metadata** field for subsequent steps.
 - **phase** and **secondary_phase** under **Status** indicate the status and next status of the training job, respectively. In the example, **Creating** indicates that the training job is being created.
5. Call the API for [querying details about a training job](#) to query the job status using the job ID.

a. Request body:

URI: GET `https://{ma_endpoint}/v2/{project_id}/training-jobs/{training_job_id}`

Request header: X-Auth-Token

→ `MIIZmgYJKoZlhvcNAQcCoIIzizCCGYcCAQExDTALBgIghkgBZQMEAgEwgXXXXXX...`

Set the following parameter based on site requirements:

Set **training_job_id** to the training job ID recorded in [5](#).

b. Status code **200 OK** is returned. The response body is as follows:

```

{
  "kind": "job",
  "metadata": {
    "id": "66ff6991-fd66-40b6-8101-0829a46d3731",
    "name": "test-pytorch-cpu01",
    "description": "test pytorch work cpu in mode gloo",
    "create_time": 1641892642625,
    "workspace_id": "0",
    "ai_project": "default-ai-project",
    "user_name": "hwstaff_z00424192",
  }
}

```

```

    "annotations": {
      "job_template": "Template DL",
      "key_task": "worker"
    }
  },
  "status": {
    "phase": "Running",
    "secondary_phase": "Running",
    "duration": 268000,
    "start_time": 1641892655000,
    "node_count_metrics": [
      [
        1641892645000,
        0
      ],
      [
        1641892654000,
        0
      ],
      [
        1641892655000,
        1
      ],
      [
        1641892922000,
        1
      ],
      [
        1641892923000,
        1
      ]
    ],
    "tasks": [
      "worker-0"
    ]
  }
},
"algorithm": {
  "id": "01c399ae-8593-4ef5-9e4d-085950aacde1",
  "name": "test-pytorch-cpu",
  "code_dir": "/xxxxxx-job-test-v2/pytorch/fast_example/code/cpu/",
  "boot_file": "/xxxxxx-job-test-v2/pytorch/fast_example/code/cpu/test-pytorch.py",
  "parameters": [
    {
      "name": "dist",
      "description": "",
      "i18n_description": null,
      "value": "False",
      "constraint": {
        "type": "Boolean",
        "editable": true,
        "required": false,
        "sensitive": false,
        "valid_type": "None",
        "valid_range": []
      }
    },
    {
      "name": "world_size",
      "description": "",
      "i18n_description": null,
      "value": "1",
      "constraint": {
        "type": "Integer",
        "editable": true,
        "required": false,
        "sensitive": false,
        "valid_type": "None",
        "valid_range": []
      }
    }
  ]
}

```

```

    }
  ],
  "parameters_customization": true,
  "inputs": [
    {
      "name": "data_url",
      "description": "Data source 1",
      "local_dir": "/home/ma-user/modelarts/inputs/data_url_0",
      "remote": {
        "obs": {
          "obs_url": "/xxxxxx-job-test-v2/pytorch/fast_example/data/"
        }
      }
    }
  ],
  "outputs": [
    {
      "name": "train_url",
      "description": "Output data 1",
      "local_dir": "/home/ma-user/modelarts/outputs/train_url_0",
      "remote": {
        "obs": {
          "obs_url": "/xxxxxx-job-test-v2/pytorch/fast_example/outputs/"
        }
      },
      "mode": "upload_periodically",
      "period": 30
    }
  ],
  "engine": {
    "engine_id": "pytorch_1.8.0-cuda_10.2-py_3.7-ubuntu_18.04-x86_64",
    "engine_name": "PyTorch",
    "engine_version": "pytorch_1.8.0-cuda_10.2-py_3.7-ubuntu_18.04-x86_64",
    "usage": "training",
    "support_groups": "public",
    "tags": [
      {
        "key": "auto_search",
        "value": "True"
      }
    ]
  },
  "v1_compatible": false,
  "run_user": "1102"
}
},
"spec": {
  "resource": {
    "flavor_id": "modelarts.vm.cpu.8u",
    "flavor_name": "Computing CPU(8U) instance",
    "node_count": 1,
    "flavor_detail": {
      "flavor_type": "CPU",
      "billing": {
        "code": "modelarts.vm.cpu.8u",
        "unit_num": 1
      }
    },
    "flavor_info": {
      "cpu": {
        "arch": "x86",
        "core_num": 8
      },
      "memory": {
        "size": 32,
        "unit": "GB"
      },
      "disk": {
        "size": 50,
        "unit": "GB"
      }
    }
  }
}

```

```

    }
  },
  "log_export_path": {
    "obs_url": "/xxxxxx-job-test-v2/pytorch/fast_example/log/"
  },
  "is_hosted_log": true
}
}

```

You can learn about the version details of the training job based on the response. The **status** value is **Running**, indicating that the training job is running.

6. Call the API for [querying the logs of a specified task in a training job \(OBS link\)](#) to obtain the OBS path of the training job logs.

- a. Request body:

URI format: GET https://{ma_endpoint}/v2/{project_id}/training-jobs/{training_job_id}/tasks/{task_id}/logs/url

Request header:

X-Auth-

Token → **MIIZmgYJKoZlhvcNAQcCollIZizCCGYcCAQExDTALBglghkgBZQM EAgEwgXXXXXX...**

Content-Type → **text/plain**

Set the following parameters based on site requirements:

- **task_id** indicates the name of the training job. Generally, set it to **work-0**.
- **Content-Type** can be set either to **text/plain** or **application/octet-stream**. **text/plain** indicates that a temporary OBS preview URL is returned. **application/octet-stream** indicates that a temporary OBS download URL is returned.

- b. Status code **200 OK** is returned. The response body is as follows:

```

{
  "obs_url": "https://modelarts-training-log-xxxxxx.obs.xxxxxx.com:443/66ff6991-fd66-40b6-8101-0829a46d3731/worker-0/modelarts-job-66ff6991-fd66-40b6-8101-0829a46d3731-worker-0.log?AWSAccessKeyId=GFGTBKOZEND83QEMZMV&Expires=1641896599&Signature=BedFZHEU1oCmqlI912UL9mXlhkg%3D"
}

```

The returned field indicates the OBS path of logs. You can copy the value to the browser to view the result.

7. Call the API for [querying the running metrics of a specified task in a training job](#) to view detailed metrics of the job.

- a. Request body:

URI format: GET https://{ma_endpoint}/v2/{project_id}/training-jobs/{training_job_id}/metrics/{task_id}

Request header: X-Auth-Token

→ **MIIZmgYJKoZlhvcNAQcCollIZizCCGYcCAQExDTALBglghkgBZQM EAgEwgXXXXXX...**

Set the bold parameters based on site requirements.

- b. Status code **200 OK** is returned. The response body is as follows:

```

{
  "metrics": [

```

```
{
  "metric": "cpuUsage",
  "value": [
    -1,
    -1,
    28.622,
    35.053,
    39.988,
    40.069,
    40.082,
    40.094
  ]
},
{
  "metric": "memUsage",
  "value": [
    -1,
    -1,
    0.544,
    0.641,
    0.736,
    0.737,
    0.738,
    0.739
  ]
},
{
  "metric": "npuUtil",
  "value": [
    -1,
    -1,
    -1,
    -1,
    -1,
    -1,
    -1,
    -1
  ]
},
{
  "metric": "npuMemUsage",
  "value": [
    -1,
    -1,
    -1,
    -1,
    -1,
    -1,
    -1,
    -1
  ]
},
{
  "metric": "gpuUtil",
  "value": [
    -1,
    -1,
    -1,
    -1,
    -1,
    -1,
    -1,
    -1
  ]
},
{
  "metric": "gpuMemUsage",
  "value": [
    -1,
```

```
-1,  
-1,  
-1,  
-1,  
-1,  
-1,  
-1,  
-1  
  ]  
  }  
 ]  
 }
```

You can view the metrics such as the CPU usage.

8. Call the API for **deleting a training job** to delete the job if it is no longer needed.
 - a. Request body:
URI: DELETE `https://{ma_endpoint}/v2/{project_id}/training-jobs/{training_job_id}`
Request header: X-Auth-Token
→ **MIIZmgYJKoZlhvcNAQcCollizCCGYcCAQExDTALBglghkgBZQMEAgEwgXXXXXX...**
Set the bold parameters based on site requirements.
 - b. Status code **202 No Content** is returned, indicating that the job is successfully deleted.

10.3 Managing ModelArts Authorization

This section describes how to manage agency authorization by calling ModelArts APIs.

Overview

The process of managing ModelArts authorization is as follows:

1. , which will be added in a request header for authentication.
2. Call the API for **creating a ModelArts agency** to create an agency for ModelArts-dependent services, such as OBS and SWR.
3. Call the API for **configuring authorization** to configure ModelArts authorization. The administrator can use this API to set an agency for IAM users and set the access key of the current user.

NOTE

ModelArts functions, such as data management, training management, development environment, and real-time services, can be used only after being authorized.

4. Call the API for **obtaining the authorization list** to view the authorization.
5. Call the API for **deleting authorization** to delete the authorization of a specified user or all users.

Prerequisites

- You have obtained the **ModelArts endpoint**.
- The following information is available: region where ModelArts is deployed, project name and ID, account name and ID, and username and ID.

Procedure

1. Call the API for [creating a ModelArts agency](#) to create an agency for ModelArts-dependent services, such as OBS, SWR, and IEF.

- a. Request body:

URI: POST `https://{endpoint}/v2/{project_id}/agency`

Request header:

- X-auth-Token
→ *MIIZmgYJKoZlHvcNAQcCollZizCCGYcCAQExDTALBglghkgBZQMEAgEwgXXXXXX...*
- Content-Type → application/json

Request body:

```
{
  "agency_name_suffix" : "iam-user01"
}
```

Set the italic parameters based on site requirements.

- *endpoint*: ModelArts endpoint
- *project_id*: Your project ID
- **X-auth-Token**: Token obtained in the previous step
- **agency_name_suffix**: Customized suffix of the agency name

- b. Response body with status code **200 OK** returned (indicating that **ma_agency_iam-user01** has been created):

```
{
  "agency_name": "ma_agency_iam-user01"
}
```

2. Call the API for [configuring authorization](#) to configure ModelArts authorization. The administrator can use this API to set an agency for IAM users and set the access key of the current user.

- a. Request body:

URI: POST `https://{endpoint}/v2/{project_id}/authorizations`

Request header:

- X-auth-Token
→ *MIIZmgYJKoZlHvcNAQcCollZizCCGYcCAQExDTALBglghkgBZQMEAgEwgXXXXXX...*
- Content-Type → application/json

Request body:

```
{
  "user_id": "****af917080f5d21f55c018ba19****",
  "type": "agency",
  "content": "ma_agency_iam-user01"
}
```

Set the italic parameters based on site requirements. Set **user_id** to the IAM user ID and **content** to the ModelArts agency created in the previous step.

- b. Response body with status code **200 OK** returned (indicating that the authorization configuration is complete):

```
{
  "result": true
}
```

3. Call the API for **obtaining the authorization list** to view the authorization.

- a. Request body:

URI: GET `https://{endpoint}/v2/{project_id}/authorizations`

Request header: X-auth-Token

→ *MIIZmgYJKoZlhvcNAQcCollZizCCGYcCAQExDTALBglghkgBZQMEAgEwgXXXXXX...*

Set the italic parameters based on site requirements.

- b. Response body with status code **200 OK** returned:

```
{
  "auth": [
    {
      "create_time": 1622804433221,
      "user_id": "all-users",
      "user_name": "all-users",
      "type": "agency",
      "content": "modelarts_agency"
    },
    {
      "create_time": 1625457065365,
      "user_id": "****af917080f5d21f55c018ba19****",
      "user_name": null,
      "type": "agency",
      "content": "ma_agency_iam-user01"
    }
  ],
  "total_count": 2
}
```

Obtain the authorization information based on the response.

4. Call the API for **deleting authorization** to delete the authorization of a specified user or all users.

- a. Request body:

URI: DELETE `https://{endpoint}/v2/{project_id}/authorizations?user_id=****d80fb058844ae8b82aa66d9fe****`

Request header: X-auth-Token

→ *MIIZmgYJKoZlhvcNAQcCollZizCCGYcCAQExDTALBglghkgBZQMEAgEwgXXXXXX...*

Set the italic parameters based on site requirements. Set

*****d80fb058844ae8b82aa66d9fe***** to the IAM user ID of the specified user.

- b. Response body with status code **200 OK** returned (indicating that the authorization has been deleted):

```
{
  "result": true
}
```

- c. If **user_id** is set to **all-users**, the authorization of all IAM users will be deleted. Response body with status code **200 OK** returned (indicating that the authorization has been deleted):

```
{
  "result": true,
```

```
"success_message": "Delete all-users auth info successfully!"  
}
```

11 Common Parameters

11.1 Status Code

[Table 11-1](#) describes the status codes.

Table 11-1 Status codes

Status Code	Code	Status Code Description
100	Continue	The client continues sending the request. This provisional response informs the client that 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 of HTTP.
200	OK	The request has been fulfilled.
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	Non-authoritative information. The request is successful.

Status Code	Code	Status Code Description
204	NoContent	The request has been fulfilled, but the HTTP response does not contain a response body. The status code is returned in response to an HTTP OPTIONS request.
205	Reset Content	The server has fulfilled the request, but the requester is required to reset the content.
206	Partial Content	The server has successfully processed 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	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	The request is invalid. Do not retry the request before modification.

Status Code	Code	Status Code Description
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 request has been rejected. The server has received and understood the request; yet it refused to respond, because the request is set to deny access. Do not retry the request before modification.
404	NotFound	The requested resource cannot be found. Do not retry the request before modification.
405	MethodNotAllowed	The request contains one or more methods not supported for the resource. Do not retry the request before modification.
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 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 time later.
409	Conflict	The request could not be processed due to a conflict with the current state of the resource. This status code indicates that the resource that the client attempts to create already exists, or the requested update failed due to a conflict.

Status Code	Code	Status Code Description
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.
422	UnprocessableEntity	The request is well-formed but is unable to be processed due to semantic errors.
429	TooManyRequests	The client has sent excessive number of requests to the server within a given time (exceeding the limit on the access frequency of the client), or the server has received an excessive number of requests within a given time (beyond its processing capability). In this case, the client should resend the request after the time specified in the Retry-After header of the response has elapsed.

Status Code	Code	Status Code Description
500	InternalServerError	The server is able to receive but unable to understand the request.
501	Not Implemented	The server does not support the requested function.
502	Bad Gateway	The server acting as a gateway or proxy has received an invalid request from a remote server.
503	ServiceUnavailable	The requested service is invalid. Do not retry the request before modification.
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.

11.2 Error Codes

Status Code	Error Codes	Error Message	Description	Solution
100	ModelArts.0116	The expected {0} exceeds the quota limit.	The expected {0} exceeds the quota limit.	Contact the administrator to increase the quota.
100	ModelArts.6768	SSH configuration is missing, target image {0} only support SSH dev service.	SSH configuration is missing, target image {0} only support SSH dev service.	Select a proper image.
200	ModelArts.4606	The sample has been reviewed.	The sample has been reviewed.	Proceed with follow-up operations.
200	ModelArts.4900	Training job prepared.	The training job has been prepared.	Proceed with follow-up operations.

Status Code	Error Codes	Error Message	Description	Solution
200	ModelArts.4902	Exporting labels.	Labels are being exported.	Wait until labels are exported.
200	ModelArts.4904	Labels exported.	Labels have been exported.	Proceed with follow-up operations.
200	ModelArts.4910	Model import task submitted.	The model import task has been submitted.	Proceed with follow-up operations.
200	ModelArts.4912	Model import task executed.	The model import task has been executed.	Proceed with follow-up operations.
200	ModelArts.4914	Auto labeling task submitted.	The auto labeling task has been submitted.	Wait until the task is complete.
200	ModelArts.4916	Auto labeling task executed.	The auto labeling task has been executed.	Proceed with follow-up operations.
200	ModelArts.4918	Import task submitted.	The import task has been submitted.	Wait until the task is complete.
200	ModelArts.4920	Import task executed.	The import task has been executed.	Proceed with follow-up operations.
200	ModelArts.4926	Collecting hard examples submitted.	Collecting hard examples has been submitted.	Wait until the task is complete.
200	ModelArts.4950	Auto inference task submitted.	The auto inference task has been submitted.	Wait until the task is complete.
200	ModelArts.4952	Auto inference task executed.	The auto inference task has been executed.	Proceed with follow-up operations.

Status Code	Error Codes	Error Message	Description	Solution
200	ModelArts.4960	Auto grouping task submitted.	The auto grouping task has been submitted.	Wait until the task is complete.
200	ModelArts.4962	Auto grouping task executed.	The auto grouping task has been executed.	Proceed with follow-up operations.
200	ModelArts.4964	Auto grouping task is already running.	The auto grouping task is already running.	Please wait.
204	ModelArts.2776	NotReadyToFetchLog	not ready for fetch log, wait for training to return data	Please try again later.
204	ModelArts.2777	NotReadyToFetchMetric	not ready for fetch metric, wait for training to return data	Please try again later.
400	ModelArts.0101	Invalid Argument.	Invalid parameter.	Enter the correct parameter as prompted.
400	ModelArts.0102	The workspace does not exist.	The workspace does not exist.	Check the current workspace.
400	ModelArts.0104	Parameter error.	Parameter error.	Check the parameter settings.
400	ModelArts.0107	The values of the request parameters ({0},{1}) are invalid.	Invalid parameter values ({0}, {1}).	Check whether the parameter values are valid.
400	ModelArts.0113	Create Notebook failed. Quota: {} is exhausted.	Failed to create the notebook instance due to the exhausted quota.	Contact the administrator for a higher quota.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.0118	QuotaInsufficient	The expected %s exceeds the quota limit.	Release idle resources, or apply for more quotas
400	ModelArts.0204	Token must contain projectId info.	No project ID included in the token.	Ensure that the project token is used.
400	ModelArts.0210	Invalid project ID.	The project ID is invalid.	Check the project ID.
400	ModelArts.0420	Failed to query agency.	Failed to query agency.	System error. Contact technical support.
400	ModelArts.0421	Failed to create agency.	Failed to create agency.	System error. Contact technical support.
400	ModelArts.0422	Failed to query users.	Failed to query users.	System error. Contact technical support.
400	ModelArts.0423	Disagree with service statement.	Disagree with service statement.	System error. Contact technical support.
400	ModelArts.0424	User authorization information already exists.	User authorization information already exists.	Authorization already exists. To update the authorization, delete the original one and add new authorization.
400	ModelArts.0425	Failed to add user authorization information.	Failed to add user authorization information.	System error. Contact technical support.
400	ModelArts.0426	Failed to delete user authorization information.	Failed to delete user authorization information.	System error. Contact technical support.
400	ModelArts.0427	Failed to get user authorization information.	Failed to get user authorization information.	System error. Contact technical support.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.0428	Failed to update agency.	Failed to update agency.	System error. Contact technical support.
400	ModelArts.0429	The authorization does not exist. Check whether the user_id is correct.	The authorization does not exist. Check whether the user_id is correct.	Check whether the user_id is correct.
400	ModelArts.0430	Failed to query agency grants.	Failed to query agency grants.	System error. Contact technical support.
400	ModelArts.0431	Failed to query agency quotas.	Failed to query agency quotas.	System error. Contact technical support.
400	ModelArts.1002	OBS operation failed.	OBS operation failed.	Ensure that the OBS service is available.
400	ModelArts.1120	Parameter verification failed.	Parameter verification failed.	System error. Contact technical support.
400	ModelArts.2750	DuplicateEvaluationName	Duplicate evaluation job (name %s).	Change evaluation job name.
400	ModelArts.2751	DuplicateTrainingJobName	Duplicate training job (name %s).	Change training job name.
400	ModelArts.2752	DuplicateAlgorithmNameAndVersion	An algorithm with the same name and version already exists.	Check the validity of the algorithm information in the request.
400	ModelArts.2753	InputsOutputsMessageError	input and output channel information is not standardized.	Check the validity of the input and output channel information in the request.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.2754	ModelartsHostedNumError	Only one output channel can be configured as ModelArts hosted output.	Check the validity of the output path information in the request.
400	ModelArts.2755	AiAlgorithmNotFound	algorithm not found.	Check whether the training project information in the request is valid.
400	ModelArts.2756	HasSameNameWithMarketAlgorithm	Duplicate name with subscribed algorithm.	Check whether the training project information in the request is valid.
400	ModelArts.2757	CodeDirError	code directory path error.	Check whether the training project information in the request is valid.
400	ModelArts.2758	BootFileDirError	code startup file path error.	Check whether the training project information in the request is valid.
400	ModelArts.2759	TooManyCodeFiles	Insufficient quota, too many code files, support uploading up to %s files.	Please check the number of code files.
400	ModelArts.2760	CodeFolderTooDeep	Insufficient quota, too many levels of code directories, supporting up to %s directories	Please check the code directory depth.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.2761	CodeFilesSize Toolarge	Insufficient quota, the code file takes up too much storage space, supporting up to %s GB	Please check the code directory depth.
400	ModelArts.2762	DuplicateFlavorCode	Duplicate flavor code (%s).	Change flavor code message.
400	ModelArts.2763	DoNotHaveFlavor	The selected flavor is invalid.	Check the validity of the flavor in the request.
400	ModelArts.2764	QueryParams Error	Invalid query parameter (%s).	Check the validity of the parameter in the request.
400	ModelArts.2765	StringParameterError	character variables must be within the preset parameters.	Please enter the variable value based on the preset range.
400	ModelArts.2766	DoNotHaveDefaultValue	Missing default value.	Please enter the default value.
400	ModelArts.2767	ParameterKey Repeat	parameter key repeat.	Please rename the parameter name.
400	ModelArts.2768	DefaultValueTypeError	Default value does not match parameter type.	Please check the default parameter type.
400	ModelArts.2769	DefaultValuesNotInRange	default value is not in the setting range.	Please ensure that the default value is within the range.
400	ModelArts.2770	RangeError	Parameter range error.	Please enter another value range.
400	ModelArts.2771	NotHavePermission	No permission to modify.	Please check the user permission.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.2772	OriginalAlgorithmCodeInfoError	Current algorithm creation failed. To ensure the availability of the algorithm, please reselect the code directory and startup file.	Please reselect the code directory and startup file.
400	ModelArts.2773	GetObsObjectContextError	get obs object content error.	Please check the file permission.
400	ModelArts.2774	ParseParameterError	parse parameter error.	Hyperparameters are parsed as beta. If the parsing result is improper, manually enter the value.
400	ModelArts.2775	TrainingJobNotFound	Training job (ID: %s) not found.	Check whether the training job information in the request is valid.
400	ModelArts.2778	EvalJobNotFound	Model evaluation job (ID: %s) not found.	Check whether the job ID in the request is valid.
400	ModelArts.2779	EvalVerNotFound	Model evaluation job version (ID: %s) not found.	Check whether the evaluation job version information in the request is valid.
400	ModelArts.2780	EngineNotFound	AI engine(id:%s) not found.	Check whether the engine informer in the request is valid.
400	ModelArts.2781	FlavorNotFound	Flavor (%s) not found.	Check whether the flavor in the request is valid.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.2782	DatasetVersionNotFound	Dataset version (ID: %s) not found.	Check whether the dataset version ID in the request is valid.
400	ModelArts.2783	DatasetNotFound	Dataset (ID: %s) not found.	Check whether the dataset ID in the request is valid.
400	ModelArts.2787	InvalidJsonBody	Invalid JSON request body.	Check the request body.
400	ModelArts.2788	InvalidParameter	Invalid parameter(%s).	Check the request parameter.
400	ModelArts.2789	UnsupportedFilterBy	Unsupported filtering item.	Check whether the filtering item is valid.
400	ModelArts.2790	UnsupportedSortBy	Unsupported sorting item.	Check whether the sorting item is valid.
400	ModelArts.2791	InvalidObsPath	Invalid obs path: %s, reason: %s.	Check the obs path.
400	ModelArts.2792	BootFileNotBelongToPackage	The boot_file does not belong to the package_id directory.	Check whether boot_file and package_id are set correctly.
400	ModelArts.2793	FlavorOutOfRange	The number of flavor instances (%d) exceeds the maximum (%d).	Check the flavor.
400	ModelArts.2794	InvalidModelURL	Invalid model URL.	Check the model URL.
400	ModelArts.2795	GetModel	Failed to obtain the model: %s	Check whether the model is valid.
400	ModelArts.2797	EvalResultsNotPrepared	The evaluation results is not generated.	Wait for the evaluation results generated.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.2798	Unsupported Model	Unsupported model id:%s	Check whether the model id is valid.
400	ModelArts.2801	FederationNot Found	federation ID (%s) not found.	Check whether the federation id is valid.
400	ModelArts.2802	InvalidCodeDir	Invalid codeDir path: %s.	Check the codeDir path.
400	ModelArts.2804	InsufficientPermissionToAuthorize	The current user does not have sufficient permission to authorize all sub-users of the tenant.	Please confirm user's iam permission
400	ModelArts.2805	FederationInviteUserError	The participant to be invited contains the IAM user ID of the current account(%s).	Ensure that the user list does not contain the IAM user ID of the current account.
400	ModelArts.2806	MustHaveCmdAndImage	Custom image jobs command and engine.url cannot be empty.	Check whether the command and engine.url is valid.
400	ModelArts.2807	UnsupportedConditionKey	Unsupported condition key (%s)	Check whether the key of search is valid.
400	ModelArts.2808	UnsupportedConditionOperator	Unsupported condition operator (%s)	Check whether the operator of search is valid.
400	ModelArts.2809	UnsupportedConditionValue	Unsupported condition value (%s)	Check whether the value of search is valid.
400	ModelArts.2810	SWRQueryImageInfoError	Custom image query failure	Check whether the image_url is valid.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.2811	SWRCreateShareError	Custom image share-create failure	Check whether the image_url is valid.
400	ModelArts.2812	Unsupported ActionType	Unsupported action type:%s	Check whether the action type is valid.
400	ModelArts.2814	AlgorithmNot Match	Job algorithm configuration(%s) does not match algorithm management configuration.	Please check the parameter is valid.
400	ModelArts.2815	ExceedQuotaLimit	Exceed quota limit of resource(%s).	Please release the resources in use or apply for an expanded quota.
400	ModelArts.2816	ParamRangeError	The parameter(%s) exceeds the upper limit %s.	Please enter another value range.
400	ModelArts.2817	AiAlgorithmShareError	algorithm(%s) is not ready to share.	Please wait for the algorithm to complete creation.
400	ModelArts.2818	LogFileNotGenerated	the log file is not generated	Please view the log after the log is generated.
400	ModelArts.2819	LogFileHasCleared	the log file has been cleared	Please select obs url to upload your logs when creating training jobs.
400	ModelArts.2820	LogPathNotFound	the log storage path cannot be found	please use old API to view the log
400	ModelArts.2821	DuplicateAlgorithmName	Duplicate algorithm (name %s).	The algorithm name already exists, Please change algorithm name.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.2822	LogExportPathNotSet	The log export path is not set.	Please set the log export path.
400	ModelArts.2823	DomainIdNotMatched	Policy doesn't allow to be performed. domain id %s, config domain id %s.	Please check domain id in configfile and Request Header.
400	ModelArts.2824	SearchIndicatorsError	The search indicators is duplicate.	Check whether the search indicators is duplicate.
400	ModelArts.2825	ParamTypeError	The parameter (%s) value type error.	Ensure that the parameter value type is correct.
400	ModelArts.2826	SearchParamsError	The search parameters error.	Please enter another search parameter.
400	ModelArts.2827	FailureAnalysisNotPossible	Unable to perform failure analysis on jobs that did not fail.	Please perform a failure analysis on the failed training job.
400	ModelArts.2828	FailureAnalysisFailed	Failed to get the reason for training job failure.	Unknown failure. Contact technical support.
400	ModelArts.2829	DuplicateEngineId	Duplicate training engine (name %s).	Change training engine name.
400	ModelArts.2830	MustHaveEngineId	Custom image jobs engine_id cannot be empty.	Check whether the engine_id is valid.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.2831	FeedbackFailureAnalysisNotPossible	Failure to provide feedback on failure results.	Please give feedback on the failed job for which the reason has been analyzed.
400	ModelArts.2832	EngineNeedUpgrade	The AI engine version which you are using is too old, it has been removed.	Please use the latest version of AI engine.
400	ModelArts.2833	ExceedFreeFlavorQuota	The job quantity exceeds the maximum number (%d) of jobs that can be created by a single user of the free flavor.	Please delete other free flavor job.
400	ModelArts.2834	UnsupportedRemoteDataType	Does not support data type (%s).	Change to a supported data type.
400	ModelArts.2835	GetSizeOfChannelTimeout	Get size of input channel time out.	Please get size of folder in local file system or obs service.
400	ModelArts.2837	FrozenAccount	The account has been restricted or frozen.	Please check the account.
400	ModelArts.2838	WorkspacePolicyLimit	Policy doesn't allow (%s) to be performed	Check the configuration of the fine-grained policy in IAM
400	ModelArts.2839	InvalidMindSporeConfig	Invalid MindSpore advanced configuration (%s).	Check the validity of the information in the request.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.2840	UnsupportedFlavor	Engine or resource policy specified cannot be supported on this flavor.	Please reselect flavor.
400	ModelArts.2841	DuplicateTrainingExperimentName	Duplicate training experiment (name: %s).	Change training experiment name.
400	ModelArts.2842	TrainingExperimentNotFound	Training job (id: %s) not found.	Check whether the training experiment information in the request is valid.
400	ModelArts.2843	QueryVPCEPFailed	Query vpc endpoint failed for reason (%s).	vpcep request failed, Contact technical support.
400	ModelArts.2844	ChangeVPCEPPolicyFailed	Change vpc endpoint policy failed for reason (%s).	vpcep request failed, Contact technical support.
400	ModelArts.2845	ImageSizeExceeded	SWR image size exceeded (%dGB > %dGB).	Please modify your swr image.
400	ModelArts.2846	TrainingExperimentTypeMismatch	The type of training job (%s) does not match the type of training experiment (%s).	Please select a training experiment that matches the training job type.
400	ModelArts.2847	IEFNodeUnavailable	The edge node (%s) current status is %s, cannot perform federated training.	Please select an available edge node.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.2848	IEFClientError	Please check whether the parameter edge_node_id(%s) is correct, error detail: %s.	Please select an available edge node.
400	ModelArts.2849	FederationParticipantsReachLimit	Exceeded maximum federation participants limit.	Please select other federations with participants below the limit.
400	ModelArts.2850	FederationDuplicateActivityName	Duplicate federation activity (name %s).	The federation activity name already exists, please change it.
400	ModelArts.2851	FederationDuplicateParticipant	you have already joined this federation activity(id %s).	Please select a federation activity that you have not yet joined.
400	ModelArts.2853	FederationOperationForbidden	Prohibition of federation activity status transitions from %s to %s.	This operation is prohibited.
400	ModelArts.2854	UnsupportedRemoteType	Unsupported input/output channel remote type.	Change to a supported channel remote type.
400	ModelArts.2855	RunUserDeny	Prohibition of run_user < %d when the training job is not in the dedicated pool.	This operation is prohibited.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.2856	NotOwnImagePublishGalleryError	Unable to publish this algorithm to gallery, because image of the algorithm is not own to you.	Please use the algorithm that image is own to you.
400	ModelArts.2857	InvalidLocalCodeDir	The local_code_dir parameter verification failed for reason(%s)	Check the validity of the information in the request.
400	ModelArts.2858	Unsupported Operation	Current operation not supported, (%s).	Please refer to hints.
400	ModelArts.2859	InvalidWorkingDir	The working_dir parameter verification failed for reason(%s)	Check the validity of the information in the request.
400	ModelArts.2879	JobRuntimeTypeNotDebug	Job runtime_type is not debug	Please adjust job runtime_type to debug
400	ModelArts.2880	ImageSaveJobExist	ImageSaveJob is exist	Please wait ImageSaveJob to be completed
400	ModelArts.2883	ImageSaveJobSecPhaseNotRunning	training job secondary_phase is not running	Please check training job secondary_phase
400	ModelArts.2884	ImageSwrAlreadyExist	image tag and image name is exist	Please check the image tag and name is unique in namespace
400	ModelArts.2885	AccessImageSwrException	the access of user swr is exception	Please check the image namespace is exist or user authorization is enough

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.2886	NotTrainingPool	Pool (%s) is not a training pool	Please check whether the pool supports the training job type
400	ModelArts.2887	PoolNotRunning	Pool (%s) is not running	Please check the status of the pool
400	ModelArts.2889	GetOsPoolFailed	Get pool (%s) failed.%s	Please check the validity of the pool in the request
400	ModelArts.2891	WorkspaceSetError	workspace access error,error message is %s, error code is %s	Please try to solve by error message
400	ModelArts.3015	Config json file does not conform to the specification.	The model configuration file does not comply with file specifications.	Check whether config.json or initial_config complies with the specifications.
400	ModelArts.3016	Failed to parse config json file because of unsupported fields, types, or formats({0}, {1}).	Failed to parse the configuration file because the field, type, or format is not supported.	Check whether the field is correct as prompted.
400	ModelArts.3017	Failed to publish model because source location noncompliance.	Failed to publish the model because the model source path does not meet specifications.	Check whether the source_location value complies with the specifications.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.3023	Invalid labels({0}) of model.	Invalid model label {0}.	Ensure that a model label starts with a letter character and contains a maximum of 64 characters, including letters, digits, and underscores (_).
400	ModelArts.3024	The templateInputs field is incorrectly configured, check if the input is consistent with the template requirements.	Incorrect configuration. Check whether the configuration complies with the template requirements.	The parameter configuration does not comply with the template requirements. Modify the configuration.
400	ModelArts.3025	User ({0}) has only a single share permission and cannot be shared with all users.	User {0} does not have permission to share data with all users.	Only the administrator or accounts in the whitelist have permission to share data with all users.
400	ModelArts.3026	Failed to delete infer format, infer format ({0}) already be used by template.	Failed to delete input/output mode {0} because it is being used by a template.	Find the template that is using the input/output mode, disassociate the mode from the template, and delete the mode again.
400	ModelArts.3027	The model used by the template ({0}) is private and belongs to other users and cannot be shared.	Input/Output mode {0} used by the template cannot be shared.	Select another mode.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.3028	Failed to publish model, please select input and output mode.	Failed to publish the model. Select an input/output mode.	Select another input/output mode and publish the model again.
400	ModelArts.3029	Model publishing failed because the Template model must provide the template ID used.	Failed to publish the model because the template ID is unavailable.	Enter the template ID.
400	ModelArts.3030	Model publishing failed, template {0} does not support modifying its built-in input and output mode.	Failed to publish the model because the built-in input/output mode of template {0} cannot be changed.	Do not change the input/output mode of a template when publishing the template model.
400	ModelArts.3031	Cancel sharing failed, {0} has not been shared with other tenants.	Failed to cancel the sharing because image {0} is not shared with other users.	Ensure that image {0} is not shared with other users.
400	ModelArts.3032	Image {0} cannot be shared to admin {1}, please check the image permissions.	Failed to share image {0} with the administrator {0}. Check the image permission.	Perform operations according to the error message.
400	ModelArts.3033	Template publishing failed, please provide template label.	Failed to publish the template model because its label is left blank.	Set the model label.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.3034	Template model failed to be published, template field is required.	Failed to publish the template model because the template field is left blank.	Set template.
400	ModelArts.3035	Profile preview failed, sourceLocation and previewConfig fields need to provide at least one.	Failed to preview the configuration file.	Either source_location or preview_config must be specified.
400	ModelArts.3036	Parameter verification failed, parameter {0} is required.	Failed to check parameter {0} because it is left blank.	Set parameters according to the error message.
400	ModelArts.3037	Parameter {0}:{1} is invalid.	Invalid parameter {0}:{1}.	Modify passwords as prompted.
400	ModelArts.3038	When isSpecific is true, domainId is required.	Failed to perform the operation because domain_id is left blank.	Set domain_id as prompted.
400	ModelArts.3039	Template input input_id cannot be repeated.	Failed to perform the operation because input_id is repetitive.	Ensure that the input_id value is unique.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.3040	When the installer is yum or apt-get, and the version constraint exists, the version constraint can only be EXACT.	Failed to perform the operation because restraint can only be EXACT.	Change the restraint value to EXACT.
400	ModelArts.3041	When the installer is conda or pip, and the version constraint exists, the version constraint can only be EXACT, ATLEAST, or ATMOST.	Failed to perform the operation because restraint can only be EXACT, ATLEAST, or ATMOST.	Change the restraint value to EXACT, ATLEAST, or ATMOST.
400	ModelArts.3063	Mode unshared failed, the template using this mode has been shared, you need to cancel the sharing of the template first.	Failed to cancel mode sharing because the template using the mode is being shared.	Perform operations according to the error message.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.3070	There is a model name that does not belong to this tenant: ({0}).	The model does not belong to user {0}.	The domain ID of the subscribed model is different from the entered domain ID. Contact the model subscription service and check whether the entered domain ID is correct.
400	ModelArts.3072	The domain id is the same and does not support subscribing to your own model.	You cannot subscribe to your own model.	Use another account to subscribe to the model.
400	ModelArts.3074	Unable to delete subscribed model.	Failed to delete the subscribed model.	Unsubscribe from the model and try again.
400	ModelArts.3076	Model does not support deploying edge services and cannot broadcast.	Failed to perform the operation because the model cannot be deployed as an edge service.	The install_type of the model does not support edge service.
400	ModelArts.3077	Model is a subscription model that cannot be broadcast.	Failed to perform the operation because the model has been subscribed.	The model_source of the model has been set to subscribe.
400	ModelArts.3078	The model status is not normal and cannot be broadcast.	Failed to perform the operation because the model status is abnormal.	Check the model status.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.3079	Purchased license, can't cancel subscription.	Failed to unsubscribe from the model because a license has been configured for the model.	Delete the license and unsubscribe from the model.
400	ModelArts.3514	Field [input_params] must be set for the model that is used to deploy the batch service.	The input_params parameter for the batch service model is left blank.	Set input_params.
400	ModelArts.3515	The model {id} cannot be used for deploying services because it is still being published.	Failed to deploy the service because model {ID} is not ready.	Try again after the model is ready.
400	ModelArts.3516	Failed to deploy services because the version of the model that is used for deploying services is the same.	Failed to deploy the service due to duplicate model versions.	Correct the model version list to ensure that no duplicate model versions exist.
400	ModelArts.3518	The {service_type} service has not been enabled or authorized.	Service {type} has not been enabled or authorized.	Enable or authorize the service.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.3519	Services only in one of states in [deploying, running, stopping, deleting] can be updated.	Services in the deploying, running, stopping, or deleting state cannot be updated.	Do not update services in the deploying, running, stopping, or deleting state.
400	ModelArts.3520	A maximum of {number} {service_type} services are allowed.	The total number of {type} services has reached the maximum allowed limit {quantity}.	Delete the services that are no longer used or contact service O&M personnel to increase the quantity quota.
400	ModelArts.3521	A maximum of {number} {service_type} services in running status are allowed.	The total number of {type} services in the running state has reached the maximum allowed limit {quantity}.	Delete the services that are no longer used or contact service O&M personnel to increase the quantity quota.
400	ModelArts.3522	A maximum of {number} {service_type} service instances are allowed.	The total number of {type} service instances has reached the maximum allowed limit {quantity}.	Delete the service instances that are no longer used or contact service O&M personnel to increase the quantity quota.
400	ModelArts.3523	Service {name} has been subscribed.	Service {name} has been subscribed.	Directly use the subscribed service.
400	ModelArts.3524	Failed to subscribe to the service because {reason}.	Failed to subscribe to the service because {Reason}.	Locate the fault based on the failure cause.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.3525	Forbidden to update service {name} because it is a shared service.	Failed to update service {name} because it is a shared service.	Do not update a shared service.
400	ModelArts.3527	Forbidden to subscribe to service {id} because it is a {service_type} service.	Failed to subscribe to service {ID} because it is a {type} service.	The service of this type cannot be subscribed to.
400	ModelArts.3528	Forbidden to subscribe to service {id} because it is released by yourself.	Failed to subscribe to service {ID} because it is your service.	You cannot subscribe to services published by yourself.
400	ModelArts.3529	Model {id} under service {id} does not exist.	Model {ID} of service {ID} does not exist.	Ensure that the model ID is correctly set and that the model belongs to the target service.
400	ModelArts.3531	Service {id} has created the QR code.	A QR code has been created for service {ID}.	The QR code has been created for the service and does not need to be created again.
400	ModelArts.3532	No QR code found for service {id}.	A QR code has not been created for service {ID}.	Create a QR code for the service and try again.
400	ModelArts.3533	The selected edge node does not support {GPU/NPU}.	The selected edge node flavor does not support {GPU/NPU}.	Add the target hardware support to the selected edge node, or use another edge node that supports the target hardware.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.3534	The security group must contain at least one inbound rule to allow the TCP request from source address 0.0.0.0/0 and port 8080.	The security group must contain at least one inbound rule to permit the requests whose protocol is TCP, source address is 0.0.0.0/0, and port number is 8080.	Add the inbound rule to the security group.
400	ModelArts.3535	Subservice {name} of service {name} does not exist.	Subservice {name} of service {name} does not exist.	Ensure that the subservice name is correctly set and that the subservice belongs to the target service.
400	ModelArts.3536	Subservice {name} of service {name} has been subscribed.	You have subscribed to subservice {name} of service {name}.	Directly use the subscribed subservice.
400	ModelArts.3538	The requirement notification of service {name} should not exceed one.	At most one requirement notification is allowed for service {name}.	Check the configuration and ensure that no more than one requirement notification is configured for service {name}.
400	ModelArts.3540	Operation failed because you are not the owner of service {name}.	You are not the owner of service {name}.	This operation can be performed only by the service owner.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.3541	Error code {error_code} of service {name} does not exist.	Error code {error code} of service {name} does not exist.	Ensure that the error code is correctly set and that the error code belongs to the target service.
400	ModelArts.3542	Error code {error_code} of service {name} already exists.	Error code {error code} of service {name} already exists.	Change the error code and try again.
400	ModelArts.3544	URL domain CNAME resolution failed.	Failed to parse the CNAME domain.	Check whether the URL is correct.
400	ModelArts.3545	Invalid certificate or private key.	Invalid certificate or private key.	Check whether the certificate or private key is correctly configured.
400	ModelArts.3547	Cluster {id} does not exist.	Cluster {ID} does not exist.	Check whether the cluster ID is correct.
400	ModelArts.3548	Path {path} is not a valid regex.	Path {path} is not a valid regular expression.	Check whether the path is correct.
400	ModelArts.3549	Error code {code} is duplicated.	Duplicate error code {error code} in the request body.	Modify the error code to ensure that the error code is unique.
400	ModelArts.3550	Domain {name} has been used.	Domain {noun} has been used.	Change the domain name and try again.
400	ModelArts.3551	OBS path {path} does not exist.	OBS path {path} does not exist.	Check whether the OBS path is correct.
400	ModelArts.3552	The length of [resource_id] must be in the range of [0, 64].	The resource_id value is not in the range from 0 to 64.	Check whether RESOURCE_SPEC_CODE is too long.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.3553	Service {name} not subscribed yet.	Service {name} has not been subscribed to.	Subscribe to the service and try again.
400	ModelArts.3554	Cluster name {name} has been used.	Cluster name {name} has been used.	Change the cluster name and try again.
400	ModelArts.3556	Forbidden to share ai-service {id} because ai-service is unsharable.	AI service {ID} cannot be shared.	AI services cannot be shared.
400	ModelArts.3559	Source error code {error_code} of service {name} already exists.	Source error code {error_code} of service {name} already exists.	Change the source error code and try again.
400	ModelArts.3560	Project {project_id} has not subscribed to subservice {id}.	Subservice {ID} has not been subscribed for project {ID}.	Subscribe to the subservice and try again.
400	ModelArts.3561	Insufficient node quota.	Insufficient node quota.	Contact service O&M personnel to increase the quota or delete unnecessary nodes to release resources.
400	ModelArts.3562	The value of field {due_time} must be a future time.	The due_time value must be later than the current time.	Set the due_time parameter to a time later than the current time.
400	ModelArts.3564	The sample collection task has been created for service {id}.	Sample collection for service {ID} has been created.	Do not create the task again.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.3565	No sample collection task is created for service {id}.	Sample collection for service {ID} has not been created.	Create a service sample collection task and try again.
400	ModelArts.3566	Forbidden to modify the default workspace.	The default workspace cannot be modified.	Do not modify the default workspace.
400	ModelArts.3567	OBS error occurs because {reason}.	OBS error because {cause}.	Locate the fault based on the failure cause.
400	ModelArts.3568	OBS client error occurs.	The OBS client is abnormal.	Contact service O&M personnel.
400	ModelArts.3572	Invalid OBS URL {url}.	Invalid OBS path {path}.	Check whether the OBS path is correct.
400	ModelArts.3573	Failed to obtain the bucket name from OBS URL {url}.	Failed to obtain the bucket name from OBS path {path}.	Check whether the OBS path is correct.
400	ModelArts.3574	Failed to validate the correctness of OBS URL {url}.	Failed to check OBS path {path}.	Check whether the OBS path is correct.
400	ModelArts.3576	Failed to query the model quota.	Failed to obtain the model quota.	Contact service O&M personnel.
400	ModelArts.3577	Failed to update the model quota.	Failed to update the model quota.	Contact service O&M personnel.
400	ModelArts.3578	Insufficient model quota. Increase the quota or delete any unnecessary services.	Insufficient model quota.	Increase the quota or delete the models that are no longer used, and try again.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.3580	Model not purchased.	The model has not been purchased.	Purchase the model and try again.
400	ModelArts.3582	Failed to create edge configurations	Failed to create edge configurations	Contact service O&M personnel.
400	ModelArts.3583	Failed to create the edge application due to IEF error.	Failed to create the edge application due to an IEF platform error.	Contact service O&M personnel.
400	ModelArts.3584	Failed to update the edge application.	Failed to update the edge application.	Contact service O&M personnel.
400	ModelArts.3585	The host port has been used by another edge application.	The host port has been used by another edge application.	Use another port or contact service O&M personnel.
400	ModelArts.3586	Failed to create the edge application.	Failed to create the edge application.	Contact service O&M personnel.
400	ModelArts.3587	Failed to delete the edge application.	Failed to delete the edge application.	Contact service O&M personnel.
400	ModelArts.3588	Failed to create edge volumes and environments.	Failed to create edge volumes and environments.	Contact service O&M personnel.
400	ModelArts.3589	Model {id} cannot be used to deploy {service_type} service.	Model {ID} cannot be used to deploy the {type} service.	Select another deployment type.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.3590	Invalid expression rule. Create a rule using Spring Expression Language and the desired result type is Boolean.	Invalid expression rule.	Ensure that the rule complies with the Spring expression language syntax, and the expected result is of Boolean type.
400	ModelArts.3591	There are duplicate rules in the extend-configs.	Duplicate rules in extend-configs.	Modify the expression specifications to prevent duplicate expressions.
400	ModelArts.3592	Incorrect model health configuration.	Incorrect model health check parameters.	Check whether the health check parameters of the model are correct.
400	ModelArts.3593	The model has been expired.	The model has expired.	Purchase or subscribe to the model again.
400	ModelArts.3594	Failed to query IEF instances.	Failed to obtain the edge instance.	Contact service O&M personnel.
400	ModelArts.3595	Failed to create the IEF application mesh.	Failed to create the application mesh.	Contact service O&M personnel.
400	ModelArts.3596	Failed to update the IEF application mesh.	Failed to update the application mesh.	Contact service O&M personnel.
400	ModelArts.3597	Please authorize the development environment to use and store your AK and SK.	Failed to use the service because the development environment has not been authorized to store and use your AK and SK.	Authorize the development environment to store and use your AK and SK.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.3598	Only nodes in the [Running] state can be selected.	Only running nodes can be selected.	Select a running node and try again.
400	ModelArts.3610	Parameter {name} cannot be empty.	Parameter {name} cannot be left blank.	Set the parameter and try again.
400	ModelArts.3611	Failed to query the batch task run log.	Failed to obtain batch processing task logs.	Contact service O&M personnel.
400	ModelArts.3612	Failed to {create/update} the service payload due to error code {code}. Please try later or submit a service ticket for professional technical support.	Failed to {create/update} service workload. The error code is {error code}. Try again later or submit a service ticket.	Try again or contact service O&M personnel.
400	ModelArts.3613	Failed to register the API. Please try later or submit a service ticket for professional technical support.	Failed to register the service API. Try again later or submit a service ticket.	Try again or contact service O&M personnel.
400	ModelArts.3620	Failed to get the number of associated services.	Failed to obtain the number of associated services.	Contact service O&M personnel.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.3811	Failed to query the agency because {reason}.	Failed to obtain the agency because {Reason}.	Locate the fault based on the failure cause.
400	ModelArts.3812	Failed to create AI resource specification code {code} because {reason}.	Failed to create resource flavor code {code} because {Reason}.	Locate the fault based on the failure cause.
400	ModelArts.3813	Failed to update AI resource specification code {code} because {reason}.	Failed to update resource flavor code {code} because {Reason}.	Locate the fault based on the failure cause.
400	ModelArts.3814	Failed to delete AI resource specification code {code} because {reason}.	Failed to delete resource flavor code {code} because {Reason}.	Locate the fault based on the failure cause.
400	ModelArts.3815	Failed to delete AI subservice {id} because {reason}.	Failed to delete subservice {ID} because {Reason}.	Locate the fault based on the failure cause.
400	ModelArts.3816	Failed to create the AI proxy mapping because {reason}.	Failed to create the proxy mapping because {reason}.	Locate the fault based on the failure cause.
400	ModelArts.3817	Failed to update the AI proxy mapping because {reason}.	Failed to update the proxy mapping path because {reason}.	Locate the fault based on the failure cause.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.3818	The agency of service {name} has been defined.	The agency of service {name} has been configured.	Do not repeat the operation.
400	ModelArts.3822	The batch service task does not exist.	The batch service does not exist.	Enter the correct batch service ID.
400	ModelArts.3825	Failed to generate the SDR data because {reason}.	Failed to generate CDR data because {Reason}.	Locate the fault based on the failure cause.
400	ModelArts.3826	Failed to create the API because {reason}.	Failed to create the API because {Reason}.	Locate the fault based on the failure cause.
400	ModelArts.3827	Failed to subscribe to the API because {reason}.	Failed to subscribe to the API because {Reason}.	Locate the fault based on the failure cause.
400	ModelArts.3828	Failed to unsubscribe from the API because {reason}.	Failed to unsubscribe from the API because {Reason}.	Locate the fault based on the failure cause.
400	ModelArts.3829	Failed to modify the API subscription status because {reason}.	Failed to update the API subscription because {Reason}.	Locate the fault based on the failure cause.
400	ModelArts.3830	Operation not allowed. Change the charging mode from postpaid to prepaid in CBC.	You are not allowed to change the billing mode from postpaid to prepaid.	Change the billing mode in CBC.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.3831	Operation not allowed. Change the charging mode from prepaid to postpaid in CBC.	You are not allowed to change the billing mode from prepaid to postpaid.	Change the billing mode in CBC.
400	ModelArts.3832	The subscription configuration is not modified and does not need to be updated. Please check.	Failed to perform the operation because the subscription configuration has not modified.	Change the subscription configuration and try again.
400	ModelArts.3833	Failed to delete the API because {reason}.	Failed to delete the API because {Reason}.	Locate the fault based on the failure cause.
400	ModelArts.3834	Failed to update metadata in CBC because {reason}.	Failed to update the CBC metadata because {Reason}.	Locate the fault based on the failure cause.
400	ModelArts.3835	Resource specification code {code} of service {name} does not exist.	Resource flavor code {code} of service {name} does not exist.	Ensure that the resource flavor code is correctly set and that the code belongs to the target service.
400	ModelArts.3836	Resource specification code {code} of project {id} does not exist.	Resource flavor code {code} of project {1} does not exist.	Ensure that the resource flavor code is correctly set and that the code belongs to the target project.
400	ModelArts.3837	Resource ID {id} of project {id} does not exist.	Resource {ID} of project {1} does not exist.	Ensure that the resource ID is correctly set and that the resource belongs to the target project.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.3838	The hard-select has not been enabled for service {id}.	Hard example filtering of service {0} has not been enabled.	Enable hard example filtering for the service and try again.
400	ModelArts.3839	The hard-select has been enabled for service {id}.	Hard example filtering of service {0} has been enabled.	You do not need to enable it again.
400	ModelArts.3840	Only one algorithm can be set for the service model.	Service {0} has more than one labeling type.	Modify the configuration to ensure that there is only one labeling type.
400	ModelArts.3841	The value of [dataset_type] must be one of values in [image_classification, object_detection].	The labeling type is not object detection or image classification.	Change the labeling type to object detection or image classification.
400	ModelArts.3842	The labeling type of service algorithm {name} does not match that of label task {type}.	The labeling type of service algorithm {0} does not match that of label task {1}.	Modify the labeling type or contact service O&M personnel.
400	ModelArts.3843	Failed to query the information about dataset {id}.	Failed to obtain database {0}.	Contact service O&M personnel.
400	ModelArts.3844	Failed to import samples to dataset {id}.	Failed to import the sample to dataset {0}.	Contact service O&M personnel.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.3845	No sample is available for import under the OBS path {path} of service {id}.	No sample is available to be imported to OBS path {path} of service {ID}.	Try again or contact service O&M personnel.
400	ModelArts.3846	Failed to add the OBS bucket lifecycle rule for service {id}.	Failed to add the lifecycle rule for the OBS bucket of service {ID}.	Contact service O&M personnel.
400	ModelArts.3847	Failed to delete the OBS bucket lifecycle rule for service {id}.	Failed to delete the lifecycle rule for the OBS bucket of service {ID}.	Contact service O&M personnel.
400	ModelArts.3848	Service {id} in the information of dataset {import_type} already exists.	The service {ID} of the {import_type} dataset already exists.	Check the request and select the correct service ID.
400	ModelArts.3849	Service {id} in the information of dataset {import_type} does not exist.	The service {ID} of the {import_type} dataset does not exist.	Check whether the service ID is correct.
400	ModelArts.3850	The hard-sample-select task {id} of service {id} does not exist.	Hard example task {ID} of service {ID} does not exist.	Ensure that the hard example task ID is correctly set and that the task belongs to the target service.
400	ModelArts.3851	The status of hard-sample-select task {name} cannot be changed from {status} to {status}.	The status {status value} of the hard example task {name} cannot be updated to {status value}.	Try again or contact service O&M personnel.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.3852	Failed to start hard-sample-select task {id} because related OBS files have been deleted.	Failed to start hard example task {ID} because key OBS files have been deleted.	Try again or contact service O&M personnel.
400	ModelArts.3853	The hard-sample-select quota of project {id} does not exist.	The quota for selecting hard samples for project {ID} does not exist.	Create the quota and try again.
400	ModelArts.3854	The hard-sample-select quota of project {id} already exists.	The quota for selecting hard samples for project {ID} already exists.	You do not need to create it again.
400	ModelArts.3856	A maximum of {number} free services are allowed.	The number of free services has reached the maximum allowed limit {Number}.	Delete unnecessary free services to release resources.
400	ModelArts.3857	A maximum of {number} free service instances are allowed.	The number of free services has reached the maximum allowed limit {Number}.	Delete unnecessary free services to release resources.
400	ModelArts.3862	Subservice {id} has been added to whitelist URL {url}.	The URL {URL} has been added to the whitelist of the sub-service {ID}.	Do not repeat the operation.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.3863	Whitelist URL {id} of service {name} does not exist.	The whitelist URL {ID} of the service {name} does not exist.	Ensure that the URL ID in the whitelist is correct and that the URL belongs to the corresponding service.
400	ModelArts.3864	Subservice {id} of service {name} does not exist.	The sub-service {ID} of service {name} does not exist.	Ensure that the sub-service ID is correct and that the sub-service belongs to the corresponding service.
400	ModelArts.3865	The size of the download whitelist of service {name} exceeds the maximum number {number} allowed.	The number of the whitelist records downloaded by service {name} exceeds the upper limit {value}.	Contact service O&M personnel.
400	ModelArts.3866	Domain ID {id} and user ID {id} already exist.	The domain ID {ID} and user ID {ID} already exist.	Do not repeat the operation.
400	ModelArts.3867	Domain ID {id} does not exist.	The domain ID {ID} does not exist.	Check the request and enter the correct domain ID.
400	ModelArts.3868	Domain ID {id} and user ID {id} do not exist.	The domain ID {ID} and user ID {ID} do not exist.	Check the request and enter the correct domain ID and user ID.
400	ModelArts.3869	Failed to delete the task type because {reason}.	Failed to delete the task type. Fault cause: {fault cause}.	Locate the fault based on the failure cause.
400	ModelArts.3870	Failed to subscribe to the API because {reason}.	Failed to subscribe to the API. Fault cause: {fault cause}.	Contact service O&M personnel.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.3871	Forbidden to subscribe to postpaid QPS for the task type.	You are not allowed to subscribe to postpaid QPS for the task type.	Check the request parameter or contact service O&M personnel.
400	ModelArts.3872	Forbidden to change the charging mode for the task type from prepaid to postpaid QPS.	You are not allowed to change the billing mode of the task type from prepaid mode to postpaid QPS mode.	Check the request parameter or contact service O&M personnel.
400	ModelArts.3873	No corresponding subservice is found in cluster {name}.	The cluster {0} does not have the corresponding sub-service.	Check the request parameter or contact service O&M personnel.
400	ModelArts.3880	Failed to create the AS policy because {reason}.	Failed to create the AS policy. Fault cause: {fault cause}.	Locate the fault based on the failure cause.
400	ModelArts.3890	The total number of AS policies exceeds the maximum number {number} allowed.	The number of AS policies exceeds the upper limit {0}.	Delete unnecessary AS policies.
400	ModelArts.3892	Calling failed due to invalid CBS request.	Failed to call the API due to the invalid CBS request.	Try again or contact service O&M personnel.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.3894	The specification configuration information of runtime environment {name} does not exist.	The specification configurations of the running environment {0} does not exist.	Set the flavor configurations for the running environment and try again.
400	ModelArts.3895	The specification configuration information of runtime environment {name} already exists.	The specification configurations of the running environment {0} already exist.	Do not repeat the operation.
400	ModelArts.3896	Resource specification code {0} for user {1} already exists.	The records of the resource specification code {0} in project {1} already exist.	Do not repeat the operation.
400	ModelArts.3897	Failed to {create/obtain/remove} the AS policy due to error code {code}.	Failed to request {create/obtain/delete} the AS. Error code: {0}.	Try again or contact service O&M personnel.
400	ModelArts.3950	Failed to obtain the AI service list.	Failed to obtain the AI service list.	Contact service O&M personnel.
400	ModelArts.3951	Failed to parse file {name}.	Failed to parse the file {name}.	Ensure that the file name is correct.
400	ModelArts.3952	Insufficient resources. Please try later.	Insufficient resources.	Try again later or contact service O&M personnel.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.3953	Failed to delete online service {id} because it has been associated with AI service {name}. Please disassociate it and try again.	Failed to delete real-time service {ID} because it has been associated with AI service {name}.	Cancel the association between the real-time service and all AI services and try again.
400	ModelArts.3954	AI resource specification {id} does not exist.	The cloud resource specification {ID} does not exist.	Ensure that the cloud resource specification ID is correct.
400	ModelArts.3955	Field [resource_id_components] of AI resource specification {id} cannot be empty.	The ID components in the cloud resource specification {ID} cannot be left blank.	Set the resource_id_components parameter and try again.
400	ModelArts.4105	Incorrect JSON format of the input data.	The JSON format of the request body is incorrect.	Use a request body in correct JSON format.
400	ModelArts.4106	Invalid authorization request. Your account is restricted.	Your account has been suspended.	Ensure your account balance is sufficient.
400	ModelArts.4111	Invalid Region header.	Invalid Region header.	Ensure the region header is correct.
400	ModelArts.4122	Incorrect username or password.	The username or password is incorrect.	Check the username and password.
400	ModelArts.4126	Account locked.	The account has been locked.	Wait until the account is unlocked and try again.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.4128	Weak password is forbidden.	Do not use a weak password.	Configure a secure password.
400	ModelArts.4129	System error.	An error occurred. The taskEntity of a recovery task cannot be null.	Check taskEntity and ensure it is not null.
400	ModelArts.4200	Invalid request.	The values of the request parameters are invalid.	Check whether the request parameters are correct.
400	ModelArts.4201	Invalid request URL because no service ID or request URI included.	The request URL does not contain a service ID.	Use a correct request URL.
400	ModelArts.4202	Empty request.	The request is empty.	Ensure the request body is not empty.
400	ModelArts.4203	Invalid request because requested service %s is unavailable.	The requested service has not been started.	Ensure the service is started.
400	ModelArts.4205	A subdirectory must be specified as the dataset input or output path.	A subdirectory must be specified as the dataset input or output path.	A subdirectory must be specified as the dataset input or output path.
400	ModelArts.4301	The labeling job does not exist.	The labeling job does not exist, and the task has been stopped.	Check the labeling job.
400	ModelArts.4311	OBS bucket does not exist	The OBS bucket does not exist.	Ensure that the bucket name is correct and that the bucket name exists in OBS.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.4312	OBS path does not exist	Incorrect or invalid bucket name.	Ensure that the bucket name is correct and that the bucket name exists in OBS.
400	ModelArts.4313	OBS path is invalid	Invalid characters in the OBS path.	Ensure that the OBS path consists of valid characters, including digits, letters, hyphens (-), underscores (_), and slashes (/).
400	ModelArts.4314	Obs error	OBS access error.	Ensure that you have the permission to access OBS and that the OBS path is valid.
400	ModelArts.4315	Invalid OBS path.	The OBS path is invalid.	Ensure the data URL does not contain //.
400	ModelArts.4316	OBS buckets are not supported.	OBS buckets are not supported.	Select a PFS bucket for the parallel file system.
400	ModelArts.4337	OperationCenter error.	An error occurred in OperationCenter, leading to a failure to send alarms to IES.	Check the alarms.
400	ModelArts.4338	The resource not exists	The dataset ID or labeling task ID does not exist.	Ensure that the input dataset ID or labeling task ID is correct.
400	ModelArts.4340	Import path does not contain valid file	No valid file exists in the import path.	Ensure that the file in the OBS path is valid.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.4342	Dataset publish with splitting annotated samples error	Incorrect splitting for labeled samples.	Ensure that the labeled samples and labels meet splitting criteria.
400	ModelArts.4343	Dataset is publishing, the operation is forbidden	Do not switch, import, synchronize, or publish a dataset version because there is an ongoing publish task.	Perform the operations after the ongoing publish task is complete.
400	ModelArts.4344	Dataset is being deleted, annotation is forbidden	No labeling task is allowed because the dataset is being deleted.	Ensure that the dataset ID is correct.
400	ModelArts.4345	File not found	The HDFS file does not exist.	Ensure that the OBS path is correct and that the file is available in the OBS path.
400	ModelArts.4346	Failed to obtain clusters.	Obtaining MRS clusters failed.	Check the MRS clusters.
400	ModelArts.4347	List files failed	Failed to obtain HDFS files.	Ensure that the OBS path is correct and that the files are available in the OBS path.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.4349	Dataset is switching version, the operation is forbidden	Do not switch, import, synchronize, or publish a dataset version because there is an ongoing version switching task.	Perform the operations after the ongoing version switching task is complete.
400	ModelArts.4350	The work_path is too long, please select shorter folder	The value of work_path exceeded the limit.	Change the value of work_path to a valid one.
400	ModelArts.4351	Dataset already exists	A dataset whose name is the value of dataset_name already exists.	Change the value of dataset_name.
400	ModelArts.4352	Dataset does not exist	The dataset ID does not exist.	Ensure that the imported dataset ID is correct.
400	ModelArts.4353	Dataset version does not exist	The dataset version ID does not exist.	Check dataset version parameters.
400	ModelArts.4355	Sync data source task exist	A data synchronization task is being executed for the dataset.	Perform the operations after the ongoing data synchronization task is complete.
400	ModelArts.4356	Dataset already has running import task	A data import task is being executed for the dataset.	Perform the operations after the ongoing data import task is complete.
400	ModelArts.4357	Parse AI annotation result file name error	Failed to parse the labeling file name.	Ensure that the file name in the OBS auto labeling result path is correct.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.4358	Invalid export path	Invalid OBS export path.	Ensure that the export path is correct.
400	ModelArts.4359	Export task does not exist	The export task ID does not exist.	Ensure that the export task ID is correct.
400	ModelArts.4361	Import AI annotation error	Failed to synchronize the auto labeling task result.	Ensure that the auto labeling task result is correct.
400	ModelArts.4362	Import data error	Failed to import data.	Ensure that the authentication information and the request parameters for creating an import task are correct.
400	ModelArts.4364	Dataset workPath subdir already exists	The work_path subdirectory already exists in the dataset directory.	Ensure that the subdirectory in the dataset directory is correct.
400	ModelArts.4365	Dataset labels not set	The label set of the dataset is empty.	Create labels for the dataset.
400	ModelArts.4368	Parse pc bin file error	Failed to parse the point cloud binary file.	Ensure that the point cloud binary file is not damaged.
400	ModelArts.4369	Parse pc prelabel json file error	Failed to parse the point cloud pre-labeling file.	Ensure that the point cloud pre-labeling file is correct.
400	ModelArts.4370	One dataset version can be released in a minute, please try again later	Frequent dataset version creation.	Do not frequently create dataset versions.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.4371	Dataset version already exists	The value of version_name is the same as an existing version.	Change the value of version_name.
400	ModelArts.4372	Valid image not found	No point cloud data image found.	Ensure that the point cloud data image is correct.
400	ModelArts.4374	Invalid path	Invalid OBS path.	Ensure that the OBS path is correct and that the file is available in the OBS path.
400	ModelArts.4375	Parse pc obs image error	Failed to parse the point cloud data image in OBS.	Ensure that the point cloud data image is correct.
400	ModelArts.4376	Unsupported pc pcd format error	The PCD attribute is not supported.	Check the point cloud data in PCD format.
400	ModelArts.4377	Pc pcd format error	Failed to parse the point cloud data in PCD format.	Ensure that the PCD file is valid.
400	ModelArts.4378	Parse pc pcd file error	Failed to parse the point cloud data in PCD format.	Ensure that the PCD file is correct.
400	ModelArts.4379	Unsupported pc file format error	The point cloud data format is not supported.	Ensure that the imported point cloud data is in BIN or PCD format.
400	ModelArts.4380	Parse kitti calibration file error	Failed to parse the Kitti data.	Ensure that the data is correct.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.4381	Dataset is publishing, annotation is forbidden	Sample labeling is not allowed because a publish task is being executed in the dataset.	Wait until the publish task is complete.
400	ModelArts.4382	Generate 2d image error	Failed to generate a 2D image.	Ensure that the point cloud data is correct.
400	ModelArts.4384	Invalid export parameter	Invalid parameter.	Ensure that all mandatory parameters are included and valid.
400	ModelArts.4391	Task name is invalid	An auto deploy task whose name is the task_name value already exists in the dataset.	Ensure that the task name is correct.
400	ModelArts.4392	Task failed	Failed to create or run the dataset.	Ensure that the parameters and task resources are correct.
400	ModelArts.4393	Task stopped	The dataset task has been stopped.	Check the status of the dataset task with the specified ID.
400	ModelArts.4396	Parameter is invalid	Invalid parameter.	Ensure that the import task parameters are correct.
400	ModelArts.4398	Invalid system language.	The system language is invalid.	Check whether the system language is supported.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.4399	Type match error	The dataset type does not match the algorithm model.	Ensure that the algorithm model is correct.
400	ModelArts.4400	Can not get table schema	Failed to obtain the schema information from the OBS directory.	Ensure that the CSV file in the OBS directory is correct.
400	ModelArts.4401	Failed to export data.	Exporting data failed. The dataset file cannot be copied to the repository.	Take other measures to export data.
400	ModelArts.4404	Can not get table schema from DLI	Failed to obtain the schema information from DLI.	Ensure that the request parameters are correct.
400	ModelArts.4405	Dataset must contains labels to start automation labeling	Labeled samples must be available for enabling auto labeling.	Add new labeled samples to the dataset.
400	ModelArts.4406	Dataset must contains labels with at least five images to start automation labeling	The number of samples to be labeled for each label cannot be less than 5.	Ensure that each label has at least five samples.
400	ModelArts.4407	Dataset must contains unlabeled files	The dataset must contain samples that have not been labeled.	Ensure that there are unlabeled samples in the dataset.
400	ModelArts.4408	Dataset contains labels shape not bndbox	The label shape cannot be bounding box.	Ensure that the shape attribute is correct.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.4409	Running tasks has exceed the max threshold	The number of tasks that are being executed in the dataset has reached the maximum allowed limit.	Try again later.
400	ModelArts.4410	Label not found	The label does not exist.	Ensure that the label name is correct.
400	ModelArts.4411	Label already exists	The new label list contains existing label names.	Ensure that the label names are correct.
400	ModelArts.4412	Label shortcut already exists	The label shortcut already exists.	Change the label shortcut.
400	ModelArts.4413	Label is incompatible with annotation rules	The label attribute does not comply with specifications.	Ensure that the label attribute complies with specifications.
400	ModelArts.4414	Triplet label's from_entity or to_entity does not exist	The entity label does not exist in the triplet label.	Ensure that the entity label in the triplet label is available.
400	ModelArts.4415	Entity label can not be deleted because it is used by triplet label	Failed to delete the entity label because it is being used in the triplet label.	Wait until the label is not used in the triplet label and try again.
400	ModelArts.4416	Sync tags error	Failed to synchronize labels in the team labeling task.	Ensure that the synchronization task parameters are correct.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.4417	Update sample labels failed when upload sample and labels	Failed to update the sample labels.	Ensure that the labels are correct.
400	ModelArts.4418	Label property mask_gray_value already exists	Duplicate tag mask_gray_value.	Ensure that the tag mask_gray_value is correct.
400	ModelArts.4420	Sample not found	The sample with the specified ID does not exist.	Ensure that the sample ID is correct.
400	ModelArts.4421	Upload sample failed	Failed to upload the sample to the dataset.	Ensure that the uploaded sample type and data are correct.
400	ModelArts.4422	Sample already exists	The sample that is being imported already exists.	Enter that the imported sample is correct.
400	ModelArts.4423	Get sample size failed, please input size manually	Incorrect size of the image obtained from the object detection dataset.	Manually set the image size.
400	ModelArts.4425	Label property mask_gray_value is incompatible with defined value	The mask_gray_value value cannot be changed.	Ensure that the mask_gray_value value is not changed.
400	ModelArts.4426	Dataset must contain tags to start auto deploy	The label set of the dataset cannot be empty.	Create labels for the dataset.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.44 27	Dataset must contain tags with at least five images to start auto deploy	The number of samples to be labeled for each label cannot be less than 5.	Ensure that each label has at least five samples.
400	ModelArts.44 28	The label attribute name already exists.	The label attribute name already exists.	Configure another attribute name.
400	ModelArts.44 29	The label attribute value already exists.	The label attribute value already exists.	Configure another attribute value.
400	ModelArts.44 30	The label attribute or value does not exist.	The label attribute or value does not exist.	Check the label attribute and value.
400	ModelArts.45 02	The IAM agency name already exists, please delete the agency in IAM first and retry	The default IAM agency already exists.	Delete the IAM agency and create a new one.
400	ModelArts.45 03	Failed to create the DLI agency.	Creating the DLI agency failed.	Check the DLI agency.
400	ModelArts.45 50	Workspace clearing error.	An error occurred in clearing the workspace.	Handle carbon reader exceptions.
400	ModelArts.45 51	You are not allowed to delete workspace resources.	You do not have permission to delete workspace resources.	Check your permissions.
400	ModelArts.45 52	ModelArts workspace error.	An error occurred in the ModelArts workspace.	Check identity authentication in the workspace.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.4601	The workforce does not exist	The team ID does not exist.	Ensure that the team ID is correct.
400	ModelArts.4602	The workforce already exists	A team whose name is the workforce_name value already exists.	Ensure that the workforce_name value is correct.
400	ModelArts.4603	Update workforce state failed	Failed to delete the team.	Check whether the team has been deleted.
400	ModelArts.4604	The worker does not exist	The team member specified by work_id does not exist.	Ensure that the workforce_id and work_id values are correct.
400	ModelArts.4605	The worker already exists	The team member whose value is the email value already exists.	Ensure that the email value is correct.
400	ModelArts.4607	Failed to reset the password.	Requesting to reset the password failed.	The username is incorrect, or the password reset email cannot be sent to the account.
400	ModelArts.4609	Change password failed	Failed to change the account password.	Ensure that the new password complies with specifications.
400	ModelArts.4612	Task not found	The task ID does not exist.	Ensure that the task ID is correct.
400	ModelArts.4615	Workforce task is unfinished	The version cannot be switched because the team labeling task has not been completed.	Wait until the team labeling task is complete and try again.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.4616	Too many unfinished tasks.	There are too many unfinished tasks.	Finish tasks.
400	ModelArts.4617	The number of manager should be one	The team manager already exists.	Ensure that the roles of the team members are correct.
400	ModelArts.4618	Can not delete yourself	You cannot delete yourself from the team.	Ensure that the request parameters are correct.
400	ModelArts.4619	Workforce task does not exist	No team labeling task whose ID is the workforce_task_id value is allowed.	Ensure that the workforce_task_id value is correct.
400	ModelArts.4620	The workforce task already exists	A team labeling task whose name is the task_name value already exists.	Ensure that the task_name value is correct.
400	ModelArts.4621	Failed to start the team labeling task.	The team labeling task cannot be started.	Check the task.
400	ModelArts.4622	Invalid n_clusters, should less than the total number of samples	The number of groups has reached the maximum allowed limit.	Check whether the number of groups is less than the total number of samples.
400	ModelArts.4623	Workforce task is checking	Only one review task is allowed at a time.	Wait until all existing review tasks are complete and try again.
400	ModelArts.4624	The member task is not checked.	The member task is not checked.	Check the task.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.4625	There are no unchecked samples.	There are no unchecked samples.	Obtain unchecked samples.
400	ModelArts.4627	Request notify too frequently	Frequent notification request submission.	Try again later.
400	ModelArts.4628	Can not delete worker who has task	The team member cannot be deleted because the labeling task has not been completed.	Notify the member to complete the task.
400	ModelArts.4631	The task does not exist.	The task does not exist.	Check the task.
400	ModelArts.4633	The IAM user has been associated.	The IAM user has been associated.	Ensure the IAM user has not been associated.
400	ModelArts.4634	Failed to frequently reset the password.	Frequently resetting the password failed.	Request to reset the password up to once within 300s.
400	ModelArts.4635	The IAM account has been associated.	The IAM account has been associated.	Ensure the IAM account has not been associated.
400	ModelArts.4636	The task does not exist.	The task does not exist.	Ensure the task is available.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.4637	Deleting the default labeling task is forbidden.	Do not delete the default labeling task of a dataset.	When you create a dataset of the old version, the system automatically creates a labeling task for the dataset. If you delete the default labeling task, the dataset may become unavailable. To delete the task, delete the dataset.
400	ModelArts.4650	Interactive operations not found	Incorrect interactive_operations value.	Ensure that the interactive_operations value is correct.
400	ModelArts.4651	Get obs sample error	Failed to read the sample from OBS.	Ensure that the sample in OBS is correct.
400	ModelArts.4654	Inference execution error.	An error occurred during inference execution.	Check inference logs.
400	ModelArts.4656	Sample oversized.	The sample is oversized.	Ensure the size of a sample is not larger than 12,582,912 bytes.
400	ModelArts.4700	Task does not exist	The task ID does not exist.	Ensure that the task ID is correct.
400	ModelArts.4701	Can not start data analysis task, dataset version annotation type must be 'Image Classification' or 'Object Detection'	The dataset type does not support feature analysis.	Ensure that the dataset type is correct.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.4702	Can not start data analysis task, dataset version must be 'Default' format	The dataset format does not support feature analysis.	Use the default dataset format.
400	ModelArts.4703	Can not start data analysis task, dataset version must contains annotated samples	Feature analysis cannot be performed because there is no labeled sample in the dataset.	Ensure that there are labeled samples in the dataset.
400	ModelArts.4704	Currently unable to start data analysis task, Please try again later	Feature analysis cannot be performed because a dataset version is being published.	Try again later.
400	ModelArts.4706	Can not start data analysis task, dataset version status must be normal	Feature analysis cannot be performed because the dataset version is not in normal state.	Ensure that the current dataset version is in normal state.
400	ModelArts.4709	Data preprocessing task already exists	A training task whose name is the name value already exists.	Ensure that the name value is correct in the request for creating in the task.
400	ModelArts.4710	Dataset export file error	Failed to publish or export the dataset.	Ensure that the task export or publish parameters are correct.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.4711	Publishing requires splitting but annotated samples do not satisfied for splitting rules	Failed to publish because the dataset label samples do not meet splitting requirements.	Ensure that the number of dataset labels and the number of labeled samples meet the splitting requirements.
400	ModelArts.4712	Dataset publish version failed	Failed to publish the dataset version.	Ensure that the task export or publish parameters are correct.
400	ModelArts.4714	The number of dataset versions has reached the quota limit.	The number of dataset versions has reached the quota limit.	Ensure the number of dataset versions does not exceed the quota limit.
400	ModelArts.4715	Incorrect data analysis result.	The data analysis result is incorrect.	Check the result of data analysis for handling OBS errors.
400	ModelArts.4716	Abnormal AI Gallery.	AI Gallery is not running properly. Calling the AI Gallery API getVersionDetail failed.	Rectify the fault.
400	ModelArts.4717	Failed to execute the fulfillment task by AI Gallery.	AI Gallery failed to execute the fulfillment task.	Obtain details about the fulfillment task.
400	ModelArts.4760	Failed to encrypt data.	Encrypting data failed.	Check the input.
400	ModelArts.4800	The label task already exists	A labeling task whose name is the task_name value already exists.	Ensure that the task_name value is correct.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.4801	The label task not exists	The labeling task specified by the ID does not exist.	Ensure that the imported task ID is correct.
400	ModelArts.4820	Unsupported operation, label task type is %s	The labeling task type does not support the operation.	Ensure that the labeling task type is correct.
400	ModelArts.4822	The number of labelers must exceed 1	At least two annotators are available in the team.	Check the number of annotators in the team.
400	ModelArts.4823	Import to dataset version error	No data can be imported to a table dataset version.	Ensure that the dataset type is correct.
400	ModelArts.4824	Dataset version status must be normal	Abnormal dataset version state.	Check the version publish state of the dataset.
400	ModelArts.4825	Dataset version is empty	No sample is available in the dataset version.	Check the version publish result of the dataset.
400	ModelArts.4827	Dataset version delete failed	Failed to delete the dataset version because there is a labeling task based on the dataset version.	Check whether there is any labeling task that is created using the dataset version.
400	ModelArts.4828	Failed to delete the version.	The version in the current state cannot be deleted.	Change the version state.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.4851	Task version not found	The version specified by version_id does not exist.	Ensure that the version_id value is correct.
400	ModelArts.4852	Error in downloading task version logs.	An error occurred in downloading task version logs.	Check the training log URL.
400	ModelArts.4853	Insufficient quota for data feature tasks.	The quota for data feature tasks is insufficient.	Delete unnecessary tasks.
400	ModelArts.4854	Insufficient quota for data feature versions.	The quota for data feature versions is insufficient.	Delete unnecessary versions.
400	ModelArts.4860	Insufficient quota for real-time services.	The quota for real-time services is insufficient.	Check the remaining quota for real-time services.
400	ModelArts.4861	Insufficient quota for batch services.	The quota for batch services is insufficient.	Check the remaining quota for batch services.
400	ModelArts.4870	HBase error.	An error occurred in HBase.	Rectify the fault in batch deletion.
400	ModelArts.4902	Access prohibited because the app-auth API was not associated with APIG app %s.	The target app is not authorized.	Ensure AppCode is allowed to access the target service.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.4903	Access prohibited because the app-auth API was not associated with a signature key.	The app authentication API is not associated with a signature key.	Internal error. Contact technical support.
400	ModelArts.4904	Access prohibited because the app-auth API was not associated with signature key %s.	The app authentication API is not associated with the signature key.	Internal error. Contact technical support.
400	ModelArts.4905	Failed to check the authorization request header.	Checking the authorization request header failed.	Use the correct AppCode URL or contact technical support.
400	ModelArts.4907	Failed to submit the training job.	Submitting the training job failed.	Check whether the algorithm is available.
400	ModelArts.4909	Failed to execute the training job.	Executing the training job failed.	Check training logs to identify the failure cause.
400	ModelArts.4911	Failed to create the model import task.	Creating the model import task failed because the task ID is left blank.	Specify the task ID.
400	ModelArts.4913	Failed to execute the model import task.	Executing the model import task failed due to task timeout.	Identify the failure cause of task timeout.
400	ModelArts.4915	Failed to create the auto labeling task.	Creating the auto labeling task failed.	Identify the failure cause of task creation.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.4917	Failed to execute the auto labeling task.	Executing the auto labeling task failed due to task timeout.	Identify the failure cause of task timeout.
400	ModelArts.4921	Failed to execute the import task.	Executing the import task failed.	Identify the failure cause.
400	ModelArts.4927	Failed to submit hard example collection.	Submitting hard example collection failed because the task ID is left blank.	Specify the task ID.
400	ModelArts.4929	Failed to collect hard examples.	Collecting hard examples failed due to task timeout.	Identify the failure cause of task timeout.
400	ModelArts.4934	Invalid model.	The model is invalid.	Use a valid model.
400	ModelArts.4935	Failed to create the medical task.	Creating the medical task failed.	Identify the failure cause of task creation.
400	ModelArts.4951	Failed to create the auto inference task.	Creating the auto inference task failed.	Identify the failure cause of task creation.
400	ModelArts.4953	Failed to execute the auto inference task.	Executing the auto inference task failed due to task timeout.	Identify the failure cause of task timeout.
400	ModelArts.4963	Failed to execute the auto grouping task.	Executing the auto grouping task failed due to task timeout.	Identify the failure cause of task timeout.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.4991	Failed to obtain the job status.	The specifications have exceeded the quota for free specifications.	Switch to billed specifications or wait until resources are released.
400	ModelArts.4992	Resources sold out.	The resources have been sold out.	Use available resources.
400	ModelArts.4999	System error.	An error occurred in the system.	Wait until the UI adapts to the new mechanism.
400	ModelArts.6301	The instance already exists.	The instance already exists.	Enter another instance name.
400	ModelArts.6302	The instance count already reaches the maximum value.	The maximum number of instances has been reached.	Delete unnecessary instances.
400	ModelArts.6303	The field does not support sorting.	This field does not support sorting.	Delete the field from the sorting parameters.
400	ModelArts.6304	Please stop the instance before deleting.	Failed to delete the instance in the running state.	Refresh the page. Stop the instance and then delete it.
400	ModelArts.6305	The instance is already running.	The instance is in the running state.	Do not start it again.
400	ModelArts.6306	The instance is starting.	Other operations cannot be performed because the instance is being started.	Perform other operations later.
400	ModelArts.6307	The instance has already been stopped.	The instance has been stopped.	Refresh the page and view the instance state.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.6308	The instance is stopping.	The instance is being stopped.	Refresh the page and view the instance state.
400	ModelArts.6309	The instance does not exist.	The instance does not exist.	Ensure that the instance has been deleted.
400	ModelArts.6316	The param 'storage' is needed for creating notebook instance.	Incorrect parameters during instance creation.	Add the storage parameter.
400	ModelArts.6317	The 'path' parameter is required when the storage type is OBS.	Incorrect parameters during instance creation.	Add the path parameter.
400	ModelArts.6318	The param 'path' needs to end with /.	Incorrect parameters during instance creation.	Ensure that the value of the path parameter ends with a slash (/).
400	ModelArts.6319	There is no ak/sk in the global settings. Please add it by access ModelArts console.	The AK/SK have not been configured in the global settings of ModelArts.	Configure the AK/SK in the global settings of ModelArts.
400	ModelArts.6320	Access obs error. Reason is {}.	OBS access error.	Ensure that the OBS path is correct.
400	ModelArts.6323	The flavor is not supported, please check it.	The flavor is not supported.	Ensure that the flavor is correctly configured.
400	ModelArts.6324	This location type is not supported. Please check it.	The storage type is not supported.	Ensure that the storage type is correctly configured.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.6326	The instance is frozen. Check whether your account is in arrears in Billing Center.	The instance has been frozen.	Check the account balance and top up the account.
400	ModelArts.6327	The instance is not in running. Please refresh the page and start it.	The instance is not in the running state.	Refresh the page and try again.
400	ModelArts.6328	Sorry, you currently do not have permission for this flavor. Please apply it firstly.	You do not have permission to use the flavor.	Contact technical support.
400	ModelArts.6329	Sorry, this flavor specification is sold out. Please try others.	The flavor is sold out.	Use another flavor or try again later.
400	ModelArts.6330	The flavor type of profile does not match the type of flavor. Please check it.	The types are not matched.	Ensure that the flavor type matches the profile type.
400	ModelArts.6331	The evs volume size ranges from {} GB to {} GB.	The EVS disk size exceeds the upper limit.	Check the EVS disk size.
400	ModelArts.6332	Incorrect parameter type. The '{}' parameter must be of the '{}' type.	Parameter type error.	Ensure that the parameter type meets the requirements.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.6333	The notebook is being restored. Refresh the page and wait for several minutes. The fault may be caused by instance overload.	Notebook instance fault.	The instance is recovering. Try again later.
400	ModelArts.6334	Invalid value for name or description. The character `{}` is not allowed.	The description parameter is incorrect.	Enter a valid description value.
400	ModelArts.6335	`{}` is a required property.	Parameter missing.	Ensure that the corresponding parameter is available.
400	ModelArts.6336	`{}` property value is incorrect.	Incorrect parameter value.	Ensure that the corresponding parameter is correct.
400	ModelArts.6341	The repository does not exist or has been deleted.	The Git repository does not exist or has been deleted.	Check the corresponding parameter.
400	ModelArts.6343	The repository name already exists.	The Git repository already exists.	Check the corresponding parameter.
400	ModelArts.6344	Delete failed. Please delete the associated development environment instance first.	Failed to delete the repository.	Delete the corresponding notebook instance first.
400	ModelArts.6345	Currently the OBS type instance does not support git capability.	The Git repository cannot be used for OBS instances.	Use notebook instances of the EVS type.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.6346	The timestamp of auto stop must be later than the current timestamp.	This parameter cannot be modified.	Check the corresponding parameter and ensure that the automatic end time is later than the current time.
400	ModelArts.6347	When the instance is in the '{}' status, the '{}' parameter cannot be modified.	Failed to modify the parameter.	Check the corresponding parameter.
400	ModelArts.6348	The 'duration' parameter is mandatory for enabling auto stop.	The description parameter is missing.	Configure the duration parameter.
400	ModelArts.6349	The 'duration' parameter is incorrect. The value ranges from {} to {}.	The value of duration parameter exceeds the threshold.	Configure the duration parameter.
400	ModelArts.6350	Failed to access OBS because of incorrect AK/SK or insufficient permissions.	Access to OBS is rejected.	Ensure that the AK/SK is correct and you have the permission to access OBS.
400	ModelArts.6353	Failed to verify the AK/SK. Please check and try again.	Incorrect AK/SK.	Ensure that the AK/SK in the global settings of ModelArts is correct.
400	ModelArts.6354	The AK/SK do not belong to the user. Please check and enter the correct ones.	Incorrect AK/SK.	Ensure that the AK/SK in the global settings of ModelArts is correct.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.6355	The resource is initializing. Please wait one minute and try again.	Resources in the dedicated resource pool are being initialized.	Try again later.
400	ModelArts.6357	The operation is not allowed because another operation is being performed on the instance or the instance is in the target state.	The operation is not allowed.	Try again later.
400	ModelArts.6358	The path parameter is incorrect. It cannot be the root directory of an OBS bucket, but must be a specific directory in the OBS bucket.	The OBS path parameter is incorrect.	Ensure that the path parameter is correct.
400	ModelArts.6360	Failed to start the notebook due to insufficient background resources. Try again later.	Failed to start the notebook due to insufficient background resources. Try again later.	Try again later or contact technical support.
400	ModelArts.6361	ModelArts internal service or configuration error. Submit a service ticket to get professional technical support.	Internal error.	System error. Contact technical support.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.6371	Currently, notebooks support only OBS buckets whose Storage Class is Standard. Change the OBS bucket and try again.	Notebook instances support only standard OBS buckets.	Use standard OBS buckets.
400	ModelArts.6374	The credential can be added only when type is set to AK/SK or agency.	The authentication type can only be AK/SK or agency.	Ensure that the authentication type is AK/SK or agency.
400	ModelArts.6376	Please make sure agency exists.	The IAM agency does not exist.	Ensure that the agency exists on IAM.
400	ModelArts.6377	The user id cannot be left blank.	The user ID cannot be left blank.	Ensure that the user ID parameter is available.
400	ModelArts.6378	The agency name cannot be left blank.	The agency name cannot be left blank.	Ensure that the agency name parameter is available.
400	ModelArts.6379	No IAM agency created in Settings. Create an IAM agency on the ModelArts management console.	No agency is created in the global settings.	Create an agency in the global settings.
400	ModelArts.6400	Invalid request payload {0}, please check.	Invalid request payload {0}, please check.	Check the request body.
400	ModelArts.6528	No EVS available. Please try again later.	The EVS disk has been sold out.	Try again later or contact technical support.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.6600	Check whether the parameter is valid.	Failed to verify the parameters.	Ensure that the parameter is valid.
400	ModelArts.6651	Unsupport entity.	Unsupported project objects.	Ensure that the project object is supported.
400	ModelArts.6652	Please delete resources from the project first.	Failed to delete the project resources.	Delete the datasets and notebook instances under the project and ensure that all training jobs have been completed.
400	ModelArts.6690	The image size cannot exceed 8 MB.	The image size exceeds 8 MB.	Ensure that the image size does not exceed 8 MB.
400	ModelArts.6716	The specified flavor {0} is already sold out.	The specified flavor {0} is already sold out.	Select another flavor or contact technical support.
400	ModelArts.6737	Conflict tag {0} for image {1}, tag cannot be duplicated.	Conflict tag {0} for image {1}, tag cannot be duplicated.	Enter a proper image tag.
400	ModelArts.6763	Cannot manage the pool, The cluster version is earlier than v1.15.	Cannot manage the pool, The cluster version is earlier than v1.15.	Please upgrade the cluster version.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.6771	Cannot register the image since the size of the registered image {0}/{1}: {2} is {3} bytes, which has reached the maximum of {4} bytes. If the size of the image is too large, the image will fail to be decompressed	Cannot register the image since the size of the registered image {0}/{1}: {2} is {3} bytes, which has reached the maximum of {4} bytes. If the size of the image is too large, the image will fail to be decompressed	Check the image size.
400	ModelArts.6782	Mounting OBS bucket or folder does not support in current authentication mode, add AK/SK in ModelArts global settings to enabled it.	Mounting OBS bucket or folder does not support in current authentication mode, add AK/SK in ModelArts global settings to enabled it.	Please add AK/SK to ModelArts Global Settings.
400	ModelArts.6789	The current user does not have the permission to use the SSH key pair {0}, please update the key pair of the instance and try again.	The current user does not have the permission to use the SSH key pair {0}, please update the key pair of the instance and try again.	Update the key pair of the instance and restart the instance.

Status Code	Error Codes	Error Message	Description	Solution
400	ModelArts.6792	Instance {0} does not support to update flavor to {1}, please try another flavor.	Instance {0} does not support to update flavor to {1}, please try another flavor.	Select another flavor.
400	ModelArts.6797	Specified image {0} cannot be found.	Specified image {0} cannot be found.	Check the image access permission.
400	ModelArts.6870	Specified flavor {0} does not support current instance feature {1}.	Specified flavor {0} does not support current instance feature {1}.	Select a proper flavor.
401	ModelArts.0112	Failed to execute the policy.	The user role is not allowed to execute the policy.	Check user role permissions.
401	ModelArts.0201	Token doesn't exist.	The token is empty.	Add a correct token in the request header.
401	ModelArts.0202	Invalid token header because the token has expired.	The token has expired.	Use a valid token.
401	ModelArts.0203	Invalid token.	The token cannot be parsed.	Use a valid token.
401	ModelArts.0204	Failed to parse the token.	Parsing the token failed.	Use a valid token.
401	ModelArts.0205	No authorization request header.	The authorization request header is missing.	Use the correct AppCode URL or contact technical support.

Status Code	Error Codes	Error Message	Description	Solution
401	ModelArts.0206	Incorrect authorization request header format.	The authorization request header is in an incorrect format.	Use the correct AppCode URL or contact technical support.
401	ModelArts.0207	Token must contain project info of current region.	Add project information to the token.	Add project information to the token.
401	ModelArts.0208	Failed to check the authorization request header.	Checking the authorization request header failed.	Use the correct AppCode URL or contact technical support.
401	ModelArts.0209	No key appld in header.	The appld request header is missing.	Use the correct APIG URL.
401	ModelArts.0210	No key apild in header.	The apild request header is missing.	Use the correct APIG URL.
401	ModelArts.0212	Invalid token header because the token does not contain a project.	The token does not contain any project information.	Use a project-level token.
401	ModelArts.0428	The user authorization does not exist.	The user authorization does not exist.	Check whether the user agency is authorized.
401	ModelArts.3801	User credential (AK and SK) does not exist.	The user certificate does not exist.	Add the AK/SK again.
401	ModelArts.4117	User is unverified.	User is unverified.	Please check the user has been authorized by IAM.

Status Code	Error Codes	Error Message	Description	Solution
401	ModelArts.4119	Invalid RBAC.	The project ID in the token is left blank.	Add the correct project ID in the token.
401	ModelArts.4127	Token authentication expired.	The token authentication has expired.	Obtain the user token again.
401	ModelArts.4201	Invalid token.	The token is invalid.	Check the token of the account to which the user belongs.
401	ModelArts.4203	Invalid request because %s.	You do not have the access permission.	Ensure you have the access permission.
401	ModelArts.4504	Iam error	IAM error.	Ensure that the AK/SK or token is correct.
401	ModelArts.4505	The operation on the IAM user is not allowed.	No operation is allowed on the IAM user list.	Do not perform operations on the IAM user list.
401	ModelArts.4506	Failed to check the IAM service ticket.	Checking the IAM service ticket failed.	Check the IAM service ticket.
401	ModelArts.4626	Failed to update the team or team manager.	Updating the team or team manager failed.	Identify the failure cause.
401	ModelArts.4629	Failed to delete the member.	Deleting a member failed.	Identify the failure cause.
401	ModelArts.4630	No permission.	The user does not have permission to perform the operation.	Ensure the user has permission to perform the operation.
401	ModelArts.4632	Unregistered member.	The member has not been registered.	Ensure the member is registered.

Status Code	Error Codes	Error Message	Description	Solution
401	ModelArts.4713	Forbidden operation.	This operation is not allowed because there are uncompleted asynchronous tasks of dataset hashing.	Ensure all asynchronous tasks of dataset hashing are complete.
401	ModelArts.4750	Error in reading files from OBS.	An error occurred in reading files from OBS.	Identify the failure cause.
401	ModelArts.4804	Failed to create a label task by the member.	The team member failed to create a label task.	Do not create a label task as a team member.
401	ModelArts.4826	OBS KMS encryption buckets are not supported.	OBS KMS encryption buckets are not supported.	Do not use OBS KMS encryption buckets.
401	ModelArts.4901	Failed to prepare a training job.	Preparing a training job failed.	Authorizing onObs failed. Check the IAM proxy or OBS.
401	ModelArts.4961	Failed to submit an auto grouping job.	Submitting an auto grouping job failed.	Authorizing onObs failed. Check the IAM proxy or OBS.
401	ModelArts.6201	The user's account has been suspended.	The user account is frozen.	Check the account balance and top up the account.
401	ModelArts.6203	The user's account has been restricted.	The user account is restricted.	Check the account balance and top up the account.
401	ModelArts.6608	Please refresh user info.	Failed to obtain the user information.	Refresh the user information.

Status Code	Error Codes	Error Message	Description	Solution
401	ModelArts.6620	Please refresh the user token.	Invalid user token.	Refresh the user token.
403	ModelArts.0108	You are not authorized to perform the ({0}) operation.	You do not have permission to perform {0}.	Check whether you are authorized to perform operations on OBS or APIs.
403	ModelArts.0112	Policy doesn't allow {} {} to be performed.	The permission on the workspace is restricted.	Ensure that you have the permission on the workspace.
403	ModelArts.0203	Invalid token.	Invalid token.	Obtain a new token and try again.
403	ModelArts.0206	Invalid AK/SK.	Invalid AK/SK.	Check whether the token is valid.
403	ModelArts.0210	The project id in the request URL does not match the token.	The project ID in the request URL and the token do not match.	Use the correct project ID to generate a token.
403	ModelArts.2784	InvalidToken	Invalid token.	Check the request token.
403	ModelArts.2785	Authorization NotFound	You cannot use ModelArts because you have not authorized ModelArts.	Please go to the Settings page to authorize ModelArts.
403	ModelArts.2852	FederationOperationDeny	Federation participant does not have permission to do this.	Contact the federation owner to do this.

Status Code	Error Codes	Error Message	Description	Solution
403	ModelArts.3010	Failed to copy model, you don't have permission to copy the model ({0}).	You do not have the permission to copy model {0}.	Check whether the entered token or model ID is correct.
403	ModelArts.3075	Model ({0}) is a subscription model and cannot create a new version.	Failed to create a new version because model {0} has been subscribed.	No new version can be created for a subscribed model.
403	ModelArts.3555	Forbidden to access ECS.	You do not have the permission to access the ECS.	Authorize access to the ECS and try again.
403	ModelArts.3936	The app-auth API {id} does not belong to service {id}.	The API {ID} that supports application authentication does not belong to the service {ID}.	Check the request parameter or contact service O&M personnel.
403	ModelArts.4116	User has been owed, please check the account status and balance.	User has been owed, please check the account status and balance.	Please check account balance.
403	ModelArts.4118	User has been suspended, please check the account balance.	User has been suspended, please check the account balance.	Please check the account balance.
403	ModelArts.4310	OBS action is forbidden, please check iam agency or OBS and so on	You do not have permission to access OBS.	Grant the permission to access OBS.
403	ModelArts.4335	Iam agency is invalid	Invalid IAM agency.	Check the IAM permission.

Status Code	Error Codes	Error Message	Description	Solution
403	ModelArts.4336	The user hasn't permission	Restricted user permission.	Check the user permission.
403	ModelArts.4348	Check dli agency failed	Failed to create the agency for checking DLI.	Check the agency permission for checking DLI.
403	ModelArts.4419	File is too large	The size of the file imported from OBS or to be parsed has reached the maximum allowed limit.	Ensure that the file is correct.
403	ModelArts.4500	The number of iam agencies has reached the maximum	The number of IAM agencies exceeded the upper limit.	Delete unused IAM agencies.
403	ModelArts.4501	The iam agency create action is forbidden	Failed to create the IAM agency.	Ensure that you have permission to create an IAM agency.
403	ModelArts.4600	The worker action is forbidden	The team member cannot perform the operation.	Check whether the team member has permission to perform the operation.
403	ModelArts.4613	Task not finish	Failed to obtain the task result because the task execution has not been completed.	Try again later.
403	ModelArts.4803	Dataset has unfinished label tasks, the operation is forbidden	Failed to delete data because the labeling task in the dataset has not been completed.	Ensure that all labeling tasks in the dataset are complete.

Status Code	Error Codes	Error Message	Description	Solution
403	ModelArts.4821	Task is publishing, the operation is forbidden	The auto labeling task cannot be started because a labeling task is being published.	Try again after the labeling task is published.
403	ModelArts.4850	Process task is initializing, the operation is forbidden	Failed to delete the task because the processor task is being initialized.	Check the task status.
403	ModelArts.6653	Contact the project owner.	You are not authorized to operate the project.	Contact the project owner to obtain the authorization.
404	ModelArts.2786	TokenProjectIDNotMatch	The project ID (%s) in the request does not match that(%s) in the token.	Change the request token or change the project ID in the request.
404	ModelArts.2836	PathNotFound	Path not found.	Please confirm that the PATH is correct
404	ModelArts.2881	ImageSaveJobNotExist	ImageSaveJob is not exist	Please check task_id
404	ModelArts.2890	PoolNotFound	Pool (%s) does not exist	Please check whether the pool exists
404	ModelArts.3001	Model ({0}) does not exist.	Model {0} does not exist.	Check whether the entered model ID is correct.
404	ModelArts.3003	Failed to get model names list.	Failed to obtain the model name list.	Check whether the account is correct.
404	ModelArts.3019	Infer format ({0}) does not exist.	Input/Output mode {0} does not exist.	Change the input/output mode.

Status Code	Error Codes	Error Message	Description	Solution
404	ModelArts.3021	Template ({0}) does not exist.	Template {0} does not exist.	Select another template.
404	ModelArts.3071	The subscription ({0}) does not exist.	Subscription {0} does not exist.	Subscribe to the model and try again.
404	ModelArts.3080	Model optimization job ({0}) does not exist.	Model optimization task {0} does not exist.	The task is unavailable.
404	ModelArts.3502	Service {name} does not exist.	The service {name} does not exist.	Ensure that the service name is correct.
404	ModelArts.3507	Model {name} does not exist.	The model {ID} does not exist.	Ensure that the model ID is correct.
404	ModelArts.3923	Application {id} does not exist.	The application {ID} does not exist.	Ensure that the application ID is correct.
404	ModelArts.3935	The app-auth API {id} does not exist.	The API {ID} that supports application authentication does not exist.	Ensure that the interface ID is correct.
404	ModelArts.4901	The app-auth API does not exist.	The app authentication API does not exist.	Ensure the app URL is correct.
404	ModelArts.6623	Ensure that the dataset already exists.	Failed to find the corresponding dataset.	Ensure that the dataset already exists.
404	ModelArts.6624	Ensure that the storage path already exists.	Failed to find the corresponding storage path.	Ensure that the storage path already exists.

Status Code	Error Codes	Error Message	Description	Solution
404	ModelArts.6650	Ensure that the project already exists.	Failed to find the corresponding project.	Ensure that the project already exists.
405	ModelArts.2803	MethodNotAllowed	Method not allowed.	Please confirm that the HTTP Method is correct
408	ModelArts.4205	Connection timed out.	Accessing the backend server timed out.	Try again later.
408	ModelArts.6100	The request timed out. Please try again.	Timeout.	System error. Contact technical support.
409	ModelArts.2796	NotAllowedToTerminate	The job cannot be terminated.	Check whether the job status is valid.
409	ModelArts.2813	NotAllowedToRestart	The job cannot be restarted.	Check whether the action type is valid.
409	ModelArts.2878	JobStatusChanged	Job status changed	Please try again later.
409	ModelArts.3002	Model ({0}, {1}) already exists.	Model ({0}, {1}) already exists.	Change the model name or version and try again.
409	ModelArts.3020	Infer format ({0}) already exists.	Input/Output mode {0} already exists.	Change the input/output mode.
409	ModelArts.3022	Template ({0}) already exists.	Template {0} already exists.	Select another template.
409	ModelArts.3073	model ({0}) already exists, no subscription required.	Model {0} already exists.	Do not subscribe to another model with the same name.
409	ModelArts.3503	Service {name} already exists.	The service {name} already exists.	Change the service name and try again.

Status Code	Error Codes	Error Message	Description	Solution
409	ModelArts.3929	Failed to create the application because {reason}.	Failed to create the application. Fault cause: {fault cause}.	Locate the fault based on the failure cause.
413	ModelArts.4104	Invalid request body because the size exceeds the maximum size allowed.	The size of the request body is invalid.	Use a request body smaller than 12 MB.
417	ModelArts.6654	Project with the same name already exists.	A project with the same name already exists.	Use another project name.
429	ModelArts.2799	TooManyRequests	services too many requests	Please try again later
429	ModelArts.4206	Excessive number of requests to %s.	The maximum number of requests has been reached.	Try again later.
429	ModelArts.4395	Too many dataset requests	Frequent dataset or labeling task creation in a unit time.	Try again later.
429	ModelArts.6101	The system is busy now. Please try again later.	The system is busy.	The system is busy. Please try again. If the retry still fails, contact technical support.
500	ModelArts.0010	Internal error.	Internal error.	Contact R&D and O&M personnel.
500	ModelArts.0109	Unauthorized account.	Unauthorized account.	Only the administrator can call alarm APIs.
500	ModelArts.2950	Unknown	Unknown error.	System error. Contact technical support.

Status Code	Error Codes	Error Message	Description	Solution
500	ModelArts.2951	DatasetService	Dataset service error.	System error. Contact technical support.
500	ModelArts.2952	ExtServiceDB	Database service error.	System error. Contact technical support.
500	ModelArts.2953	ExtServiceIAM	IAM service error.	System error. Contact technical support.
500	ModelArts.2954	ExtServiceWorkspace	Workspace service error.	System error. Contact technical support.
500	ModelArts.2955	ExtServiceAOM	AOM service abnormal	System error. Contact technical support.
500	ModelArts.2956	ExtServiceAIG	AIG service abnormal	System error. Contact technical support.
500	ModelArts.2957	ExtServiceSMN	SMN service abnormal	System error. Contact technical support.
500	ModelArts.2958	ExtServiceOBS	OBS service abnormal	System error. Contact technical support.
500	ModelArts.2959	ExtServiceMAOS	MAOS service abnormal	System error. Contact technical support.
500	ModelArts.3006	Failed to publish model.	Failed to publish the model.	The model metadata failed to be stored to the database. Contact R&D and O&M personnel.
500	ModelArts.3009	Failed to delete model, model ({0}) already deploy service.	Failed to delete model {0} because it has been deployed as a service.	Delete the deployed service and try again.

Status Code	Error Codes	Error Message	Description	Solution
500	ModelArts.3043	User ({0}) does not have obs: object: PutObjectAcl permission.	User {0} does not have the obs:object:PutObjectAcl permission.	Add the permission and try again.
500	ModelArts.3044	Model file ({0}) is larger than 20G and cannot be imported.	Failed to import model file {0}.	Ensure that the file is at most 20 GB in size.
500	ModelArts.3045	Parameter ({0}) is null.	Parameter {0} is left blank.	Check the parameter setting as prompted.
500	ModelArts.3047	ExeML model({0}) cannot be converted.	Failed to convert model {0}.	ExeML models cannot be converted.
500	ModelArts.3048	Model({0}) is being imported and cannot be converted.	Failed to convert model {0} because the model status is abnormal.	Convert the model after the model status changes to normal.
500	ModelArts.3049	Cannot convert models belonging to other users.	Models of other users cannot be converted.	You can only convert your models.
500	ModelArts.3050	Get user temporary credential failed.	Failed to obtain the temporary user certificate.	Check whether the token is valid.
500	ModelArts.3052	Chip Type ({0}) not support.	Chip type {0} is not supported.	Check whether the chip is of Ascend, GPU, Arm, or general type.
500	ModelArts.3053	Model Type ({0}) not support.	Model type {0} is not supported.	This type of model is not supported.

Status Code	Error Codes	Error Message	Description	Solution
500	ModelArts.3054	Task ({0}) does not exist.	Task {0} does not exist.	The task is unavailable.
500	ModelArts.3055	Task ({0}) is running cannot be deleted.	Failed to delete task {0} because it is running.	Delete the task after it is complete.
500	ModelArts.3056	Task name ({0}) not meeting the specification.	Task name {0} does not comply with specifications.	Ensure that a task name starts with a lowercase letter, ends with a lowercase letter or digit, and contains 2 to 24 characters, including lowercase letters, digits, and hyphens (-).
500	ModelArts.3057	Task description ({0}) not meeting the specification.	Task description {0} does not comply with specifications.	Ensure that task description contains 1 to 100 characters and cannot contain the following characters: &,"<>=.
500	ModelArts.3058	Task input ({0}) not meeting the specification.	Task input {0} does not comply with specifications.	Perform operations according to the error message.
500	ModelArts.3059	Task output ({0}) not meeting the specification.	Task output {0} does not comply with specifications.	Ensure that the task output is a valid HTTP or HTTPS address.
500	ModelArts.3060	Task spec ({0}) not meeting the specification.	Advanced task option {0} does not comply with specifications.	Ensure that the advanced option does not contain the following characters: ()&\$? <>. Ensure that the value is less than 4000.

Status Code	Error Codes	Error Message	Description	Solution
500	ModelArts.3061	get Task ({0}) log parameter not meeting the specification.	Log parameters of task {0} do not comply with specifications.	Ensure that the offset or lines value is greater than or equal to 0.
500	ModelArts.3062	Task ({0}) exists.	Task {0} already exists.	Change the name and try again.
500	ModelArts.3064	DL Framework Type ({0}) not support.	Deep learning framework {0} is not supported.	This type of framework is not supported.
500	ModelArts.3065	The image of the model is being built so the runtime cannot be updated.	Failed to update runtime because the model image is being created.	Wait until the image is created.
500	ModelArts.3301	Failed to build image.	Failed to create the image.	Check the image, or contact R&D and O&M personnel to rectify the fault.
500	ModelArts.3501	Failed to deploy the service.	Failed to deploy the service.	Contact service O&M personnel.
500	ModelArts.3504	Failed to delete the service.	Failed to delete the service.	Contact service O&M personnel.
500	ModelArts.3505	Failed to update the service.	Failed to query the service.	Contact service O&M personnel.
500	ModelArts.3506	Model {id} does not exist.	Failed to update the service.	Contact service O&M personnel.
500	ModelArts.3508	Failed to query the service monitoring information.	Failed to obtain the service monitoring information.	Contact service O&M personnel.

Status Code	Error Codes	Error Message	Description	Solution
500	ModelArts.3509	Failed to query edge nodes.	Failed to query the node.	Contact service O&M personnel.
500	ModelArts.3511	Failed to query edge tags.	Failed to query the edge label.	Contact service O&M personnel.
500	ModelArts.3537	Failed to handle the CBC request.	Failed to process the CBC request.	Contact service O&M personnel.
500	ModelArts.3539	Failed to initialize service URL {url}.	Failed to initialize the service URL {URL}.	Ensure that the service URL is correct.
500	ModelArts.3543	Invalid CBC request.	The CBC request is invalid.	Try again or contact service O&M personnel.
500	ModelArts.3546	Failed to create the cluster.	Failed to create the dedicated resource pool.	Contact service O&M personnel.
500	ModelArts.3579	Failed to query the model information.	Failed to query the model information.	Contact service O&M personnel.
500	ModelArts.3581	Failed to create edge certificates.	Failed to create the edge certificate.	Contact service O&M personnel.
500	ModelArts.3801	invalid token, can not get template ak/sk.	Failed to obtain the template AK/SK due to an invalid token.	Obtain a new token and try again.
500	ModelArts.3802	Failed to create API group {name} because {reason}.	Failed to create the API group {name}. Fault cause: {fault cause}.	Locate the fault based on the failure cause.

Status Code	Error Codes	Error Message	Description	Solution
500	ModelArts.3803	Failed to register API {name} because {reason}.	Failed to register the API {name}. Fault cause: {fault cause}.	Locate the fault based on the failure cause.
500	ModelArts.3804	upload file to obs failed.	Failed to upload the file to OBS.	Contact R&D and O&M personnel.
500	ModelArts.3805	Failed to create AI service {name} because {reason}.	Failed to create the AI service {name}. Fault cause: {fault cause}.	Locate the fault based on the failure cause.
500	ModelArts.3806	Failed to update AI service {name} because {reason}.	Failed to update the AI service {name}. Fault cause: {fault cause}.	Locate the fault based on the failure cause.
500	ModelArts.3807	Failed to delete AI service {name} because {reason}.	Failed to delete the AI service {name}. Fault cause: {fault cause}.	Locate the fault based on the failure cause.
500	ModelArts.3808	Failed to create AI subservice {name} because {reason}.	Failed to create the sub-service {name}. Fault cause: {fault cause}.	Locate the fault based on the failure cause.
500	ModelArts.3809	Failed to update AI subservice {name} because {reason}.	Failed to update the sub-service {name}. Fault cause: {fault cause}.	Locate the fault based on the failure cause.
500	ModelArts.3810	Failed to create the agency because {reason}.	Failed to create the agency. Fault cause: {fault cause}.	Locate the fault based on the failure cause.

Status Code	Error Codes	Error Message	Description	Solution
500	ModelArts.3819	Failed to delete the agency definition.	Failed to delete the service agency definition.	Contact service O&M personnel.
500	ModelArts.3858	Failed to bind task type to API {id} because {reason}.	Failed to bind task type to API {ID}. Fault cause: {fault cause}.	Locate the fault based on the failure cause.
500	ModelArts.3859	Failed to unbind task type from API {id} because {reason}.	Failed to unbind the task type from API {ID}. Fault cause: {fault cause}.	Locate the fault based on the failure cause.
500	ModelArts.3860	Failed to create the task type because {reason}.	Failed to create the task type. Fault cause: {fault cause}.	Locate the fault based on the failure cause.
500	ModelArts.3861	Failed to update the task type because {reason}.	Failed to update the task. Fault cause: {fault cause}.	Locate the fault based on the failure cause.
500	ModelArts.3891	Failed to operate the serverless cluster because {reason}.	Failed to operate the serverless cluster. Fault cause: {fault cause}.	Locate the fault based on the failure cause.
500	ModelArts.3893	Failed to forward the CBC request to the cloud service.	Failed to forward the CBC request to the cloud service.	Contact service O&M personnel.
500	ModelArts.3901	license id ({0}) is not exist.	License ID {0} does not exist.	Check whether the license ID is valid.

Status Code	Error Codes	Error Message	Description	Solution
500	ModelArts.3902	Failed to get license info because ({0}).	Failed to obtain the license because {Reason}.	Locate the fault based on the failure cause.
500	ModelArts.3903	Renew license failed because ({0}).	Failed to renew the license because {Reason}.	Locate the fault based on the failure cause.
500	ModelArts.3904	Upload data failed because ({0}).	Failed to report the used quota because {Reason}.	Locate the fault based on the failure cause.
500	ModelArts.3905	Failed to get app key.	Failed to obtain the app key.	Contact R&D and O&M personnel.
500	ModelArts.3906	Failed to create license because ({0}).	Failed to create the license because {Reason}.	Locate the fault based on the failure cause.
500	ModelArts.3907	Failed to delete license because ({0}).	Failed to delete the license because {Reason}.	Locate the fault based on the failure cause.
500	ModelArts.3908	Failed to get CBC resources.	Failed to obtain CBC resources.	Check the license ID or the associated license configuration.
500	ModelArts.3909	access CBC failed.	Failed to access CBC.	Contact R&D and O&M personnel.

Status Code	Error Codes	Error Message	Description	Solution
500	ModelArts.3910	Check CBC auth failed because ({0}).	Failed to authorize CBC operations because {Reason}.	Locate the fault based on the failure cause. Renewal authentication fails because cloudservicetype is not provided. Unsubscription authentication fails because an instance is being deployed.
500	ModelArts.3921	Failed to create the app-auth API because {reason}.	Failed to register the API that supports application authentication. Fault cause: {fault cause}.	Locate the fault based on the failure cause.
500	ModelArts.3924	Failed to create the application code for application {id} because {reason}.	Failed to create the AppCode of the application {ID}. Fault cause: {fault cause}.	Locate the fault based on the failure cause.
500	ModelArts.3925	Failed to delete app-auth api {id} because {reason}.	Failed to delete API {ID} that supports application authentication. Fault cause: {fault cause}.	Locate the fault based on the failure cause.
500	ModelArts.3926	Failed to reset the secret of application {id} because {reason}.	Failed to reset the private key of the application {ID}. Fault cause: {fault cause}.	Locate the fault based on the failure cause.

Status Code	Error Codes	Error Message	Description	Solution
500	ModelArts.3927	Failed to reset the application code of application {id} because {reason}.	Failed to reset the AppCode of the application {ID}. Fault cause: {fault cause}.	Locate the fault based on the failure cause.
500	ModelArts.3928	Failed to obtain the app-auth API information because {reason}.	Failed to obtain information about the API that supports application authentication. Fault cause: {fault cause}.	Locate the fault based on the failure cause.
500	ModelArts.3932	Failed to obtain the application information because {reason}.	Failed to obtain the application information. Fault cause: {fault cause}.	Locate the fault based on the failure cause.
500	ModelArts.3933	Failed to delete application {id} because {reason}.	Failed to delete the application {ID}. Fault cause: {fault cause}.	Locate the fault based on the failure cause.
500	ModelArts.3934	Failed to delete the application code of application {id} because {reason}.	Failed to delete the AppCode of the application {ID}. Fault cause: {fault cause}.	Locate the fault based on the failure cause.
500	ModelArts.3937	Failed to authorize the app-auth API to the application because {reason}.	Failed to authorize the API to the application. Fault cause: {fault cause}.	Locate the fault based on the failure cause.

Status Code	Error Codes	Error Message	Description	Solution
500	ModelArts.3938	Failed to unbind the app-auth API from the application because {reason}.	Failed to cancel the authorization from the API to the application. Fault cause: {fault cause}.	Locate the fault based on the failure cause.
500	ModelArts.4302	Gateway forwarding error because %s.	Accessing the service is rejected.	Ensure the service to be accessed is running properly.
500	ModelArts.4354	Insufficient quota	The number of created datasets has reached the maximum allowed limit.	Delete unused datasets or apply for a higher quota.
500	ModelArts.4360	Sync data source error	Failed to synchronize the data source.	Check the synchronization task status and result.
500	ModelArts.4367	Update dataset state failed	Failed to delete the dataset.	Check the dataset status.
500	ModelArts.4373	No migratory job	Failed to create the dataset migration task.	Check whether the source dataset of the migration task is valid.
500	ModelArts.4424	Delete sample failed	Failed to delete the dataset sample.	Ensure that the sample to be deleted is correct and that there is unfinished import or synchronization task.
500	ModelArts.4503	Failed to obtain the backend service because %s.	The service does not exist.	Ensure the service to be accessed is available.

Status Code	Error Codes	Error Message	Description	Solution
500	ModelArts.4505	Internal backend service error because reading timed out.	Internal server error.	Ensure the service to be accessed is running properly.
500	ModelArts.4614	Task failed	Failed to run the task.	Check the execution result of the processor task.
500	ModelArts.4801	Failed to obtain the consul %s.	Obtaining service instances for Consul failed.	Internal error. Contact technical support.
500	ModelArts.6102	The server has encountered an error and fails to process the request. Please try again later or submit a service ticket.	Internal error.	System error. Contact technical support.
500	ModelArts.6200	No resources available. Please try again later or submit a service ticket.	Internal error.	System error. Contact technical support.
500	ModelArts.6691	System error. Check the service status.	The algorithm service is abnormal.	System error. Check the service state.
500	ModelArts.6692	System error. Check the service status.	The dataset service is abnormal.	System error. Check the service state.
500	ModelArts.6699	Internal error.	Network error.	Check network connectivity.
501	ModelArts.4383	Unsupported operation, dataset %s type is %s	The operation cannot be performed in the dataset.	Check whether the dataset supports the current operation.

Status Code	Error Codes	Error Message	Description	Solution
501	ModelArts.4385	Auto deploy task exist	Failed to start the deploy task because an auto deploy task is being executed in the dataset.	Start a new deploy task after the current task is complete.
501	ModelArts.4386	Auto labeling task exist	Failed to start the auto labeling task because an auto labeling task is being executed in the dataset.	Start a new auto labeling task after the current task is complete.
501	ModelArts.4387	Pre-label task exist	Failed to start the pre-labeling task because a pre-labeling task is being executed in the dataset.	Start a new pre-labeling task after the current task is complete.
501	ModelArts.4394	Task is running	Failed to obtain the migration result because a migration task is being executed in the dataset.	Wait until the migration task is complete.
503	ModelArts.4302	Gateway forwarding error because %s.	The backend service is unavailable.	Internal error. Contact technical support.
503	ModelArts.4700	Unavailable gateway.	The gateway is being stopped.	The platform is being upgraded. Try again later.
504	ModelArts.4503	Failed to obtain the backend service because %s.	The response from the backend service timed out.	Ensure the service to be accessed is running properly.

11.3 Obtaining a Project ID and Name

Scenarios

A project ID or name is required for some requests when an API is called. Therefore, obtain the project ID and name before calling the API. Use either of the following methods:

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

Obtaining a Project ID and Name from the Console

To do so, perform the following operations:

1. Log in to the console.
2. In the upper right corner, click your account avatar icon and choose **My Settings** from the drop-down list.
3. On the **My Settings** page, go to the **Project List** tab page, which is displayed by default. View the project ID and name in the project list.

Obtaining a Project ID by Calling an API

The API for obtaining a project ID is **GET <https://{iam-endpoint}/v3/projects>**. To obtain *{iam-endpoint}*, see [Request URI Endpoint](#).

The following is an example response. For example, if ModelArts is deployed in the **xxx** region, the value of **name** in the response body is **xxx**. The value of **id** in **projects** is the project ID.

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

11.4 Obtaining an Account Name and ID

When you call APIs, certain requests require the account name and ID. To obtain an account name and ID, do as follows:

1. Sign up and log in to the console.
2. Hover the cursor on the username and choose **My Credentials** from the drop-down list.

On the **API Credentials** page, view the account name and ID.

Figure 11-1 Viewing the user name and ID



11.5 Obtaining a Username and ID

When you call APIs, certain requests require the username and ID. To obtain a username and ID, do as follows:

1. Log in to the management console after registration.
2. In the upper right corner, click your account avatar icon and choose **My Settings** from the drop-down list.

On the **My Settings** page, view the username and ID.