Application Orchestration Service

API Reference

Date   2020-11-05

Contents

1 Before You Start

1.1 Overview

Welcome to the Application Orchestration Service (AOS). AOS enables enterprises to automate application cloudification and orchestrate mainstream cloud services. It also supports service provisioning, and application creation, replication, and migration in a few clicks.

This document describes how to use APIs to perform operations on AOS, such as template creation, deletion, and query. For details about all supported operations, see **API Overview**.

If you plan to call AOS APIs, ensure that you are familiar with AOS concepts.

1.2 API Calling

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

1.3 Endpoints

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

1.4 Constraints

- The number of AOS resources that you can create is determined by your quota.
- For more information on constraints, see the description of specific APIs.

1.5 Concepts

- Account

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

- User

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

  API authentication requires information such as the account name, username, and password.

- Region

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

- AZ

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

- Project

  A project corresponds to a region. Default projects are defined to a group and have physically isolated resources (including computing, storage, and network resources) across regions. Users can be granted permissions in a default project to access all resources in the region under their accounts. If you need more refined access control, create subprojects under a default project and create resources in subprojects. Then you can assign users the permissions required to access only the resources in the specific subprojects.

  Figure 1-1 Project isolation model
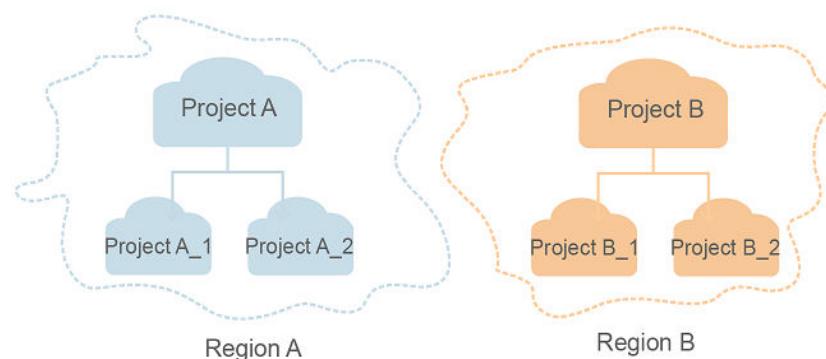
2 API Overview

Category	Description
Template management APIs	AOS template management APIs, which include APIs for creating, querying, updating, downloading, and deleting AOS templates. These APIs enable you to create, update, and delete templates; query templates, template lists, and template inputs.
Stack management APIs	AOS stack management APIs, which include APIs for creating, querying, and deleting stacks. These APIs enable you to create and delete stacks, execute stack lifecycle; query stacks, stack lists, stack elements, stack element lists, stack input and output, stack execution records, and lists of stack execution records.

Template Management APIs

AOS template management APIs include APIs for creating, querying, updating, and deleting AOS templates. These APIs enable you to create, update, download, and delete templates; query templates, template lists, and template inputs.

Table 2-1 Template management APIs

API	Description
Creating a template	This API is used to create a template by uploading a local template to the server.
Querying a template list	This API is used to query a template list based on the parameters provided.

API	Description
Updating a template	This API is used to update a template through either uploading local binary files or providing a template URL. The update can be performed only when the template has not been used to create any stacks or stacks created from this template have been deleted.
Deleting a template	This API is used to delete a specific template. The deletion can be performed only when the template has not been used to create a stack or stacks created from this template have been deleted.
Downloading a template	This API is used to download a specified template.
Querying a template	This API is used to query details about a specified template, including the template name, version, description, creation time, and update time.
Querying the input parameters of a template	This API is used to query the input parameters of a specific template.

Stack Management APIs

AOS stack management APIs include APIs for creating, querying, and deleting stacks. These APIs enable you to create and delete stacks, execute stack lifecycle; query stacks, stack lists, stack elements, stack element lists, stack input and output, stack execution records, and lists of stack execution records.

Table 2-2 Stack management APIs

API	Description
Creating a stack	This API is used to create a stack. Stack inputs consist of templates and input parameters. ● A template defines the framework of a stack, and determines the structure of the nodes in the stack, the relationships between nodes, and the value or source of the attributes of each node. ● Input parameters are one of the sources of the node attribute values in the template. They are defined in the input fields of the template and are triggered by the **get_input** function in the template.
Deleting a stack	This API is used to delete a specific stack.

API	Description
Executing a stack lifecycle	This API is used to perform lifecycle operations for a specific stack.
Querying the stack list	This API is used to query a stack list.
Querying a stack	This API is used to query details about a specified stack, including the stack name, stack description, template ID, template name, and stack status.
Querying a stack element list	This API is used to query a stack element list.
Querying a stack element	This API is used to query details about an element in a stack.
Querying a stack output	This API is used to query the output of a specified stack.
Querying a stack input	This API is used to query the input of a specific stack.
Querying an execution record of a stack	This API is used to query a stack execution record.
Querying a stack execution record list	This API is used to query the latest execution record list of a stack.

3 Calling APIs

3.1 Making an API Request

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

Request URI

A request URI is in the following format:

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

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

Table 3-1 URI parameter description

Parameter	Description
URI-scheme	Protocol used to transmit requests. All APIs use HTTPS.
Endpoint	Domain name or IP address of the server bearing the REST service. The endpoint varies between services in different regions. It can be obtained from **Regions and Endpoints**. For example, the endpoint of IAM in the **UAE-Abu Dhabi** region is **iam.ae-ad-1.myhuaweicloud.com**.
resource-path	Access path of an API for performing a specified operation. Obtain the path from the URI of an API. For example, the **resource-path** of the API used to obtain a user token is **/v3/auth/tokens**.

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

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

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

Figure 3-1 Example URI
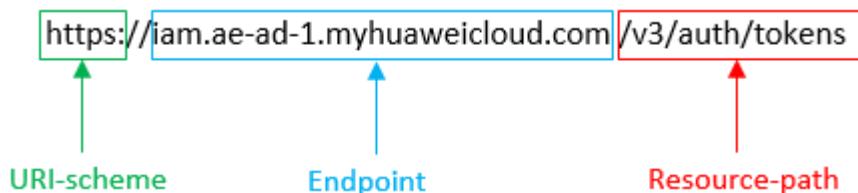
NOTE

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

Request Methods

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

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.

Method	Description
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.ae-ad-1.myhuaweicloud.com/v3/auth/tokens

Request Header

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

Common request header fields are as follows.

Table 3-3 Common request header fields

Parameter	Description	Mandatory	Example Value
Host	Specifies the server domain name and port number of the resources being requested. The value can be obtained from the URL of the service API. The value is in the format of *Hostname:Port number*. If the port number is not specified, the default port is used. The default port number for **https** is **443**.	No This field is mandatory for AK/SK authentication.	code.test.com or code.test.com: 443
Content-Type	Specifies the type (or format) of the message body. The default value **application/json** is recommended. Other values of this field will be provided for specific APIs if any.	Yes	application/json
Content-Length	Specifies the length of the request body. The unit is byte.	No	3495

Parameter	Description	Mandatory	Example Value
X-Project-Id	Specifies the project ID. Obtain the project ID by following the instructions in **Obtaining a Project ID**.	No	e9993fc787d94b6c886cbaa340f9c0f4
X-Auth-Token	Specifies the user token. It is a response to the API for **obtaining a user token** (only this API does not require authentication). After the request is processed, the value of **X-Subject-Token** in the response header is the token value.	No This field is mandatory for token authentication.	The following is part of an example token: MIIPAgYJKoZIhvcNAQcCo...ggg1BBIINPXsidG9rZ

📖 **NOTE**

In addition to supporting authentication using tokens, APIs support authentication using AK/SK, which uses SDKs 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 in the request.

For more details, see "Authentication Using AK/SK" in **Authentication**.

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

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

(Optional) Request Body

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

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

In the case of the API used to **obtain a user token**, the request parameters and parameter description can be obtained from the API request. The following provides an example request with a body included. Replace *username*, *domainname*, ******** (login password), and *xxxxxxxxxxxxxxxx* (project name) with the actual values. Obtain a project name from **Regions and Endpoints**.

📖 NOTE

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

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

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

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:

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

Token-based Authentication

📖 NOTE

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

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

You can obtain a token by calling the **Obtaining User Token** API. When you call the API, set **auth.scope** in the request body to **project**.

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

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

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

AK/SK-based Authentication

> 📖 **NOTE**
>
> AK/SK-based authentication supports API requests with a body not larger than 12 MB. For API requests with a larger body, token-based authentication is recommended.

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

- AK: access key ID, which is a unique identifier used in conjunction with a secret access key to sign requests cryptographically.

- SK: secret access key used in conjunction with an AK to sign requests cryptographically. It identifies a request sender and prevents the request from being modified.

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

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

3.3 Returned Values

Status Code

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

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

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

Response Header

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

Figure 3-2 shows the response header for the API of **obtaining a user token**, in which **x-subject-token** is the desired user token. Use the token to authenticate the calling of other APIs.

Figure 3-2 Header of the response to the request for obtaining a user token

(Optional) Response Body

A response body is usually returned in a structured format corresponding to the **Content-Type** in the response header and is used to transfer content other than the response header.

The following shows part of the response body for the API to **obtain a user token**.

```
{
    "token": {
        "expires_at": "2019-02-13T06:52:13.855000Z",
```

```
        "methods": [
           "password"
        ],
        "catalog": [
           {
              "endpoints": [
                 {
                    "region_id": "xxxxxxx",
......
```

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

```
{
    "code": "SVCSTG.AOS.4001110",
    "extend": "The template already exists!",
    "message": "Bad request, name of template is already used.",
    "showdetail": false
}
```

In the preceding information, **code** indicates an error code. **extend** is the error message set according to the actual situation. **message** provides the default description of the error. When **showdetail** is set to **true**, **extend** will be displayed. When **showdetail** is set to **false**, **message** will be displayed.

4 API

4.1 Creating a Template

Function

This API is used to create a template by uploading a local template to the server.

URI

POST /v2/templates

Request

- Request parameters

Table 4-1 Request parameters

Parameter	Mandatory	Type	Description
resource	Yes	String	Template content. For details, see **UploadResource parameters**.
archive_content	No	String or Binary	The file must be in the ZIP, TAR.GZ, TGZ, YAML, or JSON format. When **content_type** is set to **yaml** or **json**, the parameter type is **String** and the value is a YAML or JSON string. If the value of **content_type** is set to **file** or left blank, the parameter type is **Binary** and the value is the binary content of the file.

Table 4-2 UploadResource parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Template name. The value must be 1 to 64 characters long and start with a letter, but cannot end with a hyphen (-). Only lowercase letters, digits, and hyphens are allowed.
description	No	String	Template description. The value supports a maximum of 255 characters. A Chinese character is regarded as multiple characters. The following characters are not allowed: <>()#%&/
origin_template_url	No	String	URL of the source template. This parameter is required when the template is uploaded by URL. The package must be in the .zip, .tar.gz, or .tgz format. The package name must comply with the regular expression ^[A-Za-z0-9][A-Za-z0-9_-]{0,254}$.
scope	No	String	Template application scope. Only the default value **domain** is supported.
vendor	No	String	Template provider. The value supports a maximum of 64 characters. A Chinese character is regarded as one character. The following characters are not allowed: <>()#%&/
version	Yes	String	Template version. The value supports a maximum of 24 characters and must meet the regular expression ^[a-zA-Z0-9_]+([-.][a-zA-Z0-9]+)*$.

Parameter	Mandatory	Type	Description
main_file_name	No	String	Name of the main file of a template. If this parameter is left blank, the default value **blueprint.yaml** is used. The value supports a maximum of 255 characters. If this parameter is not left blank, the file name extension must be **.yaml** and the file name must comply with the regular expression ^[a-zA-Z0-9_]+([-.][a-zA-Z0-9]+)*$.
project_id	No	String	Project ID contained in the template to be uploaded.
filename	No	String	Name of the compressed package file. This parameter must be specified when the Internet Explorer is used. The value must be a string that supports a maximum of 64 characters.
dsl_version	No	String	If this parameter is not left blank, the DSL version of the uploaded template must be the same as the specified version. Otherwise, the template cannot be uploaded. DSL version of the template. Only cloud_tosca_version_1_0 is supported. By default, this parameter is left blank.
content_type	No	String	Template file type. The value can be **yaml**, **json**, or **file**.

☐ NOTE

Parameters in the preceding two template upload scenarios cannot be mixed up. Otherwise, an error will be reported.

Table 4-3 Request headers

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	**multipart/form-data**.

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Valid token (a user token with the **op_cred** permission and a short validity period). This parameter is mandatory on the account side.

📖 **NOTE**

For AOS API HTTP headers, if only parameters (in JSON format) are transmitted, **Content-Type** must be set to **Application/json**. Unless otherwise specified, other fields are the same as those listed in the preceding table.

- Example request

```
curl -k -X POST -H "X-Auth-Token: $token" -H "Content-Type:multipart/form-data" -F
'resource={"name": "test-agent-11","description":"aos","version": "1.0.1"}' -F "archive_content=@test-
agent-1-1.0.tar.gz" https://aos.ae-ad-1.myhuaweicloud.com/v2/templates
```

Response

- Response when the request is successful
  - Response parameters

    Table 4-4 Response parameter when the request is successful

Parameter	Type	Description
id	String	Template ID

  - Example response

    ```
    {
        "id": "d078e49c-124c-7706-bcbb-81885661f431"
    }
    ```

- Response when the request fails
  - Response parameters

    Table 4-5 Response parameters when the request fails

Parameter	Type	Description
message	String	Detailed reason why the request fails.
code	String	Code for the reason why the request fails.
extend	String	Extended reason why the request fails. This field is reserved for future use.

📖 **NOTE**

Unless otherwise specified, subsequent AOS API errors are returned in the following form:

– Example response

```
{
    "message": "Parameters validate err, The name should not be empty.",
    "code": "02000005",
    "extend": ""
}
```

Status Code

● Normal

Table 4-6 Status code

Status Code	Description
201	The template is created successfully.

● Abnormal

Table 4-7 Status code

Status Code	Description
400	The request cannot be parsed by the server due to incorrect parameters. Unless being modified, the request should not be sent again.
409	The request cannot be completed due to a conflict with the current resource.
500	The server fails to process the request due to an unexpected condition.

4.2 Querying a Template List

Function

This API is used to query a template list based on the parameters provided.

URI

GET /v2/templates

Table 4-8 Parameter description

Parameter	Mandatory	Description
name	No	Template name. This parameter is applicable to scenarios where the template ID is queried based on the template name.
offset	No	Start offset. The default value is **0**.
limit	No	Total number of queries. This parameter can be set only when **offset** is set.
dsl_version	No	DSL version of the template.
check_used	No	Template usage condition. If this parameter is left blank, **false** is used by default.
include_stack	No	If this parameter is set to **true**, solution templates are queried. If this parameter is set to **false**, common templates are queried. If this parameter is left blank, all templates are queried.

Request

N/A

Response

- Response parameters

  For details about response parameters when the request fails, see **Table 4-5**.

Table 4-9 Response parameters

Parameter	Type	Description
[Array element]	Array of objects	Array elements are template details. For details, see **Table 4-10**.

Table 4-10 Template details

Parameter	Type	Description
id	String	Template ID.
name	String	Template name.

Parameter	Type	Description
version	String	Template version.
vendor	String	Template vendor.
description	String	Template description.
project_id	String	ID of the project to which the template belongs.
template_url	String	URL where the template package is stored.
origin_template_url	String	Original URL of the template package.
converge_url	String	URL of the template conversion package, which is used for template preview.
create_at	String	Template creation time.
update_at	String	Template update time.
scope	String	Template application scope. • **project**: available in the project (Currently, templates of this scope cannot be created.) • **domain**: available within an account • **public**: available globally, namely, available between accounts
domain	String	Name of the account to which the template belongs.
main_file_name	String	Name of the main file. If this parameter is left blank, the default value **blueprint.yaml** is used.
need_clutser	Boolean	Check whether a Cloud Container Engine (CCE) cluster is required. Currently, all AOS applications must be deployed on CCE clusters.
dsl_version	String	DSL version of the template.
is_used	Boolean	Check whether the template is in use (that is, check whether there are any stack created using this template). If the value of this parameter is **true**, the template is in use.
is_new_template	Boolean	Whether it is a new template.

Parameter	Type	Description
include_stack	Boolean	Whether the template contains stack elements (AOS.Stack).

Table 4-11 Response headers

Parameter	Type	Description
Content-Range	String	Pagination information. The values of items include the start number, end number, and total number. The value depends on the number of user templates. For example, **items 0-24/66**.

- Example response

```
[
  {
    "id": "f9f5839d-73e2-f22b-f20b-fb11bc1abb66",
    "name": "test-z",
    "version": "v1",
    "vendor": "aos-team",
    "description": "",
    "project_id": "",
    "template_url": "https://swr-api-server.manage.svc.cluster.local:20202/swr/v2/domains/aos-team/
namespaces/aos-teame002/repositories/stack_templates/packages/test-z/versions/v1/file_paths/test-
staless-container.tar.gz",
    "converge_url": "https://swr-api-server.manage.svc.cluster.local:20202/swr/v2/domains/aos-team/
namespaces/aos-teame002/repositories/stack_templates/packages/test-z/versions/v1/file_paths/test-z-
v1-converge.tar.gz",
    "origin_template_url": "",
    "create_at": "2018-12-16T01:15:39Z",
    "update_at": "2018-12-16T01:15:41Z",
    "scope": "domain",
    "domain": "aos-team",
    "main_file_name": "blueprint.yaml",
    "need_cluster": true,
    "dsl_version": "cloud_tosca_version_",
    "is_used": true,
    "is_new_template": false,
    "include_stack": false
  },
  {
    "id": "fd3f1e6d-cc93-c3c3-d015-f69c3ff3bc63",
    "name": "test-z",
    "version": "v2",
    "vendor": "aos-team",
    "description": "",
    "project_id": "",
    "template_url": "https://swr-api-server.manage.svc.cluster.local:20202/swr/v2/domains/aos-team/
namespaces/aos-teame002/repositories/stack_templates/packages/test-z/versions/v2/file_paths/
blueprint.zip",
    "converge_url": "https://swr-api-server.manage.svc.cluster.local:20202/swr/v2/domains/aos-team/
namespaces/aos-teame002/repositories/stack_templates/packages/test-z/versions/v2/file_paths/test-z-
v2-converge.tar.gz",
    "origin_template_url": "",
    "create_at": "2018-12-16T04:09:24Z",
    "update_at": "2018-12-16T04:09:26Z",
    "scope": "domain",
    "domain": "aos-team",
```

```
        "main_file_name": "blueprint.yaml",
        "need_cluster": true,
        "dsl_version": "cloud_tosca_version_",
        "is_used": true,
        "is_new_template": false,
        "include_stack": false
    }
]
```

Status Code

- Normal

Table 4-12 Status code

Status Code	Description
200	The template list query succeeds.

- Abnormal

Table 4-13 Status code

Status Code	Description
500	The server fails to process the request due to an unexpected condition.

4.3 Updating a Template

Function

This API is used to update a template.

NOTICE

The update can be performed only when the template has not been used to create a stack.

The upload modes are as follows:

- For local upload, upload the binary data of the local file directly.
- For URL upload, enter the URL of the template file.

URI

PUT /v2/templates/{template_id}

Table 4-14 Parameter description

Parameter	Mandatory	Description
template_id	Yes	ID of the template to be updated.

Request

- Request parameters

Table 4-15 Request parameters

Parameter	Mandatory	Type	Description
resource	Yes	String	Template content. For details, see **Table 4-2**.
archive_content	No	String or Binary	The file must be in the ZIP, TAR.GZ, TGZ, YAML, or JSON format. When **content_type** is set to **yaml** or **json**, the parameter type is **String** and the value is a YAML or JSON string. If the value of **content_type** is set to **file** or left blank, the parameter type is **Binary** and the value is the binary content of the file.

- Example request

```
{
    "name": "helloworld",
    "description": "helloworld blueprint new",
    "version": "1.0"
}
curl -k -X PUT -H "X-Auth-Token: $token" -H "Content-Type:multipart/form-data" -F
"archive_content=@test-agent-1-1.0.tar.gz" https://aos.ae-ad-1.myhuaweicloud.com/v2/templates/
d078e49c-124c-7706-bcbb-81885661f431
```

Response

- Response parameters

Table 4-16 Response parameter when the request is successful

Parameter	Type	Description
id	String	Template ID.

- Example response

The following shows the response when the request is successful. For details about the response when the request fails, see **Table 4-5**.

```
{
    "id": "d078e49c-124c-7706-bcbb-81885661f431"
}
```

Status Code

- Normal

Table 4-17 Status code

Status Code	Description
200	The template is updated successfully.

- Abnormal

Table 4-18 Status code

Status Code	Description
400	The request cannot be parsed by the server due to incorrect parameters. Unless being modified, the request should not be sent again.
404	The requested resource does not exist.
409	The request cannot be completed due to a conflict with the current resource.
500	The server fails to process the request due to an unexpected condition.

4.4 Deleting a Template

Function

This API is used to delete a template.

> **NOTICE**
>
> The deletion can be performed only when the template is not used to create a stack.

URI

DELETE /v2/templates/{template_id}

Table 4-19 Parameter description

Parameter	Mandatory	Description
template_id	Yes	Template ID

Request

N/A

Response

If the deletion is successful, the response body is empty with only an HTTP header.

Response messages when the deletion fails are listed in **Table 4-5**.

Status Code

- Normal

  Table 4-20 Status code

Status Code	Description
204	The template is deleted successfully.

- Abnormal

  Table 4-21 Status code

Status Code	Description
400	The request parameter is incorrect.
404	The requested resource does not exist.
409	A stack that uses the template exists.
500	The server encountered an unexpected condition which prevented it from fulfilling the request.

4.5 Downloading a Template

Function

This API is used to download a template.

URI

GET /v2/templates/{template_id}/archive

Table 4-22 Parameter description

Parameter	Mandatory	Description
template_id	Yes	Template ID

Request

N/A

Response

- Response parameters

Table 4-23 Response header

Parameter	Type	Description
Content-Type	String	Message body type (format): application/octet-stream;charset=UTF-8

Response content:

The content of the template file is as follows:

```
tosca_definitions_version: cloud_tosca_version_1_0
description: Sample template for IaaS resource
inputs:
  az:
    default: ae-ad-1a
    description: Availability Zone
    type: Cloud.ECS.AvailabilityZone.Name
    label: ECS
  flavor:
    default: s2.large.2
    description: ECS specifications
    label: ECS
  image:
    default: 1189efbf-d48b-46ad-a823-94b942e2a000
    description: ID of the image used by the ECS
    type: Cloud.ECS.Image.Id
    label: ECS
  sshKey-name:
    description: SSH key pair
    label: ECS
  mycloudserver_rootVolume_volumeType:
    description: System disk type
    label: ECS
  subnet-cidr:
    default: 192.168.1.0/24
    description: CIDR of the subnet
    label: Subnet
  subnet-gateway:
    default: 192.168.1.1
    description: Gateway of the subnet
    label: Subnet
  vm-name:
    default: my-cloudserver
    description: ECS name
```

```
        label: ECS
      vpc-cidr:
        default: 192.168.0.0/16
        description: Available subnet ranges under VPC
        label: VPC
  node_templates:
    mycloudserver:
      properties:
        availabilityZone:
          get_input: az
        flavor:
          get_input: flavor
        imageId:
          get_input: image
        name:
          get_input: vm-name
        nics:
          - subnetId:
              get_reference: mysubnet
        publicIP:
          eip:
            bandwidth:
              shareType: PER
              size: 1
            ipType: 5_bgp
        sshKeyName:
          get_input: sshKey-name
        vpcId:
          get_reference: myvpc
        rootVolume:
          volumeType:
            get_input: mycloudserver_rootVolume_volumeType
      requirements:
        - nics.subnetId:
            node: mysubnet
        - vpcId:
            node: myvpc
      type: Cloud.ECS.CloudServer
    mysubnet:
      properties:
        availabilityZone:
          get_input: az
        cidr:
          get_input: subnet-cidr
        dnsList:
          - 192.168.0.1
          - 192.168.0.0
        gateway:
          get_input: subnet-gateway
        name: ''
        vpcId:
          get_reference: myvpc
      requirements:
        - vpcId:
            node: myvpc
            relationship: Cloud.Relationships.ContainedIn
      type: Cloud.VPC.Subnet
    myvpc:
      properties:
        cidr:
          get_input: vpc-cidr
        name: ''
      type: Cloud.VPC.VPC
  outputs:
    ecs-eip:
      value:
        get_attribute:
          - mycloudserver
          - floatingIpId
```

```
ecs-id:
  value:
    get_attribute:
      - mycloudserver
      - refID
subnet-name:
  value:
    get_attribute:
      - mysubnet
      - refName
vpc-name:
  value:
    get_attribute:
      - myvpc
      - refName
```

Status Code

- Normal

Table 4-24 Status code

Status Code	Description
200	The template is successfully downloaded.

- Abnormal

Table 4-25 Status code

Status Code	Description
400	The request cannot be parsed by the server due to incorrect parameters. Unless being modified, the request should not be sent again.
404	The requested resource does not exist.
409	The request cannot be completed due to a conflict with the current resource.
500	The server encountered an unexpected condition which prevented it from fulfilling the request.

4.6 Querying a Template

Function

This API is used to query a template.

URI

GET /v2/templates/{template_id}

Table 4-26 Parameter description

Parameter	Mandatory	Description
template_id	Yes	Template ID

Request

N/A

Response

- Response parameters

  Response messages when the request fails are listed in **Table 4-5**.

  Table 4-27 Response parameters

Parameter	Type	Description
id	String	Template ID.
name	String	Template name.
version	String	Template version.
vendor	String	Template publisher.
description	String	Template description.
project_id	String	ID of the project to which the template belongs.
template_url	String	URL where the template package is stored.
origin_template_url	String	Original URL of the template package.
converge_url	String	URL of the template conversion package, which is used for template preview.
create_at	String	Template creation time.
update_at	String	Template update time.
scope	String	Template application scope. - **project**: available within a project (Currently, templates of this scope cannot be created anymore.) - **domain**: available within an account - **public**: available globally, namely, available between accounts

Parameter	Type	Description
domain	String	Name of the account to which the template belongs.
main_file_name	String	Name of the main file. If this parameter is left blank, the default value **blueprint.yaml** is used.
need_clutser	Boolean	Determine whether Cloud Container Engine (CCE) clusters are required. Currently, all AOS applications must be deployed on CCE clusters.
dsl_version	String	DSL version of the template.
is_used	Boolean	Check whether the template is in use (that is, check whether there are any stack created using this template). If the value of this parameter is **true**, the template is in use.
is_new_template	Boolean	Whether it is a new template.
include_stack	Boolean	Whether the template contains stack elements (AOS.Stack).

- Example response

```
{
    "id": "fd3f1e6d-cc93-c3c3-d015-f69c3ff3bc63",
    "name": "test-z",
    "version": "v2",
    "vendor": "aos-team",
    "description": "",
    "project_id": "",
    "template_url": "https://swr-api-server.manage.svc.cluster.local:20202/swr/v2/domains/aos-team/
namespaces/aos-teame002/repositories/stack_templates/packages/test-z/versions/v2/file_paths/
blueprint.zip",
    "converge_url": "https://swr-api-server.manage.svc.cluster.local:20202/swr/v2/domains/aos-team/
namespaces/aos-teame002/repositories/stack_templates/packages/test-z/versions/v2/file_paths/test-z-
v2-converge.tar.gz",
    "origin_template_url": "",
    "create_at": "2017-12-16T04:09:24Z",
    "update_at": "2017-12-16T04:09:26Z",
    "scope": "domain",
    "domain": "aos-team",
    "main_file_name": "blueprint.yaml",
    "need_cluster": true,
    "dsl_version": "cloud_tosca_version_"
}
```

Status Code

- Normal

Table 4-28 Status code

Status Code	Description
200	The template query succeeds.

- Abnormal

Table 4-29 Status code

Status Code	Description
400	The request parameter is incorrect.
404	The requested resource does not exist.

4.7 Querying the Input Parameters of a Template

Function

This API is used to query the input parameter of a template.

URI

GET /v2/templates/{template_id}/inputs

Table 4-30 Parameter description

Parameter	Mandatory	Description
template_id	Yes	Template ID

Request

N/A

Response

- Response parameters
  For details about response parameters when the request fails, see **Table 4-5**.

Table 4-31 Response parameters

Parameter	Type	Description
[Customized key]	Map<String,Object>	The key indicates the name of a template input parameter. The value indicates the attribute information of the template input parameter. For details, see **Table 4-32**.

Table 4-32 Data structure of the attribute information of a template input parameter

Parameter	Type	Description
required	Boolean	Whether the parameter can be left blank. The options are as follows: **true** or **false**.
default	Integer or String or Float or Boolean	Default parameter in the template. Multiple types supported, such as integer, string, and boolean. It can be selected based on the **type** field.
description	String	Description of the parameter.
type	String	Input parameter types of a template. ● **string**: character string ● **integer**: integer ● **boolean**: Boolean ● **enum**: Enumeration. If an enumeration is used, the input must contain the **valid_values** field of the **constraints** field. ● **float**: floating point number ● **secret**: key ● **password**: password ● **ip**: IP address
constraints	Object	Constraints of the parameter value. For details, see **Table 4-33**.
label	String	Label group to which the parameter belongs.

Parameter	Type	Description
advanced	Boolean	Whether the parameter is an advanced configuration. The default value must be available. The parameter is hidden by default on the GUI.
immutable	Boolean	Whether the parameter can be modified.

Table 4-33 Parameter constraint structure description

Parameter	Type	Description
in_range	Array of integers	The value must be greater than or equal to the minimum value, and less than or equal to the maximum value.
regex	String	The input value must comply with the regular expression.
valid_values	Array of integers, string, float, or boolean	The input value must be one of the given values.
equal	Integer, string, float, or Boolean	The input value must be equal to this value.
greater_or_equal	Integer, string, float, or Boolean	The input value must be greater than or equal to this value.
greater_than	Integer, string, float, or Boolean	The input value must be greater than this value.
less_or_equal	Integer, string, float, or Boolean	The input value must be less than or equal to this value.
less_than	Integer, string, float, or Boolean	The input value must be less than this value.

- Example response

```
{
    "key1": {
        "default": "192.168.12.22:30303/test_yj/frontend:latest",
        "immutable": false,
        "type": "string"
    },
    "key2": {
        "default": 2,
        "immutable": false,
        "type": "integer"
    },
    "key3": {
        "immutable": true,
        "type": "string"
```

```
    },
    "key4": {
        "type": "float",
        "constraints": {
            "greater_or_equal": 0,
            "greater_than": 0,
            "less_or_equal": 2,
            "less_than": 2
        }
    }
}
```

Status Code

- Normal

Table 4-34 Status code

Status Code	Description
200	Querying the input parameter is successful.

- Abnormal

Table 4-35 Status code

Status Code	Description
400	The request parameter is incorrect.
404	The requested resource cannot be found.
500	The server fails to process the request due to an unexpected condition.

4.8 Creating a Stack

Function

This API is used to create a stack.

📖 **NOTE**

The stack input consists of the template and input parameters.

- Template defines the framework of the stack, and determines the structure of the internal nodes of the stack, the relationships between nodes, and the value or source of the attributes of each node.
- Input parameters are one of the sources of the node attribute values in the template. They are defined in the input fields of the template and are triggered by the **get_input** function in the template.

URI

POST /v2/stacks

Request

- Request parameters

  Table 4-36 Request parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Stack name Restrictions are the same as the **name** field in **Table 4-2**.
project_id	Yes	String	ID of the project to which the stack belongs. The default value is the project to which the user token belongs. If this parameter is set, the value must be the same as the project information in the user token. The value supports a maximum of 63 character and cannot start or end with a hyphen (-). Only lowercase letters, digits, and hyphens are allowed.
description	No	String	Stack description Restrictions are the same as the **description** field in **Table 4-2**.
template_id	Yes	String	ID of the template used by the stack. The value supports a maximum of 64 characters.
inputs_json	No	Object	Input parameter of the template used by the stack. Each key-value structure in this parameter corresponds to a **key** field in the **inputs**, and the input value is **value**. If a key is not specified, the default value specified by the key is used. If no default value is specified, an error is reported. Example: "{'template_input_key': 'value'}"
force	No	Boolean	Force tailoring ID of the stack. This parameter is applicable to tailoring scenarios.

Parameter	Mandatory	Type	Description
labels	No	Object	Stack label. Each internal **key-value** indicates a label. The **key**/**value** field supports a maximum of 64 characters, and must comply with the regular expression ^[a-zA-Z0-9_]+[-.a-zA-Z0-9_]*$.
namespace	No	String	This parameter must be specified when CCE resources and applications are orchestrated. If the value is not read from CCE, **default** is used by default.
cluster_id	No	String	This parameter must be specified when CCE resources and applications are orchestrated. ID of the cluster to which the stack belongs.
action_parameters	No	Object	After a stack is created, lifecycle configurations are automatically executed and installed. Restrictions are the same as those in **Table 4-37**.

Table 4-37 Lifecycle configurations

Parameter	Mandatory	Type	Description
failure_strategy	No	String	Failure strategy. The default value is **DoNothing**. The value can be **DoNothing** or **Rollback**.
auto_create	Yes	Boolean	The default value is **true**.
timeout	No	Integer	Lifecycle timeout period (in minutes). The default value is 60 minutes. The value supports a maximum of 1440 minutes (that is, one day).

📖 **NOTE**

> When the **inputs_json** parameter is set to the attribute of a VM application and a valid URL is not entered in the **url** or **name** field, the **name**/**version**/**repository** field is used to access SWR for fuzzy search. The efficiency is low when there are a large number of files in the software repository. A feasible solution is to fill in the **url** field to avoid fuzzy search or specify the **name**, **version**, **repository** parameters in detail to reduce the search scope.
>
> - **url**: http://SWR.url/xxx/file.tgz (recommended)
> - **name**: file.tgz; **version**: v1; **repository**: myProject (good)
> - **name**: file.tgz (low-efficiency)

- Example request

```
{
    "name": "aos-lm-verydows-t1",
    "template_id": "b417b635-6ab2-1fed-b593-b129a9923ba2",
    "project_id": "10556c2112784111b8bf10512591189e",
    "cluster_id": "46c547ca-e96c-11e7-98b9-0255ac102005",
    "inputs_json": {
        "ecsng-image": "192.168.3.88:20202/default/verydows-ecsng:2.1.1",
        "mysql-image": "192.168.3.88:20202/tangtao/verydows-mysql:5.6.35",
        "redis-image": "192.168.3.88:20202/default/verydows-redis:4.0.2_passwd",
        "verydows-image": "192.168.3.88:20202/default/verydows:2.3.1",
        "verydows-install-image": "192.168.3.88:20202/default/verydows:2.3.1",
        "mysql-name": "verydows-mysql-t1"
    }
}
```

Response

- Response parameters

  A response parameter is a stack structure or response structure.

  – The stack structure is returned when the creation is successful.

  – The error response structure is returned when the creation fails.

  The stack structure is shown in **Table 3 Response parameters** and the returned incorrect response structure is shown in **Table 4-5**.

  Table 4-38 Response parameters

Parameter	Type	Description
name	String	Stack name
guid	String	Stack ID
description	String	Stack description
project_id	String	ID of the project to which the stack belongs
domain_id	String	Account to which the stack belongs
template_id	String	Template ID
template_name	String	Template name

Parameter	Type	Description
inputs_json	String	Input parameter of the template used by the stack
status	String	Stack status
create_at	String	Stack creation time
update_at	String	Stack update time
force	Boolean	Force tailoring ID of the stack. This parameter is applicable to tailoring scenarios.
labels	String	Stack label
cluster_id	String	ID of the cluster where the stack is located
cluster_name	String	Name of the cluster where the stack is located
namespace	String	Namespace to which the stack belongs
template_version	String	Version of the template used by the stack
dsl_version	String	Syntax version of the template used by the stack

- Example response when the request is successful

```
{
    "force": true,
    "guid": "d956ea24-e979-11e7-9764-0255ac103504",
    "namespace": "default",
    "project_id": "c261c106051448479eb7c66549866011",
    "cluster_name": "test-not-delete",
    "cluster_id": "46c547ca-e96c-11e7-98b9-0255ac102005",
    "domain_id": "ef09dd6cc93f4653b5557181b802ee71",
    "name": "aos-lm-verydows-t1",
    "description": "",
    "status": "",
    "template_id": "b417b635-6ab2-1fed-b593-b129a9923ba2",
    "previous_template_id": "",
    "template_name": "verydows",
    "healthy": true,
    "healthy_message": "",
    "inputs_json": "{\"ecsng-container-port\":80,\"ecsng-image\":\"192.168.3.88:20202/default/
verydows-ecsng:2.1.1\",\"imagePullPolicy\":\"IfNotPresent\",\"mysql-container-port\":3306,\"mysql-
database\":\"verydows\",\"mysql-image\":\"192.168.3.88:20202/tangtao/verydows-mysql:5.6.35\",
\"mysql-name\":\"verydows-mysql-t1\",\"mysql-password\":\"******\",\"mysql-port\":13306,\"mysql-
root-password\":\"******\",\"mysql-user\":\"verydows\",\"redis-container-port\":6379,\"redis-image\":
\"192.168.3.88:20202/default/verydows-redis:4.0.2_passwd\",\"redis-password\":\"******\",\"redis-port\":
16379,\"verydows-container-port\":80,\"verydows-http-host\":\"\",\"verydows-image\":
\"192.168.3.88:20202/default/verydows:2.3.1\",\"verydows-install-image\":\"192.168.3.88:20202/
```

```
default/verydows:2.3.1\",\"verydows-port\":8088}",
    "previous_inputs_json": "",
    "create_at": "",
    "update_at": "",
    "labels": "",
    "template_version": "2.0.1",
    "dsl_version": "cloud_tosca_version_1_0"
}
```

- Example response when the request fails

```
{
    "message": "The request body is invalid, please check. Error: The stack name is already existed.",
    "code": "02030008",
    "extend": ""
}
```

Status Code

- Normal

  Table 4-39 Status code

Status Code	Description
201	The stack is successfully created.

- Abnormal

  Table 4-40 Status code

Status Code	Description
400	The request parameter is invalid or more than one package exists.
404	The package defined in the template does not exist.
500	The server fails to process the request due to an unexpected condition.

4.9 Deleting a Stack

Function

This API is used to delete a stack.

URI

DELETE /v2/stacks/{stack_id}

Table 4-41 Parameter description

Parameter	Mandatory	Description
stack_id	Yes	Stack ID, which has a length of 1 to 64 characters
force	No	Whether to forcibly delete a stack. The value can be **true** or **false**. The forcible deletion operation can be performed only when the deletion operation has been performed before.

Request

N/A

Response

- Response parameters

  A response parameter is a character string or response structure.
  - Null is returned when the deletion starts.
  - The failure response structure is returned when the deletion fails.

  If a success response is returned, it indicates that the deletion is started. To check whether the deletion is successful, you need to perform operations in **Querying the Execution Record of a Stack**.

  If the deletion is successful, the stack cannot be queried. The cause of the deletion failure can be obtained from the returned value of this interface. The failure response structure returned is shown in **Table 4-5**.

- Example response
  ```
  {}
  ```

Status Code

- Normal

  Table 4-42 Status code

Status Code	Description
200	The stack deletion starts.

- Abnormal

  Table 4-43 Status code

Status Code	Description
404	The requested resource does not exist.

Status Code	Description
500	The server encountered an unexpected condition which prevented it from fulfilling the request.

4.10 Executing a Stack Lifecycle

Function

This API is used to perform lifecycle operations for a specific stack.

URI

PUT /v2/stacks/{stack_id}/actions

Table 4-44 Parameter description

Parameter	Mandatory	Description
stack_id	Yes	Stack ID, which has a length of 1 to 64 characters

Request

- Request parameters

  The request parameters are applicable only to scenarios where the rollback fails.

  Table 4-45 Request parameters

Parameter	Mandatory	Type	Description
failure_strategy	No	String	Failure strategy. The default value is **DoNothing**. The value can be **DoNothing** or **Rollback**.
nodes	No	Array of objects	For details about parameters of a stack element lifecycle, see **Table 4-46**.

Parameter	Mandatory	Type	Description
timeout	No	Integer	Lifecycle timeout period (in minutes). The default value is 60 minutes. This value has a length of 0 to 1440 minutes (a maximum of one day, that is, 24 x 60 minutes).
lifecycle	Yes	String	Lifecycle name The value can be **create**, **delete**, **upgrade**, **rollback**, or **retry**.
inputs	No	String	The stack configuration parameters need to be specified during the upgrade. Each key-value structure represents an input and corresponds to the inputs in the template.
template_id	No	String	Uses the new template to execute the lifecycle.

◻ NOTE

1. Only the **inputs** upgrade mode is supported during the upgrade.

2. The **failure_strategy** parameter supports rollback only in the lifecycle of **create**.

3. The **agent**, **job**, **DynamicVolume**, **FlexVolume**, **DynamicNodePool**, **NodePool**, and **AutoScaler** elements do not support upgrade or reinstallation upon installation failure.

Table 4-46 Node parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Node name
parameters	No	Object	Node lifecycle parameters, in which **ScaleParams** parameters (**Table 4-47**) are used for the scaling operations.

Table 4-47 Scaling parameters

Parameter	Mandatory	Type	Description
instances	Yes	Integer	Number of nodes. The value ranges from 0 to 100.

📖 NOTE

Scaling lifecycle is supported only for stateless applications whose number of instances is defined in **inputs**. If the **inputs** section is shared by multiple applications, the scale operation is supported only when all these applications are scaled at the same time.

- Creation request example
```
{
    "lifecycle": "create",
    "timeout": 20,
    "failure_strategy": "Rollback"
}
```

- Rollback request example
```
{
    "lifecycle": "rollback"
}
```

- Configuration parameter upgrade request example
```
{
    "lifecycle": "upgrade",
    "inputs": {
        "param1": "value1"
    }
}
```

- Retry request example
```
{
    "lifecycle": "retry",
    "timeout": 20
}
```

Response

- Response parameters

Table 4-48 Response parameters

Parameter	Type	Description
action_id	String	Execution record ID
last_action_id	String	Record ID of the previous execution

- Example response
```
{
    "action_id" : "string",
    "last_action_id" : "string",
}
```

Status Code

- Normal

Table 4-49 Status code

Status Code	Description
200	Executing the stack lifecycle is successful.

- Abnormal

Table 4-50 Status code

Status Code	Description
400	The request parameter is incorrect.
404	The stack or the element does not exist.
500	The server fails to process the request due to an unexpected condition.

4.11 Querying a Stack List

Function

This API is used to query a stack list.

URI

GET /v2/stacks?project_id={project_id}

Table 4-51 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Project to which the stack belongs. This parameter is mandatory for common user queries. The value must be 1 to 64 characters long.
labels	No	Label of the stack to be queried. The format is "key1=value1,key2=value2". The label names and values must meet the requirements of the **labels** field in **Table 4-36**.

Parameter	Mandatory	Description
dsl_version	No	DSL version of the template. The stack of the corresponding version can be queried. Only cloud_tosca_version_1_0 is supported. By default, this parameter is left blank.
template_id	No	Template ID. The stack that uses the specified template ID can be queried. This parameter cannot coexist with **template_name**.
template_name	No	Template name. The stack that uses the specified template name can be queried. This parameter cannot coexist with **template_id**.
template_version	No	Template version. The stack that uses the specified template version is queried. You need to specify **template_name** at the same time.
include_stack	No	**true**: Solution stacks are queried. **false**: Common stacks are queried. - If this parameter is left blank, all stacks are queried.
name	No	The stack that uses the specified name is queried.
status	No	Stack status. **Running**: Running. **Stopped**: All resources are stopped. **Abnormal**: The stack is abnormal. **Pending**: Initializing. **Processing**: Processing. **PartialStopped**: Some resources are stopped. **Unknow**: Unknown.

Request

N/A

Response

- Response parameters
  A response parameter is a stack structure list or response structure.

– The stack structure list is returned when the query is successful.

– The failure response structure is returned when the query fails.

For the stack structure, see **Table 4-52**. For the failure response structure, see **Table 4-5**.

Table 4-52 Response parameters

Parameter	Type	Description
[Array element]	Array of objects	Each element indicates the stack details. For details, see **Table 4-56**.

- Success response example

  Only a single stack structure is provided here. The returned URL is **JSONArray([{"name": "stack1", …}, {"name": "stack2", …}])**.

```
{
    "force": false,
    "guid": "6c2dc1bc-66cc-11e7-8a31-0255ac101c16",
    "namespace": "csc-system",
    "project_id": "3de3c020647e447fb9c9dbe1f0fc8b85",
    "cluster_name": "cluster01",
    "cluster_id": "d019ddfa-61e2-11e7-80ba-0255ac100705",
    "domain_id": "c952eaaa5bdc43668d31aff2bb74a72d",
    "name": "b-lss-f98q0",
    "description": "",
    "status": "Running",
    "template_id": "6307541e-11e9-8d57-0682-0a0ab82e8450",
    "previous_template_id": "",
    "template_name": "paasconsole",
    "healthy": true,
    "healthy_message": "",
    "inputs_json": "{\"app_path\":\"/\",\"container_port\":18081,\"custom_tag_key0\":\"netype\",
\"custom_tag_val0\":\"com.itpaas.platformservice.ncss\",\"extend_info\":\"{}\",\"image_name\":
\"10.101.248.47:20202/aos-team/cam-default/dw_broker_lss_suse11sp3:v600r001c11\",\"pod_name\":
\"b-lss-f98q0\",\"role_name\":\"manage\",\"secret_name\":\"default-ingress-ssl\",\"service_guid\":
\"beef926a-e170-0419-c220-16d90bce9951\",\"sso_info\":\"{\\\"iam_external_domain\\\":\\
\"10.101.248.93\\\",\\\"iam_external_addr\\\":\\\"10.101.248.93\\\",\\\"iam_internal_ip\\\":\\\"iam.kube-
system.svc.cluster.local\\\",\\\"iam_port\\\":\\\"31943\\\"}\",\"volume_container_path\":\"/home/
agentlib/uniagentconf\",\"volume_host_path\":\"/home/agentlib/uniagentconf\"}",
    "previous_inputs_json": "",
    "create_at": "2017-07-12T06:36:23Z",
    "update_at": "2017-07-12T06:36:39Z",
    "labels": "",
    "object": {
        "metadata": {
            "name": "b-lss-f98q0",
            "namespace": "3de3c020647e447fb9c9dbe1f0fc8b85",
            "selfLink": "/apis/paas/v1alpha1/namespaces/3de3c020647e447fb9c9dbe1f0fc8b85/stacks/b-
lss-f98q0",
            "uid": "6c2dc1bc-66cc-11e7-8a31-0255ac101c16",
            "resourceVersion": "1442425",
            "creationTimestamp": "2017-07-12T06:36:23Z",
            "labels": {
                "stackname": "b-lss-f98q0"
            },
            "annotations": {
                "cluster-id": "d019ddfa-61e2-11e7-80ba-0255ac100705",
                "namespace": "csc-system",
                "paas-iam.alpha.kubernetes.io/domain-id": "c952eaaa5bdc43668d31aff2bb74a72d",
                "stack.kubernetes.io/revision": "1"
            }
        },
        "spec": {
```

```
            "selector": {
               "matchLabels": {
                  "stackname": "b-lss-f98q0"
               }
            },
            "template": {
               "type": "Aos",
               "data": "******"
            }
         },
         "status": {
            "phase": "CreateSucceeded",
            "elementStatuses": {
               "paas-console-app": {
                  "runtimeProperties": {
                     "CfeCluster": "{\"clusterID\":\"d019ddfa-61e2-11e7-80ba-0255ac100705\",
\"namespace\":\"csc-system\"}",
                     "ConfigMap": "{\"selfLink\":\"/api/v1/namespaces/csc-system/configmaps/b-lss-
f98q0\",\"name\":\"b-lss-f98q0\",\"id\":\"7248cfe7-66cc-11e7-9a88-286ed489ceaa\"}",
                     "Deployment": "{\"selfLink\":\"/apis/extensions/v1beta1/namespaces/csc-system/
deployments/b-lss-f98q0\",\"name\":\"b-lss-f98q0\",\"id\":\"7253bff2-66cc-11e7-9a88-286ed489ceaa
\"}",
                     "Ingress": "{\"selfLink\":\"/apis/extensions/v1beta1/namespaces/csc-system/
ingresses/b-lss-f98q0\",\"annotations\":{\"ingress.beta.kubernetes.io/role\":\"manage\",
\"ingress.beta.kubernetes.io/type\":\"external\",\"ingress.kubernetes.io/external-address\":
\"10.101.204.188:30283\",\"ingress.kubernetes.io/external-ssl-address\":\"10.101.204.188:30282\"},
\"rules\":[{\"host\":\"b-lss-f98q0-csc-system.10.101.204.188.xip.io\",\"http\":{\"paths\":[{\"path\":\"/\",
\"backend\":{\"serviceName\":\"b-lss-f98q0\",\"servicePort\":32158}}]}}]}",
                     "Service": "{\"selfLink\":\"/api/v1/namespaces/csc-system/services/b-lss-f98q0\",\"type
\":\"NodePort\",\"ports\":[{\"name\":\"\",\"protocol\":\"TCP\",\"port\":18081,\"targetPort\":
18081,\"nodePort\":32158}],\"selector\":{\"app\":\"b-lss-f98q0\",\"stack-name\":\"b-lss-f98q0\"},
\"clusterIP\":\"10.247.248.87\",\"ExternalName\":\"\",\"sessionAffinity\":\"None\"}"
                  }
               }
            },
            "lastExecutionStatus": {
               "actionName": "create",
               "progress": 100,
               "objectStatus": {
                  "phase": "Succeeded",
                  "updateAt": "2017-07-12T06:36:37Z"
               },
               "subObjectStatuses": {
                  "host-dir": {
                     "phase": "Succeeded",
                     "updateAt": "2017-07-12T06:36:23Z"
                  },
                  "paas-console": {
                     "phase": "Succeeded",
                     "updateAt": "2017-07-12T06:36:23Z"
                  },
                  "paas-console-app": {
                     "phase": "Succeeded",
                     "updateAt": "2017-07-12T06:36:36Z"
                  }
               }
            }
         }
      },
      "health": {
         "message": "",
         "healthy": true
      }
   }
```

- Error response example

```
{
   "message": "Internal error.please check the error log and handle it.",
   "code": "02000051",
```

```
    "extend": ""
}
```

Status Code

- Normal

  Table 4-53 Status code

Status Code	Description
200	Querying the stack list is successful.

- Abnormal

  Table 4-54 Status code

Status Code	Description
500	The server fails to process the request due to an unexpected condition.

4.12 Querying a Stack

Function

This API is used to query a stack.

URI

GET /v2/stacks/{stack_id}

Table 4-55 Parameter description

Parameter	Mandatory	Description
stack_id	Yes	Stack ID, which has a length of 1 to 64 characters

Request

N/A

Response

- Response parameters

  A response parameter is a stack structure or response structure.

  – The stack structure is returned when the query is successful.

– The failure response structure is returned when the query fails.

The returned stack structure is shown in **Table 4-56**, and the returned failure response structure is shown in **Table 4-5**.

Table 4-56 Response parameters

Parameter	Type	Description
cluster_id	String	ID of the cluster where the stack is located.
cluster_name	String	Name of the cluster where the stack is located.
create_at	String	Stack creation time.
description	String	Stack description.
domain_id	String	Account to which the stack belongs.
dsl_version	String	Version of the template syntax used by the stack.
force	Boolean	Label for stack forcible tailoring.
guid	String	Stack ID.
health	Object	Health check results of a stack. This parameter is not returned when the stack is being created. For details, see **Table 4-57**.
inputs_json	String	Input parameters of the template used by a stack. Each key-value pair in this parameter corresponds to the key and value in the inputs. If a key is not specified, the default value is used. If no default value is specified, an error is reported.
labels	Map<String,String>	Stack label.
name	String	Stack name.
namespace	String	Namespace to which the stack belongs.
object	Object	Stack running data. This parameter is not returned when the stack is being created. For details, see **Table 4-58**.
project_id	String	ID of the project to which the stack belongs.

Parameter	Type	Description
status	String	Stack status.
template_id	String	Template ID.
template_version	String	Version of the template used by the stack.
update_at	String	Stack update time.

Table 4-57 Data structure of the health field

Parameter	Type	Description
healthy	Boolean	Health status
message	String	Description of the health status

Table 4-58 Data structure of the object field

Parameter	Type	Description
apiVersion	String	API version. The value is **paas/v1alpha1**.
kind	String	Type. The value is **Stack**.
metadata	Object	Metadata of a stack. For details, see **Table 4-59**.
spec	Object	Stack specifications. For details, see **Table 4-60**.
status	Object	Stack execution status. For details, see **Table 4-63**.

Table 4-59 Data structure of the object.metadata field

Parameter	Type	Description
annotations	Object	Special attributes of a stack. Currently, there are three fields. ● **paas-iam.alpha.kubernetes.io/ domain-id**: consistent with the top-level **domain_id** field ● **namespace**: consistent with the top-level **namespace** field ● **cluster-id**: consistent with the top-level **cluster_id** field
labels	Map<String,String>	Stack label.
name	String	Stack name, which is passed as **action_id** when lifecycle operation is executed.
namespace	String	Stack project ID, which is the same as the top-level **project_id** field.
resourceVersion	String	Resource version.
selfLink	String	API address for accessing the resource.
uid	String	Stack resource ID, which is the same as the top-level **guid** field.

Table 4-60 Data structure of the object.spec field

Parameter	Type	Description
selector	Object	This parameter contains a JSONObject field named **matchLabels**. A key-value pair is included. The key is **stackname**, and the value is the **name** field at the top level. For details, see **Table 4-61**.
template	Object	AOS template attributes. For details, see **Table 4-62**.

Table 4-61 Data structure of the object.spec.selector field

Parameter	Type	Description
matchLabels	Map<String,String>	Label information

Table 4-62 Data structure of the object.spec.template field

Parameter	Type	Description
reference	Object	Reference information
type	String	The value is **Aos**, indicating an AOS template.

Table 4-63 Data structure of the object.status field

Parameter	Type	Description
elementStatuses	Map<String,Object>	Execution status and attributes of each stack element. For details, see **Table 4-65**.
healthyStatus	object	Cache of the last stack health check status. For details, see **Table 4-66**.
lastExecutionSta-tus	object	Latest execution status. For details, see **Table 4-70**.
phase	String	Stack execution status. The status can be succeeded, executing, suspended, canceled, canceling, forcibly canceling, suspending, retrying, or running. For details, see **Table 4-64**.

Table 4-64 phase parameter description

phase Parameter Value	Description
Running	Running.
Pending	Initializing.
Terminating	Deleting.
CreateFailed	Creation failed.
DeleteSucceeded	Deletion successful.

phase Parameter Value	Description
DarklaunchSucceeded	Dark launch successful.
UpgradeFailed	Update failed.
DarklaunchFailed	Dark launch failed.
RollbackFailed	Failed to restore the stack.
DeleteFailed	Deletion failed.
UpdateFailed	Updating failed.
ForceCancelled	Forcibly canceled.
CreateSucceeded	Creation successful.
RollbackSucceeded	Stack restored successfully.
UpgradeSucceeded	Upgrade successfully.
UpdateSucceeded	Updating successfully.
ReconfigFailed	Failed to edit.
ReconfigSucceeded	Editing successful.
Creating	Creating.
Deleting	Deleting.
Upgrading	Upgrading.
Darklaunching	Performing dark launch.
Rollbacking	Restoring the stack.
Updating	Updating.
Reconfiging	Editing.
CreatePaused	Creation paused.
DeletePaused	Deletion paused.
UpgradePaused	Upgrade paused.
DarkLaunchPaused	Dark launch paused.
RollbackPaused	Stack recovery paused.
UpdatePaused	Updating paused.
ReconfigPaused	Editing paused.
CreateCancelling	Canceling creation.
DeleteCancelling	Canceling deletion.
UpgradeCancelling	Canceling upgrade.

phase Parameter Value	Description
DarklaunchCancelling	Canceling dark launch.
RollbackCancelling	Canceling stack recovery.
UpdateCancelling	Canceling update.
ReconfigCancelling	Canceling editing.
CreateCancelled	Canceling creation.
DeleteCancelled	Canceling deletion.
UpgradeCancelled	Canceling upgrade.
RollbackCancelled	Stack recovery canceled.
ReconfigCancelled	Editing canceled.
CancelFailed	Cancellation failed.

Table 4-65 Data structure of the object.status.elementStatuses field

Parameter	Type	Description
runtimeProperties	Object	Information in runtimeProperties

Table 4-66 Data structure of the object.status.healthyStatus field

Parameter	Type	Description
elements	Map<String,Object>	Health check results of each element. For details, see **Table 4-67**.
elementsReference	Object	Element reference information. For details, see **Table 4-69**.
healthyCheckTimestamp	String	Timestamp of the health check in the **RFC3339** format.
status	String	Status of an element after the health check, for example, **Running**.

Table 4-67 Data structure of the object.status.healthyStatus.elements field

Parameter	Type	Description
message	String	Health check results of elements.

Parameter	Type	Description
runtimes	Object	Each key-value indicates a feature of the health check result. For details, see **Table 4-68**.
status	String	Status of an element after the health check.

Table 4-68 Data structure of the object.status.healthyStatus.elements.runtimes field

Parameter	Type	Description
fatigue	Integer	Element status
instanceCount	Integer	Number of instances
timestamp	String	Timestamp

Table 4-69 Data structure of the object.status.healthyStatus.elementsReference field

Parameter	Type	Description
APIVersion	String	API version of an element
FieldPath	String	Field path of an element
Kind	String	Element type
Name	String	Element name
Namespace	String	Namespace of an element
ResourceVersion	String	Resource version of an element
UID	String	Element UID

Table 4-70 Data structure of the object.status.lastExecutionStatus field

Parameter	Type	Description
actionName	String	Operation name.
objectStatus	Object	Overall execution progress of the operation. For details, see **Table 4-71**.
progress	Integer	Execution progress.

Parameter	Type	Description
subObjectStatuses	Map<String,Object>	Each key-value pair indicates the execution progress of the node corresponding to the key. For details, see **Table 4-72**.

Table 4-71 Data structure of the object.status.lastExecutionStatus.objectStatus field

Parameter	Type	Description
message	String	Detailed error information.
phase	String	Execution status of an operation step.
updateAt	String	UTC timestamp of the information update in the **RFC3339** format.

Table 4-72 Data structure of the object.status.lastExecutionStatus.subObjectStatuses field

Parameter	Type	Description
createAt	String	UTC timestamp of the information creation in the **RFC3339** format.
message	String	Detailed error information.
phase	String	Execution status of an operation step.
reason	String	Brief error description in the **CamelCase** format.
subActionName	String	Name of an element suboperation.
updateAt	String	UTC timestamp of the information update in the **RFC3339** format.

- Example response

```
{
  "force": true,
  "guid": "e167e354-c4d7-11ea-972c-0255ac1003cf",
  "namespace": "default",
  "project_id": "d383836f2d134980beb6a8cc5fced7ed",
  "cluster_name": "",
  "cluster_id": "",
  "domain_id": "4e9f3643e4cc44aeb8b7473e06ab309b",
  "name": "aoss4iit-b9152f94",
  "description": "",
  "status": "Running",
  "template_id": "779531a0-8949-b015-53a5-62764f56134b",
```

```
      "template_name": "no-delete-vpc-period",
      "inputs_json": "{\"Cloud.DomainId\":\"4e9f3643e4cc44aeb8b7473e06ab309b\",\"Cloud.NoValue\":
\"\",\"Cloud.PeriodNum\":1,\"Cloud.PeriodType\":\"month\",\"Cloud.ProjectId\":
\"d383836f2d134980beb6a8cc5fced7ed\",\"Cloud.Region\":\"ae-ad-1\",\"Cloud.StackName\":\"aoss4iit-
b9152f94\",\"Cloud.UserId\":\"f24e2f88f137420a9cfe8464e5f0a1ff\"}",
      "create_at": "2020-07-08T08:18:57Z",
      "update_at": "2020-07-13T07:10:06Z",
      "labels": "{\"creator\":\"vpc1\"}",
      "dsl_version": "cloud_tosca_version_1_0",
      "object": {
        "kind": "Stack",
        "apiVersion": "paas/v1alpha1",
        "metadata": {
          "name": "aoss4iit-b9152f94",
          "namespace": "d383836f2d134980beb6a8cc5fced7ed",
          "selfLink": "/apis/paas/v1alpha1/namespaces/d3838sadweea4980beb6a8cc5fced7ed/stacks/
aoss4iit-b9152f94",
          "uid": "e167e354-c4d7-11ea-972c-0255ac1003cf",
          "resourceVersion": "274630605",
          "labels": {
            "stackname": "aoss4iit-b9152f94"
          },
          "annotations": {
            "cluster-id": "",
            "namespace": "default",
            "paas-iam.alpha.kubernetes.io/domain-id": "4e9f3643e4cc44asswb7473e06ab309b",
          }
        },
        "spec": {
          "selector": {
            "matchLabels": {
              "stackname": "aoss4iit-b9152f94"
            }
          },
          "template": {
            "type": "Aos",
            "reference": {
              "namespace": "d383836f2d134980beb6a8cc5fced7ed",
              "name": "aoss4iit-b9152f94-713114f1-1430-89fe-5112-992a1830d457",
              "uid": "aoss4iit-b9152f94-713114f1-1430-89fe-5112-992a1830d457"
            }
          }
        },
        "status": {
          "phase": "CreateSucceeded",
          "elementStatuses": {
            "vpcvp4ud": {
              "runtimeProperties": {
                "refID": "42531670-5561-4722-bdca-1007aa25710f",
                "refName": "vpcvp4ud-72b2dee4"
              }
            }
          },
          "lastExecutionStatus": {
            "actionName": "create",
            "progress": 100,
            "objectStatus": {
              "phase": "Succeeded",
              "updateAt": "2020-07-13T07:10:11.789231Z"
            },
            "subObjectStatuses": {
              "vpcvp4ud": {
                "subActionName": "create",
                "phase": "Succeeded",
                "updateAt": "2020-07-13T07:10:10.820633Z",
                "createAt": "2020-07-13T07:10:06.787787Z"
              }
            }
          },
```

```
        "healthyStatus": {
            "healthyCheckTimestamp": "2020-07-14T09:40:42+08:00",
            "status": "Running",
            "elements": {
                "vpcvp4ud": {
                    "status": "healthy",
                    "runtimes": {
                        "fatigue": "1",
                        "instanceCount": "1",
                        "timestamp": "2020-07-14T09:40:42+08:00"
                    }
                }
            },
            "elementsReference": {
                "Kind": "",
                "Namespace": "d383836f2d134980beb6a8cc5fced7ed",
                "Name": "aoss4iit-b9152f94-health-e545b9a7-565d-0284-c498-7e2d801ae968",
                "UID": "01427996-c4d8-11ea-972c-0255ac1003cf",
                "APIVersion": "",
                "ResourceVersion": "",
                "FieldPath": ""
            }
        }
    }
},
"health": {
    "message": "",
    "healthy": true
}
}
```

Status Code

- Normal

Table 4-73 Status Code

Status Code	Description
200	Querying the stack is successful.

- Abnormal

Table 4-74 Status Code

Status Code	Description
404	The requested resource cannot be found.
500	The server fails to process the request due to an unexpected condition.

4.13 Querying a Stack Element List

Function

This API is used to query a stack element list.

URI

GET /v2/stacks/{stack_id}/elements

Table 4-75 Parameter description

Parameter	Mandatory	Description
stack_id	Yes	Stack ID, which supports a maximum of 64 characters
element_type	No	Stack element type, such as **Application**, **SoftwareComponent**, **Service**, **AppGroup**, **Port**, **Cluster**, or **Volume**. Other types can also be the types defined in the template. The value supports a maximum of 64 characters.

Request

N/A

Response

- Response parameters
  A response parameter is a stack element structure list or response structure.
  - The stack element structure list is returned when the query is successful.
  - The failure response structure is returned when the query fails.
  Table 4-5 shows the structure of a failure response.

Table 4-76 Response parameters

Parameter	Type	Description
[Array element]	Array of objects	Each element in the array indicates the details about an element in the stack. For details, see **Table 4-77**.

Table 4-77 Stack element details

Parameter	Type	Description
id	String	Element name
description	String	Element description
properties	Object	Element attribute, which corresponds to the template content

Parameter	Type	Description
runtime_properties	Object	Element runtime attribute
relationships	Object	Relationships between elements. For details, see **Table 4-78**.
stack_id	String	Stack **guid** to which the element belongs
type	String	Element type
create_at	String	Element creation time
update_at	String	Element update time
type_hierarchy	String	Stack element type
action_status	Object	Status of a stack element. For details, see **Table 4-79**.

Table 4-78 Stack Nodes Relationship structure

Parameter	Type	Description
name	String	Relationship name
type	String	Relationship type
target_id	String	Name of the target node of a relationship
type_hierarchy	Array of strings	Inherited structure of the relationship
properties	Object	Attribute of a relationship. The attribute contains a key whose name is **connection_type** and the type is **string**, and indicates the connection type of the relationship.

Table 4-79 Data structure of the action_status field

Parameter	Type	Description
subActionName	String	Name of an element suboperation.

Parameter	Type	Description
phase	String	Stack execution status. The status can be succeeded, executing, suspended, canceled, canceling, forcibly canceling, suspending, retrying, or running. For details, see **Table 4-64**.
createAt	String	UTC timestamp of the information creation in the **RFC3339** format.
updateAt	String	UTC timestamp of the information update in the **RFC3339** format.

- Example response

```
[
  {
    "id": "subnet",
    "stack_id": "a067769b-c1bb-11ea-8a77-0255ac1003d8",
    "relationships": [
      {
        "name": "vpcId",
        "type": "Cloud.Relationships.ContainedIn",
        "type_hierarchy": [
          "Cloud.Relationships.Root",
          "Cloud.Relationships.DependsOn",
          "Cloud.Relationships.ContainedIn"
        ],
        "properties": {},
        "target_id": "vpc"
      }
    ],
    "type_hierarchy": [
      "Cloud.Node.Root",
      "Cloud.VPC.Subnet"
    ],
    "properties": {
      "cidr": "192.168.1.0/24",
      "dhcpEnable": true,
      "gateway": "192.168.1.1",
      "name": "dfsdf",
      "vpcId": {
        "get_attribute": [
          "vpc",
          "refID"
        ]
      }
    },
    "type": "Cloud.VPC.Subnet",
    "runtime_properties": {
      "neutron_network_id": "940bb729-260c-44fe-ab40-e98367865530",
      "neutron_subnet_id": "dc893222-ab37-4358-b162-3316c97c4958",
      "refID": "940bb729-260c-44fe-ab40-e98367865530",
      "refName": "dfsdf",
      "vpcId": "f78881f0-2944-44d4-9325-966f5f30369e"
    },
    "action_status": {
      "subActionName": "create",
      "phase": "Succeeded",
      "updateAt": "2020-07-09T08:10:26.830795Z",
      "createAt": "2020-07-09T08:10:22.270746Z"
    },
    "create_at": "2020-07-09T08:10:18Z",
```

```
      "update_at": "2020-07-09T08:10:18Z"
  },
  {
    "id": "vpc",
    "stack_id": "a067769b-c1bb-11ea-8a77-0255ac1003d8",
    "relationships": [],
    "type_hierarchy": [
      "Cloud.Node.Root",
      "Cloud.VPC.VPC"
    ],
    "properties": {
      "cidr": "192.168.0.0/16",
      "name": "sdfs"
    },
    "type": "Cloud.VPC.VPC",
    "runtime_properties": {
      "refID": "f78881f0-2944-44d4-9325-966f5f30369e",
      "refName": "sdfs"
    },
    "action_status": {
      "subActionName": "create",
      "phase": "Succeeded",
      "updateAt": "2020-07-09T08:10:22.270648Z",
      "createAt": "2020-07-09T08:10:18.278579Z"
    },
    "create_at": "2020-07-09T08:10:18Z",
    "update_at": "2020-07-09T08:10:18Z"
  }
]
```

Status Code

- Normal

  Table 4-80 Status code

Status Code	Description
200	Querying the stack element list is successful.

- Abnormal

  Table 4-81 Status code

Status Code	Description
404	The requested stack does not exist.
500	The server fails to process the request due to an unexpected condition.

4.14 Querying a Stack Element

Function

This API is used to query a stack element.

URI

GET /v2/stacks/{stack_id}/elements/{node_id}

Table 4-82 Parameter description

Parameter	Mandatory	Description
stack_id	Yes	Stack ID, which supports a maximum of 64 characters
node_id	Yes	Stack element ID, which supports a maximum of 64 characters

Request

N/A

Response

- Response parameters

  A response parameter is a stack element structure or response structure.

  – The stack element structure is returned when the query is successful.

  – The response structure is returned when the query fails.

  The stack element structure is shown in **Table 2 Response parameters**, and the returned response structure is shown in **Table 4-5**.

  Table 4-83 Response parameters

Parameter	Type	Description
instances	Array of objects	Structure of **instances**. For details, see **Table 4-84**. Only application instances are returned in this structure. This field is empty for cloud services.
id	String	Element name.
description	String	Element description.
properties	Object	Element attribute, which corresponds to the template content.
runtime_properties	Object	Element runtime attribute.
relationships	Array of objects	Relationships between elements. For details, see **Table 4-78**.

Parameter	Type	Description
stack_id	String	Stack **guid** to which the element belongs.
type	String	Element type.
create_at	String	Element creation time.
update_at	String	Element update time.
type_hierarchy	String	Stack element type.
action_status	Object	Status of a stack element. For details, see **Table 4-79**.

Table 4-84 Data structure of the instances field

Parameter	Type	Description
items	Array<Map<String,Map<String,String>>>	Information about all instances of an element

- Example response

```
{
    "create_at": "2020-07-21T05:45:19Z",
    "description": "",
    "id": "redis-app",
    "properties": {},
    "relationships": [],
    "runtime_properties": {},
    "stack_id": "0404c915-4a0c-260f-e146-eea4b95ee578",
    "type": "Cloud.AOS.Stack",
    "update_at": "2020-07-21T05:45:19Z",
    "instances": {
        "items": []
    }
}
```

Status Code

- Normal

Table 4-85 Status code

Status Code	Description
200	Querying the stack element is successful.

- Abnormal

Table 4-86 Status code

Status Code	Description
404	The requested stack or element does not exist.
500	The server fails to process the request due to an unexpected condition.

4.15 Querying a Stack Output

Function

This API is used to query a stack output.

URI

GET /v2/stacks/{stack_id}/outputs

Table 4-87 Parameter description

Parameter	Mandatory	Description
stack_id	Yes	Stack ID, which has a length of 1 to 64 characters

Request

N/A

Response

● Response parameters

Table 4-88 Response parameters

Parameter	Type	Description
outputs	Object	Stack output information. For each key-value pair, the key indicates the output's name. The value indicates a description. The structure of the description is shown in **Table 4-89**.

Table 4-89 Output parameters

Parameter	Type	Description
value	String	Output value, which is generated based on the internal definition of the template and the runtime attribute of the stack
description	String	Output description

📖 NOTE

When the value of the output is defined in the template as **get_attribute: [app, ips]**, the value returned when the output is queried is the character string array of the IP address.

- Example response

```
{
    "outputs": {
        "address_ip": {
            "value": [
                "10.175.11.126"
            ],
            "description": "ip of mysql-instance"
        },
        "address_port": {
            "value": "32138",
            "description": "port of mysql-instance"
        },
        "paasword": {
            "value": "root",
            "description": "paasword of mysql-instance"
        },
        "user_name": {
            "value": "root",
            "description": "user_name of mysql-instance"
        }
    }
}
```

Status Code

- Normal

Table 4-90 Status Code

Status Code	Description
200	Querying the stack output is successful.

- Abnormal

Table 4-91 Status code

Status Code	Description
404	The stack does not exist.
500	The server encountered an unexpected condition which prevented it from fulfilling the request.

4.16 Querying Stack Input

Function

This API is used to query stack input.

URI

GET /v2/stacks/{stack_id}/inputs

Table 4-92 Parameter description

Parameter	Mandatory	Description
stack_id	Yes	Stack ID, which has a length of 1 to 64 characters

Request

N/A

Response

● Response parameters

Table 4-93 Response parameters

Parameter	Type	Description
inputs	Object	Stack input information. For each key-value pair, the key indicates the input name. The value indicates the input value.

◻ **NOTE**

When the input parameter is defined as of the **secret/password** type in the template input, the input value returned in the query interface is encrypted.

- Example response

```
{
    "inputs": {
        "provison_mysql_passwd": "******",
        "publish_image_name": "10.175.9.211:20202/aos-team/mysql:latest"
    }
}
```

Status Code

- Normal

Table 4-94 Status code

Status Code	Description
200	Querying the stack input information is successful.

- Abnormal

Table 4-95 Status code

Status Code	Description
404	The stack does not exist.
500	The server fails to process the request due to an unexpected condition.

4.17 Querying the Execution Record of a Stack

Function

This API is used to query a stack execution record.

URI

GET /v2/stacks/{stack_id}/actions/{action_id}

Table 4-96 Parameter description

Parameter	Mandatory	Value Range	Description
stack_id	Yes	String	Stack ID. The value must be 1 to 64 characters long.
action_id	Yes	String	Name of the execution record. The value must be 1 to 64 characters long.

Request

N/A

Response

- Response parameters

  For the description about response parameters, see **Table 4-97**.

Table 4-97 Response parameters

Parameter	Type	Description
kind	String	API type. The value is fixed at **Execution** and cannot be changed.
apiVersion	String	API version. The value is fixed at **paas/v1alpha1** and cannot be changed.
metadata	Object	Execution metadata. For details, see **Table 4-98**.
spec	Object	Execution attributes. For details, see **Table 4-99**.
status	Object	Execution status. For details, see **Table 4-102**.

Table 4-98 Execution Metadata structure

Parameter	Type	Description
name	String	Execution name, which is parsed as **action_id** when the lifecycle operation is executed.
labels	String	Execution label, which is the same as that of the stack during creation. An additional **execution-hash** field is added.
namespace	String	Project ID of the stack, which is the same as the namespace field of the stack.

Parameter	Type	Description
annotations	Map<String,String>	Special attributes of a stack. Currently, there are three fields. • **paas-iam.alpha.kubernetes.io/domain-id**: consistent with the top-level **domain_id** field • **namespace**: consistent with the top-level **namespace** field • **cluster-id**: consistent with the top-level **cluster_id** field
resourceVersion	String	Resource version.
selfLink	String	API address for accessing the resource.
uid	String	Stack resource ID, which is the same as the top-level **guid** field.

Table 4-99 Structure of Execution Spec

Parameter	Type	Description
actionName	String	Lifecycle name of the execution.
changedObjectData	String	Node attribute modified by the execution, which is Base64-encoded.
disableStrategy	String	Perform the operation disabling policy.
location	String	Location of the specified Execution executor.
metadata	Object	Metadata of K8SStackSpec. For details, see **Table 4-100**.
objectData	String	Node attribute during execution operation, that is, Base64-encoded template instantiation data.
objectReference	Object	The **objectReference** parameter of **ExecutionSpec**. For details, see **Table 4-101**.
onFailureStrategy	String	Timeout retry logic of the execution.
source	String	Action source. The options are **System** and **User**.

Parameter	Type	Description
subObjectNam es	String	Name of the to-be-executed object specified by the execution. If this parameter is not blank, the lifecycle of the specified object is executed. If this parameter is blank, the lifecycle of all objects is executed.
timeoutInMinut es	Integer	Timeout time of the execution.

Table 4-100 Execution Spec Metadata structure

Parameter	Type	Description
labels	Object	Execution label, which is the same as **stack.spec.selector.matchlabels** during creation. Each internal key-value structure indicates a label with **key** as the name and **value** as the value.
namespace	String	The value of this parameter is the same as that of the stack namespace.
Other fields provided by K8S	-	-

Table 4-101 Execution Spec ObjectReference structure

Parameter	Type	Description
kind	Object	Consistent with **kind** of the stack.
namespace	String	Consistent with **namespace** of the stack.
uid	String	Consistent with **uid** of the stack.

Table 4-102 Stack Status ExecutionStatus structure

Parameter	Type	Description
actionName	String	Operation name. The value can be **create**, **delete**, **upgrade**, **rollback**, **reconfig**, or **retry**.

Parameter	Type	Description
progress	Integer	Execution progress.
objectStatus	Object	Overall execution progress of the operation. For details, see **Table 4-103**.
subObjectStatuses	Object	Execution progress of an operation sub-step. Each key-value pair indicates the execution progress of the node corresponding to the key. The value is shown in **Table 4-104**.

Table 4-103 Stack Status ExecutionStatus ActionStatus structure

Parameter	Type	Description
phase	String	Execution status of an operation step.
message	String	Detailed error information.
reason	String	Brief error description in the **CamelCase** format.
updateAt	String	UTC timestamp of the information update in the **RFC3339** format.

Table 4-104 Stack Status ExecutionStatus SubActionStatus structure

Parameter	Type	Description
phase	String	Execution status of an operation step.
message	String	Detailed error information.
reason	String	Brief error description in the **CamelCase** format.
updateAt	String	UTC timestamp of the information update in the **RFC3339** format.
subActionName	String	Name of an element suboperation.

Parameter	Type	Description
createAt	String	UTC timestamp of the information creation in the **RFC3339** format.

- Example response

```
{
    "kind": "Execution",
    "apiVersion": "paas/v1alpha1",
    "metadata": {
        "name": "trim-4074942376",
        "namespace": "aos",
        "selfLink": "/apis/paas/v1alpha1/namespaces/aos/executions/trim-4074942376",
        "uid": "ccb8fec5-cc70-11e6-8448-0242ac001004",
        "resourceVersion": "48180",
        "creationTimestamp": "2016-12-27T20:12:32Z",
        "labels": {
            "execution-hash": "4074942376",
            "stackname": "trim"
        },
        "annotations": {
            "paas-iam.alpha.kubernetes.io/domain-id": "ea816a0cc3204ee09efb6d585bee4c58",
            "stack.kubernetes.io/revision": "1"
        },
        "enable": true
    },
    "spec": {
        "metadata": {
            "namespace": "aos",
            "creationTimestamp": null,
            "labels": {
                "execution-hash": "4074942376",
                "stackname": "trim"
            },
            "enable": true
        },
        "objectReference": {
            "namespace": "aos",
            "name": "trim",
            "uid": "cb2cdea0-cc70-11e6-8448-0242ac001004"
        },
        "actionName": "create",
        "timeoutInMinutes": 60,
        "onFailureStrategy": "DoNothing",
        "disableStrategy": "Cancel",
        "source": "User",
        "objectData": "*****",
        "location": "172.16.0.0"
    },
    "status": {
        "actionName": "create",
        "progress": 100,
        "objectStatus": {
            "phase": "Succeeded",
            "updateAt": "2016-12-27T20:12:47Z"
        },
        "subObjectStatuses": {
            "hello-app": {
                "phase": "Succeeded",
                "updateAt": "2016-12-27T20:12:33Z"
            },
            "hello-component": {
                "phase": "Succeeded",
                "updateAt": "2016-12-27T20:12:47Z"
            },
```

```
        "hello-package": {
          "phase": "Succeeded",
          "updateAt": "2016-12-27T20:12:34Z"
        }
      }
    }
  }
}
```

Status Code

- Normal

Table 4-105 Status code

Status Code	Description
200	Querying the execution record is successful.

- Abnormal

Table 4-106 Status code

Status Code	Description
400	The request parameter is incorrect.
404	The stack does not exist.
500	The server fails to process the request due to an unexpected condition.

4.18 Querying a Stack Execution Record List

Function

This API is used to query the latest execution record list of a stack.

URI

GET /v2/stacks/{stack_id}/actions{?last}

Table 4-107 Path parameter description

Parameter	Mandatory	Description
stack_id	Yes	Stack ID, which supports a maximum of 64 characters

Table 4-108 Query parameters

Parameter	Mandatory	Type	Description
last	No	Boolean	Whether to return only the last execution record. If this parameter is left blank, the default value **false** is used. **true**: Only the latest execution record is returned. **false**: The latest 10 execution records are returned.

Request

N/A

Response

- Response parameters

  The response parameter is a stack execution record list or failure response structure.

  – The stack execution record list is returned when the query is successful.

  – The failure response structure is returned when the query fails.

  The execution record structure is shown in **Table 4-97**, and the failure response structure is shown in **Table 4-5**.

- Example response

```
[
  {
    "apiVersion": "paas/v1alpha1",
    "kind": "Execution",
    "metadata": {
      "annotations": {
        "aos.security.io/account": "******",
        "aosmeta.cluster.id": "9c34840a-d8c2-11e7-ba71-0255ac101306",
        "aosmeta.cluster.name": "aos",
        "aosmeta.description": "",
        "aosmeta.domain.id": "e00222f923fa4806be2df93191e83880",
        "aosmeta.force": "false",
        "aosmeta.labels": "",
        "aosmeta.namespace": "default",
        "aosmeta.project.id": "b17be51d45d74323bc39eb60315ba0df",
        "aosmeta.signature": "1",
        "aosmeta.template.id": "39368a8a-7743-c58b-e663-f3f9e80049eb",
        "aosmeta.template.name": "tosca-hello",
        "aosmeta.template.version": "nginx",
        "cluster-id": "9c34840a-d8c2-11e7-ba71-0255ac101306",
        "cluster-version": "v1.5.0-r000000",
        "namespace": "default",
        "paas-iam.alpha.kubernetes.io/domain-id": "e00222f923fa4806be2df93191e83880",
        "stack.kubernetes.io/revision": "2"
      },
      "creationTimestamp": "2018-01-30T13:41:00Z",
      "labels": {
        "execution-hash": "1686326958",
```

```
                                  "stackname": "aos-nginx"
                              },
                              "name": "aos-nginx-1686326958",
                              "namespace": "b17be51d45d74323bc39eb60315ba0df",
                              "resourceVersion": "29574440",
                              "selfLink": "/apis/paas/v1alpha1/namespaces/b17be51d45d74323bc39eb60315ba0df/
                    executions/aos-nginx-1686326958",
                              "uid": "357f52cf-05c3-11e8-9daa-0255ac1006d2"
                          },
                          "spec": {
                              "actionName": "scale",
                              "disableStrategy": "Cancel",
                              "location": "172.16.6.229",
                              "metadata": {
                                  "creationTimestamp": null,
                                  "labels": {
                                      "execution-hash": "1686326958",
                                      "stackname": "aos-nginx"
                                  },
                                  "namespace": "b17be51d45d74323bc39eb60315ba0df"
                              },
                              "objectData": "******",
                              "objectReference": {
                                  "APIVersion": "",
                                  "FieldPath": "",
                                  "Kind": "",
                                  "Name": "aos-nginx",
                                  "Namespace": "b17be51d45d74323bc39eb60315ba0df",
                                  "ResourceVersion": "",
                                  "UID": "7b22745e-05c2-11e8-b90c-0255ac100ece"
                              },
                              "onFailureStrategy": "DoNothing",
                              "source": "User",
                              "timeoutInMinutes": 5
                          },
                          "status": {
                              "actionName": "scale",
                              "objectStatus": {
                                  "phase": "Succeeded",
                                  "updateAt": "2018-01-30T13:41:13.997416Z"
                              },
                              "progress": 100,
                              "subObjectStatuses": {
                                  "nginx-app": {
                                      "phase": "Succeeded",
                                      "subActionName": "scale",
                                      "updateAt": "2018-01-30T13:41:13.289541Z"
                                  }
                              }
                          }
                      },
                      {
                          "apiVersion": "paas/v1alpha1",
                          "kind": "Execution",
                          "metadata": {
                              "annotations": {
                                  "aos.security.io/account": "******",
                                  "aosmeta.cluster.id": "9c34840a-d8c2-11e7-ba71-0255ac101306",
                                  "aosmeta.cluster.name": "aos",
                                  "aosmeta.description": "",
                                  "aosmeta.domain.id": "e00222f923fa4806be2df93191e83880",
                                  "aosmeta.force": "false",
                                  "aosmeta.labels": "",
                                  "aosmeta.namespace": "default",
                                  "aosmeta.project.id": "b17be51d45d74323bc39eb60315ba0df",
                                  "aosmeta.signature": "1",
                                  "aosmeta.template.id": "39368a8a-7743-c58b-e663-f3f9e80049eb",
                                  "aosmeta.template.name": "tosca-hello",
                                  "aosmeta.template.version": "nginx",
```

```
                    "cluster-id": "9c34840a-d8c2-11e7-ba71-0255ac101306",
                    "cluster-version": "v1.5.0-r000000",
                    "namespace": "default",
                    "paas-iam.alpha.kubernetes.io/domain-id": "e00222f923fa4806be2df93191e83880",
                    "stack.kubernetes.io/revision": "1"
                },
                "creationTimestamp": "2018-01-30T13:38:11Z",
                "labels": {
                    "execution-hash": "2513653053",
                    "stackname": "aos-nginx"
                },
                "name": "aos-nginx-2513653053",
                "namespace": "b17be51d45d74323bc39eb60315ba0df",
                "resourceVersion": "29574154",
                "selfLink": "/apis/paas/v1alpha1/namespaces/b17be51d45d74323bc39eb60315ba0df/
executions/aos-nginx-2513653053",
                "uid": "d0d17824-05c2-11e8-9daa-0255ac1006d2"
            },
            "spec": {
                "actionName": "create",
                "disableStrategy": "Cancel",
                "location": "172.16.6.229",
                "metadata": {
                    "creationTimestamp": null,
                    "labels": {
                        "execution-hash": "2513653053",
                        "stackname": "aos-nginx"
                    },
                    "namespace": "b17be51d45d74323bc39eb60315ba0df"
                },
                "objectData": "******",
                "objectReference": {
                    "APIVersion": "",
                    "FieldPath": "",
                    "Kind": "",
                    "Name": "aos-nginx",
                    "Namespace": "b17be51d45d74323bc39eb60315ba0df",
                    "ResourceVersion": "",
                    "UID": "7b22745e-05c2-11e8-b90c-0255ac100ece"
                },
                "onFailureStrategy": "DoNothing",
                "source": "User",
                "timeoutInMinutes": 60
            },
            "status": {
                "actionName": "create",
                "objectStatus": {
                    "phase": "Succeeded",
                    "updateAt": "2018-01-30T13:38:25.137462Z"
                },
                "progress": 100,
                "subObjectStatuses": {
                    "nginx-app": {
                        "phase": "Succeeded",
                        "subActionName": "create",
                        "updateAt": "2018-01-30T13:38:25.033433Z"
                    },
                    "nginx-container": {
                        "phase": "Succeeded",
                        "subActionName": "create",
                        "updateAt": "2018-01-30T13:38:12.162796Z"
                    }
                }
            }
        }
    }
]
```

Status Code

- Normal

  Table 4-109 Status code

Status Code	Description
200	Querying the stack input information is successful.

- Abnormal

  Table 4-110 Status code

Status Code	Description
400	The request parameter is incorrect.
404	The stack does not exist.
500	The server fails to process the request due to an unexpected condition.

5 Appendix

5.1 Status Code

Table 5-1 describes status codes.

Table 5-1 Status codes

Status Code	Code	Description
100	Continue	The client should continue with its request. This interim response is used to inform the client that some requests have been received and have not been rejected by the server.
101	Switching Protocols	The protocol should be switched. The protocol can only be switched to a newer protocol. For example, the current HTTP protocol is switched to a later version.
201	Created	The request for creating resources 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.
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.

Status Code	Code	Description
206	Partial Content	The server has successfully processed the partial GET request.
300	Multiple Choices	There are multiple options for the location of the requested resource. The response contains a list of resource characteristics and addresses from which a user terminal (such as a browser) can choose the most appropriate one.
301	Moved Permanently	The requested resource has been assigned with a new permanent URI. This 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	This HTTP status code is no longer used.
400	BadRequest	The request is invalid. The client should not repeat the request without modifications.
401	Unauthorized	This status code is returned after the client provides the authentication information, indicating that the authentication information is incorrect or invalid.
402	Payment Required	This status code is reserved for future use.
403	Forbidden	Access denied. The server understands the request, but refuses to fulfill it. The client should not repeat the request without modifications.
404	NotFound	The requested resource cannot be found. The client should not repeat the request without modifications.

Status Code	Code	Description
405	MethodNotAllowed	The method specified in the request is not supported by the requested resource. The client should not repeat the request without modifications.
406	Not Acceptable	The server cannot fulfill the request based on the content characteristics of the request.
407	Proxy Authentication Required	This status code is similar to 401, but indicates that the client must authenticate itself with the proxy first.
408	Request Time-out	The client does not send a request within the time that the server was prepared to wait. The client may repeat the request without modifications later.
409	Conflict	The request cannot be processed due to a conflict. This status code indicates that the resource that the client attempts to create already exits, or the request fails to be processed because of the update of the conflict request.
410	Gone	The requested resource cannot be found. 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 server refuses to process a request because the request entity is too large. The server may disable the connection to prevent the client from sending requests consecutively. If the server cannot process the request temporarily, the response will contain a Retry-After field.
414	Request-URI Too Large	The Request-URI is too long for the server to process.
415	Unsupported Media Type	The server cannot process the media format in the request.
416	Requested range not satisfiable	The requested range is invalid.

Status Code	Code	Description
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 sends excessive requests to the server within a given time (exceeding the limit on the access frequency of the client), or the server receives excessive requests within a given time (beyond its processing capability). In this case, the client should repeat requests after the time specified in the Retry-After header of the response expires.
500	InternalServerError	The server is able to receive the request but unable to understand the request.
501	Not Implemented	The server does not support the function required to fulfill the request.
502	Bad Gateway	The server acting as a gateway or proxy receives an invalid response from a remote server.
503	ServiceUnavailable	The requested service is invalid. The client should not repeat the request without modifications.
504	ServerTimeout	The request cannot be fulfilled within a given time. This status code is returned to the client only when the **Timeout** parameter is specified in the request.
505	HTTP Version not supported	The server does not support the HTTP protocol version used in the request.

5.2 Error Code

If an error occurs in API calling, no result is returned. Identify the causes of errors based on the error codes of each API. If an error occurs in API calling, HTTP status code 4*xx* or 5*xx* is returned. The response body contains the specific error code and information.

Format of an Error Response Body

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

```
{
    "code": "SVCSTG.AOS.4001110",
```

```
    "extend": "The template already exists!",
    "message": "Bad request, name of template is already used.",
    "showdetail": false
}
```

In the preceding information, **code** indicates an error code. **extend** is the error message set according to the actual situation. **message** provides the default description of the error. When **showdetail** is set to true, **extend** will be displayed. When **showdetail** is set to false, **message** will be displayed.

Error Code Description

Status Code	Error Code	Description
400	SVCSTG.AOS. 4001000	Request error. When a template is created, the template file is specified in both origin_template_url and archive_content.
400	SVCSTG.AOS. 4001001	Request error. No template source file is specified during template creation.
500	SVCSTG.AOS. 5001002	Internal server error.
500	SVCSTG.AOS. 5001003	Internal server error.
400	SVCSTG.AOS. 4001004	Request error. The entered template file is invalid.
400	SVCSTG.AOS. 4001005	The size of the template file exceeds the limit.
500	SVCSTG.AOS. 5001006	Internal server error.
400	SVCSTG.AOS. 4001007	Request error. The template file is empty.
400	SVCSTG.AOS. 4001008	Request error. The template name cannot be updated.
400	SVCSTG.AOS. 4001009	Request error. The template scope cannot be updated.
409	SVCSTG.AOS. 4091010	Request error. A stack has been created for the template.
400	SVCSTG.AOS. 4001011	The template file name already exists in the software repository. Please use another name.
500	SVCSTG.AOS. 5001012	Internal server error.
400	SVCSTG.AOS. 4001013	Request error. The main file of the template cannot be updated.

Status Code	Error Code	Description
400	SVCSTG.AOS. 4001014	Request error. The template file is empty.
400	SVCSTG.AOS. 4001015	Request error. The entered template picture is invalid.
400	SVCSTG.AOS. 4001016	The size of the template picture exceeds the limit.
500	SVCSTG.AOS. 5001006	Internal server error.
400	SVCSTG.AOS. 4001007	Request error. The template picture is empty.
400	SVCSTG.AOS. 4001100	Request error. The **archive_content** parameter in the request is incorrect.
400	SVCSTG.AOS. 4001101	Request error. The request entity is too large.
400	SVCSTG.AOS. 4001102	Request error. The resource parameter in the request is incorrect.
400	SVCSTG.AOS. 4001103	Request error. The resource parameter in the request body is invalid.
400	SVCSTG.AOS. 4001104	Request error. The template name is invalid.
400	SVCSTG.AOS. 4001105	Request error. The template description is invalid.
400	SVCSTG.AOS. 4001106	Request error. The version field is invalid.
400	SVCSTG.AOS. 4001107	Request error. The vendor field is invalid.
400	SVCSTG.AOS. 4001108	Request error. The template scope is invalid.
500	SVCSTG.AOS. 5001109	Internal server error.
400	SVCSTG.AOS. 4001110	Request error. The template name already exists.
400	SVCSTG.AOS. 4001111	Request error. The template file in the template URL must end with .zip, .tar.gz or .tgz.
500	SVCSTG.AOS. 5001112	Database operation failed. Failed to save the template to the database.

Status Code	Error Code	Description
500	SVCSTG.AOS. 5001113	Internal server error.
500	SVCSTG.AOS. 5001114	Internal server error.
400	SVCSTG.AOS. 4001115	Request error. The main file of the template is invalid.
403	SVCSTG.AOS. 4031116	Unauthorized operation. The number of templates exceeds the limit.
400	SVCSTG.AOS. 4001117	Request error. The template version is not supported.
400	SVCSTG.AOS. 4001118	Request error. The template file type is invalid.
400	SVCSTG.AOS. 4001119	Failed to verify the template. The template is invalid. Please check.
400	SVCSTG.AOS. 4001120	Request error. The project ID is invalid.
500	SVCSTG.AOS. 5001121	Internal server error.
400	SVCSTG.AOS. 4001122	Request error. The code format of the template file.
400	SVCSTG.AOS. 4001123	Request error. The **image_content** parameter in the request is incorrect.
500	SVCSTG.AOS. 5001200	Internal server error.
400	SVCSTG.AOS. 4001201	Request error. The template scope is invalid.
400	SVCSTG.AOS. 4001400	Request error. The requested stack does not exist in the database.
400	SVCSTG.AOS. 4001500	Request error. The template ID is invalid.
500	SVCSTG.AOS. 5001501	Failed to delete the template from the database.
400	SVCSTG.AOS. 4001502	Request error. The parameter that determines whether a template needs to be deleted is incorrect.
500	SVCSTG.AOS. 5001503	Internal server error.

Status Code	Error Code	Description
500	SVCSTG.AOS. 5001504	Failed to create the historical record of the template after the template is deleted successfully.
400	SVCSTG.AOS. 4001600	Request error. The **local** parameter in the request is incorrect.
500	SVCSTG.AOS. 5001700	Internal server error.
404	SVCSTG.AOS. 4041701	Request error. The requested picture does not exist.
400	SVCSTG.AOS. 4001800	Request error. The request body is too large.
400	SVCSTG.AOS. 4001801	Failed to verify the template. The template is invalid. Please check.
400	SVCSTG.AOS. 4001900	Template convergence parsing failed. The template is invalid. Please check.

Table 5-2 other_error

Status Code	Error Code	Description
400	SVCSTG.AOS. 4004600	Request body error. Please check.
400	SVCSTG.AOS. 4004601	Request error. The entered type is invalid.
500	SVCSTG.AOS. 5004602	Failed to update application status.
400	SVCSTG.AOS. 4004603	The request parameter is invalid. Please check.
500	SVCSTG.AOS. 5004604	Failed to generate CDR data.
500	SVCSTG.AOS. 5004605	Failed to obtain CDR data.
500	SVCSTG.AOS. 5004606	Internal server error.
500	SVCSTG.AOS. 5004700	Failed to query the stack quota.
500	SVCSTG.AOS. 5004701	Failed to update the stack quota.

Status Code	Error Code	Description
400	SVCSTG.AOS.4004702	Request error. The account ID is invalid.
500	SVCSTG.AOS.5004703	Internal server error.
400	SVCSTG.AOS.4004704	Request error. The quota is invalid.
500	SVCSTG.AOS.5004800	Failed to obtain statistical data.
500	SVCSTG.AOS.5004801	Internal server error.
500	SVCSTG.AOS.5004901	Internal server error.
500	SVCSTG.AOS.5004902	Internal server error.
500	SVCSTG.AOS.5004903	Internal server error.
500	SVCSTG.AOS.5004904	Internal server error.
500	SVCSTG.AOS.5004905	Internal server error.
500	SVCSTG.AOS.5004906	Internal server error.
500	SVCSTG.AOS.5004907	Internal server error.
500	SVCSTG.AOS.5004908	Internal server error.

Table 5-3 common_error

Status Code	Error Code	Description
500	SVCSTG.AOS.5000000	Authentication failed.
500	SVCSTG.AOS.5000001	Internal authentication failed.
401	SVCSTG.AOS.4010002	The authentication failed as the client is not authorized.

Status Code	Error Code	Description
500	SVCSTG.AOS. 5000003	Internal authentication failed.
500	SVCSTG.AOS. 5000004	Internal authentication failed.
500	SVCSTG.AOS. 5000005	Internal authentication failed.
401	SVCSTG.AOS. 4010005	Internal authentication failed.
403	SVCSTG.AOS. 4030006	No access permission.
400	SVCSTG.AOS. 4000007	Request error. The requested path cannot be found.
401	SVCSTG.AOS. 4010008	Internal authentication failed.
429	SVCSTG.AOS. 4290009	Server busy. Please try again later.
400	SVCSTG.AOS. 4000010	Request error. The request parameter is incorrect. Please check.
400	SVCSTG.AOS. 4000011	Request error. The request parameter is incorrect. Please check.
404	SVCSTG.AOS. 4040012	The template does not exist.
404	SVCSTG.AOS. 4040013	The stack does not exist.
403	SVCSTG.AOS. 4030014	Failed to access as your account has been frozen.
403	SVCSTG.AOS. 4030015	Failed to access as your account has been restricted. If your account balance is insufficient, recharge your account first.
400	SVCSTG.AOS. 4000100	Request error. The URL address of the template file is not provided.
400	SVCSTG.AOS. 4000101	Request error. No authentication information is provided.
400	SVCSTG.AOS. 4000102	The URL of the template file is invalid.
500	SVCSTG.AOS. 5000103	Failed to obtain the template file.

Status Code	Error Code	Description
400	SVCSTG.AOS. 4000104	Request error. The template file is too large.
500	SVCSTG.AOS. 5000105	Internal server error.
400	SVCSTG.AOS. 4000106	Request error. The namespace information is not carried.
400	SVCSTG.AOS. 4000107	Request failed. The software package name is not contained.
400	SVCSTG.AOS. 4000108	Request failed. The template file name is not carried.
500	SVCSTG.AOS. 5000109	Internal server error.
500	SVCSTG.AOS. 5000110	Internal server error.
500	SVCSTG.AOS. 5000111	Internal server error.
500	SVCSTG.AOS. 5000112	Internal server error.
500	SVCSTG.AOS. 5000113	Internal server error.
500	SVCSTG.AOS. 5000114	Internal server error.
404	SVCSTG.AOS. 4040115	The template file is empty.
409	SVCSTG.AOS. 4090116	Request error. The template file already exists in the software repository.
400	SVCSTG.AOS. 4000117	Request error. No template file can be uploaded.
500	SVCSTG.AOS. 5000118	Failed to delete the template file.
500	SVCSTG.AOS. 5000119	Failed to delete the template file.
400	SVCSTG.AOS. 4000120	The software package name is empty when the software package is queried.
404	SVCSTG.AOS. 4040121	The specified software package or image package does not exist.

Status Code	Error Code	Description
400	SVCSTG.AOS. 4000122	Request error. Multiple software packages that meet the specified criteria have been found.
500	SVCSTG.AOS. 5000123	The URL of the software package is invalid.
500	SVCSTG.AOS. 5000124	Failed to create the template or failed to obtain the namespace.
500	SVCSTG.AOS. 5000125	Failed to create the template or failed to create the namespace.
500	SVCSTG.AOS. 5000126	Failed to create the template or failed to add the domain permission for the namespace.
400	SVCSTG.AOS. 4000127	No domain information is provided.
500	SVCSTG.AOS. 5000128	Operation failed. Failed to obtain the OBS client.
500	SVCSTG.AOS. 5000129	Failed to create the template or failed to query the bucket.
500	SVCSTG.AOS. 5000130	Failed to create the template or failed to create the bucket.
500	SVCSTG.AOS. 5000131	Failed to create the template or failed to store the template file.
500	SVCSTG.AOS. 5000132	Deletion failed.
400	SVCSTG.AOS. 4000133	Request error. The template file path cannot be found.
500	SVCSTG.AOS. 5000134	Internal server error.
400	SVCSTG.AOS. 4000200	Request error. The template file does not exist.
400	SVCSTG.AOS. 4000201	Request error. The template file type is not supported.
500	SVCSTG.AOS. 5000202	Internal server error.
500	SVCSTG.AOS. 5000203	Internal server error.
500	SVCSTG.AOS. 5000204	Internal server error.

Status Code	Error Code	Description
500	SVCSTG.AOS. 5000205	No information is returned when the template is verified.
500	SVCSTG.AOS. 5000206	Failed to verify the template or failed to read the response body.
500	SVCSTG.AOS. 5000207	Failed to verify the template.
500	SVCSTG.AOS. 5000208	Internal server error.
500	SVCSTG.AOS. 5000209	Internal server error.
500	SVCSTG.AOS. 5000210	Internal server error.
500	SVCSTG.AOS. 5000211	Internal server error.
500	SVCSTG.AOS. 5000212	Internal server error.
500	SVCSTG.AOS. 5000213	Internal server error.
500	SVCSTG.AOS. 5000214	Internal server error.
500	SVCSTG.AOS. 5000215	Internal server error.
500	SVCSTG.AOS. 5000216	Internal server error.
500	SVCSTG.AOS. 5000217	Internal server error.
500	SVCSTG.AOS. 5000300	Internal server error.
500	SVCSTG.AOS. 5000301	Failed to create a stack.
500	SVCSTG.AOS. 5000302	Internal server error.
500	SVCSTG.AOS. 5000303	Internal server error.
500	SVCSTG.AOS. 5000304	Failed to obtain stack information.
500	SVCSTG.AOS. 5000305	Internal server error.

Status Code	Error Code	Description
500	SVCSTG.AOS. 5000306	Internal server error.
500	SVCSTG.AOS. 5000307	Failed to obtain the stack instance.
500	SVCSTG.AOS. 5000308	Internal server error.
500	SVCSTG.AOS. 5000309	Internal server error.
500	SVCSTG.AOS. 5000310	Failed to obtain the stack execution data.
500	SVCSTG.AOS. 5000311	Internal server error.
500	SVCSTG.AOS. 5000312	Internal server error.
500	SVCSTG.AOS. 5000313	Internal server error.
500	SVCSTG.AOS. 5000314	Internal server error.
500	SVCSTG.AOS. 5000315	Internal server error.
500	SVCSTG.AOS. 5000316	Failed to obtain stack information.
500	SVCSTG.AOS. 5000317	Internal server error.
500	SVCSTG.AOS. 5000318	Internal server error.
400	SVCSTG.AOS. 4000319	Request error. The obtained resource type is invalid.
500	SVCSTG.AOS. 5000320	Failed to obtain the resource by label.
500	SVCSTG.AOS. 5000321	Internal server error.
500	SVCSTG.AOS. 5000322	Failed to obtain the resource by name.
500	SVCSTG.AOS. 5000323	Failed to obtain the cluster.
500	SVCSTG.AOS. 5000324	Internal server error.

Status Code	Error Code	Description
500	SVCSTG.AOS. 5000325	Internal server error.
400	SVCSTG.AOS. 4000326	Request error. The cluster is unavailable.
400	SVCSTG.AOS. 4000327	Request error. The namespace is unavailable.
500	SVCSTG.AOS. 5000328	Internal server error.
409	SVCSTG.AOS. 4090329	Request error. The K8S request already exists.
500	SVCSTG.AOS. 5000400	Internal server error.
500	SVCSTG.AOS. 5000401	Internal server error.
500	SVCSTG.AOS. 5000402	Internal server error.
500	SVCSTG.AOS. 5000500	Failed to obtain the chart package.
500	SVCSTG.AOS. 5000501	Internal server error.
500	SVCSTG.AOS. 5000502	Internal server error.
400	SVCSTG.AOS. 4000503	Request error. The number of obtained chart packages is different from that of chart packages in the current domain.
400	SVCSTG.AOS. 4000504	Request error. The address of the chart package is empty.
400	SVCSTG.AOS. 4000505	Request error. The specified chart package does not exist.
400	SVCSTG.AOS. 4000506	Request error. The name of the chart package in the namespace is not unique.
400	SVCSTG.AOS. 4000507	Request error. Neither the name nor the address of the chart package is specified.
400	SVCSTG.AOS. 4000600	Request error. The template file name is empty.
400	SVCSTG.AOS. 4000601	Request error. The template file name is invalid.

Status Code	Error Code	Description
400	SVCSTG.AOS. 4000602	The template metadata is invalid.

Table 5-4 stacks_error

Status Code	Error Code	Description
400	SVCSTG.AOS.4002000	Request parameter error. Please check.
400	SVCSTG.AOS.4002001	Request error. The stack name is invalid.
400	SVCSTG.AOS.4002002	Request error. The stack description is invalid.
400	SVCSTG.AOS.4002003	Request error. The project ID is invalid.
400	SVCSTG.AOS.4002004	Request error. The template ID is invalid.
500	SVCSTG.AOS.5002005	Internal server error.
409	SVCSTG.AOS.4092006	Request error. The stack name already exists. Please use another name.
500	SVCSTG.AOS.5002007	Internal server error.
400	SVCSTG.AOS.4002008	Request error. The input parameter is invalid.
400	SVCSTG.AOS.4002009	Request error. The input parameter is invalid. Please check.
400	SVCSTG.AOS.4002010	Request error. The input parameter in the template is invalid.
400	SVCSTG.AOS.4002011	Request error. The input parameters exceed the maximum length.
500	SVCSTG.AOS.5002012	Internal server error.
400	SVCSTG.AOS.4002013	Request error. The input parameter is invalid. Please check.
500	SVCSTG.AOS.5002014	Internal server error.
400	SVCSTG.AOS.4002015	Request error. The attribute parameter of the package is invalid.
500	SVCSTG.AOS.5002016	Internal server error.
500	SVCSTG.AOS.5002017	Internal server error.
400	SVCSTG.AOS.4002018	Request error. The attribute of the externally-dependent node is invalid.

Status Code	Error Code	Description
409	SVCSTG.AOS.4092019	Request error. The stack where the external node resides is not unique.
400	SVCSTG.AOS.4002020	Request error. The stack where the external node resides is not found.
400	SVCSTG.AOS.4002021	Request error. The external node is not in the stack specified by the template.
500	SVCSTG.AOS.5002022	Internal server error.
500	SVCSTG.AOS.5002023	Database error.
400	SVCSTG.AOS.4002024	Request error. The stack label is invalid.
403	SVCSTG.AOS.4032025	No operation permission. The number of stacks reaches the limit.
500	SVCSTG.AOS.5002026	Internal server error.
400	SVCSTG.AOS.4002027	Request error. The scale parameter cannot be defined in both input and node.
400	SVCSTG.AOS.4002028	Request error. Define the scaling parameters in inputs.
400	SVCSTG.AOS.4002029	Failed to create the stack or failed to instantiate the template.
400	SVCSTG.AOS.4002030	Request error. The template is invalid. The node to be tailored in the template is not marked as deploy false.
400	SVCSTG.AOS.4002031	Failed to create the authorization relationship.
400	SVCSTG.AOS.4002100	Request error. The stack ID is invalid.
404	SVCSTG.AOS.4042101	The desired stack does not exist.
500	SVCSTG.AOS.5002102	Failed to delete a stack.
400	SVCSTG.AOS.4002103	Request error. The request parameter is invalid.
400	SVCSTG.AOS.4002200	Request error. The project ID is invalid.
500	SVCSTG.AOS.5002201	Internal server error.
400	SVCSTG.AOS.4002400	Request error. The request parameter is incorrect. Please check.
400	SVCSTG.AOS.4002401	Request error. Updating stack names is not supported.
500	SVCSTG.AOS.5002402	Internal server error.

Status Code	Error Code	Description
500	SVCSTG.AOS.5002403	Internal server error.
400	SVCSTG.AOS.4002500	Request error. The request body is invalid.
40	SVCSTG.AOS.4002501	Request error. The lifecycle parameter is invalid.
400	SVCSTG.AOS.4002502	Request error. Specify the number of instances for scaling.
500	SVCSTG.AOS.5002503	Internal server error.
500	SVCSTG.AOS.5002504	Internal server error.
400	SVCSTG.AOS.4002505	Request error. The number of instances specified for scaling must be in the range of 0-100.
400	SVCSTG.AOS.4002506	Request error. The timeout value must be in the range of 1-1440 minutes.
400	SVCSTG.AOS.4002507	Request error. The rollback policy is invalid after the stack fails to be created.
400	SVCSTG.AOS.4002508	Request error. The policy specified for the lifecycle is not supported.
500	SVCSTG.AOS.5002509	Internal server error.
500	SVCSTG.AOS.5002510	Internal server error.
500	SVCSTG.AOS.5002511	Failed to obtain the number of instances for scaling.
500	SVCSTG.AOS.5002512	Failed to parse the scaling instance.
400	SVCSTG.AOS.4002513	Request error. The node to be upgraded is not in the original stack.
500	SVCSTG.AOS.5002514	Internal server error.
400	SVCSTG.AOS.4002515	Request error. The stack cannot be rolled back because it has not been upgraded before.
500	SVCSTG.AOS.5002516	Internal server error.
500	SVCSTG.AOS.5002517	Internal server error.
500	SVCSTG.AOS.5002518	The stack containing the helm application does not have the value attribute.
500	SVCSTG.AOS.5002519	The stack containing the helm application does not have the chart attribute.

Status Code	Error Code	Description
500	SVCSTG.AOS.5002520	The stack containing the helm application does not have the version, name, or URL attribute.
500	SVCSTG.AOS.5002521	Internal server error.
400	SVCSTG.AOS.4002522	Request error. The node to be upgraded is not a software component.
400	SVCSTG.AOS.4002523	Request error. The node to be upgraded is not included in any application.
500	SVCSTG.AOS.5002524	Internal server error.
400	SVCSTG.AOS.4002525	Request error. Application upgrade, rollback, and scaling are not supported.
500	SVCSTG.AOS.5002526	The stack does not have the package attribute.
500	SVCSTG.AOS.5002527	The package attribute of the stack does not contain the package_type attribute.
500	SVCSTG.AOS.5002528	The node of the request body does not contain the package_spec attribute.
500	SVCSTG.AOS.5002529	The package_spce attribute in the node of the request body is invalid.
500	SVCSTG.AOS.5002530	The node attribute of the request body does not contain the name attribute.
500	SVCSTG.AOS.5002531	The node attribute of the request body does not contain the version attribute.
500	SVCSTG.AOS.5002532	The node attribute of the request body does not contain the URL attribute.
500	SVCSTG.AOS.5002533	The stack does not contain the process_spec attribute.
500	SVCSTG.AOS.5002534	Internal server error.
500	SVCSTG.AOS.5002535	Internal server error.
400	SVCSTG.AOS.4002536	The node attribute of the request body does not contain the URL attribute.
500	SVCSTG.AOS.5002537	The node attribute of the request body does not contain the container_spec attribute.
500	SVCSTG.AOS.5002538	The container_spec attribute in the node attribute of the request body is invalid.

Status Code	Error Code	Description
500	SVCSTG.AOS.5002539	The stack attribute does not contain the container_spce attribute.
500	SVCSTG.AOS.5002540	Internal server error.
500	SVCSTG.AOS.5002541	Internal server error.
500	SVCSTG.AOS.5002542	The package type is incorrect.
500	SVCSTG.AOS.5002543	Internal server error.
500	SVCSTG.AOS.5002544	Internal server error.
500	SVCSTG.AOS.5002545	Failed to obtain the stopped node.
500	SVCSTG.AOS.5002546	Internal server error.
400	SVCSTG.AOS.4002547	Request error. The application cannot be started or stopped.
400	SVCSTG.AOS.4002548	Request error. No stack execution record is found. Retry is not allowed.
400	SVCSTG.AOS.4002549	Request error. The scaling parameter is invalid.
500	SVCSTG.AOS.5002550	Internal server error.
400	SVCSTG.AOS.4002551	Request error. The policy does not support topology changes.
500	SVCSTG.AOS.5002552	Failed to obtain the lifecycle of the last stack execution.
500	SVCSTG.AOS.5002553	Internal server error.
500	SVCSTG.AOS.5002554	Internal server error.
400	SVCSTG.AOS.4002555	Request error. The cluster of version 1.5 does not support the upgrade of StatefulSets.
400	SVCSTG.AOS.4002556	Request error. The StatefulSet that is configured with the OnDelete upgrade policy cannot be upgraded.
400	SVCSTG.AOS.4002557	Request error. The cluster of version 1.5 does not support the upgrade of the DaemonSet application.
500	SVCSTG.AOS.5002558	Internal server error.
500	SVCSTG.AOS.5002559	Internal server error.
500	SVCSTG.AOS.5002560	Internal server error.
500	SVCSTG.AOS.5002561	Internal server error.

Status Code	Error Code	Description
400	SVCSTG.AOS.4002562	Request error. The node name is invalid.
400	SVCSTG.AOS.4002563	Request error. The node type is changed during an upgrade.
400	SVCSTG.AOS.4002564	Request error. The node relationship is changed during an upgrade.
500	SVCSTG.AOS.5002565	Internal server error.
400	SVCSTG.AOS.4002566	Request error. The node attribute which should remain unchanged is changed during an upgrade.
400	SVCSTG.AOS.4002567	Request error. The label attribute of the application is changed during an upgrade.
400	SVCSTG.AOS.4002568	Request error. The number of services bound to the node increases or decreases during an upgrade.
400	SVCSTG.AOS.4002569	Request error. A node is deleted during an upgrade, but its associated nodes are not deleted.
400	SVCSTG.AOS.4002570	Request error. The node cannot be deleted during an upgrade.
400	SVCSTG.AOS.4002571	Request error. The tosca_version of the template is changed.
400	SVCSTG.AOS.4002572	Request error. The new node type is not allowed.
400	SVCSTG.AOS.4002573	Request error. The new node has established a relationship with the original node.
400	SVCSTG.AOS.4002574	Request error. The cancellation action is not supported.
400	SVCSTG.AOS.4002575	Request error. The automatic rollback action is not supported.
400	SVCSTG.AOS.4002576	Request error. The stack parameter remains unchanged. Upgrade is not allowed.
400	SVCSTG.AOS.4002577	Request error. The stack parameter is incorrect. Upgrade and scaling are not allowed.
400	SVCSTG.AOS.4002600	Request error. The node_type of the request is not specified.

Status Code	Error Code	Description
500	SVCSTG.AOS.5002601	Failed to obtain template data.
400	SVCSTG.AOS.4002700	Request error. The node ID is invalid.
500	SVCSTG.AOS.5002701	Internal server error.
400	SVCSTG.AOS.4002702	Request error. The instance ID of the node is invalid.
400	SVCSTG.AOS.4003000	Request error. The action ID is invalid.
500	SVCSTG.AOS.5003200	Internal server error.
500	SVCSTG.AOS.5003201	Internal server error.
500	SVCSTG.AOS.5003202	Internal server error.
500	SVCSTG.AOS.5003203	Internal server error.
500	SVCSTG.AOS.5003300	Internal server error.
500	SVCSTG.AOS.5003301	Internal server error.
500	SVCSTG.AOS.5003302	Internal server error.
500	SVCSTG.AOS.5003303	Internal server error.
500	SVCSTG.AOS.5003304	Internal server error.
500	SVCSTG.AOS.5003400	Internal server error.
500	SVCSTG.AOS.5003401	Internal server error.
500	SVCSTG.AOS.5003500	Internal server error.
400	SVCSTG.AOS.4003600	The request policy type is incorrect.
400	SVCSTG.AOS.4003700	Request error. The stack parameter remains unchanged.
500	SVCSTG.AOS.5003900	Internal server error.
500	SVCSTG.AOS.5003901	The node attribute is invalid.
500	SVCSTG.AOS.5003902	The process_spec attribute of the node is invalid.
400	SVCSTG.AOS.4003903	Request error. No name specified for process_spec of certain nodes in the template.
500	SVCSTG.AOS.5003904	Internal server error.
500	SVCSTG.AOS.5003905	Internal server error.
500	SVCSTG.AOS.5003906	The external node is empty.
500	SVCSTG.AOS.5003907	Internal server error.

Status Code	Error Code	Description
400	SVCSTG.AOS.4003908	Template error. The specified external node cannot be of the ExternalNode type.
400	SVCSTG.AOS.4003909	Template error. The specified external node does not exist.
400	SVCSTG.AOS.4003910	Request error. The node itself is specified as the external node.
500	SVCSTG.AOS.5003911	Internal server error.
500	SVCSTG.AOS.5003912	Internal server error.
500	SVCSTG.AOS.5003913	Internal server error.
500	SVCSTG.AOS.5003914	Internal server error.
400	SVCSTG.AOS.4003915	Request error. The unchangeable input parameter cannot be upgraded.
500	SVCSTG.AOS.5003916	Internal server error.
500	SVCSTG.AOS.5003917	The request parameter has not been registered.
400	SVCSTG.AOS.4003918	Invalid URL set for process_spec of certain nodes in the template.
500	SVCSTG.AOS.5003919	Internal server error.
400	SVCSTG.AOS.4004000	The stack does not require a health check.
500	SVCSTG.AOS.5004001	The stack fails the health check.

5.3 Obtaining a Project ID

Scenarios

A project ID is required for some URLs when an API is called. Therefore, you need to obtain a project ID in advance. Two methods are available:

- **Obtain the Project ID by Calling an API**
- **Obtain the Project ID from the Console**

Obtain the Project ID by Calling an API

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

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

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

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

Obtain a Project ID from the Console

To obtain a project ID from the console, perform the following operations:

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

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

5.4 Obtaining an Account ID

An account ID (domain ID) is required for some URLs when an API is called. To obtain an account ID, perform the following operations:

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

A Change History

Table A-1 Change history

Release Date	Description
2020-11-05	This issue is the first official release.