

ServiceStage

API Reference

Issue	01
Date	2025-07-10



Copyright © Huawei Cloud Computing Technologies Co., Ltd. 2025. All rights reserved.

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

Trademarks and Permissions



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

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

Notice

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

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

Huawei Cloud Computing Technologies Co., Ltd.

Address: Huawei Cloud Data Center Jiaoxinggong Road
Qianzhong Avenue
Gui'an New District
Gui Zhou 550029
People's Republic of China

Website: <https://www.huaweicloud.com/intl/en-us/>

Contents

1 Before You Start.....	1
2 API Overview.....	4
3 Calling APIs.....	19
3.1 Making an API Request.....	19
3.2 Authentication.....	22
3.3 Response.....	23
4 Application Management V3 APIs.....	25
4.1 Environment.....	25
4.1.1 Creating an Environment.....	25
4.1.2 Obtaining All Environments.....	31
4.1.3 Deleting an Environment Based on the Environment ID.....	34
4.1.4 Modifying an Environment Based on the Environment ID.....	36
4.1.5 Obtaining Environment Details Based on the Environment ID.....	42
4.1.6 Modifying an Environment Resource Based on the Environment ID.....	47
4.1.7 Querying a Managed Resource Based on the Environment ID.....	50
4.1.8 Updating an Environment Based on the Environment ID.....	52
4.2 Application.....	54
4.2.1 Creating an Application.....	54
4.2.2 Obtaining All Applications.....	57
4.2.3 Modifying Application Information Based on the Application ID.....	60
4.2.4 Deleting an Application Based on the Application ID.....	64
4.2.5 Obtaining Application Details Based on the Application ID.....	65
4.2.6 Obtaining Application Configurations Based on the Application ID.....	67
4.2.7 Modifying Application Configurations Based on the Application ID.....	70
4.2.8 Deleting Application Configurations Based on the Application ID.....	73
4.3 Component.....	74
4.3.1 Creating an Application Component.....	75
4.3.2 Importing a CCE Workload to an Application to Create a Component.....	101
4.3.3 Obtaining All Components of an Application.....	105
4.3.4 Obtaining All Components.....	117
4.3.5 Modifying Component Information Based on the Component ID.....	130
4.3.6 Deleting a Component Based on the Component ID.....	155

4.3.7 Obtaining Component Information Based on the Component ID.....	157
4.3.8 Delivering a Component Task Based on the Component ID.....	183
4.3.9 Obtaining Records Based on the Component ID.....	186
4.3.10 Refreshing Component Information Based on the Component ID.....	190
4.3.11 Obtaining Component Configuration Information.....	192
4.3.12 Creating a Component Based on a Component Template.....	194
4.3.13 Parsing a Component Template Package.....	197
4.4 Deployment Job.....	202
4.4.1 Obtaining Job Details.....	202
4.5 Configuration Management.....	206
4.5.1 Creating a Configuration Group.....	206
4.5.2 Obtaining a Configuration Group.....	209
4.5.3 Creating a Configuration File.....	211
4.5.4 Importing a Configuration File.....	215
4.5.5 Obtaining Configuration File Information.....	217
4.5.6 Obtaining Configuration File Information Based on the Configuration File ID.....	221
4.5.7 Deleting a Configuration File Based on the Configuration File ID.....	224
4.5.8 Modifying a Configuration File Based on the Configuration File ID.....	225
4.5.9 Obtaining the Configuration File Record Based on the Configuration File ID.....	229
4.5.10 Obtaining the Configuration File Record Information Based on the Record ID.....	232
4.5.11 Deleting a Configuration File Record Based on the Record ID.....	235
4.5.12 Obtaining Configuration Group Details Based on the Group ID.....	237
4.5.13 Deleting a Configuration Group Based on the Group ID.....	238
4.6 Add-on Management.....	240
4.6.1 Creating an Add-on.....	240
4.6.2 Querying the Add-on List.....	242
4.6.3 Querying Add-on Metadata.....	245
4.6.4 Querying Add-on Details Based on the Add-on ID.....	247
4.6.5 Delivering an Add-on Job Based on the Add-on ID.....	250
4.6.6 Deleting an Add-on Based on the Add-on ID.....	252
4.7 Release Management.....	253
4.7.1 Old Release Management.....	253
4.7.1.1 Creating a Release Task.....	253
4.7.1.2 Editing a Release Task Based on the Release Task ID.....	277
4.7.1.3 Deleting a Release Task Based on the Release Task ID.....	301
4.7.1.4 Obtaining the Release Task List.....	303
4.7.1.5 Obtaining Information About a Release Task Based on the Release Task ID.....	305
4.7.1.6 Obtaining the Deployment Information About a Release Task Based on the Release Task ID.....	330
4.7.1.7 Executing a Release Task Based on the Release Task ID.....	333
4.7.1.8 Rolling Back a Release Task Based on the Release Task ID.....	334
4.7.1.9 Aborting a Release Task Based on the Release Task ID.....	335
4.7.2 New Release Management.....	337

4.7.2.1 Creating a Release Task.....	337
4.7.2.2 Obtaining the Release Task List.....	390
4.7.2.3 Editing a Release Task.....	393
4.7.2.4 Operating a Release Task Based on the Release Task ID.....	447
4.7.2.5 Viewing Release Task Details Based on the Release Task ID.....	450
4.7.2.6 Creating a Trigger Policy Based on the Release Task ID.....	477
4.7.2.7 Modifying a Trigger Policy Based on the Release Task ID.....	479
4.7.2.8 Querying a Trigger Policy Based on the Release Task ID.....	482
4.7.2.9 Parsing a Release Task Template Package.....	484
4.8 System Configuration.....	493
4.8.1 Technology Stack.....	493
4.8.1.1 Creating a Technology Stack.....	493
4.8.1.2 Querying a Technology Stack.....	497
4.8.1.3 Querying a Built-in Image.....	501
4.8.1.4 Querying a Technology Stack Based on the Technology Stack ID.....	502
4.8.1.5 Modifying a Technology Stack Based on the Technology Stack ID.....	505
4.8.1.6 Deleting a Technology Stack Based on the Technology Stack ID.....	509
4.8.1.7 Releasing or Canceling the Release of a Technology Stack.....	511
5 Git Repository Access APIs.....	514
5.1 Obtaining a Git Repository Authorization List.....	514
5.2 Obtaining an Authorization Redirection URL.....	516
5.3 Creating OAuth Authorization.....	518
5.4 Creating Private Token Authorization.....	522
5.5 Creating Password Authorization.....	524
5.6 Deleting Repository Authorization.....	527
5.7 Obtaining a Repository Namespace.....	528
5.8 Obtaining Repository Information Based on the Clone URL.....	530
5.9 Obtaining All Projects in a Namespace.....	532
5.10 Creating a Software Repository Project.....	534
5.11 Obtaining a Project Branch.....	536
5.12 Obtaining a Project Tag.....	538
5.13 Creating a Project Tag.....	539
5.14 Deleting a Project Tag.....	542
5.15 Obtaining Project Commits.....	543
5.16 Obtaining a Project Hook.....	545
5.17 Creating a Project Hook.....	547
5.18 Deleting a Project Hook.....	549
5.19 Obtaining a Repository File Directory.....	551
5.20 Obtaining Repository File Contents.....	553
5.21 Creating a Repository File.....	555
5.22 Modifying Repository File Contents.....	557
5.23 Deleting a Repository File.....	560

6 CSE APIs.....	562
6.1 API Calling.....	562
6.2 Dynamic Configuration.....	562
6.2.1 Importing Configurations.....	562
6.2.2 Exporting Configurations.....	567
6.3 Engine Management.....	571
6.3.1 Querying Flavors Supported by an Exclusive Microservice Engine.....	571
6.3.2 Querying the Exclusive Microservice Engine List.....	573
6.3.3 Creating an Exclusive Microservice Engine.....	580
6.3.4 Querying Details About an Exclusive Microservice Engine.....	584
6.3.5 Deleting an Exclusive Microservice Engine.....	592
6.3.6 Querying Details About an Exclusive Microservice Engine Job.....	594
7 ServiceComb APIs.....	600
7.1 API Calling.....	600
7.2 Authentication.....	600
7.2.1 Obtaining the User Token of an Exclusive Microservice Engine.....	600
7.3 Microservice.....	602
7.3.1 Querying Information About a Microservice.....	603
7.3.2 Deleting Definition Information About a Microservice.....	607
7.3.3 Querying Information About All Microservices.....	610
7.3.4 Creating Static Information for a Microservice.....	615
7.3.5 Deleting Static Information About Microservices in Batches.....	622
7.3.6 Modifying Extended Attributes of a Microservice.....	626
7.3.7 Querying the Unique Service or Schema ID of a Microservice.....	629
7.4 Schema.....	633
7.4.1 Querying a Microservice Schema.....	633
7.4.2 Modifying a Microservice Schema.....	636
7.4.3 Querying All Schema Information About a Microservice.....	639
7.5 Microservice Instance.....	642
7.5.1 Registering a Microservice Instance.....	642
7.5.2 Querying a Microservice Instance Based on service_id.....	647
7.5.3 Deregistering a Microservice Instance.....	651
7.5.4 Querying Details About a Microservice Instance.....	654
7.5.5 Modifying the Extended Information About a Microservice Instance.....	659
7.5.6 Modifying Status of a Microservice Instance.....	662
7.5.7 Sending Heartbeat Information.....	665
7.5.8 Querying a Microservice Instance by Filter Criteria.....	667
7.5.9 Querying Microservice Instances in Batches.....	672
7.6 Dependency.....	679
7.6.1 Querying All Providers of a Microservice.....	680
7.7 Configuration Management.....	684
7.7.1 Creating a Configuration.....	684

7.7.2 Modifying a Configuration.....	688
7.7.3 Querying a Configuration.....	690
7.7.4 Querying the Configuration List.....	693
7.7.5 Deleting a Configuration.....	697
7.7.6 Deleting All Configurations.....	700
8 Examples.....	703
8.1 Creating and Managing an Environment.....	703
8.2 Creating and Managing an Application.....	705
8.3 Creating and Managing a Component.....	707
8.4 Microservice Registry and Discovery.....	713
9 Data Structure.....	716
9.1 Parameter Description.....	716
9.1.1 HealthCheck.....	716
9.1.2 MicroServiceInstance.....	717
9.1.3 HeartbeatSetElement.....	718
9.1.4 InstanceHbRst.....	718
9.1.5 DelServicesResponse.....	718
9.1.6 MetricData.....	718
9.1.7 FunctionMetricInfo.....	719
9.1.8 InstanceMetricInfo.....	720
9.2 Common Request Parameters.....	720
9.2.1 MicroService.....	720
9.2.2 Properties.....	722
9.2.3 DependencyMicroService.....	722
9.2.4 Rule.....	723
9.2.5 AddOrUpdateRule.....	724
9.2.6 DataCenterInfo.....	724
9.2.7 Schema.....	725
9.3 Common Response Parameters.....	725
9.3.1 WatchMicroServiceKey.....	725
9.3.2 SuccessdResponse.....	726
9.3.3 ServiceInfo.....	726
9.3.4 AggregateMetricInfo.....	727
9.3.5 ServiceDetail.....	728
9.3.6 TenantProject.....	729
9.3.7 Error.....	729
10 Permissions and Supported Actions.....	730
10.1 Introduction.....	730
11 Appendix.....	734
11.1 Status Codes.....	734
11.2 ServiceStage Error Codes.....	735

11.3 CSE Error Codes.....	739
11.4 ServiceComb Error Codes.....	745
11.5 Obtaining a Project ID.....	750
11.6 Obtaining an Account ID.....	751
12 Historical APIs.....	752
12.1 Application Management V2 APIs.....	752
12.1.1 Meta.....	752
12.1.1.1 Obtaining All Supported Runtime Systems of Application Components.....	752
12.1.1.2 Obtaining All Supported Flavors of Application Resources.....	754
12.1.2 Environment.....	756
12.1.2.1 Creating an Environment.....	756
12.1.2.2 Obtaining All Environments.....	760
12.1.2.3 Modifying Environment Information.....	764
12.1.2.4 Deleting an Environment Based on the Environment ID.....	767
12.1.2.5 Obtaining Environment Details Based on the Environment ID.....	769
12.1.2.6 Modifying Environment Resources.....	771
12.1.3 Application.....	776
12.1.3.1 Creating an Application.....	776
12.1.3.2 Obtaining All Applications.....	778
12.1.3.3 Modifying Application Information.....	781
12.1.3.4 Deleting an Application Based on the Application ID.....	783
12.1.3.5 Obtaining Application Details Based on the Application ID.....	785
12.1.3.6 Modifying Application Configurations.....	787
12.1.3.7 Deleting Application Configurations.....	790
12.1.3.8 Obtaining Application Configurations.....	792
12.1.4 Component.....	794
12.1.4.1 Creating an Application Component.....	795
12.1.4.2 Obtaining All Components of an Application.....	802
12.1.4.3 Modifying Component Information Based on the Component ID.....	808
12.1.4.4 Deleting a Component Based on the Component ID.....	815
12.1.4.5 Obtaining Component Information Based on the Component ID.....	817
12.1.5 Instance.....	821
12.1.5.1 Creating an Application Component Instance.....	821
12.1.5.2 Obtaining All Component Instances.....	834
12.1.5.3 Querying the Operations Performed on a Component Instance.....	838
12.1.5.4 Modifying a Component Instance.....	841
12.1.5.5 Deleting a Component Instance.....	846
12.1.5.6 Querying Instance Details Based on the Instance ID.....	848
12.1.5.7 Obtaining Component Instance Snapshots.....	853
12.1.6 Deployment Jobs.....	855
12.1.6.1 Obtaining Job Details.....	855
12.2 Build APIs.....	860

12.2.1 Obtaining the Build Job List Based on the Component Name..... 860

12.2.2 Obtaining the Build Job Instance List Based on the Build Job ID..... 862

12.2.3 Obtaining the Build Phase List Based on the Build Job Instance ID..... 864

12.2.4 Obtaining Build Logs Based on the Build Job Instance ID..... 867

12.3 Querying Configurations..... 869

12.4 Creating a Dependency Between Services..... 872

12.5 Reporting Service Metrics..... 874

1 Before You Start

Welcome to ServiceStage. ServiceStage is an application management and O&M platform that lets you deploy, roll out, monitor, and maintain applications all in one place. It supports technology stacks such as Java, PHP, Python, Node.js, Docker, and Tomcat, and supports microservice applications such as Apache ServiceComb Java Chassis (Java chassis) and Spring Cloud, making it easier to migrate enterprise applications to the cloud.

This document describes how to use application programming interfaces (APIs) to perform operations on ServiceStage, such as managing environments, applications, and components. For details about all supported operations, see [API Overview](#).

Before calling ServiceStage APIs, ensure that you are familiar with ServiceStage concepts. For details, see [Service Overview](#).

In addition, ServiceStage provides enhanced APIs of Cloud Service Engine (CSE). For details, see [CSE APIs](#). To use native CSE APIs, see [ServiceComb APIs](#).

Endpoints

An endpoint is the **request address** for calling an API. Endpoints vary depending on services and regions. For the endpoints of ServiceStage, see [Regions and Endpoints](#).

Concepts

- Account
An account is created upon successful registration with the cloud platform. The account has full access permissions for all of its cloud services and resources. It can be used to reset user passwords and grant user permissions. The account is a payment entity and should not be used directly to perform routine management. For security purposes, create Identity and Access Management (IAM) users and grant them permissions for routine management.
- User
A user is created in Identity and Access Management (IAM) to use cloud services. Each user has its own identity credentials (password and access keys).

An IAM user can view the account ID and user ID on the [My Credentials](#) page of the console. The account name, username, and password will be required for API authentication.

- Region

Regions are divided from the dimensions of geographical location and network latency. Public services, such as Elastic Cloud Server (ECS), Elastic Volume Service (EVS), Virtual Private Cloud (VPC), Elastic IP (EIP), and Image Management Service (IMS), are shared within the same region. Regions are classified into universal and dedicated regions. A universal region provides universal cloud services for common tenants. A dedicated region provides specific services for specific tenants.

For details, see [Region and AZ](#).

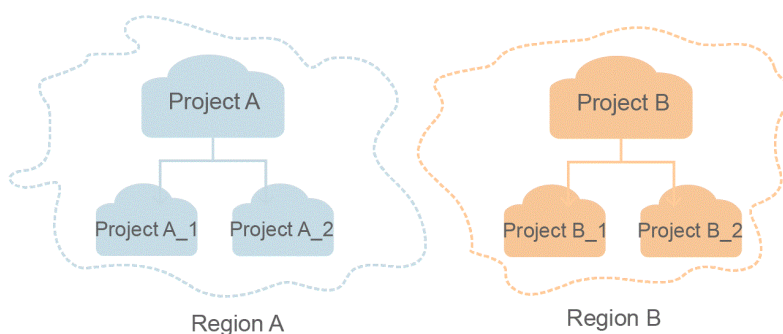
- AZ

An availability zone (AZ) comprises one or more physical data centers equipped with independent ventilation, fire, water, and electricity facilities. Compute, 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 support cross-AZ high-availability systems.

- Project

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

Figure 1-1 Project isolating model



You can view the project ID on the [My Credentials](#) page.

- Enterprise project

Enterprise projects group and manage resources across regions. Resources in enterprise projects are logically isolated. An enterprise project can contain resources of multiple regions, and resources can be added to or removed from enterprise projects.

For details about how to obtain the enterprise project ID and features, see [Enterprise Management User Guide](#).

2 API Overview

Table 2-1 Overview

Type	Subtype	Description
Application management V3 APIs	Environment APIs	APIs related to environments. An environment is a collection of basic compute (such as CCE and ECS), network (such as ELB and EIP), and middleware (such as DCS and RDS) resources, used for component deployment and running. ServiceStage combines multiple basic resources into an environment, including development, test, pre-production, and production environments. Managing resources and deploying components by environment simplifies O&M management.
	Application APIs	APIs related to applications. An application is a service system with functions and consists of one or more components.

Type	Subtype	Description
	Component APIs	APIs related to components. A component is a service feature implementation of an application. It is carried by code or software packages and can be independently deployed and run in an environment.
	Configuration Management APIs	APIs related to configurations. In ServiceStage, a configuration is a file. By creating a unified configuration file, you can fill the system variables (such as the IP address, port number, database address, and application name associated with the environment) of the environment and application in the configuration to generate a configuration file. When a component is associated with a configuration file for deployment, the system variables are automatically replaced with the actual value. This implements multi-environment use with one-time configuration through file mounting.

Type	Subtype	Description
	Technology Stack APIs	APIs related to the technology stack. A technology stack includes the operating system, framework, and runtime system on which component running depends. It consists of attributes such as the technology stack name, type, status, and version. The version number complies with the semantic versioning specifications . ServiceStage provides and manages the stack lifecycle. You only need to focus on service development to improve application hosting experience.
	Deployment Job API	APIs related to deployment jobs.
	Add-on Management APIs	APIs related to add-on management.
	Old Release Management APIs	APIs related to old release management.
	New Release Management APIs	APIs related to new release management.
Git repository access APIs	Git Repository Access APIs	APIs related to Git repository authorization, namespaces, projects, branches, tags, commits, hooks, repository file directories, and contents.
CSE APIs	Dynamic Configuration	APIs related to configuration import to and export from the configuration management center.
	Engine Management	APIs related to exclusive microservice engine creation, query, and deletion.

Type	Subtype	Description
ServiceComb APIs	Authentication	APIs related to microservice identity authentication.
	Microservice	APIs related to microservice management.
	Schema	APIs related to microservice contract query and upload.
	Microservice Instance	APIs related to microservice instance registry and heartbeat reporting.
	Dependency	APIs related to microservice dependency query.
	Configuration Management	APIs related to configuration management of the configuration center.

Environment APIs

Table 2-2 Environment APIs

API	Description
Creating an Environment	Create an environment.
Obtaining All Environments	Obtain all environments.
Deleting an Environment Based on the Environment ID	Delete an environment based on the environment ID.
Modifying an Environment Based on the Environment ID	Modify an environment based on the environment ID.
Obtaining Environment Details Based on the Environment ID	Obtain environment details based on the environment ID.
Modifying an Environment Resource Based on the Environment ID	Modify an environment resource based on environment ID.
Querying a Managed Resource Based on the Environment ID	Query a managed resource based on the environment ID.

API	Description
Updating an Environment Based on the Environment ID	Update an environment based on the environment ID.

Application APIs

Table 2-3 Application APIs

API	Description
Creating an Application	Create an application.
Obtaining All Applications	Obtain all applications.
Modifying Application Information Based on the Application ID	Modify application information based on the application ID.
Deleting an Application Based on the Application ID	Delete an application based on the application ID.
Obtaining Application Details Based on the Application ID	Obtain application details based on the application ID.
Obtaining Application Configurations Based on the Application ID	Obtain application configurations based on the application ID.
Modifying Application Configurations Based on the Application ID	Modify application configurations based on the application ID.
Deleting Application Configurations Based on the Application ID	Delete application configurations based on the application ID.

Component APIs

Table 2-4 Component APIs

API	Description
Creating an Application Component	Create an application component.
Importing a CCE Workload to an Application to Create a Component	Import a CCE workload to an application to create a component.
Obtaining All Components of an Application	Obtain all components of an application.
Obtaining All Components	Obtain all components.

API	Description
Modifying Component Information Based on the Component ID	Modify component information based on the component ID.
Deleting a Component Based on the Component ID	Delete a component based on the component ID.
Obtaining Component Information Based on the Component ID	Obtain component information based on the component ID.
Delivering a Component Task Based on the Component ID	Deliver a component action based on the component ID.
Obtaining Records Based on the Component ID	Obtain records based on the component ID.
Refreshing Component Information Based on the Component ID	Refresh the component status and access mode based on the component ID.
Obtaining Component Configuration Information	Obtain information about the application, environment, and enterprise project to which all components belong.
Creating a Component Based on a Component Template	Create a component based on a component template.
Parsing a Component Template Package	Parse a component template package.

Technology Stack APIs

Table 2-5 Technology stack APIs

API	Description
Querying a Technology Stack	Query technology stack information.
Querying a Built-in Image	Obtain the built-in basic images used for building ServiceStage technology stacks.
Creating a Technology Stack	Create a custom technology stack.
Querying a Technology Stack Based on the Technology Stack ID	Query a technology stack based on the technology stack ID.
Modifying a Technology Stack Based on the Technology Stack ID	Modify a technology stack based on the technology stack ID.
Deleting a Technology Stack Based on the Technology Stack ID	Delete a technology stack based on the technology stack ID.

API	Description
Releasing or Canceling the Release of a Technology Stack	Release or cancel the release of a technology stack.

Configuration Management APIs

Table 2-6 Configuration management APIs

API	Description
Creating a Configuration Group	Create a configuration group to manage configuration files.
Obtaining a Configuration Group	Obtain all created configuration groups.
Creating a Configuration File	Create a configuration file.
Importing a Configuration File	Import a configuration file in .zip format.
Obtaining Configuration File Information	Obtain the configuration file information.
Obtaining Configuration File Information Based on the Configuration File ID	Obtain the details about a configuration file based on the configuration file ID.
Deleting a Configuration File Based on the Configuration File ID	Delete a configuration file based on the configuration file ID.
Modifying a Configuration File Based on the Configuration File ID	Modify a configuration file based on the configuration file ID.
Obtaining the Configuration File Record Based on the Configuration File ID	Obtain the configuration file record based on the configuration file ID.
Obtaining the Configuration File Record Information Based on the Record ID	Obtain the configuration file record information based on the record ID.
Deleting a Configuration File Record Based on the Record ID	Delete a configuration file record based on the record ID.
Obtaining Configuration Group Details Based on the Group ID	Obtain the configuration group details based on the group ID.
Deleting a Configuration Group Based on the Group ID	Delete a configuration group based on the group ID.

Deployment Job API

Table 2-7 Deployment job API

API	Description
Obtaining Job Details	Obtain job details.

Add-on Management APIs

Table 2-8 Add-on management APIs

API	Description
Creating an Add-on	Create an add-on.
Querying the Add-on List	Query the add-on list in an environment.
Querying Add-on Metadata	Query the add-on metadata in an environment.
Querying Add-on Details Based on the Add-on ID	Query details about an add-on in an environment.
Delivering an Add-on Job Based on the Add-on ID	Deliver an add-on job based on the add-on ID.
Deleting an Add-on Based on the Add-on ID	Delete an add-on based on the add-on ID.

Old Release Management APIs

Table 2-9 Old release management APIs

API	Description
Creating a Release Task	Create a release task.
Editing a Release Task Based on the Release Task ID	Edit a release task based on the release task ID.
Deleting a Release Task Based on the Release Task ID	Delete a release task based on the release task ID.
Obtaining the Release Task List	Obtain the release task list.
Obtaining Information About a Release Task Based on the Release Task ID	Obtain information about a release task based on the release task ID.

API	Description
Obtaining the Deployment Information About a Release Task Based on the Release Task ID	Obtain the deployment information about a release task based on the release task ID.
Executing a Release Task Based on the Release Task ID	Execute a release task based on the release task ID.
Rolling Back a Release Task Based on the Release Task ID	Roll back a release task based on the release task ID.
Aborting a Release Task Based on the Release Task ID	Abort a release task based on the release task ID.

New Release Management APIs

Table 2-10 New release management APIs

API	Description
Creating a Release Task	Create a release task.
Obtaining the Release Task List	Obtain the release task list.
Editing a Release Task	Edit a release task.
Operating a Release Task Based on the Release Task ID	Operate a release task based on the release task ID.
Viewing Release Task Details Based on the Release Task ID	View release task details based on the release task ID.
Creating a Trigger Policy Based on the Release Task ID	Create a trigger policy based on the release task ID.
Modifying a Trigger Policy Based on the Release Task ID	Modify a release task trigger policy based on the release task ID.
Querying a Trigger Policy Based on the Release Task ID	Query a release task trigger policy based on the release task ID.
Parsing a Release Task Template Package	Parse a release task template package.

Git Repository Access APIs

Table 2-11 Git repository access APIs

API	Description
Obtaining a Git Repository Authorization List	Obtain the Git repository authorization list.
Obtaining an Authorization Redirection URL	Obtain an authorization redirection URL.
Creating OAuth Authorization	Create OAuth authorization.
Creating Private Token Authorization	Create private token authorization.
Creating Password Authorization	Create password authorization for a Git repository.
Deleting Repository Authorization	Delete repository authorization based on the name.
Obtaining a Repository Namespace	Obtain namespaces of a repository.
Obtaining Repository Information Based on the Clone URL	Obtain repository information based on the clone URL.
Obtaining All Projects in a Namespace	Obtain all projects in a namespace.
Creating a Software Repository Project	Create a software repository project.
Obtaining a Project Branch	Obtain a project branch.
Obtaining a Project Tag	Obtain a project tag.
Creating a Project Tag	Create a project tag.
Deleting a Project Tag	Delete a project tag.
Obtaining Project Commits	Obtain the latest ten project commits.
Obtaining a Project Hook	Obtain a project hook.

API	Description
Creating a Project Hook	Create a project hook.
Deleting a Project Hook	Delete a project hook.
Obtaining a Repository File Directory	Obtain a repository file directory.
Obtaining Repository File Contents	Obtain repository file contents.
Creating a Repository File	Create a repository file.
Modifying Repository File Contents	Modify repository file contents.
Deleting a Repository File	Delete repository file contents.

Dynamic Configuration

Table 2-12 Dynamic configuration APIs

API	Description
Importing Configurations	Import configurations to the configuration management center.
Exporting Configurations	Export configurations from the configuration management center.

Engine Management

Table 2-13 Engine management APIs

API	Description
Querying Flavors Supported by an Exclusive Microservice Engine	Query the flavors supported by an exclusive microservice engine.
Querying the Exclusive Microservice Engine List	Query the exclusive microservice engine list.
Creating an Exclusive Microservice Engine	Create an exclusive microservice engine.

API	Description
Querying Details About an Exclusive Microservice Engine	Query details about an exclusive microservice engine.
Deleting an Exclusive Microservice Engine	Delete an exclusive microservice engine.
Querying Details About an Exclusive Microservice Engine Job	Query details about an exclusive microservice engine job.

Authentication

Table 2-14 Authentication APIs

API	Description
Obtaining the User Token of an Exclusive Microservice Engine	This API is intended only for exclusive microservice engines with security authentication enabled. Before accessing APIs of an exclusive microservice engine with security authentication enabled, you need to call this API to obtain a user token. This token is the unique credential for accessing the APIs.

Microservice

Table 2-15 Microservice APIs

API	Description
Querying Information About a Microservice	Query the definition information about a microservice based on serviceld .
Deleting Definition Information About a Microservice	Delete the definition and related information about a microservice, and deregisters all instances of the microservice.
Querying Information About All Microservices	Query information about microservices that meet filter criteria.

API	Description
Creating Static Information for a Microservice	Create static information for a microservice before registering a microservice instance. The registered instance is associated with the static information based on the serviceId field. One service corresponds to multiple instances. The serviceId field can be customized. If serviceId is not customized, the system generates a random serviceId .
Deleting Static Information About Microservices in Batches	Delete the definitions and related information about microservices in batches, and deregisters all instances of the microservices.
Modifying Extended Attributes of a Microservice	Modify static information about a microservice. To update some fields in the static information, input all static information (including the fields that do not need to be updated) in JSON format.
Querying the Unique Service or Schema ID of a Microservice	Query the unique serviceId or schema ID of a microservice based on filter criteria.

Schema

Table 2-16 Schema APIs

API	Description
Querying a Microservice Schema	Query a microservice schema based on service_id and schema_id .
Modifying a Microservice Schema	Modify a microservice schema based on schema_id .
Querying All Schema Information About a Microservice	Query all schema information (including schemaid and summary) about a microservice.

Microservice Instance

Table 2-17 Microservice instance APIs

API	Description
Registering a Microservice Instance	Register a microservice instance after a microservice is created. Information about the instance must be provided during registration. instanceId can be customized. If it is customized, all contents are overwritten upon re-registration. If it is not customized, the system automatically generates an ID. If the endpoint content is duplicate, the original ID is used.
Querying a Microservice Instance Based on service_id	Query all instances of a microservice based on service_id after the instances are registered.
Deregistering a Microservice Instance	Deregister an instance based on instance_id after the instance is registered.
Querying Details About a Microservice Instance	Query details about an instance based on service_id and instance_id after the instance is registered.
Modifying the Extended Information About a Microservice Instance	Add or update the extended information about a microservice instance based on instance_id after the instance is registered.
Modifying Status of a Microservice Instance	Modify the status of a microservice instance based on instance_id after the instance is registered.
Sending Heartbeat Information	Send heartbeat information from service provider to service center, so that the service center can check whether service instances are normal.
Querying a Microservice Instance by Filter Criteria	Query a microservice instance based on microservice version rules or field filter criteria after the instance is registered.
Querying Microservice Instances in Batches	Query microservice instances in batches based on microservice version rules or field filter criteria after the instances are registered.

Dependency

Table 2-18 Dependency APIs

API	Description
Querying All Providers of a Microservice	Query all providers of a microservice based on consumer_id . For example, if service A depends on services B and C, you can query the information about services B and C based on service A ID.

Configuration Management

Table 2-19 Configuration management APIs

API	Description
Creating a Configuration	Create a configuration.
Modifying a Configuration	Modify a configuration.
Querying a Configuration	Query a configuration.
Querying the Configuration List	Query the configuration list.
Deleting a Configuration	Delete a configuration.
Deleting All Configurations	Delete all configurations.

3 Calling APIs

3.1 Making an API Request

This section describes the structure of a RESTful API request, and uses the IAM API for [creating an IAM user](#) as an example to describe how to call an API.

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.

- **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 **CN-Hong Kong** is **iam.ap-southeast-1.myhuaweicloud.com**.
- **resource-path:** Access path of an API for performing a specified operation. Obtain the value from the URI of an API. For example, **resource-path** of the API for creating an IAM user is **/v3.0/OS-USER/users**.
- **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, you need to create an IAM user. Since IAM is a global service, use the endpoint of any region (for example, iam.ap-southeast-1.myhuaweicloud.com of CN-Hong Kong) and obtain **resource-path (/v3.0/OS-USER/users)** in the URI of the API for [creating an IAM user as an administrator](#). Then, construct the request as follows:

```
https://iam.ap-southeast-1.myhuaweicloud.com/v3.0/OS-USER/users
```

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

HTTP-based request methods, which are also called operations or actions, indicate the type of operations the service is requesting.

- **GET**: requests the server to return specified resources.
- **PUT**: requests the server to update specified resources.
- **POST**: requests the server to add resources or perform special operations.
- **DELETE**: requests the server to delete specified resources, for example, an object.
- **HEAD**: requests a server resource header.
- **PATCH**: requests the server to update partial content of a specified resource. If the resource does not exist, a new resource will be created. If the resource is unavailable, the PATCH method is used to create a resource.

For example, in the URI of the API for [creating an IAM user as an administrator](#), the request method is **POST**. The request is as follows:

```
POST https://iam.ap-southeast-1.myhuaweicloud.com/v3.0/OS-USER/users
```

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:

- **Content-Type**: specifies the request body type or format. This field is mandatory and its default value is **application/json;charset=utf8**. Other values of this field will be provided for specific APIs if any.
- **Authorization**: signature authentication information. This field is optional. When AK/SK-based authentication is enabled, this field is automatically specified when SDK is used to sign the request. For details, see [AK/SK-based Authentication](#).
- **X-Sdk-Date**: time when a request is sent. This field is optional. When AK/SK-based authentication is enabled, this field is automatically specified when SDK is used to sign the request. For details, see [AK/SK-based Authentication](#).

- **X-Auth-Token**: specifies a user token only for token-based API authentication. The user token is a response to the API for [obtaining a user token](#). This API is the only one that does not require authentication.
- **X-Project-ID**: specifies a subproject ID. This parameter is optional and can be used in multi-project scenarios. The **X-Project-ID** field is mandatory in the request header for accessing resources in a subproject through AK/SK-based authentication.
- **X-Domain-ID**: account ID, which is optional. When you call APIs of global services using AK/SK-based authentication, **X-Domain-ID** needs to be configured in the request header.

For the API for [creating an IAM user as an administrator](#), if AK/SK-based authentication is enabled, the request with the header is as follows:

```
POST https://iam.ap-southeast-1.myhuaweicloud.com/v3.0/OS-USER/users
Content-Type: application/json
X-Sdk-Date: 20240416T095341Z
Authorization: SDK-HMAC-SHA256 Access=*****, SignedHeaders=content-type;host;x-sdk-date,
Signature=*****
```

Request Body

A request body is generally sent in structured format. It corresponds to **Content-Type** in the request header and transfers content except the request header. If the request body needs to support Chinese characters, set **Content-type** to **UTF-8**. For example, **Content-Type: application/json; charset=utf-8**.

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

For the API for [creating an IAM user as an administrator](#), you can check the required request parameters and their description in the API request. The following provides an example request with a body included. Replace the fields in bold with the actual values.

- **accountid**: ID of the account to which the IAM user belongs.
- **username**: IAM username to be created.
- **email**: email address of the IAM user.
- *********: login password of the IAM user.

```
POST https://iam.ap-southeast-1.myhuaweicloud.com/v3.0/OS-USER/users
Content-Type: application/json
X-Sdk-Date: 20240416T095341Z
Authorization: SDK-HMAC-SHA256 Access=*****, SignedHeaders=content-type;host;x-sdk-date,
Signature=*****

{
  "user": {
    "domain_id": "accountid",
    "name": "username",
    "password": "*****",
    "email": "email",
    "description": "IAM User Description"
  }
}
```

If all data required for the API request is available, you can send the request to call the API through curl, Postman, or coding.

3.2 Authentication

You can use either of the following authentication methods to call APIs:

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

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.
- You can use the AK/SK in either permanent or temporary access key. However, when the AK/SK of the temporary access key is used, the **X-Security-Token** field must be carried. The field value is the **security_token** of the temporary access key.

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

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

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

NOTICE

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

Token-based Authentication

NOTE

- The validity period of a token is 24 hours. If a token is used for authentication, cache it to prevent frequent API calling.
- Ensure that the token is valid when you use it. Using a token that will soon expire may cause API calling failures.

A token specifies temporary permissions in a computer system.

During API authentication using a token, the token is added to requests to get permissions for calling the API.

When calling the API for [obtaining a user token](#), you must set **auth.scope** in the request body to **project**.

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

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 **ABCDEFGF....**, add **X-Auth-Token: ABCDEFG....** to a request as follows:

```
POST https://iam.ap-southeast-1.myhuaweicloud.com/v3.0/OS-USER/users
Content-Type: application/json
X-Auth-Token: ABCDEFG....
```

3.3 Response

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

Status Code

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

If status code 201 is returned for the API for [creating an IAM user as an administrator](#), the request is successful.

Response Header

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

For the API for [creating an IAM user as an administrator](#), the message header shown in [Figure 3-2](#) is returned.

Figure 3-2 Response header

```
"X-Frame-Options": "SAMEORIGIN",
"X-IAM-ETag-id": "2562365939-d8f6f12921974cb097338ac11fcec8a",
"Transfer-Encoding": "chunked",
"Strict-Transport-Security": "max-age=31536000; includeSubdomains;",
"Server": "api-gateway",
"X-Request-Id": "af2953f2bcc67a42325a69a19e6c32a2",
"X-Content-Type-Options": "nosniff",
"Connection": "keep-alive",
"X-Download-Options": "noopen",
"X-XSS-Protection": "1; mode=block;",
"X-IAM-Trace-Id": "token_ null_af2953f2bcc67a42325a69a19e6c32a2",
"Date": "Tue, 21 May 2024 09:03:40 GMT",
"Content-Type": "application/json; charset=utf8"
```

Response Body

A response body is generally returned in structured format. It corresponds to **Content-Type** in the response header and transfers content except the response header.

For the API for [creating an IAM user as an administrator](#), the following message body is returned. The following describes part of the response body.

```
{
  "user": {
    "id": "c131886aec...",
    "name": "IAMUser",
    "description": "IAM User Description",
    "areacode": "",
    "phone": "",
    "email": "****@***.com",
    "status": null,
    "enabled": true,
    "pwd_status": false,
    "access_mode": "default",
    "is_domain_owner": false,
    "xuser_id": "",
    "xuser_type": "",
    "password_expires_at": null,
    "create_time": "2024-05-21T09:03:41.000000",
    "domain_id": "d78cbac1.....",
    "xdomain_id": "30086000.....",
    "xdomain_type": "",
    "xdomain_project_id": null
  }
}
```

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

```
{
  "error_msg": "Request body is invalid.",
  "error_code": "IAM.0011"
}
```

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

4 Application Management V3 APIs

4.1 Environment

4.1.1 Creating an Environment

Function

This API is used to create an environment.

URI

POST /v3/{project_id}/cas/environments

Table 4-1 Path parameter

Parameter	Mandator y	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .

Request

Table 4-2 Request header parameters

Parameter	Mandator y	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .

Table 4-3 Request body parameters

Parameter	Mandatory	Type	Description
config_mode	No	String	Configuration mode. • ui
name	Yes	String	Environment name. The value contains 2 to 64 characters consisting of letters, digits, hyphens (-), or underscores (_). It starts with a letter and ends with a letter or digit. Letters are case sensitive.
deploy_mode	No	String	Environment type. • virtualmachine • container: Kubernetes • mixed: VM and Kubernetes To use a custom server in the created environment, select virtualmachine or mixed .
type	No	String	Whether the Kubernetes environment is of the HA type. • normal: non-HA environment. • high: HA environment.
resource_create_mode	No	String	Resource configuration mode. • none: resource management. • iac: resource creation. To use a custom server in the created environment, select none .
resources	No	Array of objects	Resource list. See Table 4-4 .

Parameter	Mandatory	Type	Description
description	No	String	Environment description. The value can contain up to 128 characters.
enterprise_project_id	No	String	Enterprise project ID. If this parameter is not set, the default enterprise project is default and the ID is 0 .
vpc_id	Yes	String	VPC ID.
labels	No	Array of objects	Label. You can use TMS to filter resources by labels. See Table 4-5 . Each environment can have a maximum of 20 labels.
environment_azs	No	Array of objects	List of AZs. When type is set to high , you need to set two AZs.

Table 4-4 resources

Parameter	Mandatory	Type	Description
id	Yes	String	Resource ID. (To ensure that the API can be called properly, enter the correct resource ID.)
name	No	String	Resource name.

Parameter	Mandatory	Type	Description
type	Yes	String	Resource type. • eip • elb • cce • ecs • dcs • rds • cse: ServiceComb engine • nacos: registry/configuration center • vpc • subnet • customServer • acl • gaussdb • dms_rocketmq • css_es: Elasticsearch

Table 4-5 labels

Parameter	Mandatory	Type	Description
key	Yes	String	Label name. Labels are used to identify cloud resources. When you have multiple cloud resources of the same type, you can use labels to classify them based on usage, owner, or environment. The value contains 1 to 36 characters, including digits, letters, underscores (_), or hyphens (-).
value	Yes	String	Label value. The value contains 1 to 43 characters, including digits, letters, underscores (_), periods (.), or hyphens (-).

Response

Table 4-6 Response parameters

Parameter	Type	Description
id	String	Environment ID.
name	String	Environment name.
description	String	Environment description.
project_id	String	Project ID.
enterprise_project_id	String	Enterprise project ID.
deploy_mode	String	Environment type. ● virtualmachine ● container: Kubernetes ● mixed: VM and Kubernetes.
vpc_id	String	VPC ID.
creator	String	Environment creator.
create_time	Integer	Environment creation time.
update_time	Integer	Environment update time.
labels	Array of objects	Label. You can use TMS to query resources by labels. See Table 4-7 .
type	String	Whether the Kubernetes environment is of the HA type. ● normal: non-HA environment. ● high: HA environment.
resource_create_mode	String	Resource configuration mode. ● none: resource management.
config_mode	String	Configuration mode. ● ui
env_status	String	Non-real-time environment resource status. ● REFRESHING ● UNKNOWN ● NORMAL ● PARTIALLY_NORMAL

Table 4-7 labels

Parameter	Type	Description
key	String	Label name.
value	String	Label value.

Example Request

Create a VM environment **env-test**. The ID of the VPC where the environment is located is **0c72428b-cd6c-4283-a560-210d3edb49f7**.

```
{
  "name": "env-test",
  "description": "",
  "enterprise_project_id": "0",
  "vpc_id": "0c72428b-cd6c-4283-a560-210d3edb49f7",
  "labels": [
    {
      "key": "aaaaa",
      "value": "a5"
    }
  ],
  "deploy_mode": "virtualmachine"
}
```

Example Response

```
{
  "id": "c93a5313-d141-4b9f-97ce-b4ce2b3988bd",
  "name": "test",
  "description": "",
  "project_id": "13ae26238d724e54947af3e0fcbe9c3",
  "enterprise_project_id": "0",
  "vpc_id": "0c72428b-cd6c-4283-a560-210d3edb49f7",
  "creator": "ss-test",
  "create_time": 1681384236023,
  "update_time": 1681384236023,
  "deploy_mode": "virtualmachine",
  "labels": []
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.1.2 Obtaining All Environments

Function

This API is used to obtain all created environments.

URI

GET /v3/{project_id}/cas/environments

Table 4-8 Path parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .

Table 4-9 Query parameters

Parameter	Mandatory	Type	Description
limit	No	Integer	Number of records to be queried. Value range: 0–100.
offset	No	Integer	Query offset, which indicates the result after the Nth data record is queried.
order_by	No	String	Sorting field. Numerated values: • create_time (default) • name • update_time If the transferred value is not within the supported enumerated value range, the default sorting field is used.
order	No	String	Sorting order. • desc (default) • asc
name	No	String	Environment name.
environment_id	No	String	Environment ID.

Parameter	Mandatory	Type	Description
enterprise_project_id	No	String	Enterprise project ID.

Request

Table 4-10 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

Table 4-11 Response parameters

Parameter	Type	Description
count	Integer	Total number of environments.
environments	Array of objects	Environment parameters. See Table 4-12 .

Table 4-12 environments

Parameter	Type	Description
id	String	Environment ID.
name	String	Environment name.
description	String	Environment description.
project_id	String	Project ID.

Parameter	Type	Description
enterprise_project_id	String	Enterprise project ID.
deploy_mode	String	Environment type. - virtualmachine - container: Kubernetes - mixed: VM and Kubernetes.
vpc_id	String	VPC ID.
creator	String	Environment creator.
create_time	Integer	Environment creation time.
update_time	Integer	Environment update time.
labels	Array of objects	Label. You can use TMS to filter resources by labels. See Table 4-13 .
type	String	Whether the Kubernetes environment is of the HA type. - normal: non-HA environment. - high: HA environment.
resource_create_mode	String	Resource configuration mode. none: resource management.
config_mode	String	Configuration mode. ui
env_status	String	Non-real-time environment resource status. - REFRESHING - UNKNOWN - NORMAL - PARTIALLY_NORMAL

Table 4-13 labels

Parameter	Type	Description
key	String	Label name.
value	String	Label value.

Example Request

None

Example Response

```
{
  "environments": [
    {
      "id": "00078e9d-a61c-476e-ac63-a10c9cb2638e",
      "name": "development-env",
      "description": "",
      "project_id": "e7d2e9c589e5445e808a8ff0d1235aca",
      "enterprise_project_id": "0",
      "deploy_mode": "virtualmachine",
      "vpc_id": "29d55020-ae0e-4a18-871c-93e6976ee7bd",
      "creator": "ss-test",
      "create_time": 1610418873730,
      "update_time": 1610418873730,
      "labels": null
    }
  ],
  "count": 1
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.1.3 Deleting an Environment Based on the Environment ID

Function

This API is used to delete an environment based on the environment ID.

URI

DELETE /v3/{project_id}/cas/environments/{environment_id}

Table 4-14 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .

Parameter	Mandatory	Type	Description
environment_id	Yes	String	Environment ID. See Obtaining All Environments .

Request

Table 4-15 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

None

Example Request

None

Example Response

None

Status Code

Status Code	Description
204	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.1.4 Modifying an Environment Based on the Environment ID

Function

This API is used to modify an environment based on the environment ID.

URI

PUT /v3/{project_id}/cas/environments/{environment_id}

Table 4-16 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
environment_id	Yes	String	Environment ID. See Obtaining All Environments .

Request

Table 4-17 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Table 4-18 Request body parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Environment name. The value contains 2 to 64 characters consisting of letters, digits, hyphens (-), and underscores (_). It starts with a letter and ends with a letter or digit. Letters are case sensitive.

Parameter	Mandatory	Type	Description
description	No	String	Environment description. The value can contain up to 128 characters.
labels	No	Array of objects	Label. You can use TMS to filter resources by labels. See Table 4-19 . Each environment can have a maximum of 20 labels.
enterprise_project_id	No	String	Enterprise project ID. If this parameter is not set, the default enterprise project is default and the ID is 0 .
configuration	No	Object	Environment variable of the environment. See Table 4-20 .

Table 4-19 labels

Parameter	Mandatory	Type	Description
key	Yes	String	Label name. Labels are used to identify cloud resources. When you have multiple cloud resources of the same type, you can use labels to classify them based on usage, owner, or environment. The value contains 1 to 36 characters, including digits, letters, underscores (_), or hyphens (-).
value	Yes	String	Label value. The value contains 1 to 43 characters, including digits, letters, underscores (_), periods (.), or hyphens (-).

Table 4-20 configuration

Parameter	Mandatory	Type	Description
resource_envs	No	Array of objects	Resource variable bound to the environment. See Table 4-21 .

Parameter	Mandatory	Type	Description
envs	No	Array of objects	Environment variable of the environment. See Table 4-22 .

Table 4-21 resource_env

Parameter	Mandatory	Type	Description
resource_id	Yes	String	ID of the resource associated with the resource variable bound to the environment.
resource_name	Yes	String	Name of the resource associated with the resource variable bound to the environment.
resource_type	Yes	String	Type of the resource associated with the resource variable bound to the environment.
envs	Yes	Array of objects	Resource variable bound to the environment. See Table 4-22 .

Table 4-22 config_env

Parameter	Mandatory	Type	Description
name	Yes	String	Key of the environment variable of the environment.
value	Yes	String	Value of the environment variable of the environment.

Response

Table 4-23 Response parameters

Parameter	Type	Description
id	String	Environment ID.
name	String	Environment name.
description	String	Environment description.

Parameter	Type	Description
project_id	String	Project ID.
enterprise_project_id	String	Enterprise project ID.
deploy_mode	String	Environment type.
vpc_id	String	VPC ID.
creator	String	Environment creator.
create_time	Integer	Environment creation time.
update_time	Integer	Environment update time.
labels	Array of objects	Label. You can use TMS to filter resources by labels. See Table 4-24 .
resources	Array of objects	Resource. See Table 4-25 .
resource_create_mode	String	Resource configuration mode. • none: resource management.
config_mode	String	Configuration mode. • ui
configuration	Object	Environment variable of the environment. See Table 4-27 .

Table 4-24 labels

Parameter	Type	Description
key	String	Label name.
value	String	Label value.

Table 4-25 resources

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

Parameter	Type	Description
type	String	Resource type. • vpc • eip • elb • cce • ecs • as • cse: ServiceComb engine • nacos: registry/configuration center • dcs • rds • subnet • acl • gaussdb • dms_rocketmq • customServer • css_es: Elasticsearch • custom_k8s: custom Kubernetes cluster
ip	String	IP address of the custom server when the resource type is set to customServer .
parameters	Map<String, String>	Resource parameters. See Table 4-26 .
alias	String	Resource alias.

Table 4-26 parameters

Parameter	Type	Description
kube_config	String	Kubernetes connection configuration file. This parameter is returned when the resource type is custom_k8s .

Table 4-27 configuration

Parameter	Type	Description
resource_envs	Array of objects	Resource variable bound to the environment. See Table 4-28 .

Parameter	Type	Description
envs	Array of objects	Environment variable of the environment. See Table 4-29 .

Table 4-28 resource_env

Parameter	Type	Description
resource_id	String	ID of the resource associated with the resource variable bound to the environment.
resource_name	String	Name of the resource associated with the resource variable bound to the environment.
resource_type	String	Type of the resource associated with the resource variable bound to the environment.
envs	Array of objects	Resource variable bound to the environment. See Table 4-29 .

Table 4-29 config_env

Parameter	Type	Description
name	String	Key of the environment variable of the environment.
value	String	Value of the environment variable of the environment.

Example Request

Modify the environment name to **development-env**, leave the description blank, and add a label whose key-value pair is **"test": "value"** to the environment.

```
{
  "name": "development-env",
  "description": "",
  "labels": [
    {
      "key": "test",
      "value": "value"
    }
  ]
}
```

Example Response

```
{
  "id": "00078e9d-a61c-476e-ac63-a10c9cb2638e",
  "name": "development-env",
  "description": "",

```

```
{
  "project_id": "e7d2e9c589e5445e808a8ff0d1235aca",
  "enterprise_project_id": "0",
  "deploy_mode": "virtualmachine",
  "vpc_id": "29d55020-ae0e-4a18-871c-93e6976ee7bd",
  "creator": "ss-test",
  "create_time": 1610418873730,
  "update_time": 1610418873730,
  "resources": [
    {
      "id": "329b135e-7b31-4ea6-afb9-b69017bd0a80",
      "type": "ecs",
      "name": "ecs-test"
    }
  ],
  "labels": [
    {
      "key": "test",
      "value": "value"
    }
  ]
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.1.5 Obtaining Environment Details Based on the Environment ID

Function

This API is used to obtain environment details based on the environment ID.

URI

GET /v3/{project_id}/cas/environments/{environment_id}

Table 4-30 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .

Parameter	Mandatory	Type	Description
environment_id	Yes	String	Environment ID. See Obtaining All Environments .

Request

Table 4-31 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

Table 4-32 Response parameters

Parameter	Type	Description
id	String	Environment ID.
name	String	Environment name.
description	String	Environment description.
project_id	String	Project ID.
enterprise_project_id	String	Enterprise project ID.
deploy_mode	String	Environment type. - virtualmachine - container: Kubernetes - mixed: VM and Kubernetes.
vpc_id	String	VPC ID.
creator	String	Environment creator.
create_time	Integer	Environment creation time.

Parameter	Type	Description
update_time	Integer	Environment update time.
labels	Array of objects	Label. You can use TMS to filter resources by labels. See Table 4-33 .
resources	Array of objects	Resource. See Table 4-34 .
type	String	Whether the Kubernetes environment is of the HA type. • normal: non-HA environment. • high: HA environment.
resource_create_mode	String	Resource configuration mode. • none: resource management.
config_mode	String	Configuration mode. • ui
env_status	String	Environment resource status (non-real-time status). • REFRESHING • UNKNOWN • NORMAL • PARTIALLY_NORMAL
configuration	Object	Environment variable of the environment. See Table 4-35 .

Table 4-33 labels

Parameter	Type	Description
key	String	Label name.
value	String	Label value.

Table 4-34 resources

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

Parameter	Type	Description
type	String	Resource type. • vpc • eip • elb • cce • ecs • as • cse: ServiceComb engine • nacos: registry/configuration center • dcs • rds • subnet • acl • customServer • gaussdb • dms_rocketmq • css_es: Elasticsearch
core	String	Number of cores of the custom server. This parameter is returned when the resource type is customServer .
gib	String	Memory of the custom server, in Gib. This parameter is returned when the resource type is customServer .

Table 4-35 configuration

Parameter	Type	Description
resource_envs	Array of objects	Resource variable bound to the environment. See Table 4-36 .
envs	Array of objects	Environment variable of the environment. See Table 4-37 .

Table 4-36 resource_env

Parameter	Type	Description
resource_id	String	ID of the resource associated with the resource variable bound to the environment.

Parameter	Type	Description
resource_name	String	Name of the resource associated with the resource variable bound to the environment.
resource_type	String	Type of the resource associated with the resource variable bound to the environment.
envs	Array of objects	Resource variable bound to the environment. See Table 4-37 .

Table 4-37 config_env

Parameter	Type	Description
name	String	Key of the environment variable of the environment.
value	String	Value of the environment variable of the environment.

Example Request

None

Example Response

```
{
  "id": "ea011e01-2eb5-453f-87bf-874e4a855abe",
  "name": "dev-env",
  "description": "develop environment",
  "project_id": "bf8523d898b64e4eb956e3be3555ca16",
  "enterprise_project_id": "0",
  "deploy_mode": "virtualmachine",
  "vpc_id": "234241234124xxvasf2342xxxxxxxxxx",
  "resources": [
    {
      "id": "329b135e-7b31-4ea6-afb9-b69017bd0a80",
      "type": "ecs",
      "name": "ecs-test"
    }
  ],
  "creator": "ss-test",
  "create_time": 1578984198394,
  "update_time": 1578984198394,
  "labels": null
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request

Status Code	Description
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.1.6 Modifying an Environment Resource Based on the Environment ID

Function

This API is used to modify an environment resource based on environment ID.

URI

PUT /v3/{project_id}/cas/environments/{environment_id}/resources

Table 4-38 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
environment_id	Yes	String	Environment ID. See Obtaining All Environments .

Request

Table 4-39 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .

Table 4-40 Request body parameters

Parameter	Mandatory	Type	Description
resources	Yes	Array of objects	Resource to be added. See Table 4-41 .

Table 4-41 resources

Parameter	Mandatory	Type	Description
id	Yes	String	Resource ID. (To ensure that the API can be called properly, enter the correct resource ID.)
name	No	String	Resource name.
type	Yes	String	Resource type. • eip • elb • cce • ecs • as • cse: ServiceComb engine • nacos: registry/configuration center • dcs • rds • subnet • acl • gaussdb • dms_rocketmq • css_es: Elasticsearch

Response

Table 4-42 Response parameters

Parameter	Type	Description
resources	Array of objects	Resource. See Table 4-43 .

Table 4-43 schemas

Parameter	Type	Description
id	String	Resource ID.
name	String	Resource name.
type	String	Resource type. • eip • elb • cce • ecs • as • cse • dcs • rds • gaussdb • dms_rocketmq • css_es: Elasticsearch

Example Request

Modify the ECS whose ID is **67835bb3-1235-4cc9-be71-becbb2b4ca0d** and name is **ecs-a5a9**.

```
{
  "resources": [
    {
      "id": "67835bb3-1235-4cc9-be71-becbb2b4ca0d",
      "name": "ecs-a5a9",
      "type": "ecs"
    }
  ]
}
```

Example Response

```
{
  "resources": [
    {
      "id": "67835bb3-1235-4cc9-be71-becbb2b4ca0d",
      "type": "ecs",
      "name": "ecs-a5a9"
    }
  ]
}
```

Status Code

Status Code	Description
200	OK

Status Code	Description
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.1.7 Querying a Managed Resource Based on the Environment ID

Function

This API is used to query a managed resource based on the environment ID.

URI

GET /v3/{project_id}/cas/environments/{environment_id}/resources

Table 4-44 Path parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
environment_id	Yes	String	Environment ID. See Obtaining All Environments .

Request

Table 4-45 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

Table 4-46 Response parameters

Parameter	Type	Description
resources	Array of objects	Resource. See Table 4-47 .

Table 4-47 schemas

Parameter	Type	Description
id	String	Resource ID.
name	String	Resource name.
type	String	Resource type. • eip • elb • cce • ecs • dcs • rds • cse: ServiceComb engine • nacos: registry/configuration center • subnet • acl • gaussdb • dms_rocketmq • css_es: Elasticsearch

Example Request

None

Example Response

```
{
  "resources": [
    {
      "id": "67835bb3-1235-4cc9-be71-becbb2b4ca0d",
      "type": "ecs",
      "name": "ecs-a5a9"
    }
  ]
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.1.8 Updating an Environment Based on the Environment ID

Function

This API is used to deliver an environment update job.

URI

POST /v3/{project_id}/cas/environments/{environment_id}/refresh

Table 4-48 Path parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
environment_id	Yes	String	Environment ID. See Obtaining All Environments .

Request

Table 4-49 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

Table 4-50 Response parameters

Parameter	Type	Description
job_id	String	Update job ID.
environment_id	String	Environment ID.

Example Request

None

Example Response

None

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.2 Application

4.2.1 Creating an Application

Function

An application is a service system with functions and consists of one or more components.

This API is used to create an application.

URI

POST /v3/{project_id}/cas/applications

Table 4-51 Path parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .

Request

Table 4-52 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Table 4-53 Request body parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Application name. The value contains 2 to 64 characters consisting of letters, digits, hyphens (-), or underscores (_). It starts with a letter and ends with a letter or digit.
description	No	String	Application description. The value can contain up to 128 characters.
enterprise_project_id	No	String	Enterprise project ID. If this parameter is not set, the default enterprise project is default and the ID is 0 .
labels	No	Array of objects	Label. You can use TMS to filter resources by labels. See Table 4-54 . Each application can have a maximum of 20 labels.

Table 4-54 labels

Parameter	Mandatory	Type	Description
key	Yes	String	Label name. Labels are used to identify cloud resources. When you have multiple cloud resources of the same type, you can use labels to classify them based on usage, owner, or environment. The value contains 1 to 36 characters, including digits, letters, underscores (_), or hyphens (-).
value	Yes	String	Label value. The value contains 1 to 43 characters, including digits, letters, underscores (_), periods (.), or hyphens (-).

Response

Table 4-55 Response parameters

Parameter	Type	Description
id	String	Application ID.
name	String	Application name.
description	String	Application description.
creator	String	Application creator.
project_id	String	Project ID.
enterprise_project_id	String	Enterprise project ID.
create_time	Integer	Application creation time.
update_time	Integer	Application update time.
labels	Array of objects	Label. You can use TMS to filter resources by labels. See Table 4-56 .

Table 4-56 labels

Parameter	Type	Description
key	String	Label name.
value	String	Label value.

Example Request

Create application **app-xpmtii**, set the description to **test**, and set the label key-value pair to **"key":"var"**.

```
{
  "name": "app-xpmtii",
  "description": "test",
  "labels": [
    {
      "key": "key",
      "value": "var"
    }
  ]
}
```

Example Response

```
{
  "id": "dae29983-c1e9-437d-a6be-5256076fe605",
  "name": "app-xpmtii",
  "description": "test",
  "project_id": "063a3158b1a34710b36ad208b9497d00",
}
```

```
{
  "enterprise_project_id": "0",
  "creator": "ss-test",
  "create_time": 1679468852302,
  "update_time": 1679468852302,
  "labels": [
    {
      "key": "key",
      "value": "var"
    }
  ]
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.2.2 Obtaining All Applications

Function

This API is used to obtain all created applications.

URI

GET /v3/{project_id}/cas/applications

Table 4-57 Path parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. See Obtaining a Project ID .

Table 4-58 Query parameters

Parameter	Mandatory	Type	Description
limit	No	integer	Number of records to be queried. Value range: 0–100.

Parameter	Mandatory	Type	Description
offset	No	integer	Query offset, which indicates the result after the Nth data record is queried.
order_by	No	String	Sorting field. Numerated values: • create_time (default) • name • update_time If the transferred value is not within the supported enumerated value range, the default sorting field is used.
order	No	String	Sorting order. • desc (default) • asc

Request

Table 4-59 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

Table 4-60 Response parameters

Parameter	Type	Description
count	Integer	Total number of applications.
applications	Array of object	Application information. See Table 4-61 .

Table 4-61 applications

Parameter	Type	Description
id	String	Application ID.
name	String	Application name.
description	String	Application description.
creator	String	Application creator.
project_id	String	Project ID.
enterprise_project_id	String	Enterprise project ID.
create_time	Integer	Application creation time.
update_time	Integer	Application update time.
component_count	Integer	Number of components deployed in an application.
labels	Array of objects	Label. You can use TMS to filter resources by labels. See Table 4-62 .

Table 4-62 labels

Parameter	Type	Description
key	String	Label name.
value	String	Label value.

Example Request

None

Example Response

```
{
  "count": 1,
  "applications": [
    {
      "id": "dae29983-c1e9-437d-a6be-5256076fe605",
      "name": "app-xpmtii",
      "description": "test",
      "project_id": "063a3158b1a34710b36ad208b9497d00",
      "enterprise_project_id": "0",
      "creator": "ss-test",
      "create_time": 1679468852302,
      "update_time": 1679468852302,
      "labels": [
        {
          "key": "key",
          "value": "var"
        }
      ]
    }
  ]
}
```

```
    ],  
    "component_count": 0  
  }  
]  
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.2.3 Modifying Application Information Based on the Application ID

Function

This API is used to modify application information based on the application ID.

URI

PUT /v3/{project_id}/cas/applications/{application_id}

Table 4-63 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
application_id	Yes	String	Application ID. See Obtaining All Applications .

Request

Table 4-64 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Table 4-65 Request body parameters

Parameter	Mandatory	Type	Description
name	No	String	Application name. The value contains 2 to 64 characters consisting of letters, digits, hyphens (-), or underscores (_). It starts with a letter and ends with a letter or digit.
description	No	String	Application description. The value can contain up to 128 characters.
enterprise_project_id	No	String	Enterprise project ID. If this parameter is not set, the default enterprise project is default and the ID is 0 .
labels	No	Array of objects	Label. You can use TMS to filter resources by labels. See Table 4-66 . Each application can have a maximum of 20 labels.

Table 4-66 labels

Parameter	Man dato ry	Type	Description
key	Yes	String	Label name. Labels are used to identify cloud resources. When you have multiple cloud resources of the same type, you can use labels to classify them based on usage, owner, or environment. The value contains 1 to 36 characters, including digits, letters, underscores (_), or hyphens (-).
value	Yes	String	Label value. The value contains 1 to 43 characters, including digits, letters, underscores (_), periods (.), or hyphens (-).

Response

Table 4-67 Response parameters

Parameter	Type	Description
id	String	Application ID.
name	String	Application name.
description	String	Application description.
creator	String	Application creator.
project_id	String	Project ID.
enterprise_project_id	String	Enterprise project ID.
create_time	Integer	Application creation time.
update_time	Integer	Application update time.
component_count	Integer	Number of components deployed in an application.
labels	Array of objects	Label. You can use TMS to filter resources by labels. See Table 4-68 .

Table 4-68 labels

Parameter	Type	Description
key	String	Label name.
value	String	Label value.

Example Request

Modify the application name to **app-xpmtii2**, description to **test2**, and label key-value pair to **"key":"var"**.

```
{
  "name": "app-xpmtii2",
  "description": "test2",
  "labels": [
    {
      "key": "key",
      "value": "var"
    }
  ]
}
```

Example Response

```
{
  "id": "dae29983-c1e9-437d-a6be-5256076fe605",
  "name": "app-xpmtii2",
  "description": "test2",
  "project_id": "063a3158b1a34710b36ad208b9497d00",
  "creator": "ss-test",
  "create_time": 1679468852302,
  "update_time": 1679470928765,
  "component_count": 0,
  "labels": [
    {
      "key": "key",
      "value": "var"
    }
  ]
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.2.4 Deleting an Application Based on the Application ID

Function

This API is used to delete an application based on the application ID.

URI

DELETE /v3/{project_id}/cas/applications/{application_id}

Table 4-69 Path parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
application_id	Yes	String	Application ID. See Obtaining All Applications .

Request

Table 4-70 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

None

Example Request

None

Example Response

None

Status Code

Status Code	Description
204	OK
400	Bad Request
404	Not Found
409	Conflict
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.2.5 Obtaining Application Details Based on the Application ID

Function

This API is used to obtain application details based on the application ID.

URI

GET /v3/{project_id}/cas/applications/{application_id}

Table 4-71 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
application_id	Yes	String	Application ID. See Obtaining All Applications .

Request

Table 4-72 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

Table 4-73 Response parameters

Parameter	Type	Description
id	String	Application ID.
name	String	Application name.
description	String	Application description.
creator	String	Application creator.
project_id	String	Project ID.
enterprise_project_id	String	Enterprise project ID.
create_time	Integer	Application creation time.
update_time	Integer	Application update time.
component_count	Integer	Number of components deployed in an application.
labels	Array of objects	Label. You can use TMS to filter resources by labels. See Table 4-74 .

Table 4-74 labels

Parameter	Type	Description
key	String	Label name.
value	String	Label value.

Example Request

None

Example Response

```
{
  "id": "e6129197-f9ca-4d60-b84d-a5e71cd2d317",
  "name": "app-test-0317_03",
  "description": "",
  "project_id": "063a3158b1a34710b36ad208b9497d00",
  "creator": "ss-test",
  "create_time": 1679036601735,
  "update_time": 1679036601735,
  "labels": null,
  "component_count": 0
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.2.6 Obtaining Application Configurations Based on the Application ID

Function

This API is used to obtain application configurations based on the application ID.

URI

GET /v3/{project_id}/cas/applications/{application_id}/configuration

Table 4-75 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .

Parameter	Mandatory	Type	Description
application_id	Yes	String	Application ID. See Obtaining All Applications .

Table 4-76 Query parameter

Parameter	Mandatory	Type	Description
environment_id	No	String	Environment ID. See Obtaining All Environments .

Request

Table 4-77 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

Table 4-78 Response parameter

Parameter	Type	Description
configuration	Array of objects	Configuration list. See Table 4-79 .

Table 4-79 configuration

Parameter	Type	Description
application_id	String	Application ID.

Parameter	Type	Description
environment_id	String	Environment ID.
configuration	Object	Environment variable configuration. See Table 4-80 .

Table 4-80 configurationEnv

Parameter	Type	Description
env	Array of object	Environment variable. See Table 4-81 .

Table 4-81 env

Parameter	Type	Description
name	String	Variable name.
value	String	Variable value.

Example Request

None

Example Response

```
{
  "configuration": [
    {
      "application_id": "fdd7889d-eef8-4db1-befc-7037b02d949e",
      "environment_id": "0fa75dfe-0d32-4bc1-848b-8008cf3f2567",
      "configuration": {
        "env": [
          {
            "name": "app-env",
            "value": "env"
          }
        ]
      }
    }
  ]
}
```

Status Code

Status Code	Description
200	OK

Status Code	Description
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.2.7 Modifying Application Configurations Based on the Application ID

Function

This API is used to modify application configurations based on the application ID.

URI

PUT /v3/{project_id}/cas/applications/{application_id}/configuration

Table 4-82 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
application_id	Yes	String	Application ID. See Obtaining All Applications .

Request

Table 4-83 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Table 4-84 Request body parameters

Parameter	Mandatory	Type	Description
environment_id	Yes	String	Environment ID. See Obtaining All Environments .
configuration	Yes	Object	Environment variable configuration. See Table 4-85 .

Table 4-85 configuration

Parameter	Mandatory	Type	Description
env	Yes	Array of objects	Environment variable. See Table 4-86 .

Table 4-86 env

Parameter	Mandatory	Type	Description
name	Yes	String	Application environment variable name, which must be unique in the same application environment. The value contains 1 to 64 characters, including letters, digits, underscores (_), hyphens (-), or periods (.). It starts with a letter, underscore (_), or hyphen (-).
value	Yes	String	Application environment variable value.

Response

Table 4-87 Response parameters

Parameter	Type	Description
application_id	String	Application ID.
environment_id	String	Environment ID.

Parameter	Type	Description
configuration	Object	Environment variable configuration. See Table 4-88 .

Table 4-88 configuration

Parameter	Type	Description
env	Array of object	Environment variable. See Table 4-89 .

Table 4-89 env

Parameter	Type	Description
name	String	Variable name.
value	String	Variable value.

Example Request

Modify the environment whose ID is **0fa75dfe-0d32-4bc1-848b-8008cf3f2567** by changing the name to **app-env** and the value to **env**.

```
{
  "environment_id": "0fa75dfe-0d32-4bc1-848b-8008cf3f2567",
  "configuration": {
    "env": [
      {
        "name": "app-env",
        "value": "env"
      }
    ]
  }
}
```

Example Response

```
{
  "application_id": "fdd7889d-eef8-4db1-befc-7037b02d949e",
  "environment_id": "0fa75dfe-0d32-4bc1-848b-8008cf3f2567",
  "configuration": {
    "env": [
      {
        "name": "app-env",
        "value": "env"
      }
    ]
  }
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.2.8 Deleting Application Configurations Based on the Application ID

Function

This API is used to delete application configurations based on the application ID.

URI

DELETE /v3/{project_id}/cas/applications/{application_id}/configuration

Table 4-90 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
application_id	Yes	String	Application ID. See Obtaining All Applications .

Table 4-91 Query parameter

Parameter	Mandatory	Type	Description
environment_id	Yes	String	Environment ID. See Obtaining All Environments .

Request

Table 4-92 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

None

Example Request

None

Example Response

None

Status Code

Status Code	Description
204	OK
400	Bad Request
404	Not Found
409	Conflict
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.3 Component

4.3.1 Creating an Application Component

Function

This API is used to create an application component.

URI

POST /v3/{project_id}/cas/applications/{application_id}/components

Table 4-93 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
application_id	Yes	String	Application ID. See Obtaining All Applications .

Request

Table 4-94 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Table 4-95 Request body parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Component name. The value contains 2 to 64 characters consisting of letters, digits, hyphens (-), or underscores (_). It starts with a letter and ends with a letter or digit.

Parameter	Mandatory	Type	Description
workload_name	No	String	Workload name.
labels	No	Array of objects	Label. For container-deployed components, you can query a workload by label. For VM-deployed components, you can configure a label of the corresponding microservice name to use graceful startup and shutdown. See Table 4-96 .
create_mode	No	String	Component creation mode. - If this parameter is not transferred (default), create and deploy a component. - If this parameter is transferred and its value is drafts, create a drafted component.
runtime_stack	Yes	Object	Runtime. See Table 4-122 . The value can be obtained from the response of the API for Querying a Technology Stack .
environment_id	No	String	Environment ID, which can be obtained from the information returned by Obtaining All Environments .
description	No	String	Component description. The value can contain up to 128 characters.
source	Yes	Object	Source of the code or software package. See Table 4-123 . This parameter is optional when create_mode is set to drafts .
build	No	Object	Component building. This parameter is mandatory when the component is deployed in containers and no image is used. For details, see Table 4-124 .
limit_cpu	No	Number	Maximum CPU limit. Unit: core.
limit_memory	No	Number	Maximum memory size. Unit: GiB.
request_cpu	No	Number	Requested CPU resources. Unit: core.
request_memory	No	Number	Requested memory. Unit: GiB.

Parameter	Mandatory	Type	Description
version	Yes	String	Component version number. The value can contain up to 32 characters. It needs to meet the following rule: <code>^([0-9]+)(.[0-9]+){2,3}\$</code> This parameter is optional when create_mode is set to drafts .
envs	No	Array of objects	Environment variable list. See Table 4-120 .
replica	Yes	Integer	Number of instances. This parameter is optional when create_mode is set to drafts .
storages	No	Array of objects	Storage. See Table 4-97 .
deploy_strategy	No	Object	Component deployment. See Table 4-100 .
command	No	Object	Startup command. See Table 4-104 .
post_start	No	Object	Post-start processing. See Table 4-105 .
pre_stop	No	Object	Pre-stop processing. See Table 4-105 .
mesher	No	Object	Multi-language access service mesh. This parameter is mandatory when Node.js is selected for the technology stack and the microservice engine is bound. See Table 4-106 .
timezone	No	String	Time zone where the component runs.
jvm_opts	No	String	JVM parameter. Example: -Xms256m -Xmx1024m. If there are multiple parameters, separate them by spaces. If the parameter is left blank, the default value is used.
tomcat_opts	No	Object	Tomcat parameter. This parameter is valid when you use the Tomcat technology stack. See Table 4-107 .

Parameter	Mandatory	Type	Description
host_aliases	No	Array of objects	Host alias. See Table 4-108 .
dns_policy	No	String	DNS policy. • Default: inherits the domain name resolution configuration of the node where the pod is located. • ClusterFirst: adds the domain name resolution configuration. • ClusterFirstWithHostNet: takes effect only for pods running in hostNetwork mode. • None: replaces the domain name resolution policy.
dns_config	No	Object	DNS configuration. See Table 4-109 .
workload_kind	No	String	Workload type. • deployment • statefulset • daemonset
security_context	No	Object	Security policy. See Table 4-111 .
logs	No	Array of objects	Log collection. See Table 4-113 .
custom_metrics	No	Object	Custom metrics. See Table 4-114 .
affinity	No	Array of objects	Affinity. See Table 4-115 .
anti_affinity	No	Array of objects	Anti-affinity. See Table 4-115 .
liveness_probe	No	Object	Component liveness probe. See Table 4-117 .
readiness_probe	No	Object	Component service probe. See Table 4-117 .

Parameter	Mandatory	Type	Description
refer_resources	Yes	Array of objects	Associated resource. See Table 4-118 .
external_accesses	No	Array of objects	External access. See Table 4-126 .
tolerations	No	Array of objects	Tolerance policy. See Table 4-127 .
tolerations	No	Array of objects	Tolerance policy. See Table 4-127 .
container_name	No	String	Container name.
update_strategy	No	Object	Component upgrade strategy. See Table 4-130 .
multi_containers	No	Array of objects	Multi-container configuration. See Table 4-128 .
service_name	No	String	Service name configured for the inter-instance discovery of a StatefulSet.
ports	No	Array of objects	Port configured for the inter-instance discovery of a StatefulSet.
workload_content	No	String	Workload content.
configs	No	Array of objects	Unified component configuration. See Table 4-131 .
config_mode	No	String	Workload configuration mode. • ui • yaml • template

Parameter	Mandatory	Type	Description
enable_serment_injection	No	Boolean	Whether to enable Serment.
enable_serment_spring_register	No	Boolean	Whether Spring Cloud Huawei microservices need to be registered with registry centers through Serment.
enable_serment_dubbo_register	No	Boolean	Whether Dubbo microservices need to be registered with registry centers through Serment.
enable_serment_config	No	Boolean	Whether to enable the Serment dynamic configuration plug-in.
cell_id	No	String	Deployment cell ID.
swimlane_id	No	String	Lane ID.
sermentAgentInjectActionFlag	No	Boolean	Whether JDK 17 and JDK 21 automatically add the following JVM environment variables: JDK 17: --add-opens java.base/java.lang=ALL-UNNAMED --add-opens java.base/java.net=ALL-UNNAMED --add-opens java.base/java.math=ALL-UNNAMED --add-opens java.base/sun.net.www=ALL-UNNAMED --add-opens java.base/sun.net.www.protocol.http=ALL-UNNAMED JDK 21: --add-opens java.base/java.lang=ALL-UNNAMED --add-opens java.base/java.net=ALL-UNNAMED --add-opens java.base/java.math=ALL-UNNAMED --add-opens java.base/sun.net.www=ALL-UNNAMED --add-opens java.base/sun.net.www.protocol.http=ALL-UNNAMED -Dnet.bytebuddy.experimental=true
sys_config	No	Map<String, String>	Uses a compressed package to deploy component configuration items in the VM environment.
image_pull_secrets	No	Array of String	List of image pull secrets.
auto_lts_config	No	Object	Automatic LTS configuration. See Table 4-132 .
config_encryption	No	Object	This parameter is mandatory when the encryption key is configured for a single-container component. See Table 4-133 .

Table 4-96 labels

Parameter	Mandatory	Type	Description
key	Yes	String	Label name. The value contains 1 to 36 characters, including digits, letters, underscores (_), or hyphens (-). NOTICE The value of Key cannot be the built-in app, casid, or version of the system. Otherwise, an error will be reported during component creation and deployment.
value	Yes	String	Label value. The value contains 0 to 43 characters, including digits, letters, underscores (_), periods (.), or hyphens (-).

Table 4-97 storages

Parameter	Mandatory	Type	Description
type	Yes	String	Data storage type. • HostPath: host path for local disk mounting. • EmptyDir: temporary directory for local disk mounting. • ConfigMap: configuration item for local disk mounting. • Secret: secrets for local disk mounting. • PersistentVolumeClaim: cloud storage mounting.
name	Yes	String	Name of the storage disk.
parameters	Yes	Object	For details about the parameters of each data storage type, see Table 4-98 .
mounts	Yes	Array of objects	Mount path of data storage. See Table 4-99 .

Table 4-98 component_storage_parameters

Parameter	Mandatory	Type	Description
path	No	String	Host path. This parameter is applicable to the HostPath storage type. This parameter is mandatory when type is set to HostPath .
name	No	String	Name of a configuration item, secret, or PVC, which is applicable to ConfigMap, Secret, or PersistentVolumeClaim storage, respectively. This parameter is mandatory when type is set to ConfigMap , Secret , or PersistentVolumeClaim .
default_mode	No	Integer	Mounting permission. The value is in decimal format. Example: 384. This parameter is applicable to ConfigMap and secret storage.
medium	No	String	This parameter is applicable to EmptyDir storage. If it is not specified, the default disk medium is used. If it is set to memory , memory storage is enabled.

Table 4-99 mounts

Parameter	Mandatory	Type	Description
path	Yes	String	Application path to which the data storage is mounted. Example: /tmp.
sub_path	Yes	String	Subpath of the mount path.
read_only	Yes	Boolean	Read-only or not. • true: read-only. • false: read and write. You can modify data volumes in the application path. To prevent data loss, newly written data will not be migrated during application migration.

Table 4-100 deploy_strategy

Parameter	Mandatory	Type	Description
type	Yes	String	Deployment type. • OneBatchRelease: single-batch upgrade. • RollingRelease: rolling deployment and upgrade. • GrayRelease: dark launch upgrade.
rolling_release	No	Object	Rolling deployment parameter, which is mandatory when type is set to RollingRelease . For details, see Table 4-101 .
gray_release	No	Object	Dark launch upgrade parameter, which is mandatory when type is set to GrayRelease . For details, see Table 4-102 .

Table 4-101 rolling_release

Parameter	Mandatory	Type	Description
batches	Yes	Integer	Deployment batches.

Table 4-102 gray_release

Parameter	Mandatory	Type	Description
type	Yes	String	Dark launch policy. • WEIGHT: dark launch by traffic. • CONTENT: dark launch by content.
first_batch_weight	Yes	Integer	Proportion of first-batch dark launch traffic. This parameter is mandatory when type is set to WEIGHT .
first_batch_replica	Yes	Integer	Number of first-batch dark launch instances. This parameter is mandatory when type is set to WEIGHT .

Parameter	Mandatory	Type	Description
remaining_batch	Yes	Integer	Number of batches for remaining instance deployment. After the first batch is completed, the remaining instances will be deployed in specified batches. Example: If there are 5 instances remaining to be deployed in 3 batches, these 5 instances will be upgraded based on 2:2:1. This parameter is mandatory when type is set to WEIGHT .
deployment_mode	No	Integer	Deployment mode. 1: The upstream is a microservice gateway. 3: The upstream is a microservice. 4: The upstream is a load balancer.
replica_surge_mode	No	String	Mode of adding a dark launch instance. This parameter is mandatory when type is set to CONTENT . - MIRROR: blue-green. - EXTRA: canary (increase, then decrease) - NOSURGE: canary (decrease, then increase)
rule_match_mode	No	String	Effective mode of a dark launch rule. This parameter is mandatory when type is set to CONTENT . - ALL: All conditions are met. - ANY: Any condition is met.
rules	No	Object	Dark launch rule. See Table 4-103 . This parameter is mandatory when type is set to CONTENT .

Table 4-103 gray_rules

Parameter	Mandatory	Type	Description
type	Yes	String	Match type. Currently, only HEADER is supported.
key	Yes	String	Parameter name.
value	Yes	String	Condition value.

Parameter	Mandatory	Type	Description
condition	Yes	String	Condition type. • EQUAL • MATCH • IN: enumeration

Table 4-104 command

Parameter	Mandatory	Type	Description
command	No	Array of String	Command for controlling container running.
args	No	Array of String	Parameter for controlling container running. Example: -port=8080. If there are multiple parameters, separate them by line breaks.

Table 4-105 component_lifecycle

Parameter	Mandatory	Type	Description
type	Yes	String	Processing method. • http • command
scheme	No	String	HTTP request type, which can be HTTP or HTTPS. This parameter is valid only when type is set to http .
host	No	String	Custom IP address. If this parameter is left blank, the pod IP address is used. This parameter is valid only when type is set to http .
port	No	Integer	Port number. This parameter is valid only when type is set to http .
path	No	String	Request path. This parameter is valid only when type is set to http .
command	No	Array of String	Command list. This parameter is valid only when type is set to command .

Table 4-106 mesher

Parameter	Mandatory	Type	Description
port	Yes	Integer	Process listening port.

Table 4-107 tomcat_opt

Parameter	Mandatory	Type	Description
server_xml	Yes	String	1. Listening port in the server.xml file will take effect even if it has been configured in public network access. 2. The custom Tomcat configuration may conflict with the existing port. Check whether the custom port is occupied.

Table 4-108 host_aliases

Parameter	Mandatory	Type	Description
ip	Yes	String	IP address.
hostnames	Yes	Array of String	Host alias.

Table 4-109 dns_config

Parameter	Mandatory	Type	Description
nameservers	No	Array of String	IP address list of the DNS server of the pod.
searches	No	Array of String	(Optional) List of DNS search domains used to search for host names in the pod.
options	No	Array of objects	(Optional) List of objects. Each object may have a name (mandatory) and a value (optional). The contents of the objects here will be merged into the options generated from the specified DNS policy. Duplicate entries will be deleted. See Table 4-110 .

Table 4-110 options

Parameter	Mandatory	Type	Description
name	Yes	String	Operation name.
value	No	String	Operation value.

Table 4-111 security_context

Parameter	Mandatory	Type	Description
run_as_user	No	Integer	User for running containers. Example: To run containers as user root , set the user ID to 0 .
run_as_group	No	Integer	Owner group specified for running container processes.
capabilities	No	Object	Capability set. See Table 4-112 .

Table 4-112 capabilities

Parameter	Mandatory	Type	Description
add	No	Array of String	Grant the Linux permission to the running user.
drop	No	Array of String	Remove the Linux permission of the running user.

Table 4-113 logs

Parameter	Mandatory	Type	Description
log_path	Yes	String	Container log path.
rotate	Yes	String	Interval for dumping logs.
host_path	Yes	String	Mounted host path.

Parameter	Mandatory	Type	Description
host_extend_path	Yes	String	Host extension path. By extending the host path, you can distinguish mounting from different containers in the same host path. • None: the extended path is not used. • PodUID: extend the host path based on the pod ID. • PodName: extend the host path based on the pod name. • PodUID/ContainerName: extend the host path based on the pod ID and container name. • PodName/ContainerName: extend the host path based on the pod name and container name.

Table 4-114 custom_metric

Parameter	Mandatory	Type	Description
path	Yes	String	Collection path. Example: /metrics.
port	Yes	Integer	Collection port. Example: 9090.
dimensions	Yes	String	Monitoring dimension. Example: cpu_usage,mem_usage.

Table 4-115 component_affinity

Parameter	Mandatory	Type	Description
kind	No	String	Object kind. • node • pod
condition	No	String	Policy type. • required: The requirement must be met. • preferred: The requirement is met as much as possible.
match_expressions	No	Array of objects	Scheduling policy rule. See Table 4-116 .

Parameter	Mandatory	Type	Description
weight	No	String	Weight. - The value is 100 when condition is set to required. - Enter an integer ranging from 1 to 100 when condition is set to preferred.
topology_key	No	String	Topology key.

Table 4-116 match_expressions

Parameter	Mandatory	Type	Description
key	Yes	String	Key of the match tag.
operation	Yes	String	Match condition.
value	Yes	String	Value of the match tag.

Table 4-117 component_probe

Parameter	Mandatory	Type	Description
type	Yes	String	Type. Value: http, tcp, or command.
delay	Yes	Integer	Interval between the startup and detection.
timeout	Yes	Integer	Detection timeout interval.
scheme	No	String	Request type. Value: HTTP or HTTPS. This parameter is valid only when type is set to http .
host	No	String	Custom IP address. If this parameter is left blank, the pod IP address is used. This parameter is valid only when type is set to http .
port	No	Integer	Port number. This parameter is valid only when type is set to http or tcp .
path	No	String	Request path. This parameter is valid only when type is set to http .
command	No	Array of String	Command list. This parameter is valid only when type is set to command .

Table 4-118 refer_resources

Parameter	Mandatory	Type	Description
id	Yes	String	Resource ID.
type	Yes	String	Resource type. • vpc • eip • elb • cce • ecs • as • cse: ServiceComb engine • nacos: registry/configuration center • dcs • rds • gaussdb • dms_rocketmq • customServer • pvc • css_es: Elasticsearch
parameters	No	Object	Resource parameters. See Table 4-119 .

Table 4-119 refer_resource_parameter

Parameter	Mandatory	Type	Description
namespace	No	String	Namespace.
name	No	String	Resource name.
capacity	No	Integer	Storage capacity, in GiB.
type	No	String	Volume declaration type. • efs • obs • sfs • evs

Parameter	Mandatory	Type	Description
class	No	String	Volume declaration subtype. • csi-sfsturbo: fixed when type is set to efs. • csi-obs: fixed when type is set to obs. • csi-nas: fixed when type is set to sfs. • csi-disk: fixed when type is set to evs.
obs_volume_type	No	String	OBS volume type. Default value: STANDARD.
access_mode	No	String	Storage volume access mode. • ReadWriteOnce: A volume can be mounted as read-write by a single node. • ReadWriteMany: A volume can be mounted as read-write by many nodes.
availableZone	No	String	AZ.
volume_type	No	String	Volume type.

Table 4-120 envs

Parameter	Mandatory	Type	Description
name	Yes	String	Variable name.
value	No	String	Variable value.
value_from	No	Object	Variable reference. See Table 4-121 .

Table 4-121 value_from

Parameter	Mandatory	Type	Description
reference_type	Yes	String	Reference type. • configMapKey • secretKey
name	Yes	String	Name of the configuration item or secret.
key	No	String	Key of the configuration item or secret.
optional	No	Boolean	Whether the key of the configuration item or secret is mandatory.

Table 4-122 runtime_stack

Parameter	Mandatory	Type	Description
name	Yes	String	Technology stack name.
type	Yes	String	Technology stack type. Value: Java, Tomcat, Nodejs, Php, Docker, or Python. When deploy_mode is set to virtualmachine , only Java, Tomcat, and Node.js are supported. When deploy_mode is set to container , all of the preceding types are supported.
version	Yes	String	Technology stack version.
deploy_mode	Yes	String	Deployment mode. • container • virtualmachine

Table 4-123 source

Parameter	Mandatory	Type	Description
kind	Yes	String	Component source kind. • code • package • image
version	No	String	Version number.
url	No	String	Package address. This parameter is mandatory when kind is set to package or image .
storage	No	String	Storage mode. • devcloud: CodeArts release repository • obs • http: custom file address • swr This parameter is mandatory when kind is set to package or image .
http_username	No	String	Username. This parameter is mandatory when storage is set to http . Username and password authentication is supported.

Parameter	Mandatory	Type	Description
http_password	No	String	Password. This parameter is mandatory when storage is set to http . Username and password authentication is supported.
http_command	No	String	Address of the custom software package. This parameter is mandatory when storage is set to http .
http_headers	No	Map<String, String>	This parameter is mandatory when storage is set to http . You can customize request headers as key-value pairs, for example, {"key":"xx","value":"xx"}.
auth	No	String	Authentication mode. Value: iam or none. Default value: iam.
repo_auth	No	String	Authorization name, which can be obtained from the authorization list. For details, see Obtaining a Git Repository Authorization List .
repo_namespace	No	String	Namespace of the code repository.
repo_ref	No	String	Code branch or tag. Default value: master.
repo_type	No	String	Code repository type. Value: GitHub, BitBucket, or GitLab.
web_url	No	String	Code redirection URL. Example: https://github.com/example/demo.git.
repo_url	No	String	Code repository URL. Example: https://github.com/example/demo.git.
commit_id	No	String	In the code repository, commit ID is the unique identifier of each commit and is used to identify and reference a commit. The value contains 8 to 40 characters and starts with a lowercase letter or digit.

Table 4-124 build

Parameter	Mandatory	Type	Description
parameters	No	Map<String, Object>	This parameter is provided only when no ID is available during build creation. See Table 4-125 .

Table 4-125 parameters

Parameter	Mandatory	Type	Description
build_cmd	No	String	Compilation command. By default: 1. When build.sh exists in the root directory, the command is ./build.sh . 2. When build.sh does not exist in the root directory, the command varies depending on the OS. Example: • Java and Tomcat: mvn clean package • Node.js: npm build
dockerfile_path	No	String	Address of the Docker file. By default, the Docker file is in the root directory (.).
artifact_namespace	No	String	Build archive organization. Default value: cas_{project_id}.
cluster_id	Yes	String	ID of the cluster to be built.
node_label_selector	No	Map<String, String>	key: label key. value: label value.
environment_id	No	String	Environment ID. See Obtaining All Environments .

Table 4-126 external_accesses

Parameter	Mandatory	Type	Description
protocol	No	String	External access type. Value: http or https.
address	No	String	External access address.
forward_port	No	Integer	External access port.

Table 4-127 tolerations

Parameter	Mandatory	Type	Description
key	No	String	Taint key.

Parameter	Mandatory	Type	Description
operator	No	String	Operator. • Equal: matches the nodes with the specified taint key (mandatory) and value. If the taint value is left blank, all taints with the same key will be matched. • Exists: matches nodes with the specified taint key. In this case, the taint value cannot be specified. Not specifying any taint key means all taints are tolerated.
value	No	String	Taint value. This parameter is mandatory when operator is set to Equal .
effect	No	String	Taint policy. • NoScheduler: Only the NoSchedule taint is matched. • PreferNoSchedule: Only the PreferNoSchedule taint is matched. • NoExecute: Only the NoExecute taint is matched. • If this parameter is not set, all taint effects are matched.
toleration_seconds	No	Integer	Tolerance time window. This parameter can be set only when effect is set to NoExecute . Within the tolerance time window, pods still run on the node with taints. After the time expires, the pods will be evicted.

Table 4-128 multi_containers

Parameter	Mandatory	Type	Description
container_name	Yes	String	Container name.
source	Yes	Object	Source of the code or software package. See Table 4-123 . In multi-container mode, only images are supported.
limit_cpu	No	Number	Maximum CPU limit. Unit: core.
limit_memory	No	Number	Maximum memory size. Unit: GiB.

Parameter	Mandatory	Type	Description
request_cpu	No	Number	Requested CPU resources. Unit: core.
request_memory	No	Number	Requested memory. Unit: GiB.
envs	No	Array of objects	Environment variable list. See Table 4-120 .
liveness_probe	No	Object	Container liveness probe. See Table 4-117 .
readiness_probe	No	Object	Container service probe. See Table 4-117 .
startup_probe	No	Object	Container startup probe. See Table 4-117 .
command	No	Object	Startup command. See Table 4-104 .
post_start	No	Object	Post-start processing. See Table 4-105 .
pre_stop	No	Object	Pre-stop processing. See Table 4-105 .
logs	No	Array of objects	Log collection. See Table 4-113 .
storages	No	Array of objects	Storage. See Table 4-97 .
security_context	No	Object	Security policy. See Table 4-111 .
config_encryption	No	Object	This parameter is mandatory when the encryption key is configured for a multi-container component. See Table 4-133 .

Table 4-129 ports

Parameter	Mandatory	Type	Description
name	Yes	String	Port name.
port	Yes	String	Access port.
target_port	Yes	String	Container port.

Table 4-130 update_strategy

Parameter	Mandatory	Type	Description
type	No	String	Component upgrade policy. Options: • RollingUpdate (default): New pods are created gradually, and then old pods are deleted. • Recreate: Old pods are deleted, and then new pods are created.
max_unavailable	No	• String • Integer	Maximum number of unavailable pods. That is, the number or percentage of pods that can be deleted based on the value of replica . The default value is 25% . This parameter can be set when workload_kind is set to deployment . • String (%): the maximum percentage of unavailable pods to the total number of pods. • Integer: the maximum number of unavailable pods.
max_surge	No	• String • Integer	Maximum surge. That is, the number or percentage of pods allowed based on the value of replica . The default value is 25% . This parameter can be set only when workload_kind is set to deployment . • String (%) • Integer
termination_period_seconds	No	Integer	Scale-in time window. That is, the time window for executing commands before a workload stops. The default value is 30s.
min_ready_seconds	No	Integer	Minimum ready time of a pod. That is, the minimum ready time of a new pod without any container crash. A pod is considered available only after the minimum ready time is exceeded. The default value is 0 .

Parameter	Mandatory	Type	Description
progress_deadline_seconds	No	Integer	Maximum upgrade duration. That is, the time (in seconds) the system waits for before reporting a component upgrade failure. If this parameter is specified, the value of this parameter must be greater than that of min_ready_seconds .
revision_history_limit	No	Integer	Maximum number of pods to retain. That is, the number of old pods to retain to allow rollback. The default value is 10 .

Table 4-131 configs

Parameter	Mandatory	Type	Description
config_id	Yes	String	Unified configuration ID.
mount_path	Yes	String	Mount path.

Table 4-132 auto_lts_config

Parameter	Mandatory	Type	Description
enable	Yes	Boolean	Whether to enable automatic LTS configuration.
lts_log_path	No	Array of String	Access configuration log path.

Table 4-133 config_encryption

Parameter	Mandatory	Type	Description
key_id	Yes	String	User key ID in DEW.

Response

Table 4-134 Response parameters

Parameter	Type	Description
job_id	String	Job ID.
component_id	String	Application component ID.

Example Request

- Scenario 1: Create a component in the CCE container environment

Create a component named **mycomponent**. The technology stack is Java 1.2.0. The component source is the **weather-1.0.0.jar** package stored in the **bucket00001** bucket of OBS. The resource used for deployment is the CCE whose ID is **fd92bfab-73b3-11ee-90f0-0255ac1001b3**.

```
{
  "name": "mycomponent",
  "description": "",
  "labels": [
    {
      "key": "com-key",
      "value": "com-value"
    }
  ],
  "version": "2023.1102.17105",
  "environment_id": "b489f7b4-2539-45a1-ba71-4728961434ca",
  "runtime_stack": {
    "name": "OpenJDK8",
    "version": "1.2.0",
    "type": "Java",
    "deploy_mode": "container"
  },
  "source": {
    "kind": "package",
    "url": "obs://bucket0001/weather-1.0.0.jar",
    "version": "",
    "storage": "obs"
  },
  "tomcat_opts": {
    "server_xml": ""
  },
  "refer_resources": [
    {
      "id": "fd92bfab-73b3-11ee-90f0-0255ac1001b3",
      "type": "cce",
    }
  ],
  "replica": 1,
  "limit_cpu": 0.25,
  "limit_memory": 0.5,
  "request_cpu": 0.25,
  "request_memory": 0.5
}
```

- Scenario 2: Create a component in the ECS VM environment

Create a component named **mycomponent**. The technology stack is Java 1.2.0. The component source is the **weather-1.0.0.jar** package stored in the **bucket00001** bucket of OBS. The resource used for deployment is the ECS whose ID is **67835bb3-1235-4cc9-be71-becbb2b4ca0d**.

```
{
  "name": "mycomponent",
  "description": "",
  "labels": [
    {
      "key": "com-key",
      "value": "com-value"
    }
  ],
  "version": "2023.1102.17540",
  "environment_id": "9a075df5-104f-4e21-9dbf-ffc3572effdf",
  "runtime_stack": {
    "name": "OpenJDK8",
    "version": "1.2.0",
    "type": "Java",
    "deploy_mode": "virtualmachine"
  },
  "source": {
    "kind": "package",
    "url": "obs://bucket0001/weather-1.0.0.jar",
    "version": "",
    "storage": "obs"
  },
  "tomcat_opts": {
    "server_xml": ""
  },
  "refer_resources": [
    {
      "id": "23598aa9-7a6b-43c7-a53a-39ca5e7290fb",
      "type": "ecs"
    }
  ],
  "replica": 1,
  "external_accesses": [
    {
      "protocol": "http",
      "address": "192.168.0.169",
      "forward_port": 80
    }
  ]
}
```

Example Response

```
{
  "job_id": "JOB8be53b2a-af9a-4e2b-8505-096688a52237",
  "component_id": "b8702b0f-94d3-4822-98a1-56815632a0a0"
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.3.2 Importing a CCE Workload to an Application to Create a Component

Function

In CCE, a workload is a set of pods that work together in a certain mode to complete a specific application or service.

This API is used to import a workload created on CCE to an application for creating a component.

URI

POST /v3/{project_id}/cas/applications/{application_id}/components

Table 4-135 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
application_id	Yes	String	Application ID. See Obtaining All Applications .

Request

Table 4-136 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Table 4-137 Request body parameter

Parameter	Mandatory	Type	Description
create_mode	Yes	String	Component creation mode. To import a workload created on CCE for creating components, set this parameter to import .
name	Yes	String	Application component name. The value contains 2 to 64 characters consisting of letters, digits, hyphens (-), and underscores (_). It starts with a letter and ends with a letter or digit.
workload_name	Yes	String	Name of the workload created on CCE.
labels	No	Array of objects	Label. For container-based deployment, you can query workloads by label. See Table 4-138 .
runtime_stack	Yes	Object	Runtime. Fixed value: Docker.
environment_id	Yes	String	Environment ID.
description	No	String	Description. The value can contain up to 128 characters.
version	Yes	String	Component version number. The value can contain up to 32 characters. It needs to meet the following rule: $\wedge([0-9]+)(.[0-9]+){2,3}\$$
workload_kind	Yes	String	Workload type. • deployment • statefulset
refer_resources	Yes	Array of objects	Associated resource. See Table 4-139 . If the imported workload requires microservice dark launch, association with the ServiceComb or Nacos engine is needed.
external_accesses	No	Array of Objects	External access. See Table 4-141 .

Table 4-138 labels

Parameter	Mandatory	Type	Description
key	Yes	String	Label name.
value	Yes	String	Label value.

Table 4-139 refer_resources

Parameter	Mandatory	Type	Description
id	Yes	String	Resource ID.
type	Yes	String	Resource type. • vpc • eip • elb • cce • dcs • rds • cse: ServiceComb engine • nacos: registry/configuration center • gaussdb • dms_rocketmq • css_es: Elasticsearch
parameters	No	Object	Resource parameters. See Table 4-140 .

Table 4-140 refer_resource_parameter

Parameter	Mandatory	Type	Description
namespace	No	String	Namespace.

Table 4-141 external_accesses

Parameter	Mandatory	Type	Description
protocol	No	String	External access type. Value: http or https.
address	No	String	External access address.

Parameter	Mandatory	Type	Description
forward_port	No	Integer	External access port.

Response

Table 4-142 Response parameters

Parameter	Type	Description
instance_id	String	Component instance ID.

Example Request

Create a stateless component named **testcyf-import-deployment** by importing the **test-cyf-deployment** workload from the CCE resource whose ID is **1d23eab7-da00-11ee-a1e1-0255ac1001ba**.

```
{
  "name": "testcyf-import-deployment",
  "description": "",
  "labels": [],
  "version": "2024.0506.16306",
  "environment_id": "9cff7557-4ebb-481b-bc1e-6a6573b5b8d2",
  "workload_name": "test-cyf-deployment",
  "workload_kind": "deployment",
  "runtime_stack": {
    "name": "Docker",
    "version": "",
    "type": "Docker",
    "deploy_mode": "container"
  },
  "refer_resources": [{
    "id": "1d23eab7-da00-11ee-a1e1-0255ac1001ba",
    "type": "cce",
    "parameters": {
      "namespace": "default"
    }
  }],
  "create_mode": "import"
}
```

Example Response

```
{
  "instance_id": "7897df44-e9c3-486f-9406-7a55e9aa55e2"
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.3.3 Obtaining All Components of an Application

Function

This API is used to obtain all components of an application.

URI

GET /v3/{project_id}/cas/applications/{application_id}/components

Table 4-143 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
application_id	Yes	String	Application ID. See Obtaining All Applications .

Table 4-144 Query parameters

Parameter	Mandatory	Type	Description
limit	No	integer	Number of records to be queried. Value range: 0–100.
offset	No	integer	Query offset, which indicates the result after the Nth data record is queried.

Parameter	Mandatory	Type	Description
order_by	No	String	Sorting field. Numerated values: • create_time (default) • name • update_time If the transferred value is not within the supported enumerated value range, the default sorting field is used.
order	No	String	Sorting order. • desc (default) • asc

Request

Table 4-145 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

Table 4-146 Response parameters

Parameter	Type	Description
count	Integer	Total number of components.
components	Array of objects	Component parameters. See Table 4-147 .

Table 4-147 components

Parameter	Type	Description
id	String	Component ID.
name	String	Component name.
labels	Array of objects	Label. For container-deployed components, you can query a workload by label. For VM-deployed components, you can configure a label of the corresponding microservice name to use graceful startup and shutdown. See Table 4-152 .
status	Object	Component status. See Table 4-150 .
runtime_stack	Object	Technology stack. See Table 4-149 .
application_name	String	Application name.
application_id	String	Application ID.
environment_name	String	Environment name.
environment_id	String	Environment ID.
version	String	Component version.
source	Object	Source of the code or software package. See Table 4-148 .
creator	String	Component creator.
platform_type	String	Type of the platform where an application runs. Values: • cce • vmapp
external_accesses	Array of objects	External access. See Table 4-153 .
workload_kind	String	Workload type. • deployment • statefulset • daemonset
is_multi_containers	Boolean	Whether multi-container is set.
enterprise_project_id	String	Enterprise project ID.
engine_id	String	Engine ID.

Parameter	Type	Description
swimlane_tag	String	Lane tag.
swimlane	Object	Lane. See Table 4-154 .
refer_resources	Array of Objects	Associated resource. See Table 4-156 .
config_mode	String	Component creation mode. • yaml • ui • template
template_source	Object	Source of the component template package. See Table 4-162 .

Table 4-148 source

Parameter	Type	Description
kind	String	Component source kind. • code • package • image
version	String	Version number.
url	String	Package address. This parameter is mandatory when kind is set to package or image .
storage	String	Storage mode. • devcloud: CodeArts release repository • obs • http: custom file address • swr This parameter is mandatory when kind is set to package or image .
auth	String	Authentication mode. Value: iam or none. Default value: iam.
repo_auth	String	Authorization name, which can be obtained from the authorization list.
repo_namespace	String	Namespace of the code repository.
repo_ref	String	Code branch or tag. Default value: master.

Parameter	Type	Description
repo_type	String	Code repository type. Value: GitHub, BitBucket, or GitLab.
web_url	String	Code redirection URL. Example: https://github.com/example/demo.git.
repo_url	String	Code repository URL. Example: https://github.com/example/demo.git.
http_username	String	Username. This parameter is mandatory when storage is set to http . Username and password authentication is supported.
http_password	String	Password. This parameter is mandatory when storage is set to http . Username and password authentication is supported.
http_command	String	Address of the custom software package. This parameter is mandatory when storage is set to http .
http_headers	Map<String, String>	This parameter is mandatory when storage is set to http . You can customize request headers as key-value pairs, for example, {"key": "xx", "value": "xx"}.

Table 4-149 runtime_stack

Parameter	Type	Description
name	String	Technology stack name. Example: OpenJDK8.
type	String	Technology stack type. Value: Java, Tomcat, Nodejs, Php, Docker, or Python. When deploy_mode is set to virtualmachine , only Java, Tomcat, and Node.js are supported. When deploy_mode is set to container , all of the preceding types are supported.
version	String	Technology stack version. Example: 1.1.4.
deploy_mode	String	Deployment mode. • container • virtualmachine

Table 4-150 status

Parameter	Type	Description
component_status	String	Component status. - INITIALIZING - UPGRADING - FAILED - RUNNING - DOWN - DELETING - DELETED - RESERVED - STARTING - STOPPING - STOPPED - RESTARTING - PENDING - UNKNOWN - PARTIALLY_FAILED
available_replica	Integer	Available instances.
replica	Integer	Total instances.
fail_detail	String	Failure cause. - cluster_deleted - cluster_unavailable - cluster_inaccessible - namespace_deleted - namespace_unavailable - namespace_inaccessible - resource_deleted - create_failed - delete_failed
last_job_id	String	ID of the last executed job.
create_time	Integer	Creation time.
update_time	Integer	Update time.
creator	String	Creator.
artifact	Object	Artifact information. See Table 4-151 .

Table 4-151 artifact

Parameter	Type	Description
type	String	Kind of the component source package. • package • image
url	String	Software package or image address.

Table 4-152 labels

Parameter	Type	Description
key	String	Label name.
value	String	Label value.

Table 4-153 external_accesses

Parameter	Type	Description
protocol	String	External access type. Value: http or https.
address	String	External access address.
forward_port	Integer	External access port.

Table 4-154 swimlane

Parameter	Type	Description
id	String	Lane ID.
name	String	Lane name.
swimlane_group_id	String	Lane group ID.
type	String	Lane type. • BASE: baseline lane. • GRAY: dark launch lane.
tag	String	Lane tag, which is used to control traffic routing.
route_status	String	Route status. • ENABLE • DISABLED

Parameter	Type	Description
rule_match_mode	String	Mode of matching routing rules. This parameter is valid when gray_release_type is set to CONTENT . - ALL: matches all rules. - ANY: matches any rule.
rules	Array of objects	Routing rules. This parameter is valid when gray_release_type is set to CONTENT . See Table 4-155 .
weight	Integer	Lane traffic weight. The value ranges from 0 to 100 . This parameter is valid when gray_release_type is set to WEIGHT .
project_id	String	Project ID.
create_time	Integer	Creation time.
update_time	Integer	Update time.
creator	String	Creator.
swimlane_group	Object	Lane group. See Table 4-159 .
engine_ids	Array of String	Engine ID list.

Table 4-155 rules

Parameter	Type	Description
type	String	Matching type of dark launch policies. Currently, only HEADER is supported, that is, matching by header.
key	String	Key.
value	String	Value.
condition	String	Match condition. - EXACT - PREFIX - REGEX

Table 4-156 refer_resources

Parameter	Type	Description
id	String	Resource ID.

Parameter	Type	Description
instance_id	String	Component instance ID.
type	String	Resource type. • vpc • eip • elb • cce • ecs • customServer • dcs • rds • cse: ServiceComb engine. • nacos: registry/configuration center. • pvc • gaussdb • dms_rocketmq • css_es: Elasticsearch
refer_type	String	Resource type. • platform: deployment type. • service: service type.
refer_alias	String	Resource alias.
parameters	Object	Resource parameters. See Table 4-157 .
selectors	Objects	Resource selector. See Table 4-158 .

Table 4-157 refer_resource_parameter

Parameter	Type	Description
namespace	String	Namespace.
name	String	Resource name.
capacity	Integer	Storage capacity, in GiB.
type	String	Volume declaration type. This parameter is specified when type is set to pvc in Table 4-156 . • efs • obs • sfs • evs

Parameter	Type	Description
class	String	Volume declaration subtype. • csi-sfsturbo: fixed when type is set to efs. • csi-obs: fixed when type is set to obs. • csi-nas: fixed when type is set to sfs. • csi-disk: fixed when type is set to evs.
pv_name	String	Volume name. This parameter is displayed when type is set to efs .
share_id	String	Volume handle. This parameter is displayed when type is set to efs .
share_export_location	String	Shared path of the volume. This parameter is displayed when type is set to efs .
share_proto	String	File system type of the volume. This parameter is displayed when type is set to efs . Default value: NFS.
obs_volume_type	String	OBS volume type. Default value: STANDARD.
access_mode	String	Storage volume access mode. • ReadWriteOnce: A volume can be mounted as read-write by a single node. • ReadWriteMany: A volume can be mounted as read-write by many nodes.
availableZone	String	AZ.
volume_type	String	Volume type.

Table 4-158 selectors

Parameter	Type	Description
name	String	Resource selector name.
flavor	String	Resource selector flavor.
labels	Map<String, String>	Label.

Table 4-159 swimlane_group

Parameter	Type	Description
id	String	Lane group ID.
name	String	Lane group name.
description	String	Lane group description.
creator	String	Creator.
project_id	String	Project ID.
gray_release_type	String	Dark launch type of the lane group. - WEIGHT: full-link dark launch based on the traffic ratio. - CONTENT: full-link dark launch based on content.
engine_ids	Array of string	Engine IDs of the gateway's target services.
gateways	Array of objects	List of target services. See Table 4-160 .
create_time	Integer	Creation time.
update_time	Integer	Update time.

Table 4-160 gateways

Parameter	Type	Description
id	String	Gateway ID.
name	String	Gateway name.
engine_id	String	ID of the engine used by the target services.
target_services	Array of objects	List of target services in the gateway service management. See Table 4-161 .

Table 4-161 target_services

Parameter	Type	Description
id	String	Target service ID.
name	String	Target service name.
host	String	Host information of the target service.

Table 4-162 template_source

Parameter	Type	Description
storage	String	Storage mode of the component template package. • devcloud: CodeArts release repository • obs • http: custom file address
url	String	Address of the component template package. This parameter is valid only when the component template package storage mode is http .
http_username	String	Username. This parameter is valid only when the component template package storage mode is http and the security authentication mode is username and password authentication.
http_password	String	Password. This parameter is valid only when the component template package storage mode is http and the security authentication mode is username and password authentication.
http_headers	Array of objects	Request header. See Table 4-163 . This parameter is valid only when the component template package storage mode is http and the security authentication mode is custom header authentication.

Table 4-163 http_headers

Parameter	Type	Description
key	String	Request key.
value	String	Request value.

Example Request

None

Example Response

```
{
  "components": [
    {
      "source": {
```

```
{
  "kind": "package",
  "url": "obs://bucket0001/weather-1.0.0.jar",
  "storage": "obs"
},
{
  "name": "mycomponent",
  "id": "b8702b0f-94d3-4822-98a1-56815632a0a0",
  "labels": null,
  "runtime_stack": {
    "name": "OpenJDK8",
    "deploy_mode": "virtualmachine",
    "version": "1.1.1"
  },
  "external_accesses": null,
  "status": {
    "component_status": "RUNNING",
    "available_replica": 1,
    "replica": 1,
    "fail_detail": null,
    "last_job_id": "JOB8be53b2a-af9a-4e2b-8505-096688a52237",
    "creator": "ss-test",
    "create_time": 1679556221028,
    "update_time": 1679556318779
  },
  "environment_name": "sudo",
  "environment_id": "4d084044-0b80-4641-963c-b9c9f4092a4f",
  "application_name": "app-39mbbn",
  "application_id": "fc092465-a5fb-4a52-bc65-b735f18366d8",
  "creator": "ss-test",
  "platform_type": "cce",
  "version": "2023.0323.15181"
}
],
"count": 1
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.3.4 Obtaining All Components

Function

This API is used to obtain all application components.

URI

GET /v3/{project_id}/cas/components

Table 4-164 Path parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .

Table 4-165 Query parameters

Parameter	Mandatory	Type	Description
limit	No	integer	Number of records to be queried. Value range: 0–100.
offset	No	integer	Query offset, which indicates the result after the Nth data record is queried.
order_by	No	String	Sorting field. Numerated values: • create_time (default) • name • update_time If the transferred value is not within the supported enumerated value range, the default sorting field is used.
order	No	String	Sorting order. • desc (default) • asc
application_name	No	String	Application name.
component_name	No	String	Component name.

Request

Table 4-166 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

Table 4-167 Response parameters

Parameter	Type	Description
count	Integer	Total number of components.
components	Array of objects	Component parameters. See Table 4-168 .

Table 4-168 components

Parameter	Type	Description
id	String	Component ID.
name	String	Component name.
labels	Array of objects	Label. For container-deployed components, you can query a workload by label. For VM-deployed components, you can configure a label of the corresponding microservice name to use graceful startup and shutdown. See Table 4-173 .
status	Object	Component status. See Table 4-171 .
runtime_stack	Object	Technology stack. See Table 4-170 .
application_name	String	Application name.
application_id	String	Application ID.
environment_name	String	Environment name.
environment_id	String	Environment ID.
version	String	Version number.

Parameter	Type	Description
source	Object	Source of the code or software package. See Table 4-169 .
creator	String	Creator.
platform_type	String	Type of the platform where an application runs. Values: • cce • vmapp
external_accesses	Array of Objects	External access. See Table 4-174 .
workload_kind	String	Workload type. • deployment • statefulset • daemonset
is_multi_containers	Boolean	Whether multi-container is set.
enterprise_project_id	String	Enterprise project ID.
engine_id	String	Engine ID.
swimlane_tag	String	Lane tag.
swimlane	Object	Lane. See Table 4-175 .
refer_resources	Array of Objects	Associated resource. See Table 4-177 .
config_mode	String	Component creation mode. • yaml • ui • template
template_source	Object	Source of the component template package. See Table 4-183 .

Table 4-169 source

Parameter	Type	Description
kind	String	Component source kind. • code • package • image

Parameter	Type	Description
version	String	Version number.
url	String	Package address. This parameter is mandatory when kind is set to package or image .
storage	String	Storage mode, for example, obs.
auth	String	Authentication mode, for example, iam.
repo_auth	String	Authorization name.
repo_namespace	String	Namespace of the code repository.
repo_ref	String	Code branch or tag. Example: master.
repo_type	String	Code repository type. Value: GitHub, BitBucket, or GitLab.
web_url	String	Code redirection URL. Example: https://github.com/example/demo.git.
repo_url	String	Code repository URL. Example: https://github.com/example/demo.git.
http_username	String	Username. This parameter is mandatory when storage is set to http . Username and password authentication is supported.
http_password	String	Password. This parameter is mandatory when storage is set to http . Username and password authentication is supported.
http_command	String	Address of the custom software package. This parameter is mandatory when storage is set to http .
http_headers	Map<String, String>	This parameter is mandatory when storage is set to http . You can customize request headers as key-value pairs, for example, {"key": "xx", "value": "xx"}.

Table 4-170 runtime_stack

Parameter	Type	Description
name	String	Technology stack name.

Parameter	Type	Description
type	String	Technology stack type. Value: Java, Tomcat, Nodejs, Php, Docker, or Python. When deploy_mode is set to virtualmachine , only Java, Tomcat, and Node.js are supported. When deploy_mode is set to container , all of the preceding types are supported.
version	String	Technology stack version. Example: 1.1.4.
deploy_mode	String	Deployment mode. • container • virtualmachine

Table 4-171 status parameters

Parameter	Type	Description
component_status	String	Component status. • INITIALIZING • UPGRADING • FAILED • RUNNING • DOWN • DELETING • DELETED • RESERVED • STARTING • STOPPING • STOPPED • RESTARTING • PENDING • UNKNOWN • PARTIALLY_FAILED
available_replica	Integer	Available instances.
replica	Integer	Total instances.

Parameter	Type	Description
fail_detail	String	Failure cause. • cluster_deleted • cluster_unavailable • cluster_inaccessible • namespace_deleted • namespace_unavailable • namespace_inaccessible • resource_deleted • create_failed • delete_failed
last_job_id	String	ID of the last executed job.
create_time	Integer	Creation time.
update_time	Integer	Update time.
creator	String	Creator.
artifact	Object	Artifact information. See Table 4-172 .

Table 4-172 artifact

Parameter	Type	Description
type	String	Component source type. • package • image
url	String	Software package or image address.

Table 4-173 labels

Parameter	Type	Description
key	String	Label name.
value	String	Label value.

Table 4-174 external_accesses

Parameter	Type	Description
protocol	String	External access type. Value: http or https.

Parameter	Type	Description
address	String	External access address.
forward_port	Integer	External access port.

Table 4-175 swimlane

Parameter	Type	Description
id	String	Lane ID.
name	String	Lane name.
swimlane_group_id	String	Lane group ID.
type	String	Lane type. • BASE: baseline lane. • GRAY: dark launch lane.
tag	String	Lane tag, which is used to control traffic routing.
route_status	String	Route status. • ENABLE • DISABLED
rule_match_mode	String	Mode of matching routing rules. This parameter is valid when gray_release_type is set to CONTENT . • ALL: matches all rules. • ANY: matches any rule.
rules	Array of objects	Routing rules. This parameter is valid when gray_release_type is set to CONTENT . See Table 4-176 .
weight	Integer	Lane traffic weight. The value ranges from 0 to 100 . This parameter is valid when gray_release_type is set to WEIGHT .
project_id	String	Project ID.
create_time	Integer	Creation time.
update_time	Integer	Update time.
creator	String	Creator.
swimlane_group	Object	Lane group. See Table 4-180 .
engine_ids	Array of String	Engine ID list.

Table 4-176 rules

Parameter	Type	Description
type	String	Type of dark launch matching rule. Currently, only HEADER is supported, that is, matching by header.
key	String	Key.
value	String	Value.
condition	String	Match condition. • EXACT • PREFIX • REGEX

Table 4-177 refer_resources

Parameter	Type	Description
id	String	Resource ID.
instance_id	String	Component instance ID.
type	String	Resource type. • vpc • eip • elb • cce • ecs • customServer • dcs • rds • cse: ServiceComb engine. • nacos: registry/configuration center. • pvc • gaussdb • dms_rocketmq • css_es: Elasticsearch
refer_type	String	Resource type. • platform: deployment type. • service: service type.
refer_alias	String	Resource alias.
parameters	Object	Resource parameters. See Table 4-178 .

Parameter	Type	Description
selectors	Objects	Resource selector. See Table 4-179 .

Table 4-178 refer_resource_parameter

Parameter	Type	Description
namespace	String	Namespace.
name	String	Resource name.
capacity	Integer	Storage capacity, in GiB.
type	String	Volume declaration type. This parameter is specified when type is set to pvc in Table 4-177 . • efs • obs • sfs • evs
class	String	Volume declaration subtype. • csi-sfsturbo: fixed when type is set to efs. • csi-obs: fixed when type is set to obs. • csi-nas: fixed when type is set to sfs. • csi-disk: fixed when type is set to evs.
pv_name	String	Volume name. This parameter is displayed when type is set to efs .
share_id	String	Volume handle. This parameter is displayed when type is set to efs .
share_export_location	String	Shared path of the volume. This parameter is displayed when type is set to efs .
share_proto	String	File system type of the volume. This parameter is displayed when type is set to efs . Default value: NFS.
obs_volume_type	String	OBS volume type. Default value: STANDARD.
access_mode	String	Storage volume access mode. • ReadWriteOnce: A volume can be mounted as read-write by a single node. • ReadWriteMany: A volume can be mounted as read-write by many nodes.
availableZone	String	AZ.

Parameter	Type	Description
volume_type	String	Volume type.

Table 4-179 selectors

Parameter	Type	Description
name	String	Resource selector name.
flavor	String	Resource selector flavor.
labels	Map<String, String>	Label.

Table 4-180 swimlane_group

Parameter	Type	Description
id	String	Lane group ID.
name	String	Lane group name.
description	String	Lane group description.
creator	String	Creator.
project_id	String	Project ID.
gray_release_type	String	Dark launch type of the lane group. - WEIGHT: full-link dark launch based on the traffic ratio. - CONTENT: full-link dark launch based on content.
engine_ids	Array of string	Engine IDs of the gateway's target services.
gateways	Array of objects	List of target services. See Table 4-181 .
create_time	Integer	Creation time.
update_time	Integer	Update time.

Table 4-181 gateways

Parameter	Type	Description
id	String	Gateway ID.
name	String	Gateway name.
engine_id	String	ID of the engine used by the target services.
target_services	Array of objects	List of target services in the gateway service management. See Table 4-182 .

Table 4-182 target_services

Parameter	Type	Description
id	String	Target service ID.
name	String	Target service name.
host	String	Host information of the target service.

Table 4-183 template_source

Parameter	Type	Description
storage	String	Storage mode of the component template package. • devcloud: CodeArts release repository • obs • http: custom file address
url	String	Address of the component template package. This parameter is valid only when the component template package storage mode is http .
http_username	String	Username. This parameter is valid only when the component template package storage mode is http and the security authentication mode is username and password authentication.
http_password	String	Password. This parameter is valid only when the component template package storage mode is http and the security authentication mode is username and password authentication.

Parameter	Type	Description
http_headers	Array of objects	Request header. See Table 4-184 . This parameter is valid only when the component template package storage mode is http and the security authentication mode is custom header authentication.

Table 4-184 http_headers

Parameter	Type	Description
key	String	Request key.
value	String	Request value.

Example Request

None

Example Response

```
{
  "components": [
    {
      "source": {
        "kind": "package",
        "url": "obs://bucket0001/weather-1.0.0.jar",
        "storage": "obs"
      },
      "name": "mycomponent",
      "id": "b8702b0f-94d3-4822-98a1-56815632a0a0",
      "labels": null,
      "runtime_stack": {
        "name": "OpenJDK8",
        "deploy_mode": "virtualmachine",
        "version": "1.1.1"
      },
      "external_accesses": null,
      "status": {
        "component_status": "RUNNING",
        "available_replica": 1,
        "replica": 1,
        "fail_detail": null,
        "last_job_id": "JOB8be53b2a-af9a-4e2b-8505-096688a52237",
        "creator": "ss-test",
        "create_time": 1679556221028,
        "update_time": 1679556318779
      },
      "environment_name": "sudo",
      "environment_id": "4d084044-0b80-4641-963c-b9c9f4092a4f",
      "application_name": "app-39mbbn",
      "application_id": "fc092465-a5fb-4a52-bc65-b735f18366d8",
      "creator": "ss-test",
      "platform_type": "cce",
      "version": "2023.0323.15181"
    }
  ],
}
```

```
"count": 1  
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.3.5 Modifying Component Information Based on the Component ID

Function

This API is used to modify component information based on the component ID.

URI

PUT /v3/{project_id}/cas/applications/{application_id}/components/
{component_id}

Table 4-185 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
application_id	Yes	String	Application ID. See Obtaining All Applications .
component_id	Yes	String	Component ID. See Obtaining All Components of an Application .

Request

Table 4-186 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Table 4-187 Request body parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Application component name. This value cannot be modified. The value contains 2 to 64 characters consisting of letters, digits, hyphens (-), and underscores (_). It starts with a letter and ends with a letter or digit.
labels	No	Array of objects	Label. For container-deployed components, you can query a workload by label. For VM-deployed components, you can configure a label of the corresponding microservice name to use graceful startup and shutdown. See Table 4-188 .
runtime_stack	Yes	Object	Runtime. This value cannot be modified. See Table 4-214 . The value can be obtained from the response of the API for Querying a Technology Stack .
description	No	String	Description. The value can contain up to 128 characters.
source	Yes	Object	Source of the code or software package. See Table 4-215 .
build	No	Object	Component build. See Table 4-216 .

Parameter	Mandatory	Type	Description
limit_cpu	No	Number	Maximum CPU limit. Unit: core.
limit_memory	No	Number	Maximum memory size. Unit: GiB.
request_cpu	No	Number	Requested CPU resources. Unit: core.
request_memory	No	Number	Requested memory. Unit: GiB.
version	Yes	String	Component version number. The value can contain up to 32 characters. It needs to meet the following rule: $^([0-9]+)(.[0-9]+){2,3}$$
envs	No	Array of objects	Environment variable list. See Table 4-212 .
replica	Yes	Integer	Number of instances. This parameter can be modified only when the scaling function is used.
storages	No	Array of objects	Storage. See Table 4-189 .
deploy_strategy	No	Object	Component deployment. See Table 4-192 .
command	No	Object	Startup command. See Table 4-196 .
post_start	No	Object	Post-start processing. See Table 4-197 .
pre_stop	No	Object	Pre-stop processing. See Table 4-197 .
mesher	No	Object	Multi-language access service mesh. This parameter is mandatory when Node.js is selected for the technology stack and the CSE engine is bound. See Table 4-198 .
timezone	No	String	Time zone where the component runs. Example: Asia/Shanghai.

Parameter	Mandatory	Type	Description
jvm_opts	No	String	JVM parameter. Example: -Xms256m -Xmx1024m. If there are multiple parameters, separate them by spaces. If the parameter is left blank, the default value is used.
tomcat_opts	No	Object	Tomcat parameter. This parameter is valid when you use the Tomcat technology stack. See Table 4-199 .
host_aliases	No	Array of objects	Host alias. See Table 4-200 .
dns_policy	No	String	DNS policy. • Default: inherits the domain name resolution configuration of the node where the pod is located. • ClusterFirst: adds the domain name resolution configuration. • ClusterFirstWithHostNet: takes effect only for pods running in hostNetwork mode. • None: replaces the domain name resolution policy.
dns_config	No	Object	DNS configuration. See Table 4-201 .
workload_kind	No	String	Workload type. • deployment • statefulset • daemonset
security_context	No	Object	Security policy. See Table 4-203 .
logs	No	Array of objects	Log collection. See Table 4-205 .
custom_metric	No	Object	Custom metrics. See Table 4-206 .
affinity	No	Array of objects	Affinity. See Table 4-207 .

Parameter	Mandatory	Type	Description
anti_affinity	No	Array of objects	Anti-affinity. See Table 4-207 .
liveness_probe	No	Object	Component liveness probe. See Table 4-209 .
readiness_probe	No	Object	Component service probe. See Table 4-209 .
startup_probe	No	Object	Component startup probe. See Table 4-209 .
refer_resources	No	Array of objects	Associated resource. See Table 4-210 .
external_accesses	No	Array of objects	External access. See Table 4-218 .
tolerations	No	Array of Objects	Tolerance policy. See Table 4-220 .
container_name	No	String	Container name.
update_strategy	No	Object	Component upgrade strategy. See Table 4-221 .
multi_containers	No	Array of Objects	Multi-container configuration. See Table 4-222 .
workload_content	No	String	Workload content.
configs	No	Array of Objects	Unified component configuration. See Table 4-223 .

Parameter	Mandatory	Type	Description
config_mode	No	String	Workload configuration mode. • ui • yaml • template
enable_serment_injection	No	Boolean	Whether to enable Serment.
enable_serment_spring_register	No	Boolean	Whether Spring Cloud Huawei microservices need to be registered with registry centers through Serment.
enable_serment_dubbo_register	No	Boolean	Whether Dubbo microservices need to be registered with registry centers through Serment.
enable_serment_config	No	Boolean	Whether to enable the Serment dynamic configuration plug-in.
cell_id	No	String	Deployment cell ID.
swimlane_id	No	String	Lane ID.
sermentAgentInjectActionFlag	No	Boolean	Whether JDK 17 and JDK 21 automatically add the following JVM environment variables: JDK 17: --add-opens java.base/java.lang=ALL-UNNAMED --add-opens java.base/java.net=ALL-UNNAMED --add-opens java.base/java.math=ALL-UNNAMED --add-opens java.base/sun.net.www=ALL-UNNAMED --add-opens java.base/sun.net.www.protocol.http=ALL-UNNAMED JDK 21: --add-opens java.base/java.lang=ALL-UNNAMED --add-opens java.base/java.net=ALL-UNNAMED --add-opens java.base/java.math=ALL-UNNAMED --add-opens java.base/sun.net.www=ALL-UNNAMED --add-opens java.base/sun.net.www.protocol.http=ALL-UNNAMED -Dnet.bytebuddy.experimental=true
sys_config	No	Map<String, String>	Uses a compressed package to deploy component configuration items in the VM environment.
image_pull_secrets	No	Array of String	List of image pull secrets.
template_source	No	Object	Source of the component template package. See Table 4-224 .

Parameter	Mandatory	Type	Description
input_values	No	Map<String, String>	Template package input information.
auto_lts_config	No	Object	Automatic LTS configuration. See Table 4-219 .

Table 4-188 labels

Parameter	Mandatory	Type	Description
key	Yes	String	Label name.
value	Yes	String	Label value.

Table 4-189 storages

Parameter	Mandatory	Type	Description
type	Yes	String	Data storage type. • HostPath: host path for local disk mounting. • EmptyDir: temporary directory for local disk mounting. • ConfigMap: configuration item for local disk mounting. • Secret: secrets for local disk mounting. • PersistentVolumeClaim: cloud storage mounting.
name	Yes	String	Name of the storage disk.
parameters	Yes	Object	For details about the parameters of each data storage type, see Table 4-190 .
mounts	Yes	Array of objects	Mount path of data storage. See Table 4-191 .

Table 4-190 component_storage_parameters

Parameter	Mandatory	Type	Description
path	No	String	Host path. This parameter is applicable to the HostPath storage type. This parameter is mandatory when type is set to HostPath .
name	No	String	Name of a configuration item, secret, or PVC, which is applicable to ConfigMap, Secret, or PersistentVolumeClaim storage, respectively. This parameter is mandatory when type is set to ConfigMap , Secret , or PersistentVolumeClaim .
default_mode	No	Integer	Mounting permission. The value is in decimal format. Example: 384. This parameter is applicable to ConfigMap and secret storage.
medium	No	String	This parameter is applicable to EmptyDir storage. If it is not specified, the default disk medium is used. If it is set to memory , memory storage is enabled.

Table 4-191 mounts

Parameter	Mandatory	Type	Description
path	Yes	String	Mount path.
sub_path	Yes	String	Subpath of the mount path.
read_only	Yes	Boolean	Read-only or not.

Table 4-192 deploy_strategy

Parameter	Mandatory	Type	Description
type	Yes	String	Deployment type. • OneBatchRelease: single-batch upgrade. • RollingRelease: rolling deployment and upgrade. • GrayRelease: dark launch upgrade.

Parameter	Mandatory	Type	Description
rolling_release	No	Object	Rolling deployment parameter, which is mandatory when type is set to RollingRelease . For details, see Table 4-193 .
gray_release	No	Object	Dark launch upgrade parameter, which is mandatory when type is set to GrayRelease . For details, see Table 4-194 .

Table 4-193 rolling_release

Parameter	Mandatory	Type	Description
batches	Yes	Integer	Deployment batches.

Table 4-194 gray_release

Parameter	Mandatory	Type	Description
type	Yes	String	Dark launch policy. • WEIGHT: dark launch by traffic. • CONTENT: dark launch by content.
first_batch_weight	Yes	Integer	Proportion of first-batch dark launch traffic. This parameter is mandatory when type is set to WEIGHT .
first_batch_replica	Yes	Integer	Number of first-batch dark launch instances. This parameter is mandatory when type is set to WEIGHT .
remaining_batches	Yes	Integer	Number of batches for remaining instance deployment. After the first batch is completed, the remaining instances will be deployed in specified batches. Example: If there are 5 instances remaining to be deployed in 3 batches, these 5 instances will be upgraded based on 2:2:1. This parameter is mandatory when type is set to WEIGHT .

Parameter	Mandatory	Type	Description
deployment_mode	No	Integer	Deployment mode. 1: The upstream is a microservice gateway. 3: The upstream is a microservice. 4: The upstream is a load balancer.
replica_surge_mode	No	String	Mode of adding a dark launch instance. This parameter is mandatory when type is set to CONTENT . - MIRROR: blue-green. - EXTRA: canary (increase, then decrease) - NOSURGE: canary (decrease, then increase)
rule_match_mode	No	String	Effective mode of a dark launch rule. This parameter is mandatory when type is set to CONTENT . - ALL: All conditions are met. - ANY: Any condition is met.
rules	No	Object	Dark launch rule. This parameter is mandatory when type is set to CONTENT . See Table 4-195 .

Table 4-195 gray_rules

Parameter	Mandatory	Type	Description
type	Yes	String	Match type. Currently, only HEADER is supported.
key	Yes	String	Parameter name.
value	Yes	String	Condition value.
condition	Yes	String	Condition type. - EQUAL - MATCH - IN: enumeration

Table 4-196 command

Parameter	Man dato ry	Type	Description
command	No	Array of String	Command for controlling container running.
args	No	Array of String	Parameter for controlling container running. Example: -port=8080. If there are multiple parameters, separate them by line breaks.

Table 4-197 component_lifecycle

Parameter	Man dato ry	Type	Description
type	Yes	String	Processing method. • http • command
scheme	No	String	HTTP request type. Value: HTTP or HTTPS. This parameter is valid only when type is set to http .
host	No	String	Custom IP address. If this parameter is left blank, the pod IP address is used. This parameter is valid only when type is set to http .
port	No	Integer	Port. This parameter is valid only when type is set to http .
path	No	String	Request path. This parameter is valid only when type is set to http .
command	No	Array of String	Command list. This parameter is valid only when type is set to command .

Table 4-198 mesher

Parameter	Man dato ry	Type	Description
port	Yes	Integer	Process listening port.

Table 4-199 tomcat_opt

Parameter	Man dato ry	Type	Description
server_xml	Yes	String	1. Listening port in the server.xml file will take effect even if it has been configured in public network access. 2. The custom Tomcat configuration may conflict with the existing port. Check whether the custom port is occupied.

Table 4-200 host_aliases

Parameter	Man dato ry	Type	Description
ip	Yes	String	IP address.
hostnames	Yes	Array of String	Host alias.

Table 4-201 dns_config

Parameter	Man dator y	Type	Description
nameservers	No	Array of String	IP address list of the DNS server of the pod.
searches	No	Array of String	(Optional) List of DNS search domains used to search for host names in the pod.
options	No	Array of object s	An optional list of objects where each object may have a name property (required) and a value property (optional). This property is merged into the options generated from the specified DNS policy. Duplicate entries will be deleted. See Table 4-202 .

Table 4-202 options

Parameter	Mandatory	Type	Description
name	Yes	String	Operation name.
value	No	String	Operation value.

Table 4-203 security_context

Parameter	Mandatory	Type	Description
run_as_user	No	Integer	User for running containers. Example: To run containers as user root , set the user ID to 0 .
run_as_group	No	Integer	Owner group specified for running container processes.
capabilities	No	Object	Capability set. See Table 4-204 .

Table 4-204 capabilities

Parameter	Mandatory	Type	Description
add	No	Array of String	Grant the Linux permission to the running user.
drop	No	Array of String	Remove the Linux permission of the running user.

Table 4-205 logs

Parameter	Mandatory	Type	Description
log_path	Yes	String	Container log path.
rotate	Yes	String	Interval for dumping logs.
host_path	Yes	String	Mounted host path.

Parameter	Mandatory	Type	Description
host_extend_path	Yes	String	Host extension path. By extending the host path, you can distinguish mounting from different containers in the same host path. • None: the extended path is not used. • PodUID: extend the host path based on the pod ID. • PodName: extend the host path based on the pod name. • PodUID/ContainerName: extend the host path based on the pod ID and container name. • PodName/ContainerName: extend the host path based on the pod name and container name.

Table 4-206 custom_metric

Parameter	Mandatory	Type	Description
path	Yes	String	Collection path. Example: /metrics.
port	Yes	Integer	Collection port. Example: 9090.
dimensions	Yes	String	Monitoring dimension. Example: cpu_usage,mem_usage.

Table 4-207 component_affinity

Parameter	Mandatory	Type	Description
kind	No	String	Object kind. • node • pod
condition	No	String	Policy type. • required: The requirement must be met. • preferred: The requirement is met as much as possible.
match_expressions	No	Array of objects	Scheduling policy rule. See Table 4-208 .

Parameter	Mandatory	Type	Description
weight	No	String	Weight. - The value is 100 when condition is set to required. - Enter an integer ranging from 1 to 100 when condition is set to preferred.
topology_key	No	String	Topology key.

Table 4-208 match_expressions

Parameter	Mandatory	Type	Description
key	Yes	String	Key of the match tag.
operation	Yes	String	Match condition.
value	Yes	String	Value of the match tag.

Table 4-209 component_probe

Parameter	Mandatory	Type	Description
type	Yes	String	Type. Value: http, tcp, or command.
delay	Yes	Integer	Interval between the startup and detection.
timeout	Yes	Integer	Detection timeout interval.
scheme	No	String	Request type. Value: HTTP or HTTPS. This parameter is valid only when type is set to http .
host	No	String	Custom IP address. If this parameter is left blank, the pod IP address is used. This parameter is valid only when type is set to http .
port	No	Integer	Port number. This parameter is valid only when type is set to http or tcp .
path	No	String	Request path. This parameter is valid only when type is set to http .
command	No	Array of String	Command list. This parameter is valid only when type is set to command .

Table 4-210 refer_resources

Parameter	Mandatory	Type	Description
id	Yes	String	Resource ID.
type	Yes	String	Resource type. • vpc • eip • elb • cce • ecs • as • cse: ServiceComb engine • nacos: registry/configuration center • dcs • rds • gaussdb • dms_rocketmq • css_es: Elasticsearch
parameters	No	Object	Resource parameter, which is mandatory when type is cce . See Table 4-211 .

Table 4-211 refer_resource_parameter

Parameter	Mandatory	Type	Description
namespace	No	String	Namespace.
name	No	String	Resource name.
capacity	No	Integer	Storage capacity, in GB.
type	No	String	Volume declaration type. • efs • obs • sfs • evs

Parameter	Mandatory	Type	Description
class	No	String	Volume declaration subtype. • csi-sfsturbo: fixed when type is set to efs. • csi-obs: fixed when type is set to obs. • csi-nas: fixed when type is set to sfs. • csi-disk: fixed when type is set to evs.
obs_volume_type	No	String	OBS volume type. Default value: STANDARD.
access_mode	No	String	Storage volume access mode. • ReadWriteOnce: A volume can be mounted as read-write by a single node. • ReadWriteMany: A volume can be mounted as read-write by many nodes.
availableZone	No	String	AZ.
volume_type	No	String	Volume type.

Table 4-212 envs

Parameter	Mandatory	Type	Description
name	Yes	String	Variable name.
value	No	String	Variable value.
value_from	No	Object	Variable reference. See Table 4-213 .

Table 4-213 value_from

Parameter	Mandatory	Type	Description
reference_type	Yes	String	Reference type. • configMapKey • secretKey
name	Yes	String	Name of the configuration item or secret.
key	No	String	Key of the configuration item or secret.
optional	No	Boolean	Whether the key of the configuration item or secret is mandatory. Value: true and false.

Table 4-214 runtime_stack

Parameter	Mandatory	Type	Description
name	Yes	String	Technology stack name.
type	Yes	String	Technology stack type. Value: Java, Tomcat, Nodejs, Php, Docker, or Python. When deploy_mode is set to virtualmachine , only Java, Tomcat, and Node.js are supported. When deploy_mode is set to container , all of the preceding types are supported.
version	Yes	String	Technology stack version.
deploy_mode	Yes	String	Deployment mode. • container • virtualmachine

Table 4-215 source

Parameter	Mandatory	Type	Description
kind	Yes	String	Component source kind. • code • package • image
version	No	String	Version number.
url	No	String	Package address. This parameter is mandatory when kind is set to package or image .
storage	No	String	Storage mode. • devcloud: CodeArts release repository • obs • http: custom file address • swr This parameter is mandatory when kind is set to package or image .
auth	No	String	Authentication mode. Value: iam or none. Default value: iam.
repo_auth	No	String	Authorization name, which can be obtained from the authorization list.
repo_namespace	No	String	Namespace of the code repository.

Parameter	Mandatory	Type	Description
repo_ref	No	String	Code branch or tag. Default value: master.
repo_type	No	String	Code repository type. Value: GitHub, BitBucket, or GitLab.
web_url	No	String	Code redirection URL. Example: https://github.com/example/demo.git.
repo_url	No	String	Code repository URL. Example: https://github.com/example/demo.git.
commit_id	No	String	In the code repository, commit ID is the unique identifier of each commit and is used to identify and reference a commit. The value contains 8 to 40 characters and starts with a lowercase letter or digit.
http_username	No	String	Username. This parameter is mandatory when storage is set to http . Username and password authentication is supported.
http_password	No	String	Password. This parameter is mandatory when storage is set to http . Username and password authentication is supported.
http_command	No	String	Address of the custom software package. This parameter is mandatory when storage is set to http .
http_headers	No	Map<String, String>	This parameter is mandatory when storage is set to http . You can customize request headers as key-value pairs, for example, {"key": "xx", "value": "xx"}.

Table 4-216 build

Parameter	Mandatory	Type	Description
parameters	No	Map<String, Object>	This parameter is provided only when no ID is available during build creation. See Table 4-217 .

Table 4-217 parameters

Parameter	Mandatory	Type	Description
build_cmd	No	String	Compilation command. By default: 1. When build.sh exists in the root directory, the command is ./build.sh . 2. When build.sh does not exist in the root directory, the command varies depending on the OS. Example: • Java and Tomcat: mvn clean package • Nodejs: npm build
dockerfile_path	No	String	Address of the Docker file. By default, the Docker file is in the root directory (./).
artifact_namespace	No	String	Build archive organization. Default value: cas_{project_id} .
cluster_id	Yes	String	ID of the cluster to be built.
node_label_selector	No	Map<String, String>	key: label key. value: label value.
environment_id	No	String	Environment ID.

Table 4-218 external_accesses

Parameter	Mandatory	Type	Description
protocol	No	String	External access type. Value: http or https.
address	No	String	External access address.
forward_port	No	Integer	External access port.

Table 4-219 auto_lts_config

Parameter	Mandatory	Type	Description
enable	Yes	Boolean	Whether to enable automatic LTS configuration.
lts_log_path	No	Array of String	Access configuration log path.

Table 4-220 tolerations

Parameter	Mandatory	Type	Description
key	No	String	Taint key.
operator	No	String	Operator. • Equal: matches the nodes with the specified taint key (mandatory) and value. If the taint value is left blank, all taints with the same key will be matched. • Exists: matches nodes with the specified taint key. In this case, the taint value cannot be specified. Not specifying any taint key means all taints are tolerated.
value	No	String	Taint value. This parameter is mandatory when operator is set to Equal .
effect	No	String	Taint policy. • NoScheduler: Only the NoSchedule taint is matched. • PreferNoSchedule: Only the PreferNoSchedule taint is matched. • NoExecute: Only the NoExecute taint is matched. • If this parameter is not set, all taint effects are matched.
toleration_seconds	No	Integer	Tolerance time window. This parameter can be set only when effect is set to NoExecute . Within the tolerance time window, pods still run on the node with taints. After the time expires, the pods will be evicted.

Table 4-221 update_strategy

Parameter	Mandatory	Type	Description
type	No	String	Component upgrade policy. Options: • RollingUpdate (default): New pods are created gradually, and then old pods are deleted. • Recreate: Old pods are deleted, and then new pods are created.
max_unavailable	No	• String • Integer	Maximum number of unavailable pods. That is, the number or percentage of pods that can be deleted based on the value of replica . The default value is 25% . This parameter can be set when workload_kind is set to deployment . • String (%): the maximum percentage of unavailable pods to the total number of pods. • Integer: the maximum number of unavailable pods.
max_surge	No	• String • Integer	Maximum surge. That is, the number or percentage of pods allowed based on the value of replica . The default value is 25% . This parameter can be set only when workload_kind is set to deployment . • String (%) • Integer
termination_period_seconds	No	Integer	Scale-in time window. That is, the time window for executing commands before a workload stops. The default value is 30s.
min_ready_seconds	No	Integer	Minimum ready time of a pod. That is, the minimum ready time of a new pod without any container crash. A pod is considered available only after the minimum ready time is exceeded. The default value is 0 .

Parameter	Mandatory	Type	Description
progress_dead_line_seconds	No	Integer	Maximum upgrade duration. That is, the time (in seconds) the system waits for before reporting a component upgrade failure. If this parameter is specified, the value of this parameter must be greater than that of min_ready_seconds .
revision_history_limit	No	Integer	Maximum number of pods to retain. That is, the number of old pods to retain to allow rollback. The default value is 10 .

Table 4-222 multi_containers

Parameter	Mandatory	Type	Description
container_name	Yes	String	Container name.
source	Yes	Object	Source of the code or software package. See Table 4-215 . In multi-container mode, only images are supported.
limit_cpu	No	Number	Maximum CPU limit. Unit: core.
limit_memory	No	Number	Maximum memory size. Unit: GiB.
request_cpu	No	Number	Requested CPU resources. Unit: core.
request_memory	No	Number	Requested memory. Unit: GiB.
envs	No	Array of objects	Environment variable list. See Table 4-212 .
liveness_probe	No	Object	Container liveness probe. See Table 4-209 .
readiness_probe	No	Object	Container service probe. See Table 4-209 .
startup_probe	No	Object	Container startup probe. See Table 4-209 .
command	No	Object	Startup command. See Table 4-196 .

Parameter	Mandatory	Type	Description
post_start	No	Object	Post-start processing. See Table 4-197 .
pre_stop	No	Object	Pre-stop processing. See Table 4-197 .
logs	No	Array of objects	Log collection. See Table 4-205 .
storages	No	Array of objects	Storage. See Table 4-189 .
security_context	No	Object	Security policy. See Table 4-203 .

Table 4-223 configs

Parameter	Mandatory	Type	Description
config_id	Yes	String	Unified configuration ID.
mount_path	Yes	String	Mount path.

Table 4-224 template_source

Parameter	Mandatory	Type	Description
storage	Yes	String	Storage mode of the component template package. • devcloud: CodeArts release repository • obs • http: custom file address • devcloud_repository: CodeArts artifact
url	Yes	String	Address of the component template package. This parameter is valid only when the component template package storage mode is http .
http_username	No	String	Username. This parameter is valid only when the component template package storage mode is http and the security authentication mode is username and password authentication.

Parameter	Mandatory	Type	Description
http_password	No	String	Password. This parameter is valid only when the component template package storage mode is http and the security authentication mode is username and password authentication.
http_headers	No	Array of objects	Request header. See Table 4-225 . This parameter is valid only when the component template package storage mode is http and the security authentication mode is custom header authentication.

Table 4-225 http_headers

Parameter	Mandatory	Type	Description
key	Yes	String	Request key.
value	Yes	String	Request value.

Table 4-226 config_encryption

Parameter	Mandatory	Type	Description
key_id	Yes	String	User key ID in DEW.

Response

Table 4-227 Response parameters

Parameter	Type	Description
job_id	String	Job ID.

Example Request

Modify the component named **mycomponent**. The component source is the **weather-1.0.0.jar** package stored in OBS **bucket0001**.

```
{  
  "name": "mycomponent",
```

```
{
  "description": "",
  "labels": [
    {
      "key": "com-key",
      "value": "com-value"
    }
  ],
  "version": "2023.0323.15181",
  "runtime_stack": {
    "name": "OpenJDK8",
    "version": "1.1.1",
    "type": "Java",
    "deploy_mode": "virtualmachine"
  },
  "source": {
    "kind": "package",
    "url": "obs://bucket0001/weather-1.0.0.jar",
    "version": "",
    "storage": "obs"
  },
  "tomcat_opts": {
    "server_xml": ""
  },
  "refer_resources": [
    {
      "id": "67835bb3-1235-4cc9-be71-becbb2b4ca0d",
      "type": "ecs"
    }
  ],
  "replica": 1,
  "external_accesses": []
}
```

Example Response

```
{
  "job_id": "JOB1c8e20ec-1b30-4ee4-9a36-35a18b5e8b7e"
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.3.6 Deleting a Component Based on the Component ID

Function

This API is used to delete a component based on the component ID.

URI

DELETE /v3/{project_id}/cas/applications/{application_id}/components/{component_id}

Table 4-228 Path parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
application_id	Yes	String	Application ID. See Obtaining All Applications .
component_id	Yes	String	Component ID. See Obtaining All Components of an Application .

Request

Table 4-229 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

None

Example Request

None

Example Response

None

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
409	Conflict
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.3.7 Obtaining Component Information Based on the Component ID

Function

This API is used to obtain component information based on the component ID.

URI

GET /v3/{project_id}/cas/applications/{application_id}/components/{component_id}

Table 4-230 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
application_id	Yes	String	Application ID. See Obtaining All Applications .
component_id	Yes	String	Component ID. See Obtaining All Components of an Application .

Table 4-231 Query parameter

Parameter	Mandatory	Type	Description
version	No	String	Component version. See Obtaining All Components of an Application .

Request

Table 4-232 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

Table 4-233 Response parameters

Parameter	Type	Description
name	String	Application component name. The value contains 2 to 64 characters consisting of letters, digits, hyphens (-), and underscores (_). It starts with a letter and ends with a letter or digit. Letters are case insensitive.
workload_name	String	Workload name.
labels	Array of objects	Label. For container-deployed components, you can query a workload by label. For VM-deployed components, you can configure a label of the corresponding microservice name to use graceful startup and shutdown. See Table 4-234 .

Parameter	Type	Description
runtime_stack	Object	Runtime. See Table 4-260 . The value can be obtained from the response of the API for Querying a Technology Stack .
environment_id	String	Environment ID.
application_id	String	Application ID.
description	String	Description. The value can contain up to 128 characters.
source	Object	Source of the code or software package. See Table 4-261 .
build	Object	Component build. See Table 4-262 .
limit_cpu	Number	Maximum CPU limit. Unit: core.
limit_memory	Number	Maximum memory size. Unit: GiB.
request_cpu	Number	Requested CPU resources. Unit: core.
request_memory	Number	Requested memory. Unit: GiB.
version	String	Component version number. The value can contain up to 32 characters. It needs to meet the following rule: $\wedge([0-9]+)(.[0-9]+){2,3}\$$
envs	Array of objects	Environment variable list. See Table 4-258 .
replica	Integer	Number of instances.
storages	Array of objects	Storage. See Table 4-235 .
deploy_strategy	Object	Component deployment. See Table 4-238 .
command	Object	Startup command. See Table 4-242 .
post_start	Object	Post-start processing. See Table 4-243 .
pre_stop	Object	Pre-stop processing. See Table 4-243 .
mesher	Object	Multi-language access service mesh. This parameter is mandatory when Node.js is selected for the technology stack and the CSE engine is bound. See Table 4-244 .
timezone	String	Time zone where the component runs. Example: Asia/Shanghai.

Parameter	Type	Description
jvm_opts	String	JVM parameter. Example: -Xms256m -Xmx1024m. If there are multiple parameters, separate them by spaces. If the parameter is left blank, the default value is used.
tomcat_opts	Object	Tomcat parameter. This parameter is valid when you use the Tomcat technology stack. See Table 4-245 .
host_aliases	Array of objects	Host alias. See Table 4-246 .
dns_policy	String	DNS policy. • Default: inherits the domain name resolution configuration of the node where the pod is located. • ClusterFirst: adds the domain name resolution configuration. • ClusterFirstWithHostNet: takes effect only for pods running in hostNetwork mode. • None: replaces the domain name resolution policy.
dns_config	Object	DNS configuration. See Table 4-247 .
workload_kind	String	Workload type. • deployment • statefulset • daemonset
security_context	Object	Security policy. See Table 4-249 .
logs	Array of objects	Log collection. See Table 4-251 .
custom_metrics	Object	Custom metrics. See Table 4-252 .
affinity	Array of objects	Affinity. See Table 4-253 .
anti_affinity	Array of objects	Anti-affinity. See Table 4-253 .
liveness_probe	Array of objects	Component liveness probe. See Table 4-255 .
readiness_probe	Object	Component service probe. See Table 4-255 .

Parameter	Type	Description
refer_resources	Array of objects	Associated resource. See Table 4-256 .
status	Object	Component status. See Table 4-264 .
tolerations	Array of Objects	Tolerance policy. See Table 4-266 .
container_name	String	Container name.
update_strategy	Object	Component upgrade strategy. See Table 4-268 .
multi_containers	Array of Objects	Multi-container configuration. See Table 4-267 .
service_name	String	Service name configured for the inter-instance discovery of a StatefulSet.
ports	Array of Objects	Port configured for the inter-instance discovery of a StatefulSet.
enterprise_project_id	String	Enterprise project ID.
swimlane_tag	String	Lane tag.
swimlane_id	String	Lane ID.
refer_resources	Array of Objects	Associated resource. See Table 4-256 .
runtime_stack_id	String	Runtime ID.
component_id	String	Component ID.
deploy_type	String	Deployment type. • OneBatchRelease • RollingRelease • GrayRelease
platform_type	String	Type of the platform where an application runs. Values: • cce • vmapp
config_mode	String	Workload configuration mode. • ui • yaml • template

Parameter	Type	Description
workload_content	String	Workload content.
workload_name	String	Workload name.
instance_id	String	Component instance ID.
sys_config	Map<String, String>	Uses a compressed package to deploy component configuration items in the VM environment.
configs	Array of Objects	Configuration file list. See Table 4-269 .
enable_serment_injection	Boolean	Whether to enable Serment.
enable_serment_spring_register	Boolean	Whether Spring Cloud Huawei microservices need to be registered with registry centers through Serment.
enable_serment_dubbo_register	Boolean	Whether Dubbo microservices need to be registered with registry centers through Serment.
enable_serment_config	Boolean	Whether to enable the Serment dynamic configuration plug-in.
image_pull_secrets	Array of String	List of image pull secrets.
template_source	Object	Source of the component template package. See Table 4-270 .
template_content	Object	Component template package content. See Table 4-272 .
resource_content	Object	Details about component template package resource. See Table 4-273 .
auto_lts_config	Object	Automatic LTS configuration. See Table 4-276 .
config_encryption	Object	This parameter is mandatory when the encryption key is configured for a single-container component. See Table 4-277 .

Table 4-234 labels

Parameter	Type	Description
key	String	Label name.

Parameter	Type	Description
value	String	Label value.

Table 4-235 storages

Parameter	Type	Description
type	String	Data storage type. • HostPath: host path for local disk mounting. • EmptyDir: temporary directory for local disk mounting. • ConfigMap: configuration item for local disk mounting. • Secret: secrets for local disk mounting. • PersistentVolumeClaim: cloud storage mounting.
name	String	Name of the storage disk.
parameters	Object	For details about the parameters of each data storage type, see Table 4-236 .
mounts	Array of objects	Mount path of data storage. See Table 4-237 .

Table 4-236 component_storage_parameters

Parameter	Type	Description
path	String	Host path. This parameter is applicable to the HostPath storage type.
name	String	Name of a configuration item, secret, or PVC, which is applicable to ConfigMap, Secret, or PersistentVolumeClaim storage, respectively.
default_mode	Integer	Mounting permission. The value is in decimal format. Example: 384. This parameter is applicable to ConfigMap and secret storage.
medium	String	This parameter is applicable to the EmptyDir storage type. Example: memory.

Table 4-237 mounts

Parameter	Type	Description
path	String	Mount path.
sub_path	String	Subpath of the mount path.
read_only	Boolean	Read-only or not.

Table 4-238 deploy_strategy

Parameter	Type	Description
type	String	Deployment type. • OneBatchRelease: single-batch upgrade. • RollingRelease: rolling deployment and upgrade. • GrayRelease: dark launch upgrade.
rolling_release	Object	Rolling deployment parameter. See Table 4-239 .
gray_release	Object	Dark launch upgrade. See Table 4-240 .

Table 4-239 rolling_release

Parameter	Type	Description
batches	Integer	Deployment batches.

Table 4-240 gray_release

Parameter	Type	Description
type	String	Dark launch policy. • WEIGHT: dark launch by traffic. • CONTENT: dark launch by content.
first_batch_weight	Integer	Proportion of first-batch dark launch traffic. This parameter takes effect when type is set to WEIGHT .
first_batch_replica	Integer	Number of first-batch dark launch instances. This parameter takes effect when type is set to WEIGHT .

Parameter	Type	Description
remaining_batches	Integer	Number of batches for remaining instance deployment. After the first batch is completed, the remaining instances will be deployed in specified batches. Example: If there are 5 instances remaining to be deployed in 3 batches, these 5 instances will be upgraded based on 2:2:1. This parameter takes effect when type is set to WEIGHT .
deployment_mode	Integer	Deployment mode. • 1: The upstream is a microservice gateway. • 3: The upstream is a microservice. • 4: The upstream is a load balancer.
replica_surge_mode	String	Mode of adding a dark launch instance. This parameter takes effect when type is set to CONTENT . • MIRROR: blue-green. • EXTRA: canary (increase, then decrease) • NOSURGE: canary (decrease, then increase)
rule_match_mode	String	Effective mode of a dark launch rule. This parameter takes effect when type is set to CONTENT . • ALL: All conditions are met. • ANY: Any condition is met.
rules	Object	Dark launch rule. This parameter takes effect when type is set to CONTENT . See Table 4-241 .

Table 4-241 gray_rules

Parameter	Type	Description
type	String	Match type. Currently, only HEADER is supported.
key	String	Parameter name.
value	String	Condition value.
condition	String	Condition type. • EQUAL • MATCH • IN: enumeration

Table 4-242 command

Parameter	Type	Description
command	Array of String	Command for controlling container running.
args	Array of String	Parameter for controlling container running. Example: -port=8080. If there are multiple parameters, separate them by line breaks.

Table 4-243 component_lifecycle

Parameter	Type	Description
type	String	Processing method. • http • command
scheme	String	HTTP request type, which can be HTTP or HTTPS. This parameter is valid only when type is set to http .
host	String	Custom IP address. If this parameter is left blank, the pod IP address is used. This parameter is valid only when type is set to http .
port	Integer	Port number. This parameter is valid only when type is set to http .
path	String	Request path. This parameter is valid only when type is set to http .
command	Array of String	Command list. This parameter is valid only when type is set to command .

Table 4-244 mesher

Parameter	Type	Description
port	Integer	Process listening port.

Table 4-245 tomcat_opt

Parameter	Type	Description
server_xml	String	1. Listening port in the server.xml file will take effect even if it has been configured in public network access. 2. The custom Tomcat configuration may conflict with the existing port. Check whether the custom port is occupied.

Table 4-246 host_aliases

Parameter	Type	Description
ip	String	IP address.
hostname	Array of String	Host alias.

Table 4-247 dns_config

Parameter	Type	Description
nameservers	Array of String	IP address list of the DNS server of the pod.
searches	Array of String	List of DNS search domains used to search for host names in the pod.
options	Array of objects	An optional list of objects where each object may have a name property (required) and a value property (optional). This property is merged into the options generated from the specified DNS policy. Duplicate entries will be deleted. See Table 4-248 .

Table 4-248 options

Parameter	Type	Description
name	String	Operation name.
value	String	Operation value.

Table 4-249 security_context

Parameter	Type	Description
run_as_user	Integer	User for running containers. Example: To run containers as user root , set the user ID to 0 .
run_as_group	Integer	Owner group specified for running container processes.
capabilities	Object	Capability set. See Table 4-250 .

Table 4-250 capabilities

Parameter	Type	Description
add	Array of String	Grant the Linux permission to the running user.
drop	Array of String	Remove the Linux permission of the running user.

Table 4-251 logs

Parameter	Type	Description
log_path	String	Container log path.
rotate	String	Interval for dumping logs.
host_path	String	Mounted host path.
host_extend_path	String	Host extension path. By extending the host path, you can distinguish mounting from different containers in the same host path. • None: the extended path is not used. • PodUID: extend the host path based on the pod ID. • PodName: extend the host path based on the pod name. • PodUID/ContainerName: extend the host path based on the pod ID and container name. • PodName/ContainerName: extend the host path based on the pod name and container name.

Table 4-252 custom_metric

Parameter	Type	Description
path	String	Collection path. Example: /metrics.
port	Integer	Collection port. Example: 9090.
dimensions	String	Monitoring dimension. Example: cpu_usage,mem_usage.

Table 4-253 component_affinity

Parameter	Type	Description
kind	String	Object kind. • node • pod
condition	String	Policy type. • required: The requirement must be met. • preferred: The requirement is met as much as possible.
match_expressions	Array of objects	Scheduling policy rule. See Table 4-254 .
weight	String	Weight. • The value is 100 when condition is set to required. • Enter an integer ranging from 1 to 100 when condition is set to preferred.
topology_key	String	Topology key.

Table 4-254 match_expressions

Parameter	Type	Description
key	String	Key of the match tag.
operation	String	Match condition.
value	String	Value of the match tag.

Table 4-255 component_probe

Parameter	Type	Description
type	String	Type. Value: http, tcp, or command.
delay	Integer	Interval between the startup and detection.
timeout	Integer	Detection timeout interval.
scheme	String	Request type. Value: HTTP or HTTPS. This parameter is valid only when type is set to http .
host	String	Custom IP address. If this parameter is left blank, the pod IP address is used. This parameter is valid only when type is set to http .
port	Integer	Port number. This parameter is valid only when type is set to http or tcp .
path	String	Request path. This parameter is valid only when type is set to http .
command	Array of String	Command list. This parameter is valid only when type is set to command .

Table 4-256 refer_resources

Parameter	Type	Description
id	String	Resource ID.
type	String	Resource type. • vpc • eip • elb • cce • ecs • as • cse: ServiceComb engine • nacos: registry/configuration center • dcs • rds • gaussdb • dms_rocketmq • css_es: Elasticsearch
parameters	Object	Resource parameters. See Table 4-257 .

Table 4-257 refer_resource_parameter

Parameter	Type	Description
namespace	String	Namespace.
name	String	Resource name.
capacity	Integer	Storage capacity, in GiB.
type	String	Volume declaration type. It is specified when the type is set to pvc in Table 4-256 . • efs • obs • sfs • evs
class	String	Volume declaration subtype. • csi-sfsturbo: fixed when type is set to efs. • csi-obs: fixed when type is set to obs. • csi-nas: fixed when type is set to sfs. • csi-disk: fixed when type is set to evs.
pv_name	String	Volume name. This parameter is displayed when type is set to efs .
share_id	String	Volume handle. This parameter is displayed when type is set to efs .
share_export_location	String	Shared path of the volume. This parameter is displayed when type is set to efs .
share_proto	String	File system type of the volume. This parameter is displayed when type is set to efs . Default value: NFS.
obs_volume_type	String	OBS volume type. Default value: STANDARD.
access_mode	String	Storage volume access mode. • ReadWriteOnce: A volume can be mounted as read-write by a single node. • ReadWriteMany: A volume can be mounted as read-write by many nodes.
availableZone	String	AZ.
volume_type	String	Volume type.

Table 4-258 envs

Parameter	Type	Description
name	String	Variable name.
value	String	Variable value.
value_from	Object	Variable reference. See Table 4-259 .

Table 4-259 value_from

Parameter	Type	Description
reference_type	String	Reference type. - configMapKey - secretKey
name	String	Name of the configuration item or secret.
key	String	Key of the configuration item or secret.
optional	Boolean	Whether the key of the configuration item or secret is mandatory.

Table 4-260 runtime_stack

Parameter	Type	Description
name	String	Technology stack name.
type	String	Technology stack type. Value: Java, Tomcat, Nodejs, Php, Docker, or Python. When deploy_mode is set to virtualmachine , only Java, Tomcat, and Node.js are supported. When deploy_mode is set to container , all of the preceding types are supported.
version	String	Technology stack version.
deploy_mode	String	Deployment mode. - container - virtualmachine

Table 4-261 source

Parameter	Type	Description
kind	String	Component source kind. • code • package • image
version	String	Version number.
url	String	Package address. This parameter is mandatory when kind is set to package or image .
storage	String	Storage mode. • devcloud: CodeArts release repository • obs • http: custom file address • swr This parameter is mandatory when kind is set to package or image .
auth	String	Authentication mode. Value: iam or none. Default value: iam.
repo_auth	String	Authorization name, which can be obtained from the authorization list.
repo_namespace	String	Namespace of the code repository.
repo_ref	String	Code branch or tag. Default value: master.
repo_type	String	Code repository type. Value: GitHub, BitBucket, or GitLab.
web_url	String	Code redirection URL. Example: https://github.com/example/demo.git.
repo_url	String	Code repository URL. Example: https://github.com/example/demo.git.
commit_id	String	In the code repository, commit ID is the unique identifier of each commit and is used to identify and reference a commit. The value contains 8 to 40 characters and starts with a lowercase letter or digit. This parameter is returned only when Request contains Table 4-231 .

Parameter	Type	Description
http_username	No	String Username. This parameter is mandatory when storage is set to http . Username and password authentication is supported.
http_password	No	String Password. This parameter is mandatory when storage is set to http . Username and password authentication is supported.
http_command	No	String Address of the custom software package. This parameter is mandatory when storage is set to http .
http_headers	No	Map<String, String> This parameter is mandatory when storage is set to http . You can customize request headers as key-value pairs, for example, {"key": "xx", "value": "xx"}.

Table 4-262 build

Parameter	Type	Description
parameters	Map<String, Object>	This parameter is provided only when no ID is available during build creation. See Table 4-263 .

Table 4-263 parameters

Parameter	Type	Description
build_cmd	String	Compilation command. By default: 1. When build.sh exists in the root directory, the command is ./build.sh . 2. When build.sh does not exist in the root directory, the command varies depending on the OS. Example: • Java and Tomcat: mvn clean package • Nodejs: npm build
dockerfile_path	String	Address of the Docker file. By default, the Docker file is in the root directory (./).

Parameter	Type	Description
artifact_name space	String	Build archive organization. Default value: cas_{project_id}.
cluster_id	String	ID of the cluster to be built.
node_label_selector	Map<String, String>	key: label key. value: label value.
environment_id	String	Environment ID.

Table 4-264 status

Parameter	Type	Description
component_status	String	Component status. • INITIALIZING • UPGRADING • FAILED • RUNNING • DOWN • DELETING • DELETED • RESERVED • STARTING • STOPPING • STOPPED • RESTARTING • PENDING • UNKNOWN • PARTIALLY_FAILED
available_replica	Integer	Available instances.
replica	Integer	Total instances.

Parameter	Type	Description
fail_detail	String	Failure cause. • cluster_deleted • cluster_unavailable • cluster_inaccessible • namespace_deleted • namespace_unavailable • namespace_inaccessible • resource_deleted • create_failed • delete_failed
last_job_id	String	ID of the last executed task.
create_time	Integer	Creation time.
update_time	Integer	Update time.
creator	String	Creator.
artifact	Object	Artifact information. See Table 4-265 .

Table 4-265 artifact

Parameter	Type	Description
type	String	Component source type. • package • image
url	String	Software package or image address.

Table 4-266 tolerations

Parameter	Type	Description
key	String	Taint key.

Parameter	Type	Description
operator	String	Operator. • Equal: matches the nodes with the specified taint key (mandatory) and value. If the taint value is left blank, all taints with the same key will be matched. • Exists: matches nodes with the specified taint key. In this case, the taint value cannot be specified. Not specifying any taint key means all taints are tolerated.
value	String	Taint value. This parameter is mandatory when operator is set to Equal .
effect	String	Taint policy. • NoScheduler: Only the NoSchedule taint is matched. • PreferNoSchedule: Only the PreferNoSchedule taint is matched. • NoExecute: Only the NoExecute taint is matched. • If this parameter is not set, all taint effects are matched.
toleration_seconds	Integer	Tolerance time window. This parameter can be set only when effect is set to NoExecute . Within the tolerance time window, pods still run on the node with taints. After the time expires, the pods will be evicted.

Table 4-267 multi_containers

Parameter	Type	Description
container_name	String	Container name.
source	Object	Source of the code or software package. See Table 4-261 . In multi-container mode, only images are supported.
limit_cpu	Number	Maximum CPU limit. Unit: core.
limit_memory	Number	Maximum memory size. Unit: GiB.
request_cpu	Number	Requested CPU resources. Unit: core.
request_memory	Number	Requested memory. Unit: GiB.

Parameter	Type	Description
envs	Array of objects	Environment variable list. See Table 4-258 .
liveness_probe	Object	Container liveness probe. See Table 4-255 .
readiness_probe	Object	Container service probe. See Table 4-255 .
startup_probe	Object	Container startup probe. See Table 4-255 .
command	Object	Startup command. See Table 4-242 .
post_start	Object	Post-start processing. See Table 4-243 .
pre_stop	Object	Pre-stop processing. See Table 4-243 .
logs	Array of objects	Log collection. See Table 4-251 .
storages	Array of objects	Storage. See Table 4-235 .
security_context	Object	Security policy. See Table 4-249 .
config_encryption	Object	This parameter is mandatory when the encryption key is configured for a multi-container component. See Table 4-277 .

Table 4-268 update_strategy

Parameter	Type	Description
type	String	Component upgrade policy. Options: - RollingUpdate (default): New pods are created gradually, and then old pods are deleted. - Recreate: Old pods are deleted, and then new pods are created.

Parameter	Type	Description
max_unavailable	- String - Integer	Maximum number of unavailable pods. That is, the number or percentage of pods that can be deleted based on the value of replica. The default value is 25%. This parameter can be set when workload_kind is set to deployment. - String (%): the maximum percentage of unavailable pods to the total number of pods. - Integer: the maximum number of unavailable pods.
max_surge	- String - Integer	Maximum surge. That is, the number or percentage of pods allowed based on the value of replica. The default value is 25%. This parameter can be set only when workload_kind is set to deployment. - String (%) - Integer
termination_period_seconds	Integer	Scale-in time window. That is, the time window for executing commands before a workload stops. The default value is 30s.
min_ready_seconds	Integer	Minimum ready time of a pod. That is, the minimum ready time of a new pod without any container crash. A pod is considered available only after the minimum ready time is exceeded. The default value is 0.
progress_deadline_seconds	Integer	Maximum upgrade duration. That is, the time (in seconds) the system waits for before reporting a component upgrade failure. If this parameter is specified, the value of this parameter must be greater than that of min_ready_seconds.
revision_history_limit	Integer	Maximum number of pods to retain. That is, the number of old pods to retain to allow rollback. The default value is 10.

Table 4-269 configs

Parameter	Type	Description
config_id	String	Unified configuration ID.
mount_path	String	Mount path.

Table 4-270 template_source

Parameter	Type	Description
storage	String	Storage mode of the component template package. • devcloud: CodeArts release repository • obs • http: custom file address
url	String	Address of the component template package. This parameter is valid only when the component template package storage mode is http .
http_username	String	Username. This parameter is valid only when the component template package storage mode is http and the security authentication mode is username and password authentication.
http_password	String	Password. This parameter is valid only when the component template package storage mode is http and the security authentication mode is username and password authentication.
http_headers	Array of objects	Request header. See Table 4-271 . This parameter is valid only when the component template package storage mode is http and the security authentication mode is custom header authentication.

Table 4-271 http_headers

Parameter	Type	Description
key	String	Request key.
value	String	Request value.

Table 4-272 template_content

Parameter	Type	Description
template	Map<String, String>	Template package file information.

Parameter	Type	Description
input_values	Map<String, String>	Template package input information.

Table 4-273 resource_content

Parameter	Type	Description
configs	Array of objects	Unified configuration information of the template package. See Table configs .
k8s_resources	Array of objects	Template package Kubernetes resource information.

Table 4-274 configs

Parameter	Type	Description
kind	String	Resource type. The value is fixed to ServiceStageConfig .
spec	Object	Unified configuration parameter. See Table 4-275 .

Table 4-275 spec

Parameter	Type	Description
name	String	Unified configuration name. Format: • {Configuration group}/{Configuration name}.yaml • {Configuration group}/{Configuration name}.properties
description	String	Unified configuration description.
sensitive	Boolean	Encrypted or not.
content	String	Unified configuration content.

Table 4-276 auto_lts_config

Parameter	Type	Description
enable	Boolean	Whether to enable automatic LTS configuration.

Parameter	Type	Description
lts_log_path	Array of string	Access configuration log path.
lts_log_group_name	String	Log group name.
lts_log_stream_name	String	Log stream name.
lts_host_group_name	String	Host group name.
lts_access_config_name	String	Access configuration name.
lts_log_group_id	String	Log group ID.
lts_log_stream_id	String	Log stream ID.
lts_host_group_id	String	Host group ID.
lts_access_config_id	String	Access configuration ID.

Table 4-277 config_encryption

Parameter	Type	Description
key_id	String	User key ID in DEW.

Example Request

None

Example Response

```
{
  "name": "test-component",
  "description": "",
  "labels": [],
  "runtime_stack": {
    "type": "Java",
    "name": "OpenJDK8",
    "deploy_mode": "virtualmachine",
    "version": "1.1.1"
  },
  "id": "b8702b0f-94d3-4822-98a1-56815632a0a0",
  "source": {
    "kind": "package",
    "url": "obs://bucket0001/weather-1.0.0.jar",
    "storage": "obs"
  },
  "environment_id": "4d084044-0b80-4641-963c-b9c9f4092a4f",
  "application_id": "fc092465-a5fb-4a52-bc65-b735f18366d8",
}
```

```
{
  "replica": 1,
  "version": "2023.0323.15181",
  "envs": [],
  "tomcat_opts": {
    "server_xml": "",
    "http_port": 0,
    "context_path": null
  },
  "refer_resources": [
    {
      "id": "Default",
      "type": "ecs",
      "parameters": {
        "hosts": [
          "67835bb3-1235-4cc9-be71-becbb2b4ca0d"
        ]
      }
    }
  ],
  "status": {
    "component_status": "RUNNING",
    "available_replica": 1,
    "replica": 1,
    "fail_detail": null,
    "last_job_id": "JOB1c8e20ec-1b30-4ee4-9a36-35a18b5e8b7e",
    "creator": null,
    "create_time": 1679556221933,
    "update_time": 1679579448500,
  },
  "workload_kind": "deployment"
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.3.8 Delivering a Component Task Based on the Component ID

Function

This API is used to deliver a component task based on the component ID.

URI

POST /v3/{project_id}/cas/applications/{application_id}/components/
{component_id}/action

Table 4-278 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
application_id	Yes	String	Application ID. See Obtaining All Applications .
component_id	Yes	String	Component ID. See Obtaining All Components of an Application .

Request

Table 4-279 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Table 4-280 Request body parameters

Parameter	Mandatory	Type	Description
action	Yes	String	Component task type. • start • stop • restart • scale • rollback • rollback_current: roll back to the previous version. • continue_deploy • check_gray_release • modify_gray_rule • sync_workload • sync_import: Compare workload information before and after workload import. • abort_job: Abort a component deployment or upgrade task. • redeployment
parameters	No	Object	Task parameters. See Table 4-281 .

Table 4-281 parameters

Parameter	Mandatory	Type	Description
replica	No	Integer	Number of component instances.
version	No	String	Version number.
hosts	No	Array of String	Host ID list, which is mandatory when component scaling is deployed on a VM.

Response

Table 4-282 Response parameters

Parameter	Type	Description
job_id	String	Job ID.
component_id	String	Component ID.

Example Request

Start a component.

```
{
  "action": "start",
  "parameters": {
    "replica": 0,
    "hosts": [
      "string"
    ],
    "version": "string"
  }
}
```

Example Response

```
{
  "job_id": "JOBa6346016-b97d-4311-9e4c-9a40bd39ead8",
  "component_id": "29e7ea25-ce11-4cf9-9a04-3226fb26d60e"
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.3.9 Obtaining Records Based on the Component ID

Function

This API is used to obtain records based on the component ID.

URI

GET /v3/{project_id}/cas/applications/{application_id}/components/{component_id}/records

Table 4-283 Path parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
application_id	Yes	String	Application ID. See Obtaining All Applications .
component_id	Yes	String	Component ID. See Obtaining All Components of an Application .

Table 4-284 Query parameter

Parameter	Mandatory	Type	Description
limit	No	String	Number of records to be queried. Value range: 0–100.
offset	No	String	Offset.
order_by	No	String	Sorting field. Enumerated values: • create_time (default) • name • update_time If the transferred value is not within the supported enumerated value range, the default sorting field is used.
order	No	String	Sorting order. • desc (default) • asc

Request

Table 4-285 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

Table 4-286 Response parameters

Parameter	Type	Description
count	Integer	Total number of records.
records	Array of objects	Record list. See Table 4-287 .

Table 4-287 records parameters

Parameter	Type	Description
begin_time	String	Start time.
end_time	String	End time.
description	Object	Description.
instance_id	String	Instance ID.
version	String	Version number.
current_used	Boolean	Currently used parameter.
status	String	Status.
deploy_type	String	Deployment type. - OneBatchRelease - RollingRelease - GrayRelease

Parameter	Type	Description
jobs	Array of objects	Job list. See Table 4-288 .

Table 4-288 jobs

Parameter	Type	Description
sequence	Integer	Execution sequence.
deploy_type	String	Deployment type. • OneBatchRelease • RollingRelease • GrayRelease
job_id	String	Job ID.
job_info	Object	Job information. See Table 4-289 .

Table 4-289 job_info

Parameter	Type	Description
deploy_type	String	Deployment type.
source_url	String	Component source.
first_batch_weight	Integer	First execution weight.
first_batch_replica	Integer	Component instances deployed first time.
replica	Integer	Total instances.
remaining_batch	Integer	Remaining batch.

Example Request

None

Example Response

```
{
  "count": 10,
  "records": [
    {
      "begin_time": "string",
      "end_time": "string",
      "description": null,

```

```
{
  "instance_id": "string",
  "version": "string",
  "current_used": true,
  "status": "string",
  "deploy_type": "string",
  "jobs": [
    {
      "sequence": 0,
      "deploy_type": "string",
      "job_id": "string",
      "job_info": {
        "deploy_type": "string",
        "source_url": "string",
        "first_batch_weight": 0,
        "first_batch_replica": 0,
        "replica": 0,
        "remaining_batch": 0
      }
    }
  ]
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.3.10 Refreshing Component Information Based on the Component ID

Function

This API is used to refresh the component status and access mode based on the component ID.

URI

PUT /v3/{project_id}/cas/applications/{application_id}/components/
{component_id}/refresh

Table 4-290 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
application_id	Yes	String	Application ID. See Obtaining All Applications .
component_id	Yes	String	Component ID. See Obtaining All Components of an Application .

Request

Table 4-291 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

Table 4-292 Response parameters

Parameter	Type	Description
message	String	The refresh is successful. Fixed value: OK.

Example Request

None

Example Response

```
{
  "message": "OK"
}
```


Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.3.11 Obtaining Component Configuration Information

Function

This API is used to obtain information about the application, environment, and enterprise project to which all components belong.

URI

GET /v3/{project_id}/cas/components/filterOptions

Table 4-293 Path parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .

Request

Table 4-294 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

Table 4-295 Response parameters

Parameter	Type	Description
filter_options	Object	Component information parameter. See Table 4-296 .

Table 4-296 Component parameters

Parameter	Type	Description
applications	Array of objects	Application information. See Table 4-297 .
enterprise_projects	Array of String	Enterprise project information, which is the ID list of enterprise projects used by the component.
environments	Array of objects	Environment information. See Table 4-298 .

Table 4-297 Application information

Parameter	Type	Description
id	String	Application ID.
label	String	Application name.

Table 4-298 Environment information

Parameter	Type	Description
id	String	Environment ID.
label	String	Environment name.

Example Request

None

Example Response

```
{  
  "filter_options": {
```

```
{
  "environments": [
    {
      "label": "env-echotest-nodel",
      "id": "82005bc2-35d3-4fe1-87a8-4a1e1686f261"
    }
  ],
  "applications": [
    {
      "label": "wuzhiheng",
      "id": "f190e324-6dd8-4c97-ba21-53deafab7395"
    }
  ],
  "enterprise_projects": [
    "0"
  ]
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.3.12 Creating a Component Based on a Component Template

Function

This API is used to create a component based on a component template.

URI

POST /v3/{project_id}/cas/components

Table 4-299 Path parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .

Request

Table 4-300 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Table 4-301 Request body parameters

Parameter	Mandatory	Type	Description
environment_id	No	String	Environment ID. See Obtaining All Environments .
application_id	No	String	Application ID. See Obtaining All Applications .
template_source	No	Object	Source of the component template package. See Table 4-302 .
input_values	No	Map<String, String>	Template package input information.

Table 4-302 template_source

Parameter	Mandatory	Type	Description
storage	Yes	String	Storage mode of the component template package. - devcloud: CodeArts release repository - obs - http: custom file address

Parameter	Mandatory	Type	Description
url	Yes	String	Address of the component template package. This parameter is valid only when the component template package storage mode is http .
http_username	No	String	Username. This parameter is valid only when the component template package storage mode is http and the security authentication mode is username and password authentication.
http_password	No	String	Password. This parameter is valid only when the component template package storage mode is http and the security authentication mode is username and password authentication.
http_headers	No	Array of objects	Request header. See Table 4-303 . This parameter is valid only when the component template package storage mode is http and the security authentication mode is custom header authentication.

Table 4-303 http_headers

Parameter	Mandatory	Type	Description
key	Yes	String	Request key.
value	Yes	String	Request value.

Response

Table 4-304 Response parameters

Parameter	Type	Description
component_id	Integer	Component ID.
job_id	Integer	Job ID.

Example Request

Create a container-deployed component based on the component template.

```
{
  "environment_id": "2393eccc-69bc-40a9-9b6c-996a9259618d",
  "application_id": "dda56ba8-5295-3d0d-b298-72cb1da3f515",
  "template_source": {
    "storage": "obs",
    "url": "obs://apitest/demo-app-package.zip",
    "http_username": null,
    "http_password": null,
    "http_headers": []
  },
  "input_values": {
    "config_name": "demo-group/application.yaml",
    "namespace": "default",
    "dep_name": "test-demo",
    "svc_port": 8080,
    "value": "World",
    "cce_resource": "ef1b6474-906c-11ef-a773-0255ac1000a3",
    "elb_resource": "053dcc46-d3fd-4796-bba1-b73650ec0906",
    "image": "swr.ap-southeast-1.myhuaweicloud.com/ss/demo-app:latest"
  }
}
```

Example Response

```
{
  "job_id": "string",
  "component_id": "string"
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.3.13 Parsing a Component Template Package

Function

This API is used to parse a component template package.

URI

POST /v3/{project_id}/cas/components/parse-template-package

Table 4-305 Path parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .

Request

Table 4-306 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Table 4-307 Request body parameters

Parameter	Mandatory	Type	Description
template_source	No	Object	Source of the component template package. See Table 4-308 .
component_id	No	String	Component ID. • Mandatory when the component is upgraded by Modifying Component Information Based on the Component ID. See Obtaining All Components of an Application. • Not required when Creating a Component Based on a Component Template.

Table 4-308 template_source

Parameter	Mandatory	Type	Description
storage	Yes	String	Storage mode of the component template package. • devcloud: CodeArts release repository • obs • http: custom file address
url	Yes	String	Address of the component template package. This parameter is valid only when the component template package storage mode is http .
http_username	No	String	Username. This parameter is valid only when the component template package storage mode is http and the security authentication mode is username and password authentication.
http_password	No	String	Password. This parameter is valid only when the component template package storage mode is http and the security authentication mode is username and password authentication.
http_headers	No	Array of objects	Request header. See Table 4-309 . This parameter is valid only when the component template package storage mode is http and the security authentication mode is custom header authentication.

Table 4-309 http_headers

Parameter	Mandatory	Type	Description
key	Yes	String	Request key.
value	Yes	String	Request value.

Response

Table 4-310 Response parameters

Parameter	Type	Description
template	Map<String, String>	Template package file information.
variables	Map<String, Object>	Template package variable declaration information. - key - value: see Table 4-311.
input_values	Map<String, String>	Template package input information.

Table 4-311 variables

Parameter	Type	Description
type	String	Variable type.
default	String	Default variable value.
description	String	Variable description.

Example Request

Parse the **demo-app-package.zip** component template package stored in the **obs://apitest/** directory of the OBS bucket.

```
{
  "template_source": {
    "storage": "obs",
    "url": "obs://apitest/demo-app-package.zip",
    "http_username": null,
    "http_password": null,
    "http_headers": []
  }
}
```

Example Response

```
{
  "template": {
    "comp_demo/service-account.yaml": "apiVersion: v1\nkind: ServiceAccount\nmetadata:\n  name: ss-demo-app-account\n  namespace: ${var.namespace}",
    "comp_demo/role.yaml": "apiVersion: rbac.authorization.k8s.io/v1\nkind: Role\nmetadata:\n  name: ss-demo-app-role\n  namespace: ${var.namespace}\nrules:\n  - apiGroups: [ \"\" ]\n  resources: [ \"pods\" ]\n  verbs: [ \"get\", \"list\", \"watch\" ]",
    "values.yaml": "namespace: demo\ndep_name: demo-dep\nsvc_port: 8080\nvalue: World\nimage: swr.ap-southeast-1.myhuaweicloud.com/ss/demo-app:latest\nconfig_name: demo-group/application-wqd.yaml",
    "comp_demo/rolebinding.yaml": "apiVersion: rbac.authorization.k8s.io/v1\nkind: RoleBinding\nmetadata:\n  name: ss-demo-app-rolebinding\n  namespace: ${var.namespace}\nroleRef:\n  apiGroup:"
```

```
rbac.authorization.k8s.io\n kind: Role\n name: ss-demo-app-role\nsubjects:\n - kind: ServiceAccount\n name: ss-demo-app-account\n namespace: ${var.namespace}",  
  "comp_demo/service.yaml": "apiVersion: v1\nkind: Service\nmetadata:\n name: demo-app-service\n namespace: ${var.namespace}\nspec:\n selector:\n app: ${var.dep_name}\n ports:\n - name: cce-  
service-0\n targetPort: ${var.svc_port}\n port: ${var.svc_port}\n protocol: TCP\n type: ClusterIP\n",  
  "comp_demo/ss-component.yaml": "kind: ServiceStageComponent\nspec:\n name: ${var.dep_name}\n refer_resources:\n - type: cce\n id: ${var.cce_resource}\n - type: elb\n id: ${var.elb_resource}\n external_accessed:\n - address: www.demo.com\n forward_port: 8080\n id: ${var.elb_resource}\n protocol: HTTP\n type: SPECIFIED\n configs:\n - config_name: ${var.config_name}\n mount_path: /home/demo/application-wqd.yaml\n container_name: container-1",  
  "comp_demo/ss-config.yaml": "kind: ServiceStageConfig\nspec:\n name: ${var.config_name}\n description: desc\n sensitive: true\n content: |\n server:\n port: ${var.svc_port}\n demo:\n value: ${var.value}",  
  "comp_demo/secret.yaml": "apiVersion: v1\nkind: Secret\nmetadata:\n name: ss-demo-app-secret\n namespace: ${var.namespace}\ntype: Opaque\n data:\n SECRET_KEY: dmFsdWU=",  
  "comp_demo/hpa.yaml": "kind: HorizontalPodAutoscaler\napiVersion: autoscaling/v2\nmetadata:\n name: ss-demo-app-hpa\n namespace: ${var.namespace}\nspec:\n scaleTargetRef:\n kind: Deployment\n name: ${var.dep_name}\n apiVersion: apps/v1\n minReplicas: 1\n maxReplicas: 10\n metrics:\n - type: Resource\n resource:\n name: cpu\n target:\n type: Utilization\n averageUtilization: 70",  
  "comp_demo/deployment.yaml": "apiVersion: apps/v1\nkind: Deployment\nmetadata:\n name: ${var.dep_name}\n namespace: ${var.namespace}\nspec:\n selector:\n matchLabels: {}\n template:\n metadata:\n labels: {}\n spec:\n containers:\n - name: container-1\n image: ${var.image}\n imagePullPolicy: IfNotPresent\n envFrom:\n - configMapRef:\n name: ss-demo-app-configmap\n - secretRef:\n name: ss-demo-app-secret\n resources:\n requests:\n cpu: 100m\n memory: 256Mi\n limits:\n cpu: 100m\n memory: 256Mi\n serviceAccountName: ss-demo-app-account\n imagePullSecrets:\n - name: default-secret\n terminationGracePeriodSeconds: 30\n dnsPolicy: ClusterFirst\n replicas: 2\n revisionHistoryLimit: 10\n strategy:\n type: RollingUpdate\n rollingUpdate:\n maxUnavailable: 25%\n maxSurge: 25%\n progressDeadlineSeconds: 600\n",  
  "variables.yaml": "variable:\n config_name:\n type: string\n default: demo-group/application-wqd.yaml\n description: configuration file name\n namespace:\n type: string\n default: demo\n description: namespace\n dep_name:\n type: string\n default: demo-dep\n description: workload name\n svc_port:\n type: integer\n default: 8080\n description: service port number\n value:\n type: string\n default: World\n description: Hello interface response data\n cce_resource:\n type: cce\n description: CCE resource\n elb_resource:\n type: elb\n description: ELB resource\n image:\n type: string\n description: workload image",  
  "comp_demo/configmap.yaml": "apiVersion: v1\nkind: ConfigMap\nmetadata:\n name: ss-demo-app-configmap\n namespace: ${var.namespace}\ndata:\n CONFIG_KEY: \"value\"",  
  "spec.yaml": "model_version: v1.0.0\ncomponent:\n component_dir: comp_demo\n\nk8s_resources:\n order_type: CUSTOM\n custom_orders:\n - ServiceAccount\n - Role\n - RoleBinding\n - ConfigMap\n - Secret\n - Deployment\n - HorizontalPodAutoscaler\n - Service\n - Ingress",  
},  
"variable": {  
  "config_name": {  
    "type": "string",  
    "default": "demo-group/application-wqd.yaml",  
    "description": "configuration file name"  
  },  
  "namespace": {  
    "type": "string",  
    "default": "demo",  
    "description": "namespace"  
  },  
  "dep_name": {  
    "type": "string",  
    "default": "demo-dep",  
    "description": "workload name"  
  },  
  "svc_port": {  
    "type": "integer",  
    "default": 8080,  
    "description": "service port number"  
  },  
  "value": {  
    "type": "string",  
    "default": "World",  
    "description": "Hello interface response data"  
  }  
}
```

```
    },
    "cce_resource": {
      "type": "cce",
      "default": null,
      "description": "CCE resource"
    },
    "elb_resource": {
      "type": "elb",
      "default": null,
      "description": "ELB resource"
    },
    "image": {
      "type": "string",
      "default": null,
      "description": "workload image"
    }
  },
  "input_values": {
    "config_name": "demo-group/application.yaml",
    "image": "swr.ap-southeast-1.myhuaweicloud.com/ss/demo-app:latest",
    "svc_port": 8080,
    "cce_resource": null,
    "namespace": "demo",
    "dep_name": "demo-dep",
    "value": "World",
    "elb_resource": null
  }
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.4 Deployment Job

4.4.1 Obtaining Job Details

Function

This API is used to obtain job details.

URI

GET /v3/{project_id}/cas/jobs/{job_id}

Table 4-312 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
job_id	Yes	String	Deployment job ID, which is job_id in the response parameter in Creating an Application Component , Importing a CCE Workload to an Application to Create a Component , Modifying Component Information Based on the Component ID , and Delivering a Component Task Based on the Component ID .

Table 4-313 Query parameters

Parameter	Mandatory	Type	Description
limit	No	Integer	Number of specified records.
offset	No	Integer	Offset.
desc	No	String	Whether query results are displayed in descending order. Value: true (descending order) or false (ascending order).

Request

Table 4-314 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

Table 4-315 Response parameters

Parameter	Type	Description
task_count	Integer	Number of tasks.
job	Object	Job parameters. See Table 4-316 .
tasks	Array<Object>	Task parameters. See Table 4-317 .

Table 4-316 job

Parameter	Type	Description
execution_statuses	String	Execution status. • SUBMITTED • SCHEDULED • RUNNING • FAILED • SUCCEEDED • CANCELLED • PARTIALLY_FAILED • ROLLBACKING • ROLLBACKED • UNKNOWN
job_id	String	Job ID.
job_type	String	Type. • PROVISION • DEPROVISION • DAILY_OPERATION • SYSTEM_INNER • UNKNOWN

Table 4-317 task

Parameter	Type	Description
created_at	String	Creation time.
messages	String	Message.

Parameter	Type	Description
task_id	String	Task ID.
task_index	Integer	Task index.
task_name	String	Task name.
task_status	String	Task status. • SUBMITTED • SCHEDULED • RUNNING • FAILED • SUCCEEDED • CANCELLED • PARTIALLY_FAILED • SKIPPED • ROLLBACKING • ROLLBACKED • UNKNOWN
update_at	String	Update time.

Example Request

None

Example Response

```
{
  "job": {
    "execution_status": "SUCCEEDED",
    "job_id": "JOB17cb1778-07b1-4c8f-a183-c86cffbe8f9e",
    "job_type": "PROVISION"
  },
  "tasks": [
    {
      "task_id": "Task-af75dd60-0b04-4f4b-b669-0bad38fcdd97",
      "task_index": 4,
      "task_name": "VMAPP_CONTEXT_PREPARE_01",
      "task_status": "SUCCEEDED",
      "created_at": "2024-06-06T15:55:31.115Z",
      "update_at": "2024-06-06T15:55:31.136Z",
      "messages": "{}"
    },
    {
      "task_id": "Task-50110d26-3973-49e1-95d3-d46755e865af",
      "task_index": 5,
      "task_name": "VMAPP_UPGRADE_002",
      "task_status": "SUCCEEDED",
      "created_at": "2024-06-06T15:55:31.136Z",
      "update_at": "2024-06-06T15:55:31.36Z",
      "messages": "{}"
    }
  ]
},
```

```
"task_count": 2  
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.5 Configuration Management

4.5.1 Creating a Configuration Group

Function

This API is used to create a configuration group to manage configuration files.

URI

POST /v3/{project_id}/cas/config-groups

Table 4-318 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .

Request

Table 4-319 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Table 4-320 Request body parameter

Parameter	Mandatory	Type	Description
name	Yes	String	Configuration group name. The value contains 2 to 64 characters consisting of letters, digits, hyphens (-), and underscores (_). It starts with a letter and ends with a letter or digit.
description	No	String	Configuration group description. The value contains a maximum of 256 characters.
enterprise_project_id	No	String	Enterprise project ID. If this parameter is not set, the default enterprise project is default and the ID is 0 .
labels	No	Array of objects	Label. You can use TMS to filter resources by labels. See Table 4-321 .

Table 4-321 labels

Parameter	Mandatory	Type	Description
key	Yes	String	Label name.
value	Yes	String	Label value.

Response

Table 4-322 Response parameters

Parameter	Type	Description
id	String	Configuration group ID.
name	String	Configuration group name.
description	String	Configuration group description.
creator	String	Creator.
project_id	String	Project ID.
create_time	Integer	Creation time.

Example Request

Create a configuration group named **test-group**.

```
{
  "name": "test-group",
  "description": ""
}
```

Example Response

```
{
  "id": "3f10d752-80e9-4d80-ad74-454fd59e23a7",
  "project_id": "b6097e34f6694a1fa2a49052acc9718d",
  "name": "test-group",
  "description": null,
  "create_time": 1744698133769,
  "creator": "ss-test",
  "labels": [],
  "enterprise_project_id": "0",
  "enterprise_project_name": null
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.5.2 Obtaining a Configuration Group

Function

This API is used to obtain a created configuration group.

URI

GET /v3/{project_id}/cas/config-groups

Table 4-323 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .

Table 4-324 Query parameters

Parameter	Mandatory	Type	Description
name	No	String	Configuration group name.
limit	No	integer	Number of records to be queried. Value range: 0–100.
offset	No	integer	Offset.
order_by	No	String	Sorting field. By default, query results are sorted by creation time. Enumerated values: - create_time - name - update_time
order	No	String	Sorting order. - desc (default) - asc

Request

Table 4-325 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

Table 4-326 Response parameters

Parameter	Type	Description
count	Integer	Total number of configuration groups.
config_groups	Array of objects	Group list. See Table 4-327 .

Table 4-327 config_groups

Parameter	Type	Description
id	String	Configuration group ID.
project_id	String	Project ID.
name	String	Configuration group name.
description	String	Configuration group description.
creator	String	Creator.
create_time	Integer	Creation time.
enterprise_project_id	String	Enterprise project ID. If this parameter is not set, the default enterprise project is default and the ID is 0 .
labels	Array of objects	Label. You can use TMS to filter resources by labels. See Table 4-328 .

Table 4-328 labels

Parameter	Type	Description
key	String	Label name.
value	String	Label value.

Example Request

None

Example Response

```
{
  "config_groups": [
    {
      "id": "3f10d752-80e9-4d80-ad74-454fd59e23a7",
      "project_id": "b6097e34f6694a1fa2a49052acc9718d",
      "name": "test-group",
      "description": null,
      "create_time": 1744698133769,
      "creator": "ss-test",
      "config_count": 0,
      "labels": [],
      "enterprise_project_id": "0",
      "enterprise_project_name": "default"
    }
  ],
  "total": 1,
  "count": 1
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.5.3 Creating a Configuration File

Function

This API is used to create a configuration file.

URI

POST /v3/{project_id}/cas/configs

Table 4-329 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .

Request

Table 4-330 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Table 4-331 Request body parameter

Parameter	Mandatory	Type	Description
config_group_id	Yes	String	Configuration group ID.
name	Yes	String	Configuration file name. The value contains 2 to 64 characters consisting of letters, digits, hyphens (-), and underscores (_). It starts with a letter and ends with a letter or digit.
content	Yes	String	Configuration file content.
description	No	String	Configuration file description. The value can contain up to 128 characters.

Parameter	Mandatory	Type	Description
type	Yes	String	Configuration file format. • yaml • properties
sensitive	Yes	Boolean	Whether to encrypt the configuration file. • true: Encrypt. • false: Not encrypt.
config_encryption	No	Object	Set the encryption key in the configuration file. See Table 4-332 .

Table 4-332 config_encryption

Parameter	Mandatory	Type	Description
key_id	Yes	String	User key ID in DEW.

Response

Table 4-333 Response parameters

Parameter	Type	Description
id	String	Configuration file ID.
name	String	Configuration file name.
description	String	Configuration description.
creator	String	Creator.
project_id	String	Project ID.
type	String	Configuration file format. • yaml • properties
sensitive	Boolean	Whether to encrypt the configuration file. • true: Encrypt. • false: Not encrypt.
components	Array of objects	Component to which the configuration file is bound. See Table 4-334 .

Parameter	Type	Description
create_time	Integer	Creation time.
update_time	Integer	Update time.
version	String	Configuration file version.
content	String	Configuration file content.
config_group_id	String	Configuration group ID.

Table 4-334 components

Parameter	Type	Description
application_id	String	Application ID.
component_id	String	Component ID.
component_name	String	Component name.
environment_id	String	Environment ID.

Example Request

Create a configuration file named **test-config**. Do not encrypt the content and set the file format to **properties**.

```
{
  "config_group_id": "f3e6d5b4-4d7d-4009-b806-22cfed2ac6ea",
  "name": "test-config",
  "description": "",
  "type": "properties",
  "content": "test-key = test-value",
  "sensitive": false
}
```

Example Response

```
{
  "id": "d5821100-9a82-4d1c-9468-fc6b2b9ccdad",
  "config_group_id": "f3e6d5b4-4d7d-4009-b806-22cfed2ac6ea",
  "name": "test-config",
  "description": "",
  "type": "properties",
  "content": "testkey = testvalue",
  "version": "v1",
  "project_id": "2cfeefc3e4c54a5aa7548b8350e638d7",
  "sensitive": false,
  "create_time": 1717485580152,
  "update_time": 1717485580152,
  "components": [],
  "creator": "ss-test"
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.5.4 Importing a Configuration File

Function

This API is used to import a configuration file in .zip format.

URI

POST /v3/{project_id}/cas/configs/import

Table 4-335 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .

Request

Table 4-336 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type or format: multipart/form-data

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Table 4-337 Request body parameter

Parameter	Mandatory	Type	Description
upload_file	Yes	File	The configuration file to be imported must be compressed into a .zip package, and the package size cannot exceed 500 KB. The package to be imported can contain up to 500 configuration files. The folder name in the compressed package is the configuration group name, and the file name is the configuration name. The file can be in the YAML or properties format.
repetition_policy	No	String	Policy for processing the configuration file with the same name as the imported configuration file: - covered: Replace the existing configuration file with the imported configuration file. - skip: Skip the import of the configuration file with the same name.

Response

Table 4-338 Response parameters

Parameter	Type	Description
covered_num	Integer	Number of configuration files that are overwritten.
skip_num	Integer	Number of configuration files that are skipped during file import.

Parameter	Type	Description
total_num	Integer	Number of configuration files that are imported.

Example Request

Import the configuration file whose absolute path is **/root/configfile.zip**.

```
{
  "upload_file" : "/root/configfile.zip"
}
```

Example Response

```
{
  "covered_num" : 0,
  "total_num" : 1
}
```

Status Code

Status Code	Description
204	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.5.5 Obtaining Configuration File Information

Function

This API is used to obtain the configuration file information.

URI

GET /v3/{project_id}/cas/configs

Table 4-339 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .

Table 4-340 Query parameters

Parameter	Mandatory	Type	Description
name	No	String	Configuration file name.
config_group_id	No	String	Configuration group ID.
component_id	No	String	Component ID.
limit	No	integer	Number of records to be queried. Value range: 0–100.
offset	No	integer	Offset.
order_by	No	String	Sorting field. By default, query results are sorted by creation time. • create_time • name • update_time
order	No	String	Sorting order. • desc (default) • asc

Request

Table 4-341 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

Table 4-342 Response parameters

Parameter	Type	Description
count	Integer	Total number of configuration files.
configs	Array of objects	Configuration file list. See Table 4-343 .

Table 4-343 configs

Parameter	Type	Description
config_group_id	String	Configuration group ID.
id	String	Configuration file ID.
name	String	Configuration file name.
description	String	Configuration file description.
creator	String	Creator.
type	String	Configuration file type. • yaml • properties
content	String	Configuration file content.
version	String	Configuration file version.
project_id	String	Project ID.
sensitive	Boolean	Whether to encrypt the configuration file. • true: Encrypt. • false: Not encrypt.

Parameter	Type	Description
components	Array of objects	List of the components to which the configuration file is bound. See Table 4-344 .
create_time	Integer	Creation time.
update_time	Integer	Update time.

Table 4-344 components

Parameter	Type	Description
application_id	String	Application ID.
component_id	String	Component ID.
component_name	String	Component name.
environment_id	String	Environment ID.

Example Request

None

Example Response

```
{
  "configs": [{
    "id": "f5a52d93-1587-45cf-b8fe-5c7f204b98d1",
    "config_group_id": "4808b50d-d7f3-401b-9175-968cb4f9892b",
    "name": "test-config1",
    "description": null,
    "type": "properties",
    "content": "testKey = testValue",
    "version": "v1",
    "project_id": "2cfeefc3e4c54a5aa7548b8350e638d7",
    "sensitive": false,
    "create_time": 1717486892611,
    "update_time": 1717486892611,
    "components": [{
      "application_id": "8a3be48a-6401-4e8c-bbce-033745252227",
      "component_id": "4b611997-4acd-4538-a06b-32fd2475e057",
      "component_name": "component-test1",
      "environment_id": "d2d7180c-7739-4cbb-bd8e-0911ba31cf51"
    }],
    "creator": "ss-test"
  }],
  {
    "id": "2cb1897e-c23b-48a5-993c-dc7ad3c552a0",
    "config_group_id": "f3e6d5b4-4d7d-4009-b806-22cfed2ac6ea",
    "name": "test-config2",
    "description": null,
    "type": "properties",
    "content": "testKey = testValue",
    "version": "v1",
    "project_id": "2cfeefc3e4c54a5aa7548b8350e638d7",
    "sensitive": false,
```

```
{
  "create_time": 1717486590467,
  "update_time": 1717486590467,
  "components": [],
  "creator": "ss-test"
},
"count": 2
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.5.6 Obtaining Configuration File Information Based on the Configuration File ID

Function

This API is used to obtain the details about a configuration file based on the configuration file ID.

URI

GET /v3/{project_id}/cas/configs/{config_id}

Table 4-345 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
config_id	Yes	String	Configuration file ID. See Obtaining Configuration File Information .

Request

Table 4-346 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

Table 4-347 Response parameters

Parameter	Type	Description
config_group_id	String	Configuration group ID.
id	String	Configuration file ID.
name	String	Configuration file name.
description	String	Configuration file description.
creator	String	Creator.
type	String	Configuration file type. • yaml • properties
content	String	Configuration file content.
version	String	Configuration file version.
project_id	String	Project ID.
sensitive	Boolean	Whether to encrypt the configuration file. • true: Encrypt. • false: Not encrypt.
components	Array of objects	List of the components to which the configuration file is bound. See Table 4-348 .
create_time	Integer	Creation time.

Parameter	Type	Description
update_time	Integer	Update time.
enterprise_project_id	String	If this parameter is not set, the default enterprise project is default and the ID is 0 .
config_encryption	Object	Set the encryption key in the configuration file. See Table 4-349 .

Table 4-348 components

Parameter	Type	Description
application_id	String	Application ID.
component_id	String	Component ID.
component_name	String	Component name.
environment_id	String	Environment ID.

Table 4-349 config_encryption

Parameter	Type	Description
key_id	String	User key ID in DEW.

Example Request

None

Example Response

```
{
  "id": "f5a52d93-1587-45cf-b8fe-5c7f204b98d1",
  "config_group_id": "4808b50d-d7f3-401b-9175-968cb4f9892b",
  "name": "test-config1",
  "description": "",
  "type": "properties",
  "content": "testKey = testValue",
  "version": "v1",
  "project_id": "2cfeefc3e4c54a5aa7548b8350e638d7",
  "sensitive": false,
  "create_time": 1717486892611,
  "update_time": 1717486892611,
  "components": [{
    "application_id": "8a3be48a-6401-4e8c-bbce-033745252227",
    "component_id": "4b611997-4acd-4538-a06b-32fd2475e057",
    "component_name": "component-test1",
    "environment_id": "d2d7180c-7739-4cbb-bd8e-0911ba31cf51"
  }],
  "creator": "ss-test"
}
```


Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.5.7 Deleting a Configuration File Based on the Configuration File ID

Function

This API is used to delete a configuration file based on the configuration file ID.

URI

DELETE /v3/{project_id}/cas/configs/{config_id}

Table 4-350 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
config_id	Yes	String	Configuration file ID. See Obtaining Configuration File Information .

Request

Table 4-351 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

None

Example Request

None

Example Response

None

Status Code

Status Code	Description
204	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.5.8 Modifying a Configuration File Based on the Configuration File ID

Function

This API is used to modify a configuration file based on the configuration file ID.

URI

PUT /v3/{project_id}/cas/configs/{config_id}

Table 4-352 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
config_id	Yes	String	Configuration file ID. See Obtaining Configuration File Information .

Request

Table 4-353 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Table 4-354 Request body parameter

Parameter	Mandatory	Type	Description
config_group_id	Yes	String	Configuration group ID.
content	Yes	String	Configuration file content.
type	Yes	String	Configuration file format.
description	No	String	Configuration file description.
sensitive	No	Boolean	Whether to encrypt the configuration file. - true: Encrypt. - false: Not encrypt.
config_encryption	No	Object	Set the encryption key in the configuration file. See Table 4-355 .

Table 4-355 config_encryption

Parameter	Mandatory	Type	Description
key_id	Yes	String	User key ID in DEW.

Response

Table 4-356 Response parameters

Parameter	Type	Description
config_group_id	String	Configuration group ID.
id	String	Configuration file ID.
name	String	Configuration file name.
description	String	Configuration file description.
creator	String	Creator.
type	String	Configuration file type. • yaml • properties
content	String	Configuration file content.
version	String	Configuration file version.
project_id	String	Project ID.
sensitive	Boolean	Whether to encrypt the configuration file. • true: Encrypt. • false: Not encrypt.
components	Array of objects	List of the components to which the configuration file is bound. See Table 4-357 .
create_time	Integer	Creation time.
update_time	Integer	Update time.
enterprise_project_id	String	If this parameter is not set, the default enterprise project is default and the ID is 0 .
config_encryption	Object	Set the encryption key in the configuration file. See Table 4-358 .

Table 4-357 components

Parameter	Type	Description
application_id	String	Application ID.
component_id	String	Component ID.
component_name	String	Component name.
environment_id	String	Environment ID.

Table 4-358 config_encryption

Parameter	Type	Description
key_id	String	User key ID in DEW.

Example Request

Change the group ID of the configuration file to **4808b50d-d7f3-401b-9175-968cb4f9892b**, type to **properties**, and content to **testKey = testModify**, and do not encrypt the file content.

```
{
  "config_group_id": "4808b50d-d7f3-401b-9175-968cb4f9892b",
  "description": "",
  "type": "properties",
  "content": "testKey = testModify",
  "sensitive": false
}
```

Example Response

```
{
  "id": "f5a52d93-1587-45cf-b8fe-5c7f204b98d1",
  "config_group_id": "4808b50d-d7f3-401b-9175-968cb4f9892b",
  "name": "test-config1",
  "description": "",
  "type": "properties",
  "content": "testKey = testModify",
  "version": "v2",
  "project_id": "2cfeefc3e4c54a5aa7548b8350e638d7",
  "sensitive": false,
  "create_time": 1717486892611,
  "update_time": 1717491574202,
  "components": [{
    "application_id": "8a3be48a-6401-4e8c-bbce-033745252227",
    "component_id": "4b611997-4acd-4538-a06b-32fd2475e057",
    "component_name": "component-test1",
    "environment_id": "d2d7180c-7739-4cbb-bd8e-0911ba31cf51"
  }],
  "creator": "ss-test"
}
```

Status Code

Status Code	Description
204	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.5.9 Obtaining the Configuration File Record Based on the Configuration File ID

Function

This API is used to obtain the configuration file record based on the configuration file ID.

URI

GET /v3/{project_id}/cas/configs/{config_id}/histories

Table 4-359 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
config_id	Yes	String	Configuration file ID. See Obtaining Configuration File Information .

Table 4-360 Query parameters

Parameter	Mandatory	Type	Description
limit	No	integer	Number of records to be queried. Value range: 0–100.
offset	No	integer	Offset.

Parameter	Mandatory	Type	Description
order_by	No	String	Sorting field. By default, query results are sorted by creation time. Enumerated values: • create_time • name • update_time
order	No	String	Sorting order. • desc (default) • asc

Request

Table 4-361 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

Table 4-362 Response parameters

Parameter	Type	Description
count	Integer	Total number of configuration file records.
histories	Array of objects	Configuration file record list. See Table 4-363 .

Table 4-363 histories

Parameter	Type	Description
config_history_id	String	Configuration file record ID.
config_id	String	Configuration file ID.
content	String	Configuration file record content.
version	String	Configuration file record version.
create_time	Integer	Creation time.
components	Array of objects	Component to which the configuration file is bound. See Table 4-364 .
creator	String	Creator.
group_id	String	Configuration group ID.
type	String	Configuration file type.
config_name	String	Configuration file name.
description	String	Configuration file description.
sensitive	Boolean	Whether to encrypt the configuration file. • true: Encrypt. • false: Not encrypt.

Table 4-364 components

Parameter	Type	Description
application_id	String	Application ID.
component_id	String	Component ID.
component_name	String	Component name.
environment_id	String	Environment ID.

Example Request

None

Example Response

```
{
  "count": 2,
  "histories": [{
    "config_history_id": "e9c52a5f-0f7c-48a2-b89c-29dd349d1a35",
```



```
{
  "config_id": "f5a52d93-1587-45cf-b8fe-5c7f204b98d1",
  "content": "testKey = valueModify",
  "type": "properties",
  "version": "v2",
  "create_time": 1717491574202,
  "components": [{
    "application_id": "8a3be48a-6401-4e8c-bbce-033745252227",
    "component_id": "4b611997-4acd-4538-a06b-32fd2475e057",
    "component_name": "component-test",
    "environment_id": "d2d7180c-7739-4cbb-bd8e-0911ba31cf51"
  }],
  "creator": "ss-test",
  "group_id": "4808b50d-d7f3-401b-9175-968cb4f9892b",
  "config_name": "test-config1",
  "description": "",
  "sensitive": false
},
{
  "config_history_id": "adacc5dc-a497-4023-b292-3b3cf37347f8",
  "config_id": "f5a52d93-1587-45cf-b8fe-5c7f204b98d1",
  "content": "testKey = testValue",
  "type": "properties",
  "version": "v1",
  "create_time": 1717491445668,
  "components": [{
    "application_id": "8a3be48a-6401-4e8c-bbce-033745252227",
    "component_id": "4b611997-4acd-4538-a06b-32fd2475e057",
    "component_name": "component-test",
    "environment_id": "d2d7180c-7739-4cbb-bd8e-0911ba31cf51"
  }],
  "creator": "ss-test",
  "group_id": "4808b50d-d7f3-401b-9175-968cb4f9892b",
  "config_name": "test-config1",
  "description": "",
  "sensitive": false
}
}
```

Status Code

Status Code	Description
204	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.5.10 Obtaining the Configuration File Record Information Based on the Record ID

Function

This API is used to obtain the configuration file record information based on the record ID.

URI

GET /v3/{project_id}/cas//configs/{config_id}/histories/{config_history_id}

Table 4-365 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
config_id	Yes	String	Configuration file ID. See Obtaining Configuration File Information .
config_history_id	Yes	String	Configuration file record ID. See Obtaining the Configuration File Record Based on the Configuration File ID .

Request

Table 4-366 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

Table 4-367 Response parameters

Parameter	Type	Description
config_history_id	String	Configuration file record ID.
config_id	String	Configuration file ID.
content	String	Configuration file record content.

Parameter	Type	Description
version	String	Configuration file record version.
type	String	Configuration file.
create_time	Integer	Creation time.
components	Array of objects	List of the components to which the configuration file is bound. See Table 4-368 .
creator	String	Creator.
group_id	String	Configuration group ID.
config_name	String	Configuration file name.
description	String	Configuration file description.
sensitive	Boolean	Whether to encrypt the configuration file. • true: Encrypt. • false: Not encrypt.

Table 4-368 components

Parameter	Type	Description
application_id	String	Application ID.
component_id	String	Component ID.
component_name	String	Component name.
environment_id	String	Environment ID.

Example Request

None

Example Response

```
{
  "config_history_id": "139e2161-ddf3-4a90-832f-a3a5a5fa01c2",
  "config_id": "4b4cd418-3553-453a-bd2b-660bfcee7f5a",
  "content": "testkey = testValue",
  "version": "v1",
  "create_time": 1717498251010,
  "components": [],
  "creator": "ss-test",
  "group_id": "3db71eff-c08a-4da7-bd44-03791f33bbc7",
  "config_name": "test-config",
  "description": "",
  "sensitive": true,
  "type": "properties"
}
```

Status Code

Status Code	Description
204	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.5.11 Deleting a Configuration File Record Based on the Record ID

Function

This API is used to delete a configuration file record based on the record ID.

URI

DELETE /v3/{project_id}/cas/configs/{config_id}/histories/{config_history_id}

Table 4-369 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
config_id	Yes	String	Configuration file ID. See Obtaining Configuration File Information .
config_history_id	Yes	String	Configuration file record ID. See Obtaining the Configuration File Record Based on the Configuration File ID .

Request

Table 4-370 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

None

Example Request

None

Example Response

None

Status Code

Status Code	Description
204	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.5.12 Obtaining Configuration Group Details Based on the Group ID

Function

This API is used to obtain the configuration group details based on the group ID.

URI

GET /v3/{project_id}/cas/config-groups/{config_group_id}

Table 4-371 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
config_group_id	Yes	String	Configuration group ID. See Obtaining a Configuration Group .

Request

Table 4-372 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

Table 4-373 Response parameters

Parameter	Type	Description
id	String	Configuration group ID.

Parameter	Type	Description
project_id	String	Project ID.
name	String	Configuration group name.
description	String	Configuration group description.
creator	String	Creator.
create_time	Integer	Creation time.

Example Request

None

Example Response

```
{
  "id": "0f5722ea-ee97-42bf-8583-404caa0a46e5",
  "project_id": "2cfeefc3e4c54a5aa7548b8350e638d7",
  "name": "test1",
  "description": "",
  "create_time": 1717482692680,
  "creator": "ss-test"
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.5.13 Deleting a Configuration Group Based on the Group ID

Function

This API is used to delete a configuration group based on the group ID.

URI

DELETE /v3/{project_id}/cas/config-groups/{config_group_id}

Table 4-374 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
config_group_id	Yes	String	Configuration group ID. See Obtaining a Configuration Group .

Request

Table 4-375 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

None

Example Request

None

Example Response

None

Status Code

Status Code	Description
204	OK
400	Bad Request
404	Not Found

Status Code	Description
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.6 Add-on Management

4.6.1 Creating an Add-on

Function

This API is used to create an add-on.

URI

POST /v3/{project_id}/cas/environments/{environment_id}/addons

Table 4-376 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
environment_id	Yes	String	Environment ID. See Obtaining All Environments .

Request

Table 4-377 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Table 4-378 Request body parameters

Parameter	Mandatory	Type	Description
cce_id	Yes	String	ID of the CCE cluster bound to the environment. See Querying a Managed Resource Based on the Environment ID .
cse_id	No	String	ID of the microservice engine bound to the environment. See Querying a Managed Resource Based on the Environment ID . This parameter is mandatory when name is set to sermantInjector .
name	Yes	String	Add-on name. sermantInjector: Sermant Injector add-on. It is used to automatically mount Sermant Agent and connect to ServiceComb engines with security authentication disabled through Sermant Agent. You can access the engine and use functions such as application registry and discovery, full-link dark launch, graceful startup and shutdown, and label routing without modifying application code. However, microservice governance is not supported.

Response

Table 4-379 Response parameters

Parameter	Type	Description
job_id	String	Job ID.
id	String	Add-on ID.

Example Request

Create a Sermant Injector add-on and manage the microservice engine whose ID is b4e49ad5-5142-437d-8a6b-1de66b1af0db and the CCE cluster whose ID is 02f9a523-cefe-11ee-afc1-0255ac100045 for the add-on.

```
{
  "name": "sermantInjector",
  "cse_id": "b4e49ad5-5142-437d-8a6b-1de66b1af0db",
  "cce_id": "02f9a523-cefe-11ee-afc1-0255ac100045"
}
```

Example Response

```
{
  "job_id": "JOB7c165545-a415-4cc6-9b11-d7045aa24440",
  "id": "a2cdf0e2-8495-4db5-82f7-a23b97b57787"
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.6.2 Querying the Add-on List

Function

This API is used to query the add-on list in an environment.

URI

GET /v3/{project_id}/cas/environments/{environment_id}/addons

Table 4-380 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
environment_id	Yes	String	Environment ID, which is the ID in the response parameter of the API for Creating an Environment .

Request

Table 4-381 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

Table 4-382 Response parameters

Parameter	Type	Description
addons	Arrays of objects	Add-on list. See Table 4-383 .

Table 4-383 addons

Parameter	Type	Description
cce_id	String	ID of the CCE cluster bound to the add-on.
cse_id	String	ID of the microservice engine bound to the add-on.
environment_id	String	ID of the add-on environment.
id	String	Add-on ID.
job_id	String	ID of the last job executed by the add-on.

Parameter	Type	Description
name	String	Add-on name. sermantInjector: Sermant Injector add-on. It is used to automatically mount Sermant Agent and connect to ServiceComb engines with security authentication disabled through Sermant Agent. You can access the engine and use functions such as application registry and discovery, full-link dark launch, graceful startup and shutdown, and label routing without modifying application code. However, microservice governance is not supported.
status	String	Add-on status. - INITIALIZING - UPGRADING - FAILED - SUCCEEDED - DELETING
version	String	Add-on version.

Example Request

None

Example Response

```
{
  "addons": [
    {
      "status": "FAILED",
      "id": "a2cdf0e2-8495-4db5-82f7-a23b97b57787",
      "name": "sermantInjector",
      "version": "1.3.1.0",
      "cce_id": "02f9a523-cefe-11ee-afc1-0255ac100045",
      "cse_id": "b4e49ad5-5142-437d-8a6b-1de66b1af0db",
      "job_id": "JOB195d62a9-6446-4ac4-a42e-034a48d6a10b",
      "environment_id": "7e2af5b5-4223-40b7-b89b-ee9059827248"
    }
  ]
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.6.3 Querying Add-on Metadata

Function

This API is used to query the add-on metadata in an environment.

URI

GET /v3/{project_id}/cas/environments/{environment_id}/addons-metadata

Table 4-384 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
environment_id	Yes	String	Environment ID. See Obtaining All Environments .

Request

Table 4-385 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

Table 4-386 Response parameters

Parameter	Type	Description
addons	Arrays of objects	Add-on version. See Table 4-387 .

Table 4-387 addons

Parameter	Type	Description
name	String	Add-on name. sermantInjector: Sermant Injector add-on. It is used to automatically mount Sermant Agent and connect to ServiceComb engines with security authentication disabled through Sermant Agent. You can access the engine and use functions such as application registry and discovery, full-link dark launch, graceful startup and shutdown, and label routing without modifying application code. However, microservice governance is not supported.
latest_version	String	Latest add-on version.

Example Request

None

Example Response

```
{
  "addons": [
    {
      "name": "sermantInjector",
      "latest_version": "1.3.1.0"
    }
  ]
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.6.4 Querying Add-on Details Based on the Add-on ID

Function

This API is used to query details about an add-on in an environment.

URI

GET /v3/{project_id}/cas/environments/{environment_id}/addons/{addon_id}

Table 4-388 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
environment_id	Yes	String	Environment ID. See Obtaining All Environments .
addon_id	Yes	String	Add-on ID. See Querying the Add-on List .

Request

Table 4-389 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

Table 4-390 Response parameters

Parameter	Type	Description
id	String	Add-on ID.
name	String	Add-on name. sermantInjector: Sermant Injector add-on. It is used to automatically mount Sermant Agent and connect to ServiceComb engines with security authentication disabled through Sermant Agent. You can access the engine and use functions such as application registry and discovery, full-link dark launch, graceful startup and shutdown, and label routing without modifying application code. However, microservice governance is not supported.
version	String	Add-on version.

Parameter	Type	Description
cce_id	String	ID of the CCE cluster where the add-on is installed.
cse_id	String	ID of the microservice engine bound for add-on installation.
job_id	String	ID of the job executed by the add-on.
environment_id	String	ID of the add-on environment.
status	String	Add-on status. - INITIALIZING - UPGRADING - FAILED - SUCCEEDED - DELETING

Example Request

None

Example Response

```
{
  "status": "SUCCEEDED",
  "id": "d52025c9-8e8f-44c0-994e-9f42f87eebd2",
  "name": "sermantInjector",
  "version": "1.3.1.0",
  "cce_id": "7bc15618-f875-11ee-9505-0255ac100b08",
  "cse_id": "98d033a9-d631-47d7-a63f-0cd8039f7028",
  "job_id": "JOB8d13e3a3-a0cc-404a-955d-efe148e6ab6f",
  "environment_id": "48874c71-4898-442f-87fc-afff135d65d1"
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.6.5 Delivering an Add-on Job Based on the Add-on ID

Function

This API is used to deliver an add-on job based on the add-on ID.

URI

POST /v3/{project_id}/cas/environments/{environment_id}/addons/{addon_id}/action

Table 4-391 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
environment_id	Yes	String	Environment ID. See Obtaining All Environments .
addon_id	Yes	String	Add-on ID. See Querying the Add-on List .

Request

Table 4-392 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Table 4-393 Request body parameters

Parameter	Mandatory	Type	Description
action	Yes	String	Add-on job type. • restart • redeploy

Response

Table 4-394 Response parameters

Parameter	Type	Description
job_id	String	Job ID.

Example Request

Deliver an add-on redeployment job.

```
{  
  "action": "redploy"  
}
```

Example Response

```
{  
  "job_id": "JOB7e9317bb-a63a-46f4-a026-244de2f6eac3"  
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.6.6 Deleting an Add-on Based on the Add-on ID

Function

This API is used to delete an add-on based on the add-on ID.

URI

DELETE /v3/{project_id}/cas/environments/{environment_id}/addons/{addon_id}

Table 4-395 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
environment_id	Yes	String	Environment ID. See Obtaining All Environments .
addon_id	Yes	String	Add-on ID. See Querying the Add-on List .

Request

Table 4-396 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

Table 4-397 Response parameters

Parameter	Type	Description
job_id	String	Job ID.

Example Request

None

Example Response

```
{
  "job_id": "JOB7e9317bb-a63a-46f4-a026-244de2f6eac3"
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.7 Release Management

4.7.1 Old Release Management

4.7.1.1 Creating a Release Task

Function

This API is used to create a release task.

URI

POST /v3/{project_id}/cas/release-plans

Table 4-398 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .

Request

Table 4-399 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Table 4-400 Request body parameters

Parameter	Mandatory	Type	Description
action	Yes	String	Release task type. • clone • upgrade • create
desc	No	String	Description.
components	Yes	Array of objects	Component list. See Table 4-401 .
name	Yes	String	Release task name.
fail_strategy	No	String	Failure strategy. • continue • stop

Table 4-401 components

Parameter	Mandatory	Type	Description
seq	Yes	Integer	Release batch.
component_id	Yes	String	Application component ID.

Parameter	Mandatory	Type	Description
name	Yes	String	Application component name. The value contains 2 to 64 characters consisting of letters, digits, hyphens (-), and underscores (_). It starts with a letter and ends with a letter or digit.
workload_name	No	String	Workload name.
labels	No	Array of objects	Label. For container-based deployment, you can query a workload by label. For VM-based deployment, you can configure a label of the corresponding microservice name to use graceful startup and shutdown. See Table 4-402 .
create_mode	No	String	Component creation mode. This parameter is not transferred by default, indicating to create and deploy a component.
runtime_stack	Yes	Object	Runtime. See Table 4-428 . The value can be obtained from the response of the API for Querying a Technology Stack .
environment_id	Yes	String	Environment ID.
description	No	String	Description. The value can contain up to 128 characters.
source	Yes	Object	Source of the code or software package. See Table 4-429 .
build	No	Object	Component building. This parameter is mandatory when the component is deployed in containers and no image is used. For details, see Table 4-430 .
limit_cpu	No	Number	Maximum CPU limit. Unit: core.
limit_memory	No	Number	Maximum memory size. Unit: GiB.
request_cpu	No	Number	Requested CPU resources. Unit: core.
request_memory	No	Number	Requested memory. Unit: GiB.

Parameter	Mandatory	Type	Description
version	Yes	String	Component version number. The value can contain up to 32 characters. It needs to meet the following rule: $^([0-9]+)(.[0-9]+){2,3}$$
envs	No	Array of objects	Environment variable list. See Table 4-426 .
replica	Yes	Integer	Number of instances.
storages	No	Array of objects	Storage. See Table 4-403 .
deploy_strategy	No	Object	Component deployment. See Table 4-406 .
command	No	Object	Startup command. See Table 4-410 .
post_start	No	Object	Post-start processing. See Table 4-411 .
pre_stop	No	Object	Pre-stop processing. See Table 4-411 .
mesher	No	Object	Multi-language access service mesh. This parameter is mandatory when Node.js is selected for the technology stack and the microservice engine is bound. See Table 4-412 .
timezone	No	String	Time zone where the component runs.
jvm_opts	No	String	JVM parameter. Example: -Xms256m -Xmx1024m. If there are multiple parameters, separate them by spaces. If the parameter is left blank, the default value is used.
tomcat_opts	No	Object	Tomcat parameter. This parameter is valid when you use the Tomcat technology stack. See Table 4-413 .
host_aliases	No	Array of objects	Host alias. See Table 4-414 .

Parameter	Mandatory	Type	Description
dns_policy	No	String	DNS policy. • Default: inherits the domain name resolution configuration of the node where the pod is located. • ClusterFirst: adds the domain name resolution configuration. • ClusterFirstWithHostNet: takes effect only for pods running in hostNetwork mode. • None: replaces the domain name resolution policy.
dns_config	No	Object	DNS configuration. See Table 4-415 .
workload_kind	No	String	Workload type. • deployment • statefulset • daemonset
security_context	No	Object	Security policy. See Table 4-417 .
logs	No	Array of objects	Log collection. See Table 4-419 .
custom_metrics	No	Object	Custom metrics. See Table 4-420 .
affinity	No	Array of objects	Affinity. See Table 4-421 .
anti_affinity	No	Array of objects	Anti-affinity. See Table 4-421 .
liveness_probe	No	Object	Component liveness probe. See Table 4-423 .
readiness_probe	No	Object	Component service probe. See Table 4-423 .
startup_probe	No	Object	Component startup probe. See Table 4-423 .
refer_resources	Yes	Array of objects	Associated resource. See Table 4-424 .
external_accesses	No	Array of Objects	External access. See Table 4-432 .
tolerations	No	Array of Objects	Tolerance policy. See Table 4-433 .

Parameter	Mandatory	Type	Description
container_name	No	String	Container name.
update_strategy	No	Object	Component upgrade strategy. See Table 4-436 .
multi_containers	No	Array of Objects	Multi-container configuration. See Table 4-434 .
service_name	No	String	Service name configured for the inter-instance discovery of a StatefulSet.
ports	No	Array of Objects	Port configured for the inter-instance discovery of a StatefulSet.
workload_content	No	String	Workload content.
configs	No	Array of Objects	Unified component configuration. See Table 4-437 .
enable_serment_injection	No	Boolean	Whether to enable Serment.
enable_serment_spring_register	No	Boolean	Whether Spring Cloud Huawei microservices need to be registered with registry centers through Serment.
enable_serment_dubbo_register	No	Boolean	Whether Dubbo microservices need to be registered with registry centers through Serment.
enable_serment_config	No	Boolean	Whether to enable the Serment dynamic configuration plug-in.
cell_id	No	String	Deployment cell ID.
swimlane_id	No	String	Lane ID.

Parameter	Mandatory	Type	Description
sermantAgentInjectActionFlag	No	Boolean	Whether JDK 17 and JDK 21 automatically add the following JVM environment variables: JDK 17: --add-opens java.base/java.lang=ALL-UNNAMED --add-opens java.base/java.net=ALL-UNNAMED --add-opens java.base/java.math=ALL-UNNAMED --add-opens java.base/sun.net.www=ALL-UNNAMED --add-opens java.base/sun.net.www.protocol.http=ALL-UNNAMED JDK 21: --add-opens java.base/java.lang=ALL-UNNAMED --add-opens java.base/java.net=ALL-UNNAMED --add-opens java.base/java.math=ALL-UNNAMED --add-opens java.base/sun.net.www=ALL-UNNAMED --add-opens java.base/sun.net.www.protocol.http=ALL-UNNAMED -Dnet.bytebuddy.experimental=true
sys_config	No	Map<String, String>	Uses a compressed package to deploy component configuration items in the VM environment.
image_pull_secrets	No	Array of String	List of image pull secrets.
auto_lts_config	No	Object	Automatic LTS configuration. See Table 4-438 .
config_encryption	No	Object	This parameter is mandatory when the encryption key is configured for a single-container component. See Table 4-439 .

Table 4-402 labels

Parameter	Mandatory	Type	Description
key	Yes	String	Label name.
value	Yes	String	Label value.

Table 4-403 storages

Parameter	Mandatory	Type	Description
type	Yes	String	Data storage type. • HostPath: host path for local disk mounting. • EmptyDir: temporary directory for local disk mounting. • ConfigMap: configuration item for local disk mounting. • Secret: secret for local disk mounting. • PersistentVolumeClaim: cloud storage mounting.
name	Yes	String	Name of the storage disk.
parameters	Yes	Object	For details about the parameters of each data storage type, see Table 4-404 .
mounts	Yes	Array of objects	Mount path of data storage. See Table 4-405 .

Table 4-404 component_storage_parameters

Parameter	Mandatory	Type	Description
path	No	String	Host path. This parameter is applicable to the HostPath storage type. This parameter is mandatory when type is set to HostPath .
name	No	String	Name of a configuration item, secret, or PVC, which is applicable to ConfigMap, Secret, or PersistentVolumeClaim storage, respectively. This parameter is mandatory when type is set to ConfigMap , Secret , or PersistentVolumeClaim .
default_mode	No	Integer	Mounting permission. The value is in decimal format. Example: 384. This parameter is applicable to ConfigMap and secret storage.

Parameter	Mandatory	Type	Description
medium	No	String	This parameter is applicable to EmptyDir storage. If it is not specified, the default disk medium is used. If it is set to memory , memory storage is enabled.

Table 4-405 mounts

Parameter	Mandatory	Type	Description
path	Yes	String	Mount path.
sub_path	Yes	String	Subpath of the mount path.
read_only	Yes	Boolean	Read only or not.

Table 4-406 deploy_strategy

Parameter	Mandatory	Type	Description
type	Yes	String	Deployment type. - OneBatchRelease - RollingRelease - GrayRelease
rolling_release	No	Object	Rolling deployment parameter, which is mandatory when type is set to RollingRelease . For details, see Table 4-407 .
gray_release	No	Object	Dark launch upgrade parameter, which is mandatory when type is set to GrayRelease . For details, see Table 4-408 .

Table 4-407 rolling_release

Parameter	Mandatory	Type	Description
batches	Yes	Integer	Deployment batches.

Table 4-408 gray_release

Parameter	Mandatory	Type	Description
type	Yes	String	Dark launch policy. • WEIGHT: dark launch by traffic. • CONTENT: dark launch by content.
first_batch_weight	Yes	Integer	Proportion of first-batch dark launch traffic. This parameter is mandatory when type is set to WEIGHT .
first_batch_replica	Yes	Integer	Number of first-batch dark launch instances. This parameter is mandatory when type is set to WEIGHT .
remaining_batches	Yes	Integer	Number of batches for remaining instance deployment. After the first batch is completed, the remaining instances will be deployed in specified batches. Example: If there are five instances remaining to be deployed in three batches, these five instances will be upgraded based on 2:2:1. This parameter is mandatory when type is set to WEIGHT .
deployment_mode	No	Integer	Deployment mode. • 1: The upstream is a microservice gateway. • 3: The upstream is a microservice. • 4: The upstream is a load balancer.
replica_surge_mode	No	String	Mode of adding a dark launch instance. This parameter is mandatory when type is set to CONTENT . • MIRROR: blue-green. • EXTRA: canary (increase, then decrease) • NOSURGE: canary (decrease, then increase)
rule_match_mode	No	String	Effective mode of a dark launch rule. This parameter is mandatory when type is set to CONTENT . • ALL: All conditions are met. • ANY: Any condition is met.

Parameter	Mandatory	Type	Description
rules	No	Object	Dark launch rule. See Table 4-409 . This parameter is mandatory when type is set to CONTENT .

Table 4-409 gray_rules

Parameter	Mandatory	Type	Description
type	Yes	String	Match type. Currently, only HEADER is supported.
key	Yes	String	Parameter name.
value	Yes	String	Condition value.
condition	Yes	String	Condition type. • EQUAL • MATCH • IN: enumeration

Table 4-410 command

Parameter	Mandatory	Type	Description
command	No	Array of String	Command for controlling container running.
args	No	Array of String	Parameter for controlling container running. Example: -port=8080. If there are multiple parameters, separate them by line breaks.

Table 4-411 component_lifecycle

Parameter	Mandatory	Type	Description
type	Yes	String	Processing method. • http • command

Parameter	Mandatory	Type	Description
scheme	No	String	HTTP request type, which can be HTTP or HTTPS. This parameter is valid only when type is set to http .
host	No	String	Custom IP address. If this parameter is left blank, the pod IP address is used. This parameter is valid only when type is set to http .
port	No	Integer	Port number. This parameter is valid only when type is set to http .
path	No	String	Request path. This parameter is valid only when type is set to http .
command	No	Array of String	Command list. This parameter is valid only when type is set to command .

Table 4-412 mesher

Parameter	Mandatory	Type	Description
port	Yes	Integer	Process listening port.

Table 4-413 tomcat_opt

Parameter	Mandatory	Type	Description
server_xml	Yes	String	- Listening port in the server.xml file will take effect even if it has been configured in public network access. - The custom Tomcat configuration may conflict with the existing port. Check whether the custom port is occupied.

Table 4-414 host_aliases

Parameter	Mandatory	Type	Description
ip	Yes	String	IP address.

Parameter	Mandatory	Type	Description
hostname	Yes	Array of String	Host alias.

Table 4-415 dns_config

Parameter	Mandatory	Type	Description
nameservers	No	Array of String	IP address list of the DNS server of the pod.
searches	No	Array of String	(Optional) List of DNS search domains used to search for host names in the pod.
options	No	Array of objects	(Optional) List of objects. Each object may have a name (mandatory) and a value (optional). The contents of the objects here will be merged into the options generated from the specified DNS policy. Duplicate entries will be deleted. See Table 4-416 .

Table 4-416 options

Parameter	Mandatory	Type	Description
name	Yes	String	Operation name.
value	No	String	Operation value.

Table 4-417 security_context

Parameter	Mandatory	Type	Description
run_as_user	No	Integer	User for running containers. Example: To run containers as user root , set the user ID to 0 .
run_as_group	No	Integer	Owner group specified for running container processes.
capabilities	No	Object	Capability set. See Table 4-418 .

Table 4-418 capabilities

Parameter	Mandatory	Type	Description
add	No	Array of String	Grant the Linux permission to the running user.
drop	No	Array of String	Remove the Linux permission of the running user.

Table 4-419 logs

Parameter	Mandatory	Type	Description
log_path	Yes	String	Container log path.
rotate	Yes	String	Interval for dumping logs.
host_path	Yes	String	Mounted host path.
host_extend_path	Yes	String	Host extension path. By extending the host path, you can distinguish mounting from different containers in the same host path. • None: The extended path is not used. • PodUID: extend the host path based on the pod ID. • PodName: extend the host path based on the pod name. • PodUID/ContainerName: extend the host path based on the pod ID and container name. • PodName/ContainerName: extend the host path based on the pod name and container name.

Table 4-420 custom_metric

Parameter	Mandatory	Type	Description
path	Yes	String	Collection path. Example: /metrics.
port	Yes	Integer	Collection port. Example: 9090.
dimensions	Yes	String	Monitoring dimension. Example: cpu_usage,mem_usage.

Table 4-421 component_affinity

Parameter	Mandatory	Type	Description
kind	No	String	Object kind. • node • pod
condition	No	String	Policy type. • required: The requirement must be met. • preferred: The requirement is met as much as possible.
match_expressions	No	Array of objects	Scheduling policy rule. See Table 4-422 .
weight	No	String	Weight. • The value is 100 when condition is set to required. • Enter an integer ranging from 1 to 100 when condition is set to preferred.

Table 4-422 match_expressions

Parameter	Mandatory	Type	Description
key	Yes	String	Condition key.
value	Yes	String	Condition value.
operation	Yes	String	Operation.

Table 4-423 component_probe

Parameter	Mandatory	Type	Description
type	Yes	String	Type. Value: http, tcp, or command.
delay	Yes	Integer	Interval between the startup and detection.
timeout	Yes	Integer	Detection timeout interval.
scheme	No	String	Request type. Value: HTTP or HTTPS. This parameter is valid only when type is set to http .

Parameter	Mandatory	Type	Description
host	No	String	Custom IP address. If this parameter is left blank, the pod IP address is used. This parameter is valid only when type is set to http .
port	No	Integer	Port number. This parameter is valid only when type is set to http or tcp .
path	No	String	Request path. This parameter is valid only when type is set to http .
command	No	Array of String	Command list. This parameter is valid only when type is set to command .

Table 4-424 refer_resources

Parameter	Mandatory	Type	Description
id	Yes	String	Resource ID.
type	Yes	String	Resource type. • vpc • eip • elb • cce • ecs • dcs • rds • cse: ServiceComb engine. • nacos: registry/configuration center. • customServer • pvc • gaussdb • dms_rocketmq • css_es: Elasticsearch
parameters	No	Object	Resource parameters. See Table 4-425 .

Table 4-425 refer_resource_parameter

Parameter	Mandatory	Type	Description
namespace	No	String	Namespace.
name	No	String	Resource name.
capacity	No	Integer	Storage capacity, in GiB.
type	No	String	Volume declaration type. • efs • obs • sfs • evs
class	No	String	Volume declaration subtype. • csi-sfsturbo: fixed when type is set to efs. • csi-obs: fixed when type is set to obs. • csi-nas: fixed when type is set to sfs. • csi-disk: fixed when type is set to evs.
obs_volume_type	No	String	OBS volume type. Default value: STANDARD.
access_mode	No	String	Storage volume access mode. • ReadWriteOnce: A volume can be mounted as read-write by a single node. • ReadWriteMany: A volume can be mounted as read-write by many nodes.
availableZone	No	String	AZ.
volume_type	No	String	Volume type.

Table 4-426 envs

Parameter	Mandatory	Type	Description
name	Yes	String	Variable name.
value	No	String	Variable value.
value_from	No	Object	Variable reference. See Table 4-427 .

Table 4-427 value_from

Parameter	Mandatory	Type	Description
reference_type	Yes	String	Reference type. - configMapKey - secretKey
name	Yes	String	Name of the configuration item or secret.
key	No	String	Key of the configuration item or secret.
optional	No	Boolean	Whether the key of the configuration item or secret is mandatory.

Table 4-428 runtime_stack

Parameter	Mandatory	Type	Description
name	Yes	String	Technology stack name.
type	Yes	String	Technology stack type. Value: Java, Tomcat, Nodejs, Php, Docker, or Python. - For VM-based deployment, Java, Tomcat, and Nodejs are supported. - For container-based deployment, Java, Tomcat, Nodejs, Php, and Docker are supported.
version	Yes	String	Technology stack version.
deploy_mode	Yes	String	Deployment mode. - container - virtualmachine

Table 4-429 source

Parameter	Mandatory	Type	Description
kind	Yes	String	Component source kind. - code - package - image
version	No	String	Version number.

Parameter	Mandatory	Type	Description
url	No	String	Package address. This parameter is mandatory when kind is set to package or image .
storage	No	String	Storage mode. • devcloud: CodeArts release repository • obs • http: custom file address • swr This parameter is mandatory when kind is set to package or image .
http_username	No	String	Username. This parameter is mandatory when storage is set to http . Username and password authentication is supported.
http_password	No	String	Password. This parameter is mandatory when storage is set to http . Username and password authentication is supported.
http_command	No	String	Address of the custom software package. This parameter is mandatory when storage is set to http .
http_headers	No	Map<String, String>	This parameter is mandatory when storage is set to http . You can customize request headers as key-value pairs, for example, {"key": "xx", "value": "xx"}.
auth	No	String	Authentication mode. Value: iam or none. Default value: iam.
repo_auth	No	String	Authorization name, which can be obtained from the authorization list. For details, see Obtaining a Git Repository Authorization List .
repo_namespace	No	String	Namespace of the code repository.
repo_ref	No	String	Code branch or tag. Default value: master.
repo_type	No	String	Code repository type. Value: GitHub, BitBucket, or GitLab.
web_url	No	String	Code redirection URL. Example: https://github.com/example/demo.git.
repo_url	No	String	Code repository URL. Example: https://github.com/example/demo.git.

Parameter	Mandatory	Type	Description
commit_id	No	String	In the code repository, commit ID is the unique identifier of each commit and is used to identify and reference a commit. The value contains 8 to 40 characters and starts with a lowercase letter or digit.

Table 4-430 build

Parameter	Mandatory	Type	Description
parameters	No	Map<String, Object>	This parameter is provided only when no ID is available during build creation. See Table 4-431 .

Table 4-431 parameters

Parameter	Mandatory	Type	Description
build_cmd	No	String	Compilation command. By default: 1. When build.sh exists in the root directory, the command is ./build.sh . 2. When build.sh does not exist in the root directory, the command varies depending on the OS. Example: • Java and Tomcat: mvn clean package • Node.js: npm build
dockerfile_path	No	String	Address of the Docker file. By default, the Docker file is in the root directory (./).
artifact_namespace	No	String	Build archive organization. Default value: cas_{project_id}.
cluster_id	Yes	String	ID of the cluster to be built.
node_label_selector	No	Map<String, String>	key: label key. value: label value.
environment_id	No	String	Environment ID.

Table 4-432 external_accesses

Parameter	Mandatory	Type	Description
protocol	No	String	External access type. Value: http or https.
address	No	String	External access address.
forward_port	No	Integer	External access port.

Table 4-433 tolerations

Parameter	Mandatory	Type	Description
key	No	String	Taint key.
operator	No	String	Operator. - Equal: matches the nodes with the specified taint key (mandatory) and value. If the taint value is left blank, all taints with the same key will be matched. - Exists: matches nodes with the specified taint key. In this case, the taint value cannot be specified. Not specifying any taint key means all taints are tolerated.
value	No	String	Taint value. This parameter is mandatory when operator is set to Equal .
effect	No	String	Taint policy. - NoScheduler: Only the NoSchedule taint is matched. - PreferNoSchedule: Only the PreferNoSchedule taint is matched. - NoExecute: Only the NoExecute taint is matched. - If this parameter is not set, all taint effects are matched.
toleration_seconds	No	Integer	Tolerance time window. This parameter can be set only when effect is set to NoExecute . Within the tolerance time window, pods still run on the node with taints. After the time expires, the pods will be evicted.

Table 4-434 multi_containers

Parameter	Mandatory	Type	Description
container_name	Yes	String	Container name.
source	Yes	Object	Source of the code or software package. See Table 4-429 . In multi-container mode, only images are supported.
limit_cpu	No	Number	Maximum CPU limit. Unit: core.
limit_memory	No	Number	Maximum memory size. Unit: GiB.
request_cpu	No	Number	Requested CPU resources. Unit: core.
request_memory	No	Number	Requested memory. Unit: GiB.
envs	No	Array of objects	Environment variable list. See Table 4-426 .
liveness_probe	No	Object	Container liveness probe. See Table 4-423 .
readiness_probe	No	Object	Container service probe. See Table 4-423 .
startup_probe	No	Object	Container startup probe. See Table 4-423 .
command	No	Object	Startup command. See Table 4-410 .
post_start	No	Object	Post-start processing. See Table 4-411 .
pre_stop	No	Object	Pre-stop processing. See Table 4-411 .
logs	No	Array of objects	Log collection. See Table 4-419 .
storages	No	Array of objects	Storage. See Table 4-403 .
security_context	No	Object	Security policy. See Table 4-417 .
config_encryption	No	Object	This parameter is mandatory when the encryption key is configured for a multi-container component. See Table 4-439 .

Table 4-435 ports

Parameter	Mandatory	Type	Description
name	Yes	String	Port name.
port	Yes	String	Access port.
target_port	Yes	String	Container port.

Table 4-436 update_strategy

Parameter	Mandatory	Type	Description
type	No	String	Component upgrade policy. Options: - RollingUpdate (default): New pods are created gradually, and then old pods are deleted. - Recreate: Old pods are deleted, and then new pods are created.
max_unavailable	No	- String - Integer	Maximum number of unavailable pods. That is, the number or percentage of pods that can be deleted based on the value of replica . The default value is 25% . This parameter can be set when workload_kind is set to deployment . - String (%): the maximum percentage of unavailable pods to the total number of pods. - Integer: the maximum number of unavailable pods.
max_surge	No	- String - Integer	Maximum surge. That is, the number or percentage of pods allowed based on the value of replica . The default value is 25% . This parameter can be set only when workload_kind is set to deployment . - String (%) - Integer

Parameter	Mandatory	Type	Description
termination_period_seconds	No	Integer	Scale-in time window. That is, the time window for executing commands before a workload stops. The default value is 30s.
min_ready_seconds	No	Integer	Minimum ready time of a pod. That is, the minimum ready time of a new pod without any container crash. A pod is considered available only after the minimum ready time is exceeded. The default value is 0 .
progress_deadline_seconds	No	Integer	Maximum upgrade duration. That is, the time (in seconds) the system waits for before reporting a component upgrade failure. If this parameter is specified, the value of this parameter must be greater than that of min_ready_seconds .
revision_history_limit	No	Integer	Maximum number of pods to retain. That is, the number of old pods to retain to allow rollback. The default value is 10 .

Table 4-437 configs

Parameter	Mandatory	Type	Description
config_id	Yes	String	Unified configuration ID.
mount_path	Yes	String	Mount path.

Table 4-438 auto_lts_config

Parameter	Mandatory	Type	Description
enable	Yes	Boolean	Whether to enable automatic LTS configuration.
lts_log_path	No	Array of String	Access configuration log path.

Table 4-439 config_encryption

Parameter	Mandatory	Type	Description
key_id	Yes	String	User key ID in DEW.

Response

Table 4-440 Response parameters

Parameter	Type	Description
release_plan_id	String	Release task ID.

Example Request

Create a release task to upgrade components.

Example Response

```
{
  "release_plan_id": "bb09c189-d17e-480a-a0e4-3a102b3795b0"
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.7.1.2 Editing a Release Task Based on the Release Task ID

Function

This API is used to edit a release task based on the release task ID.

URI

PUT /v3/{project_id}/cas/release-plans/{release_plan_id}

Table 4-441 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
release_plan_id	Yes	String	Release task ID. See Obtaining the Release Task List .

Request

Table 4-442 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Table 4-443 Request body parameters

Parameter	Mandatory	Type	Description
desc	No	String	Description.
components	Yes	Array of objects	Component list. See Table 4-444 .
name	Yes	String	Release task name.
fail_strategy	No	String	Failure strategy. • continue • stop

Table 4-444 components

Parameter	Mandatory	Type	Description
seq	Yes	Integer	Release batch.
component_id	Yes	String	Application component ID.
name	Yes	String	Application component name. The value contains 2 to 64 characters consisting of letters, digits, hyphens (-), and underscores (_). It starts with a letter and ends with a letter or digit.
workload_name	No	String	Workload name.
labels	No	Array of objects	Label. For container-based deployment, you can query a workload by label. For VM-based deployment, you can configure a label of the corresponding microservice name to use graceful startup and shutdown. See Table 4-445 .
create_mode	No	String	Component creation mode. This parameter is not transferred by default, indicating to create and deploy a component.
runtime_stack	Yes	Object	Runtime. See Table 4-471 . The value can be obtained from the response of the API for Querying a Technology Stack .
environment_id	Yes	String	Environment ID.
description	No	String	Description. The value can contain up to 128 characters.
source	Yes	Object	Source of the code or software package. See Table 4-472 .
build	No	Object	Component building. This parameter is mandatory when the component is deployed in containers and no image is used. For details, see Table 4-473 .
limit_cpu	No	Number	Maximum CPU limit. Unit: core.
limit_memory	No	Number	Maximum memory size. Unit: GiB.
request_cpu	No	Number	Requested CPU resources. Unit: core.

Parameter	Mandatory	Type	Description
request_memory	No	Number	Requested memory. Unit: GiB.
version	Yes	String	Component version number. The value can contain up to 32 characters. It needs to meet the following rule: <code>^([0-9]+)(.[0-9]+){2,3}\$</code>
envs	No	Array of objects	Environment variable list. See Table 4-469 .
replica	Yes	Integer	Number of instances.
storages	No	Array of objects	Storage. See Table 4-446 .
deploy_strategy	No	Object	Component deployment. See Table 4-449 .
command	No	Object	Startup command. See Table 4-453 .
post_start	No	Object	Post-start processing. See Table 4-454 .
pre_stop	No	Object	Pre-stop processing. See Table 4-454 .
mesher	No	Object	Multi-language access service mesh. This parameter is mandatory when Node.js is selected for the technology stack and the microservice engine is bound. See Table 4-455 .
timezone	No	String	Time zone where the component runs.
jvm_opts	No	String	JVM parameter. Example: -Xms256m -Xmx1024m. If there are multiple parameters, separate them by spaces. If the parameter is left blank, the default value is used.
tomcat_opts	No	Object	Tomcat parameter. This parameter is valid when you use the Tomcat technology stack. See Table 4-456 .

Parameter	Mandatory	Type	Description
host_aliases	No	Array of objects	Host alias. See Table 4-457 .
dns_policy	No	String	DNS policy. • Default: inherits the domain name resolution configuration of the node where the pod is located. • ClusterFirst: adds the domain name resolution configuration. • ClusterFirstWithHostNet: takes effect only for pods running in hostNetwork mode. • None: replaces the domain name resolution policy.
dns_config	No	Object	DNS configuration. See Table 4-458 .
workload_kind	No	String	Workload type. • deployment • statefulset • daemonset
security_context	No	Object	Security policy. See Table 4-460 .
logs	No	Array of objects	Log collection. See Table 4-462 .
custom_metrics	No	Object	Custom metrics. See Table 4-463 .
affinity	No	Array of objects	Affinity. See Table 4-464 .
anti_affinity	No	Array of objects	Anti-affinity. See Table 4-464 .
liveness_probe	No	Object	Component liveness probe. See Table 4-466 .
readiness_probe	No	Object	Component service probe. See Table 4-466 .

Parameter	Mandatory	Type	Description
startup_probe	No	Object	Component startup probe. See Table 4-466 .
refer_resources	Yes	Array of objects	Associated resource. See Table 4-467 .
external_accesses	No	Array of Objects	External access. See Table 4-475 .
tolerations	No	Array of Objects	Tolerance policy. See Table 4-476 .
container_name	No	String	Container name.
update_strategy	No	Object	Component upgrade strategy. See Table 4-479 .
multi_containers	No	Array of Objects	Multi-container configuration. See Table 4-477 .
service_name	No	String	Service name configured for the inter-instance discovery of a StatefulSet.
ports	No	Array of Objects	Port configured for the inter-instance discovery of a StatefulSet.
workload_content	No	String	Workload content.
configs	No	Array of Objects	Unified component configuration. See Table 4-480 .
enable_serment_injection	No	Boolean	Whether to enable Serment.
enable_serment_spring_register	No	Boolean	Whether Spring Cloud Huawei microservices need to be registered with registry centers through Serment.

Parameter	Mandatory	Type	Description
enable_serment_dubbo_register	No	Boolean	Whether Dubbo microservices need to be registered with registry centers through Serment.
enable_serment_config	No	Boolean	Whether to enable the Serment dynamic configuration plug-in.
cell_id	No	String	Deployment cell ID.
swimlane_id	No	String	Lane ID.
sermantAgentInjectActionFlag	No	Boolean	Whether JDK 17 and JDK 21 automatically add the following JVM environment variables: JDK 17: --add-opens java.base/java.lang=ALL-UNNAMED --add-opens java.base/java.net=ALL-UNNAMED --add-opens java.base/java.math=ALL-UNNAMED --add-opens java.base/sun.net.www=ALL-UNNAMED --add-opens java.base/sun.net.www.protocol.http=ALL-UNNAMED JDK 21: --add-opens java.base/java.lang=ALL-UNNAMED --add-opens java.base/java.net=ALL-UNNAMED --add-opens java.base/java.math=ALL-UNNAMED --add-opens java.base/sun.net.www=ALL-UNNAMED --add-opens java.base/sun.net.www.protocol.http=ALL-UNNAMED -Dnet.bytebuddy.experimental=true
sys_config	No	Map<String, String>	Uses a compressed package to deploy component configuration items in the VM environment.
image_pull_secrets	No	Array of String	List of image pull secrets.
auto_lts_config	No	Object	Automatic LTS configuration. See Table 4-481 .
config_encryption	No	Object	This parameter is mandatory when the encryption key is configured for a single-container component. See Table 4-482 .

Table 4-445 labels

Parameter	Mandatory	Type	Description
key	Yes	String	Label name.
value	Yes	String	Label value.

Table 4-446 storages

Parameter	Mandatory	Type	Description
type	Yes	String	Data storage type. • HostPath: host path for local disk mounting. • EmptyDir: temporary directory for local disk mounting. • ConfigMap: configuration item for local disk mounting. • Secret: secret for local disk mounting. • PersistentVolumeClaim: cloud storage mounting.
name	Yes	String	Name of the storage disk.
parameters	Yes	Object	For details about the parameters of each data storage type, see Table 4-447 .
mounts	Yes	Array of objects	Mount path of data storage. See Table 4-448 .

Table 4-447 component_storage_parameters

Parameter	Mandatory	Type	Description
path	No	String	Host path. This parameter is applicable to the HostPath storage type. This parameter is mandatory when type is set to HostPath .
name	No	String	Name of a configuration item, secret, or PVC, which is applicable to ConfigMap, Secret, or PersistentVolumeClaim storage, respectively. This parameter is mandatory when type is set to ConfigMap , Secret , or PersistentVolumeClaim .
default_mode	No	Integer	Mounting permission. The value is in decimal format. Example: 384. This parameter is applicable to ConfigMap and secret storage.

Parameter	Mandatory	Type	Description
medium	No	String	This parameter is applicable to EmptyDir storage. If it is not specified, the default disk medium is used. If it is set to memory , memory storage is enabled.

Table 4-448 mounts

Parameter	Mandatory	Type	Description
path	Yes	String	Mount path.
sub_path	Yes	String	Subpath of the mount path.
read_only	Yes	Boolean	Read only or not.

Table 4-449 deploy_strategy

Parameter	Mandatory	Type	Description
type	Yes	String	Deployment type. - OneBatchRelease - RollingRelease - GrayRelease
rolling_release	No	Object	Rolling deployment parameter, which is mandatory when type is set to RollingRelease . For details, see Table 4-450 .
gray_release	No	Object	Dark launch upgrade parameter, which is mandatory when type is set to GrayRelease . For details, see Table 4-451 .

Table 4-450 rolling_release

Parameter	Mandatory	Type	Description
batches	Yes	Integer	Deployment batches.

Table 4-451 gray_release

Parameter	Mandatory	Type	Description
type	Yes	String	Dark launch policy. - WEIGHT: dark launch by traffic. - CONTENT: dark launch by content.
first_batch_weight	Yes	Integer	Proportion of first-batch dark launch traffic. This parameter is mandatory when type is set to WEIGHT .
first_batch_replica	Yes	Integer	Number of first-batch dark launch instances. This parameter is mandatory when type is set to WEIGHT .
remaining_batches	Yes	Integer	Number of batches for remaining instance deployment. After the first batch is completed, the remaining instances will be deployed in specified batches. Example: If there are five instances remaining to be deployed in three batches, these five instances will be upgraded based on 2:2:1. This parameter is mandatory when type is set to WEIGHT .
deployment_mode	No	Integer	Deployment mode. 1: The upstream is a microservice gateway. 3: The upstream is a microservice. 4: The upstream is a load balancer.
replica_surge_mode	No	String	Mode of adding a dark launch instance. This parameter is mandatory when type is set to CONTENT . - MIRROR: blue-green. - EXTRA: canary (increase, then decrease) - NOSURGE: canary (decrease, then increase)
rule_match_mode	No	String	Effective mode of a dark launch rule. This parameter is mandatory when type is set to CONTENT . - ALL: All conditions are met. - ANY: Any condition is met.
rules	No	Object	Dark launch rule. See Table 4-452 . This parameter is mandatory when type is set to CONTENT .

Table 4-452 gray_rules

Parameter	Mandatory	Type	Description
type	Yes	String	Match type. Currently, only HEADER is supported.
key	Yes	String	Parameter name.
value	Yes	String	Condition value.
condition	Yes	String	Condition type. • EQUAL • MATCH • IN: enumeration

Table 4-453 command

Parameter	Mandatory	Type	Description
command	No	Array of String	Command for controlling container running.
args	No	Array of String	Parameter for controlling container running. Example: -port=8080. If there are multiple parameters, separate them by line breaks.

Table 4-454 component_lifecycle

Parameter	Mandatory	Type	Description
type	Yes	String	Processing method. • http • command
scheme	No	String	HTTP request type, which can be HTTP or HTTPS. This parameter is valid only when type is set to http .

Parameter	Mandatory	Type	Description
host	No	String	Custom IP address. If this parameter is left blank, the pod IP address is used. This parameter is valid only when type is set to http .
port	No	Integer	Port number. This parameter is valid only when type is set to http .
path	No	String	Request path. This parameter is valid only when type is set to http .
command	No	Array of String	Command list. This parameter is valid only when type is set to command .

Table 4-455 mesher

Parameter	Mandatory	Type	Description
port	Yes	Integer	Process listening port.

Table 4-456 tomcat_opt

Parameter	Mandatory	Type	Description
server_xml	Yes	String	1. Listening port in the server.xml file will take effect even if it has been configured in public network access. 2. The custom Tomcat configuration may conflict with the existing port. Check whether the custom port is occupied.

Table 4-457 host_aliases

Parameter	Mandatory	Type	Description
ip	Yes	String	IP address.
hostname	Yes	Array of String	Host alias.

Table 4-458 dns_config

Parameter	Mandatory	Type	Description
nameservers	No	Array of String	IP address list of the DNS server of the pod.
searches	No	Array of String	(Optional) List of DNS search domains used to search for host names in the pod.
options	No	Array of objects	(Optional) List of objects. Each object may have a name (mandatory) and a value (optional). The contents of the objects here will be merged into the options generated from the specified DNS policy. Duplicate entries will be deleted. See Table 4-459 .

Table 4-459 options

Parameter	Mandatory	Type	Description
name	Yes	String	Operation name.
value	No	String	Operation value.

Table 4-460 security_context

Parameter	Mandatory	Type	Description
run_as_user	No	Integer	User for running containers. Example: To run containers as user root , set the user ID to 0 .
run_as_group	No	Integer	Owner group specified for running container processes.
capabilities	No	Object	Capability set. See Table 4-461 .

Table 4-461 capabilities

Parameter	Mandatory	Type	Description
add	No	Array of String	Grant the Linux permission to the running user.

Parameter	Mandatory	Type	Description
drop	No	Array of String	Remove the Linux permission of the running user.

Table 4-462 logs

Parameter	Mandatory	Type	Description
log_path	Yes	String	Container log path.
rotate	Yes	String	Interval for dumping logs.
host_path	Yes	String	Mounted host path.
host_extend_path	Yes	String	Host extension path. By extending the host path, you can distinguish mounting from different containers in the same host path. • None: The extended path is not used. • PodUID: extend the host path based on the pod ID. • PodName: extend the host path based on the pod name. • PodUID/ContainerName: extend the host path based on the pod ID and container name. • PodName/ContainerName: extend the host path based on the pod name and container name.

Table 4-463 custom_metric

Parameter	Mandatory	Type	Description
path	Yes	String	Collection path. Example: /metrics.
port	Yes	Integer	Collection port. Example: 9090.
dimensions	Yes	String	Monitoring dimension. Example: cpu_usage,mem_usage.

Table 4-464 component_affinity

Parameter	Mandatory	Type	Description
kind	No	String	Object kind. • node • pod
condition	No	String	Policy type. • required: The requirement must be met. • preferred: The requirement is met as much as possible.
match_expressions	No	Array of objects	Scheduling policy rule. See Table 4-465 .
weight	No	String	Weight. • The value is 100 when condition is set to required. • Enter an integer ranging from 1 to 100 when condition is set to preferred.

Table 4-465 match_expressions

Parameter	Mandatory	Type	Description
key	Yes	String	Condition key.
value	Yes	String	Condition value.
operation	Yes	String	Operation.

Table 4-466 component_probe

Parameter	Mandatory	Type	Description
type	Yes	String	Type. Value: http, tcp, or command.
delay	Yes	Integer	Interval between the startup and detection.
timeout	Yes	Integer	Detection timeout interval.
scheme	No	String	Request type. Value: HTTP or HTTPS. This parameter is valid only when type is set to http .

Parameter	Mandatory	Type	Description
host	No	String	Custom IP address. If this parameter is left blank, the pod IP address is used. This parameter is valid only when type is set to http .
port	No	Integer	Port number. This parameter is valid only when type is set to http or tcp .
path	No	String	Request path. This parameter is valid only when type is set to http .
command	No	Array of String	Command list. This parameter is valid only when type is set to command .

Table 4-467 refer_resources

Parameter	Mandatory	Type	Description
id	Yes	String	Resource ID.
type	Yes	String	Resource type. • vpc • eip • elb • cce • ecs • dcs • rds • cse: ServiceComb engine. • nacos: registry/configuration center. • customServer • pvc • gaussdb • dms_rocketmq • css_es: Elasticsearch
parameters	No	Object	Resource parameters. See Table 4-468 .

Table 4-468 refer_resource_parameter

Parameter	Mandatory	Type	Description
namespace	No	String	Namespace.
name	No	String	Resource name.
capacity	No	Integer	Storage capacity, in GiB.
type	No	String	Volume declaration type. • efs • obs • sfs • evs
class	No	String	Volume declaration subtype. • csi-sfsturbo: fixed when type is set to efs. • csi-obs: fixed when type is set to obs. • csi-nas: fixed when type is set to sfs. • csi-disk: fixed when type is set to evs.
obs_volume_type	No	String	OBS volume type. Default value: STANDARD.
access_mode	No	String	Storage volume access mode. • ReadWriteOnce: A volume can be mounted as read-write by a single node. • ReadWriteMany: A volume can be mounted as read-write by many nodes.
availableZone	No	String	AZ.
volume_type	No	String	Volume type.

Table 4-469 envs

Parameter	Mandatory	Type	Description
name	Yes	String	Variable name.
value	No	String	Variable value.
value_from	No	Object	Variable reference. See Table 4-470 .

Table 4-470 value_from

Parameter	Mandatory	Type	Description
reference_type	Yes	String	Reference type. - configMapKey - secretKey
name	Yes	String	Name of the configuration item or secret.
key	No	String	Key of the configuration item or secret.
optional	No	Boolean	Whether the key of the configuration item or secret is mandatory.

Table 4-471 runtime_stack

Parameter	Mandatory	Type	Description
name	Yes	String	Technology stack name.
type	Yes	String	Technology stack type. Value: Java, Tomcat, Nodejs, Php, Docker, or Python. - For VM-based deployment, Java, Tomcat, and Nodejs are supported. - For container-based deployment, Java, Tomcat, Nodejs, Php, and Docker are supported.
version	Yes	String	Technology stack version.
deploy_mode	Yes	String	Deployment mode. - container - virtualmachine

Table 4-472 source

Parameter	Mandatory	Type	Description
kind	Yes	String	Component source kind. - code - package - image
version	No	String	Version number.

Parameter	Mandatory	Type	Description
url	No	String	Package address. This parameter is mandatory when kind is set to package or image .
storage	No	String	Storage mode. • devcloud: CodeArts release repository • obs • http: custom file address • swr This parameter is mandatory when kind is set to package or image .
http_username	No	String	Username. This parameter is mandatory when storage is set to http . Username and password authentication is supported.
http_password	No	String	Password. This parameter is mandatory when storage is set to http . Username and password authentication is supported.
http_command	No	String	Address of the custom software package. This parameter is mandatory when storage is set to http .
http_headers	No	Map<String, String>	This parameter is mandatory when storage is set to http . You can customize request headers as key-value pairs, for example, {"key": "xx", "value": "xx"}.
auth	No	String	Authentication mode. Value: iam or none. Default value: iam.
repo_auth	No	String	Authorization name, which can be obtained from the authorization list. For details, see Obtaining a Git Repository Authorization List .
repo_namespace	No	String	Namespace of the code repository.
repo_ref	No	String	Code branch or tag. Default value: master.
repo_type	No	String	Code repository type. Value: GitHub, BitBucket, or GitLab.
web_url	No	String	Code redirection URL. Example: https://github.com/example/demo.git.
repo_url	No	String	Code repository URL. Example: https://github.com/example/demo.git.

Parameter	Mandatory	Type	Description
commit_id	No	String	In the code repository, commit ID is the unique identifier of each commit and is used to identify and reference a commit. The value contains 8 to 40 characters and starts with a lowercase letter or digit.

Table 4-473 build

Parameter	Mandatory	Type	Description
parameters	No	Map<String, Object>	This parameter is provided only when no ID is available during build creation. See Table 4-474 .

Table 4-474 parameters

Parameter	Mandatory	Type	Description
build_cmd	No	String	Compilation command. By default: 1. When build.sh exists in the root directory, the command is ./build.sh . 2. When build.sh does not exist in the root directory, the command varies depending on the OS. Example: • Java and Tomcat: mvn clean package • Nodejs: npm build
dockerfile_path	No	String	Address of the Docker file. By default, the Docker file is in the root directory (.).
artifact_namespace	No	String	Build archive organization. Default value: cas_{project_id}.
cluster_id	Yes	String	ID of the cluster to be built.
node_label_selector	No	Map<String, String>	key: label key. value: label value.
environment_id	No	String	Environment ID. See Obtaining All Environments .

Table 4-475 external_accesses

Parameter	Mandatory	Type	Description
protocol	No	String	External access type. Value: http or https.
address	No	String	External access address.
forward_port	No	Integer	External access port.

Table 4-476 tolerations

Parameter	Mandatory	Type	Description
key	No	String	Taint key.
operator	No	String	Operator. - Equal: matches the nodes with the specified taint key (mandatory) and value. If the taint value is left blank, all taints with the same key will be matched. - Exists: matches nodes with the specified taint key. In this case, the taint value cannot be specified. Not specifying any taint key means all taints are tolerated.
value	No	String	Taint value. This parameter is mandatory when operator is set to Equal .
effect	No	String	Taint policy. - NoScheduler: Only the NoSchedule taint is matched. - PreferNoSchedule: Only the PreferNoSchedule taint is matched. - NoExecute: Only the NoExecute taint is matched. - If this parameter is not set, all taint effects are matched.
toleration_seconds	No	Integer	Tolerance time window. This parameter can be set only when effect is set to NoExecute . Within the tolerance time window, pods still run on the node with taints. After the time expires, the pods will be evicted.

Table 4-477 multi_containers

Parameter	Mandatory	Type	Description
container_name	Yes	String	Container name.
source	Yes	Object	Source of the code or software package. See Table 4-472 . In multi-container mode, only images are supported.
limit_cpu	No	Number	Maximum CPU limit. Unit: core.
limit_memory	No	Number	Maximum memory size. Unit: GiB.
request_cpu	No	Number	Requested CPU resources. Unit: core.
request_memory	No	Number	Requested memory. Unit: GiB.
envs	No	Array of objects	Environment variable list. See Table 4-469 .
liveness_probe	No	Object	Container liveness probe. See Table 4-466 .
readiness_probe	No	Object	Container service probe. See Table 4-466 .
startup_probe	No	Object	Container startup probe. See Table 4-466 .
command	No	Object	Startup command. See Table 4-453 .
post_start	No	Object	Post-start processing. See Table 4-454 .
pre_stop	No	Object	Pre-stop processing. See Table 4-454 .
logs	No	Array of objects	Log collection. See Table 4-462 .
storages	No	Array of objects	Storage. See Table 4-446 .
security_context	No	Object	Security policy. See Table 4-460 .
config_encryption	No	Object	This parameter is mandatory when the encryption key is configured for a multi-container component. See Table 4-482 .

Table 4-478 ports

Parameter	Mandatory	Type	Description
name	Yes	String	Port name.
port	Yes	String	Access port.
target_port	Yes	String	Container port.

Table 4-479 update_strategy

Parameter	Mandatory	Type	Description
type	No	String	Component upgrade policy. Options: • RollingUpdate (default): New pods are created gradually, and then old pods are deleted. • Recreate: Old pods are deleted, and then new pods are created.
max_unavailable	No	• String • Integer	Maximum number of unavailable pods. That is, the number or percentage of pods that can be deleted based on the value of replica . The default value is 25% . This parameter can be set when workload_kind is set to deployment . • String (%): the maximum percentage of unavailable pods to the total number of pods. • Integer: the maximum number of unavailable pods.
max_surge	No	• String • Integer	Maximum surge. That is, the number or percentage of pods allowed based on the value of replica . The default value is 25% . This parameter can be set only when workload_kind is set to deployment . • String (%) • Integer

Parameter	Mandatory	Type	Description
termination_period_seconds	No	Integer	Scale-in time window. That is, the time window for executing commands before a workload stops. The default value is 30s.
min_ready_seconds	No	Integer	Minimum ready time of a pod. That is, the minimum ready time of a new pod without any container crash. A pod is considered available only after the minimum ready time is exceeded. The default value is 0 .
progress_deadline_seconds	No	Integer	Maximum upgrade duration. That is, the time (in seconds) the system waits for before reporting a component upgrade failure. If this parameter is specified, the value of this parameter must be greater than that of min_ready_seconds .
revision_history_limit	No	Integer	Maximum number of pods to retain. That is, the number of old pods to retain to allow rollback. The default value is 10 .

Table 4-480 configs

Parameter	Mandatory	Type	Description
config_id	Yes	String	Unified configuration ID.
mount_path	Yes	String	Mount path.

Table 4-481 auto_lts_config

Parameter	Mandatory	Type	Description
enable	Yes	Boolean	Whether to enable automatic LTS configuration.
lts_log_path	No	Array of String	Access configuration log path.

Table 4-482 config_encryption

Parameter	Mandatory	Type	Description
key_id	Yes	String	User key ID in DEW.

Response

Table 4-483 Response parameters

Parameter	Type	Description
release_plan_id	String	Release task ID.

Example Request

Edit the release task whose ID is **bb09c189-d17e-480a-a0e4-3a102b3795b0**.

Example Response

```
{
  "release_plan_id": "bb09c189-d17e-480a-a0e4-3a102b3795b0"
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.7.1.3 Deleting a Release Task Based on the Release Task ID

Function

This API is used to delete a release task based on the release task ID.

URI

DELETE /v3/{project_id}/cas/release-plans/{release_plan_id}

Table 4-484 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
release_plan_id	Yes	String	Release task ID. See Obtaining the Release Task List .

Request

Table 4-485 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

None

Example Request

None

Example Response

None

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found

Status Code	Description
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.7.1.4 Obtaining the Release Task List

Function

This API is used to obtain the release task list.

URI

GET /v3/{project_id}/cas/release-plans

Table 4-486 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .

Request

Table 4-487 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

Table 4-488 Response parameters

Parameter	Type	Description
release_plans	Array of objects	Release task list. See Table 4-489 .
count	Integer	Number of release tasks.

Table 4-489 release_plans

Parameter	Type	Description
id	String	Lane ID.
name	String	Lane name.
action	String	Release task type.
status	String	Release task status.
create_time	Integer	Creation time.
execute_time	Integer	Execution time.
creator	String	Creator.

Example Request

None

Example Response

```
{
  "count": 1,
  "release_plans": [
    {
      "name": "release-1zws2j",
      "id": "4cf4d39e-c11a-41b0-82eb-30defc376d52",
      "action": "create",
      "create_time": 1711952860132,
      "modify_time": 0,
      "status": "SUCCESS",
      "creator": "wzh",
      "execute_time": 1711952860379
    }
  ]
}
```

Status Code

Status Code	Description
200	OK

Status Code	Description
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.7.1.5 Obtaining Information About a Release Task Based on the Release Task ID

Function

This API is used to obtain information about a release task based on the release task ID.

URI

GET /v3/{project_id}/cas/release-plans/{release_plan_id}

Table 4-490 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
release_plan_id	Yes	String	Release task ID. See Obtaining the Release Task List .

Request

Table 4-491 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

Table 4-492 Response parameters

Parameter	Type	Description
id	String	Release task ID.
action	String	Release task type. • clone • upgrade • create
swimlane_id	String	Lane ID.
desc	String	Description.
components	Array of objects	Component list. See Table 4-493 .
name	String	Release task name.
fail_strategy	String	Failure strategy. • continue • stop
status	String	Release task status.
create_time	Integer	Creation time.
modify_time	Integer	Update time.
execute_time	Integer	Execution time.
creator	String	Creator.

Table 4-493 components

Parameter	Type	Description
seq	Integer	Release batch.
component_id	String	Application component ID.
name	String	Application component name. The value contains 2 to 64 characters consisting of letters, digits, hyphens (-), and underscores (_). It starts with a letter and ends with a letter or digit.
workload_name	String	Workload name.
labels	Array of objects	Label. For container-based deployment, you can query a workload by label. For VM-based deployment, you can configure a label of the corresponding microservice name to use graceful startup and shutdown. See Table 4-494 .
create_mode	String	Component creation mode. This parameter is not transferred by default, indicating to create and deploy a component.
runtime_stack	Object	Runtime. See Table 4-520 . The value can be obtained from the response of the API for Querying a Technology Stack .
environment_id	String	Environment ID.
description	String	Description. The value can contain up to 128 characters.
source	Object	Source of the code or software package. See Table 4-521 .
build	Object	Component building. This parameter is mandatory when the component is deployed in containers and no image is used. For details, see Table 4-522 .
limit_cpu	Number	Maximum CPU limit. Unit: core.
limit_memory	Number	Maximum memory size. Unit: GiB.
request_cpu	Number	Requested CPU resources. Unit: core.
request_memory	Number	Requested memory. Unit: GiB.
version	String	Component version number. The value can contain up to 32 characters. It needs to meet the following rule: $\wedge([0-9]+)(.[0-9]+){2,3}\$$

Parameter	Type	Description
envs	Array of objects	Environment variable list. See Table 4-518 .
replica	Integer	Number of instances.
storages	Array of objects	Storage. See Table 4-495 .
deploy_strategy	Object	Component deployment. See Table 4-498 .
command	Object	Startup command. See Table 4-502 .
post_start	Object	Post-start processing. See Table 4-503 .
pre_stop	Object	Pre-stop processing. See Table 4-503 .
mesher	Object	Multi-language access service mesh. This parameter is mandatory when Node.js is selected for the technology stack and the microservice engine is bound. See Table 4-504 .
timezone	String	Time zone where the component runs.
jvm_opts	String	JVM parameter. Example: -Xms256m -Xmx1024m. If there are multiple parameters, separate them by spaces. If the parameter is left blank, the default value is used.
tomcat_opts	Object	Tomcat parameter. This parameter is valid when you use the Tomcat technology stack. See Table 4-505 .
host_aliases	Array of objects	Host alias. See Table 4-506 .
dns_policy	String	DNS policy. • Default: inherits the domain name resolution configuration of the node where the pod is located. • ClusterFirst: adds the domain name resolution configuration. • ClusterFirstWithHostNet: takes effect only for pods running in hostNetWork mode. • None: replaces the domain name resolution policy.
dns_config	Object	DNS configuration. See Table 4-507 .

Parameter	Type	Description
workload_kind	String	Workload type. • deployment • statefulset • daemonset
security_context	Object	Security policy. See Table 4-509 .
logs	Array of objects	Log collection. See Table 4-511 .
custom_metric	Object	Custom metrics. See Table 4-512 .
affinity	Array of objects	Affinity. See Table 4-513 .
anti_affinity	Array of objects	Anti-affinity. See Table 4-513 .
liveness_probe	Object	Component liveness probe. See Table 4-515 .
readiness_probe	Object	Component service probe. See Table 4-515 .
startup_probe	Object	Component startup probe. See Table 4-515 .
refer_resources	Array of objects	Associated resource. See Table 4-516 .
external_accesses	Array of Objects	External access. See Table 4-524 .
tolerations	Array of Objects	Tolerance policy. See Table 4-525 .
container_name	String	Container name.
update_strategy	Object	Component upgrade strategy. See Table 4-528 .
multi_containers	Array of Objects	Multi-container configuration. See Table 4-526 .
service_name	String	Service name configured for the inter-instance discovery of a StatefulSet.
ports	Array of Objects	Port configured for the inter-instance discovery of a StatefulSet.

Parameter	Type	Description
workload_content	String	Workload content.
configs	Array of Objects	Unified component configuration. See Table 4-529 .
enable_sermant_injection	Boolean	Whether to enable Sermant.
enable_sermant_spring_register	Boolean	Whether Spring Cloud Huawei microservices need to be registered with registry centers through Sermant.
enable_sermant_dubbo_register	Boolean	Whether Dubbo microservices need to be registered with registry centers through Sermant.
enable_sermant_config	Boolean	Whether to enable the Sermant dynamic configuration plug-in.
cell_id	String	Deployment cell ID.
swimlane_id	String	Lane ID.
sermantAgentInjectActionFlag	Boolean	Whether JDK 17 and JDK 21 automatically add the following JVM environment variables: JDK 17: --add-opens java.base/java.lang=ALL-UNNAMED --add-opens java.base/java.net=ALL-UNNAMED --add-opens java.base/java.math=ALL-UNNAMED --add-opens java.base/sun.net.www=ALL-UNNAMED --add-opens java.base/sun.net.www.protocol.http=ALL-UNNAMED JDK 21: --add-opens java.base/java.lang=ALL-UNNAMED --add-opens java.base/java.net=ALL-UNNAMED --add-opens java.base/java.math=ALL-UNNAMED --add-opens java.base/sun.net.www=ALL-UNNAMED --add-opens java.base/sun.net.www.protocol.http=ALL-UNNAMED -Dnet.bytebuddy.experimental=true
sys_config	Map<String, String>	Uses a compressed package to deploy component configuration items in the VM environment.
image_pull_secrets	Array of String	List of image pull secrets.
auto_lts_config	Object	Automatic LTS configuration. See Table 4-530 .
config_encryption	Object	This parameter is mandatory when the encryption key is configured for a single-container component. See Table 4-531 .

Table 4-494 labels

Parameter	Type	Description
key	String	Label name.
value	String	Label value.

Table 4-495 storages

Parameter	Type	Description
type	String	Data storage type. • HostPath: host path for local disk mounting. • EmptyDir: temporary directory for local disk mounting. • ConfigMap: configuration item for local disk mounting. • Secret: secret for local disk mounting. • PersistentVolumeClaim: cloud storage mounting.
name	String	Name of the storage disk.
parameters	Object	For details about the parameters of each data storage type, see Table 4-496 .
mounts	Array of objects	Mount path of data storage. See Table 4-497 .

Table 4-496 component_storage_parameters

Parameter	Type	Description
path	String	Host path. This parameter is applicable to the HostPath storage type. This parameter is mandatory when type is set to HostPath .
name	String	Name of a configuration item, secret, or PVC, which is applicable to ConfigMap, Secret, or PersistentVolumeClaim storage, respectively. This parameter is mandatory when type is set to ConfigMap , Secret , or PersistentVolumeClaim .
default_mode	Integer	Mounting permission. The value is in decimal format. Example: 384. This parameter is applicable to ConfigMap and secret storage.

Parameter	Type	Description
medium	String	This parameter is applicable to EmptyDir storage. If it is not specified, the default disk medium is used. If it is set to memory , memory storage is enabled.

Table 4-497 mounts

Parameter	Type	Description
path	String	Mount path.
sub_path	String	Subpath of the mount path.
read_only	Boolean	Read only or not.

Table 4-498 deploy_strategy

Parameter	Type	Description
type	String	Deployment type. - OneBatchRelease - RollingRelease - GrayRelease
rolling_release	Object	Rolling deployment parameter, which is mandatory when type is set to RollingRelease . For details, see Table 4-499 .
gray_release	Object	Dark launch upgrade parameter, which is mandatory when type is set to GrayRelease . For details, see Table 4-500 .

Table 4-499 rolling_release

Parameter	Type	Description
batches	Integer	Deployment batches.

Table 4-500 gray_release

Parameter	Type	Description
type	String	Dark launch policy. • WEIGHT: dark launch by traffic. • CONTENT: dark launch by content.
first_batch_weight	Integer	Proportion of first-batch dark launch traffic. This parameter is mandatory when type is set to WEIGHT .
first_batch_replica	Integer	Number of first-batch dark launch instances. This parameter is mandatory when type is set to WEIGHT .
remaining_batches	Integer	Number of batches for remaining instance deployment. After the first batch is completed, the remaining instances will be deployed in specified batches. Example: If there are five instances remaining to be deployed in three batches, these five instances will be upgraded based on 2:2:1. This parameter is mandatory when type is set to WEIGHT .
deployment_mode	Integer	Deployment mode. • 1: The upstream is a microservice gateway. • 3: The upstream is a microservice. • 4: The upstream is a load balancer.
replica_surge_mode	String	Mode of adding a dark launch instance. This parameter is mandatory when type is set to CONTENT . • MIRROR: blue-green. • EXTRA: canary (increase, then decrease) • NOSURGE: canary (decrease, then increase)
rule_match_mode	String	Effective mode of a dark launch rule. This parameter is mandatory when type is set to CONTENT . • ALL: All conditions are met. • ANY: Any condition is met.
rules	Object	Dark launch rule. See Table 4-501 . This parameter is mandatory when type is set to CONTENT .

Table 4-501 gray_rules

Parameter	Type	Description
type	String	Match type. Currently, only HEADER is supported.
key	String	Parameter name.
value	String	Condition value.
condition	String	Condition type. • EQUAL • MATCH • IN: enumeration

Table 4-502 command

Parameter	Type	Description
command	Array of String	Command for controlling container running.
args	Array of String	Parameter for controlling container running. Example: -port=8080. If there are multiple parameters, separate them by line breaks.

Table 4-503 component_lifecycle

Parameter	Type	Description
type	String	Processing method. • http • command
scheme	String	HTTP request type, which can be HTTP or HTTPS. This parameter is valid only when type is set to http .
host	String	Custom IP address. If this parameter is left blank, the pod IP address is used. This parameter is valid only when type is set to http .
port	Integer	Port number. This parameter is valid only when type is set to http .
path	String	Request path. This parameter is valid only when type is set to http .

Parameter	Type	Description
command	Array of String	Command list. This parameter is valid only when type is set to command .

Table 4-504 mesher

Parameter	Type	Description
port	Integer	Process listening port.

Table 4-505 tomcat_opt

Parameter	Type	Description
server_xml	String	1. Listening port in the server.xml file will take effect even if it has been configured in public network access. 2. The custom Tomcat configuration may conflict with the existing port. Check whether the custom port is occupied.

Table 4-506 host_aliases

Parameter	Type	Description
ip	String	IP address.
hostname	Array of String	Host alias.

Table 4-507 dns_config

Parameter	Type	Description
nameservers	Array of String	IP address list of the DNS server of the pod.
searches	Array of String	(Optional) List of DNS search domains used to search for host names in the pod.
options	Array of objects	(Optional) List of objects. Each object may have a name (mandatory) and a value (optional). The contents of the objects here will be merged into the options generated from the specified DNS policy. Duplicate entries will be deleted. See Table 4-508 .

Table 4-508 options

Parameter	Type	Description
name	String	Operation name.
value	String	Operation value.

Table 4-509 security_context

Parameter	Type	Description
run_as_user	Integer	User for running containers. Example: To run containers as user root , set the user ID to 0 .
run_as_group	Integer	Owner group specified for running container processes.
capabilities	Object	Capability set. See Table 4-510 .

Table 4-510 capabilities

Parameter	Type	Description
add	Array of String	Grant the Linux permission to the running user.
drop	Array of String	Remove the Linux permission of the running user.

Table 4-511 logs

Parameter	Type	Description
log_path	String	Container log path.
rotate	String	Interval for dumping logs.
host_path	String	Mounted host path.

Parameter	Type	Description
host_extend_path	String	Host extension path. By extending the host path, you can distinguish mounting from different containers in the same host path. • None: The extended path is not used. • PodUID: extend the host path based on the pod ID. • PodName: extend the host path based on the pod name. • PodUID/ContainerName: extend the host path based on the pod ID and container name. • PodName/ContainerName: extend the host path based on the pod name and container name.

Table 4-512 custom_metric

Parameter	Type	Description
path	String	Collection path. Example: /metrics.
port	Integer	Collection port. Example: 9090.
dimensions	String	Monitoring dimension. Example: cpu_usage,mem_usage.

Table 4-513 component_affinity

Parameter	Type	Description
kind	String	Object kind. • node • pod
condition	String	Policy type. • required: The requirement must be met. • preferred: The requirement is met as much as possible.
match_expressions	Array of objects	Scheduling policy rule. See Table 4-514 .
weight	String	Weight. • The value is 100 when condition is set to required. • Enter an integer ranging from 1 to 100 when condition is set to preferred.

Table 4-514 match_expressions

Parameter	Type	Description
key	String	Condition key.
value	String	Condition value.
operation	String	Operation.

Table 4-515 component_probe

Parameter	Type	Description
type	String	Type. Value: http, tcp, or command.
delay	Integer	Interval between the startup and detection.
timeout	Integer	Detection timeout interval.
scheme	String	Request type. Value: HTTP or HTTPS. This parameter is valid only when type is set to http .
host	String	Custom IP address. If this parameter is left blank, the pod IP address is used. This parameter is valid only when type is set to http .
port	Integer	Port number. This parameter is valid only when type is set to http or tcp .
path	String	Request path. This parameter is valid only when type is set to http .
command	Array of String	Command list. This parameter is valid only when type is set to command .

Table 4-516 refer_resources

Parameter	Type	Description
id	String	Resource ID.

Parameter	Type	Description
type	String	Resource type. • vpc • eip • elb • cce • ecs • dcs • rds • cse: ServiceComb engine. • nacos: registry/configuration center. • customServer • pvc • gaussdb • dms_rocketmq • css_es: Elasticsearch
parameters	Object	Resource parameters. See Table 4-517 .

Table 4-517 refer_resource_parameter

Parameter	Type	Description
namespace	String	Namespace.
name	String	Resource name.
capacity	Integer	Storage capacity, in GiB.
type	String	Volume declaration type. • efs • obs • sfs • evs
class	String	Volume declaration subtype. • csi-sfsturbo: fixed when type is set to efs. • csi-obs: fixed when type is set to obs. • csi-nas: fixed when type is set to sfs. • csi-disk: fixed when type is set to evs.
obs_volume_type	String	OBS volume type. Default value: STANDARD.

Parameter	Type	Description
access_mode	String	Storage volume access mode. - ReadWriteOnce: A volume can be mounted as read-write by a single node. - ReadWriteMany: A volume can be mounted as read-write by many nodes.
availableZone	String	AZ.
volume_type	String	Volume type.

Table 4-518 envs

Parameter	Type	Description
name	String	Variable name.
value	String	Variable value.
value_from	Object	Variable reference. See Table 4-519 .

Table 4-519 value_from

Parameter	Type	Description
reference_type	String	Reference type. - configMapKey - secretKey
name	String	Name of the configuration item or secret.
key	String	Key of the configuration item or secret.
optional	Boolean	Whether the key of the configuration item or secret is mandatory.

Table 4-520 runtime_stack

Parameter	Type	Description
name	String	Technology stack name.

Parameter	Type	Description
type	String	Technology stack type. Value: Java, Tomcat, Nodejs, Php, Docker, or Python. - For VM-based deployment, Java, Tomcat, and Nodejs are supported. - For container-based deployment, Java, Tomcat, Nodejs, Php, and Docker are supported.
version	String	Technology stack version.
deploy_mode	String	Deployment mode. - container - virtualmachine

Table 4-521 source

Parameter	Mandatory	Type	Description
kind	Yes	String	Component source kind. - code - package - image
version	No	String	Version number.
url	No	String	Package address. This parameter is mandatory when kind is set to package or image .
storage	No	String	Storage mode. - devcloud: CodeArts release repository - obs - http: custom file address - swr This parameter is mandatory when kind is set to package or image .
http_username	No	String	Username. This parameter is mandatory when storage is set to http . Username and password authentication is supported.
http_password	No	String	Password. This parameter is mandatory when storage is set to http . Username and password authentication is supported.

Parameter	Mandatory	Type	Description
http_command	No	String	Address of the custom software package. This parameter is mandatory when storage is set to http .
http_headers	No	Map<String, String>	This parameter is mandatory when storage is set to http . You can customize request headers as key-value pairs, for example, {"key": "xx", "value": "xx"}.
auth	No	String	Authentication mode. Value: iam or none. Default value: iam.
repo_auth	No	String	Authorization name, which can be obtained from the authorization list. For details, see Obtaining a Git Repository Authorization List .
repo_namespace	No	String	Namespace of the code repository.
repo_ref	No	String	Code branch or tag. Default value: master.
repo_type	No	String	Code repository type. Value: GitHub, BitBucket, or GitLab.
web_url	No	String	Code redirection URL. Example: https://github.com/example/demo.git.
repo_url	No	String	Code repository URL. Example: https://github.com/example/demo.git.
commit_id	No	String	In the code repository, commit ID is the unique identifier of each commit and is used to identify and reference a commit. The value contains 8 to 40 characters and starts with a lowercase letter or digit.

Table 4-522 build

Parameter	Type	Description
parameters	Map<String, Object>	This parameter is provided only when no ID is available during build creation. See Table 4-523 .

Table 4-523 parameters

Parameter	Type	Description
build_cmd	String	Compilation command. By default: 1. When build.sh exists in the root directory, the command is ./build.sh . 2. When build.sh does not exist in the root directory, the command varies depending on the OS. Example: • Java and Tomcat: mvn clean package • Nodejs: npm build
dockerfile_path	String	Address of the Docker file. By default, the Docker file is in the root directory (.).
artifact_name_space	String	Build archive organization. Default value: cas_{project_id}.
cluster_id	String	ID of the cluster to be built.
node_label_selector	Map<String, String>	key: label key. value: label value.
environment_id	String	Environment ID. See Obtaining All Environments .

Table 4-524 external_accesses

Parameter	Type	Description
protocol	String	External access type. Value: http or https.
address	String	External access address.
forward_port	Integer	External access port.

Table 4-525 tolerations

Parameter	Type	Description
key	String	Taint key.

Parameter	Type	Description
operator	String	Operator. • Equal: matches the nodes with the specified taint key (mandatory) and value. If the taint value is left blank, all taints with the same key will be matched. • Exists: matches nodes with the specified taint key. In this case, the taint value cannot be specified. Not specifying any taint key means all taints are tolerated.
value	String	Taint value. This parameter is mandatory when operator is set to Equal .
effect	String	Taint policy. • NoScheduler: Only the NoSchedule taint is matched. • PreferNoSchedule: Only the PreferNoSchedule taint is matched. • NoExecute: Only the NoExecute taint is matched. • If this parameter is not set, all taint effects are matched.
toleration_seconds	Integer	Tolerance time window. This parameter can be set only when effect is set to NoExecute . Within the tolerance time window, pods still run on the node with taints. After the time expires, the pods will be evicted.

Table 4-526 multi_containers

Parameter	Type	Description
container_name	String	Container name.
source	Object	Source of the code or software package. See Table 4-521 . In multi-container mode, only images are supported.
limit_cpu	Number	Maximum CPU limit. Unit: core.
limit_memory	Number	Maximum memory size. Unit: GiB.
request_cpu	Number	Requested CPU resources. Unit: core.
request_memory	Number	Requested memory. Unit: GiB.
envs	Array of objects	Environment variable list. See Table 4-518 .

Parameter	Type	Description
liveness_probe	Object	Container liveness probe. See Table 4-515 .
readiness_probe	Object	Container service probe. See Table 4-515 .
startup_probe	Object	Container startup probe. See Table 4-515 .
command	Object	Startup command. See Table 4-502 .
post_start	Object	Post-start processing. See Table 4-503 .
pre_stop	Object	Pre-stop processing. See Table 4-503 .
logs	Array of objects	Log collection. See Table 4-511 .
storages	Array of objects	Storage. See Table 4-495 .
security_context	Object	Security policy. See Table 4-509 .
config_encryption	Object	This parameter is mandatory when the encryption key is configured for a multi-container component. See Table 4-531 .

Table 4-527 ports

Parameter	Type	Description
name	String	Port name.
port	String	Access port.
target_port	String	Container port.

Table 4-528 update_strategy

Parameter	Type	Description
type	String	Component upgrade policy. Options: - RollingUpdate (default): New pods are created gradually, and then old pods are deleted. - Recreate: Old pods are deleted, and then new pods are created.

Parameter	Type	Description
max_unavailable	- String - Integer	Maximum number of unavailable pods. That is, the number or percentage of pods that can be deleted based on the value of replica. The default value is 25%. This parameter can be set when workload_kind is set to deployment. - String (%): the maximum percentage of unavailable pods to the total number of pods. - Integer: the maximum number of unavailable pods.
max_surge	- String - Integer	Maximum surge. That is, the number or percentage of pods allowed based on the value of replica. The default value is 25%. This parameter can be set only when workload_kind is set to deployment. - String (%) - Integer
termination_period_seconds	Integer	Scale-in time window. That is, the time window for executing commands before a workload stops. The default value is 30s.
min_ready_seconds	Integer	Minimum ready time of a pod. That is, the minimum ready time of a new pod without any container crash. A pod is considered available only after the minimum ready time is exceeded. The default value is 0.
progress_deadline_seconds	Integer	Maximum upgrade duration. That is, the time (in seconds) the system waits for before reporting a component upgrade failure. If this parameter is specified, the value of this parameter must be greater than that of min_ready_seconds.
revision_history_limit	Integer	Maximum number of pods to retain. That is, the number of old pods to retain to allow rollback. The default value is 10.

Table 4-529 configs

Parameter	Type	Description
config_id	String	Unified configuration ID.
mount_path	String	Mount path.

Table 4-530 auto_lts_config

Parameter	Type	Description
enable	Boolean	Whether to enable automatic LTS configuration.
lts_log_path	Array of String	Access configuration log path.

Table 4-531 config_encryption

Parameter	Type	Description
key_id	String	User key ID in DEW.

Example Request

None

Example Response

```
{
  "desc": "",
  "proportion": 0,
  "components": [
    {
      "id": "30f88521-fa43-42ff-aaec-9cac4259fb16",
      "runtime_stack": {
        "url": "",
        "type": "Docker",
        "name": "Docker",
        "deploy_mode": "container",
        "version": "",
        "status": "Supported"
      },
      "source": {
        "kind": "image",
        "url": "swr.ap-southeast-1.myhuaweicloud.com/cae-group1/sch:v1",
        "storage": "swr",
        "auth": "iam"
      },
      "replica": 1,
      "version": "2024.0802.18390",
      "envs": [
        {
          "name": "MOCK_ENABLED",
          "inner": false,
          "value": "true"
        }
      ]
    }
  ]
}
```



```
    }
  ],
  "storages": [],
  "update_strategy": {
    "type": "RollingUpdate",
    "max_unavailable": "25%",
    "max_surge": "25%",
    "termination_period_seconds": 30
  },
  "command": {
    "command": [],
    "args": []
  },
  "jvm_opts": "",
  "tomcat_opts": {
    "server_xml": "",
    "http_port": 0,
    "context_path": null
  },
  "logs": [],
  "custom_metric": {
    "path": null,
    "port": 0,
    "dimensions": null
  },
  "affinity": [],
  "anti_affinity": [],
  "tolerations": [
    {
      "key": "node.kubernetes.io/not-ready",
      "operator": "Exists",
      "value": null,
      "effect": "NoExecute",
      "toleration_seconds": 300
    },
    {
      "key": "node.kubernetes.io/unreachable",
      "operator": "Exists",
      "value": null,
      "effect": "NoExecute",
      "toleration_seconds": 300
    }
  ],
  "liveness_probe": {
    "type": null,
    "delay": 0,
    "timeout": 0,
    "scheme": null,
    "host": null,
    "port": 0,
    "path": null,
    "command": null,
    "period_seconds": 0,
    "success_threshold": 0,
    "failure_threshold": 0,
    "http_headers": null
  },
  "readiness_probe": {
    "type": null,
    "delay": 0,
    "timeout": 0,
    "scheme": null,
    "host": null,
    "port": 0,
    "path": null,
    "command": null,
    "period_seconds": 0,
    "success_threshold": 0,
    "failure_threshold": 0,
  }
```

```
    "http_headers": null
  },
  "startup_probe": {
    "type": null,
    "delay": 0,
    "timeout": 0,
    "scheme": null,
    "host": null,
    "port": 0,
    "path": null,
    "command": null,
    "period_seconds": 0,
    "success_threshold": 0,
    "failure_threshold": 0,
    "http_headers": null
  },
  "refer_resources": [
    {
      "id": "7bc15618-f875-11ee-9505-0255ac100b08",
      "type": "cce",
      "refer_alias": null,
      "parameters": {
        "name": "cce-lianghuanyu",
        "namespace": "default",
        "type": "VirtualMachine"
      },
      "selectors": null
    },
    {
      "id": "ebeb26bd-3e27-49d6-bf70-bb1fa07a0500",
      "type": "cse",
      "refer_alias": null,
      "parameters": null,
      "selectors": null
    }
  ],
  "workload_kind": "deployment",
  "config_mode": "ui",
  "enable_serment_injection": false,
  "enable_serment_spring_register": false,
  "enable_serment_dubbo_register": false,
  "enable_serment_config": false,
  "sys_config": {},
  "swimlane_id": "180c158a-0b45-41b4-a8da-767e8fb8e156",
  "container_name": "sch-env-apitest-ss-lb0i2d",
  "seq": 1,
  "enterprise_project_id": "0",
  "environment_id": "48874c71-4898-442f-87fc-afff135d65d1",
  "name": "sch",
  "application_id": "f397be2a-4c49-4ed8-8144-ce13417e179d",
  "nacos_enterprise_project_id": "0"
}
},
"name": "release-jxams3",
"fail_strategy": "stop",
"action": "clone",
"swimlane_id": "180c158a-0b45-41b4-a8da-767e8fb8e156",
"id": "716eda33-dbc6-47f3-953f-29119edd5e5c",
"create_time": 1722933924775,
"modify_time": 0,
"status": "SUCCESS",
"execute_time": 1722933924876,
"creator": "wzh"
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.7.1.6 Obtaining the Deployment Information About a Release Task Based on the Release Task ID

Function

This API is used to obtain the deployment information about a release task based on the release task ID.

URI

GET /v3/{project_id}/cas/release-plans/{release_plan_id}/deploy-info

Table 4-532 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
release_plan_id	Yes	String	Release task ID. See Obtaining the Release Task List .

Request

Table 4-533 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

Table 4-534 Response parameters

Parameter	Type	Description
id	String	Release task ID.
components	Array of objects	Component release information. See Table 4-535 .

Table 4-535 job_info

Parameter	Type	Description
job_id	String	Job ID.
status	String	Component status.
job_status	String	Job status.
name	String	Component name.
id	String	Component ID.
seq	String	Release batch.
version	String	Component version.
rollback_job_id	String	Rollback job ID.
start_time	Long	Start time of component release.
end_time	Long	End time of component release.
rollback_job_status	String	Rollback job status.
rollback_start_time	String	Start time of component rollback.

Parameter	Type	Description
rollback_end_time	String	End time of component rollback.
fail_message	String	Failure message.
rollback_fail_message	String	Rollback failure message.

Example Request

None

Example Response

```
{
  "id": "716eda33-dbc6-47f3-953f-29119edd5e5c",
  "pre_tasks": [],
  "components": [
    {
      "job_id": "JOB6999c66d-cb39-4c46-9f98-893b704300e0",
      "status": "RUNNING",
      "job_status": "SUCCEEDED",
      "name": "sch",
      "id": "df49e875-000c-4d69-8f45-c4f384549e1b",
      "seq": 1,
      "version": "2024.0802.18390",
      "rollback_job_id": null,
      "start_time": 1722933924962,
      "end_time": 1722933950018,
      "rollback_job_status": null,
      "rollback_start_time": 0,
      "rollback_end_time": 0,
      "fail_message": null,
      "rollback_fail_message": null
    }
  ]
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.7.1.7 Executing a Release Task Based on the Release Task ID

Function

This API is used to execute a release task based on the release task ID.

URI

POST /v3/{project_id}/cas/release-plans/{release_plan_id}/execute

Table 4-536 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
release_plan_id	Yes	String	Release task ID. See Obtaining the Release Task List .

Request

Table 4-537 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

None

Example Request

None

Example Response

None

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.7.1.8 Rolling Back a Release Task Based on the Release Task ID

Function

This API is used to roll back a release task based on the release task ID.

URI

POST /v3/{project_id}/cas/release-plans/{release_plan_id}/rollback

Table 4-538 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
release_plan_id	Yes	String	Release task ID. See Obtaining the Release Task List .

Request

Table 4-539 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

None

Example Request

None

Example Response

None

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.7.1.9 Aborting a Release Task Based on the Release Task ID

Function

This API is used to abort a release task that is being executed based on the release task ID.

URI

POST /v3/{project_id}/cas/release-plans/{release_plan_id}/abort

Table 4-540 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
release_plan_id	Yes	String	Release task ID. See Obtaining the Release Task List .

Request

Table 4-541 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

None

Example Request

None

Example Response

None

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found

Status Code	Description
500	Internal Server Error

Error Code

For details, see [ServiceStage Error Codes](#).

4.7.2 New Release Management

4.7.2.1 Creating a Release Task

Function

This API is used to create a release task.

URI

POST /v3/{project_id}/pipeline/pipelines

Table 4-542 Path parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .

Request

Table 4-543 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Table 4-544 Request body parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Release task name.
description	No	String	Release task description.
model_version	Yes	Integer	Release task model. Only 2 is supported, indicating the new model.
config_mode	Yes	String	Release task deployment mode. • ui • template
sources	No	List of Object	Template package list. This parameter is mandatory when config_mode is set to template . See Table 4-545 .
enterprise_project_id	No	String	Enterprise project ID. If this parameter is not set, the default enterprise project is default and the ID is 0 .
stages	No	List of Object	Stage parameters. This parameter is mandatory when config_mode is ui . See Table 4-548 .

Table 4-545 template_sources

Parameter	Mandatory	Type	Description
template_source	Yes	Object	Template package information. See Table 4-546 .
input_values	No	Map<String, String>	Template package input information.

Table 4-546 template_source

Parameter	Mandatory	Type	Description
storage	Yes	String	Storage mode of the component template package. • devcloud: CodeArts release repository • obs • http: custom file address
url	Yes	String	Address of the component template package. This parameter is valid only when the component template package storage mode is http .
http_username	No	String	Username. This parameter is valid only when the component template package storage mode is http and the security authentication mode is username and password authentication.
http_password	No	String	Password. This parameter is valid only when the component template package storage mode is http and the security authentication mode is username and password authentication.
http_headers	No	Array of objects	Request header. See Table 4-547 . This parameter is valid only when the component template package storage mode is http and the security authentication mode is custom header authentication.

Table 4-547 http_headers

Parameter	Mandatory	Type	Description
key	Yes	String	Request key.
value	Yes	String	Request value.

Table 4-548 stages

Parameter	Mandatory	Type	Description
name	Yes	String	Stage name, which is unique in a release task.
display_name	No	String	Display name of a stage. If this parameter is left blank, the value is the same as the stage name.
task_group	Yes	Array of String	Task group. The parameters in the task group must correspond to the group_index field in Table 4-549 .
tasks	Yes	Array of object	Task parameters. See Table 4-549 .

Table 4-549 tasks

Parameter	Mandatory	Type	Description
name	Yes	String	Task name, which is unique in a single stage.
display_name	No	String	Display name of a task. If this parameter is left blank, the value is the same as the task name.
order_index	Yes	Integer	Task execution sequence. Tasks with the same group_index value belong to the same task group. The execution sequence is unique in a task group.
group_index	Yes	String	Task group. If the task groups are the same, the tasks are in the same group. The value is the same as that of task_group in Table 4-548 .
task_type_id	Yes	Object	Task execution parameters. See Table 4-550 .

Parameter	Mandatory	Type	Description
input	No	Object	Task execution parameter, which is determined based on the task type. - For a build task in Table 4-550, see Table 4-551. - For a checkpoint task in Table 4-550, see Table 4-552. - For an application task in Table 4-550, see Table 4-554. - For a component task in Table 4-550, see Table 4-555.

Table 4-550 task_type_id

Parameter	Mandatory	Type	Description
category	Yes	String	Task type. - Build - CheckPoint - Environment - Component - Application
type	Yes	String	Task execution type. - create - update - upsert: create or modify. - approval

Table 4-551 build

Parameter	Mandatory	Type	Description
name	Yes	String	Build task name.
id	Yes	String	Build task ID.
enterprise_project_id	No	String	Enterprise project ID of a build task. If this parameter is not set, the default enterprise project is default and the ID is 0 .

Table 4-552 checkpoint

Parameter	Mandatory	Type	Description
mode	No	String	If this field is left blank, no person needs to be specified for approval. - all: If all approvers approve the application, the application is approved. - any: If any approver approves the application, the application is approved.
users	No	List of Object	Approver. If this parameter is left blank, no approver is required. See Table 4-553 .

Table 4-553 users

Parameter	Mandatory	Type	Description
id	Yes	String	User ID.
label	No	String	User label, which is displayed on the frontend.
name	No	String	User name, which is displayed on the frontend.

Table 4-554 application

Parameter	Mandatory	Type	Description
name	Yes	String	Application name.
description	Yes	String	Application description.
enterprise_project_id	No	String	Enterprise project ID. If this parameter is not set, the default enterprise project is default and the ID is 0 .

Table 4-555 component

Parameter	Mandatory	Type	Description
id	No	String	Application component ID.
name	Yes	String	Application component name. The value contains 2 to 64 characters consisting of letters, digits, hyphens (-), and underscores (_). It starts with a letter and ends with a letter or digit.
workload_name	No	String	Workload name.
labels	No	Array of objects	Label. For container-based deployment, you can query a workload by label. For VM-based deployment, you can configure a label of the corresponding microservice name to use graceful startup and shutdown. See Table 4-556 .
runtime_stack	Yes	Object	Runtime. See Table 4-582 . The value can be obtained from the response of the API for Querying a Technology Stack .
environment_id	Yes	String	Environment ID.
description	No	String	Description. The value can contain up to 128 characters.
source	Yes	Object	Source of the code or software package. See Table 4-583 .
build	No	Object	Component building. This parameter is mandatory when the component is deployed in containers and no image is used. For details, see Table 4-584 .
limit_cpu	No	Number	Maximum CPU limit. Unit: core.
limit_memory	No	Number	Maximum memory size. Unit: GiB.
request_cpu	No	Number	Requested CPU resources. Unit: core.
request_memory	No	Number	Requested memory. Unit: GiB.

Parameter	Mandatory	Type	Description
version	Yes	String	Component version number. The value can contain up to 32 characters. It needs to meet the following rule: <code>^([0-9]+)(.[0-9]+){2,3}\$</code>
envs	No	Array of objects	Environment variable list. See Table 4-580 .
replica	Yes	Integer	Number of instances.
storages	No	Array of objects	Storage. See Table 4-557 .
deploy_strategy	No	Object	Component deployment. See Table 4-560 .
command	No	Object	Startup command. See Table 4-564 .
post_start	No	Object	Post-start processing. See Table 4-565 .
pre_stop	No	Object	Pre-stop processing. See Table 4-565 .
mesher	No	Object	Multi-language access service mesh. This parameter is mandatory when Node.js is selected for the technology stack and the microservice engine is bound. See Table 4-566 .
timezone	No	String	Time zone where the component runs.
jvm_opts	No	String	JVM parameter. Example: -Xms256m -Xmx1024m. If there are multiple parameters, separate them by spaces. If the parameter is left blank, the default value is used.
tomcat_opts	No	Object	Tomcat parameter. This parameter is valid when you use the Tomcat technology stack. See Table 4-567 .
host_aliases	No	Array of objects	Host alias. See Table 4-568 .

Parameter	Mandatory	Type	Description
dns_policy	No	String	DNS policy. • Default: inherits the domain name resolution configuration of the node where the pod is located. • ClusterFirst: adds the domain name resolution configuration. • ClusterFirstWithHostNet: takes effect only for pods running in hostNetWork mode. • None: replaces the domain name resolution policy.
dns_config	No	Object	DNS configuration. See Table 4-569 .
workload_kind	No	String	Workload type. • deployment • statefulset • daemonset
security_context	No	Object	Security policy. See Table 4-571 .
logs	No	Array of objects	Log collection. See Table 4-573 .
custom_metric	No	Object	Custom metrics. See Table 4-574 .
affinity	No	Array of objects	Affinity. See Table 4-575 .
anti_affinity	No	Array of objects	Anti-affinity. See Table 4-575 .
liveness_probe	No	Object	Component liveness probe. See Table 4-577 .
readiness_probe	No	Object	Component service probe. See Table 4-577 .
startup_probe	No	Object	Component startup probe. See Table 4-577 .

Parameter	Mandatory	Type	Description
refer_resources	Yes	Array of objects	Associated resource. See Table 4-578 .
external_accesses	No	Array of Objects	External access. See Table 4-586 .
tolerations	No	Array of Objects	Tolerance policy. See Table 4-587 .
container_name	No	String	Container name.
update_strategy	No	Object	Component upgrade strategy. See Table 4-590 .
multi_containers	No	Array of Objects	Multi-container configuration. See Table 4-588 .
service_name	No	String	Service name configured for the inter-instance discovery of a StatefulSet.
ports	No	Array of Objects	Port configured for the inter-instance discovery of a StatefulSet.
workload_content	No	String	Workload content.
configs	No	Array of Objects	Unified component configuration. See Table 4-591 .
config_mode	No	String	Workload configuration mode.
enable_serment_injection	No	Boolean	Whether to enable Serment.
enable_serment_spring_register	No	Boolean	Whether Spring Cloud Huawei microservices need to be registered with registry centers through Serment.

Parameter	Mandatory	Type	Description
enable_serment_dubbo_register	No	Boolean	Whether Dubbo microservices need to be registered with registry centers through Serment.
enable_serment_config	No	Boolean	Whether to enable the Serment dynamic configuration plug-in.
cell_id	No	String	Deployment cell ID.
swimlane_id	No	String	Lane ID.
sermantAgentInjectActionFlag	No	Boolean	Whether JDK 17 and JDK 21 automatically add the following JVM environment variables: JDK 17: --add-opens java.base/java.lang=ALL-UNNAMED --add-opens java.base/java.net=ALL-UNNAMED --add-opens java.base/java.math=ALL-UNNAMED --add-opens java.base/sun.net.www=ALL-UNNAMED --add-opens java.base/sun.net.www.protocol.http=ALL-UNNAMED JDK 21: --add-opens java.base/java.lang=ALL-UNNAMED --add-opens java.base/java.net=ALL-UNNAMED --add-opens java.base/java.math=ALL-UNNAMED --add-opens java.base/sun.net.www=ALL-UNNAMED --add-opens java.base/sun.net.www.protocol.http=ALL-UNNAMED -Dnet.bytebuddy.experimental=true
sys_config	No	Map<String, String>	Uses a compressed package to deploy component configuration items in the VM environment.
image_pull_secrets	No	Array of String	List of image pull secrets.
auto_lts_config	No	Object	Automatic LTS configuration. See Table 4-592 .
config_encryption	No	Object	This parameter is mandatory when the encryption key is configured for a single-container component. See Table 4-593 .

Table 4-556 labels

Parameter	Mandatory	Type	Description
key	Yes	String	Label name.
value	Yes	String	Label value.

Table 4-557 storages

Parameter	Mandatory	Type	Description
type	Yes	String	Data storage type. • HostPath: host path for local disk mounting. • EmptyDir: temporary directory for local disk mounting. • ConfigMap: configuration item for local disk mounting. • Secret: secret for local disk mounting. • PersistentVolumeClaim: cloud storage mounting.
name	Yes	String	Name of the storage disk.
parameters	Yes	Object	For details about the parameters of each data storage type, see Table 4-558 .
mounts	Yes	Array of objects	Mount path of data storage. See Table 4-559 .

Table 4-558 component_storage_parameters

Parameter	Mandatory	Type	Description
path	No	String	Host path. This parameter is applicable to the HostPath storage type. This parameter is mandatory when type is set to HostPath .
name	No	String	Name of a configuration item, secret, or PVC, which is applicable to ConfigMap, Secret, or PersistentVolumeClaim storage, respectively. This parameter is mandatory when type is set to ConfigMap , Secret , or PersistentVolumeClaim .
default_mode	No	Integer	Mounting permission. The value is in decimal format. Example: 384. This parameter is applicable to ConfigMap and secret storage.

Parameter	Mandatory	Type	Description
medium	No	String	This parameter is applicable to EmptyDir storage. If it is not specified, the default disk medium is used. If it is set to memory , memory storage is enabled.

Table 4-559 mounts

Parameter	Mandatory	Type	Description
path	Yes	String	Mount path.
sub_path	Yes	String	Subpath of the mount path.
read_only	Yes	Boolean	Read only or not.

Table 4-560 deploy_strategy

Parameter	Mandatory	Type	Description
type	Yes	String	Deployment type. - OneBatchRelease - RollingRelease - GrayRelease
rolling_release	No	Object	Rolling deployment parameter, which is mandatory when type is set to RollingRelease . For details, see Table 4-561 .
gray_release	No	Object	Dark launch upgrade parameter, which is mandatory when type is set to GrayRelease . For details, see Table 4-562 .

Table 4-561 rolling_release

Parameter	Mandatory	Type	Description
batches	Yes	Integer	Deployment batches.

Table 4-562 gray_release

Parameter	Mandatory	Type	Description
type	Yes	String	Dark launch policy. - WEIGHT: dark launch by traffic. - CONTENT: dark launch by content.
first_batch_weight	Yes	Integer	Proportion of first-batch dark launch traffic. This parameter is mandatory when type is set to WEIGHT .
first_batch_replica	Yes	Integer	Number of first-batch dark launch instances. This parameter is mandatory when type is set to WEIGHT .
remaining_batches	Yes	Integer	Number of batches for remaining instance deployment. After the first batch is completed, the remaining instances will be deployed in specified batches. Example: If there are five instances remaining to be deployed in three batches, these five instances will be upgraded based on 2:2:1. This parameter is mandatory when type is set to WEIGHT .
deployment_mode	No	Integer	Deployment mode. 1: The upstream is a microservice gateway. 3: The upstream is a microservice. 4: The upstream is a load balancer.
replica_surge_mode	No	String	Mode of adding a dark launch instance. This parameter is mandatory when type is set to CONTENT . - MIRROR: blue-green. - EXTRA: canary (increase, then decrease) - NOSURGE: canary (decrease, then increase)
rule_match_mode	No	String	Effective mode of a dark launch rule. This parameter is mandatory when type is set to CONTENT . - ALL: All conditions are met. - ANY: Any condition is met.
rules	No	Object	Dark launch rule. See Table 4-563 . This parameter is mandatory when type is set to CONTENT .

Table 4-563 gray_rules

Parameter	Mandatory	Type	Description
type	Yes	String	Match type. Currently, only HEADER is supported.
key	Yes	String	Parameter name.
value	Yes	String	Condition value.
condition	Yes	String	Condition type. • EQUAL • MATCH • IN: enumeration

Table 4-564 command

Parameter	Mandatory	Type	Description
command	No	Array of String	Command for controlling container running.
args	No	Array of String	Parameter for controlling container running. Example: -port=8080. If there are multiple parameters, separate them by line breaks.

Table 4-565 component_lifecycle

Parameter	Mandatory	Type	Description
type	Yes	String	Processing method. • http • command
scheme	No	String	HTTP request type, which can be HTTP or HTTPS. This parameter is valid only when type is set to http .

Parameter	Mandatory	Type	Description
host	No	String	Custom IP address. If this parameter is left blank, the pod IP address is used. This parameter is valid only when type is set to http .
port	No	Integer	Port number. This parameter is valid only when type is set to http .
path	No	String	Request path. This parameter is valid only when type is set to http .
command	No	Array of String	Command list. This parameter is valid only when type is set to command .

Table 4-566 mesher

Parameter	Mandatory	Type	Description
port	Yes	Integer	Process listening port.

Table 4-567 tomcat_opt

Parameter	Mandatory	Type	Description
server_xml	Yes	String	- Listening port in the server.xml file will take effect even if it has been configured in public network access. - The custom Tomcat configuration may conflict with the existing port. Check whether the custom port is occupied.

Table 4-568 host_aliases

Parameter	Mandatory	Type	Description
ip	Yes	String	IP address.
hostname	Yes	Array of String	Host alias.

Table 4-569 dns_config

Parameter	Mandatory	Type	Description
nameservers	No	Array of String	IP address list of the DNS server of the pod.
searches	No	Array of String	(Optional) List of DNS search domains used to search for host names in the pod.
options	No	Array of objects	(Optional) List of objects. Each object may have a name (mandatory) and a value (optional). The contents of the objects here will be merged into the options generated from the specified DNS policy. Duplicate entries will be deleted. See Table 4-570 .

Table 4-570 options

Parameter	Mandatory	Type	Description
name	Yes	String	Operation name.
value	No	String	Operation value.

Table 4-571 security_context

Parameter	Mandatory	Type	Description
run_as_user	No	Integer	User for running containers. Example: To run containers as user root , set the user ID to 0 .
run_as_group	No	Integer	Owner group specified for running container processes.
capabilities	No	Object	Capability set. See Table 4-572 .

Table 4-572 capabilities

Parameter	Mandatory	Type	Description
add	No	Array of String	Grant the Linux permission to the running user.

Parameter	Mandatory	Type	Description
drop	No	Array of String	Remove the Linux permission of the running user.

Table 4-573 logs

Parameter	Mandatory	Type	Description
log_path	Yes	String	Container log path.
rotate	Yes	String	Interval for dumping logs.
host_path	Yes	String	Mounted host path.
host_extend_path	Yes	String	Host extension path. By extending the host path, you can distinguish mounting from different containers in the same host path. • None: The extended path is not used. • PodUID: extend the host path based on the pod ID. • PodName: extend the host path based on the pod name. • PodUID/ContainerName: extend the host path based on the pod ID and container name. • PodName/ContainerName: extend the host path based on the pod name and container name.

Table 4-574 custom_metric

Parameter	Mandatory	Type	Description
path	Yes	String	Collection path. Example: /metrics.
port	Yes	Integer	Collection port. Example: 9090.
dimensions	Yes	String	Monitoring dimension. Example: cpu_usage,mem_usage.

Table 4-575 component_affinity

Parameter	Mandatory	Type	Description
kind	No	String	Object kind. • node • pod
condition	No	String	Policy type. • required: The requirement must be met. • preferred: The requirement is met as much as possible.
match_expressions	No	Array of objects	Scheduling policy rule. See Table 4-576 .
weight	No	String	Weight. • The value is 100 when condition is set to required. • Enter an integer ranging from 1 to 100 when condition is set to preferred.

Table 4-576 match_expressions

Parameter	Mandatory	Type	Description
key	Yes	String	Condition key.
value	Yes	String	Condition value.
operation	Yes	String	Operation.

Table 4-577 component_probe

Parameter	Mandatory	Type	Description
type	Yes	String	Type. Value: http, tcp, or command.
delay	Yes	Integer	Interval between the startup and detection.
timeout	Yes	Integer	Detection timeout interval.
scheme	No	String	Request type. Value: HTTP or HTTPS. This parameter is valid only when type is set to http .

Parameter	Mandatory	Type	Description
host	No	String	Custom IP address. If this parameter is left blank, the pod IP address is used. This parameter is valid only when type is set to http .
port	No	Integer	Port number. This parameter is valid only when type is set to http or tcp .
path	No	String	Request path. This parameter is valid only when type is set to http .
command	No	Array of String	Command list. This parameter is valid only when type is set to command .

Table 4-578 refer_resources

Parameter	Mandatory	Type	Description
id	Yes	String	Resource ID.
type	Yes	String	Resource type. • vpc • eip • elb • cce • ecs • dcs • rds • cse: ServiceComb engine. • nacos: registry/configuration center. • customServer • pvc • gaussdb • dms_rocketmq • css_es: Elasticsearch
parameters	No	Object	Resource parameters. See Table 4-579 .

Table 4-579 refer_resource_parameter

Parameter	Mandatory	Type	Description
namespace	No	String	Namespace.
name	No	String	Resource name.
capacity	No	Integer	Storage capacity, in GiB.
type	No	String	Volume declaration type. • efs • obs • sfs • evs
class	No	String	Volume declaration subtype. • csi-sfsturbo: fixed when type is set to efs. • csi-obs: fixed when type is set to obs. • csi-nas: fixed when type is set to sfs. • csi-disk: fixed when type is set to evs.
obs_volume_type	No	String	OBS volume type. Default value: STANDARD.
access_mode	No	String	Storage volume access mode. • ReadWriteOnce: A volume can be mounted as read-write by a single node. • ReadWriteMany: A volume can be mounted as read-write by many nodes.
availableZone	No	String	AZ.
volume_type	No	String	Volume type.

Table 4-580 envs

Parameter	Mandatory	Type	Description
name	Yes	String	Variable name.
value	No	String	Variable value.
value_from	No	Object	Variable reference. See Table 4-581 .

Table 4-581 value_from

Parameter	Mandatory	Type	Description
reference_type	Yes	String	Reference type. - configMapKey - secretKey
name	Yes	String	Name of the configuration item or secret.
key	No	String	Key of the configuration item or secret.
optional	No	Boolean	Whether the key of the configuration item or secret is mandatory.

Table 4-582 runtime_stack

Parameter	Mandatory	Type	Description
name	Yes	String	Technology stack name.
type	Yes	String	Technology stack type. Value: Java, Tomcat, Nodejs, Php, Docker, or Python. - For VM-based deployment, Java, Tomcat, and Nodejs are supported. - For container-based deployment, Java, Tomcat, Nodejs, Php, and Docker are supported.
version	Yes	String	Technology stack version.
deploy_mode	Yes	String	Deployment mode. - container - virtualmachine

Table 4-583 source

Parameter	Mandatory	Type	Description
kind	Yes	String	Component source kind. - code - package - image
version	No	String	Version number.

Parameter	Mandatory	Type	Description
url	No	String	Package address. This parameter is mandatory when kind is set to package or image .
storage	No	String	Storage mode. • devcloud: CodeArts release repository • obs • http: custom file address • swr This parameter is mandatory when kind is set to package or image .
http_username	No	String	Username. This parameter is mandatory when storage is set to http . Username and password authentication is supported.
http_password	No	String	Password. This parameter is mandatory when storage is set to http . Username and password authentication is supported.
http_command	No	String	Address of the custom software package. This parameter is mandatory when storage is set to http .
http_headers	No	Map<String, String>	This parameter is mandatory when storage is set to http . You can customize request headers as key-value pairs, for example, {"key": "xx", "value": "xx"}.
auth	No	String	Authentication mode. Value: iam or none. Default value: iam.
repo_auth	No	String	Authorization name, which can be obtained from the authorization list. For details, see Obtaining a Git Repository Authorization List .
repo_namespace	No	String	Namespace of the code repository.
repo_ref	No	String	Code branch or tag. Default value: master.
repo_type	No	String	Code repository type. Value: GitHub, BitBucket, or GitLab.
web_url	No	String	Code redirection URL. Example: https://github.com/example/demo.git.
repo_url	No	String	Code repository URL. Example: https://github.com/example/demo.git.

Parameter	Mandatory	Type	Description
commit_id	No	String	In the code repository, commit ID is the unique identifier of each commit and is used to identify and reference a commit. The value contains 8 to 40 characters and starts with a lowercase letter or digit.

Table 4-584 build

Parameter	Mandatory	Type	Description
parameters	No	Map<String, Object>	This parameter is provided only when no ID is available during build creation. See Table 4-585 .

Table 4-585 parameters

Parameter	Mandatory	Type	Description
build_cmd	No	String	Compilation command. By default: 1. When build.sh exists in the root directory, the command is ./build.sh . 2. When build.sh does not exist in the root directory, the command varies depending on the OS. Example: • Java and Tomcat: mvn clean package • Node.js: npm build
dockerfile_path	No	String	Address of the Docker file. By default, the Docker file is in the root directory (./).
artifact_namespace	No	String	Build archive organization. Default value: cas_{project_id}.
cluster_id	Yes	String	ID of the cluster to be built.
node_label_selector	No	Map<String, String>	key: label key. value: label value.
environment_id	No	String	Environment ID.

Table 4-586 external_accesses

Parameter	Mandatory	Type	Description
protocol	No	String	External access type. Value: http or https.
address	No	String	External access address.
forward_port	No	Integer	External access port.

Table 4-587 tolerations

Parameter	Mandatory	Type	Description
key	No	String	Taint key.
operator	No	String	Operator. - Equal: matches the nodes with the specified taint key (mandatory) and value. If the taint value is left blank, all taints with the same key will be matched. - Exists: matches nodes with the specified taint key. In this case, the taint value cannot be specified. Not specifying any taint key means all taints are tolerated.
value	No	String	Taint value. This parameter is mandatory when operator is set to Equal .
effect	No	String	Taint policy. - NoScheduler: Only the NoSchedule taint is matched. - PreferNoSchedule: Only the PreferNoSchedule taint is matched. - NoExecute: Only the NoExecute taint is matched. - If this parameter is not set, all taint effects are matched.
toleration_seconds	No	Integer	Tolerance time window. This parameter can be set only when effect is set to NoExecute . Within the tolerance time window, pods still run on the node with taints. After the time expires, the pods will be evicted.

Table 4-588 multi_containers

Parameter	Mandatory	Type	Description
container_name	Yes	String	Container name.
source	Yes	Object	Source of the code or software package. See Table 4-583 . In multi-container mode, only images are supported.
limit_cpu	No	Number	Maximum CPU limit. Unit: core.
limit_memory	No	Number	Maximum memory size. Unit: GiB.
request_cpu	No	Number	Requested CPU resources. Unit: core.
request_memory	No	Number	Requested memory. Unit: GiB.
envs	No	Array of objects	Environment variable list. See Table 4-580 .
liveness_probe	No	Object	Container liveness probe. See Table 4-577 .
readiness_probe	No	Object	Container service probe. See Table 4-577 .
startup_probe	No	Object	Container startup probe. See Table 4-577 .
command	No	Object	Startup command. See Table 4-564 .
post_start	No	Object	Post-start processing. See Table 4-565 .
pre_stop	No	Object	Pre-stop processing. See Table 4-565 .
logs	No	Array of objects	Log collection. See Table 4-573 .
storages	No	Array of objects	Storage. See Table 4-557 .
security_context	No	Object	Security policy. See Table 4-571 .
config_encryption	No	Object	This parameter is mandatory when the encryption key is configured for a multi-container component. See Table 4-593 .

Table 4-589 ports

Parameter	Mandatory	Type	Description
name	Yes	String	Port name.
port	Yes	String	Access port.
target_port	Yes	String	Container port.

Table 4-590 update_strategy

Parameter	Mandatory	Type	Description
type	No	String	Component upgrade policy. Options: • RollingUpdate (default): New pods are created gradually, and then old pods are deleted. • Recreate: Old pods are deleted, and then new pods are created.
max_unavailable	No	• String • Integer	Maximum number of unavailable pods. That is, the number or percentage of pods that can be deleted based on the value of replica . The default value is 25% . This parameter can be set when workload_kind is set to deployment . • String (%): the maximum percentage of unavailable pods to the total number of pods. • Integer: the maximum number of unavailable pods.
max_surge	No	• String • Integer	Maximum surge. That is, the number or percentage of pods allowed based on the value of replica . The default value is 25% . This parameter can be set only when workload_kind is set to deployment . • String (%) • Integer

Parameter	Mandatory	Type	Description
termination_period_seconds	No	Integer	Scale-in time window. That is, the time window for executing commands before a workload stops. The default value is 30s.
min_ready_seconds	No	Integer	Minimum ready time of a pod. That is, the minimum ready time of a new pod without any container crash. A pod is considered available only after the minimum ready time is exceeded. The default value is 0 .
progress_deadline_seconds	No	Integer	Maximum upgrade duration. That is, the time (in seconds) the system waits for before reporting a component upgrade failure. If this parameter is specified, the value of this parameter must be greater than that of min_ready_seconds .
revision_history_limit	No	Integer	Maximum number of pods to retain. That is, the number of old pods to retain to allow rollback. The default value is 10 .

Table 4-591 configs

Parameter	Mandatory	Type	Description
config_id	Yes	String	Unified configuration ID.
mount_path	Yes	String	Mount path.

Table 4-592 auto_lts_config

Parameter	Mandatory	Type	Description
enable	Yes	Boolean	Whether to enable automatic LTS configuration.
lts_log_path	No	Array of String	Access configuration log path.

Table 4-593 config_encryption

Parameter	Mandatory	Type	Description
key_id	Yes	String	User key ID in DEW.

Response

Parameter	Type	Description
name	String	Release task name.
description	String	Release task description.
model_version	Integer	Release task model. Only 2 is supported, indicating the new model.
config_mode	String	Release task deployment mode. • ui • template
id	String	Release task ID.
fail_mode	String	Best effort policy. • stop • continue
enterprise_project_id	String	Enterprise project ID.
pipeline_declaration	List of Object	Stage parameters. This parameter is mandatory when config_mode is ui . See Table 4-594 .

Table 4-594 pipeline_declaration

Parameter	Type	Description
sources	List of Object	Template package list. See Table 4-595 .
stages	List of Object	Stage parameters. See Table 4-598 .

Table 4-595 template_sources

Parameter	Type	Description
template_source	Object	Template package information. See Table 4-596 .

Parameter	Type	Description
input_values	Map<String, String>	Template package input information.

Table 4-596 template_source

Parameter	Type	Description
storage	String	Storage mode of the component template package. • devcloud: CodeArts release repository • obs • http: custom file address
url	String	Address of the component template package. This parameter is valid only when the component template package storage mode is http .
http_username	String	Username. This parameter is valid only when the component template package storage mode is http and the security authentication mode is username and password authentication.
http_password	String	Password. This parameter is valid only when the component template package storage mode is http and the security authentication mode is username and password authentication.
http_headers	Array of objects	Request header. See Table 4-597 . This parameter is valid only when the component template package storage mode is http and the security authentication mode is custom header authentication.

Table 4-597 http_headers

Parameter	Type	Description
key	String	Request key.
value	String	Request value.

Table 4-598 stages

Parameter	Type	Description
name	String	Stage name, which is unique in a release task.
display_name	String	Display name of a stage. If this parameter is left blank, the value is the same as the stage name.
task_group	Array of String	Task group. The parameters in the task group must correspond to the group_index field in Table 4-599 .
tasks	Array of object	Task parameters. See Table 4-599 .

Table 4-599 tasks

Parameter	Type	Description
name	String	Task name, which is unique in a single stage.
display_name	String	Display name of a task. If this parameter is left blank, the value is the same as the task name.
order_index	Integer	Task execution sequence. Tasks with the same group_index value belong to the same task group. The execution sequence is unique in a task group.
group_index	String	Task group. If the task groups are the same, the tasks are in the same group. The value is the same as that of task_group in Table 4-598 .
task_type_id	Object	Task execution parameters. See Table 4-600 .
input	Object	Task execution parameter, which is determined based on the task type. • For a build task in Table 4-550, see Table 4-551. • For a checkpoint task in Table 4-550, see Table 4-552. • For an application task in Table 4-550, see Table 4-554. • For a component task in Table 4-550, see Table 4-555.

Table 4-600 task_type_id

Parameter	Type	Description
category	String	Task type. • Build • CheckPoint • Environment • Component • Application
type	String	Task execution type. • create • update • upsert: create or modify. • approval

Table 4-601 build

Parameter	Type	Description
name	String	Build task name.
id	String	Build task ID.
enterprise_project_id	String	Enterprise project ID of a build task.

Table 4-602 checkpoint

Parameter	Type	Description
mode	String	If this field is left blank, no person needs to be specified for approval. • all: If all approvers approve the application, the application is approved. • any: If any approver approves the application, the application is approved.
users	List of Object	Approver. If this parameter is left blank, no approver is required. See Table 4-603 .

Table 4-603 users

Parameter	Type	Description
id	String	User ID.
label	String	User label, which is displayed on the frontend.
name	String	User name, which is displayed on the frontend.

Table 4-604 application

Parameter	Type	Description
name	String	Application name.
description	String	Application description.
enterprise_project_id	String	Enterprise project ID.

Table 4-605 component

Parameter	Type	Description
id	String	Application component ID.
name	String	Application component name. The value contains 2 to 64 characters consisting of letters, digits, hyphens (-), and underscores (_). It starts with a letter and ends with a letter or digit.
workload_name	String	Workload name.
labels	Array of objects	Label. For container-based deployment, you can query a workload by label. For VM-based deployment, you can configure a label of the corresponding microservice name to use graceful startup and shutdown. See Table 4-606 .
runtime_stack	Object	Runtime. See Table 4-632 . The value can be obtained from the response of the API for Querying a Technology Stack .
environment_id	String	Environment ID.

Parameter	Type	Description
description	String	Description. The value can contain up to 128 characters.
source	Object	Source of the code or software package. See Table 4-633 .
build	Object	Component building. This parameter is mandatory when the component is deployed in containers and no image is used. For details, see Table 4-634 .
limit_cpu	Number	Maximum CPU limit. Unit: core.
limit_memory	Number	Maximum memory size. Unit: GiB.
request_cpu	Number	Requested CPU resources. Unit: core.
request_memory	Number	Requested memory. Unit: GiB.
version	String	Component version number. The value can contain up to 32 characters. It needs to meet the following rule: $\wedge([0-9]+)(.[0-9]+)\{2,3\}\$$
envs	Array of objects	Environment variable list. See Table 4-630 .
replica	Integer	Number of instances.
storages	Array of objects	Storage. See Table 4-607 .
deploy_strategy	Object	Component deployment. See Table 4-610 .
command	Object	Startup command. See Table 4-614 .
post_start	Object	Post-start processing. See Table 4-615 .
pre_stop	Object	Pre-stop processing. See Table 4-615 .
mesher	Object	Multi-language access service mesh. This parameter is mandatory when Node.js is selected for the technology stack and the microservice engine is bound. See Table 4-616 .
timezone	String	Time zone where the component runs.
jvm_opts	String	JVM parameter. Example: -Xms256m -Xmx1024m. If there are multiple parameters, separate them by spaces. If the parameter is left blank, the default value is used.

Parameter	Type	Description
tomcat_opts	Object	Tomcat parameter. This parameter is valid when you use the Tomcat technology stack. See Table 4-617 .
host_aliases	Array of objects	Host alias. See Table 4-618 .
dns_policy	String	DNS policy. • Default: inherits the domain name resolution configuration of the node where the pod is located. • ClusterFirst: adds the domain name resolution configuration. • ClusterFirstWithHostNet: takes effect only for pods running in hostNetWork mode. • None: replaces the domain name resolution policy.
dns_config	Object	DNS configuration. See Table 4-619 .
workload_kind	String	Workload type. • deployment • statefulset • daemonset
security_context	Object	Security policy. See Table 4-621 .
logs	Array of objects	Log collection. See Table 4-623 .
custom_metric	Object	Custom metrics. See Table 4-624 .
affinity	Array of objects	Affinity. See Table 4-625 .
anti_affinity	Array of objects	Anti-affinity. See Table 4-625 .
liveness_probe	Object	Component liveness probe. See Table 4-627 .
readiness_probe	Object	Component service probe. See Table 4-627 .
startup_probe	Object	Component startup probe. See Table 4-627 .
refer_resources	Array of objects	Associated resource. See Table 4-628 .
external_accesses	Array of Objects	External access. See Table 4-636 .
tolerations	Array of Objects	Tolerance policy. See Table 4-637 .

Parameter	Type	Description
container_name	String	Container name.
update_strategy	Object	Component upgrade strategy. See Table 4-640 .
multi_containers	Array of Objects	Multi-container configuration. See Table 4-638 .
service_name	String	Service name configured for the inter-instance discovery of a StatefulSet.
ports	Array of Objects	Port configured for the inter-instance discovery of a StatefulSet.
workload_content	String	Workload content.
configs	Array of Objects	Unified component configuration. See Table 4-591 .
config_mode	String	Workload configuration mode.
enable_sermant_injection	Boolean	Whether to enable Sermant.
enable_sermant_spring_register	Boolean	Whether Spring Cloud Huawei microservices need to be registered with registry centers through Sermant.
enable_sermant_dubbo_register	Boolean	Whether Dubbo microservices need to be registered with registry centers through Sermant.
enable_sermant_config	Boolean	Whether to enable the Sermant dynamic configuration plug-in.
cell_id	String	Deployment cell ID.
swimlane_id	String	Lane ID.
sermantAgentInjectActionFlag	Boolean	Whether JDK 17 and JDK 21 automatically add the following JVM environment variables: JDK 17: --add-opens java.base/java.lang=ALL-UNNAMED --add-opens java.base/java.net=ALL-UNNAMED --add-opens java.base/java.math=ALL-UNNAMED --add-opens java.base/sun.net.www=ALL-UNNAMED --add-opens java.base/sun.net.www.protocol.http=ALL-UNNAMED JDK 21: --add-opens java.base/java.lang=ALL-UNNAMED --add-opens java.base/java.net=ALL-UNNAMED --add-opens java.base/java.math=ALL-UNNAMED --add-opens java.base/sun.net.www=ALL-UNNAMED --add-opens java.base/sun.net.www.protocol.http=ALL-UNNAMED -Dnet.bytebuddy.experimental=true

Parameter	Type	Description
sys_config	Map<String, String>	Uses a compressed package to deploy component configuration items in the VM environment.
image_pull_secrets	Array of String	List of image pull secrets.
auto_lts_config	Object	Automatic LTS configuration. See Table 4-592 .
config_encryption	Object	This parameter is mandatory when the encryption key is configured for a single-container component. See Table 4-593 .

Table 4-606 labels

Parameter	Type	Description
key	String	Label name.
value	String	Label value.

Table 4-607 storages

Parameter	Type	Description
type	String	Data storage type. • HostPath: host path for local disk mounting. • EmptyDir: temporary directory for local disk mounting. • ConfigMap: configuration item for local disk mounting. • Secret: secret for local disk mounting. • PersistentVolumeClaim: cloud storage mounting.
name	String	Name of the storage disk.
parameters	Object	For details about the parameters of each data storage type, see Table 4-608 .
mounts	Array of objects	Mount path of data storage. See Table 4-609 .

Table 4-608 component_storage_parameters

Parameter	Type	Description
path	String	Host path. This parameter is applicable to the HostPath storage type. This parameter is mandatory when type is set to HostPath .
name	String	Name of a configuration item, secret, or PVC, which is applicable to ConfigMap, Secret, or PersistentVolumeClaim storage, respectively. This parameter is mandatory when type is set to ConfigMap , Secret , or PersistentVolumeClaim .
default_mode	Integer	Mounting permission. The value is in decimal format. Example: 384. This parameter is applicable to ConfigMap and secret storage.
medium	String	This parameter is applicable to EmptyDir storage. If it is not specified, the default disk medium is used. If it is set to memory , memory storage is enabled.

Table 4-609 mounts

Parameter	Type	Description
path	String	Mount path.
sub_path	String	Subpath of the mount path.
read_only	Boolean	Read only or not.

Table 4-610 deploy_strategy

Parameter	Type	Description
type	String	Deployment type. - OneBatchRelease - RollingRelease - GrayRelease
rolling_release	Object	Rolling deployment parameter, which is mandatory when type is set to RollingRelease . For details, see Table 4-611 .
gray_release	Object	Dark launch upgrade parameter, which is mandatory when type is set to GrayRelease . For details, see Table 4-612 .

Table 4-611 rolling_release

Parameter	Type	Description
batches	Integer	Deployment batches.

Table 4-612 gray_release

Parameter	Type	Description
type	String	Dark launch policy. - WEIGHT: dark launch by traffic. - CONTENT: dark launch by content.
first_batch_weight	Integer	Proportion of first-batch dark launch traffic. This parameter is mandatory when type is set to WEIGHT .
first_batch_replica	Integer	Number of first-batch dark launch instances. This parameter is mandatory when type is set to WEIGHT .
remaining_batches	Integer	Number of batches for remaining instance deployment. After the first batch is completed, the remaining instances will be deployed in specified batches. Example: If there are five instances remaining to be deployed in three batches, these five instances will be upgraded based on 2:2:1. This parameter is mandatory when type is set to WEIGHT .
deployment_mode	Integer	Deployment mode. 1: The upstream is a microservice gateway. 3: The upstream is a microservice. 4: The upstream is a load balancer.
replica_surge_mode	String	Mode of adding a dark launch instance. This parameter is mandatory when type is set to CONTENT . - MIRROR: blue-green. - EXTRA: canary (increase, then decrease) - NOSURGE: canary (decrease, then increase)
rule_match_mode	String	Effective mode of a dark launch rule. This parameter is mandatory when type is set to CONTENT . - ALL: All conditions are met. - ANY: Any condition is met.

Parameter	Type	Description
rules	Object	Dark launch rule. See Table 4-613 . This parameter is mandatory when type is set to CONTENT .

Table 4-613 gray_rules

Parameter	Type	Description
type	String	Match type. Currently, only HEADER is supported.
key	String	Parameter name.
value	String	Condition value.
condition	String	Condition type. • EQUAL • MATCH • IN: enumeration

Table 4-614 command

Parameter	Type	Description
command	Array of String	Command for controlling container running.
args	Array of String	Parameter for controlling container running. Example: -port=8080. If there are multiple parameters, separate them by line breaks.

Table 4-615 component_lifecycle

Parameter	Type	Description
type	String	Processing method. • http • command
scheme	String	HTTP request type, which can be HTTP or HTTPS. This parameter is valid only when type is set to http .
host	String	Custom IP address. If this parameter is left blank, the pod IP address is used. This parameter is valid only when type is set to http .

Parameter	Type	Description
port	Integer	Port number. This parameter is valid only when type is set to http .
path	String	Request path. This parameter is valid only when type is set to http .
command	Array of String	Command list. This parameter is valid only when type is set to command .

Table 4-616 mesher

Parameter	Type	Description
port	Integer	Process listening port.

Table 4-617 tomcat_opt

Parameter	Type	Description
server_xml	String	1. Listening port in the server.xml file will take effect even if it has been configured in public network access. 2. The custom Tomcat configuration may conflict with the existing port. Check whether the custom port is occupied.

Table 4-618 host_aliases

Parameter	Type	Description
ip	String	IP address.
hostname	Array of String	Host alias.

Table 4-619 dns_config

Parameter	Type	Description
nameservers	Array of String	IP address list of the DNS server of the pod.
searches	Array of String	(Optional) List of DNS search domains used to search for host names in the pod.

Parameter	Type	Description
options	Array of objects	(Optional) List of objects. Each object may have a name (mandatory) and a value (optional). The contents of the objects here will be merged into the options generated from the specified DNS policy. Duplicate entries will be deleted. See Table 4-620 .

Table 4-620 options

Parameter	Type	Description
name	String	Operation name.
value	String	Operation value.

Table 4-621 security_context

Parameter	Type	Description
run_as_user	Integer	User for running containers. Example: To run containers as user root , set the user ID to 0 .
run_as_group	Integer	Owner group specified for running container processes.
capabilities	Object	Capability set. See Table 4-622 .

Table 4-622 capabilities

Parameter	Type	Description
add	Array of String	Grant the Linux permission to the running user.
drop	Array of String	Remove the Linux permission of the running user.

Table 4-623 logs

Parameter	Type	Description
log_path	String	Container log path.
rotate	String	Interval for dumping logs.
host_path	String	Mounted host path.

Parameter	Type	Description
host_extend_path	String	Host extension path. By extending the host path, you can distinguish mounting from different containers in the same host path. • None: The extended path is not used. • PodUID: extend the host path based on the pod ID. • PodName: extend the host path based on the pod name. • PodUID/ContainerName: extend the host path based on the pod ID and container name. • PodName/ContainerName: extend the host path based on the pod name and container name.

Table 4-624 custom_metric

Parameter	Type	Description
path	String	Collection path. Example: /metrics.
port	Integer	Collection port. Example: 9090.
dimensions	String	Monitoring dimension. Example: cpu_usage,mem_usage.

Table 4-625 component_affinity

Parameter	Type	Description
kind	String	Object kind. • node • pod
condition	String	Policy type. • required: The requirement must be met. • preferred: The requirement is met as much as possible.
match_expressions	Array of objects	Scheduling policy rule. See Table 4-626 .
weight	String	Weight. • The value is 100 when condition is set to required. • Enter an integer ranging from 1 to 100 when condition is set to preferred.

Table 4-626 match_expressions

Parameter	Type	Description
key	String	Condition key.
value	String	Condition value.
operation	String	Operation.

Table 4-627 component_probe

Parameter	Type	Description
type	String	Type. Value: http, tcp, or command.
delay	Integer	Interval between the startup and detection.
timeout	Integer	Detection timeout interval.
scheme	String	Request type. Value: HTTP or HTTPS. This parameter is valid only when type is set to http .
host	String	Custom IP address. If this parameter is left blank, the pod IP address is used. This parameter is valid only when type is set to http .
port	Integer	Port number. This parameter is valid only when type is set to http or tcp .
path	String	Request path. This parameter is valid only when type is set to http .
command	Array of String	Command list. This parameter is valid only when type is set to command .

Table 4-628 refer_resources

Parameter	Type	Description
id	String	Resource ID.

Parameter	Type	Description
type	String	Resource type. • vpc • eip • elb • cce • ecs • dcs • rds • cse: ServiceComb engine. • nacos: registry/configuration center. • customServer • pvc • gaussdb • dms_rocketmq • css_es: Elasticsearch
parameters	Object	Resource parameters. See Table 4-629 .

Table 4-629 refer_resource_parameter

Parameter	Type	Description
namespace	String	Namespace.
name	String	Resource name.
capacity	Integer	Storage capacity, in GiB.
type	String	Volume declaration type. • efs • obs • sfs • evs
class	String	Volume declaration subtype. • csi-sfsturbo: fixed when type is set to efs. • csi-obs: fixed when type is set to obs. • csi-nas: fixed when type is set to sfs. • csi-disk: fixed when type is set to evs.
obs_volume_type	String	OBS volume type. Default value: STANDARD.

Parameter	Type	Description
access_mode	String	Storage volume access mode. - ReadWriteOnce: A volume can be mounted as read-write by a single node. - ReadWriteMany: A volume can be mounted as read-write by many nodes.
availableZone	String	AZ.
volume_type	String	Volume type.

Table 4-630 envs

Parameter	Type	Description
name	String	Variable name.
value	String	Variable value.
value_from	Object	Variable reference. See Table 4-631 .

Table 4-631 value_from

Parameter	Type	Description
reference_type	String	Reference type. - configMapKey - secretKey
name	String	Name of the configuration item or secret.
key	String	Key of the configuration item or secret.
optional	Boolean	Whether the key of the configuration item or secret is mandatory.

Table 4-632 runtime_stack

Parameter	Type	Description
name	String	Technology stack name.

Parameter	Type	Description
type	String	Technology stack type. Value: Java, Tomcat, Nodejs, Php, Docker, or Python. - For VM-based deployment, Java, Tomcat, and Nodejs are supported. - For container-based deployment, Java, Tomcat, Nodejs, Php, and Docker are supported.
version	String	Technology stack version.
deploy_mode	String	Deployment mode. - container - virtualmachine

Table 4-633 source

Parameter	Type	Description
kind	String	Kind of the component source package. - code - package - image
version	String	Version number.
url	String	Package address. This parameter is mandatory when kind is set to package or image .
storage	String	Storage mode. - devcloud: CodeArts release repository - obs - http: custom file address - swr This parameter is mandatory when kind is set to package or image .
auth	String	Authentication mode. Value: iam or none. Default value: iam.
repo_auth	String	Authorization name, which can be obtained from the authorization list. For details, see Obtaining a Git Repository Authorization List .
repo_namespace	String	Namespace of the code repository.
repo_ref	String	Code branch or tag. Default value: master.

Parameter	Type	Description
repo_type	String	Code repository type. Value: GitHub, BitBucket, or GitLab.
web_url	String	Code redirection URL. Example: https://github.com/example/demo.git.
repo_url	String	Code repository URL. Example: https://github.com/example/demo.git.
http_username	String	Username. This parameter is mandatory when storage is set to http . Username and password authentication is supported.
http_password	String	Password. This parameter is mandatory when storage is set to http . Username and password authentication is supported.
http_command	String	Address of the custom software package. This parameter is mandatory when storage is set to http .
http_headers	Map<String, String>	This parameter is mandatory when storage is set to http . You can customize request headers as key-value pairs, for example, {"key": "xx", "value": "xx"}.

Table 4-634 build

Parameter	Type	Description
parameters	Map<String, Object>	This parameter is provided only when no ID is available during build creation. See Table 4-635 .

Table 4-635 parameters

Parameter	Type	Description
build_cmd	String	Compilation command. By default: - When build.sh exists in the root directory, the command is ./build.sh. - When build.sh does not exist in the root directory, the command varies depending on the OS. Example: - Java and Tomcat: mvn clean package - Node.js: npm build
dockerfile_path	String	Address of the Docker file. By default, the Docker file is in the root directory (.).

Parameter	Type	Description
artifact_name space	String	Build archive organization. Default value: cas_{project_id}.
cluster_id	String	ID of the cluster to be built.
node_label_selector	Map<String, String>	key: label key. value: label value.
environment_id	String	Environment ID.

Table 4-636 external_accesses

Parameter	Type	Description
protocol	String	External access type. Value: http or https.
address	String	External access address.
forward_port	Integer	External access port.

Table 4-637 tolerations

Parameter	Type	Description
key	String	Taint key.
operator	String	Operator. - Equal: matches the nodes with the specified taint key (mandatory) and value. If the taint value is left blank, all taints with the same key will be matched. - Exists: matches nodes with the specified taint key. In this case, the taint value cannot be specified. Not specifying any taint key means all taints are tolerated.
value	String	Taint value. This parameter is mandatory when operator is set to Equal .

Parameter	Type	Description
effect	String	Taint policy. • NoScheduler: Only the NoSchedule taint is matched. • PreferNoSchedule: Only the PreferNoSchedule taint is matched. • NoExecute: Only the NoExecute taint is matched. • If this parameter is not set, all taint effects are matched.
toleration_seconds	Integer	Tolerance time window. This parameter can be set only when effect is set to NoExecute . Within the tolerance time window, pods still run on the node with taints. After the time expires, the pods will be evicted.

Table 4-638 multi_containers

Parameter	Type	Description
container_name	String	Container name.
source	Object	Source of the code or software package. See Table 4-633 . In multi-container mode, only images are supported.
limit_cpu	Number	Maximum CPU limit. Unit: core.
limit_memory	Number	Maximum memory size. Unit: GiB.
request_cpu	Number	Requested CPU resources. Unit: core.
request_memory	Number	Requested memory. Unit: GiB.
envs	Array of objects	Environment variable list. See Table 4-630 .
liveness_probe	Object	Container liveness probe. See Table 4-627 .
readiness_probe	Object	Container service probe. See Table 4-627 .
startup_probe	Object	Container startup probe. See Table 4-627 .
command	Object	Startup command. See Table 4-614 .
post_start	Object	Post-start processing. See Table 4-615 .
pre_stop	Object	Pre-stop processing. See Table 4-615 .

Parameter	Type	Description
logs	Array of objects	Log collection. See Table 4-623 .
storages	Array of objects	Storage. See Table 4-607 .
security_context	Object	Security policy. See Table 4-621 .
config_encryption	Object	This parameter is mandatory when the encryption key is configured for a multi-container component. See Table 4-593 .

Table 4-639 ports

Parameter	Type	Description
name	String	Port name.
port	String	Access port.
target_port	String	Container port.

Table 4-640 update_strategy

Parameter	Type	Description
type	String	Component upgrade policy. Options: - RollingUpdate (default): New pods are created gradually, and then old pods are deleted. - Recreate: Old pods are deleted, and then new pods are created.
max_unavailable	- String - Integer	Maximum number of unavailable pods. That is, the number or percentage of pods that can be deleted based on the value of replica . The default value is 25% . This parameter can be set when workload_kind is set to deployment . - String (%): the maximum percentage of unavailable pods to the total number of pods. - Integer: the maximum number of unavailable pods.

Parameter	Type	Description
max_surge	- String - Integer	Maximum surge. That is, the number or percentage of pods allowed based on the value of replica . The default value is 25% . This parameter can be set only when workload_kind is set to deployment . - String (%) - Integer
termination_period_seconds	Integer	Scale-in time window. That is, the time window for executing commands before a workload stops. The default value is 30s.
min_ready_seconds	Integer	Minimum ready time of a pod. That is, the minimum ready time of a new pod without any container crash. A pod is considered available only after the minimum ready time is exceeded. The default value is 0 .
progress_deadline_seconds	Integer	Maximum upgrade duration. That is, the time (in seconds) the system waits for before reporting a component upgrade failure. If this parameter is specified, the value of this parameter must be greater than that of min_ready_seconds .
revision_history_limit	Integer	Maximum number of pods to retain. That is, the number of old pods to retain to allow rollback. The default value is 10 .

Table 4-641 configs

Parameter	Type	Description
config_id	String	Unified configuration ID.
mount_path	String	Mount path.

Table 4-642 auto_lts_config

Parameter	Type	Description
enable	Boolean	Whether to enable automatic LTS configuration.
lts_log_path	Array of String	Access configuration log path.

Table 4-643 config_encryption

Parameter	Type	Description
key_id	String	User key ID in DEW.

Example Request

Use the template package to create a release task named **release-template**.

```
{
  "model_version": 2,
  "name": "release-template",
  "config_mode": "template",
  "description": "",
  "enterprise_project_id": "0",
  "sources": [
    {
      "template_source": {
        "storage": "obs",
        "url": "obs://obs-zyh/create-component.zip",
        "http_username": "",
        "http_password": "",
        "http_headers": []
      },
      "input_values": {
        "env_name": "demo",
        "image1": "swr.ap-southeast-1.myhuaweicloud.com/cse/java-cce:v3",
        "application_id": "demo",
        "component_name1": "demo",
        "cce_id": "demo"
      }
    }
  ]
}
```

Example Response

```
{
  "id": "4105b37a-947e-4aca-90d8-cec7c04c6f38",
  "enterprise_project_id": "0",
  "name": "release-gn7mre11",
  "enterprise_project": "default",
  "model_version": 2,
  "creator": "zhangyuhao-3",
  "status": null,
  "revisor": "zhangyuhao-3",
  "create_at": 1739865312582,
  "update_at": 1739865967776,
  "fail_mode": "stop",
  "strategy_type": 0,
  "description": "test",
  "pipeline_declaration": {
    "stages": [
      {
        "id": 15,
        "name": "75foki",
        "approval_type": 0,
        "display_name": "Stage"
        "type": null,
        "pipeline_id": null,
        "order_index": 0,
        "status": null,
        "task_group": [
          "0"
        ]
      }
    ]
  }
}
```

```
    "tasks": [
      {
        "id": 17,
        "name": "bkdi55",
        "display_name": "Create application-187",
        "task_type_id": {
          "category": "Application",
          "owner": null,
          "provider": null,
          "type": "upsert"
        },
        "input": {
          "displayName": "Create application-187",
          "originalName": "bkdi55",
          "name": "xxz",
          "description": "",
          "enterprise_project_id": "0",
          "checked": true
        },
        "output": {},
        "status": null,
        "order_index": 0,
        "skipped": false,
        "group_index": "0",
        "delay_time": 0,
        "error_details": null,
        "stage_name": "75foki"
      }
    ]
  },
  "config_mode": "ui"
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.7.2.2 Obtaining the Release Task List

Function

This API is used to obtain the release task list.

URI

GET /v3/{project_id}/pipeline/pipelines

Table 4-644 Path parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .

Request

Table 4-645 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

Table 4-646 Response

Parameter	Type	Description
pipelines	Array of Object	Release task information. See Table 4-647 .

Table 4-647 pipeline

Parameter	Type	Description
id	String	Release task ID.
name	String	Release task name.
description	String	Release task description.
create_time	Integer	Creation time of a release task.
update_time	Integer	Update time of a release task.

Parameter	Type	Description
model_version	Integer	Release task model. Only 2 is supported, indicating the new model.
config_mode	String	Release task deployment mode. • ui • template
status	Integer	Release task status. • 1: initialization successful • 2: release successful • 3: release failed • 4: releasing • 5: to be approved • 7: pause • 9: skip • 10: rolling back • 11: rollback successful • 12: rollback failed • 13: stopping • 14: restarting
creator	String	Creator.
last_trigger_at	Integer	Last execution time.
enterprise_project_id	String	Enterprise project ID.

Example Request

None

Example Response

```
[
  {
    "id": "4105b37a-947e-4aca-90d8-cec7c04c6f38",
    "name": "release-gn7mre",
    "description": "",
    "create_time": 1739865312582,
    "update_time": 1739865312582,
    "version": 1,
    "config_mode": "ui",
    "creator": "testcreator-3",
    "model_version": 2,
    "status": 1,
    "last_trigger_at": 0,
    "enterprise_project_id": "0",
    "enterprise_project": null,
  }
]
```

```
    "extend_info": null
  }
]
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.7.2.3 Editing a Release Task

Function

This API is used to edit a release task.

URI

PUT /v3/{project_id}/pipeline/pipelines/{pipeline_id}

Table 4-648 Path parameter

Parameter	Mand atory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
pipeline_id	Yes	String	Release task ID. See Obtaining the Release Task List .

Request

Table 4-649 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Table 4-650 Request body parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Pipeline name.
description	No	String	Release task description.
model_version	Yes	Integer	Release task model. Only 2 is supported, indicating the new model.
config_mode	Yes	String	Release task deployment mode. • ui • template
sources	No	List of Object	Template package list. This parameter is mandatory when config_mode is set to template . See Table 4-651 .
enterprise_project_id	No	String	Enterprise project ID. If this parameter is not set, the default enterprise project is default and the ID is 0 .
stages	No	List of Object	Stage parameters. This parameter is mandatory when config_mode is ui . See Table 4-654 .

Table 4-651 template_sources

Parameter	Mandatory	Type	Description
template_source	Yes	Object	Template package information. See Table 4-652 .
input_values	No	Map<String, String>	Template package input information.

Table 4-652 template_source

Parameter	Mandatory	Type	Description
storage	Yes	String	Storage mode of the component template package. - devcloud: CodeArts release repository - obs - http: custom file address
url	Yes	String	Address of the component template package. This parameter is valid only when the component template package storage mode is http .
http_username	No	String	Username. This parameter is valid only when the component template package storage mode is http and the security authentication mode is username and password authentication.
http_password	No	String	Password. This parameter is valid only when the component template package storage mode is http and the security authentication mode is username and password authentication.
http_headers	No	Array of objects	Request header. See Table 4-653 . This parameter is valid only when the component template package storage mode is http and the security authentication mode is custom header authentication.

Table 4-653 http_headers

Parameter	Mandatory	Type	Description
key	Yes	String	Request key.
value	Yes	String	Request value.

Table 4-654 stages

Parameter	Mandatory	Type	Description
name	Yes	String	Stage name, which is unique in a release task.
display_name	No	String	Display name of a stage. If this parameter is left blank, the value is the same as the stage name.
task_group	Yes	Array of String	Task group. The parameters in the task group must correspond to the group_index field in Table 4-655 .
tasks	Yes	Array of object	Task parameters. See Table 4-655 .

Table 4-655 tasks

Parameter	Mandatory	Type	Description
name	Yes	String	Task name, which is unique in a single stage.
display_name	No	String	Display name of a task. If this parameter is left blank, the value is the same as the task name.
order_index	Yes	Integer	Task execution sequence. Tasks with the same group_index value belong to the same task group. The execution sequence is unique in a task group.
group_index	Yes	String	Task group. If the task groups are the same, the tasks are in the same group. The value is the same as that of task_group in Table 4-654 .
task_type_id	Yes	Object	Task execution parameters. See Table 4-656 .

Parameter	Mandatory	Type	Description
input	No	Object	Task execution parameter, which is determined based on the task type. - For a build task in Table 4-656, see Table 4-657. - For a checkpoint task in Table 4-656, see Table 4-658. - For an application task in Table 4-656, see Table 4-660. - For a component task in Table 4-656, see Table 4-661.

Table 4-656 task_type_id

Parameter	Mandatory	Type	Description
category	Yes	String	Task type. - Build - CheckPoint - Environment - Component - Application
type	Yes	String	Task execution type. - create - update - upsert: create or modify. - approval

Table 4-657 build

Parameter	Mandatory	Type	Description
name	Yes	String	Build task name.
id	Yes	String	Build task ID.
enterprise_project_id	No	String	Enterprise project ID of a build task. If this parameter is not set, the default enterprise project is default and the ID is 0 .

Table 4-658 checkpoint

Parameter	Mandatory	Type	Description
mode	No	String	If this field is left blank, no person needs to be specified for approval. - all: If all approvers approve the application, the application is approved. - any: If any approver approves the application, the application is approved.
users	No	List of Object	Approver. If this parameter is left blank, no approver is required. See Table 4-659 .

Table 4-659 users

Parameter	Mandatory	Type	Description
id	Yes	String	User ID.
label	No	String	User label, which is displayed on the frontend.
name	No	String	User name, which is displayed on the frontend.

Table 4-660 application

Parameter	Mandatory	Type	Description
name	Yes	String	Application name.
description	Yes	String	Application description.
enterprise_project_id	No	String	Enterprise project ID. If this parameter is not set, the default enterprise project is default and the ID is 0 .

Table 4-661 component

Parameter	Mandatory	Type	Description
id	No	String	Application component ID.
name	Yes	String	Application component name. The value contains 2 to 64 characters consisting of letters, digits, hyphens (-), and underscores (_). It starts with a letter and ends with a letter or digit.
workload_name	No	String	Workload name.
labels	No	Array of objects	Label. For container-based deployment, you can query a workload by label. For VM-based deployment, you can configure a label of the corresponding microservice name to use graceful startup and shutdown. See Table 4-662 .
runtime_stack	Yes	Object	Runtime. See Table 4-688 . The value can be obtained from the response of the API for Querying a Technology Stack .
environment_id	Yes	String	Environment ID.
description	No	String	Description. The value can contain up to 128 characters.
source	Yes	Object	Source of the code or software package. See Table 4-689 .
build	No	Object	Component building. This parameter is mandatory when the component is deployed in containers and no image is used. For details, see Table 4-690 .
limit_cpu	No	Number	Maximum CPU limit. Unit: core.
limit_memory	No	Number	Maximum memory size. Unit: GiB.
request_cpu	No	Number	Requested CPU resources. Unit: core.
request_memory	No	Number	Requested memory. Unit: GiB.

Parameter	Mandatory	Type	Description
version	Yes	String	Component version number. The value can contain up to 32 characters. It needs to meet the following rule: <code>^([0-9]+)(.[0-9]+){2,3}\$</code>
envs	No	Array of objects	Environment variable list. See Table 4-686 .
replica	Yes	Integer	Number of instances.
storages	No	Array of objects	Storage. See Table 4-663 .
deploy_strategy	No	Object	Component deployment. See Table 4-666 .
command	No	Object	Startup command. See Table 4-670 .
post_start	No	Object	Post-start processing. See Table 4-671 .
pre_stop	No	Object	Pre-stop processing. See Table 4-671 .
mesher	No	Object	Multi-language access service mesh. This parameter is mandatory when Node.js is selected for the technology stack and the microservice engine is bound. See Table 4-672 .
timezone	No	String	Time zone where the component runs.
jvm_opts	No	String	JVM parameter. Example: -Xms256m -Xmx1024m. If there are multiple parameters, separate them by spaces. If the parameter is left blank, the default value is used.
tomcat_opts	No	Object	Tomcat parameter. This parameter is valid when you use the Tomcat technology stack. See Table 4-673 .
host_aliases	No	Array of objects	Host alias. See Table 4-674 .

Parameter	Mandatory	Type	Description
dns_policy	No	String	DNS policy. • Default: inherits the domain name resolution configuration of the node where the pod is located. • ClusterFirst: adds the domain name resolution configuration. • ClusterFirstWithHostNet: takes effect only for pods running in hostNetwork mode. • None: replaces the domain name resolution policy.
dns_config	No	Object	DNS configuration. See Table 4-675 .
workload_kind	No	String	Workload type. • deployment • statefulset • daemonset
security_context	No	Object	Security policy. See Table 4-677 .
logs	No	Array of objects	Log collection. See Table 4-679 .
custom_metrics	No	Object	Custom metrics. See Table 4-680 .
affinity	No	Array of objects	Affinity. See Table 4-681 .
anti_affinity	No	Array of objects	Anti-affinity. See Table 4-681 .
liveness_probe	No	Object	Component liveness probe. See Table 4-683 .
readiness_probe	No	Object	Component service probe. See Table 4-683 .
startup_probe	No	Object	Component startup probe. See Table 4-683 .

Parameter	Mandatory	Type	Description
refer_resources	Yes	Array of objects	Associated resource. See Table 4-684 .
external_accesses	No	Array of Objects	External access. See Table 4-692 .
tolerations	No	Array of Objects	Tolerance policy. See Table 4-693 .
container_name	No	String	Container name.
update_strategy	No	Object	Component upgrade strategy. See Table 4-696 .
multi_containers	No	Array of Objects	Multi-container configuration. See Table 4-694 .
service_name	No	String	Service name configured for the inter-instance discovery of a StatefulSet.
ports	No	Array of Objects	Port configured for the inter-instance discovery of a StatefulSet.
workload_content	No	String	Workload content.
configs	No	Array of Objects	Unified component configuration. See Table 4-697 .
config_mode	No	String	Workload configuration mode.
enable_serment_injection	No	Boolean	Whether to enable Serment.
enable_serment_spring_register	No	Boolean	Whether Spring Cloud Huawei microservices need to be registered with registry centers through Serment.

Parameter	Mandatory	Type	Description
enable_serment_dubbo_register	No	Boolean	Whether Dubbo microservices need to be registered with registry centers through Serment.
enable_serment_config	No	Boolean	Whether to enable the Serment dynamic configuration plug-in.
cell_id	No	String	Deployment cell ID.
swimlane_id	No	String	Lane ID.
sermantAgentInjectActionFlag	No	Boolean	Whether JDK 17 and JDK 21 automatically add the following JVM environment variables: JDK 17: --add-opens java.base/java.lang=ALL-UNNAMED --add-opens java.base/java.net=ALL-UNNAMED --add-opens java.base/java.math=ALL-UNNAMED --add-opens java.base/sun.net.www=ALL-UNNAMED --add-opens java.base/sun.net.www.protocol.http=ALL-UNNAMED JDK 21: --add-opens java.base/java.lang=ALL-UNNAMED --add-opens java.base/java.net=ALL-UNNAMED --add-opens java.base/java.math=ALL-UNNAMED --add-opens java.base/sun.net.www=ALL-UNNAMED --add-opens java.base/sun.net.www.protocol.http=ALL-UNNAMED -Dnet.bytebuddy.experimental=true
sys_config	No	Map<String, String>	Uses a compressed package to deploy component configuration items in the VM environment.
image_pull_secrets	No	Array of String	List of image pull secrets.
auto_lts_config	No	Object	Automatic LTS configuration. See Table 4-698 .
config_encryption	No	Object	This parameter is mandatory when the encryption key is configured for a single-container component. See Table 4-699 .

Table 4-662 labels

Parameter	Mandatory	Type	Description
key	Yes	String	Label name.
value	Yes	String	Label value.

Table 4-663 storages

Parameter	Mandatory	Type	Description
type	Yes	String	Data storage type. • HostPath: host path for local disk mounting. • EmptyDir: temporary directory for local disk mounting. • ConfigMap: configuration item for local disk mounting. • Secret: secret for local disk mounting. • PersistentVolumeClaim: cloud storage mounting.
name	Yes	String	Name of the storage disk.
parameters	Yes	Object	For details about the parameters of each data storage type, see Table 4-664 .
mounts	Yes	array of objects	Mount path of data storage. See Table 4-665 .

Table 4-664 component_storage_parameters

Parameter	Mandatory	Type	Description
path	No	String	Host path. This parameter is applicable to the HostPath storage type. This parameter is mandatory when type is set to HostPath .
name	No	String	Name of a configuration item, secret, or PVC, which is applicable to ConfigMap, Secret, or PersistentVolumeClaim storage, respectively. This parameter is mandatory when type is set to ConfigMap , Secret , or PersistentVolumeClaim .
default_mode	No	Integer	Mounting permission. The value is in decimal format. Example: 384. This parameter is applicable to ConfigMap and secret storage.

Parameter	Mandatory	Type	Description
medium	No	String	This parameter is applicable to EmptyDir storage. If it is not specified, the default disk medium is used. If it is set to memory , memory storage is enabled.

Table 4-665 mounts

Parameter	Mandatory	Type	Description
path	Yes	String	Mount path.
sub_path	Yes	String	Subpath of the mount path.
read_only	Yes	Boolean	Read-only or not.

Table 4-666 deploy_strategy

Parameter	Mandatory	Type	Description
type	Yes	String	Deployment type. - OneBatchRelease - RollingRelease - GrayRelease
rolling_release	No	Object	Rolling deployment parameter, which is mandatory when type is set to RollingRelease . For details, see Table 4-667 .
gray_release	No	Object	Dark launch upgrade parameter, which is mandatory when type is set to GrayRelease . For details, see Table 4-668 .

Table 4-667 rolling_release

Parameter	Mandatory	Type	Description
batches	Yes	Integer	Deployment batches.

Table 4-668 gray_release

Parameter	Mandatory	Type	Description
type	Yes	String	Dark launch policy. - WEIGHT: dark launch by traffic. - CONTENT: dark launch by content.
first_batch_weight	Yes	Integer	Proportion of first-batch dark launch traffic. This parameter is mandatory when type is set to WEIGHT .
first_batch_replica	Yes	Integer	Number of first-batch dark launch instances. This parameter is mandatory when type is set to WEIGHT .
remaining_batches	Yes	Integer	Number of batches for remaining instance deployment. After the first batch is completed, the remaining instances will be deployed in specified batches. Example: If there are five instances remaining to be deployed in three batches, these five instances will be upgraded based on 2:2:1. This parameter is mandatory when type is set to WEIGHT .
deployment_mode	No	Integer	Deployment mode. 1: The upstream is a microservice gateway. 3: The upstream is a microservice. 4: The upstream is a load balancer.
replica_surge_mode	No	String	Mode of adding a dark launch instance. This parameter is mandatory when type is set to CONTENT . - MIRROR: blue-green. - EXTRA: canary (increase, then decrease) - NOSURGE: canary (decrease, then increase)
rule_match_mode	No	String	Effective mode of a dark launch rule. This parameter is mandatory when type is set to CONTENT . - ALL: All conditions are met. - ANY: Any condition is met.
rules	No	Object	Dark launch rule. See Table 4-669 . This parameter is mandatory when type is set to CONTENT .

Table 4-669 gray_rules

Parameter	Mandatory	Type	Description
type	Yes	String	Match type. Currently, only HEADER is supported.
key	Yes	String	Parameter name.
value	Yes	String	Condition value.
condition	Yes	String	Condition type. • EQUAL • MATCH • IN: enumeration

Table 4-670 command

Parameter	Mandatory	Type	Description
command	No	Array of String	Command for controlling container running.
args	No	Array of String	Parameter for controlling container running. Example: -port=8080. If there are multiple parameters, separate them by line breaks.

Table 4-671 component_lifecycle

Parameter	Mandatory	Type	Description
type	Yes	String	Processing method. • http • command
scheme	No	String	HTTP request type, which can be HTTP or HTTPS. This parameter is valid only when type is set to http .

Parameter	Mandatory	Type	Description
host	No	String	Custom IP address. If this parameter is left blank, the pod IP address is used. This parameter is valid only when type is set to http .
port	No	Integer	Port number. This parameter is valid only when type is set to http .
path	No	String	Request path. This parameter is valid only when type is set to http .
command	No	Array of String	Command list. This parameter is valid only when type is set to command .

Table 4-672 mesher

Parameter	Mandatory	Type	Description
port	Yes	Integer	Process listening port.

Table 4-673 tomcat_opt

Parameter	Mandatory	Type	Description
server_xml	Yes	String	- Listening port in the server.xml file will take effect even if it has been configured in public network access. - The custom Tomcat configuration may conflict with the existing port. Check whether the custom port is occupied.

Table 4-674 host_aliases

Parameter	Mandatory	Type	Description
ip	Yes	String	IP address.
hostname	Yes	Array of String	Host alias.

Table 4-675 dns_config

Parameter	Mandatory	Type	Description
nameservers	No	Array of String	IP address list of the DNS server of the pod.
searches	No	Array of String	(Optional) List of DNS search domains used to search for host names in the pod.
options	No	Array of objects	(Optional) List of objects. Each object may have a name (mandatory) and a value (optional). The contents of the objects here will be merged into the options generated from the specified DNS policy. Duplicate entries will be deleted. See Table 4-676 .

Table 4-676 options

Parameter	Mandatory	Type	Description
name	Yes	String	Operation name.
value	No	String	Operation value.

Table 4-677 security_context

Parameter	Mandatory	Type	Description
run_as_user	No	Integer	User for running containers. Example: To run containers as user root , set the user ID to 0 .
run_as_group	No	Integer	Owner group specified for running container processes.
capabilities	No	Object	Capability set. See Table 4-678 .

Table 4-678 capabilities

Parameter	Mandatory	Type	Description
add	No	Array of String	Grant the Linux permission to the running user.

Parameter	Mandatory	Type	Description
drop	No	Array of String	Remove the Linux permission of the running user.

Table 4-679 logs

Parameter	Mandatory	Type	Description
log_path	Yes	String	Container log path.
rotate	Yes	String	Interval for dumping logs.
host_path	Yes	String	Mounted host path.
host_extend_path	Yes	String	Host extension path. By extending the host path, you can distinguish mounting from different containers in the same host path. • None: The extended path is not used. • PodUID: extend the host path based on the pod ID. • PodName: extend the host path based on the pod name. • PodUID/ContainerName: extend the host path based on the pod ID and container name. • PodName/ContainerName: extend the host path based on the pod name and container name.

Table 4-680 custom_metric

Parameter	Mandatory	Type	Description
path	Yes	String	Collection path. Example: /metrics.
port	Yes	Integer	Collection port. Example: 9090.
dimensions	Yes	String	Monitoring dimension. Example: cpu_usage,mem_usage.

Table 4-681 component_affinity

Parameter	Mandatory	Type	Description
kind	No	String	Object kind. • node • pod
condition	No	String	Policy type. • required: The requirement must be met. • preferred: The requirement is met as much as possible.
match_expressions	No	Array of objects	Scheduling policy rule. See Table 4-682 .
weight	No	String	Weight. • The value is 100 when condition is set to required. • Enter an integer ranging from 1 to 100 when condition is set to preferred.

Table 4-682 match_expressions

Parameter	Mandatory	Type	Description
key	Yes	String	Condition key.
value	Yes	String	Condition value.
operation	Yes	String	Operation.

Table 4-683 component_probe

Parameter	Mandatory	Type	Description
type	Yes	String	Type. Value: http, tcp, or command.
delay	Yes	Integer	Interval between the startup and detection.
timeout	Yes	Integer	Detection timeout interval.
scheme	No	String	Request type. Value: HTTP or HTTPS. This parameter is valid only when type is set to http .

Parameter	Mandatory	Type	Description
host	No	String	Custom IP address. If this parameter is left blank, the pod IP address is used. This parameter is valid only when type is set to http .
port	No	Integer	Port number. This parameter is valid only when type is set to http or tcp .
path	No	String	Request path. This parameter is valid only when type is set to http .
command	No	Array of String	Command list. This parameter is valid only when type is set to command .

Table 4-684 refer_resources

Parameter	Mandatory	Type	Description
id	Yes	String	Resource ID.
type	Yes	String	Resource type. • vpc • eip • elb • cce • ecs • dcs • rds • cse: ServiceComb engine. • nacos: registry/configuration center. • customServer • pvc • gaussdb • dms_rocketmq • css_es: Elasticsearch
parameters	No	Object	Resource parameters. See Table 4-685 .

Table 4-685 refer_resource_parameter

Parameter	Mandatory	Type	Description
namespace	No	String	Namespace.
name	No	String	Resource name.
capacity	No	Integer	Storage capacity, in GiB.
type	No	String	Volume declaration type. • efs • obs • sfs • evs
class	No	String	Volume declaration subtype. • csi-sfsturbo: fixed when type is set to efs. • csi-obs: fixed when type is set to obs. • csi-nas: fixed when type is set to sfs. • csi-disk: fixed when type is set to evs.
obs_volume_type	No	String	OBS volume type. Default value: STANDARD.
access_mode	No	String	Storage volume access mode. • ReadWriteOnce: A volume can be mounted as read-write by a single node. • ReadWriteMany: A volume can be mounted as read-write by many nodes.
availableZone	No	String	AZ.
volume_type	No	String	Volume type.

Table 4-686 envs

Parameter	Mandatory	Type	Description
name	Yes	String	Variable name.
value	No	String	Variable value.
value_from	No	Object	Variable reference. See Table 4-687 .

Table 4-687 value_from

Parameter	Mandatory	Type	Description
reference_type	Yes	String	Reference type. - configMapKey - secretKey
name	Yes	String	Name of the configuration item or secret.
key	No	String	Key of the configuration item or secret.
optional	No	Boolean	Whether the key of the configuration item or secret is mandatory.

Table 4-688 runtime_stack

Parameter	Mandatory	Type	Description
name	Yes	String	Technology stack name.
type	Yes	String	Technology stack type. Value: Java, Tomcat, Nodejs, Php, Docker, or Python. - For VM-based deployment, Java, Tomcat, and Nodejs are supported. - For container-based deployment, Java, Tomcat, Nodejs, Php, and Docker are supported.
version	Yes	String	Technology stack version.
deploy_mode	Yes	String	Deployment mode. - container - virtualmachine

Table 4-689 source

Parameter	Mandatory	Type	Description
kind	Yes	String	Component source kind. - code - package - image
version	No	String	Version number.

Parameter	Mandatory	Type	Description
url	No	String	Package address. This parameter is mandatory when kind is set to package or image .
storage	No	String	Storage mode. • devcloud: CodeArts release repository • obs • http: custom file address • swr This parameter is mandatory when kind is set to package or image .
http_username	No	String	Username. This parameter is mandatory when storage is set to http . Username and password authentication is supported.
http_password	No	String	Password. This parameter is mandatory when storage is set to http . Username and password authentication is supported.
http_command	No	String	Address of the custom software package. This parameter is mandatory when storage is set to http .
http_headers	No	Map<String, String>	This parameter is mandatory when storage is set to http . You can customize request headers as key-value pairs, for example, {"key": "xx", "value": "xx"}.
auth	No	String	Authentication mode. Value: iam or none. Default value: iam.
repo_auth	No	String	Authorization name, which can be obtained from the authorization list. For details, see Obtaining a Git Repository Authorization List .
repo_namespace	No	String	Namespace of the code repository.
repo_ref	No	String	Code branch or tag. Default value: master.
repo_type	No	String	Code repository type. Value: GitHub, BitBucket, or GitLab.
web_url	No	String	Code redirection URL. Example: https://github.com/example/demo.git.
repo_url	No	String	Code repository URL. Example: https://github.com/example/demo.git.

Parameter	Mandatory	Type	Description
commit_id	No	String	In the code repository, commit ID is the unique identifier of each commit and is used to identify and reference a commit. The value contains 8 to 40 characters and starts with a lowercase letter or digit.

Table 4-690 build

Parameter	Mandatory	Type	Description
parameters	No	Map<String, Object>	This parameter is provided only when no ID is available during build creation. See Table 4-691 .

Table 4-691 parameters

Parameter	Mandatory	Type	Description
build_cmd	No	String	Compilation command. By default: 1. When build.sh exists in the root directory, the command is ./build.sh . 2. When build.sh does not exist in the root directory, the command varies depending on the OS. Example: • Java and Tomcat: mvn clean package • Nodejs: npm build
dockerfile_path	No	String	Address of the Docker file. By default, the Docker file is in the root directory (./).
artifact_namespace	No	String	Build archive organization. Default value: cas_{project_id}.
cluster_id	Yes	String	ID of the cluster to be built.
node_label_selector	No	Map<String, String>	key: label key. value: label value.
environment_id	No	String	Environment ID.

Table 4-692 external_accesses

Parameter	Mandatory	Type	Description
protocol	No	String	External access type. Value: http or https.
address	No	String	External access address.
forward_port	No	Integer	External access port.

Table 4-693 tolerations

Parameter	Mandatory	Type	Description
key	No	String	Taint key.
operator	No	String	Operator. - Equal: matches the nodes with the specified taint key (mandatory) and value. If the taint value is left blank, all taints with the same key will be matched. - Exists: matches nodes with the specified taint key. In this case, the taint value cannot be specified. Not specifying any taint key means all taints are tolerated.
value	No	String	Taint value. This parameter is mandatory when operator is set to Equal .
effect	No	String	Taint policy. - NoScheduler: Only the NoSchedule taint is matched. - PreferNoSchedule: Only the PreferNoSchedule taint is matched. - NoExecute: Only the NoExecute taint is matched. - If this parameter is not set, all taint effects are matched.
toleration_seconds	No	Integer	Tolerance time window. This parameter can be set only when effect is set to NoExecute . Within the tolerance time window, pods still run on the node with taints. After the time expires, the pods will be evicted.

Table 4-694 multi_containers

Parameter	Mandatory	Type	Description
container_name	Yes	String	Container name.
source	Yes	Object	Source of the code or software package. See Table 4-689 . In multi-container mode, only images are supported.
limit_cpu	No	Number	Maximum CPU limit. Unit: core.
limit_memory	No	Number	Maximum memory size. Unit: GiB.
request_cpu	No	Number	Requested CPU resources. Unit: core.
request_memory	No	Number	Requested memory. Unit: GiB.
envs	No	Array of objects	Environment variable list. See Table 4-686 .
liveness_probe	No	Object	Container liveness probe. See Table 4-683 .
readiness_probe	No	Object	Container service probe. See Table 4-683 .
startup_probe	No	Object	Container startup probe. See Table 4-683 .
command	No	Object	Startup command. See Table 4-670 .
post_start	No	Object	Post-start processing. See Table 4-671 .
pre_stop	No	Object	Pre-stop processing. See Table 4-671 .
logs	No	Array of objects	Log collection. See Table 4-679 .
storages	No	Array of objects	Storage. See Table 4-663 .
security_context	No	Object	Security policy. See Table 4-677 .
config_encryption	No	Object	This parameter is mandatory when the encryption key is configured for a multi-container component. See Table 4-699 .

Table 4-695 ports

Parameter	Mandatory	Type	Description
name	Yes	String	Port name.
port	Yes	String	Access port.
target_port	Yes	String	Container port.

Table 4-696 update_strategy

Parameter	Mandatory	Type	Description
type	No	String	Component upgrade policy. Options: - RollingUpdate (default): New pods are created gradually, and then old pods are deleted. - Recreate: Old pods are deleted, and then new pods are created.
max_unavailable	No	- String - Integer	Maximum number of unavailable pods. That is, the number or percentage of pods that can be deleted based on the value of replica . The default value is 25% . This parameter can be set when workload_kind is set to deployment . - String (%): the maximum percentage of unavailable pods to the total number of pods. - Integer: the maximum number of unavailable pods.
max_surge	No	- String - Integer	Maximum surge. That is, the number or percentage of pods allowed based on the value of replica . The default value is 25% . This parameter can be set only when workload_kind is set to deployment . - String (%) - Integer

Parameter	Mandatory	Type	Description
termination_period_seconds	No	Integer	Scale-in time window. That is, the time window for executing commands before a workload stops. The default value is 30s.
min_ready_seconds	No	Integer	Minimum ready time of a pod. That is, the minimum ready time of a new pod without any container crash. A pod is considered available only after the minimum ready time is exceeded. The default value is 0 .
progress_deadline_seconds	No	Integer	Maximum upgrade duration. That is, the time (in seconds) the system waits for before reporting a component upgrade failure. If this parameter is specified, the value of this parameter must be greater than that of min_ready_seconds .
revision_history_limit	No	Integer	Maximum number of pods to retain. That is, the number of old pods to retain to allow rollback. The default value is 10 .

Table 4-697 configs

Parameter	Mandatory	Type	Description
config_id	Yes	String	Unified configuration ID.
mount_path	Yes	String	Mount path.

Table 4-698 auto_lts_config

Parameter	Mandatory	Type	Description
enable	Yes	Boolean	Whether to enable automatic LTS configuration.
lts_log_path	No	Array of String	Access configuration log path.

Table 4-699 config_encryption

Parameter	Mandatory	Type	Description
key_id	Yes	String	User key ID in DEW.

Response

Parameter	Type	Description
name	String	Release task name.
description	String	Release task description.
model_version	Integer	Release task model. Only 2 is supported, indicating the new model.
config_mode	String	Release task deployment mode. • ui • template
id	String	Release task ID.
fail_mode	String	Best effort policy. • stop • continue
enterprise_project_id	String	Enterprise project ID.
pipeline_declaration	List of Object	Stage parameters. This parameter is mandatory when config_mode is ui . See Table 4-700 .

Table 4-700 pipeline_declaration

Parameter	Type	Description
sources	List of Object	Template package list. See Table 4-701 .
stages	List of Object	Stage parameters. See Table 4-704 .

Table 4-701 template_sources

Parameter	Type	Description
template_source	Object	Template package information. See Table 4-702 .

Parameter	Type	Description
input_values	Map<String, String>	Template package input information.

Table 4-702 template_source

Parameter	Type	Description
storage	String	Storage mode of the component template package. • devcloud: CodeArts release repository • obs • http: custom file address
url	String	Address of the component template package. This parameter is valid only when the component template package storage mode is http .
http_username	String	Username. This parameter is valid only when the component template package storage mode is http and the security authentication mode is username and password authentication.
http_password	String	Password. This parameter is valid only when the component template package storage mode is http and the security authentication mode is username and password authentication.
http_headers	Array of objects	Request header. See Table 4-703 . This parameter is valid only when the component template package storage mode is http and the security authentication mode is custom header authentication.

Table 4-703 http_headers

Parameter	Type	Description
key	String	Request key.
value	String	Request value.

Table 4-704 stages

Parameter	Type	Description
name	String	Stage name, which is unique in a release task.
display_name	String	Display name of a stage. If this parameter is left blank, the value is the same as the stage name.
task_group	Array of String	Task group. The parameters in the task group must correspond to the group_index field in Table 4-705 .
tasks	Array of objects	Task parameters. See Table 4-705 .

Table 4-705 tasks

Parameter	Type	Description
name	String	Task name, which is unique in a single stage.
display_name	String	Display name of a task. If this parameter is left blank, the value is the same as the task name.
order_index	Integer	Task execution sequence. Tasks with the same group_index value belong to the same task group. The execution sequence is unique in a task group.
group_index	String	Task group. If the task groups are the same, the tasks are in the same group. The value is the same as that of task_group in Table 4-704 .
task_type_id	Object	Task execution parameters. See Table 4-706 .
input	Object	Task execution parameter, which is determined based on the task type. • For a build task in Table 4-656, see Table 4-657. • For a checkpoint task in Table 4-656, see Table 4-658. • For an application task in Table 4-656, see Table 4-660. • For a component task in Table 4-656, see Table 4-661.

Table 4-706 task_type_id

Parameter	Type	Description
category	String	Task type. • Build • CheckPoint • Environment • Component • Application
type	String	Task execution type. • create • update • upsert: create or modify. • approval

Table 4-707 build

Parameter	Type	Description
name	String	Build task name.
id	String	Build task ID.
enterprise_project_id	String	Enterprise project ID of a build task.

Table 4-708 checkpoint

Parameter	Type	Description
mode	String	If this field is left blank, no person needs to be specified for approval. • all: If all approvers approve the application, the application is approved. • any: If any approver approves the application, the application is approved.
users	List of Object	Approver. If this parameter is left blank, no approver is required. See Table 4-709 .

Table 4-709 users

Parameter	Type	Description
id	String	User ID.
label	String	User label, which is displayed on the frontend.
name	String	User name, which is displayed on the frontend.

Table 4-710 application

Parameter	Type	Description
name	String	Application name.
description	String	Application description.
enterprise_project_id	String	Enterprise project ID.

Table 4-711 component

Parameter	Type	Description
id	String	Application component ID.
name	String	Application component name. The value contains 2 to 64 characters consisting of letters, digits, hyphens (-), and underscores (_). It starts with a letter and ends with a letter or digit.
workload_name	String	Workload name.
labels	Array of objects	Label. For container-based deployment, you can query a workload by label. For VM-based deployment, you can configure a label of the corresponding microservice name to use graceful startup and shutdown. See Table 4-712 .
runtime_stack	Object	Runtime. See Table 4-738 . The value can be obtained from the response of the API for Querying a Technology Stack .
environment_id	String	Environment ID.
description	String	Description. The value can contain up to 128 characters.

Parameter	Type	Description
source	Object	Source of the code or software package. See Table 4-739 .
build	Object	Component building. This parameter is mandatory when the component is deployed in containers and no image is used. For details, see Table 4-740 .
limit_cpu	Number	Maximum CPU limit. Unit: core.
limit_memory	Number	Maximum memory size. Unit: GiB.
request_cpu	Number	Requested CPU resources. Unit: core.
request_memory	Number	Requested memory. Unit: GiB.
version	String	Component version number. The value can contain up to 32 characters. It needs to meet the following rule: $\wedge([0-9]+)(.[0-9]+){2,3}\$$
envs	Array of objects	Environment variable list. See Table 4-736 .
replica	Integer	Number of instances.
storages	Array of objects	Storage. See Table 4-713 .
deploy_strategy	Object	Component deployment. See Table 4-716 .
command	Object	Startup command. See Table 4-720 .
post_start	Object	Post-start processing. See Table 4-721 .
pre_stop	Object	Pre-stop processing. See Table 4-721 .
mesher	Object	Multi-language access service mesh. This parameter is mandatory when Node.js is selected for the technology stack and the microservice engine is bound. See Table 4-722 .
timezone	String	Time zone where the component runs.
jvm_opts	String	JVM parameter. Example: -Xms256m -Xmx1024m. If there are multiple parameters, separate them by spaces. If the parameter is left blank, the default value is used.

Parameter	Type	Description
tomcat_opts	Object	Tomcat parameter. This parameter is valid when you use the Tomcat technology stack. See Table 4-723 .
host_aliases	Array of objects	Host alias. See Table 4-724 .
dns_policy	String	DNS policy. • Default: inherits the domain name resolution configuration of the node where the pod is located. • ClusterFirst: adds the domain name resolution configuration. • ClusterFirstWithHostNet: takes effect only for pods running in hostNetWork mode. • None: replaces the domain name resolution policy.
dns_config	Object	DNS configuration. See Table 4-725 .
workload_kind	String	Workload type. • deployment • statefulset • daemonset
security_context	Object	Security policy. See Table 4-727 .
logs	Array of objects	Log collection. See Table 4-729 .
custom_metric	Object	Custom metrics. See Table 4-730 .
affinity	Array of objects	Affinity. See Table 4-731 .
anti_affinity	Array of objects	Anti-affinity. See Table 4-731 .
liveness_probe	Object	Component liveness probe. See Table 4-733 .
readiness_probe	Object	Component service probe. See Table 4-733 .
startup_probe	Object	Component startup probe. See Table 4-733 .
refer_resources	Array of objects	Associated resource. See Table 4-734 .

Parameter	Type	Description
external_accesses	Array of Objects	External access. See Table 4-742 .
tolerations	Array of Objects	Tolerance policy. See Table 4-743 .
container_name	String	Container name.
update_strategy	Object	Component upgrade strategy. See Table 4-746 .
multi_containers	Array of Objects	Multi-container configuration. See Table 4-744 .
service_name	String	Service name configured for the inter-instance discovery of a StatefulSet.
ports	Array of Objects	Port configured for the inter-instance discovery of a StatefulSet.
workload_content	String	Workload content.
configs	Array of Objects	Unified component configuration. See Table 4-697 .
config_mode	String	Workload configuration mode.
enable_sermant_injection	Boolean	Whether to enable Sermant.
enable_sermant_spring_register	Boolean	Whether Spring Cloud Huawei microservices need to be registered with registry centers through Sermant.
enable_sermant_dubbo_register	Boolean	Whether Dubbo microservices need to be registered with registry centers through Sermant.
enable_sermant_config	Boolean	Whether to enable the Sermant dynamic configuration plug-in.
cell_id	String	Deployment cell ID.
swimlane_id	String	Lane ID.

Parameter	Type	Description
sermantAgentInjectActionFlag	Boolean	Whether JDK 17 and JDK 21 automatically add the following JVM environment variables: JDK 17: --add-opens java.base/java.lang=ALL-UNNAMED --add-opens java.base/java.net=ALL-UNNAMED --add-opens java.base/java.math=ALL-UNNAMED --add-opens java.base/sun.net.www=ALL-UNNAMED --add-opens java.base/sun.net.www.protocol.http=ALL-UNNAMED JDK 21: --add-opens java.base/java.lang=ALL-UNNAMED --add-opens java.base/java.net=ALL-UNNAMED --add-opens java.base/java.math=ALL-UNNAMED --add-opens java.base/sun.net.www=ALL-UNNAMED --add-opens java.base/sun.net.www.protocol.http=ALL-UNNAMED -Dnet.bytebuddy.experimental=true
sys_config	Map<String, String>	Uses a compressed package to deploy component configuration items in the VM environment.
image_pull_secrets	Array of String	List of image pull secrets.
auto_lts_config	Object	Automatic LTS configuration. See Table 4-698 .
config_encryption	Object	This parameter is mandatory when the encryption key is configured for a single-container component. See Table 4-699 .

Table 4-712 labels

Parameter	Type	Description
key	String	Label name.
value	String	Label value.

Table 4-713 storages

Parameter	Type	Description
type	String	Data storage type. • HostPath: host path for local disk mounting. • EmptyDir: temporary directory for local disk mounting. • ConfigMap: configuration item for local disk mounting. • Secret: secret for local disk mounting. • PersistentVolumeClaim: cloud storage mounting.
name	String	Name of the storage disk.
parameters	Object	For details about the parameters of each data storage type, see Table 4-714 .
mounts	Array of objects	Mount path of data storage. See Table 4-715 .

Table 4-714 component_storage_parameters

Parameter	Type	Description
path	String	Host path. This parameter is applicable to the HostPath storage type. This parameter is mandatory when type is set to HostPath .
name	String	Name of a configuration item, secret, or PVC, which is applicable to ConfigMap, Secret, or PersistentVolumeClaim storage, respectively. This parameter is mandatory when type is set to ConfigMap , Secret , or PersistentVolumeClaim .
default_mode	Integer	Mounting permission. The value is in decimal format. Example: 384. This parameter is applicable to ConfigMap and secret storage.
medium	String	This parameter is applicable to EmptyDir storage. If it is not specified, the default disk medium is used. If it is set to memory , memory storage is enabled.

Table 4-715 mounts

Parameter	Type	Description
path	String	Mount path.

Parameter	Type	Description
sub_path	String	Subpath of the mount path.
read_only	Boolean	Read-only or not.

Table 4-716 deploy_strategy

Parameter	Type	Description
type	String	Deployment type. - OneBatchRelease - RollingRelease - GrayRelease
rolling_release	Object	Rolling deployment parameter, which is mandatory when type is set to RollingRelease . For details, see Table 4-717 .
gray_release	Object	Dark launch upgrade parameter, which is mandatory when type is set to GrayRelease . For details, see Table 4-718 .

Table 4-717 rolling_release

Parameter	Type	Description
batches	Integer	Deployment batches.

Table 4-718 gray_release

Parameter	Type	Description
type	String	Dark launch policy. - WEIGHT: dark launch by traffic. - CONTENT: dark launch by content.
first_batch_weight	Integer	Proportion of first-batch dark launch traffic. This parameter is mandatory when type is set to WEIGHT .
first_batch_replica	Integer	Number of first-batch dark launch instances. This parameter is mandatory when type is set to WEIGHT .

Parameter	Type	Description
remaining_batch	Integer	Number of batches for remaining instance deployment. After the first batch is completed, the remaining instances will be deployed in specified batches. Example: If there are five instances remaining to be deployed in three batches, these five instances will be upgraded based on 2:2:1. This parameter is mandatory when type is set to WEIGHT .
deployment_mode	Integer	Deployment mode. 1: The upstream is a microservice gateway. 3: The upstream is a microservice. 4: The upstream is a load balancer.
replica_surge_mode	String	Mode of adding a dark launch instance. This parameter is mandatory when type is set to CONTENT . - MIRROR: blue-green. - EXTRA: canary (increase, then decrease) - NOSURGE: canary (decrease, then increase)
rule_match_mode	String	Effective mode of a dark launch rule. This parameter is mandatory when type is set to CONTENT . - ALL: All conditions are met. - ANY: Any condition is met.
rules	Object	Dark launch rule. See Table 4-719 . This parameter is mandatory when type is set to CONTENT .

Table 4-719 gray_rules

Parameter	Type	Description
type	String	Match type. Currently, only HEADER is supported.
key	String	Parameter name.
value	String	Condition value.
condition	String	Condition type. - EQUAL - MATCH - IN: enumeration

Table 4-720 command

Parameter	Type	Description
command	Array of String	Command for controlling container running.
args	Array of String	Parameter for controlling container running. Example: -port=8080. If there are multiple parameters, separate them by line breaks.

Table 4-721 component_lifecycle

Parameter	Type	Description
type	String	Processing method. • http • command
scheme	String	HTTP request type, which can be HTTP or HTTPS. This parameter is valid only when type is set to http .
host	String	Custom IP address. If this parameter is left blank, the pod IP address is used. This parameter is valid only when type is set to http .
port	Integer	Port number. This parameter is valid only when type is set to http .
path	String	Request path. This parameter is valid only when type is set to http .
command	Array of String	Command list. This parameter is valid only when type is set to command .

Table 4-722 mesher

Parameter	Type	Description
port	Integer	Process listening port.

Table 4-723 tomcat_opt

Parameter	Type	Description
server_xml	String	1. Listening port in the server.xml file will take effect even if it has been configured in public network access. 2. The custom Tomcat configuration may conflict with the existing port. Check whether the custom port is occupied.

Table 4-724 host_aliases

Parameter	Type	Description
ip	String	IP address.
hostname	Array of String	Host alias.

Table 4-725 dns_config

Parameter	Type	Description
nameservers	Array of String	IP address list of the DNS server of the pod.
searches	Array of String	(Optional) List of DNS search domains used to search for host names in the pod.
options	Array of objects	(Optional) List of objects. Each object may have a name (mandatory) and a value (optional). The contents of the objects here will be merged into the options generated from the specified DNS policy. Duplicate entries will be deleted. See Table 4-726 .

Table 4-726 options

Parameter	Type	Description
name	String	Operation name.
value	String	Operation value.

Table 4-727 security_context

Parameter	Type	Description
run_as_user	Integer	User for running containers. Example: To run containers as user root , set the user ID to 0 .
run_as_group	Integer	Owner group specified for running container processes.
capabilities	Object	Capability set. See Table 4-728 .

Table 4-728 capabilities

Parameter	Type	Description
add	Array of String	Grant the Linux permission to the running user.
drop	Array of String	Remove the Linux permission of the running user.

Table 4-729 logs

Parameter	Type	Description
log_path	String	Container log path.
rotate	String	Interval for dumping logs.
host_path	String	Mounted host path.
host_extend_path	String	Host extension path. By extending the host path, you can distinguish mounting from different containers in the same host path. • None: The extended path is not used. • PodUID: extend the host path based on the pod ID. • PodName: extend the host path based on the pod name. • PodUID/ContainerName: extend the host path based on the pod ID and container name. • PodName/ContainerName: extend the host path based on the pod name and container name.

Table 4-730 custom_metric

Parameter	Type	Description
path	String	Collection path. Example: /metrics.
port	Integer	Collection port. Example: 9090.
dimensions	String	Monitoring dimension. Example: cpu_usage,mem_usage.

Table 4-731 component_affinity

Parameter	Type	Description
kind	String	Object kind. • node • pod
condition	String	Policy type. • required: The requirement must be met. • preferred: The requirement is met as much as possible.
match_expressions	Array of objects	Scheduling policy rule. See Table 4-732 .
weight	String	Weight. • The value is 100 when condition is set to required. • Enter an integer ranging from 1 to 100 when condition is set to preferred.

Table 4-732 match_expressions

Parameter	Type	Description
key	String	Condition key.
value	String	Condition value.
operation	String	Operation.

Table 4-733 component_probe

Parameter	Type	Description
type	String	Type. Value: http, tcp, or command.

Parameter	Type	Description
delay	Integer	Interval between the startup and detection.
timeout	Integer	Detection timeout interval.
scheme	String	Request type. Value: HTTP or HTTPS. This parameter is valid only when type is set to http .
host	String	Custom IP address. If this parameter is left blank, the pod IP address is used. This parameter is valid only when type is set to http .
port	Integer	Port number. This parameter is valid only when type is set to http or tcp .
path	String	Request path. This parameter is valid only when type is set to http .
command	Array of String	Command list. This parameter is valid only when type is set to command .

Table 4-734 refer_resources

Parameter	Type	Description
id	String	Resource ID.
type	String	Resource type. • vpc • eip • elb • cce • ecs • dcs • rds • cse: ServiceComb engine. • nacos: registry/configuration center. • customServer • pvc • gaussdb • dms_rocketmq • css_es: Elasticsearch
parameters	Object	Resource parameters. See Table 4-735 .

Table 4-735 refer_resource_parameter

Parameter	Type	Description
namespace	String	Namespace.
name	String	Resource name.
capacity	Integer	Storage capacity, in GiB.
type	String	Volume declaration type. • efs • obs • sfs • evs
class	String	Volume declaration subtype. • csi-sfsturbo: fixed when type is set to efs. • csi-obs: fixed when type is set to obs. • csi-nas: fixed when type is set to sfs. • csi-disk: fixed when type is set to evs.
obs_volume_type	String	OBS volume type. Default value: STANDARD.
access_mode	String	Storage volume access mode. • ReadWriteOnce: A volume can be mounted as read-write by a single node. • ReadWriteMany: A volume can be mounted as read-write by many nodes.
availableZone	String	AZ.
volume_type	String	Volume type.

Table 4-736 envs

Parameter	Type	Description
name	String	Variable name.
value	String	Variable value.
value_from	Object	Variable reference. See Table 4-737 .

Table 4-737 value_from

Parameter	Type	Description
reference_type	String	Reference type. • configMapKey • secretKey
name	String	Name of the configuration item or secret.
key	String	Key of the configuration item or secret.
optional	Boolean	Whether the key of the configuration item or secret is mandatory.

Table 4-738 runtime_stack

Parameter	Type	Description
name	String	Technology stack name.
type	String	Technology stack type. Value: Java, Tomcat, Nodejs, Php, Docker, or Python. • For VM-based deployment, Java, Tomcat, and Nodejs are supported. • For container-based deployment, Java, Tomcat, Nodejs, Php, and Docker are supported.
version	String	Technology stack version.
deploy_mode	String	Deployment mode. • container • virtualmachine

Table 4-739 source

Parameter	Mandatory	Type	Description
kind	Yes	String	Component source kind. • code • package • image
version	No	String	Version number.
url	No	String	Package address. This parameter is mandatory when kind is set to package or image .

Parameter	Mandatory	Type	Description
storage	No	String	Storage mode. • devcloud: CodeArts release repository • obs • http: custom file address • swr This parameter is mandatory when kind is set to package or image .
http_username	No	String	Username. This parameter is mandatory when storage is set to http . Username and password authentication is supported.
http_password	No	String	Password. This parameter is mandatory when storage is set to http . Username and password authentication is supported.
http_command	No	String	Address of the custom software package. This parameter is mandatory when storage is set to http .
http_headers	No	Map<String, String>	This parameter is mandatory when storage is set to http . You can customize request headers as key-value pairs, for example, {"key":"xx","value":"xx"}.
auth	No	String	Authentication mode. Value: iam or none. Default value: iam.
repo_auth	No	String	Authorization name, which can be obtained from the authorization list. For details, see Obtaining a Git Repository Authorization List .
repo_namespace	No	String	Namespace of the code repository.
repo_ref	No	String	Code branch or tag. Default value: master.
repo_type	No	String	Code repository type. Value: GitHub, BitBucket, or GitLab.
web_url	No	String	Code redirection URL. Example: https://github.com/example/demo.git.
repo_url	No	String	Code repository URL. Example: https://github.com/example/demo.git.

Parameter	Mandatory	Type	Description
commit_id	No	String	In the code repository, commit ID is the unique identifier of each commit and is used to identify and reference a commit. The value contains 8 to 40 characters and starts with a lowercase letter or digit.

Table 4-740 build

Parameter	Type	Description
parameters	Map<String, Object>	This parameter is provided only when no ID is available during build creation. See Table 4-741 .

Table 4-741 parameters

Parameter	Type	Description
build_cmd	String	Compilation command. By default: 1. When build.sh exists in the root directory, the command is ./build.sh . 2. When build.sh does not exist in the root directory, the command varies depending on the OS. Example: • Java and Tomcat: mvn clean package • Nodejs: npm build
dockerfile_path	String	Address of the Docker file. By default, the Docker file is in the root directory (./).
artifact_name_space	String	Build archive organization. Default value: cas_{project_id}.
cluster_id	String	ID of the cluster to be built.
node_label_selector	Map<String, String>	key: label key. value: label value.
environment_id	String	Environment ID.

Table 4-742 external_accesses

Parameter	Type	Description
protocol	String	External access type. Value: http or https.

Parameter	Type	Description
address	String	External access address.
forward_port	Integer	External access port.

Table 4-743 tolerations

Parameter	Type	Description
key	String	Taint key.
operator	String	Operator. • Equal: matches the nodes with the specified taint key (mandatory) and value. If the taint value is left blank, all taints with the same key will be matched. • Exists: matches nodes with the specified taint key. In this case, the taint value cannot be specified. Not specifying any taint key means all taints are tolerated.
value	String	Taint value. This parameter is mandatory when operator is set to Equal .
effect	String	Taint policy. • NoScheduler: Only the NoSchedule taint is matched. • PreferNoSchedule: Only the PreferNoSchedule taint is matched. • NoExecute: Only the NoExecute taint is matched. • If this parameter is not set, all taint effects are matched.
toleration_seconds	Integer	Tolerance time window. This parameter can be set only when effect is set to NoExecute . Within the tolerance time window, pods still run on the node with taints. After the time expires, the pods will be evicted.

Table 4-744 multi_containers

Parameter	Type	Description
container_name	String	Container name.

Parameter	Type	Description
source	Object	Source of the code or software package. See Table 4-739 . In multi-container mode, only images are supported.
limit_cpu	Number	Maximum CPU limit. Unit: core.
limit_memory	Number	Maximum memory size. Unit: GiB.
request_cpu	Number	Requested CPU resources. Unit: core.
request_memory	Number	Requested memory. Unit: GiB.
envs	Array of objects	Environment variable list. See Table 4-736 .
liveness_probe	Object	Container liveness probe. See Table 4-733 .
readiness_probe	Object	Container service probe. See Table 4-733 .
startup_probe	Object	Container startup probe. See Table 4-733 .
command	Object	Startup command. See Table 4-720 .
post_start	Object	Post-start processing. See Table 4-721 .
pre_stop	Object	Pre-stop processing. See Table 4-721 .
logs	Array of objects	Log collection. See Table 4-729 .
storages	Array of objects	Storage. See Table 4-713 .
security_context	Object	Security policy. See Table 4-727 .
config_encryption	Object	This parameter is mandatory when the encryption key is configured for a multi-container component. See Table 4-699 .

Table 4-745 ports

Parameter	Type	Description
name	String	Port name.
port	String	Access port.
target_port	String	Container port.

Table 4-746 update_strategy

Parameter	Type	Description
type	String	Component upgrade policy. Options: - RollingUpdate (default): New pods are created gradually, and then old pods are deleted. - Recreate: Old pods are deleted, and then new pods are created.
max_unavailable	- String - Integer	Maximum number of unavailable pods. That is, the number or percentage of pods that can be deleted based on the value of replica . The default value is 25% . This parameter can be set when workload_kind is set to deployment . - String (%): the maximum percentage of unavailable pods to the total number of pods. - Integer: the maximum number of unavailable pods.
max_surge	- String - Integer	Maximum surge. That is, the number or percentage of pods allowed based on the value of replica . The default value is 25% . This parameter can be set only when workload_kind is set to deployment . - String (%) - Integer
termination_period_seconds	Integer	Scale-in time window. That is, the time window for executing commands before a workload stops. The default value is 30s.
min_ready_seconds	Integer	Minimum ready time of a pod. That is, the minimum ready time of a new pod without any container crash. A pod is considered available only after the minimum ready time is exceeded. The default value is 0 .
progress_deadline_seconds	Integer	Maximum upgrade duration. That is, the time (in seconds) the system waits for before reporting a component upgrade failure. If this parameter is specified, the value of this parameter must be greater than that of min_ready_seconds .

Parameter	Type	Description
revision_history_limit	Integer	Maximum number of pods to retain. That is, the number of old pods to retain to allow rollback. The default value is 10 .

Table 4-747 configs

Parameter	Type	Description
conifg_id	String	Unified configuration ID.
mount_path	String	Mount path.

Table 4-748 auto_lts_config

Parameter	Type	Description
enable	Boolean	Whether to enable automatic LTS configuration.
lts_log_path	Array of String	Access configuration log path.

Table 4-749 config_encryption

Parameter	Type	Description
key_id	String	User key ID in DEW.

Example Request

Modify release task **release-template**.

```
{
  "model_version": 2,
  "name": "release-template",
  "config_mode": "template",
  "description": "",
  "enterprise_project_id": "0",
  "sources": [
    {
      "template_source": {
        "storage": "obs",
        "url": "obs://obs-zyh/create-component.zip",
        "http_username": "",
        "http_password": "",
        "http_headers": []
      },
      "input_values": {
        "env_name": "demo",
        "image1": "swr.ap-southeast-1.myhuaweicloud.com/cse/java-cce:v3",
        "application_id": "demo",

```

```
        "component_name1": "demo",  
        "cce_id": "demo"  
      }  
    ]  
  }  
}
```

Example Response

```
{  
  "id": "4105b37a-947e-4aca-90d8-cec7c04c6f38",  
  "enterprise_project_id": "0",  
  "name": "release-gn7mre11",  
  "enterprise_project": "default",  
  "model_version": 2,  
  "creator": "zhangyuhao-3",  
  "status": null,  
  "revisor": "zhangyuhao-3",  
  "create_at": 1739865312582,  
  "update_at": 1739865967776,  
  "fail_mode": "stop",  
  "strategy_type": 0,  
  "description": "test",  
  "pipeline_declaration": {  
    "stages": [  
      {  
        "id": 15,  
        "name": "75foki",  
        "approval_type": 0,  
        "display_name": "Stage"  
        "type": null,  
        "pipeline_id": null,  
        "order_index": 0,  
        "status": null,  
        "task_group": [  
          "0"  
        ],  
        "tasks": [  
          {  
            "id": 17,  
            "name": "bkdi55",  
            "display_name": "Create application-187",  
            "task_type_id": {  
              "category": "Application",  
              "owner": null,  
              "provider": null,  
              "type": "upsert"  
            },  
            "input": {  
              "displayName": "Create application-187",  
              "originalName": "bkdi55",  
              "name": "xxz",  
              "description": "",  
              "enterprise_project_id": "0",  
              "checked": true  
            },  
            "output": {},  
            "status": null,  
            "order_index": 0,  
            "skipped": false,  
            "group_index": "0",  
            "delay_time": 0,  
            "error_details": null,  
            "stage_name": "75foki"  
          }  
        ]  
      }  
    ],  
    "config_mode": "ui"  
  }  
}
```

```
}  
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.7.2.4 Operating a Release Task Based on the Release Task ID

Function

This API is used to release, stop, and continue a release task.

URI

POST /v3/{project_id}/pipeline/pipelines/{pipeline_id}/action

Table 4-750 Path parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
pipeline_id	Yes	String	Release task ID. See Obtaining the Release Task List .

Table 4-751 Query parameters

Parameter	Mandatory	Type	Description
action_id	Yes	String	Action type. • start • stop • retry • approval • continue • rollback

Request

Table 4-752 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Table 4-753 Request body parameters

Parameter	Mandatory	Type	Description
selected_tasks	No	List of object	Task to be executed. When action_id is set to start , if this parameter is not specified or is set to null , all tasks are executed. See Table 4-754 .
task_name	No	String	Name of the task to be executed. This parameter is mandatory when action_id is set to approval .

Parameter	Mandatory	Type	Description
approval_type	No	String	Approval type. This parameter is mandatory when action_id is set to approval , indicating that the application is approved or refused. • agree • refuse
message	No	String	Approval information.

Table 4-754 selected_tasks

Parameter	Mandatory	Type	Description
stage_name	Yes	String	Stage name.
tasks	Yes	List of string	Name of the task to be executed.

Response

Parameter	Type	Description
id	String	Release task execution ID.
pipeline_id	String	Release task ID.
status	Integer	Release task status. • 1: initialization successful • 2: release successful • 3: release failed • 4: releasing • 5: to be approved • 7: pause • 9: skip • 10: rolling back • 11: rollback successful • 12: rollback failed • 13: stopping • 14: restarting

Example Request

Start a release task.

```
{
  "selected_tasks": [
    {
      "stage_name": "899tih",
      "tasks": [
        "ujgdh4"
      ]
    },
    {
      "stage_name": "uipfzw",
      "tasks": [
        "ax8hus"
      ]
    }
  ]
}
```

Example Response

```
{
  "id": 286541,
  "pipeline_id": "caf409b1-10df-483a-b14d-7b05f7e12c88",
  "status": 4
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.7.2.5 Viewing Release Task Details Based on the Release Task ID

Function

This API is used to view release task details based on the release task ID.

URI

GET /v3/{project_id}/pipeline/pipelines/{pipeline_id}

Table 4-755 Path parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
pipeline_id	Yes	String	Release task ID. See Obtaining the Release Task List .

Request

Table 4-756 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

Table 4-757 Response parameters

Parameter	Type	Description
name	String	Release task name.
description	String	Release task description.
model_version	Integer	Release task model. Only 2 is supported, indicating the new model.
config_mode	String	Release task deployment mode. - ui - template
id	String	Release task ID.

Parameter	Type	Description
fail_mode	String	Best effort policy. • stop • continue
enterprise_project_id	String	Enterprise project ID.
pipeline_declaration	List of Object	Stage parameters. This parameter is mandatory when config_mode is ui . See Table 4-758 .

Table 4-758 pipeline_declaration

Parameter	Type	Description
sources	List of Object	Template package list. See Table 4-759 .
stages	List of Object	Stage parameters. See Table 4-762 .

Table 4-759 template_sources

Parameter	Type	Description
template_source	Object	Template package information. See Table 4-760 .
input_values	Map<String, String>	Template package input information.

Table 4-760 template_source

Parameter	Type	Description
storage	String	Storage mode of the component template package. • devcloud: CodeArts release repository • obs • http: custom file address
url	String	Address of the component template package. This parameter is valid only when the component template package storage mode is http .

Parameter	Type	Description
http_username	String	Username. This parameter is valid only when the component template package storage mode is http and the security authentication mode is username and password authentication.
http_password	String	Password. This parameter is valid only when the component template package storage mode is http and the security authentication mode is username and password authentication.
http_headers	Array of objects	Request header. See Table 4-761 . This parameter is valid only when the component template package storage mode is http and the security authentication mode is custom header authentication.

Table 4-761 http_headers

Parameter	Type	Description
key	String	Request key.
value	String	Request value.

Table 4-762 stages

Parameter	Type	Description
name	String	Stage name, which is unique in a release task.
display_name	String	Display name of a stage. If this parameter is left blank, the value is the same as the stage name.
task_group	Array of String	Task group. The parameters in the task group must correspond to the group_index field in Table 4-763 .
tasks	Array of object	Task parameters. See Table 4-763 .

Table 4-763 tasks

Parameter	Type	Description
name	String	Task name, which is unique in a single stage.
display_name	String	Display name of a task. If this parameter is left blank, the value is the same as the task name.
order_index	Integer	Task execution sequence. Tasks with the same group_index value belong to the same task group. The execution sequence is unique in a task group.
group_index	String	Task group. If the task groups are the same, the tasks are in the same group. The value is the same as that of task_group in Table 4-762 .
task_type_id	Object	Task execution parameters. See Table 4-764 .
input	Object	Task execution parameter, which is determined based on the task type. • For a build task in Table 4-764, see Table 4-765. • For a checkpoint task in Table 4-764, see Table 4-766. • For an application task in Table 4-764, see Table 4-768. • For a component task in Table 4-764, see Table 4-769.

Table 4-764 task_type_id

Parameter	Type	Description
category	String	Task type. • Build • CheckPoint • Environment • Component • Application
type	String	Task execution type. • create • update • upsert: create or modify. • approval

Table 4-765 build

Parameter	Type	Description
name	String	Build task name.
id	String	Build task ID.
enterprise_project_id	String	Enterprise project ID of a build task.

Table 4-766 checkpoint

Parameter	Type	Description
mode	String	If this field is left blank, no person needs to be specified for approval. - all: If all approvers approve the application, the application is approved. - any: If any approver approves the application, the application is approved.
users	List of Object	Approver. If this parameter is left blank, no approver is required. See Table 4-767 .

Table 4-767 users

Parameter	Type	Description
id	String	User ID.
label	String	User label, which is displayed on the frontend.
name	String	User name, which is displayed on the frontend.

Table 4-768 application

Parameter	Type	Description
name	String	Application name.
description	String	Application description.
enterprise_project_id	String	Enterprise project ID.

Table 4-769 component

Parameter	Type	Description
id	String	Application component ID.
name	String	Application component name. The value contains 2 to 64 characters consisting of letters, digits, hyphens (-), and underscores (_). It starts with a letter and ends with a letter or digit.
workload_name	String	Workload name.
labels	Array of objects	Label. For container-based deployment, you can query a workload by label. For VM-based deployment, you can configure a label of the corresponding microservice name to use graceful startup and shutdown. See Table 4-770 .
runtime_stack	Object	Runtime. See Table 4-796 . The value can be obtained from the response of the API for Querying a Technology Stack .
environment_id	String	Environment ID.
description	String	Description. The value can contain up to 128 characters.
source	Object	Source of the code or software package. See Table 4-797 .
build	Object	Component building. This parameter is mandatory when the component is deployed in containers and no image is used. For details, see Table 4-798 .
limit_cpu	Number	Maximum CPU limit. Unit: core.
limit_memory	Number	Maximum memory size. Unit: GiB.
request_cpu	Number	Requested CPU resources. Unit: core.
request_memory	Number	Requested memory. Unit: GiB.
version	String	Component version number. The value can contain up to 32 characters. It needs to meet the following rule: ^([0-9]+)(.[0-9]+){2,3}\$
envs	Array of objects	Environment variable list. See Table 4-794 .
replica	Integer	Number of instances.

Parameter	Type	Description
storages	Array of objects	Storage. See Table 4-771 .
deploy_strategy	Object	Component deployment. See Table 4-774 .
command	Object	Startup command. See Table 4-778 .
post_start	Object	Post-start processing. See Table 4-779 .
pre_stop	Object	Pre-stop processing. See Table 4-779 .
mesher	Object	Multi-language access service mesh. This parameter is mandatory when Node.js is selected for the technology stack and the microservice engine is bound. See Table 4-780 .
timezone	String	Time zone where the component runs.
jvm_opts	String	JVM parameter. Example: -Xms256m -Xmx1024m. If there are multiple parameters, separate them by spaces. If the parameter is left blank, the default value is used.
tomcat_opts	Object	Tomcat parameter. This parameter is valid when you use the Tomcat technology stack. See Table 4-781 .
host_aliases	Array of objects	Host alias. See Table 4-782 .
dns_policy	String	DNS policy. • Default: inherits the domain name resolution configuration of the node where the pod is located. • ClusterFirst: adds the domain name resolution configuration. • ClusterFirstWithHostNet: takes effect only for pods running in hostNetWork mode. • None: replaces the domain name resolution policy.
dns_config	Object	DNS configuration. See Table 4-783 .
workload_kind	String	Workload type. • deployment • statefulset • daemonset
security_context	Object	Security policy. See Table 4-785 .

Parameter	Type	Description
logs	Array of objects	Log collection. See Table 4-787 .
custom_metric	Object	Custom metrics. See Table 4-788 .
affinity	Array of objects	Affinity. See Table 4-789 .
anti_affinity	Array of objects	Anti-affinity. See Table 4-789 .
liveness_probe	Object	Component liveness probe. See Table 4-791 .
readiness_probe	Object	Component service probe. See Table 4-791 .
startup_probe	Object	Component startup probe. See Table 4-791 .
refer_resources	Array of objects	Associated resource. See Table 4-792 .
external_accesses	Array of Objects	External access. See Table 4-800 .
tolerations	Array of Objects	Tolerance policy. See Table 4-801 .
container_name	String	Container name.
update_strategy	Object	Component upgrade strategy. See Table 4-804 .
multi_containers	Array of Objects	Multi-container configuration. See Table 4-802 .
service_name	String	Service name configured for the inter-instance discovery of a StatefulSet.
ports	Array of Objects	Port configured for the inter-instance discovery of a StatefulSet.
workload_content	String	Workload content.
configs	Array of Objects	Unified component configuration. See Table 4-805 .
config_mode	String	Workload configuration mode.
enable_sermant_injection	Boolean	Whether to enable Sermant.
enable_sermant_spring_register	Boolean	Whether Spring Cloud Huawei microservices need to be registered with registry centers through Sermant.

Parameter	Type	Description
enable_serment_dubbo_register	Boolean	Whether Dubbo microservices need to be registered with registry centers through Serment.
enable_serment_config	Boolean	Whether to enable the Serment dynamic configuration plug-in.
cell_id	String	Deployment cell ID.
swimlane_id	String	Lane ID.
sermentAgentInjectActionFlag	Boolean	Whether JDK 17 and JDK 21 automatically add the following JVM environment variables: JDK 17: --add-opens java.base/java.lang=ALL-UNNAMED --add-opens java.base/java.net=ALL-UNNAMED --add-opens java.base/java.math=ALL-UNNAMED --add-opens java.base/sun.net.www=ALL-UNNAMED --add-opens java.base/sun.net.www.protocol.http=ALL-UNNAMED JDK 21: --add-opens java.base/java.lang=ALL-UNNAMED --add-opens java.base/java.net=ALL-UNNAMED --add-opens java.base/java.math=ALL-UNNAMED --add-opens java.base/sun.net.www=ALL-UNNAMED --add-opens java.base/sun.net.www.protocol.http=ALL-UNNAMED -Dnet.bytebuddy.experimental=true
sys_config	Map<String, String>	Uses a compressed package to deploy component configuration items in the VM environment.
image_pull_secrets	Array of String	List of image pull secrets.
auto_lts_config	Object	Automatic LTS configuration. See Table 4-806 .
config_encryption	Object	This parameter is mandatory when the encryption key is configured for a single-container component. See Table 4-807 .

Table 4-770 labels

Parameter	Type	Description
key	String	Label name.
value	String	Label value.

Table 4-771 storages

Parameter	Type	Description
type	String	Data storage type. • HostPath: host path for local disk mounting. • EmptyDir: temporary directory for local disk mounting. • ConfigMap: configuration item for local disk mounting. • Secret: secret for local disk mounting. • PersistentVolumeClaim: cloud storage mounting.
name	String	Name of the storage disk.
parameters	Object	For details about the parameters of each data storage type, see Table 4-772 .
mounts	Array of objects	Mount path of data storage. See Table 4-773 .

Table 4-772 component_storage_parameters

Parameter	Type	Description
path	String	Host path. This parameter is applicable to the HostPath storage type. This parameter is mandatory when type is set to HostPath .
name	String	Name of a configuration item, secret, or PVC, which is applicable to ConfigMap, Secret, or PersistentVolumeClaim storage, respectively. This parameter is mandatory when type is set to ConfigMap , Secret , or PersistentVolumeClaim .
default_mode	Integer	Mounting permission. The value is in decimal format. Example: 384. This parameter is applicable to ConfigMap and secret storage.
medium	String	This parameter is applicable to EmptyDir storage. If it is not specified, the default disk medium is used. If it is set to memory , memory storage is enabled.

Table 4-773 mounts

Parameter	Type	Description
path	String	Mount path.

Parameter	Type	Description
sub_path	String	Subpath of the mount path.
read_only	Boolean	Read-only or not.

Table 4-774 deploy_strategy

Parameter	Type	Description
type	String	Deployment type. - OneBatchRelease - RollingRelease - GrayRelease
rolling_release	Object	Rolling deployment parameter, which is mandatory when type is set to RollingRelease . For details, see Table 4-775 .
gray_release	Object	Dark launch upgrade parameter, which is mandatory when type is set to GrayRelease . For details, see Table 4-776 .

Table 4-775 rolling_release

Parameter	Type	Description
batches	Integer	Deployment batches.

Table 4-776 gray_release

Parameter	Type	Description
type	String	Dark launch policy. - WEIGHT: dark launch by traffic. - CONTENT: dark launch by content.
first_batch_weight	Integer	Proportion of first-batch dark launch traffic. This parameter is mandatory when type is set to WEIGHT .
first_batch_replica	Integer	Number of first-batch dark launch instances. This parameter is mandatory when type is set to WEIGHT .

Parameter	Type	Description
remaining_batch	Integer	Number of batches for remaining instance deployment. After the first batch is completed, the remaining instances will be deployed in specified batches. Example: If there are five instances remaining to be deployed in three batches, these five instances will be upgraded based on 2:2:1. This parameter is mandatory when type is set to WEIGHT .
deployment_mode	Integer	Deployment mode. • 1: The upstream is a microservice gateway. • 3: The upstream is a microservice. • 4: The upstream is a load balancer.
replica_surge_mode	String	Mode of adding a dark launch instance. This parameter is mandatory when type is set to CONTENT . • MIRROR: blue-green. • EXTRA: canary (increase, then decrease) • NOSURGE: canary (decrease, then increase)
rule_match_mode	String	Effective mode of a dark launch rule. This parameter is mandatory when type is set to CONTENT . • ALL: All conditions are met. • ANY: Any condition is met.
rules	Object	Dark launch rule. See Table 4-777 . This parameter is mandatory when type is set to CONTENT .

Table 4-777 gray_rules

Parameter	Type	Description
type	String	Match type. Currently, only HEADER is supported.
key	String	Parameter name.
value	String	Condition value.
condition	String	Condition type. • EQUAL • MATCH • IN: enumeration

Table 4-778 command

Parameter	Type	Description
command	Array of String	Command for controlling container running.
args	Array of String	Parameter for controlling container running. Example: -port=8080. If there are multiple parameters, separate them by line breaks.

Table 4-779 component_lifecycle

Parameter	Type	Description
type	String	Processing method. • http • command
scheme	String	HTTP request type, which can be HTTP or HTTPS. This parameter is valid only when type is set to http .
host	String	Custom IP address. If this parameter is left blank, the pod IP address is used. This parameter is valid only when type is set to http .
port	Integer	Port number. This parameter is valid only when type is set to http .
path	String	Request path. This parameter is valid only when type is set to http .
command	Array of String	Command list. This parameter is valid only when type is set to command .

Table 4-780 mesher

Parameter	Type	Description
port	Integer	Process listening port.

Table 4-781 tomcat_opt

Parameter	Type	Description
server_xml	String	1. Listening port in the server.xml file will take effect even if it has been configured in public network access. 2. The custom Tomcat configuration may conflict with the existing port. Check whether the custom port is occupied.

Table 4-782 host_aliases

Parameter	Type	Description
ip	String	IP address.
hostname	Array of String	Host alias.

Table 4-783 dns_config

Parameter	Type	Description
nameservers	Array of String	IP address list of the DNS server of the pod.
searches	Array of String	(Optional) List of DNS search domains used to search for host names in the pod.
options	Array of objects	(Optional) List of objects. Each object may have a name (mandatory) and a value (optional). The contents of the objects here will be merged into the options generated from the specified DNS policy. Duplicate entries will be deleted. See Table 4-784 .

Table 4-784 options

Parameter	Type	Description
name	String	Operation name.
value	String	Operation value.

Table 4-785 security_context

Parameter	Type	Description
run_as_user	Integer	User for running containers. Example: To run containers as user root , set the user ID to 0 .
run_as_group	Integer	Owner group specified for running container processes.
capabilities	Object	Capability set. See Table 4-786 .

Table 4-786 capabilities

Parameter	Type	Description
add	Array of String	Grant the Linux permission to the running user.
drop	Array of String	Remove the Linux permission of the running user.

Table 4-787 logs

Parameter	Type	Description
log_path	String	Container log path.
rotate	String	Interval for dumping logs.
host_path	String	Mounted host path.
host_extend_path	String	Host extension path. By extending the host path, you can distinguish mounting from different containers in the same host path. • None: The extended path is not used. • PodUID: extend the host path based on the pod ID. • PodName: extend the host path based on the pod name. • PodUID/ContainerName: extend the host path based on the pod ID and container name. • PodName/ContainerName: extend the host path based on the pod name and container name.

Table 4-788 custom_metric

Parameter	Type	Description
path	String	Collection path. Example: /metrics.
port	Integer	Collection port. Example: 9090.
dimensions	String	Monitoring dimension. Example: cpu_usage,mem_usage.

Table 4-789 component_affinity

Parameter	Type	Description
kind	String	Object kind. • node • pod
condition	String	Policy type. • required: The requirement must be met. • preferred: The requirement is met as much as possible.
match_expressions	Array of objects	Scheduling policy rule. See Table 4-790 .
weight	String	Weight. • The value is 100 when condition is set to required. • Enter an integer ranging from 1 to 100 when condition is set to preferred.

Table 4-790 match_expressions

Parameter	Type	Description
key	String	Condition key.
value	String	Condition value.
operation	String	Operation.

Table 4-791 component_probe

Parameter	Type	Description
type	String	Type. Value: http, tcp, or command.

Parameter	Type	Description
delay	Integer	Interval between the startup and detection.
timeout	Integer	Detection timeout interval.
scheme	String	Request type. Value: HTTP or HTTPS. This parameter is valid only when type is set to http .
host	String	Custom IP address. If this parameter is left blank, the pod IP address is used. This parameter is valid only when type is set to http .
port	Integer	Port number. This parameter is valid only when type is set to http or tcp .
path	String	Request path. This parameter is valid only when type is set to http .
command	Array of String	Command list. This parameter is valid only when type is set to command .

Table 4-792 refer_resources

Parameter	Type	Description
id	String	Resource ID.
type	String	Resource type. • vpc • eip • elb • cce • ecs • dcs • rds • cse: ServiceComb engine. • nacos: registry/configuration center. • customServer • pvc • gaussdb • dms_rocketmq • css_es: Elasticsearch
parameters	Object	Resource parameters. See Table 4-793 .

Table 4-793 refer_resource_parameter

Parameter	Type	Description
namespace	String	Namespace.
name	String	Resource name.
capacity	Integer	Storage capacity, in GiB.
type	String	Volume declaration type. • efs • obs • sfs • evs
class	String	Volume declaration subtype. • csi-sfsturbo: fixed when type is set to efs. • csi-obs: fixed when type is set to obs. • csi-nas: fixed when type is set to sfs. • csi-disk: fixed when type is set to evs.
obs_volume_type	String	OBS volume type. Default value: STANDARD.
access_mode	String	Storage volume access mode. • ReadWriteOnce: A volume can be mounted as read-write by a single node. • ReadWriteMany: A volume can be mounted as read-write by many nodes.
availableZone	String	AZ.
volume_type	String	Volume type.

Table 4-794 envs

Parameter	Type	Description
name	String	Variable name.
value	String	Variable value.
value_from	Object	Variable reference. See Table 4-795 .

Table 4-795 value_from

Parameter	Type	Description
reference_type	String	Reference type. • configMapKey • secretKey
name	String	Name of the configuration item or secret.
key	String	Key of the configuration item or secret.
optional	Boolean	Whether the key of the configuration item or secret is mandatory.

Table 4-796 runtime_stack

Parameter	Type	Description
name	String	Technology stack name.
type	String	Technology stack type. Value: Java, Tomcat, Nodejs, Php, Docker, or Python. • For VM-based deployment, Java, Tomcat, and Nodejs are supported. • For container-based deployment, Java, Tomcat, Nodejs, Php, and Docker are supported.
version	String	Technology stack version.
deploy_mode	String	Deployment mode. • container • virtualmachine

Table 4-797 source

Parameter	Mandatory	Type	Description
kind	Yes	String	Component source kind. • code • package • image
version	No	String	Version number.
url	No	String	Package address. This parameter is mandatory when kind is set to package or image .

Parameter	Mandatory	Type	Description
storage	No	String	Storage mode. • devcloud: CodeArts release repository • obs • http: custom file address • swr This parameter is mandatory when kind is set to package or image .
http_username	No	String	Username. This parameter is mandatory when storage is set to http . Username and password authentication is supported.
http_password	No	String	Password. This parameter is mandatory when storage is set to http . Username and password authentication is supported.
http_command	No	String	Address of the custom software package. This parameter is mandatory when storage is set to http .
http_headers	No	Map<String, String>	This parameter is mandatory when storage is set to http . You can customize request headers as key-value pairs, for example, {"key":"xx","value":"xx"}.
auth	No	String	Authentication mode. Value: iam or none. Default value: iam.
repo_auth	No	String	Authorization name, which can be obtained from the authorization list. For details, see Obtaining a Git Repository Authorization List .
repo_namespace	No	String	Namespace of the code repository.
repo_ref	No	String	Code branch or tag. Default value: master.
repo_type	No	String	Code repository type. Value: GitHub, BitBucket, or GitLab.
web_url	No	String	Code redirection URL. Example: https://github.com/example/demo.git.
repo_url	No	String	Code repository URL. Example: https://github.com/example/demo.git.

Parameter	Mandatory	Type	Description
commit_id	No	String	In the code repository, commit ID is the unique identifier of each commit and is used to identify and reference a commit. The value contains 8 to 40 characters and starts with a lowercase letter or digit.

Table 4-798 build

Parameter	Type	Description
parameters	Map<String, Object>	This parameter is provided only when no ID is available during build creation. See Table 4-799 .

Table 4-799 parameters

Parameter	Type	Description
build_cmd	String	Compilation command. By default: 1. When build.sh exists in the root directory, the command is ./build.sh . 2. When build.sh does not exist in the root directory, the command varies depending on the OS. Example: • Java and Tomcat: mvn clean package • Nodejs: npm build
dockerfile_path	String	Address of the Docker file. By default, the Docker file is in the root directory (./).
artifact_name_space	String	Build archive organization. Default value: cas_{project_id}.
cluster_id	String	ID of the cluster to be built.
node_label_selector	Map<String, String>	key: label key. value: label value.
environment_id	String	Environment ID.

Table 4-800 external_accesses

Parameter	Type	Description
protocol	String	External access type. Value: http or https.

Parameter	Type	Description
address	String	External access address.
forward_port	Integer	External access port.

Table 4-801 tolerations

Parameter	Type	Description
key	String	Taint key.
operator	String	Operator. • Equal: matches the nodes with the specified taint key (mandatory) and value. If the taint value is left blank, all taints with the same key will be matched. • Exists: matches nodes with the specified taint key. In this case, the taint value cannot be specified. Not specifying any taint key means all taints are tolerated.
value	String	Taint value. This parameter is mandatory when operator is set to Equal .
effect	String	Taint policy. • NoScheduler: Only the NoSchedule taint is matched. • PreferNoSchedule: Only the PreferNoSchedule taint is matched. • NoExecute: Only the NoExecute taint is matched. • If this parameter is not set, all taint effects are matched.
toleration_seconds	Integer	Tolerance time window. This parameter can be set only when effect is set to NoExecute . Within the tolerance time window, pods still run on the node with taints. After the time expires, the pods will be evicted.

Table 4-802 multi_containers

Parameter	Type	Description
container_name	String	Container name.

Parameter	Type	Description
source	Object	Source of the code or software package. See Table 4-797 . In multi-container mode, only images are supported.
limit_cpu	Number	Maximum CPU limit. Unit: core.
limit_memory	Number	Maximum memory size. Unit: GiB.
request_cpu	Number	Requested CPU resources. Unit: core.
request_memory	Number	Requested memory. Unit: GiB.
envs	Array of objects	Environment variable list. See Table 4-794 .
liveness_probe	Object	Container liveness probe. See Table 4-791 .
readiness_probe	Object	Container service probe. See Table 4-791 .
startup_probe	Object	Container startup probe. See Table 4-791 .
command	Object	Startup command. See Table 4-778 .
post_start	Object	Post-start processing. See Table 4-779 .
pre_stop	Object	Pre-stop processing. See Table 4-779 .
logs	Array of objects	Log collection. See Table 4-787 .
storages	Array of objects	Storage. See Table 4-771 .
security_context	Object	Security policy. See Table 4-785 .
config_encryption	Object	This parameter is mandatory when the encryption key is configured for a multi-container component. See Table 4-807 .

Table 4-803 ports

Parameter	Type	Description
name	String	Port name.
port	String	Access port.
target_port	String	Container port.

Table 4-804 update_strategy

Parameter	Type	Description
type	String	Component upgrade policy. Options: - RollingUpdate (default): New pods are created gradually, and then old pods are deleted. - Recreate: Old pods are deleted, and then new pods are created.
max_unavailable	- String - Integer	Maximum number of unavailable pods. That is, the number or percentage of pods that can be deleted based on the value of replica . The default value is 25% . This parameter can be set when workload_kind is set to deployment . - String (%): the maximum percentage of unavailable pods to the total number of pods. - Integer: the maximum number of unavailable pods.
max_surge	- String - Integer	Maximum surge. That is, the number or percentage of pods allowed based on the value of replica . The default value is 25% . This parameter can be set only when workload_kind is set to deployment . - String (%) - Integer
termination_period_seconds	Integer	Scale-in time window. That is, the time window for executing commands before a workload stops. The default value is 30s.
min_ready_seconds	Integer	Minimum ready time of a pod. That is, the minimum ready time of a new pod without any container crash. A pod is considered available only after the minimum ready time is exceeded. The default value is 0 .
progress_deadline_seconds	Integer	Maximum upgrade duration. That is, the time (in seconds) the system waits for before reporting a component upgrade failure. If this parameter is specified, the value of this parameter must be greater than that of min_ready_seconds .

Parameter	Type	Description
revision_history_limit	Integer	Maximum number of pods to retain. That is, the number of old pods to retain to allow rollback. The default value is 10 .

Table 4-805 configs

Parameter	Type	Description
conifg_id	String	Unified configuration ID.
mount_path	String	Mount path.

Table 4-806 auto_lts_config

Parameter	Type	Description
enable	Boolean	Whether to enable automatic LTS configuration.
lts_log_path	Array of String	Access configuration log path.

Table 4-807 config_encryption

Parameter	Type	Description
key_id	String	User key ID in DEW.

Example Request

None

Example Response

```
{
  "id": "4105b37a-947e-4aca-90d8-cec7c04c6f38",
  "enterprise_project_id": "0",
  "name": "release-gn7mre11",
  "enterprise_project": "default",
  "model_version": 2,
  "creator": "zhangyuhao-3",
  "status": null,
  "revisor": "zhangyuhao-3",
  "create_at": 1739865312582,
  "update_at": 1739865967776,
  "fail_mode": "stop",
  "strategy_type": 0,
  "description": "test",
  "pipeline_declaration": {
    "stages": [
```

```
{
  "id": 15,
  "name": "75foki",
  "approval_type": 0,
  "display_name": "Stage"
  "type": null,
  "pipeline_id": null,
  "order_index": 0,
  "status": null,
  "task_group": [
    "0"
  ],
  "tasks": [
    {
      "id": 17,
      "name": "bkdi55",
      "display_name": "Create application-187",
      "task_type_id": {
        "category": "Application",
        "owner": null,
        "provider": null,
        "type": "upsert"
      },
      "input": {
        "displayName": "Create application-187",
        "originalName": "bkdi55",
        "name": "xxz",
        "description": "",
        "enterprise_project_id": "0",
        "checked": true
      },
      "output": {},
      "status": null,
      "order_index": 0,
      "skipped": false,
      "group_index": "0",
      "delay_time": 0,
      "error_details": null,
      "stage_name": "75foki"
    }
  ]
},
"config_mode": "ui"
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.7.2.6 Creating a Trigger Policy Based on the Release Task ID

Function

This API is used to create a release task trigger policy based on the release task ID.

URI

POST /v3/{project_id}/pipeline/pipelines/{pipeline_id}/hooks

Table 4-808 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
pipeline_id	Yes	String	Release task ID. See Obtaining the Release Task List .

Request

Table 4-809 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Table 4-810 Request body parameters

Parameter	Mandatory	Type	Description
selected_tasks	No	List of object	Task to be executed. If this parameter is not transferred or is set to null , all tasks are executed. See Table 4-811 .

Parameter	Mandatory	Type	Description
strategy_type	Yes	Integer	Trigger policy. • 0: manual trigger • 1: scheduled trigger
hooks	No	List of object	Scheduling policy. This parameter is mandatory when strategy_type is set to 1. See Table 4-812 .

Table 4-811 selected_tasks

Parameter	Mandatory	Type	Description
stage_name	Yes	String	Stage name.
tasks	Yes	List of string	Name of the task to be executed.

Table 4-812 hooks

Parameter	Mandatory	Type	Description
schedule	Yes	String	Regular expression of the scheduling.
timezone_operator	Yes	String	Hour offset from the Coordinated Universal Time (UTC).
timezone_hour	Yes	Integer	Direction of the offset, that is, whether the time is advanced (+) or delayed (-).

Example Request

Create a scheduled trigger policy for the release task.

```
{
  "selected_tasks": [
    {
      "stage_name": "75foki",
      "tasks": [
        "bkdi55"
      ]
    }
  ],
  "strategy_type": 1,
  "hooks": [
    {
      "schedule": "0 00-59/10 00-00 ? * MON **",

```

```
    "timezone_operator": "+",
    "timezone_hour": 8
  }
]
```

Example Response

None

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.7.2.7 Modifying a Trigger Policy Based on the Release Task ID

Function

This API is used to modify a release task trigger policy based on the release task ID.

URI

PUT /v3/{project_id}/pipeline/pipelines/{pipeline_id}/hooks

Table 4-813 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
pipeline_id	Yes	String	Release task ID. See Obtaining the Release Task List .

Request

Table 4-814 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Table 4-815 Request body parameters

Parameter	Mandatory	Type	Description
selected_tasks	No	List of object	Task to be executed. If this parameter is not transferred or is set to null , all tasks are executed. See Table 4-816 .
strategy_type	Yes	Integer	Execution policy. • 0: manual trigger • 1: scheduled trigger
hooks	No	List of object	Scheduling policy. This parameter is mandatory when strategy_type is set to 1 . See Table 4-817 .

Table 4-816 selected_tasks

Parameter	Mandatory	Type	Description
stage_name	Yes	String	Stage name.
tasks	Yes	List of string	Name of the task to be executed.

Table 4-817 hooks

Parameter	Mandatory	Type	Description
schedule	Yes	String	Regular expression of the scheduling.
timezone_operator	Yes	String	Hour offset from the Coordinated Universal Time (UTC).
timezone_hour	Yes	Integer	Direction of the offset, that is, whether the time is advanced (+) or delayed (-).

Example Request

Modify a scheduled trigger policy for the release task.

```
{
  "selected_tasks": [
    {
      "stage_name": "75foki",
      "tasks": [
        "bkdi55"
      ]
    }
  ],
  "strategy_type": 1,
  "hooks": [
    {
      "schedule": "0 00-59/10 00-00 ? * MON **",
      "timezone_operator": "+",
      "timezone_hour": 8
    }
  ]
}
```

Example Response

None

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.7.2.8 Querying a Trigger Policy Based on the Release Task ID

Function

This API is used to query a release task trigger policy based on the release task ID.

URI

GET /v3/{project_id}/pipeline/pipelines/{pipeline_id}/hooks

Table 4-818 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
pipeline_id	Yes	String	Release task ID. See Obtaining the Release Task List .

Request

Table 4-819 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

Table 4-820 Response

Parameter	Type	Description
strategy_type	Integer	Execution policy. 0: manual trigger 1: scheduled trigger

Parameter	Type	Description
hooks	List of object	Scheduling policy. This parameter is mandatory when strategy_type is set to 1 . See Table 4-821 .

Table 4-821 hooks

Parameter	Type	Description
schedule	String	Regular expression of the scheduling.
timezone_operator	String	Hour offset from the Coordinated Universal Time (UTC).
timezone_hour	Integer	Direction of the offset, that is, whether the time is advanced (+) or delayed (-).

Example Request

Query the release task trigger policy.

```
{
  "selected_tasks": [
    {
      "stage_name": "75foki",
      "tasks": [
        "bkdi55"
      ]
    }
  ],
  "strategy_type": 1,
  "hooks": [
    {
      "schedule": "0 00-59/10 00-00 ? * MON **",
      "timezone_operator": "+",
      "timezone_hour": 8
    }
  ]
}
```

Example Response

None

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found

Status Code	Description
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.7.2.9 Parsing a Release Task Template Package

Function

This API is used to parse a release task template package.

URI

GET /v3/{project_id}/pipeline/pipelines/parse-template-package

Table 4-822 Path parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .

Request

Table 4-823 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Table 4-824 Request body parameter

Parameter	Man dator y	Type	Description
sources	Yes	Array of objects	Source of the component template package. See Table 4-825 .

Table 4-825 sources

Parameter	Man dator y	Type	Description
input_values	No	Map<String, String>	Template package input information.
template_source	Yes	Object	Source of the component template package. See Table 4-826 .

Table 4-826 template_source

Parameter	Man dator y	Type	Description
storage	Yes	String	Storage mode of the component template package. • devcloud: CodeArts release repository • obs • http: custom file address
url	Yes	String	Address of the component template package. This parameter is valid only when the component template package storage mode is http .
http_username	No	String	Username. This parameter is valid only when the component template package storage mode is http and the security authentication mode is username and password authentication.

Parameter	Mandatory	Type	Description
http_password	No	String	Password. This parameter is valid only when the component template package storage mode is http and the security authentication mode is username and password authentication.
http_headers	No	Array of objects	Request header. See Table 4-827 . This parameter is valid only when the component template package storage mode is http and the security authentication mode is custom header authentication.

Table 4-827 http_headers

Parameter	Mandatory	Type	Description
key	Yes	String	Request key.
value	Yes	String	Request value.

Response

Table 4-828 Response parameter

Parameter	Type	Description
template_contents	Array of objects	Template package parsing content.

Table 4-829 template_contents

Parameter	Type	Description
template	Map<String, String>	Template package file information.

Parameter	Type	Description
variables	Map<String, Object>	Template package variable declaration information. • key • value: see Table 4-830.
input_values	Map<String, String>	Template package input information.
package_name	String	Package name.

Table 4-830 variables

Parameter	Type	Description
type	String	Variable type.
default	String	Default variable value.
description	String	Variable description.

Example Request

Parse the **demo-app-package.zip** release task template package stored in the **obs://apitest/** directory of the OBS bucket.

```
{
  "sources": [
    {
      "template_source": {
        "storage": "obs",
        "url": "obs://apitest/demo-app-package.zip",
        "http_username": "",
        "http_password": "",
        "http_headers": []
      },
      "input_values": {}
    }
  ],
  "pipeline_id": null
}
```

Example Response

```
{
  "template_contents": [
    {
      "template": {
        "comp1/spec.yaml": "model_version: v1.0.0\n\ncomponent:\n  component_dir: comp_demo/\n\nk8s_resources:\n  order_type: CUSTOM\n  custom_orders:\n    - ServiceAccount\n    - Role\n    - RoleBinding\n    - ConfigMap\n    - Secret\n    - Deployment\n    - HorizontalPodAutoscaler\n    - Service\n    - Ingress",
        "comp2/variables.yaml": "variable:\n  namespace:\n    type: string\n    default: demo\n  description:\n  dep_name:\n    type: string\n    default: demo-dep\n  description: workload\n  name:\n  svc_port:\n    type: integer\n    default: 8080\n  description: service port number\n  value:\n"
```



```
type: string\n default: World\n description: Hello interface response data\n cce_resource:\n type: cce\n\n description: CCE resource\n image:\n type: string\n description: workload image",\n  "values.yaml": "namespace: demo\ndep_name: demo-dep\nsvc_port: 8080\nvalue: World\n\nimage: swr.ap-southeast-1.myhuaweicloud.com/ss/demo-app:latest",\n  "comp3/comp_demo/ss-component.yaml": "kind: ServiceStageComponent\nspec:\n name: zyh-component-3\n environment_name: default/env-ayl\n refer_resources:\n - type: cce\n id: c90a1197-c26c-11ef-8078-0255ac10009a",\n  "comp2/comp_demo/deployment.yaml": "apiVersion: apps/v1\nkind: Deployment\nmetadata:\n name: zyh-test-dep-2\n namespace: default\nspec:\n selector:\n matchLabels: {}\n template:\n metadata:\n labels: {}\n spec:\n containers:\n - name: container-1\n image: swr.ap-southeast-1.myhuaweicloud.com/ss/test3:v1\n imagePullPolicy: IfNotPresent\n resources:\n requests:\n cpu: 100m\n memory: 256Mi\n limits:\n cpu: 100m\n memory: 256Mi\n imagePullSecrets:\n - name: default-secret\n terminationGracePeriodSeconds: 30\n dnsPolicy: ClusterFirst\n replicas: 2\n revisionHistoryLimit: 10\n strategy:\n type: RollingUpdate\n rollingUpdate:\n maxUnavailable: 25%\n maxSurge: 25%\n progressDeadlineSeconds: 600\n",\n  "comp1/comp_demo/ss-component.yaml": "kind: ServiceStageComponent\nspec:\n name: zyh-component-1\n environment_name: default/env-ayl\n refer_resources:\n - type: cce\n id: c90a1197-c26c-11ef-8078-0255ac10009a",\n  "comp1/variables.yaml": "variable:\n namespace:\n type: string\n default: demo\n\n description: namespace\n dep_name:\n type: string\n default: demo-dep\n description: workload name\n svc_port:\n type: integer\n default: 8080\n description: service port number\n value:\n type: string\n default: World\n description: Hello interface response data\n cce_resource:\n type: cce\n\n description: CCE resource\n image:\n type: string\n description: workload image",\n  "comp1/comp_demo/deployment.yaml": "apiVersion: apps/v1\nkind: Deployment\nmetadata:\n name: zyh-test-dep-1\n namespace: default\nspec:\n selector:\n matchLabels: {}\n template:\n metadata:\n labels: {}\n spec:\n containers:\n - name: container-1\n image: swr.ap-southeast-1.myhuaweicloud.com/ss/test3:v1\n imagePullPolicy: IfNotPresent\n resources:\n requests:\n cpu: 100m\n memory: 256Mi\n limits:\n cpu: 100m\n memory: 256Mi\n imagePullSecrets:\n - name: default-secret\n terminationGracePeriodSeconds: 30\n dnsPolicy: ClusterFirst\n replicas: 2\n revisionHistoryLimit: 10\n strategy:\n type: RollingUpdate\n rollingUpdate:\n maxUnavailable: 25%\n maxSurge: 25%\n progressDeadlineSeconds: 600\n",\n  "comp2/values.yaml": "namespace: demo\ndep_name: demo-dep\nsvc_port: 8080\nvalue: World\n\nimage: swr.ap-southeast-1.myhuaweicloud.com/ss/demo-app:latest",\n  "comp3/spec.yaml": "model_version: v1.0.0\n\ndependency:\n - component_name: zyh-component-2\n application_name: default/test-application-zyh\n - component_name: zyh-component-1\n application_name: default/test-application-zyh\n\ncomponent:\n component_dir: comp_demo\n\nnk8s_resources:\n order_type: CUSTOM\n custom_orders:\n - ServiceAccount\n - Role\n - RoleBinding\n - ConfigMap\n - Secret\n - Deployment\n - HorizontalPodAutoscaler\n - Service\n - Ingress",\n  "comp1/values.yaml": "namespace: demo\ndep_name: demo-dep\nsvc_port: 8080\nvalue: World\n\nimage: swr.ap-southeast-1.myhuaweicloud.com/ss/demo-app:latest",\n  "comp3/variables.yaml": "variable:\n namespace:\n type: string\n default: demo\n\n description: namespace\n dep_name:\n type: string\n default: demo-dep\n description: workload name\n svc_port:\n type: integer\n default: 8080\n description: service port number\n value:\n type: string\n default: World\n description: Hello interface response data\n cce_resource:\n type: cce\n\n description: CCE resource\n image:\n type: string\n description: workload image",\n  "comp2/spec.yaml": "model_version: v1.0.0\n\nmodel_type: Application",\n  "comp3/comp_demo/deployment.yaml": "apiVersion: apps/v1\nkind: Deployment\nmetadata:\n name: zyh-test-dep-3\n namespace: default\nspec:\n selector:\n matchLabels: {}\n template:\n metadata:\n labels: {}\n spec:\n containers:\n - name: container-1\n image: swr.ap-southeast-1.myhuaweicloud.com/ss/test3:v1\n imagePullPolicy: IfNotPresent\n resources:\n requests:\n cpu: 100m\n memory: 256Mi\n limits:\n cpu: 100m\n memory: 256Mi\n imagePullSecrets:\n - name: default-secret\n terminationGracePeriodSeconds: 30\n dnsPolicy: ClusterFirst\n replicas: 2\n revisionHistoryLimit: 10\n strategy:\n type: RollingUpdate\n rollingUpdate:\n maxUnavailable: 25%\n maxSurge: 25%\n progressDeadlineSeconds: 600\n",\n  "comp3/values.yaml": "namespace: demo\ndep_name: demo-dep\nsvc_port: 8080\nvalue: World\n\nimage: swr.ap-southeast-1.myhuaweicloud.com/ss/demo-app:latest",
```

```
"application.yaml": {"kind": "ServiceStageApplication\nspec:\n  name: test-application-zyh\n  enterprise_project_id: 0",
"comp2/comp_demo/ss-component.yaml": {"kind": "ServiceStageComponent\nspec:\n  name: zyh-component-2\n  environment_name: default/env-ayl\n  refer_resources:\n    - type: cce\n    id: c90a1197-c26c-11ef-8078-0255ac10009a"
},
"variable": {
  "comp2/dep_name": {
    "parsed_type": {
      "type": "string",
      "properties": null,
      "item_type": null
    },
    "type": "string",
    "sensitive": false,
    "default": "demo-dep",
    "description": "workload name"
  },
  "comp1/image": {
    "parsed_type": {
      "type": "string",
      "properties": null,
      "item_type": null
    },
    "type": "string",
    "sensitive": false,
    "default": null,
    "description": "workload image"
  },
  "image": {
    "parsed_type": {
      "type": "string",
      "properties": null,
      "item_type": null
    },
    "type": "string",
    "sensitive": false,
    "default": null,
    "description": "workload image"
  },
  "comp2/value": {
    "parsed_type": {
      "type": "string",
      "properties": null,
      "item_type": null
    },
    "type": "string",
    "sensitive": false,
    "default": "World",
    "description": "Hello interface response data"
  },
  "comp3/cce_resource": {
    "parsed_type": {
      "type": "cce",
      "properties": null,
      "item_type": null
    },
    "type": "cce",
    "sensitive": false,
    "default": null,
    "description": "CCE resource"
  },
  "comp1/namespace": {
    "parsed_type": {
      "type": "string",
      "properties": null,
      "item_type": null
    },
    "type": "string",
```

```
"sensitive": false,
"default": "demo",
"description": "namespace"
},
"comp1/value": {
  "parsed_type": {
    "type": "string",
    "properties": null,
    "item_type": null
  },
  "type": "string",
  "sensitive": false,
  "default": "World",
  "description": "Hello interface response data"
},
"comp3/namespace": {
  "parsed_type": {
    "type": "string",
    "properties": null,
    "item_type": null
  },
  "type": "string",
  "sensitive": false,
  "default": "demo",
  "description": "namespace"
},
"svc_port": {
  "parsed_type": {
    "type": "integer",
    "properties": null,
    "item_type": null
  },
  "type": "integer",
  "sensitive": false,
  "default": 8080,
  "description": "service port number"
},
"cce_resource": {
  "parsed_type": {
    "type": "cce",
    "properties": null,
    "item_type": null
  },
  "type": "cce",
  "sensitive": false,
  "default": null,
  "description": "CCE resource"
},
"comp1/svc_port": {
  "parsed_type": {
    "type": "integer",
    "properties": null,
    "item_type": null
  },
  "type": "integer",
  "sensitive": false,
  "default": 8080,
  "description": "service port number"
},
"comp3/value": {
  "parsed_type": {
    "type": "string",
    "properties": null,
    "item_type": null
  },
  "type": "string",
  "sensitive": false,
  "default": "World",
  "description": "Hello interface response data"
```

```
},
"comp1/cce_resource": {
  "parsed_type": {
    "type": "cce",
    "properties": null,
    "item_type": null
  },
  "type": "cce",
  "sensitive": false,
  "default": null,
  "description": "CCE resource"
},
"comp3/image": {
  "parsed_type": {
    "type": "string",
    "properties": null,
    "item_type": null
  },
  "type": "string",
  "sensitive": false,
  "default": null,
  "description": "workload image"
},
"dep_name": {
  "parsed_type": {
    "type": "string",
    "properties": null,
    "item_type": null
  },
  "type": "string",
  "sensitive": false,
  "default": "demo-dep",
  "description": "workload name"
},
"comp2/cce_resource": {
  "parsed_type": {
    "type": "cce",
    "properties": null,
    "item_type": null
  },
  "type": "cce",
  "sensitive": false,
  "default": null,
  "description": "CCE resource"
},
"comp1/dep_name": {
  "parsed_type": {
    "type": "string",
    "properties": null,
    "item_type": null
  },
  "type": "string",
  "sensitive": false,
  "default": "demo-dep",
  "description": "workload name"
},
"comp3/dep_name": {
  "parsed_type": {
    "type": "string",
    "properties": null,
    "item_type": null
  },
  "type": "string",
  "sensitive": false,
  "default": "demo-dep",
  "description": "workload name"
},
"comp3/svc_port": {
  "parsed_type": {
```

```
        "type": "integer",
        "properties": null,
        "item_type": null
    },
    "type": "integer",
    "sensitive": false,
    "default": 8080,
    "description": "service port number"
},
"comp2/svc_port": {
    "parsed_type": {
        "type": "integer",
        "properties": null,
        "item_type": null
    },
    "type": "integer",
    "sensitive": false,
    "default": 8080,
    "description": "service port number"
},
"namespace": {
    "parsed_type": {
        "type": "string",
        "properties": null,
        "item_type": null
    },
    "type": "string",
    "sensitive": false,
    "default": "demo",
    "description": "namespace"
},
"comp2/image": {
    "parsed_type": {
        "type": "string",
        "properties": null,
        "item_type": null
    },
    "type": "string",
    "sensitive": false,
    "default": null,
    "description": "workload image"
},
"value": {
    "parsed_type": {
        "type": "string",
        "properties": null,
        "item_type": null
    },
    "type": "string",
    "sensitive": false,
    "default": "World",
    "description": "Hello interface response data"
},
"comp2/namespace": {
    "parsed_type": {
        "type": "string",
        "properties": null,
        "item_type": null
    },
    "type": "string",
    "sensitive": false,
    "default": "demo",
    "description": "namespace"
}
},
"input_values": {
    "comp2/dep_name": "demo-dep",
    "comp1/image": "swr.ap-southeast-1.myhuaweicloud.com/ss/demo-app:latest",
    "image": "swr.ap-southeast-1.myhuaweicloud.com/ss/demo-app:latest",
```

```
    "comp2/value": "World",
    "comp3/cce_resource": null,
    "comp1/namespace": "demo",
    "comp1/value": "World",
    "comp3/namespace": "demo",
    "svc_port": 8080,
    "cce_resource": null,
    "comp1/svc_port": 8080,
    "comp3/value": "World",
    "comp1/cce_resource": null,
    "comp3/image": "swr.ap-southeast-1.myhuaweicloud.com/ss/demo-app:latest",
    "dep_name": "demo-dep",
    "comp2/cce_resource": null,
    "comp1/dep_name": "demo-dep",
    "comp3/dep_name": "demo-dep",
    "comp3/svc_port": 8080,
    "comp2/svc_port": 8080,
    "namespace": "demo",
    "comp2/image": "swr.ap-southeast-1.myhuaweicloud.com/ss/demo-app:latest",
    "value": "World",
    "comp2/namespace": "demo"
  },
  "package_name": "demo-app-package.zip"
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.8 System Configuration

4.8.1 Technology Stack

4.8.1.1 Creating a Technology Stack

Function

This API is used to create a custom technology stack.

URI

POST /v3/{project_id}/cas/runtimestacks

Table 4-831 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .

Request

Table 4-832 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Table 4-833 Request body parameter

Parameter	Mandatory	Type	Description
name	Yes	String	Technology stack name.
deploy_mode	Yes	String	Deployment mode. - container - virtualmachine
type	Yes	String	Technology stack type. - If deploy_mode is set to container, Java and Tomcat are supported. - If deploy_mode is set to virtualmachine, Java and Tomcat are supported.
version	Yes	String	Version number.
spec	No	Object	Technology stack parameters. See Table 4-834 .
description	No	String	Technology stack description.

Table 4-834 spec

Parameter	Mandatory	Type	Description
sdk	No	String	Tool package of the technology stack.
os	No	String	OS of the technology stack.
parameters	No	Map<String, String>	Parameters of the image related to the technology stack. Key and value description: - os_url: OS image path, which is used as the basic image of the technology stack of the container-based deployment type. Example value: swr.ap-southeast-1.myhuaweicloud.com/cyf/euleros_x86_64:2.9.5 - os_type: OS image type of the technology stack of the container-based deployment type. The value can be system or custom. - jdk_url: path for storing the JDK software package. Example: obs://cyf/bisheng-jdk-8u412-linux-aarch64.tar.gz. - tomcat_url: path for storing the Tomcat software package. Example value: obs://cyf/apache-tomcat-8.5.100.tar.gz.

Response

Table 4-835 Response parameters

Parameter	Type	Description
id	String	Technology stack ID.
name	String	Runtime system name.
project_id	String	Tenant's project ID.
deploy_mode	String	Deployment mode. - container - virtualmachine
version	String	Version number.

Parameter	Type	Description
type	String	Technology stack type. • If deploy_mode is set to container, Java and Tomcat are supported. • If deploy_mode is set to virtualmachine, Java and Tomcat are supported.
version	String	Technology stack version.
spec	Object	Technology stack parameters. See Table 4-836 .
description	String	Technology stack description.
system	Integer	Whether the stack is a system technology stack. • 1: system technology stack. • 0: custom technology stack.
status	String	Technology stack status. • Supported • Deprecated • Disable
creator	String	Creator.
create_time	Integer	Creation time.
update_time	Integer	Update time.
component_count	Integer	Number of components associated with the technology stack.

Table 4-836 spec

Parameter	Type	Description
os	String	OS of the technology stack.
sdk	String	Tool package of the technology stack.
parameters	Map<String,String>	Address of the image related to the technology stack.

Example Request

Create a custom Java technology stack version of the VM-based deployment type.

```
{
  "name": "test",
  "deploy_mode": "virtualmachine",
  "type": "Java",
```

```
{
  "version": "1.0.0",
  "description": "1234",
  "spec": {
    "parameters": {
      "jdk_url": "obs://cyf/bisheng-jdk-8u412-linux-aarch64.tar.gz"
    }
  }
}
```

Example Response

```
{
  "id": "cf4727f0-0d6f-43d5-b1b8-f2c1f09d1611",
  "name": "test",
  "project_id": "578ac30b81034b89a7255b3af26db9c9",
  "deploy_mode": "virtualmachine",
  "type": "Java",
  "version": "1.0.0",
  "description": null,
  "spec": {
    "os": null,
    "sdk": null,
    "parameters": {
      "jdk_url": "obs://cyf/bisheng-jdk-8u412-linux-aarch64.tar.gz"
    }
  },
  "system": 0,
  "status": "Disable",
  "creator": "cyf",
  "create_time": 1722934506431,
  "update_time": 1722934506431
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
409	Conflict
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.8.1.2 Querying a Technology Stack

Function

This API is used to query technology stack information.

URI

GET /v3/{project_id}/cas/runtimestacks

Table 4-837 Path parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .

Request

Table 4-838 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

Table 4-839 Response parameters

Parameter	Type	Description
runtime_stacks	Array of objects	Technology stack. See Table 4-840 .
total	Integer	Total number of technology stacks.

Table 4-840 runtimestack

Parameter	Type	Description
id	String	Technology stack ID.
name	String	Runtime system name.
project_id	String	Tenant's project ID.

Parameter	Type	Description
deploy_mode	String	Deployment mode. • container • virtualmachine
type	String	Technology stack type. • If deploy_mode is set to container, Java and Tomcat are supported. • If deploy_mode is set to virtualmachine, Java and Tomcat are supported.
version	String	Technology stack version.
spec	Object	Technology stack parameters. See Table 4-841 .
creator	String	Creator.
system	Integer	Whether the stack is a system technology stack. • 1: system technology stack. • 0: custom technology stack.
status	String	Technology stack status. • Supported • Deprecated • Disable
create_time	Integer	Creation time.
update_time	Integer	Update time.
component_count	Integer	Number of components associated with the technology stack.

Table 4-841 spec

Parameter	Type	Description
os	String	OS of the technology stack.
sdk	String	Tool package of the technology stack.
parameters	Map<String,String>	Address of the image related to the technology stack.

Example Request

None

Example Response

```
{
  "runtime_stacks": [{
    "id": "1c5e7000-f2f2-4fc2-988f-0a22ca67940f",
    "name": "OpenJDK11",
    "project_id": "578ac30b81034b89a7255b3af26db9c9",
    "deploy_mode": "virtualmachine",
    "type": "Java",
    "version": "1.3.7",
    "spec": {
      "os": "Linux",
      "sdk": "OpenJDK-11.0.23",
      "parameters": null
    },
    "system": 1,
    "status": "Supported",
    "create_time": 1722416061588,
    "update_time": 1722416098865,
    "component_count": 3
  },
  {
    "id": "b153f6b8-9335-46a1-913e-c2d8f966d4b2",
    "name": "CustomJdk",
    "project_id": "578ac30b81034b89a7255b3af26db9c9",
    "deploy_mode": "virtualmachine",
    "type": "Java",
    "version": "1.0.0",
    "spec": {
      "os": null,
      "sdk": null,
      "parameters": {
        "jdk_url": "obs://cyf/bisheng-jdk-8u412-linux-aarch64.tar.gz"
      }
    },
    "system": 0,
    "status": "Disable",
    "creator": "cyf",
    "create_time": 1722932102641,
    "update_time": 1722932102641,
    "component_count": 0
  }
],
  "total": 2
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
409	Conflict
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.8.1.3 Querying a Built-in Image

Function

This API is used to obtain the built-in basic images used for building ServiceStage technology stacks.

URI

GET /v3/{project_id}/cas/innerimages

Table 4-842 Path parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .

Request

Table 4-843 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

Table 4-844 Response parameter

Parameter	Type	Description
runtime_stacks	Array of objects	Runtime system stack. See Table 4-845 .

Table 4-845 runtime_stacks

Parameter	Type	Description
type	String	Technology stack type. • Nodejs • Java • Tomcat • Python • Php
url	String	Image address.

Example Request

None

Example Response

```
{
  "runtime_stacks": [{
    "url": "swr.ap-southeast-1.myhuaweicloud.com/op_svc_cse/openjdk-x86_64:17-1.3.6",
    "type": "Java"
  }]
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
409	Conflict
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.8.1.4 Querying a Technology Stack Based on the Technology Stack ID

Function

This API is used to query a technology stack based on the technology stack ID.

URI

GET /v3/{project_id}/cas/runtimestacks/{runtimestack_id}

Table 4-846 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
runtimestack_id	Yes	String	Technology stack ID. See Querying a Technology Stack .

Request

Table 4-847 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

Table 4-848 Response parameters

Parameter	Type	Description
id	String	Technology stack ID.
name	String	Runtime system name.
project_id	String	Tenant's project ID.
deploy_mode	String	Deployment mode. • container • virtualmachine
version	String	Version number.

Parameter	Type	Description
type	String	Technology stack type. - If deploy_mode is set to container, Java and Tomcat are supported. - If deploy_mode is set to virtualmachine, Java and Tomcat are supported.
version	String	Technology stack version.
spec	Object	Technology stack parameters. See Table 4-849 .
description	String	Description.
system	Integer	Whether the stack is a system technology stack. 1: system technology stack. 0: custom technology stack.
status	String	Technology stack status. - Supported - Deprecated - Disable
creator	String	Creator.
create_time	Integer	Creation time.
update_time	Integer	Update time.
component_count	Integer	Number of components associated with the technology stack.

Table 4-849 spec

Parameter	Type	Description
os	String	OS of the technology stack.
sdk	String	Tool package of the technology stack.
parameters	Map<String,String>	Address of the image related to the technology stack.

Example Request

None

Example Response

```
{
  "id": "cf4727f0-0d6f-43d5-b1b8-f2c1f09d1611",
}
```

```
{
  "name": "test",
  "project_id": "578ac30b81034b89a7255b3af26db9c9",
  "deploy_mode": "virtualmachine",
  "type": "Java",
  "version": "1.0.0",
  "description": null,
  "spec": {
    "os": null,
    "sdk": null,
    "parameters": {
      "jdk_url": "obs://cyf/bisheng-jdk-8u412-linux-aarch64.tar.gz"
    }
  },
  "system": 0,
  "status": "Disable",
  "creator": "cyf",
  "create_time": 1722934506431,
  "update_time": 1722934506431
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
409	Conflict
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.8.1.5 Modifying a Technology Stack Based on the Technology Stack ID

Function

This API is used to modify a technology stack based on the technology stack ID.

URI

PUT /v3/{project_id}/cas/runtimestacks/{runtimestack_id}

Table 4-850 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .

Parameter	Mandatory	Type	Description
runtimestack_id	Yes	String	Technology stack ID. See Querying a Technology Stack .

Request

Table 4-851 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Table 4-852 Request body parameter

Parameter	Mandatory	Type	Description
name	Yes	String	Technology stack name.
deploy_mode	Yes	String	Deployment mode. - container - virtualmachine
type	Yes	String	Technology stack type. - If deploy_mode is set to container, Java and Tomcat are supported. - If deploy_mode is set to virtualmachine, Java and Tomcat are supported.
version	Yes	String	Version number.
spec	No	Object	Technology stack parameters. See Table 4-853 .
description	No	String	Technology stack description.

Table 4-853 spec

Parameter	Mandatory	Type	Description
sdk	No	String	Tool package of the technology stack.
os	No	String	OS of the technology stack.
parameters	No	Map<String, String>	Parameters of the image related to the technology stack.

Response

Table 4-854 Response parameters

Parameter	Type	Description
id	String	Technology stack ID.
name	String	Runtime system name.
project_id	String	Tenant's project ID.
deploy_mode	String	Deployment mode. • container • virtualmachine
version	String	Version number.
type	String	Technology stack type. • If deploy_mode is set to container, Java and Tomcat are supported. • If deploy_mode is set to virtualmachine, Java and Tomcat are supported.
version	String	Technology stack version.
spec	Object	Technology stack parameters. See Table 4-855 .
description	String	Description.
system	Integer	Whether the stack is a system technology stack. • 1: system technology stack. • 0: custom technology stack.
status	String	Technology stack status. • Supported • Deprecated • Disable

Parameter	Type	Description
creator	String	Creator.
create_time	Integer	Creation time.
update_time	Integer	Update time.
component_count	Integer	Number of components associated with the technology stack.

Table 4-855 spec

Parameter	Type	Description
os	String	OS of the technology stack.
sdk	String	Tool package of the technology stack.
parameters	Map<String,String>	Address of the image related to the technology stack.

Example Request

Edit the name and description of the custom Java technology stack version of the VM-based deployment type.

```
{
  "name": "CustomJdk",
  "deploy_mode": "virtualmachine",
  "type": "Java",
  "version": "1.0.0",
  "description": "12345",
  "spec": {
    "parameters": {
      "jdk_url": "obs://cyf/bisheng-jdk-8u412-linux-aarch64.tar.gz"
    }
  }
}
```

Example Response

```
{
  "id": "cf4727f0-0d6f-43d5-b1b8-f2c1f09d1611",
  "name": "test",
  "project_id": "578ac30b81034b89a7255b3af26db9c9",
  "deploy_mode": "virtualmachine",
  "type": "Java",
  "version": "1.0.0",
  "description": null,
  "spec": {
    "os": null,
    "sdk": null,
    "parameters": {
      "jdk_url": "obs://cyf/bisheng-jdk-8u412-linux-aarch64.tar.gz"
    }
  },
  "system": 0,
}
```

```
"status": "Disable",  
"creator": "cyf",  
"create_time": 1722934506431,  
"update_time": 1722934506431  
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
409	Conflict
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.8.1.6 Deleting a Technology Stack Based on the Technology Stack ID

Function

This API is used to delete a technology stack based on the technology stack ID.

URI

DELETE /v3/{project_id}/cas/runtimestacks/{runtimestack_id}

Table 4-856 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
runtimestack_id	Yes	String	Technology stack ID. See Querying a Technology Stack .

Request

Table 4-857 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Response

None

Example Request

None

Example Response

None

Status Code

Status Code	Description
204	OK
400	Bad Request
404	Not Found
409	Conflict
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

4.8.1.7 Releasing or Canceling the Release of a Technology Stack

Function

This API is used to release or cancel the release of a technology stack.

URI

POST /v3/{project_id}/cas/runtimestacks/action

Table 4-858 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .

Request

Table 4-859 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain the token, see Authentication .

Table 4-860 Request body parameter

Parameter	Mandatory	Type	Description
action	Yes	String	Operation performed on technology stacks. - Supported: releases the technology stacks. - Disable: cancels the release of the technology stacks.
parameters	Yes	Object	See Table 4-861 .

Table 4-861 parameters

Parameter	Mandatory	Type	Description
runtimestacks	Yes	Array of objects	Technology stacks to be performed. For details about the parameter format, see Table 4-862 .

Table 4-862 runtimestack_item

Parameter	Mandatory	Type	Description
id	Yes	String	Technology stack ID.

Response

None

Example Request

Release a technology stack.

```
{
  "action": "Supported",
  "parameters": {
    "runtimestacks": [
      {
        "id": "b153f6b8-9335-46a1-913e-c2d8f966d4b2"
      }
    ]
  }
}
```

Example Response

None

Status Code

Status Code	Description
204	OK
400	Bad Request
404	Not Found
409	Conflict

Status Code	Description
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

5 Git Repository Access APIs

5.1 Obtaining a Git Repository Authorization List

Function

This API is used to obtain a Git repository authorization list.

URI

GET /v1/{project_id}/git/auths

[Table 5-1](#) describes the parameters.

Table 5-1 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .

Request

Table 5-2 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .

Response

Table 5-3 Response parameters

Parameter	Type	Description
authorizations	Array<Object>	Authorization structure. See Table 5-4 .
count	Integer	Number of authorized repositories.

Table 5-4 authorizations

Parameter	Type	Description
name	String	Authorized repository.
repo_type	String	Repository type. Value: GitHub, BitBucket, or GitLab.
repo_host	String	Repository address.
repo_home	String	Repository homepage.
repo_user	String	Repository username.
avatar	String	Repository avatar.
token_type	String	Repository authorization type.
create_time	Long	Repository authorization creation timestamp.
update_time	Long	Repository authorization update timestamp.
status	Integer	Repository authorization status. 0: normal 1: overdue 2: unauthenticated
tag	String	Type of the authorized environment. Default value: null.

Example Request

None

Example Response

```
{
  "authorizations": [
    {
      "name": "test",
      "repo_type": "github",
      "repo_host": "https://api.github.com",
      "repo_home": "https://github.com/tom-repo",
      "repo_user": "tom-repo",
      "avatar": "https://avatars.githubusercontent.com/u/73919200?v=4",
      "token_type": "oauth",
      "create_time": 1649731565642,
      "update_time": 1649731565642,
      "tag": null,
      "status": 0
    }
  ],
  "count": 1
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request

Error Code

The error code format is SVCSTG.REPO.[Error_ID], for example, SVCSTG.REPO.0401. For details, see [ServiceStage Error Codes](#).

5.2 Obtaining an Authorization Redirection URL

Function

This API is used to obtain an authorization redirection URL.

URI

GET /v1/{project_id}/git/auths/{repo_type}/redirect

[Table 5-5](#) describes the parameters.

Table 5-5 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
repo_type	Yes	String	Repository type. Value: GitHub, BitBucket, or GitLab.

Table 5-6 Query parameters

Parameter	Mandatory	Type	Description
tag	No	String	Site label. Example: ?tag=intl, which indicates an international site.

Request

Table 5-7 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .

Response

Table 5-8 Response parameter

Parameter	Type	Description
url	String	Authorization redirection URL.

Example Request

None

Example Response

```
{
  "url": "https://github.com/login/oauth/authorize?client_id=2993bccf255673ba****&redirect_uri=https%3A%2F%2Fconsole.huaweicloud.com%2Fservicestage%2Foauth%3Frepo_type%3Dgithub&state=%242a%2410%24%2FMOymg0euuGH%2FJNWIIJ3FeSVZhTUVj%2FF7r52mqpUKcYXz7f6.****&scope=read%3Auser%20repo%20write%3Arepo_hook"
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request

Error Code

The error code format is SVCSTG.REPO.[Error_ID], for example, SVCSTG.REPO.0401. For details, see [ServiceStage Error Codes](#).

5.3 Creating OAuth Authorization

Function

This API is used to create OAuth authorization.

URI

POST /v1/{project_id}/git/auths/{repo_type}/oauth

[Table 5-9](#) describes the parameters.

Table 5-9 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
repo_type	Yes	String	Repository type. Value: GitHub, BitBucket, or GitLab.

Table 5-10 Query parameters

Parameter	Mandatory	Type	Description
tag	No	String	Site label. Example: ?tag=intl, which indicates an international site.

Request

Table 5-11 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .

Table 5-12 Request body parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Authorization name.
code	Yes	String	code parameter in the redirection URL after Git repository authorization is performed.
state	Yes	String	One-off authentication code and random string received after Git repository authorization is performed.

 NOTE

To obtain the values of **code** and **state**, perform the following steps:

1. Enter the URL obtained in [Obtaining an Authorization Redirection URL](#) in the address box of the browser and press **Enter**.
2. On the login page, enter your code repository account and password.
If you have logged in to the code repository, skip this step.
3. On the login page of the Huawei Cloud console, enter the account name and password, and click **Log In**.
If you have logged in to the Huawei Cloud console, skip this step.
4. After the system displays a message indicating that the authorization fails to be created, copy the URL in the address box.

For example:

```
https://console.huaweicloud.com/servicestage/oauth?  
repo_type=github&code=04bed2df009f3fc9***c&state=%242a  
%2410%24.uQYThj5yKFs51w9s9ajeuQ0NSGshPH7IUfXoZLPlz7J%2FoAJ8B***
```

5. Decode the obtained URL to obtain the decoded URL.

For example:

```
https://console.huaweicloud.com/servicestage/oauth?  
repo_type=github&code=04bed2df009f3fc9***c&state=$2a  
$10$.uQYThj5yKFs51w9s9ajeuQ0NSGshPH7IUfXoZLPlz7J/oAJ8B***
```

Obtain the values of **code** and **state**.

The value of the obtained **state** is a one-off authentication code and random string, which can be used only once. If you need to call this API again to create OAuth authorization, follow the steps in [Obtaining an Authorization Redirection URL](#) again to obtain the URL and obtain the values of **code** and **state**.

Response

Table 5-13 Response parameters

Parameter	Type	Description
authorization	Object	Authorization structure. See Table 5-14 .

Table 5-14 authorization

Parameter	Type	Description
name	String	Authorization name.
repo_type	String	Repository type. Value: github, gitlab, , or bitbucket.
repo_host	String	Repository address.
repo_home	String	Repository homepage.
repo_user	String	Repository username.
avartar	String	Repository avatar.

Parameter	Type	Description
token_type	String	Repository authorization type.
create_time	Long	Repository authorization creation timestamp.
update_time	Long	Repository authorization update timestamp.
status	Integer	Repository authorization status. 0: normal 1: overdue 2: unauthenticated

Example Request

Create OAuth authorization **auth1**.

```
{
  "name": "auth1",
  "code": "04bed2df009f3fc9***c",
  "state": "$2a$10$.uQYThj5yKFs51w9s9ajeuQ0NSGshPH7IUfXoZLPlz7J/oAJ8B****"
}
```

Example Response

```
{
  "authorization": {
    "name": "auth1",
    "repo_type": "github",
    "repo_host": "https://api.github.com",
    "repo_home": "https://github.com/tom-repo",
    "repo_user": "repo",
    "avatar": "https://avatars.githubusercontent.com/u/73919200?v=4",
    "token_type": "oauth",
    "create_time": 1649758626858,
    "update_time": 1649758626858,
    "status": 0
  }
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request

Error Code

The error code format is SVCSTG.REPO.[Error_ID], for example, SVCSTG.REPO.0401. For details, see [ServiceStage Error Codes](#).

5.4 Creating Private Token Authorization

Function

This API is used to create private token authorization.

URI

POST /v1/{project_id}/git/auths/{repo_type}/personal

[Table 5-15](#) describes the parameters.

Table 5-15 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
repo_type	Yes	String	Repository type. Value: github, or gitlab.

Request

Table 5-16 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .

Table 5-17 Request body parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Authorization name.

Parameter	Mandatory	Type	Description
token	Yes	String	Repository token.
host	No	String	Repository address.

Response

Table 5-18 Response parameters

Parameter	Type	Description
authorization	Object	Authorization structure. See Table 5-19 .

Table 5-19 authorization

Parameter	Type	Description
name	String	Authorization name.
repo_type	String	Repository type. Value: github, or gitlab.
repo_host	String	Repository address.
repo_home	String	Repository homepage.
repo_user	String	Repository username.
avatar	String	Repository avatar.
token_type	String	Repository authorization type.
create_time	Long	Repository authorization creation timestamp.
update_time	Long	Repository authorization update timestamp.
status	Integer	Repository authorization status. 0: normal 1: overdue 2: unauthenticated

Example Request

Create private token authorization **token1**.

```
{  
  "name": "token1",
```

```
{
  "token": "ghp_7QoYcNLfahSXSbhhUT8R5xHLKe*****6",
  "host": "https://github.com/tom-repo"
}
```

Example Response

```
{
  "authorization": {
    "name": "token1",
    "repo_type": "github",
    "repo_host": "https://api.github.com",
    "repo_home": "https://github.com/tom-repo",
    "repo_user": "tom-repo",
    "avatar": "https://avatars.githubusercontent.com/u/73919264?v=4",
    "token_type": "personal",
    "create_time": 1649762172340,
    "update_time": 1649762172340,
    "status": 0
  }
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request

Error Code

The error code format is SVCSTG.REPO.[Error_ID], for example, SVCSTG.REPO.0401. For details, see [ServiceStage Error Codes](#).

5.5 Creating Password Authorization

Function

This API is used to create password authorization for a Git repository.

URI

POST /v1/{project_id}/git/auths/{repo_type}/password

[Table 5-20](#) describes the parameters.

Table 5-20 Path parameters

Parameter	Mand atory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .

Parameter	Mandatory	Type	Description
repo_type	Yes	String	Repository type. Value: bitbucket.

Request

Table 5-21 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .

Table 5-22 Request body parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Authorization name.
user	Yes	String	Repository username.
password	Yes	String	Repository password.

Response

Table 5-23 Response parameters

Parameter	Type	Description
authorization	Object	Authorization structure. See Table 5-24 .

Table 5-24 authorization

Parameter	Type	Description
name	String	Authorization name.
repo_type	String	Repository type. Value: bitbucket.
repo_host	String	Repository address.
repo_home	String	Repository homepage.
repo_user	String	Repository username.
avatar	String	Avatar.
token_type	String	Authorization mode.
create_time	Long	Creation time.
update_time	Long	Update time.
status	Integer	Repository authorization status. 0: normal 1: overdue 2: unauthenticated

Example Request

Create password authorization **pass1**.

```
{
  "name": "pass1",
  "user": "bucket-ljn",
  "password": "*****"
}
```

Example Response

```
{
  "authorization": {
    "name": "pass1",
    "repo_type": "bitbucket",
    "repo_host": "https://api.bitbucket.org",
    "repo_home": "https://bitbucket.org/%7B63cc78de-*****-9505c0335f96%7D/",
    "repo_user": "bucket-ljn",
    "avatar": "https://secure.gravatar.com/avatar/05a7b4a2302750*****cf64?d=https%3A%2F%2Favatar-management--avatars.us-west-2.prod.public.atl-paas.net%2Finitials%2FB-2.png",
    "token_type": "password",
    "create_time": 1634119075119,
    "update_time": 1634119075119,
    "status": 0
  }
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request

Error Code

The error code format is SVCSTG.REPO.[Error_ID], for example, SVCSTG.REPO.0401. For details, see [ServiceStage Error Codes](#).

5.6 Deleting Repository Authorization

Function

This API is used to delete repository authorization based on the name.

URI

DELETE /v1/{project_id}/git/auths/{name}

[Table 5-25](#) describes the parameters.

Table 5-25 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
name	Yes	String	Authorization name.

Request

Table 5-26 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .

Response

None

Example Request

None

Example Response

None

Status Code

Status Code	Description
204	OK
500	Internal Server Error

Error Code

The error code format is SVCSTG.REPO.[Error_ID], for example, SVCSTG.REPO.0401. For details, see [ServiceStage Error Codes](#).

5.7 Obtaining a Repository Namespace

Function

This API is used to obtain a repository namespace.

URI

GET /v1/{project_id}/git/repos/namespaces

[Table 5-27](#) describes the parameters.

Table 5-27 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .

Request

Table 5-28 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .
X-Repo-Auth	Yes	String	Authorization name. See Obtaining a Git Repository Authorization List .

Response

Table 5-29 Response parameters

Parameter	Type	Description
namespaces	Array<Object>	Namespace structure. See Table 5-30 .

Table 5-30 namespaces

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

Example Request

None

Example Response

```
{
  "namespaces": [
    {
      "id": "zmg",
      "name": "zmg"
    }
  ]
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request

Error Code

The error code format is SVCSTG.REPO.[Error_ID], for example, SVCSTG.REPO.0401. For details, see [ServiceStage Error Codes](#).

5.8 Obtaining Repository Information Based on the Clone URL

Function

This API is used to obtain repository information based on the clone URL.

URI

GET /v1/{project_id}/git/repos/project-info

[Table 5-31](#) describes the parameters.

Table 5-31 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .

Table 5-32 Query parameters

Parameter	Mandatory	Type	Description
clone_url	Yes	String	Clone URL of a repository. See Obtaining All Projects in a Namespace .

Request

Table 5-33 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .
X-Repo-Auth	Yes	String	Authorization name. See Obtaining a Git Repository Authorization List .

Response

Table 5-34 Response parameters

Parameter	Type	Description
namespace_id	String	Namespace ID.
namespace	String	Namespace.
project_id	String	Repository project ID.
project	String	Repository project.

Example Request

Obtain repository information based on clone URL **http://xxx.gitlab.com:8090/xxx/springcloud-xxbbon.git**.

`https://Endpoint/v1/7c6a62ddb2bf45a3920d8053d3cb8fbf/git/repos/project-info?clone_url=http://xxx.gitlab.com:8090/xxx/springcloud-xxbbon.git`

Example Response

```
{
  "project": "gradle_custom_gradledir",
  "namespace_id": "zmg",
  "namespace": "zmg",
  "project_id": "79",
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request

Error Code

The error code format is SVCSTG.REPO.[Error_ID], for example, SVCSTG.REPO.0401. For details, see [ServiceStage Error Codes](#).

5.9 Obtaining All Projects in a Namespace

Function

This API is used to obtain all projects in a namespace.

URI

GET /v1/{project_id}/git/repos/{namespace}/projects

[Table 5-35](#) describes the parameters.

Table 5-35 Path parameters

Parameter	Mandator y	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
namespace	Yes	String	Namespace ID. See Obtaining a Repository Namespace .

Request

Table 5-36 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .
X-Repo-Auth	Yes	String	Authorization name. See Obtaining a Git Repository Authorization List .

Response

Table 5-37 Response parameters

Parameter	Type	Description
projects	Array<Object>	Project structure. See Table 5-38 .

Table 5-38 projects

Parameter	Type	Description
id	String	Project ID.
name	String	Project name.
clone_url	String	Clone URL.

Example Request

None

Example Response

```
{
  "projects": [
    {
      "id": "79",
      "name": "gradle_custom_gradledir",
      "clone_url": "http://***@cpe.gitlab.com:8090/zmg/gradle_custom_gradledir.git"
    }
  ]
}
```

```
]
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request

Error Code

The error code format is SVCSTG.REPO.[Error_ID], for example, SVCSTG.REPO.0401. For details, see [ServiceStage Error Codes](#).

5.10 Creating a Software Repository Project

Function

This API is used to create a software repository project.

URI

POST /v1/{project_id}/git/repos/{namespace}/projects

[Table 5-39](#) describes the parameters.

Table 5-39 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
namespace	Yes	String	Namespace ID. See Obtaining a Repository Namespace .

Request

Table 5-40 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .
X-Repo-Auth	Yes	String	Authorization name. See Obtaining a Git Repository Authorization List .

Table 5-41 Request body parameter

Parameter	Mandatory	Type	Description
name	Yes	String	Project name.

Response

Table 5-42 Response parameters

Parameter	Type	Description
id	String	Project ID.
name	String	Project name.
clone_url	String	Clone URL.

Example Request

Create software repository project **string**.

```
{
  "name": "string"
}
```

Example Response

```
{
  "id": "79",
  "name": "gradle_custom_gradledir",
  "clone_url": "http://***@cpe.gitlab.com:8090/zmg/gradle_custom_gradledir.git"
}
```


Status Code

Status Code	Description
200	OK
400	Bad Request

Error Code

The error code format is SVCSTG.REPO.[Error_ID], for example, SVCSTG.REPO.0401. For details, see [ServiceStage Error Codes](#).

5.11 Obtaining a Project Branch

Function

This API is used to obtain a project branch.

URI

GET /v1/{project_id}/git/repos/{namespace}/{project}/branches

[Table 5-43](#) describes the parameters.

Table 5-43 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
namespace	Yes	String	Namespace ID. See Obtaining a Repository Namespace .
project	Yes	String	Repository project ID. If the value contains a slash (/), replace it with a colon (:). See Obtaining All Projects in a Namespace .

Request

Table 5-44 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .
X-Repo-Auth	Yes	String	Authorization name. See Obtaining a Git Repository Authorization List .

Response

Table 5-45 Response parameter

Parameter	Type	Description
branches	Array<String>	Project branch.

Example Request

None

Example Response

```
{
  "branches": [
    "main"
  ]
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request

Error Code

The error code format is SVCSTG.REPO.[Error_ID], for example, SVCSTG.REPO.0401. For details, see [ServiceStage Error Codes](#).

5.12 Obtaining a Project Tag

Function

This API is used to obtain a project tag.

URI

GET /v1/{project_id}/git/repos/{namespace}/{project}/tags

[Table 5-46](#) describes the parameters.

Table 5-46 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
namespace	Yes	String	Namespace ID or URL code name. See Obtaining a Repository Namespace .
project	Yes	String	Repository project ID or URL code name. If the value contains a slash (/), replace it with a colon (:). See Obtaining All Projects in a Namespace .

Request

Table 5-47 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .

Parameter	Mandatory	Type	Description
X-Repo-Auth	Yes	String	Authorization name. See Obtaining a Git Repository Authorization List .

Response

Table 5-48 Response parameter

Parameter	Type	Description
tags	Array<String>	Project tag.

Example Request

None

Example Response

```
{
  "tags": [
    "test1"
  ]
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request

Error Code

The error code format is SVCSTG.REPO.[Error_ID], for example, SVCSTG.REPO.0401. For details, see [ServiceStage Error Codes](#).

5.13 Creating a Project Tag

Function

This API is used to create a project tag.

URI

POST /v1/{project_id}/git/repos/{namespace}/{project}/tags

[Table 5-49](#) describes the parameters.

Table 5-49 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
namespace	Yes	String	Namespace ID or URL code name. See Obtaining a Repository Namespace .
project	Yes	String	Repository project ID or URL code name. If the value contains a slash (/), replace it with a colon (:). See Obtaining All Projects in a Namespace .

Table 5-50 Query parameters

Parameter	Mandatory	Type	Description
ref	Yes	String	Branch name, tag name, or commit SHA value.

Request

Table 5-51 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .
X-Repo-Auth	Yes	String	Authorization name. See Obtaining a Git Repository Authorization List .

Table 5-52 Request body parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Tag name.
description	Yes	String	Description.

Response

Table 5-53 Response parameter

Parameter	Type	Description
name	String	Project tag name.

Example Request

Create project tag **test1** and set the tag description to **test version 1**.

```
{
  "name": "test1",
  "description": "test version 1"
}
```

Example Response

```
{
  "name": "test1"
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request

Error Code

The error code format is SVCSTG.REPO.[Error_ID], for example, SVCSTG.REPO.0401. For details, see [ServiceStage Error Codes](#).

5.14 Deleting a Project Tag

Function

This API is used to delete a project tag.

URI

DELETE /v1/{project_id}/git/repos/{namespace}/{project}/tags/{tag_name}

[Table 5-54](#) describes the parameters.

Table 5-54 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
namespace	Yes	String	Namespace ID or URL code name. See Obtaining a Repository Namespace .
project	Yes	String	Repository project ID or URL code name. If the value contains a slash (/), replace it with a colon (:). See Obtaining All Projects in a Namespace .
tag_name	Yes	String	Tag name. See Obtaining a Project Tag .

Request

Table 5-55 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .
X-Repo-Auth	Yes	String	Authorization name. See Obtaining a Git Repository Authorization List .

Response

None

Example Request

None

Example Response

None

Status Code

Status Code	Description
204	OK
400	Bad Request

Error Code

The error code format is SVCSTG.REPO.[Error_ID], for example, SVCSTG.REPO.0401. For details, see [ServiceStage Error Codes](#).

5.15 Obtaining Project Commits

Function

This API is used to obtain the latest ten project commits.

URI

GET /v1/{project_id}/git/repos/{namespace}/{project}/commits

[Table 5-56](#) describes the parameters.

Table 5-56 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
namespace	Yes	String	Namespace ID or URL code name. See Obtaining a Repository Namespace .

Parameter	Mandatory	Type	Description
project	Yes	String	Repository project ID or URL code name. If the value contains a slash (/), replace it with a colon (:). See Obtaining All Projects in a Namespace .

Table 5-57 Query parameters

Parameter	Mandatory	Type	Description
ref	No	String	Branch or tag name. If it is not provided, use the default branch.

Request

Table 5-58 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .
X-Repo-Auth	Yes	String	Authorization name. See Obtaining a Git Repository Authorization List .

Response

Table 5-59 Response parameter

Parameter	Type	Description
commits	Array<Object>	Project commits structure. See Table 5-60 .

Table 5-60 commits

Parameter	Type	Description
sha	String	Commit SHA value.
message	String	Commit description.
authored_date	String	Import time.

Example Request

None

Example Response

```
{
  "commits": [
    {
      "sha": "6dcb09b5b57875f334f61aebd695e2e4193db5e",
      "message": "Fix all the bugs",
      "authored_date": "Fix all the bugs"
    }
  ]
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request

Error Code

The error code format is SVCSTG.REPO.[Error_ID], for example, SVCSTG.REPO.0401. For details, see [ServiceStage Error Codes](#).

5.16 Obtaining a Project Hook

Function

This API is used to obtain a project hook.

URI

GET /v1/{project_id}/git/repos/{namespace}/{project}/hooks

[Table 5-61](#) describes the parameters.

Table 5-61 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
namespace	Yes	String	Namespace ID or URL code name. See Obtaining a Repository Namespace .
project	Yes	String	Repository project ID or URL code name. If the value contains a slash (/), replace it with a colon (:). See Obtaining All Projects in a Namespace .

Request

Table 5-62 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .
X-Repo-Auth	Yes	String	Authorization name. See Obtaining a Git Repository Authorization List .

Response

Table 5-63 Response parameter

Parameter	Type	Description
hooks	Array of objects	Project hooks. See Table 5-64 .

Table 5-64 hooks

Parameter	Type	Description
id	String	Hook ID.
type	String	Hook type.
callback_url	String	Callback URL.

Example Request

None

Example Response

```
{
  "hooks": [
    {
      "id": "1753",
      "type": "github",
      "callback_url": "https://example.com/webhook"
    }
  ]
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request

Error Code

The error code format is SVCSTG.REPO.[Error_ID], for example, SVCSTG.REPO.0401. For details, see [ServiceStage Error Codes](#).

5.17 Creating a Project Hook

Function

This API is used to create a project hook.

URI

POST /v1/{project_id}/git/repos/{namespace}/{project}/hooks

[Table 5-65](#) describes the parameters.

Table 5-65 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
namespace	Yes	String	Namespace ID or URL code name. See Obtaining a Repository Namespace .
project	Yes	String	Repository project ID or URL code name. If the value contains a slash (/), replace it with a colon (:). See Obtaining All Projects in a Namespace .

Request

Table 5-66 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .
X-Repo-Auth	Yes	String	Authorization name. See Obtaining a Git Repository Authorization List .

Table 5-67 Request body parameters

Parameter	Mandatory	Type	Description
secret	Yes	String	Character string used to verify received payloads.
url	Yes	String	URL that is called back when a hook is triggered.

Response

Table 5-68 Response parameters

Parameter	Type	Description
id	String	Hook ID.
type	String	Hook type.
callback_url	String	Callback URL.

Example Request

Create a project hook, verify the **secrete** parameter of the received payloads string, and set the callback URL to **https://example.com/webhook** when the hook is triggered.

```
{
  "secret": "*****",
  "url": "https://example.com/webhook"
}
```

Example Response

```
{
  "id": "1573",
  "type": "github",
  "callback_url": "https://example.com/webhook"
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request

Error Code

The error code format is SVCSTG.REPO.[Error_ID], for example, SVCSTG.REPO.0401. For details, see [ServiceStage Error Codes](#).

5.18 Deleting a Project Hook

Function

This API is used to delete a project hook.

URI

DELETE /v1/{project_id}/git/repos/{namespace}/{project}/hooks/{hook_id}

[Table 5-69](#) describes the parameters.

Table 5-69 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
namespace	Yes	String	Namespace ID or URL code name. See Obtaining a Repository Namespace .
project	Yes	String	Repository project ID or URL code name. If the value contains a slash (/), replace it with a colon (:). See Obtaining All Projects in a Namespace .
hook_id	Yes	String	Hook ID. See Obtaining a Project Hook .

Request

Table 5-70 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .
X-Repo-Auth	Yes	String	Authorization name. See Obtaining a Git Repository Authorization List .

Response

None

Example Request

None

Example Response

None

Status Code

Status Code	Description
204	OK
400	Bad Request

Error Code

The error code format is SVCSTG.REPO.[Error_ID], for example, SVCSTG.REPO.0401. For details, see [ServiceStage Error Codes](#).

5.19 Obtaining a Repository File Directory

Function

This API is used to obtain a repository file directory.

URI

GET /v1/{project_id}/git/files/{namespace}/{project}/trees

[Table 5-71](#) describes the parameters.

Table 5-71 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
namespace	Yes	String	Namespace ID or URL code name. See Obtaining a Repository Namespace .
project	Yes	String	Repository project ID or URL code name. If the value contains a slash (/), replace it with a colon (:). See Obtaining All Projects in a Namespace .

Table 5-72 Query parameters

Parameter	Mandatory	Type	Description
ref	Yes	String	Branch name, tag name, or commit SHA value.

Request

Table 5-73 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .
X-Repo-Auth	Yes	String	Authorization name. See Obtaining a Git Repository Authorization List .

Response

Table 5-74 Response parameter

Parameter	Type	Description
paths	Array<String>	File path.

Example Request

None

Example Response

```
{
  "paths": [
    "files/whitespace"
  ]
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request

Error Code

The error code format is SVCSTG.REPO.[Error_ID], for example, SVCSTG.REPO.0401. For details, see [ServiceStage Error Codes](#).

5.20 Obtaining Repository File Contents

Function

This API is used to obtain repository file contents.

URI

GET /v1/{project_id}/git/files/{namespace}/{project}/{path}

[Table 5-75](#) describes the parameters.

Table 5-75 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
namespace	Yes	String	Namespace ID or URL code name. See Obtaining a Repository Namespace .
project	Yes	String	Repository project ID or URL code name. If the value contains a slash (/), replace it with a colon (:). See Obtaining All Projects in a Namespace .
path	Yes	String	File path. If the value contains a slash (/), replace it with a colon (:). See Obtaining a Repository File Directory .

Table 5-76 Query parameters

Parameter	Mandatory	Type	Description
ref	Yes	String	Branch name, tag name, or commit SHA value.

Request

Table 5-77 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .
X-Repo-Auth	Yes	String	Authorization name. See Obtaining a Git Repository Authorization List .

Response

Table 5-78 Response parameters

Parameter	Type	Description
path	String	File path.
sha	String	Commit SHA value.
encoding	String	Encoding mode. Value: Base64 or text/plain.
content	String	File contents.

Example Request

None

Example Response

```
{  
  "path": "app/models/key.rb",  
}
```

```
"sha": "4c294617b60715c1d218e61164a3abd4808a4284cbc30e6728a01ad9aada4481",  
"encoding": "base64",  
"content": "lyA9PSBTY2h1bWEgSW5mb3..."  
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request

Error Code

The error code format is SVCSTG.REPO.[Error_ID], for example, SVCSTG.REPO.0401. For details, see [ServiceStage Error Codes](#).

5.21 Creating a Repository File

Function

This API is used to create a repository file.

URI

POST /v1/{project_id}/git/files/{namespace}/{project}/{path}

[Table 5-79](#) describes the parameters.

Table 5-79 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
namespace	Yes	String	Namespace ID or URL code name. See Obtaining a Repository Namespace .
project	Yes	String	Repository project ID or URL code name. If the value contains a slash (/), replace it with a colon (:). See Obtaining All Projects in a Namespace .
path	Yes	String	File path. If the value contains a slash (/), replace it with a colon (:). See Obtaining a Repository File Directory .

Table 5-80 Query parameters

Parameter	Mandatory	Type	Description
ref	Yes	String	Branch name, tag name, or commit SHA value.

Request

Table 5-81 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .
X-Repo-Auth	Yes	String	Authorization name. See Obtaining a Git Repository Authorization List .

Table 5-82 Request body parameters

Parameter	Mandatory	Type	Description
message	Yes	String	Description.
content	Yes	String	File content after Base64 encoding.

Response

Table 5-83 Response parameter

Parameter	Type	Description
path	String	File path.

Example Request

Create a repository file, set **message** to **create a new file**, and set **content** to the content encoded using Base64.

```
{
  "message": "create a new file",
  "content": "IyA9PSBTY2hWWEgSW5mb3..."
}
```

Example Response

```
{
  "path": "app/project.rb"
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request

Error Code

The error code format is SVCSTG.REPO.[Error_ID], for example, SVCSTG.REPO.0401. For details, see [ServiceStage Error Codes](#).

5.22 Modifying Repository File Contents

Function

This API is used to modify repository file contents.

URI

PUT /v1/{project_id}/git/files/{namespace}/{project}/{path}

[Table 5-84](#) describes the parameters.

Table 5-84 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
namespace	Yes	String	Namespace ID or URL code name. See Obtaining a Repository Namespace .

Parameter	Mandatory	Type	Description
project	Yes	String	Repository project ID or URL code name. If the value contains a slash (/), replace it with a colon (:). See Obtaining All Projects in a Namespace .
path	Yes	String	File path. If the value contains a slash (/), replace it with a colon (:). See Obtaining a Repository File Directory .

Table 5-85 Query parameters

Parameter	Mandatory	Type	Description
ref	Yes	String	Branch name, tag name, or commit SHA value.

Request

Table 5-86 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .
X-Repo-Auth	Yes	String	Authorization name. See Obtaining a Git Repository Authorization List .

Table 5-87 Request body parameters

Parameter	Mandatory	Type	Description
message	Yes	String	Description.
content	Yes	String	File content after Base64 encoding.

Parameter	Mandatory	Type	Description
sha	Yes	String	Blob SHA of a file, which can be obtained through Obtaining Repository File Contents .

Response

Table 5-88 Response parameter

Parameter	Type	Description
path	String	File path.

Example Request

Modify repository file contents.

```
{
  "message": "update file",
  "content": "lyA9PSBTY2h1bWEgSW5mb3...",
  "sha": "4c294617b60715c1d218e61164a3abd4808a4284cbc30e6728a01ad9aada4481"
}
```

Example Response

```
{
  "path": "app/project.rb"
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request

Error Code

The error code format is SVCSTG.REPO.[Error_ID], for example, SVCSTG.REPO.0401. For details, see [ServiceStage Error Codes](#).

5.23 Deleting a Repository File

Function

This API is used to delete a repository file.

URI

DELETE /v1/{project_id}/git/files/{namespace}/{project}/{path}

[Table 5-89](#) describes the parameters.

Table 5-89 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
namespace	Yes	String	Namespace ID or URL code name. See Obtaining a Repository Namespace .
project	Yes	String	Repository project ID or URL code name. If the value contains a slash (/), replace it with a colon (:). See Obtaining All Projects in a Namespace .
path	Yes	String	File path. If the value contains a slash (/), replace it with a colon (:). See Obtaining a Repository File Directory .

Table 5-90 Query parameters

Parameter	Mandatory	Type	Description
ref	Yes	String	Branch name, tag name, or commit SHA value.
message	Yes	String	Commit information.
sha	Yes	String	Blob SHA of a file, which can be obtained through Obtaining Repository File Contents .

Request

Table 5-91 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .
X-Repo-Auth	Yes	String	Authorization name. See Obtaining a Git Repository Authorization List .

Response

None

Example Request

None

Example Response

None

Status Code

Status Code	Description
204	OK
400	Bad Request

Error Code

The error code format is SVCSTG.REPO.[Error_ID], for example, SVCSTG.REPO.0401. For details, see [ServiceStage Error Codes](#).

6 CSE APIs

6.1 API Calling

CSE provides REST APIs, allowing you to call APIs using HTTPS.

CSE supports APIs of configuration management and engine management.

To call the CSE APIs of a microservice engine, see [Calling APIs](#).

6.2 Dynamic Configuration

6.2.1 Importing Configurations

Function

This API is used to import configurations to the configuration management center.

NOTE

- This API applies to microservice engine 2.x.
- Replace **{Endpoint}** with the [endpoint](#) in the [request URI](#). For details, see [Calling APIs](#).

URL

POST /v1/{project_id}/kie/file

Table 6-1 Path parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant sub-project ID, which must be unique. The value contains 1 to 64 characters. See Obtaining a Project ID .

Table 6-2 Query parameters

Parameter	Mandatory	Type	Description
override	Yes	String	force : Forcible import will overwrite duplicate items. skip : Duplicate items will be skipped. abort : If duplicate items are found, the import stops.
label	No	String	To import a specified label, enter a value in the format of {Label key}:{Label value}. If this parameter is empty, the label of the body is imported.

Request

Table 6-3 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token.
x-engine-id	Yes	String	ID of the engine whose configuration is to be uploaded. See Querying the Exclusive Microservice Engine List .
X-Enterprise-Project-ID	No	String	Enterprise project ID. If this parameter is not set, the default enterprise project is default and the ID is 0 .

Table 6-4 Request body parameter

Parameter	Mandatory	Type	Description
upload_file	Yes	File	File to be imported. The file name format is fileName.json. File format: <pre>{ "data": [{ "key": "keyItem1", "value": "valueItem1", "labels": { "1": "1" }, "status": "enabled", "value_type": "text" }] }</pre>

Response

Status code: 200

Table 6-5 Response body parameters

Parameter	Type	Description
success	List< KVDoc >	List of configuration items that are successfully imported.
failure	List< DocFailedOfUpload >	List of configuration items that fail to be imported.

Table 6-6 KVDoc

Parameter	Type	Description
id	String	Configuration ID.
key	String	Configuration key.
value	String	Configuration value.
value_type	String	Type of the configuration value.
status	String	Configuration status.
create_time	Integer	Time when the configuration is created.
update_time	Integer	Time when the configuration is updated.
create_revision	Integer	Version number of the created configuration.

Parameter	Type	Description
update_revision	Integer	Version number of the updated configuration.
labels	Map<String,String>	Configuration labels.

Table 6-7 DocFailedOfUpload

Parameter	Type	Description
key	String	Key of the configuration item that fails to be imported.
labels	Map<String,String>	Labels of the configuration item that fails to be imported.
error_code	String	Error code.
error_message	String	Error message.

Status code: 400

Table 6-8 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_message	String	Error message.
detail	String	Location details.

Status code: 500

Table 6-9 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_message	String	Error message.
detail	String	Location details.

Example Request

Import a configuration file. If the same configurations exist, these configurations will be overwritten based on the body label.

```
POST https://{endpoint}/v1/{project_id}/kie/file/override=force

----WebKitFormBoundary7MA4YWxkTrZu0gW
Content-Disposition: form-data; name="upload_file"; filename="custom_1663124521493.json"
Content-Type: application/json
----WebKitFormBoundary7MA4YWxkTrZu0gW
```

Example Response

Successfully queried.

```
{
  "success": [
    {
      "id": "7e2d346e-d907-4871-bf51-a17fc7e52ad4",
      "key": "2",
      "value": "2",
      "value_type": "text",
      "create_revision": 3,
      "update_revision": 3,
      "status": "enabled",
      "create_time": 1636374809,
      "update_time": 1636374809,
      "labels": {
        "2": "2",
        "environment": "production"
      }
    },
    {
      "id": "907b3891-c691-4ae0-816f-e91eba705e28",
      "key": "1",
      "value": "1",
      "value_type": "text",
      "create_revision": 4,
      "update_revision": 4,
      "status": "enabled",
      "create_time": 1636374809,
      "update_time": 1636374809,
      "labels": {
        "1": "1",
        "environment": "production"
      }
    }
  ],
  "failure": []
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
500	Internal Server Error

Error Code

See [CSE Error Codes](#).

6.2.2 Exporting Configurations

Function

This API is used to export configurations from the configuration management center.

NOTE

- This API applies to microservice engine 2.x.
- Replace **{Endpoint}** with the [endpoint](#) in the [request URI](#). For details, see [Calling APIs](#).

URI

POST /v1/{project_id}/kie/download

Table 6-10 Path parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant sub-project ID, which must be unique. The value contains 1 to 64 characters. See Obtaining a Project ID .

Table 6-11 Query parameters

Parameter	Mandatory	Type	Description
label	No	String	Exports data filtered by label. The format is {Label key}: {Label value}.
match	No	String	Matching option of the filtered items. Value exact indicates exact matching, including the same number of labels. If the value is null, inclusive matching is used.

Request

Table 6-12 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token.
x-engine-id	Yes	String	ID of the engine whose configuration is to be downloaded. See Querying the Exclusive Microservice Engine List .
X-Enterprise-Project-ID	No	String	Enterprise project ID. If this parameter is not set, the default enterprise project is default and the ID is 0 .

Table 6-13 Request body parameter

Parameter	Mandatory	Type	Description
ids	Yes	List<String>	ID list of the configuration items to be exported. See Querying the Configuration List . When ids is empty, this API is called to export all configuration items.

Response

Status code: 200

Table 6-14 Response body parameter

Parameter	Mandatory	Type	Description
metadata	No	Object	Additional information about a configuration item.
data	Yes	List< KVCreateBody >	List of configuration items to be created.

Table 6-15 KVCreateBody

Parameter	Mandatory	Type	Description
id	Yes	String	Configuration ID.
key	Yes	String	Configuration key.
value	No	String	Configuration value.
labels	No	Map<String,String>	Configuration labels.
status	No	String	Configuration status.
value_type	No	String	Type of the configuration value.

Table 6-16 Metadata

Parameter	Mandatory	Type	Description
version	No	String	Version information.
annotations	No	Object	Additional information about a configuration item.

Status code: 400

Table 6-17 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_message	String	Error message.
detail	String	Location details.

Status code: 500

Table 6-18 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_message	String	Error message.

Parameter	Type	Description
detail	String	Location details.

Example Request

Export the configuration whose ID is **0bdfddb8-c4f9-4a88-8360-07a8a616a804**.

POST https://{endpoint}/v1/{project_id}/kie/download

```
{
  "ids":[
    "0bdfddb8-c4f9-4a88-8360-07a8a616a804"
  ]
}
```

Example Response

Successfully queried.

```
{
  "metadata": {
    "version": "2.0.0"
  },
  "data": [
    {
      "id": "0bdfddb8-c4f9-4a88-8360-07a8a616a804",
      "key": "2",
      "labels": {
        "2": "2"
      },
      "value": "2",
      "status": "enabled",
      "value_type": "text"
    },
    {
      "id": "34befef9-7f3a-4394-85f9-142622a11b1d",
      "key": "1",
      "labels": {
        "1": "1"
      },
      "value": "1",
      "status": "enabled",
      "value_type": "text"
    }
  ]
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
500	Internal Server Error

Error Code

See [CSE Error Codes](#).

6.3 Engine Management

6.3.1 Querying Flavors Supported by an Exclusive Microservice Engine

Function

This API is used to query the list of flavors supported by an exclusive microservice engine.

URI

GET /v2/{project_id}/enginemgr/flavors

Table 6-19 Path parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant sub-project ID, which must be unique. The value contains 1 to 64 characters. See Obtaining a Project ID .

Table 6-20 query parameter

Parameter	Mandatory	Type	Description
specType	No	String	- To query a 1.x CSE engine, set it to CSE. - To query a 2.x CSE engine, set it to CSE2.

Request

Table 6-21 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token.

Parameter	Mandatory	Type	Description
X-Enterprise-Project-ID	No	String	Enterprise project ID. If this parameter is not set, the default enterprise project is default and the ID is 0 .

Response

Status code: 200

Table 6-22 Response body parameters

Parameter	Type	Description
total	Integer	Total number of flavors of an exclusive microservice engine.
data	Array of Flavor objects	Flavor details of an exclusive microservice engine.

Table 6-23 Flavor

Parameter	Type	Description
flavor	String	Flavor of an exclusive microservice engine.
description	String	Flavor description of an exclusive microservice engine.

Status code: 400

Table 6-24 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Status code: 500

Table 6-25 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Example Request

Query the list of flavors supported by an exclusive microservice engine.

```
GET /v2/{project_id}/enginemgr/flavors
```

Example Response

```
{
  "total": 0,
  "data": [
    {
      "flavor": "string",
      "description": "string"
    }
  ]
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
500	Internal Server Error

Error Code

See [CSE Error Codes](#).

6.3.2 Querying the Exclusive Microservice Engine List

Function

This API is used to query the Exclusive Microservice Engine List

URI

```
GET /v2/{project_id}/enginemgr/engines
```

Table 6-26 Path parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant sub-project ID, which must be unique. The value contains 1 to 64 characters. See Obtaining a Project ID .

Table 6-27 Query parameters

Parameter	Mandatory	Type	Description
offset	No	Integer	Offset.
limit	No	Integer	Number of data records on each page.
type	No	String	Engine type: CSE or CSE_SHARE.

Request

Table 6-28 Request header parameter

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token.

Response

Status code: 200

Table 6-29 Response body parameters

Parameter	Type	Description
total	Integer	Total number of exclusive microservice engines.
data	Array of EngineInfo objects	Details about exclusive microservice engines.

Table 6-30 EngineInfo

Parameter	Type	Description
id	String	ID of an exclusive microservice engine.
name	String	Name of an exclusive microservice engine.
enterpriseProjectId	String	ID of the enterprise project to which an exclusive microservice engine belongs.
enterpriseProjectName	String	Name of the enterprise project to which an exclusive microservice engine belongs.
type	String	Engine type: CSE or CSE_SHARE.
description	String	Description of an exclusive microservice engine.
flavor	String	Flavor of an exclusive microservice engine.
payment	String	Billing mode of an exclusive microservice engine. 1 indicates pay-per-use, and 2 indicates free of charge.
authType	String	Authentication mode of an exclusive microservice engine. RBAC indicates security authentication, and NONE indicates no authentication.
status	String	Current status of an exclusive microservice engine.
externalAddress	String	Address for accessing an exclusive microservice engine in a VPC on the tenant side.
serviceEndpoint	Object	Address for accessing an exclusive microservice engine component in a VPC on the tenant side.
publicAddress	String	Public network access address of an exclusive microservice engine. You need to enable public network access.
publicServiceEndpoint	Object	Public network access address of an exclusive microservice engine component. You need to enable public network access.
totalInstance	Integer	Total number of instances supported by an exclusive microservice engine.
usedInstance	Integer	Total number of used instances.
availableInstance	Integer	Total number of available instances.
version	String	Current version of an exclusive microservice engine.

Parameter	Type	Description
latestVersion	String	Latest version of an exclusive microservice engine.
createTime	Integer	Creation time of an exclusive microservice engine.
dueTo	Integer	Expiration time of an exclusive microservice engine.
latestJobId	Integer	ID of the latest job of an exclusive microservice engine.
engineAdditionalActions	Array of String	Additional operations allowed by an exclusive microservice engine. Delete ForceDelete Upgrade Retry Configure
specType	String	Deployment type of an exclusive microservice engine. - For the 1.x engine, the value is fixed to CSE. - For the 2.x engine, the value is fixed to CSE2.
reference	Object	Additional information about an exclusive microservice engine.

Table 6-31 Endpoint

Parameter	Type	Description
kie	Object	Endpoint information of the configuration center.
serviceCenter	Object	Endpoint information of the service center.

Table 6-32 EntryPoint

Parameter	Type	Description
masterEntryPoint	String	Primary IPv4 address for accessing an exclusive microservice engine component in a VPC.

Parameter	Type	Description
masterEndpointIpv6	String	Primary IPv6 address for accessing an exclusive microservice engine component in a VPC.
slaveEndpoint	String	Secondary IPv4 address for accessing an exclusive microservice engine component in a VPC.
slaveEndpointIpv6	String	Secondary IPv6 address for accessing an exclusive microservice engine component in a VPC.
type	String	Type of an exclusive microservice engine component. CSE

Table 6-33 EngineReference

Parameter	Type	Description
vpc	String	VPC name.
vpcId	String	VPC ID. The value can contain up to 64 characters.
azList	Array of String	List of AZs for exclusive microservice engine deployment.
networkId	String	Subnet network ID of an exclusive microservice engine.
subnetCidr	String	IPv4 subnet division of an exclusive microservice engine.
subnetCidrV6	String	IPv6 subnet division of an exclusive microservice engine.
subnetGateway	String	Subnet gateway of an exclusive microservice engine.
publicIpId	String	Public IP address ID of an exclusive microservice engine. You need to enable public network access.
serviceLimit	Integer	Total number of microservices supported by a package.
instanceLimit	Integer	Total number of instances supported by a package.
inputs	Map <String,String>	Additional parameter of an exclusive microservice engine.

Status code: 400

Table 6-34 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Status code: 500

Table 6-35 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Example Request

Query the microservice engine list.

```
GET /v2/{project_id}/enginemgr/engines
```

Example Response

```
{
  "total": 1,
  "data": [{
    "id": "891bf21a-4024-4f47-b38c-bd259ca8f10a",
    "name": "test",
    "enterpriseProjectId": "0",
    "enterpriseProjectName": "default",
    "type": "CSE_SHARE",
    "description": "",
    "flavor": "cse.s1.medium2",
    "payment": "0",
    "authType": "RBAC",
    "status": "Available",
    "externalAddress": "192.168.0.169",
    "serviceEndpoint": {
      "kie": {
        "masterEntrypoint": "https://192.168.0.169:30110",
        "masterEntrypointIpv6": "https://[2407:c080:11f0:11:b11d:675c:97ab:65f6]:30110",
        "slaveEntrypoint": null,
        "slaveEntrypointIpv6": null,
        "type": "REGISTRY"
      },
      "serviceCenter": {
        "masterEntrypoint": "https://192.168.0.169:30100",
        "masterEntrypointIpv6": "https://[2407:c080:11f0:11:b11d:675c:97ab:65f6]:30100",
        "slaveEntrypoint": null,

```

```
        "slaveEntrypointIpv6": null,
        "type": "REGISTRY"
    },
    "publicAddress": "",
    "publicServiceEndpoint": {
        "kie": {
            "masterEntrypoint": "",
            "masterEntrypointIpv6": "",
            "slaveEntrypoint": "",
            "slaveEntrypointIpv6": "",
            "type": "REGISTRY"
        },
        "serviceCenter": {
            "masterEntrypoint": "",
            "masterEntrypointIpv6": "",
            "slaveEntrypoint": "",
            "slaveEntrypointIpv6": "",
            "type": "REGISTRY"
        }
    },
    "totalInstance": 200,
    "usedInstance": 0,
    "availableInstance": 200,
    "version": "2.3.1",
    "latestVersion": "2.3.3",
    "createTime": 1635576800332,
    "dueTo": 4102415999000,
    "latestJobId": 12339,
    "engineAdditionalActions": ["Retry"],
    "specType": "CSE2",
    "reference": {
        "vpc": "vpc-test",
        "vpcId": "09902850-9454-4715-9764-018f0c3701hy",
        "azList": ["test"],
        "networkId": "88550801-e892-4f8e-b21b-f7147f604f69",
        "subnetCidr": "192.168.0.0/24",
        "subnetCidrV6": "2407:c080:11f0:11::/64",
        "subnetGateway": "192.168.0.2",
        "publicIpId": null,
        "serviceLimit": 200,
        "instanceLimit": 200,
        "inputs": {
            "is_arm_cluster": "false",
            "nodeFlavor": "s6.large.2"
        }
    }
}
}}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
500	Internal Server Error

Error Code

See [CSE Error Codes](#).

6.3.3 Creating an Exclusive Microservice Engine

Function

This API is used to create an exclusive microservice engine.

URI

POST /v2/{project_id}/enginemgr/engines

Table 6-36 Path parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant sub-project ID, which must be unique. The value contains 1 to 64 characters. See Obtaining a Project ID .

Request

Table 6-37 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token.
X-Enterprise-Project-ID	No	String	Enterprise project ID. If this parameter is not set, the default enterprise project is default and the ID is 0 .

Table 6-38 Request body parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Name of an exclusive microservice engine. The value contains 3 to 24 characters, including letters, and digits, and starts with a letter but cannot end with a hyphen (-). NOTICE Microservice engine name cannot be default .
description	No	String	Description of an exclusive microservice engine. The value contains 0 to 255 characters.

Parameter	Mandatory	Type	Description
payment	Yes	String	Billing mode of an exclusive microservice engine. Set it to the fixed value 1 .
flavor	Yes	String	Flavor of an exclusive microservice engine. • cse.s1.small2: HA 100-instance engine • cse.s1.medium2: HA 200-instance engine • cse.s1.large2: HA 500-instance engine • cse.s1.xlarge2: HA 2,000-instance engine
azList	Yes	Array of String	List of AZs at the current region.
authType	Yes	String	Authentication mode of an exclusive microservice engine. RBAC indicates security authentication, and NONE indicates no authentication.
authConfig	No	Table 6-39	RBAC configuration of a microservice engine. If this parameter is not transferred, programming interface authentication is enabled by default.
vpc	Yes	String	VPC name.
vpcId	Yes	String	VPC ID. The value can contain up to 64 characters.
networkId	Yes	String	Subnet ID of an exclusive microservice engine.
subnetCidr	Yes	String	Subnet division of an exclusive microservice engine.
publicIpId	No	String	ID of the public network address for accessing an exclusive microservice engine.

Parameter	Mandatory	Type	Description
specType	Yes	String	Deployment type of an exclusive microservice engine. - For the 1.x engine, the value is fixed to CSE. - For the 2.x engine, the value is fixed to CSE2.
auth_cred	No	object	This parameter is mandatory when security authentication needs to be enabled for an exclusive microservice engine. It contains the authentication information of the engine.
inputs	No	Map<String,String>	Additional parameter of an exclusive microservice engine.

Table 6-39 authConfig

Parameter	Mandatory	Type	Description
consoleAuthEnabled	Yes	Boolean	Whether to enable console authentication.
sdkAuthEnabled	Yes	Boolean	Whether to enable programming interface authentication. If the programming interface authentication parameter sdkAuthEnabled is set to true , the console security authentication parameter consoleAuthEnabled must also be set to true .

Table 6-40 object

Parameter	Mandatory	Type	Description
pwd	Yes	String	root password for enabling security authentication.

Response

Status code: 200

Table 6-41 Response body parameters

Parameter	Type	Description
id	String	ID of an exclusive microservice engine.
name	String	Name of an exclusive microservice engine.
jobId	Integer	ID of an exclusive microservice engine job.

Status code: 400

Table 6-42 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Status code: 500

Table 6-43 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Example Request

Create exclusive microservice engine **test**: billing mode is pay-per-use, flavor is **cse.s1.small2**, AZ is **test**, security authentication is disabled, and deployment type is CSE2.

```
POST https://{endpoint}/v2/{project_id}/enginemgr/engines
{
  "name": "test",
  "description": "test",
  "payment": "1",
  "flavor": "cse.s1.small2",
  "azList": ["test"],
  "authType": "NONE",
  "vpc": "vpc-demotest",
  "vpcId": "09902850-9454-4715-9764-018f0c37022",
  "networkId": "88550801-e892-4f8e-b21b-f7147f6229",
  "subnetCidr": "192.168.0.5/26",
  "specType": "CSE2",
  "inputs": {
```



```
    "nodeFlavor": "s6.large.2"
  }
}
```

Example Response

```
{
  "id": "891bf21a-4024-4f47-b38c-bd259ca8f10a",
  "name": "test",
  "jobId": 17655
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
500	Internal Server Error

Error Code

See [CSE Error Codes](#).

6.3.4 Querying Details About an Exclusive Microservice Engine

Function

This API is used to query details about an exclusive microservice engine.

URI

GET /v2/{project_id}/enginemgr/engines/{engine_id}

Table 6-44 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant ID, which must be unique. The value contains 1 to 64 characters. See Obtaining a Project ID .
engine_id	Yes	String	ID of an exclusive microservice engine. See Querying the Exclusive Microservice Engine List .

Request

Table 6-45 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token.
X-Enterprise-Project-ID	No	String	Enterprise project ID. If this parameter is not set, the default enterprise project is default and the ID is 0 .

Response

Status code: 200

Table 6-46 Response body parameters

Parameter	Type	Description
id	String	ID of an exclusive microservice engine.
name	String	Name of an exclusive microservice engine.
projectId	String	ID of the project to which an exclusive microservice engine belongs.
enterpriseProjectId	String	ID of the enterprise project to which an exclusive microservice engine belongs. .
enterpriseProjectName	String	Name of the enterprise project to which an exclusive microservice engine belongs. .
type	String	Engine type. CSE: exclusive microservice engine.
description	String	Description of an exclusive microservice engine.
beDefault	Boolean	Whether the exclusive engine is the default engine.
flavor	String	Flavor of an exclusive microservice engine. • cse.s1.small2: HA 100-instance engine • cse.s1.medium2: HA 200-instance engine • cse.s1.large2: HA 500-instance engine • cse.s1.xlarge2: HA 2000-instance engine
payment	String	Billing mode of an exclusive microservice engine. 1 indicates pay-per-use, and 2 indicates free of charge.

Parameter	Type	Description
authType	String	Authentication mode of an exclusive microservice engine. RBAC indicates security authentication, and NONE indicates no authentication.
authConfig	Table 6-47	RBAC configuration of a microservice engine.
status	String	Current status of an exclusive microservice engine.
cceSpec	Object	CCE specification of an exclusive microservice engine.
externalEntryPoint	Object	Access address of an exclusive microservice engine.
publicAddress	String	Public network access address of an exclusive microservice engine. You need to enable public network access.
version	String	Current version of an exclusive microservice engine.
latestVersion	String	Latest version of an exclusive microservice engine.
createTime	Integer	Creation time of an exclusive microservice engine.
createUser	String	Creator of an exclusive microservice engine.
latestJobId	Integer	ID of the latest job of an exclusive microservice engine.
engineAdditionalActions	Array of String	Additional operations allowed by an exclusive microservice engine. Delete ForceDelete Upgrade Retry
specType	String	Deployment type of an exclusive microservice engine. - For the 1.x engine, the value is fixed to CSE. - For the 2.x engine, the value is fixed to CSE2.
reference	Object	Additional information about an exclusive microservice engine.
vmIds	Array of String	ID list of VMs used by an exclusive microservice engine on the resource tenant side.

Table 6-47 authConfig

Parameter	Type	Description
consoleAuthEnabled	Boolean	Whether to enable console authentication.
sdkAuthEnabled	Boolean	Whether to enable programming interface authentication.

Table 6-48 EngineReference

Parameter	Type	Description
vpc	String	VPC name.
vpcId	String	VPC ID. The value can contain up to 64 characters.
azList	Array of String	List of AZs for exclusive microservice engine deployment.
networkId	String	Subnet network ID of an exclusive microservice engine.
subnetCidr	String	IPv4 subnet division of an exclusive microservice engine.
subnetCidrV6	String	IPv6 subnet division of an exclusive microservice engine.
subnetGateway	String	Subnet gateway of an exclusive microservice engine.
publicIpId	String	Public IP address ID of an exclusive microservice engine. You need to enable public network access.
serviceLimit	Integer	Total number of microservices supported by a package.
instanceLimit	Integer	Total number of instances supported by a package.
inputs	Map <String,String>	Additional parameter of an exclusive microservice engine.

Table 6-49 Spec

Parameter	Type	Description
id	Integer	CCE specification ID of an exclusive microservice engine.
engineId	String	ID of an exclusive microservice engine.
specType	String	CCE cluster deployment type of an exclusive microservice engine.
cluster	String	CCE cluster information about an exclusive microservice engine. Currently, the value is null .
clusterId	String	CCE cluster ID of an exclusive microservice engine.
clusterNodes	Object	CCE node list of an exclusive microservice engine.
flavor	String	CCE cluster flavor of an exclusive microservice engine.
region	String	Region where the CCE cluster of an exclusive microservice engine is located.
version	String	CCE cluster version of an exclusive microservice engine.
extendParam	String	Additional parameter of the CCE cluster of an exclusive microservice engine.

Table 6-50 SpecClusterNode

Parameter	Type	Description
clusterNodes	Array of ClusterNode objects	CCE node information of an exclusive microservice engine.

Table 6-51 ClusterNode

Parameter	Type	Description
id	String	CCE node ID of an exclusive microservice engine.
az	String	AZ to which the CCE cluster of an exclusive microservice engine belongs.

Parameter	Type	Description
ip	String	CCE node IP of an exclusive microservice engine.
label	String	CCE node label of an exclusive microservice engine.
status	String	CCE node status of an exclusive microservice engine.

Table 6-52 EngineExternalEntrypoint

Parameter	Type	Description
externalAddress	String	Address for accessing an exclusive microservice engine in a VPC on the tenant side.
publicAddress	String	Public network access address of an exclusive microservice engine. You need to enable public network access.
serviceEndpoint	Object	Address for accessing an exclusive microservice engine component in a VPC on the tenant side.
publicServiceEndpoint	Object	Public network access address of an exclusive microservice engine component. You need to enable public network access.

Table 6-53 Endpoint

Parameter	Type	Description
kie	Object	Endpoint information of the configuration center.
serviceCenter	Object	Endpoint information of the service center.

Table 6-54 EntryPoint

Parameter	Type	Description
masterEntrypoint	String	Primary IPv4 address for accessing an exclusive microservice engine component in a VPC.
masterEntrypointIpv6	String	Primary IPv6 address for accessing an exclusive microservice engine component in a VPC.

Parameter	Type	Description
slaveEntrypoint	String	Secondary IPv4 address for accessing an exclusive microservice engine component in a VPC.
slaveEntrypointIpv6	String	Secondary IPv6 address for accessing an exclusive microservice engine component in a VPC.
type	String	Type of an exclusive microservice engine component. CSE

Status code: 400

Table 6-55 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Status code: 500

Table 6-56 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Example Request

Query details about an exclusive engine with specified **engine_id**.

```
GET /v2/{project_id}/enginemgr/engines/{engine_id}
```

Example Response

```
{
  "id": "891bf21a-4024-4f47-b38c-bd259ca8f10a",
  "name": "test",
  "description": "",
  "authType": "RBAC",
```

```
"flavor": "cse.s1.medium2",
"payment": "0",
"version": "2.3.1",
"latestVersion": "2.3.3",
"status": "Creating",
"beDefault": true,
"createUser": "test",
"createTime": 1635576800332,
"cceSpec": {
  "id": 7465,
  "engineId": "891bf21a-4024-4f47-b38c-bd259ca8f10a",
  "specType": "CCE",
  "cluster": null,
  "clusterId": "41115a6f-912f-11eb-9af9-0255ac100188",
  "clusterNodes": {
    "clusterNodes": [{
      "id": "c13aaf5c-2192-421c-8e03-522e2b9a06b5",
      "az": "test",
      "ip": "172.31.25.277",
      "label": "test",
      "status": "Active"
    }]
  },
  "flavor": null,
  "region": "test",
  "version": "",
  "extendParam": ""
},
"externalEntrypoint": {
  "externalAddress": "192.168.0.169",
  "publicAddress": "",
  "serviceEndpoint": {
    "kie": {
      "masterEntrypoint": "https://192.168.0.169:30110",
      "masterEntrypointIpv6": "https://[2407:c080:11f0:11:b11d:675c:97ab:65f6]:30110",
      "slaveEntrypoint": null,
      "slaveEntrypointIpv6": null,
      "type": "REGISTRY"
    },
    "serviceCenter": {
      "masterEntrypoint": "https://192.168.0.169:30100",
      "masterEntrypointIpv6": "https://[2407:c080:11f0:11:b11d:675c:97ab:65f6]:30100",
      "slaveEntrypoint": null,
      "slaveEntrypointIpv6": null,
      "type": "REGISTRY"
    }
  },
  "publicServiceEndpoint": {
    "kie": {
      "masterEntrypoint": "https://192.168.0.169:30110",
      "masterEntrypointIpv6": null,
      "slaveEntrypoint": null,
      "slaveEntrypointIpv6": null,
      "type": "REGISTRY"
    },
    "serviceCenter": {
      "masterEntrypoint": "https://192.168.0.169:30100",
      "masterEntrypointIpv6": null,
      "slaveEntrypoint": null,
      "slaveEntrypointIpv6": null,
      "type": "REGISTRY"
    }
  },
  "reference": {
    "vpc": "vpc-test",
    "vpcId": "09902850-9454-4715-9764-018f0c3701hy",
    "azList": ["test"],
    "networkId": "88550801-e892-4f8e-b21b-f7147f604f69",
```



```
{
  "subnetCidr": "192.168.0.0/24",
  "subnetCidrV6": "2407:c080:11f0:11::/64",
  "subnetGateway": "192.168.0.2",
  "publicIpId": null,
  "serviceLimit": 200,
  "instanceLimit": 200,
  "inputs": {
    "is_arm_cluster": "false",
    "nodeFlavor": "s6.large.2"
  }
},
"latestJobId": 12339,
"enterpriseProjectId": "0",
"enterpriseProjectName": "default",
"engineAdditionalActions": ["Retry"],
"specType": "CSE2",
"type": "CSE",
"projectId": "string",
"vmIds": [""]
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
500	Internal Server Error

Error Code

See [CSE Error Codes](#).

6.3.5 Deleting an Exclusive Microservice Engine

Function

This API is used to delete an exclusive microservice engine.

URI

DELETE /v2/{project_id}/enginemgr/engines/{engine_id}

Table 6-57 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant sub-project ID, which must be unique. The value contains 1 to 64 characters. See Obtaining a Project ID .

Parameter	Mandatory	Type	Description
engine_id	Yes	String	ID of an exclusive microservice engine.

Request

Table 6-58 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token.
X-Enterprise-Project-ID	No	String	Enterprise project ID. If this parameter is not set, the default enterprise project is default and the ID is 0 .

Response

Status code: 200

Table 6-59 Response body parameters

Parameter	Type	Description
id	String	ID of an exclusive microservice engine.
name	String	Name of an exclusive microservice engine.
jobId	Integer	ID of an exclusive microservice engine job.

Status code: 400

Table 6-60 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Status code: 500

Table 6-61 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Example Request

Delete an engine with specified **engine_id**.

```
DELETE /v2/{project_id}/enginemgr/engines/{engine_id}
```

Example Response

```
{
  "id": "891bf21a-4024-4f47-b38c-bd259ca8f10a",
  "name": "test",
  "jobId": 7256
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
500	Internal Server Error

Error Code

See [CSE Error Codes](#).

6.3.6 Querying Details About an Exclusive Microservice Engine Job

Function

Querying Details About an Exclusive Microservice Engine Job

URI

```
GET /v2/{project_id}/enginemgr/engines/{engine_id}/jobs/{job_id}
```

Table 6-62 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant sub-project ID, which must be unique. The value contains 1 to 64 characters. See Obtaining a Project ID .
engine_id	Yes	String	ID of an exclusive microservice engine. See Querying the Exclusive Microservice Engine List .
job_id	Yes	String	ID of an exclusive microservice engine. See Querying Details About an Exclusive Microservice Engine .

Request

Table 6-63 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token.
X-Enterprise-Project-ID	No	String	Enterprise project ID. If this parameter is not set, the default enterprise project is default and the ID is 0 .

Response

Status code: 200

Table 6-64 Response body parameters

Parameter	Type	Description
id	Integer	Job ID.
engineId	String	ID of an exclusive microservice engine.
type	String	Job type. Create Delete Upgrade Configure

Parameter	Type	Description
description	String	Job description.
status	String	Current status of a job. Init Executing Error Timeout Finished
scheduling	Integer	Whether a job is being executed. 0 : no; 1 : yes.
createUser	String	Creator of a job.
startTime	Integer	Start time of a job.
endTime	Integer	End time of a job.
context	String	Job execution context.
tasks	Array of TaskSteps objects	Job phases.

Table 6-65 TaskSteps

Parameter	Type	Description
taskName	String	Name of a phase.
taskNames	Array of String	List of procedures contained in the current phase.
status	String	Status of the phase.
startTime	Integer	Start time of a phase.
endTime	Integer	End time of a phase.
taskExecutorBrief	Object	Job metadata.
tasks	Array of Task objects	Sub-jobs in the phase.

Table 6-66 TaskExecutorBrief

Parameter	Type	Description
duration	Integer	Duration of a sub-job.

Parameter	Type	Description
description	String	Sub-job description.

Table 6-67 Task

Parameter	Type	Description
jobId	Integer	ID of the job to which the sub-job belongs.
id	Integer	Sub-job ID, which is in UUID format.
type	String	Sub-job type.
assigned	String	Executor of a sub-job.
taskName	String	Sub-job name.
engineName	String	Name of the engine to which the sub-job belongs.
taskOrder	Integer	Sequence in which sub-jobs are executed, in ascending order.
status	String	Sub-job status.
startTime	Integer	Start time of a sub-job.
endTime	Integer	End time of a sub-job.
createTime	Integer	Creation time of a sub-job.
updateTime	Integer	Update time of a sub-job.
timeout	Integer	Whether a sub-job times out.
log	String	Sub-job details, which are auxiliary information generated during the execution.
output	String	Sub-job output information.
taskExecutorBrief	Object	Sub-job metadata.

Status code: 400

Table 6-68 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.

Parameter	Type	Description
detail	String	Location details.

Status code: 500

Table 6-69 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Example Request

Query details about a job with specified **job_id** in an exclusive engine with specified **engine_id**.

```
GET /v2/{project_id}/enginemgr/engines/{engine_id}/jobs/{job_id}
```

Example Response

```
{
  "id": 12339,
  "engineId": "ad76972c-a743-4770-859c-c98a680f4d98",
  "type": "Create",
  "description": "",
  "status": "Finished",
  "scheduling": 0,
  "createUser": "test",
  "startTime": 1636540095901,
  "endTime": 1636540460230,
  "context": "",
  "tasks": [{
    "taskName": "resourcePrepare",
    "taskNames": ["TenantTaskExecutor"],
    "status": "Finished",
    "startTime": 1636540100216,
    "endTime": 1636540110215,
    "taskExecutorBrief": {
      "duration": 60000,
      "description": "Resource preparation"
    }
  }],
  "tasks": [{
    "jobId": 12339,
    "id": 89117,
    "type": "Create",
    "assigned": "",
    "taskName": "TenantTaskExecutor",
    "engineName": "test",
    "taskOrder": 0,
    "status": "Init",
    "startTime": 1636540100216,
    "endTime": 1636540110215,
    "createTime": 1636540095910,
    "updateTime": 1636540110230,
  }]
```

```
    "timeout": -1,  
    "log": "",  
    "output": "{}",  
    "taskExecutorBrief": {  
      "duration": 60000,  
      "description": "Tenant processing"  
    }  
  }  
}  
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
500	Internal Server Error

Error Code

See [CSE Error Codes](#).

7 ServiceComb APIs

7.1 API Calling

CSE provides REST APIs, allowing you to call APIs using HTTPS.

NOTICE

APIs in [Microservice](#), [Schema](#), [Microservice Instance](#), and [Dependency](#) are available only in ME-Riyadh, CN-Hong Kong, and AP-Singapore.

To call the ServiceComb APIs of an exclusive microservice engine, do as follows:

1. Log in to ServiceStage and choose **Cloud Service Engine > Engines**.
2. Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.
When calling the APIs in [Authentication](#), [Microservice](#), [Schema](#), [Microservice Instance](#), and [Dependency](#), view or click  to copy the service center address of the engine.
When calling the APIs in [Configuration Management](#), view or click  to copy the configuration center address of the engine.
3. Call the API by referring to [Calling APIs](#). In [Request URI](#), replace **{Endpoint}** with the obtained service center address.

7.2 Authentication

7.2.1 Obtaining the User Token of an Exclusive Microservice Engine

Function

This API is intended only for exclusive microservice engines with security authentication enabled.

Before accessing APIs of an exclusive microservice engine with security authentication enabled, you need to call this API to obtain a user token. This token is the unique credential for accessing the APIs.

The token must be contained in the request header when an access request is sent. The format is as follows:

```
Authorization:Bearer{Token}
```

Restrictions

None

URI

POST /v4/token

Request

Table 7-1 Request body parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Account name.
password	Yes	String	Password of the account.

Response

Status code: 200

Table 7-2 Response body parameter

Parameter	Type	Description
token	String	User token, which is valid for 12 hours.

Status code: 401

Table 7-3 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Status code: 500

Table 7-4 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Example Request

```
POST https://{Service center address}/v4/token

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

Example Response

```
Status code: 200

OK

{
  "token" :
  "****bGciOiJSUzUxMiIsInR5cCI6IkpXVCJ9.eyJhY2NvdW50Ijoicm9vdCIsImV4cCI6MTY1MDU5MTcwMSwicm9sZXMiOiSiYWRTaW4iXX0.WKwNAjaYMMCSjNX0qCGCeyh13FJRzLousxoXlThdkMwkH-
  pXEmG51_SguH0LLHOZolc8gNJq-ilQg4bxTo1s0pnQZIS3wma0qvE-
  MzaYnFguTuHM7rxD7eZdwnbUe3dhnw9xRqR1hcd-lTuBbLoL9fbED4U_63IoEDyBCJI9D_l0F86uGzpUysCvC-
  t6MrJHgi7miUaO7ZZQmSAUNhmbEoN8IIvp-QtP_cWNWtWaFO-
  eoQrmCT2FdIYiB9MCuELr9-5EGM_mFLPgs6E4fyIGiGHY7IwoGUKOCW5w6Jb0L-2JxeUe3eOI5Md5kzOIAE_EYU
  ATxCbJ5GmgpSSJf*****"
}
```

Status Code

Status Code	Description
200	OK
401	Unauthorized
500	Internal Server Error

Error Code

See [ServiceComb Error Codes](#).

7.3 Microservice

7.3.1 Querying Information About a Microservice

Function

This API is used to query the definition information about a microservice based on **service_id**.

Restrictions

None

URI

GET /v4/{project_id}/registry/microservices/{service_id}

Table 7-5 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	- For a professional microservice engine, enter the tenant project ID. The value contains 1 to 64 characters. See Obtaining a Project ID. You can also enter the fixed value default. - For an exclusive microservice engine, enter the fixed value default.
service_id	Yes	String	Microservice ID, which must be unique. The value contains 1 to 64 characters. Regular expression: <code>^[.*\$]</code> . See Querying Information About All Microservices .

Request

Table 7-6 Request header parameter

Parameter	Mandatory	Type	Description
Authorization	No	String	This parameter is mandatory if security authentication is enabled for an exclusive microservice engine. Otherwise, this parameter is not required. The token of an exclusive microservice engine with security authentication enabled is in the following format: Authorization:Bearer {Token} For details about how to obtain the token, see Obtaining the User Token of an Exclusive Microservice Engine.

Response

Status code: 200

Table 7-7 Response body parameter

Parameter	Type	Description
service	MicroService object	Microservice information.

Table 7-8 MicroService

Parameter	Type	Description
serviceId	String	Microservice ID, which must be unique. The value contains 1 to 64 characters. Regular expression: ^.*\$
environment	String	Microservice environment. Value: development, testing, acceptance, or production.

Parameter	Type	Description
appld	String	Application ID, which must be unique. The value contains 1 to 160 characters. Regular expression: <code>^[a-zA-Z0-9]\$ ^[a-zA-Z0-9][a-zA-Z0-9_-.][a-zA-Z0-9]\$</code>
serviceName	String	Microservice name, which must be unique in an application. The value contains 1 to 128 characters. Regular expression: <code>^[a-zA-Z0-9]\$ ^[a-zA-Z0-9][a-zA-Z0-9_-.][a-zA-Z0-9]\$</code>
version	String	Microservice version. The value contains 1 to 64 characters. Regular expression: <code>^[0-9]\$ ^[0-9]+(\.[0-9]+)\$</code>
description	String	Microservice description. The value contains a maximum of 256 characters.
level	String	Microservice level. Value: FRONT, MIDDLE, or BACK.
registerBy	String	Microservice registration mode. Value: SDK, PLATFORM, SIDECAR, or UNKNOWN.
schemas	Array of strings	Microservice schema content. The value must be 1 to 160 bytes long. Only digits, letters, underscores (<code>_</code>), hyphens (<code>-</code>), and periods (<code>.</code>) are allowed. An array contains a maximum of 100 schemas.
status	String	Microservice status. Value: UP or DOWN. Default value: UP. Value: • UP • DOWN
timestamp	String	Microservice registration time.
modTimestamp	String	Latest modification time (UTC).
framework	Framework object	Development framework.
paths	Array of ServicePath objects	Service path.

Table 7-9 Framework

Parameter	Type	Description
name	String	Microservice development framework. Default value: UNKNOWN.
version	String	Version of the microservice development framework.

Table 7-10 ServicePath

Parameter	Type	Description
Path	String	Route address.
Property	Object	Extended attribute. You can customize a key and value. The value must be at least 1 byte long.

Status code: 400

Table 7-11 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Status code: 500

Table 7-12 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Example Request

Query details about the microservice whose ID is **819706e21b7173306797d19922ce4231441c17c5**.

```
GET https://{endpoint}/v4/{project_id}/registry/microservices/  
819706e21b7173306797d19922ce4231441c17c5
```

Example Response

Status code: 200

Successfully queried.

```
{  
  "service": {  
    "serviceId": "819706e21b7173306797d19922ce4231441c17c5",  
    "appld": "default",  
    "serviceName": "SERVICECENTER",  
    "version": "2.4.8",  
    "level": "BACK",  
    "schemas": [  
      "servicecenter.grpc.api.ServiceCtrl",  
      "servicecenter.grpc.api.ServiceInstanceCtrl"  
    ],  
    "status": "UP",  
    "timestamp": "1616426688",  
    "modTimestamp": "1616426688",  
    "environment": "development"  
  }  
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
500	Internal Server Error

Error Code

See [ServiceComb Error Codes](#).

7.3.2 Deleting Definition Information About a Microservice

Function

This API is used to delete definition and related information about a microservice, and deregister all instances of the microservice.

Restrictions

None

URI

DELETE /v4/{project_id}/registry/microservices/{service_id}

Table 7-13 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	- For a professional microservice engine, enter the tenant project ID. The value contains 1 to 64 characters. See Obtaining a Project ID. You can also enter the fixed value default. - For an exclusive microservice engine, enter the fixed value default.
service_id	Yes	String	Microservice ID, which must be unique. The value contains 1 to 64 characters. Regular expression: ^.*\$. See Querying Information About All Microservices .

Table 7-14 Query parameter

Parameter	Mandatory	Type	Description
force	No	Boolean	Whether to forcibly delete information about a microservice. true: forcible deletion; false: non-forcible deletion. If you choose forcible deletion, all service instances are automatically deregistered and related service dependencies are deleted. If this parameter is not transferred, the service cannot be deleted when it has an instance. Default value: false.

Request

Table 7-15 Request header parameter

Parameter	Mandatory	Type	Description
Authorization	No	String	This parameter is mandatory if security authentication is enabled for an exclusive microservice engine. Otherwise, this parameter is not required. The token of an exclusive microservice engine with security authentication enabled is in the following format: Authorization:Bearer {Token} For details about how to obtain the token, see Obtaining the User Token of an Exclusive Microservice Engine.

Response

Status code: 400

Table 7-16 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Status code: 500

Table 7-17 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Example Request

Delete the microservice whose ID is **e0f0da073f2c91e8979a89ff2d7c69t6**.

```
DELETE https://{endpoint}/v4/d9f4da085f2c11e8959a00ff2d7c69b7/registry/microservices/  
e0f0da073f2c91e8979a89ff2d7c69t6
```

Example Response

None

Status Code

Status Code	Description
200	OK
400	Bad Request
500	Internal Server Error

Error Code

See [ServiceComb Error Codes](#).

7.3.3 Querying Information About All Microservices

Function

This API is used to query definition information about microservices that meet filter criteria.

Restrictions

None

URI

GET /v4/{project_id}/registry/microservices

Table 7-18 Path parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	- For a professional microservice engine, enter the tenant project ID. The value contains 1 to 64 characters. See Obtaining a Project ID. You can also enter the fixed value default. - For an exclusive microservice engine, enter the fixed value default.

Request

Table 7-19 Request header parameter

Parameter	Mandatory	Type	Description
Authorization	No	String	This parameter is mandatory if security authentication is enabled for an exclusive microservice engine. Otherwise, this parameter is not required. The token of an exclusive microservice engine with security authentication enabled is in the following format: <pre>Authorization:Bearer {Token}</pre> For details about how to obtain the token, see Obtaining the User Token of an Exclusive Microservice Engine.

Response

Status code: 200

Table 7-20 Response body parameter

Parameter	Type	Description
services	Array of MicroService objects	Microservice list.

Table 7-21 MicroService

Parameter	Type	Description
serviceId	String	Microservice ID, which must be unique. The value contains 1 to 64 characters. Regular expression: <code>^.*\$</code>
environment	String	Microservice environment. Value: development, testing, acceptance, or production. You can use the API for uploading schemas in batches to add or modify an existing schema only when the value is development , testing , or acceptance . Default value: development.
appld	String	Application ID, which must be unique. The value contains 1 to 160 characters. Regular expression: <code>^[a-zA-Z0-9]\$ ^[a-zA-Z0-9][a-zA-Z0-9_-.][a-zA-Z0-9]\$</code>
serviceName	String	Microservice name, which must be unique in an application. The value contains 1 to 128 characters. Regular expression: <code>^[a-zA-Z0-9]\$ ^[a-zA-Z0-9][a-zA-Z0-9_-.][a-zA-Z0-9]\$</code>
version	String	Microservice version. The value contains 1 to 64 characters. Regular expression: <code>^[0-9]\$ ^[0-9]+([0-9]+)\$</code>
description	String	Microservice description. The value contains a maximum of 256 characters.
level	String	Microservice level. Value: FRONT, MIDDLE, or BACK.
registerBy	String	Microservice registration mode. Value: SDK, PLATFORM, SIDECAR, or UNKNOWN.
schemas	Array of strings	Microservice schema content. The value must be 1 to 160 bytes long. Only digits, letters, underscores (<code>_</code>), hyphens (<code>-</code>), and periods (<code>.</code>) are allowed. An array contains a maximum of 100 schemas.
status	String	Microservice status. Value: UP or DOWN. Default value: UP.

Parameter	Type	Description
timestamp	String	Microservice registration time.
modTimestamp	String	Latest modification time (UTC).
framework	Framework object	Development framework.
paths	Array of ServicePath objects	Service path.

Table 7-22 Framework

Parameter	Type	Description
name	String	Microservice development framework. Default value: UNKNOWN.
version	String	Version of the microservice development framework.

Table 7-23 ServicePath

Parameter	Type	Description
Path	String	Route address.
Property	Object	Extended attribute. You can customize a key and value. The value must be at least 1 byte long.

Status code: 400

Table 7-24 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Status code: 500

Table 7-25 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Example Request

Query information about all registered microservices.

```
GET https://{endpoint}/v4/{project_id}/registry/microservices
```

Example Response

Status code: 200
Successfully queried.

```
{
  "services": [
    {
      "serviceId": "8aed80ea052ac04a64dfc79c24f2170224d074f5",
      "appld": "default",
      "serviceName": "test",
      "version": "1.0.0",
      "description": "this is a test",
      "level": "BACK",
      "status": "UP",
      "timestamp": "1650543950",
      "modTimestamp": "1650543950"
    },
    {
      "serviceId": "dcc6c1073eab3cadb47cea2e1a874b7883b02a63",
      "appld": "test",
      "serviceName": "test1",
      "version": "1.0.0",
      "level": "BACK",
      "status": "UP",
      "timestamp": "1650544223",
      "modTimestamp": "1650544223"
    }
  ]
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
500	Internal Server Error

Error Code

See [ServiceComb Error Codes](#).

7.3.4 Creating Static Information for a Microservice

Function

This API is used to create static information for a microservice before registering a microservice instance. The registered instance is associated with the static information based on **serviceId**. One service corresponds to multiple instances.

serviceId can be customized. If **serviceId** is not customized, the system generates a random service ID.

Restrictions

None

URI

POST /v4/{project_id}/registry/microservices

Table 7-26 Path parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	- For a professional microservice engine, enter the tenant project ID. The value contains 1 to 64 characters. See Obtaining a Project ID. You can also enter the fixed value default. - For an exclusive microservice engine, enter the fixed value default.

Request

Table 7-27 Request header parameter

Parameter	Mandatory	Type	Description
Authorization	No	String	This parameter is mandatory if security authentication is enabled for an exclusive microservice engine. Otherwise, this parameter is not required. The token of an exclusive microservice engine with security authentication enabled is in the following format: Authorization:Bearer {Token} For details about how to obtain the token, see Obtaining the User Token of an Exclusive Microservice Engine.

Table 7-28 Request body parameters

Parameter	Mandatory	Type	Description
service	Yes	MicroService object	Microservice information.
rules	No	Array of Rule objects	Blacklist and whitelist.
instances	No	Array of MicroServiceInstance objects	Instance information.
tags	No	Object	Extended attribute. You can customize a key and value. The value must be at least 1 byte long.

Table 7-29 MicroService

Parameter	Mandatory	Type	Description
serviceId	No	String	Microservice ID, which must be unique. The value contains 1 to 64 characters. Regular expression: <code>^[.*]\$</code>
environment	No	String	Microservice environment. Value: development, testing, acceptance, or production. You can use the API for uploading schemas in batches to add or modify an existing schema only when the value is development , testing , or acceptance . Default value: development.
appld	No	String	Application ID, which must be unique. The value contains 1 to 160 characters. Regular expression: <code>^[a-zA-Z0-9]\$ ^[a-zA-Z0-9][a-zA-Z0-9_-][a-zA-Z0-9]\$</code>
serviceName	Yes	String	Microservice name, which must be unique in an application. The value contains 1 to 128 characters. Regular expression: <code>^[a-zA-Z0-9]\$ ^[a-zA-Z0-9][a-zA-Z0-9_-][a-zA-Z0-9]\$</code>
version	No	String	Microservice version. The value contains 1 to 64 characters. Regular expression: <code>^[0-9]\$ ^[0-9]+(\.[0-9]+)\$</code>
description	No	String	Microservice description. The value contains a maximum of 256 characters.
level	No	String	Microservice level. Value: FRONT, MIDDLE, or BACK.
registerBy	No	String	Microservice registration mode. Value: SDK, PLATFORM, SIDECAR, or UNKNOWN.
schemas	No	Array of strings	Foreign key ID of a microservice access schema. The array length supports a maximum of 100 schemas.

Parameter	Mandatory	Type	Description
status	No	String	Microservice status. Value: UP or DOWN. Default value: UP.
timestamp	No	String	Microservice registration time.
modTimestamp	No	String	Latest modification time (UTC).
framework	No	Framework object	Development framework.
paths	No	Array of ServicePath objects	Service path.

Table 7-30 Framework

Parameter	Mandatory	Type	Description
name	No	String	Microservice development framework. Default value: UNKNOWN .
version	No	String	Version of the microservice development framework.

Table 7-31 ServicePath

Parameter	Mandatory	Type	Description
Path	No	String	Route address.
Property	No	Object	Extended attribute. You can customize a key and value. The value must be at least 1 byte long.

Table 7-32 Rule

Parameter	Mandatory	Type	Description
ruleId	No	String	Customized rule ID.
ruleType	No	String	Rule type. Value: WHITE or BLACK.

Parameter	Mandatory	Type	Description
attribute	No	String	If the value starts with tag_xxx , the attributes are filtered by Tag . Otherwise, the attributes are filtered by serviceld , AppId , ServiceName , Version , Description , Level , or Status .
pattern	No	String	Matching rule. The value is a regular expression containing 1 to 64 characters.
description	No	String	Rule description.
timestamp	No	String	Time when a rule is created. This parameter is used only when you query rules.
modTimestamp	No	String	Update time.

Table 7-33 MicroServiceInstance

Parameter	Mandatory	Type	Description
instanceId	No	String	Instance ID, which must be unique. The instance ID is generated by the service center.
serviceld	No	String	Microservice ID, which must be unique. During instance creation, use the service ID in the URL instead of the service ID here.
version	No	String	Microservice version.
hostName	Yes	String	Host information.
endpoints	Yes	Array of strings	Access address information.
status	No	String	Instance status. Value: UP, DOWN, STARTING, or OUTOFSERVICE. Default value: UP.
properties	No	Object	Extended attribute. You can customize a key and value. The value must be at least 1 byte long.

Parameter	Mandatory	Type	Description
healthCheck	No	HealthCheck object	Health check information.
dataCenterInfo	No	DataCenterInfo object	Data center information.
timestamp	No	String	Time when an instance is created, which is automatically generated.
modTimestamp	No	String	Update time.

Table 7-34 HealthCheck

Parameter	Mandatory	Type	Description
mode	Yes	String	Heartbeat mode. Value: push or pull.
port	No	Integer	Port.
interval	Yes	Integer	Heartbeat interval. Unit: s. If the value is less than 5s, the registration is performed at an interval of 5s.
times	Yes	Integer	Maximum retries.

Table 7-35 DataCenterInfo

Parameter	Mandatory	Type	Description
name	Yes	String	Region name.
region	Yes	String	Region.
availableZone	Yes	String	AZ.

Response

Status code: 200

Table 7-36 Response body parameter

Parameter	Type	Description
serviceId	String	Microservice ID, which must be unique.

Status code: 400

Table 7-37 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Status code: 500

Table 7-38 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Example Request

Create static information about a microservice: microservice name is **test**, microservice version is **1.0.0**, rule type is whitelist, host information of the instance is **instanceTest**, and access address is **rest:127.0.0.1:8080**.

POST https://{endpoint}/v4/{project_id}/registry/microservices

```
{
  "service": {
    "appld": "default",
    "serviceName": "test",
    "version": "1.0.0",
    "description": "this is a test"
  },
  "rules": [{
    "ruleType": "WHITE",
    "attribute": "tag_123",
    "pattern": "aaa"
  }],
  "instances": [{
    "hostName": "instanceTest",
    "endpoints": ["rest:127.0.0.1:8080"]
  }],
  "tags": {
    "test_tag1": "test_tag1",
    "test_tag2": "test_tag2",
    "test_tag3": "test_tag3"
  }
}
```

Example Response

Status code: 200

Successfully created. **serviceld** indicates the ID generated for the microservice. Subsequent operations are operated based on the microservice ID.

```
{
  "serviceld" : "8aed80ea052ac04a64dfc79c24f2170224d074f5"
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
500	Internal Server Error

Error Code

See [ServiceComb Error Codes](#).

7.3.5 Deleting Static Information About Microservices in Batches

Function

This API is used to delete the definitions and related information about microservices in batches, and deregister all instances of the microservices.

Restrictions

None

URI

DELETE /v4/{project_id}/registry/microservices

Table 7-39 Path parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	- For a professional microservice engine, enter the tenant project ID. The value contains 1 to 64 characters. See Obtaining a Project ID. You can also enter the fixed value default. - For an exclusive microservice engine, enter the fixed value default.

Table 7-40 Query parameter

Parameter	Mandatory	Type	Description
force	No	Boolean	Whether to forcibly delete static information about microservices. true: forcible deletion; false: non-forcible deletion. If you choose forcible deletion, all service instances are automatically deregistered and related service dependencies are deleted. If this parameter is not transferred, the service cannot be deleted when it has an instance. Default value: false.

Request

Table 7-41 Request header parameter

Parameter	Mandatory	Type	Description
Authorization	No	String	This parameter is mandatory if security authentication is enabled for an exclusive microservice engine. Otherwise, this parameter is not required. The token of an exclusive microservice engine with security authentication enabled is in the following format: Authorization:Bearer {Token} For details about how to obtain the token, see Obtaining the User Token of an Exclusive Microservice Engine.

Table 7-42 Request body parameter

Parameter	Mandatory	Type	Description
serviceIds	Yes	Array of strings	Service ID list. See Querying Information About All Microservices .

Response

Status code: 200

Table 7-43 Response body parameter

Parameter	Type	Description
services	Array of DelServicesRsplInfo objects	List of services to be deleted.

Table 7-44 DelServicesRspInfo

Parameter	Type	Description
serviceId	String	Microservice ID.

Status code: 400

Table 7-45 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Status code: 500

Table 7-46 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Example Request

Delete the static information about the microservices whose IDs are **id1** and **id2**.

```
DELETE https://{endpoint}/v4/{project_id}/registry/microservices
{
  "serviceIds" : [ "id1", "id2" ]
}
```

Example Response

Status code: 200

Successfully deleted.

```
{
  "services" : [ {
    "serviceId" : "id1"
  },
  {
    "serviceId" : "id2"
  }
]
```

```
]
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
500	Internal Server Error

Error Code

See [ServiceComb Error Codes](#).

7.3.6 Modifying Extended Attributes of a Microservice

Function

This API is used to modify static information about a microservice. To update some fields in the static information, input all static information (including the fields that do not need to be updated) in JSON format.

Restrictions

None

URI

PUT /v4/{project_id}/registry/microservices/{service_id}/properties

Table 7-47 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	- For a professional microservice engine, enter the tenant project ID. The value contains 1 to 64 characters. See Obtaining a Project ID. You can also enter the fixed value default. - For an exclusive microservice engine, enter the fixed value default.

Parameter	Mandatory	Type	Description
service_id	Yes	String	Microservice ID, which must be unique. The value contains 1 to 64 characters. Regular expression: <code>^\.*\$</code> . See Querying Information About All Microservices .

Request

Table 7-48 Request header parameter

Parameter	Mandatory	Type	Description
Authorization	No	String	This parameter is mandatory if security authentication is enabled for an exclusive microservice engine. Otherwise, this parameter is not required. The token of an exclusive microservice engine with security authentication enabled is in the following format: Authorization:Bearer {Token} For details about how to obtain the token, see Obtaining the User Token of an Exclusive Microservice Engine.

Table 7-49 Request body parameters

Parameter	Mandatory	Type	Description
properties	Yes	Object	Extended attribute. You can customize a key and value. The value must be at least 1 byte long.

Table 7-50 Framework

Parameter	Mandatory	Type	Description
name	No	String	Microservice development framework. Default value: UNKNOWN.
version	No	String	Version of the microservice development framework.

Response

Status code: 400

Table 7-51 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Status code: 500

Table 7-52 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Example Request

Modify extended information about a microservice with specified **service_id**.

```
PUT https://{endpoint}/v4/{project_id}/registry/microservices/{service_id}/properties
```

```
{
  "properties": {
    "a": "test"
  }
}
```

Example Response

None

Status Code

Status Code	Description
200	OK
400	Bad Request
500	Internal Server Error

Error Code

See [ServiceComb Error Codes](#).

7.3.7 Querying the Unique Service or Schema ID of a Microservice

Function

This API is used to query the unique service or schema ID of a microservice based on filter criteria.

URI

GET /v4/{project_id}/registry/existence

Table 7-53 Path parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	- For a professional microservice engine, enter the tenant project ID. The value contains 1 to 64 characters. See Obtaining a Project ID. You can also enter the fixed value default. - For an exclusive microservice engine, enter the fixed value default.

Table 7-54 Query parameters

Parameter	Mandatory	Type	Description
type	Yes	String	Request type. • microservice • schema NOTE • If type is set to microservice, env, appld, serviceName, and version are mandatory. • If type is set to schema, serviceld and schemald are mandatory.
env	No	String	Microservice environment. Value: development, testing, acceptance, or production.
appld	Yes	String	Application ID, which needs to be transferred when the resource type is microservice . The value contains 1 to 160 characters. Regular expression: <code>^[a-zA-Z0-9]\$ ^[a-zA-Z0-9][a-zA-Z0-9_-.][a-zA-Z0-9]\$</code>
serviceName	Yes	String	Microservice name, which needs to be transferred when the resource type is microservice . The value contains 1 to 128 characters. Regular expression: <code>^[a-zA-Z0-9]\$ ^[a-zA-Z0-9][a-zA-Z0-9_-.][a-zA-Z0-9]\$</code>
version	Yes	String	Microservice version, which needs to be transferred when the resource type is microservice . The value contains 1 to 64 characters. Regular expression: <code>^[0-9]\$ ^[0-9]+(\.[0-9]+)\$</code>
serviceld	Yes	String	Microservice ID, which needs to be transferred when the resource type is schema . The value contains 1 to 64 characters. Regular expression: <code>^\.*\$</code>

Parameter	Mandatory	Type	Description
schemald	Yes	String	Schema ID, which needs to be transferred when the resource type is schema . The value contains 1 to 160 characters. Regular expression: <code>^[a-zA-Z0-9]{1,160}\$ ^[a-zA-Z0-9][a-zA-Z0-9_-.]{0,158}[a-zA-Z0-9]\$</code> .

Request

Table 7-55 Request header parameter

Parameter	Mandatory	Type	Description
Authorization	No	String	This parameter is mandatory if security authentication is enabled for an exclusive microservice engine. Otherwise, this parameter is not required. The token of an exclusive microservice engine with security authentication enabled is in the following format: Authorization:Bearer {Token} For details about how to obtain the token, see Obtaining the User Token of an Exclusive Microservice Engine.

Response

Status code: 200

Table 7-56 Response header parameter

Parameter	Type	Description
X-Schema-Summary	String	Summary of a microservice schema.

Table 7-57 Response body parameters

Parameter	Type	Description
serviceld	String	When a service is queried, a service ID is returned.
schemald	String	When a schema is queried, a schema ID is returned.

Status code: 400

Table 7-58 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Status code: 500

Table 7-59 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Example Request

Query **serviceld** based on **microservice** as **type**, **default** as **appld**, **service** as **serviceName**, and **1.0.0** as **version**.

```
GET https://{endpoint}/v4/{project_id}/registry/existence?  
type=microservice&appld=default&serviceName=service&version=1.0.0
```

Example Response

Status code: 200

Successfully queried. **serviceld** or **schemald** is returned.

```
{  
  "serviceld" : "8aed80ea052ac04a64dfc79c24f2170224d074f5"  
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
500	Internal Server Error

Error Code

See [ServiceComb Error Codes](#).

7.4 Schema

7.4.1 Querying a Microservice Schema

Function

Query a microservice schema based on **service_id** and **schema_id**.

Restrictions

None

URI

GET /v4/{project_id}/registry/microservices/{service_id}/schemas/{schema_id}

Table 7-60 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	- For a professional microservice engine, enter the tenant project ID. The value contains 1 to 64 characters. See Obtaining a Project ID. You can also enter the fixed value default. - For an exclusive microservice engine, enter the fixed value default.

Parameter	Mandatory	Type	Description
service_id	Yes	String	Microservice ID, which must be unique. The value contains 1 to 64 characters. Regular expression: ^.*\$. See Querying Information About All Microservices .
schema_id	Yes	String	Schema ID. See Querying All Schema Information About a Microservice .

Request

Table 7-61 Request header parameter

Parameter	Mandatory	Type	Description
Authorization	No	String	This parameter is mandatory if security authentication is enabled for an exclusive microservice engine. Otherwise, this parameter is not required. The token of an exclusive microservice engine with security authentication enabled is in the following format: Authorization:Bearer {Token} For details about how to obtain the token, see Obtaining the User Token of an Exclusive Microservice Engine.

Response

Status code: 200

Table 7-62 Response header parameter

Parameter	Type	Description
X-Schema-Summary	String	Schema summary.

Table 7-63 Response body parameter

Parameter	Type	Description
schema	String	Schema content.

Status code: 400

Table 7-64 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Status code: 500

Table 7-65 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Example Request

Query details about a schema with specified **service_id** and **schema_id**.

```
GET https://{endpoint}/v4/{project_id}/registry/microservices/{service_id}/schemas/{schema_id}
```

Example Response

Status code: 200

Successfully queried. If the summary exists, the value of **X-Schema-Summary** in the header is the summary of the schema.

```
{
  "schema" : "---\nswagger: \"2.0\"\ninfo:\n  version: \"1.0.0\"\n  title: \"swagger definition for\ncom.service.provider.controller.ProviderImpl\"\n  x-java-interface:\n    \"cse.gen.springmvc.provider.provider.ProviderImplIntf\"\n  basePath: \"/provider\"\n  consumes:\n    \"application/json\"\n  produces:\n    \"application/json\"\n  paths:\n    /helloworld:\n      get:\n        operationId:\n          \"helloworld\"\n        produces:\n          - \"application/json\"\n        parameters:\n          - name: \"name\"\n            in: \"query\"\n            required: true\n            type: \"string\"\n        responses:\n          200:\n            description:\n              \"response of 200\"\n            schema:\n              type: \"string\"\n        }\n}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
500	Internal Server Error

Error Code

See [ServiceComb Error Codes](#).

7.4.2 Modifying a Microservice Schema

Function

Modify a microservice schema based on **schema_id**.

Restrictions

In versions earlier than 2.3.33, schema overwriting is not supported in production environment. In version 2.3.33 and later, the environment variable **SCHEMA_EDITABLE** can be used to support schema overwriting.

URI

PUT /v4/{project_id}/registry/microservices/{service_id}/schemas/{schema_id}

Table 7-66 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	- For a professional microservice engine, enter the tenant project ID. The value contains 1 to 64 characters. See Obtaining a Project ID. You can also enter the fixed value default. - For an exclusive microservice engine, enter the fixed value default.

Parameter	Mandatory	Type	Description
service_id	Yes	String	Microservice ID, which must be unique. The value contains 1 to 64 characters. Regular expression: ^.*\$. See Querying Information About All Microservices .
schema_id	Yes	String	Microservice schema ID, which must be unique. The value contains 1 to 160 characters. Regular expression: ^[a-zA-Z0-9]{1,160}\$ ^[a-zA-Z0-9][a-zA-Z0-9_-.]{0,158}[a-zA-Z0-9]\$. See Querying All Schema Information About a Microservice .

Request

Table 7-67 Request header parameter

Parameter	Mandatory	Type	Description
Authorization	No	String	This parameter is mandatory if security authentication is enabled for an exclusive microservice engine. Otherwise, this parameter is not required. The token of an exclusive microservice engine with security authentication enabled is in the following format: Authorization:Bearer {Token} For details about how to obtain the token, see Obtaining the User Token of an Exclusive Microservice Engine.

Table 7-68 Request body parameters

Parameter	Mandatory	Type	Description
schema	Yes	String	Schema content.

Parameter	Mandatory	Type	Description
summary	No	String	Schema summary.

Response

Status code: 400

Table 7-69 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Status code: 500

Table 7-70 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Example Request

Modify the content and summary of a schema with specified **service_id** and **schema_id**.

```
PUT https://{endpoint}/v4/{project_id}/registry/microservices/{service_id}/schemas/{schema_id}

{
  "schema": "---\nswagger: \"2.0\"\ninfo:\n  version: \"1.0.0\"\n  title: \"swagger definition for\ncom.service.provider.controller.ProviderImpl\"\n  x-java-interface:\n    \"cse.gen.springmvc.provider.provider.ProviderImplIntf\"\n  basePath: \"/provider\"\n  consumes:\n    \"application/json\"\n  produces:\n    \"application/json\"\n  paths:\n    /helloworld:\n      get:\n        operationId:\n          \"helloworld\"\n        produces:\n          - \"application/json\"\n        parameters:\n          - name: \"name\"\n            in: \"query\"\n            required: true\n            type: \"string\"\n        responses:\n          \"response of 200\"\n            schema:\n              type: \"string\"\n        \"summary\": \"test\"
}
```

Example Response

None

Status Code

Status Code	Description
200	OK
400	Bad Request
500	Internal Server Error

Error Code

See [ServiceComb Error Codes](#).

7.4.3 Querying All Schema Information About a Microservice

Function

Query all schema information (including **schemald** and **summary**) about a microservice.

URI

GET /v4/{project_id}/registry/microservices/{service_id}/schemas

Table 7-71 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	- For a professional microservice engine, enter the tenant project ID. The value contains 1 to 64 characters. See Obtaining a Project ID. You can also enter the fixed value default. - For an exclusive microservice engine, enter the fixed value default.
service_id	Yes	String	Microservice ID, which must be unique. The value contains 1 to 64 characters. Regular expression: $^{\wedge}.*\$$. See Querying Information About All Microservices .

Table 7-72 Query parameter

Parameter	Mandatory	Type	Description
withSchema	No	Integer	Whether to query the schema content. Default value: 0. 0: Only schema_id and summary are displayed. 1: schema_id , summary , and schema are displayed.

Request

Table 7-73 Request header parameter

Parameter	Mandatory	Type	Description
Authorization	No	String	This parameter is mandatory if security authentication is enabled for an exclusive microservice engine. Otherwise, this parameter is not required. The token of an exclusive microservice engine with security authentication enabled is in the following format: Authorization:Bearer {Token} For details about how to obtain the token, see Obtaining the User Token of an Exclusive Microservice Engine .

Response

Status code: 200

Table 7-74 Response body parameter

Parameter	Type	Description
schemas	Array of Schema objects	Schema list.

Table 7-75 Schema

Parameter	Type	Description
schemald	String	Schema ID.
schema	String	Schema content.
summary	String	Schema summary.

Status code: 400

Table 7-76 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Status code: 500

Table 7-77 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Example Request

Query the schema of a microservice with specified **service_id**, including **schema_id**, **summary**, and **schema**.

```
GET /v4/{project_id}/registry/microservices/{service_id}/schemas?withSchema=1
```

Example Response

Status code: 200

Successfully queried.

```
{
  "schemas": [{
    "schemald": "xxxxmvc",
    "schema": "---\nswagger: \"2.0\"\ninfo:\n  version: \"1.0.0\"\n  title: \"swagger definition for\ncom.service.provider.controller.ProviderImpl\"\n  x-java-interface:\n    \"cse.gen.springmvc.provider.provider.ProviderImplIntf\"\n  basePath: \"/provider\"\n  consumes:\n
```

```

{"application/json"\nproduces:\n- "application/json"\npaths:\n /helloworld:\n  get:\n    operationId:\n "helloworld"\n    produces:\n - "application/json"\n    parameters:\n - name: "name"\n in: "query"\n    required: true\n    type: "string"\n    responses:\n      200:\n        description:\n "response of 200"\n        schema:\n          type: "string"\n          "xxxx",\n          "summary": "abdcda7b4072ef2d7a5fc9aefccf03e5548029ae31c6cd5fc29da7685d6d9e14adea3"\n    }\n  }\n}

```

Status Code

Status Code	Description
200	OK
400	Bad Request
500	Internal Server Error

Error Code

See [ServiceComb Error Codes](#).

7.5 Microservice Instance

7.5.1 Registering a Microservice Instance

Function

This API is used to register a microservice instance after a microservice is created.

Information about the instance must be provided during registration.

instanceid can be customized. If it is customized, the new instance ID will overwrite the original one. If it is not customized, the system automatically generates an ID. If the endpoints are duplicate, the original ID is used.

URI

POST /v4/{project_id}/registry/microservices/{service_id}/instances

Table 7-78 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	- For a professional microservice engine, enter the tenant project ID. The value contains 1 to 64 characters. See Obtaining a Project ID. You can also enter the fixed value default. - For an exclusive microservice engine, enter the fixed value default.
service_id	Yes	String	Microservice ID, which must be unique. The value contains 1 to 64 characters. Regular expression: ^.*\$. See Querying Information About All Microservices .

Request

Table 7-79 Request header parameter

Parameter	Mandatory	Type	Description
Authorization	No	String	This parameter is mandatory if security authentication is enabled for an exclusive microservice engine. Otherwise, this parameter is not required. The token of an exclusive microservice engine with security authentication enabled is in the following format: Authorization:Bearer {Token} For details about how to obtain the token, see Obtaining the User Token of an Exclusive Microservice Engine.

Table 7-80 Request body parameter

Parameter	Mandatory	Type	Description
instance	Yes	MicroServiceInstance object	Microservice instance information.

Table 7-81 MicroServiceInstance

Parameter	Mandatory	Type	Description
instanceId	No	String	Instance ID, which must be unique. The instance ID is generated by the service center.
serviceId	No	String	Microservice ID, which must be unique. During instance creation, use the microservice ID in the URL instead of the microservice ID specified here.
version	No	String	Microservice version.
hostName	Yes	String	Host information.
endpoints	No	Array of strings	Access address information.
status	No	String	Instance status. Value: UP, DOWN, STARTING, or OUTOFSERVICE. Default value: UP.
properties	No	Object	Extended attribute. You can customize a key and value. The value must be at least 1 byte long.
healthCheck	No	HealthCheck object	Health check information.
dataCenterInfo	No	DataCenterInfo object	Data center information.
timestamp	No	String	Time when an instance is created, which is automatically generated.
modificationTimestamp	No	String	Update time.

Table 7-82 HealthCheck

Parameter	Mandatory	Type	Description
mode	Yes	String	Heartbeat mode. Value: push or pull.
port	No	Integer	Port.
interval	Yes	Integer	Heartbeat interval. Unit: s. If the value is less than 5s, the registration is performed at an interval of 5s.
times	Yes	Integer	Maximum retries.

Table 7-83 DataCenterInfo

Parameter	Mandatory	Type	Description
name	Yes	String	Region name.
region	Yes	String	Region.
availableZone	Yes	String	AZ.

Response

Status code: 200

Table 7-84 Response body parameter

Parameter	Type	Description
instanceId	String	Instance ID.

Status code: 400

Table 7-85 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Status code: 500

Table 7-86 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Example Request

Register a microservice instance: host information is **localhost**, heartbeat mode is **push**, heartbeat interval is **30s**, region is **r1**, name is **dc**, and AZ is **az1**.

```
POST https://{endpoint}/v4/{project_id}/registry/microservices/{service_id}/instances
{
  "instance": {
    "endpoints": [ "grpc://127.0.1.312:9980", "rest://127.0.0.111:8081" ],
    "hostName": "localhost",
    "status": "UP",
    "properties": {
      "_TAGS": "A, B",
      "attr1": "a",
      "nodeIP": "127.0.0.1"
    },
    "dataCenterInfo": {
      "name": "dc",
      "region": "r1",
      "availableZone": "az1"
    },
    "healthCheck": {
      "mode": "push",
      "interval": 30,
      "times": 3
    }
  }
}
```

Example Response

Status code: 200
Successfully registered.

```
{
  "instanceId": "8540bb8b693c4ad1a7fb6a756c415244"
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
500	Internal Server Error

Error Code

See [ServiceComb Error Codes](#).

7.5.2 Querying a Microservice Instance Based on service_id

Function

This API is used to query all instances of a microservice based on **service_id** after the instances are registered.

URI

GET /v4/{project_id}/registry/microservices/{service_id}/instances

Table 7-87 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	- For a professional microservice engine, enter the tenant project ID. The value contains 1 to 64 characters. See Obtaining a Project ID. You can also enter the fixed value default. - For an exclusive microservice engine, enter the fixed value default.
service_id	Yes	String	Microservice ID, which must be unique. The value contains 1 to 64 characters. Regular expression: <code>^[.*\$]</code> . See Querying Information About All Microservices .

Table 7-88 Query parameter

Parameter	Mandatory	Type	Description
tags	No	String	Tag. When there are multiple tags, separate them using commas (.). Regular expression: <code>^[a-zA-Z][a-zA-Z0-9_-.]{0,63}\$</code>

Request

Table 7-89 Request header parameters

Parameter	Mandatory	Type	Description
X-ConsumerId	No	String	Microservice consumer ID, which must be unique.
Authorization	No	String	This parameter is mandatory if security authentication is enabled for an exclusive microservice engine. Otherwise, this parameter is not required. The token of an exclusive microservice engine with security authentication enabled is in the following format: Authorization:Bearer {Token} For details about how to obtain the token, see Obtaining the User Token of an Exclusive Microservice Engine.

Response

Status code: 200

Table 7-90 Response body parameter

Parameter	Type	Description
instances	Array of MicroServiceInstance objects	Instance list.

Table 7-91 MicroServiceInstance

Parameter	Type	Description
instanceId	String	Instance ID, which must be unique. The instance ID is generated by the service center.

Parameter	Type	Description
serviceId	String	Microservice ID, which must be unique. During instance creation, use the microservice ID in the URL instead of the microservice ID specified here.
version	String	Microservice version.
hostName	String	Host information.
endpoints	Array of strings	Access address information.
status	String	Instance status. Value: UP, DOWN, STARTING, or OUTOFSERVICE. Default value: UP.
properties	Object	Extended attribute. You can customize a key and value. The value must be at least 1 byte long.
healthCheck	HealthCheck object	Health check information.
dataCenterInfo	DataCenterInfo object	Data center information.
timestamp	String	Time when an instance is created, which is automatically generated.
modificationTimestamp	String	Update time.

Table 7-92 HealthCheck

Parameter	Type	Description
mode	String	Heartbeat mode. Value: push or pull.
port	Integer	Port.
interval	Integer	Heartbeat interval. Unit: s. If the value is less than 5s, the registration is performed at an interval of 5s.
times	Integer	Maximum retries.

Table 7-93 DataCenterInfo

Parameter	Type	Description
name	String	Region name.

Parameter	Type	Description
region	String	Region.
availableZone	String	AZ.

Status code: 400

Table 7-94 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Status code: 500

Table 7-95 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Example Request

Query instance information about a microservice with specified **service_id**.

```
GET https://{endpoint}/v4/{project_id}/registry/microservices/{service_id}/instances
```

Example Response

Status code: 200

Successfully queried.

```
{
  "instances": [
    {
      "instanceId": "8540bb8b693c4ad1a7fb6a756c415244",
      "serviceId": "8aed80ea052ac04a64dfc79c24f2170224d074f5",
      "endpoints": [
        "rest:127.0.0.1:8080"
      ],
      "hostName": "hostNameTest",
      "status": "UP",
      "properties": {
```

```
{
  "engineID": "30c263e5-2eac-4da1-9c72-5abb9ac94550",
  "engineName": "cse-fkln1-HA"
},
"healthCheck": {
  "mode": "push",
  "interval": 30,
  "times": 3
},
"timestamp": "1650545035",
"modTimestamp": "1650545035",
"version": "1.0.0"
}
]
```

Status Code

Status Code	Description
200	OK
400	Bad Request
500	Internal Server Error

Error Code

See [ServiceComb Error Codes](#).

7.5.3 Deregistering a Microservice Instance

Function

This API is used to deregister an instance based on **instance_id**.

URI

DELETE /v4/{project_id}/registry/microservices/{service_id}/instances/{instance_id}

Table 7-96 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	- For a professional microservice engine, enter the tenant project ID. The value contains 1 to 64 characters. See Obtaining a Project ID. You can also enter the fixed value default. - For an exclusive microservice engine, enter the fixed value default.
service_id	Yes	String	Microservice instance ID, which must be unique. The value contains 1 to 64 characters. Regular expression: <code>^[A-Za-z0-9_-.]*\$</code> . See Querying a Microservice Instance Based on service_id .
instance_id	Yes	String	Microservice instance ID, which must be unique. The value contains 1 to 64 characters. Regular expression: <code>^[A-Za-z0-9_-.]*\$</code> . See Querying a Microservice Instance Based on service_id .

Request

Table 7-97 Request header parameter

Parameter	Mandatory	Type	Description
Authorization	No	String	This parameter is mandatory if security authentication is enabled for an exclusive microservice engine. Otherwise, this parameter is not required. The token of an exclusive microservice engine with security authentication enabled is in the following format: Authorization:Bearer {Token} For details about how to obtain the token, see Obtaining the User Token of an Exclusive Microservice Engine.

Response

Status code: 400

Table 7-98 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Status code: 500

Table 7-99 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Example Request

Deregister an instance with specified **instance_id**.

```
DELETE https://{endpoint}/v4/{project_id}/registry/microservices/{service_id}/instances/{instance_id}
```

Example Response

None

Status Code

Status Code	Description
200	OK
400	Bad Request
500	Internal Server Error

Error Code

See [ServiceComb Error Codes](#).

7.5.4 Querying Details About a Microservice Instance

Function

This API is used to query details about an instance based on **service_id** and **instance_id** after the instance is registered.

URI

```
GET /v4/{project_id}/registry/microservices/{service_id}/instances/{instance_id}
```

Table 7-100 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	- For a professional microservice engine, enter the tenant project ID. The value contains 1 to 64 characters. See Obtaining a Project ID. You can also enter the fixed value default. - For an exclusive microservice engine, enter the fixed value default.

Parameter	Mandatory	Type	Description
service_id	Yes	String	Microservice ID, which must be unique. The value contains 1 to 64 characters. Regular expression: ^.*\$. See Querying Information About All Microservices .
instance_id	Yes	String	Microservice instance ID, which must be unique. The value contains 1 to 64 characters. Regular expression: ^[A-Za-z0-9_-.]*\$. See Querying a Microservice Instance Based on service_id .

Table 7-101 Query parameter

Parameter	Mandatory	Type	Description
tags	No	String	Tag. When there are multiple tags, separate them using commas (.). Regular expression: ^[a-zA-Z][a-zA-Z0-9_-.]{0,63}\$

Request

Table 7-102 Request header parameters

Parameter	Mandatory	Type	Description
X-consumerId	No	String	Microservice consumer ID, which must be unique.

Parameter	Mandatory	Type	Description
Authorization	No	String	This parameter is mandatory if security authentication is enabled for an exclusive microservice engine. Otherwise, this parameter is not required. The token of an exclusive microservice engine with security authentication enabled is in the following format: Authorization:Bearer {Token} For details about how to obtain the token, see Obtaining the User Token of an Exclusive Microservice Engine.

Response

Status code: 200

Table 7-103 Response body parameter

Parameter	Type	Description
instance	MicroServiceInstance object	Microservice instance information.

Table 7-104 MicroServiceInstance

Parameter	Type	Description
instanceId	String	Instance ID, which must be unique. The instance ID is generated by the service center.
serviceId	String	Microservice ID, which must be unique. During instance creation, use the microservice ID in the URL instead of the microservice ID specified here.
version	String	Microservice version.
hostName	String	Host information.
endpoints	Array of strings	Access address information.

Parameter	Type	Description
status	String	Instance status. Value: UP, DOWN, STARTING, or OUTOFSERVICE. Default value: UP.
properties	Object	Extended attribute. You can customize a key and value. The value must be at least 1 byte long.
healthCheck	HealthCheck object	Health check information.
dataCenterInfo	DataCenterInfo object	Data center information.
timestamp	String	Time when an instance is created, which is automatically generated.
modificationTimestamp	String	Update time.

Table 7-105 HealthCheck

Parameter	Type	Description
mode	String	Heartbeat mode. Value: push or pull.
port	Integer	Port.
interval	Integer	Heartbeat interval. Unit: s. If the value is less than 5s, the registration is performed at an interval of 5s.
times	Integer	Maximum retries.

Table 7-106 DataCenterInfo

Parameter	Type	Description
name	String	Region name.
region	String	Region.
availableZone	String	AZ.

Status code: 400

Table 7-107 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Status code: 500

Table 7-108 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Example Request

Query details about an instance with specified **service_id** and **instance_id**.

GET https://{endpoint}/v4/{project_id}/registry/microservices/{service_id}/instances/{instance_id}

Example Response

Status code: 200

Successfully queried.

```
{
  "instance": {
    "instanceId": "4994929d6b8044b29462d4f7daa33c4f",
    "serviceId": "8aed80ea052ac04a64dfc79c24f2170224d074f5",
    "endpoints": [
      "rest:127.0.0.1:8080"
    ],
    "hostName": "hostNameTest",
    "status": "UP",
    "properties": {
      "engineID": "30c263e5-2eac-4da1-9c72-5abb9ac94550",
      "engineName": "cse-fkln1-HA"
    },
    "healthCheck": {
      "mode": "push",
      "interval": 30,
      "times": 3
    },
    "timestamp": "1650545270",
    "modTimestamp": "1650545270",
    "version": "1.0.0"
  }
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
500	Internal Server Error

Error Code

See [ServiceComb Error Codes](#).

7.5.5 Modifying the Extended Information About a Microservice Instance

Function

This API is used to add or update the extended information about a microservice instance based on **instance_id** after the instance is registered.

Restrictions

None

URI

PUT /v4/{project_id}/registry/microservices/{service_id}/instances/{instance_id}/properties

Table 7-109 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	- For a professional microservice engine, enter the tenant project ID. The value contains 1 to 64 characters. See Obtaining a Project ID. You can also enter the fixed value default. - For an exclusive microservice engine, enter the fixed value default.

Parameter	Mandatory	Type	Description
service_id	Yes	String	Microservice ID, which must be unique. The value contains 1 to 64 characters. Regular expression: ^.*\$. See Querying Information About All Microservices .
instance_id	Yes	String	Microservice instance ID, which must be unique. The value contains 1 to 64 characters. Regular expression: ^[A-Za-z0-9_.-]*\$. See Querying a Microservice Instance Based on service_id .

Request

Table 7-110 Request header parameters

Parameter	Mandatory	Type	Description
Authorization	No	String	This parameter is mandatory if security authentication is enabled for an exclusive microservice engine. Otherwise, this parameter is not required. The token of an exclusive microservice engine with security authentication enabled is in the following format: Authorization:Bearer {Token} For details about how to obtain the token, see Obtaining the User Token of an Exclusive Microservice Engine.
X-consumerId	No	String	Microservice consumer ID, which must be unique.

Table 7-111 Request body parameter

Parameter	Mandatory	Type	Description
properties	No	Object	Extended attribute. You can customize a key and value. The value must be at least 1 byte long.

Response

Status code: 400

Table 7-112 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Status code: 500

Table 7-113 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Example Request

Update the extended information of an instance with specified **instance_id** to **"attr1": "b"**.

```
PUT https://{endpoint}/v4/{project_id}/registry/microservices/{service_id}/instances/{instance_id}/properties
{
  "properties": {
    "attr1": "b"
  }
}
```

Example Response

None

Status Code

Status Code	Description
200	OK
400	Bad Request
500	Internal Server Error

Error Code

See [ServiceComb Error Codes](#).

7.5.6 Modifying Status of a Microservice Instance

Function

This API is used to update the status of a microservice instance based on **instance_id** after the instance is registered.

Restrictions

The TESTING state is supported only in version 2.3.X.

URI

PUT /v4/{project_id}/registry/microservices/{service_id}/instances/{instance_id}/status

Table 7-114 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	- For a professional microservice engine, enter the tenant project ID. The value contains 1 to 64 characters. See Obtaining a Project ID. You can also enter the fixed value default. - For an exclusive microservice engine, enter the fixed value default.

Parameter	Mandatory	Type	Description
service_id	Yes	String	Microservice ID, which must be unique. The value contains 1 to 64 characters. Regular expression: ^.*\$. See Querying Information About All Microservices .
instance_id	Yes	String	Microservice instance ID, which must be unique. The value contains 1 to 64 characters. Regular expression: ^[A-Za-z0-9_-.]*\$. See Querying a Microservice Instance Based on service_id .

Table 7-115 Query parameter

Parameter	Mandatory	Type	Description
value	Yes	String	Instance status. UP: online; OUTOFSERVICE: off-hook; STARTING: starting; DOWN: offline; TESTING: dialing test.

Request

Table 7-116 Request header parameter

Parameter	Mandatory	Type	Description
Authorization	No	String	This parameter is mandatory if security authentication is enabled for an exclusive microservice engine. Otherwise, this parameter is not required. The token of an exclusive microservice engine with security authentication enabled is in the following format: Authorization:Bearer {Token} For details about how to obtain the token, see Obtaining the User Token of an Exclusive Microservice Engine.

Response

Status code: 400

Table 7-117 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Status code: 500

Table 7-118 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Example Request

Update the status of an instance with specified **instance_id** to **UP**.

```
PUT https://{endpoint}/v4/{project_id}/registry/microservices/{service_id}/instances/{instance_id}/status?value=UP
```

Example Response

None

Status Code

Status Code	Description
200	OK
400	Bad Request
500	Internal Server Error

Error Code

See [ServiceComb Error Codes](#).

7.5.7 Sending Heartbeat Information

Function

This API is used to send heartbeat information. Service providers need to send heartbeat information to the service center, so that the center can check whether service instances are normal.

Restrictions

None

URI

PUT /v4/{project_id}/registry/microservices/{service_id}/instances/{instance_id}/heartbeat

Table 7-119 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	- For a professional microservice engine, enter the tenant project ID. The value contains 1 to 64 characters. See Obtaining a Project ID. You can also enter the fixed value default. - For an exclusive microservice engine, enter the fixed value default.
service_id	Yes	String	Microservice ID, which must be unique. The value contains 1 to 64 characters. Regular expression: <code>^.*\$</code> . See Querying Information About All Microservices .
instance_id	Yes	String	Microservice instance ID, which must be unique. The value contains 1 to 64 characters. Regular expression: <code>^[A-Za-z0-9_-.]*\$</code> . See Querying a Microservice Instance Based on service_id .

Request

Table 7-120 Request header parameter

Parameter	Mandatory	Type	Description
Authorization	No	String	This parameter is mandatory if security authentication is enabled for an exclusive microservice engine. Otherwise, this parameter is not required. The token of an exclusive microservice engine with security authentication enabled is in the following format: Authorization:Bearer {Token} For details about how to obtain the token, see Obtaining the User Token of an Exclusive Microservice Engine.

Response

Status code: 400

Table 7-121 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Status code: 500

Table 7-122 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Example Request

Send heartbeat information to check whether an instance with specified **instance_id** is healthy.

```
PUT https://{endpoint}/v4/{project_id}/registry/microservices/{service_id}/instances/{instance_id}/heartbeat
```

Example Response

None

Status Code

Status Code	Description
200	OK
400	Bad Request
500	Internal Server Error

Error Code

See [ServiceComb Error Codes](#).

7.5.8 Querying a Microservice Instance by Filter Criteria

Function

This API is used to query a microservice instance based on microservice field filter criteria after the instance is registered.

URI

GET /v4/{project_id}/registry/instances

Table 7-123 Path parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	- For a professional microservice engine, enter the tenant project ID. The value contains 1 to 64 characters. See Obtaining a Project ID. You can also enter the fixed value default. - For an exclusive microservice engine, enter the fixed value default.

Table 7-124 Query parameters

Parameter	Mandatory	Type	Description
appld	Yes	String	Application ID, which must be unique. The value contains 1 to 160 characters. Regular expression: <code>^[a-zA-Z0-9]\$ ^[a-zA-Z0-9][a-zA-Z0-9_-.][a-zA-Z0-9]\$</code>
serviceName	Yes	String	Microservice name The value contains 1 to 128 characters. Regular expression: <code>^[a-zA-Z0-9]\$ ^[a-zA-Z0-9][a-zA-Z0-9_-.][a-zA-Z0-9]\$</code>
tags	No	String	Tag. When there are multiple tags, separate them using commas (.). Regular expression: <code>^[a-zA-Z][a-zA-Z0-9_-.]{0,63}\$</code>
env	No	String	Environment information about the instance.

Request

Table 7-125 Request header parameters

Parameter	Mandatory	Type	Description
X-ConsumerId	No	String	Microservice consumer ID, which must be unique.

Parameter	Mandatory	Type	Description
Authorization	No	String	This parameter is mandatory if security authentication is enabled for an exclusive microservice engine. Otherwise, this parameter is not required. The token of an exclusive microservice engine with security authentication enabled is in the following format: Authorization:Bearer {Token} For details about how to obtain the token, see Obtaining the User Token of an Exclusive Microservice Engine.

Response

Status code: 200

Table 7-126 Response body parameter

Parameter	Type	Description
instances	Array of MicroServiceInstance objects	Instance list.

Table 7-127 MicroServiceInstance

Parameter	Type	Description
instanceId	String	Instance ID, which must be unique. The instance ID is generated by the service center.
serviceId	String	Microservice ID, which must be unique. During instance creation, use the microservice ID in the URL instead of the microservice ID specified here.
version	String	Microservice version.
hostName	String	Host information.
endpoints	Array of strings	Access address information.

Parameter	Type	Description
status	String	Instance status. Value: UP, DOWN, STARTING, or OUTOFSERVICE. Default value: UP.
properties	Object	Extended attribute. You can customize a key and value. The value must be at least 1 byte long.
healthCheck	HealthCheck object	Health check information.
dataCenterInfo	DataCenterInfo object	Data center information.
timestamp	String	Time when an instance is created, which is automatically generated.
modTimestamp	String	Update time.

Table 7-128 HealthCheck

Parameter	Type	Description
mode	String	Heartbeat mode. Value: push or pull.
port	Integer	Port.
interval	Integer	Heartbeat interval. Unit: s. If the value is less than 5s, the registration is performed at an interval of 5s.
times	Integer	Maximum retries.

Table 7-129 DataCenterInfo

Parameter	Type	Description
name	String	Region name.
region	String	Region.
availableZone	String	AZ.

Status code: 400

Table 7-130 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Status code: 500

Table 7-131 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Example Request

Query the instance of the microservice whose application ID is **default** and microservice name is **test**.

GET https://{endpoint}/v4/{project_id}/registry/instances?appId=default&serviceName=test

Example Response

Status code: 200

Successfully queried.

```
{
  "instances": [
    {
      "instanceId": "4098483294bb42d8b3c27eee0c166c1c",
      "serviceId": "8aed80ea052ac04a64dfc79c24f2170224d074f5",
      "endpoints": [
        "rest:127.0.0.1:8080"
      ],
      "hostName": "hostNameTest",
      "status": "UP",
      "properties": {
        "engineId": "30c263e5-2eac-4da1-9c72-5abb9ac94550",
        "engineName": "cse-fkln1-HA"
      },
      "healthCheck": {
        "mode": "push",
        "interval": 30,
        "times": 3
      },
      "timestamp": "1650545431",
      "modTimestamp": "1650545442",
      "version": "1.0.0"
    }
  ]
}
```



```
]
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
500	Internal Server Error

Error Code

See [ServiceComb Error Codes](#).

7.5.9 Querying Microservice Instances in Batches

Function

This API is used to query microservice instances in batches based on microservice field filter criteria after the instances are registered.

URI

POST /v4/{project_id}/registry/instances/action

Table 7-132 Path parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	- For a professional microservice engine, enter the tenant project ID. The value contains 1 to 64 characters. See Obtaining a Project ID. You can also enter the fixed value default. - For an exclusive microservice engine, enter the fixed value default.

Table 7-133 Query parameter

Parameter	Mandatory	Type	Description
type	Yes	String	Operation type. Currently, only query is supported.

Request

Table 7-134 Request header parameters

Parameter	Mandatory	Type	Description
X-consumerId	No	String	Microservice consumer ID, which must be unique.
Authorization	No	String	This parameter is mandatory if security authentication is enabled for an exclusive microservice engine. Otherwise, this parameter is not required. The token of an exclusive microservice engine with security authentication enabled is in the following format: Authorization:Bearer {Token} For details about how to obtain the token, see Obtaining the User Token of an Exclusive Microservice Engine.

Table 7-135 Request body parameters

Parameter	Mandatory	Type	Description
services	At least one of the two parameters must be specified.	Array of FindService objects	Structure for querying service information.
instances		Array of FindInstance objects	Structure for querying instance information.

Table 7-136 FindService

Parameter	Mandatory	Type	Description
service	Yes	Dependency Key object	Dependency item.
rev	No	String	Version number of the client cache, which determines the difference between the local cache and the microservice in the service center. • Input 0 for the first time. • If the input rev is the same as that of the current microservice, notModified is returned. If they are different, updated is returned. • If rev is required in the next request, use the value of rev in updated in the response body.

Table 7-137 DependencyKey

Parameter	Mandatory	Type	Description
environment	Yes	String	Environment. Value: development, testing, acceptance, or production.
appld	Yes	String	Application ID, which must be unique.
serviceName	Yes	String	Microservice name

Table 7-138 FindInstance

Parameter	Mandatory	Type	Description
instance	No	HeartbeatSet Element object	Request struct of a microservice instance.

Parameter	Mandatory	Type	Description
rev	No	String	Version number cached by the client. Version number of the client cache, which determines the difference between the local cache and the microservice instance in the service center. • Input 0 for the first time. • If the input rev is the same as that of the current microservice instance, notModified is returned. If they are different, updated is returned. • If rev is required in the next request, use the value of rev in updated in the response body.

Table 7-139 HeartbeatSetElement

Parameter	Mandatory	Type	Description
serviceId	No	String	Microservice ID.
instanceId	No	String	Microservice instance ID.

Response

Status code: 200

Table 7-140 Response body parameters

Parameter	Type	Description
services	BatchFindResult object	Batch query result structure.
instances	BatchFindResult object	Batch query result structure.

Table 7-141 BatchFindResult

Parameter	Type	Description
failed	Array of FindFailedResult objects	Query failure result structure.

Parameter	Type	Description
notModified	Array of integers	Index set corresponding to the request array.
updated	Array of FindResult objects	Query result structure list.

Table 7-142 FindFailedResult

Parameter	Type	Description
indexes	Array of integers	Index set corresponding to the request array.
error	Error object	Error struct.

Table 7-143 Error

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Table 7-144 FindResult

Parameter	Type	Description
index	Integer	Index corresponding to the request array.
rev	String	Versions returned by the server. If the versions are the same as the versions cached on the client, the instances parameter is left empty.
instances	Array of MicroServiceInstance objects	Instance list.

Table 7-145 MicroServiceInstance

Parameter	Type	Description
instanceId	String	Instance ID, which must be unique. The instance ID is generated by the service center.
serviceId	String	Microservice ID, which must be unique. During instance creation, use the microservice ID in the URL instead of the microservice ID specified here.
version	String	Microservice version.
hostName	String	Host information.
endpoints	Array of strings	Access address information.
status	String	Instance status. Value: UP, DOWN, STARTING, or OUTOFSERVICE. Default value: UP.
properties	Object	Extended attribute. You can customize a key and value. The value must be at least 1 byte long.
healthCheck	HealthCheck object	Health check information.
dataCenterInfo	DataCenterInfo object	Data center information.
timestamp	String	Time when an instance is created, which is automatically generated.
modificationTimestamp	String	Update time.

Table 7-146 HealthCheck

Parameter	Type	Description
mode	String	Heartbeat mode. Value: push or pull.
port	Integer	Port.
interval	Integer	Heartbeat interval. Unit: s. If the value is less than 5s, the registration is performed at an interval of 5s.
times	Integer	Maximum retries.

Table 7-147 DataCenterInfo

Parameter	Type	Description
name	String	Region name.
region	String	Region.
availableZone	String	AZ.

Status code: 400

Table 7-148 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Status code: 500

Table 7-149 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Example Request

Query all instances of the microservice whose application ID is **default** and microservice name is **test**.

```
POST https://{endpoint}/v4/{project_id}/registry/instances/action?type=query
{
  "services": [
    {
      "service": {
        "environment": "",
        "appld": "default",
        "serviceName": "test"
      },
      "rev": "0"
    }
  ]
}
```

Example Response

Status code: 200
Successfully queried.

```
{
  "services": {
    "updated": [
      {
        "index": 0,
        "rev": "0feb784798bca7b2fb4de8351578c4437b516c4b",
        "instances": [
          {
            "instanceId": "79cdaf47cacf43a5b2b4185527da2255",
            "serviceId": "8aed80ea052ac04a64dfc79c24f2170224d074f5",
            "endpoints": [
              "rest:127.0.0.1:8080"
            ],
            "hostName": "hostNameTest",
            "status": "UP",
            "properties": {
              "engineID": "30c263e5-2eac-4da1-9c72-5abb9ac94550",
              "engineName": "cse-fkln1-HA"
            },
            "healthCheck": {
              "mode": "push",
              "interval": 30,
              "times": 3
            },
            "timestamp": "1650546183",
            "modTimestamp": "1650546183",
            "version": "1.0.0"
          }
        ]
      }
    ]
  }
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
500	Internal Server Error

Error Code

See [ServiceComb Error Codes](#).

7.6 Dependency

7.6.1 Querying All Providers of a Microservice

Function

This API is used to query all providers of a microservice based on **consumer_id**. For example, if service A depends on services B and C, you can query the information about services B and C based on service A ID.

URI

GET /v4/{project_id}/registry/microservices/{consumer_id}/providers

Table 7-150 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	- For a professional microservice engine, enter the tenant project ID. The value contains 1 to 64 characters. See Obtaining a Project ID. You can also enter the fixed value default. - For an exclusive microservice engine, enter the fixed value default.
consumer_id	Yes	String	Consumer service ID.

Table 7-151 Query parameters

Parameter	Mandatory	Type	Description
noSelf	No	Integer	Whether to return the self-dependency relationship. 0: no; 1: yes.
sameDomain	No	Integer	Whether to return the shared-service relationship. 0: no; 1: yes.

Request

Table 7-152 Request header parameter

Parameter	Mandatory	Type	Description
Authorization	No	String	This parameter is mandatory if security authentication is enabled for an exclusive microservice engine. Otherwise, this parameter is not required. The token of an exclusive microservice engine with security authentication enabled is in the following format: Authorization:Bearer {Token} For details about how to obtain the token, see Obtaining the User Token of an Exclusive Microservice Engine.

Response

Status code: 200

Table 7-153 Response body parameter

Parameter	Type	Description
providers	MicroService object	Microservice information.

Table 7-154 Microservice

Parameter	Type	Description
serviceId	String	Microservice ID, which must be unique. The value contains 1 to 64 characters. Regular expression: ^.*\$
environment	String	Microservice environment. Value: development, testing, acceptance, or production. You can use the API for uploading schemas in batches to add or modify an existing schema only when the value is development , testing , or acceptance . Default value: development.

Parameter	Type	Description
appld	String	Application ID, which must be unique. The value contains 1 to 160 characters. Regular expression: <code>^[a-zA-Z0-9]\$ ^[a-zA-Z0-9][a-zA-Z0-9_-.][a-zA-Z0-9]\$</code>
serviceName	String	Microservice name, which must be unique in an application. The value contains 1 to 128 characters. Regular expression: <code>^[a-zA-Z0-9]\$ ^[a-zA-Z0-9][a-zA-Z0-9_-.][a-zA-Z0-9]\$</code>
version	String	Microservice version. The value contains 1 to 64 characters. Regular expression: <code>^[0-9]\$ ^[0-9]+([0-9]+)\$</code>
description	String	Microservice description. The value contains a maximum of 256 characters.
level	String	Microservice level. Value: FRONT, MIDDLE, or BACK.
registerBy	String	Microservice registration mode. Value: SDK, PLATFORM, SIDECAR, or UNKNOWN.
schemas	Array of strings	Microservice schema content. The value must be 1 to 160 bytes long. Only digits, letters, underscores (<code>_</code>), hyphens (<code>-</code>), and periods (<code>.</code>) are allowed. An array contains a maximum of 100 schemas.
status	String	Microservice status. Value: UP or DOWN. Default value: UP.
timestamp	String	Microservice registration time.
modTimestamp	String	Latest modification time (UTC).
framework	Framework object	Development framework.
paths	Array of ServicePath objects	Service path.
properties	Object	Extended attribute. You can customize a key and value.

Table 7-155 Framework

Parameter	Type	Description
name	String	Microservice development framework. Default value: UNKNOWN.
version	String	Version of the microservice development framework.

Table 7-156 ServicePath

Parameter	Type	Description
Path	String	Route address.
Property	Object	Extended attribute. You can customize a key and value. The value must be at least 1 byte long.

Status code: 400

Table 7-157 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Status code: 500

Table 7-158 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Example Request

Query the providers of a consumer service with specified **consumer_id**.

```
GET https://{endpoint}/v4/{project_id}/registry/microservices/{consumer_id}/providers
```

Example Response

Status code: 200

Successfully queried. **serviceId** or **schemaId** is returned.

```
{
  "providers": [
    {
      "serviceId": "8aed80ea052ac04a64dfc79c24f2170224d074f5",
      "appId": "default",
      "serviceName": "test",
      "version": "1.0.0",
      "description": "this is a test",
      "level": "BACK",
      "status": "UP",
      "properties": {
        "test": "test"
      },
      "timestamp": "1650543950",
      "modTimestamp": "1650544411"
    }
  ]
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
500	Internal Server Error

Error Code

See [ServiceComb Error Codes](#).

7.7 Configuration Management

7.7.1 Creating a Configuration

Function

This API is used to create a configuration.

URI

POST /v1/{project_id}/kie/kv

Table 7-159 Path parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Fixed value: default.

Request

Table 7-160 Request header parameter

Parameter	Mandatory	Type	Description
Authorization	No	String	This parameter is mandatory if security authentication is enabled for an exclusive microservice engine. Otherwise, this parameter is not required. The token of an exclusive microservice engine with security authentication enabled is in the following format: Authorization:Bearer {Token} For details about how to obtain the token, see Obtaining the User Token of an Exclusive Microservice Engine.

Table 7-161 Request body parameters

Parameter	Mandatory	Type	Description
key	Yes	String	Configuration key.
value	No	String	Configuration value.
labels	No	Map<String,String>	Configuration labels.
status	No	String	Configuration status.
value_type	No	String	Type of the configuration value. Supported types: text, yaml, json, properties, ini, and xml.

Response

Status code: 200

Table 7-162 Response body parameter

Parameter	Type	Description
kie	SingleKieInfo	Configuration information.

Table 7-163 SingleKieInfo

Parameter	Type	Description
id	String	Configuration ID.
key	String	Configuration key.
value	String	Configuration value.
value_type	String	Type of the configuration value.
status	String	Configuration status.
create_time	String	Time when the configuration is created.
update_time	String	Time when the configuration is updated.
labels	Map<String,String>	Configuration labels.
create_version	Integer	Created version number of the configuration.
update_version	Integer	Updated version number of the configuration.

Status code: 400

Table 7-164 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_message	String	Error message.

Status code: 500

Table 7-165 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_message	String	Error message.

Example Request

Create a configuration whose label is **"key1": "value1"** and **"key2": "value2"**.

```
POST https://{endpoint}/v1/{project_id}/kie/kv
{
  "key": "String",
  "value": "String",
  "labels": {
    "key1": "value1",
    "key2": "value2",
  },
  "status": "String",
  "value_type": "String"
}
```

Example Response

Status code: 200
Successfully created. The configuration information is returned.

```
{
  "id": "8a9e6a5d-8d65-48fb-a40c-5150c8479da8",
  "key": "string",
  "labels": {
    "key1": "value1",
    "key2": "value2"
  },
  "value": "string",
  "value_type": "text",
  "status": "enabled",
  "create_time": 1623139038,
  "update_time": 1623139038,
  "create_revision": 13,
  "update_revision": 13,
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
409	Conflict
500	Internal Server Error

Error Code

See [ServiceComb Error Codes](#).

7.7.2 Modifying a Configuration

Function

This API is used to modify a configuration.

URI

PUT /v1/{project_id}/kie/kv/{kv_id}

Table 7-166 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Fixed value: default.
kv_id	Yes	String	ID of the configuration to be modified. See Querying the Configuration List .

Request

Table 7-167 Request header parameter

Parameter	Mandatory	Type	Description
Authorization	No	String	This parameter is mandatory if security authentication is enabled for an exclusive microservice engine. Otherwise, this parameter is not required. The token of an exclusive microservice engine with security authentication enabled is in the following format: Authorization:Bearer {Token} For details about how to obtain the token, see Obtaining the User Token of an Exclusive Microservice Engine.

Table 7-168 Request body parameters

Parameter	Mandatory	Type	Description
value	Yes	String	Configuration value.
status	No	String	Configuration status.

Response

Status code: 200

Table 7-169 Response body parameter

Parameter	Type	Description
kie	SingleKieInfo	Configuration information.

Status code: 400

Table 7-170 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_message	String	Error message.

Status code: 500

Table 7-171 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_message	String	Error message.

Example Request

Modify the value of the configuration whose ID is **8a9e6a5d-8d65-48fb-a40c-5150c8479da8** to **this is a test** and set the configuration status to **enabled**.

```
PUT https://{endpoint}/v1/{project_id}/kie/kv/8a9e6a5d-8d65-48fb-a40c-5150c8479da8
```

```
{  
  "value": "this is a test",  
}
```

```
}
  "status": "enabled"
}
```

Example Response

Status code: 200

Successfully modified. The configuration information is returned.

```
{
  "id": "8a9e6a5d-8d65-48fb-a40c-5150c8479da8",
  "key": "string",
  "labels": {
    "key1": "value1",
    "key2": "value2"
  },
  "value": "this is a test",
  "value_type": "text",
  "status": "enabled",
  "create_time": 1623139038,
  "update_time": 1623139038,
  "create_revision": 13,
  "update_revision": 13,
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
500	Internal Server Error

Error Code

See [ServiceComb Error Codes](#).

7.7.3 Querying a Configuration

Function

This API is used to query a configuration in the configuration center.

URI

GET /v1/{project_id}/kie/kv/{kv_id}

Table 7-172 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Fixed value: default.

Parameter	Mandatory	Type	Description
kv_id	Yes	String	ID of the configuration to be queried. See Querying the Configuration List .

Request

Table 7-173 Request header parameter

Parameter	Mandatory	Type	Description
Authorization	No	String	This parameter is mandatory if security authentication is enabled for an exclusive microservice engine. Otherwise, this parameter is not required. The token of an exclusive microservice engine with security authentication enabled is in the following format: Authorization:Bearer {Token} For details about how to obtain the token, see Obtaining the User Token of an Exclusive Microservice Engine.

Response

Status code: 200

Table 7-174 Response body parameter

Parameter	Type	Description
id	String	Configuration ID.
key	String	Configuration key.
value	String	Configuration value.
value_type	String	Type of the configuration value.
status	String	Configuration status.
create_time	String	Time when the configuration is created.

Parameter	Type	Description
update_time	String	Time when the configuration is updated.
labels	Map<String,String>	Configuration labels.
create_revision	Integer	Created version number of the configuration.
update_revision	Integer	Updated version number of the configuration.

Status code: 400

Table 7-175 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_message	String	Error message.

Status code: 404

Table 7-176 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_message	String	Error message.

Status code: 500

Table 7-177 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_message	String	Error message.

Example Request

Query the configuration whose configuration ID is **8a9e6a5d-8d65-48fb-a40c-5150c847**.

GET https://{endpoint}/v1/{project_id}/kie/kv/8a9e6a5d-8d65-48fb-a40c-5150c8479da8

Example Response

Status code: 200.

Successfully queried.

```
{
  "id": "8a9e6a5d-8d65-48fb-a40c-5150c8479da8",
  "key": "string",
  "labels": {
    "key1": "value1",
    "key2": "value2"
  },
  "value": "string",
  "value_type": "text",
  "status": "enabled",
  "create_time": 1623139038,
  "update_time": 1623139038,
  "create_revision": 13,
  "update_revision": 13
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceComb Error Codes](#).

7.7.4 Querying the Configuration List

Function

This API is used to query the configuration list in the configuration center.

URI

GET /v1/{project_id}/kie/kv

Table 7-178 Path parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Fixed value: default.

Request

Table 7-179 Request header parameter

Parameter	Mandatory	Type	Description
Authorization	No	String	This parameter is mandatory if security authentication is enabled for an exclusive microservice engine. Otherwise, this parameter is not required. The token of an exclusive microservice engine with security authentication enabled is in the following format: Authorization:Bearer {Token} For details about how to obtain the token, see Obtaining the User Token of an Exclusive Microservice Engine.

Table 7-180 Query parameters

Parameter	Mandatory	Type	Description
label	No	String	Exports data filtered by label. The format is {Label key}: {Label value}.
match	No	String	Matching option of the filtered items. Value exact indicates exact matching, including the same number of labels. If the value is null, inclusive matching is used.

Parameter	Mandatory	Type	Description
revision	No	Integer	Configuration version number This parameter is used to determine the difference between the local cache configuration and the configuration in the configuration center. • Input 0 for the first time. • If the input revision value is the same as the current revision value, status code 304 is returned, but no body is returned. If they are inconsistent, both revision values are returned. • If revision is required in the next request, use the value of update_revision in the response body.

Response

Status code: 200

Table 7-181 Response body parameters

Parameter	Type	Description
total	Integer	Number of query results.
data	List< SingleKieInfo >	List of queried kie.

Table 7-182 SingleKieInfo

Parameter	Type	Description
id	String	Configuration ID.
key	String	Configuration key.
value	String	Configuration value.
value_type	String	Type of the configuration value.
status	String	Configuration status.
create_time	String	Time when the configuration is created.

Parameter	Type	Description
update_time	String	Time when the configuration is updated.
labels	Map<String,String>	Configuration labels.
create_revision	Integer	Created version number of the configuration.
update_revision	Integer	Updated version number of the configuration.

Status code: 304

None

Status code: 400

Table 7-183 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_message	String	Error message.

Status code: 500

Table 7-184 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_message	String	Error message.

Example Request

Query the configuration list.

GET https://{endpoint}/v1/{project_id}/kie/kv

Example Response

Status code: 200

Successfully queried.

```
{
  "total": 1,
  "data": [{
    "id": "8a9e6a5d-8d65-48fb-a40c-5150c8479da8",
    "key": "string",
    "labels": {
      "key1": "value1",
      "key2": "value2"
    },
    "value": "string",
    "value_type": "text",
    "status": "enabled",
    "create_time": 1623139038,
    "update_time": 1623139038,
    "create_revision": 13,
    "update_revision": 13
  }]
}
```

Status Code

Status Code	Description
200	OK
304	Not Modified
400	Bad Request
500	Internal Server Error

Error Code

See [ServiceComb Error Codes](#).

7.7.5 Deleting a Configuration

Function

This API is used to delete a configuration.

URI

DELETE /v1/{project_id}/kie/kv/{kv_id}

Table 7-185 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Fixed value: default.
kv_id	Yes	String	ID of the configuration to be deleted. See Querying the Configuration List .

Request

Table 7-186 Request header parameter

Parameter	Mandatory	Type	Description
Authorization	No	String	This parameter is mandatory if security authentication is enabled for an exclusive microservice engine. Otherwise, this parameter is not required. The token of an exclusive microservice engine with security authentication enabled is in the following format: Authorization:Bearer {Token} For details about how to obtain the token, see Obtaining the User Token of an Exclusive Microservice Engine.

Response

Status code: 200

Successfully deleted.

Status code: 404

Table 7-187 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_message	String	Error message.

Status code: 400

Table 7-188 Response body parameters

Parameter	Type	Description
error_code	String	Error code.

Parameter	Type	Description
error_message	String	Error message.

Status code: 500

Table 7-189 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_message	String	Error message.

Example Request

Delete the configuration whose configuration ID is **8a9e6a5d-8d65-48fb-a40c-5150c8479da8**.

DELETE https://{endpoint}/v1/{project_id}/kie/kv/8a9e6a5d-8d65-48fb-a40c-5150c8479da8

Example Response

Status code: 204

Successfully deleted.

Status Code

Status Code	Description
200	No Content
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceComb Error Codes](#).

7.7.6 Deleting All Configurations

Function

This API is used to delete all configurations.

URI

DELETE /v1/{project_id}/kie/kv

Table 7-190 Path parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Fixed value: default.

Request

Table 7-191 Request header parameter

Parameter	Mandatory	Type	Description
Authorization	No	String	This parameter is mandatory if security authentication is enabled for an exclusive microservice engine. Otherwise, this parameter is not required. The token of an exclusive microservice engine with security authentication enabled is in the following format: Authorization:Bearer {Token} For details about how to obtain the token, see Obtaining the User Token of an Exclusive Microservice Engine.

Table 7-192 Request body parameter

Parameter	Mandatory	Type	Description
ids	Yes	List<String>	ID list of the configurations to be deleted. See Querying the Configuration List .

Response

Status code: 204

Successfully deleted.

Status code: 404

Table 7-193 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Status code: 500

Table 7-194 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Example Request

Delete all configurations.

```
DELETE https://{endpoint}/v1/{project_id}/kie/kv
```

```
{
  "ids":[
    "id1",
    "id2"
  ]
}
```

Example Response

Status code: 204

Successfully deleted.

Status Code

Status Code	Description
204	No Content
400	Bad Request
500	Internal Server Error

Error Code

See [ServiceComb Error Codes](#).

8 Examples

8.1 Creating and Managing an Environment

Scenarios

An environment is a collection of basic compute (such as CCE and ECS), network (such as ELB and EIP), and middleware (such as DCS and RDS) resources, used for component deployment and running. ServiceStage combines multiple basic resources into an environment, including development, test, pre-production, and production environments. Managing resources and deploying components by environment simplifies O&M management.

The following describes how to call the APIs for [Creating an Environment](#), [Modifying an Environment Resource Based on the Environment ID](#), and [Obtaining Environment Details Based on the Environment ID](#).

For details about how to call APIs, see [Calling APIs](#).

Prerequisites

- You have obtained the IAM user token by calling the API for [obtaining a user token](#), and set the token to an environment variable, that is, **export Token={token}**.
- You have obtained the ServiceStage endpoint to replace **{servicestage_endpoint}** in the following procedure. For details, see [Regions and Endpoints](#).

Procedure

- Step 1** Call the API for [Creating an Environment](#) to create an environment. The following is a request example:

```
curl -k -H "Content-Type:application/json;charset=utf8" -H "X-Auth-Token:$Token" -X POST "https://{servicestage_endpoint}/v3/{project_id}/cas/environments" -d '{
  "name": "env-Kubernetes",
  "description": "test environment",
  "enterprise_project_id": "0",
  "charge_mode": "provided",
}
```



```
"vpc_id": "081c539d-ed09-43b0-81d7-a87603e528b7",
"labels": [
  {
    "key": "testKey",
    "value": "testValue"
  }
],
"deploy_mode": "container"
}]'
```

Information similar to the following is displayed.

```
{
  "id": "9cd33350-5d7f-46ec-ba6d-40ed793f0eb7",
  "name": "env-Kubernetes",
  "description": "test environment",
  "creator": "test_user",
  "enterprise_project_id": "0",
  "vpc_id": "081c539d-ed09-43b0-81d7-a87603e528b7",
  "deploy_mode": "container",
  "vm_cluster_size": 50,
  "labels": [
    {
      "key": "testKey",
      "value": "testValue"
    }
  ],
  "create_time": 1689564968283,
  "update_time": 1689564968283,
  "resources": []
}
```

Record the environment ID in bold in the command output to replace **{environment_id}** in subsequent examples.

Step 2 Call the API for [Modifying an Environment Resource Based on the Environment ID](#) to add resources to the environment.

You can add multiple types of resources, such as computing, network, and middleware resources, in the same VPC to the environment. The following is a request example:

```
curl -k -H "Content-Type:application/json;charset=utf8" -H "X-Auth-Token:$Token" -X PUT "https://
{servicestage_endpoint}/v3/{project_id}/cas/environments/{environment_id}/resources" -d '
{
  "resources": [
    {
      "id": "d2917177-3436-4e53-a5bc-706c3529e0e7",
      "name": "elb-pu6a",
      "type": "elb"
    }
  ]
}'
```

Information similar to the following is displayed.

```
{
  "resources": [
    {
      "id": "d2917177-3436-4e53-a5bc-706c3529e0e7",
      "type": "elb",
      "name": "elb-pu6a"
    }
  ]
}
```

Step 3 Call the API for [Obtaining Environment Details Based on the Environment ID](#) to query environment details based on the environment ID. The following is a request example:

```
curl -k -H "Content-Type:application/json;charset=utf8" -H "X-Auth-Token:$Token" -X GET "https://  
{servicestage_endpoint}/v3/{project_id}/cas/environments/{environment_id}"
```

Information similar to the following is displayed.

```
{  
  "id": "9cd33350-5d7f-46ec-ba6d-40ed793f0eb7",  
  "name": "env-Kubernetes",  
  "description": "test environment",  
  "creator": "test_user",  
  "enterprise_project_id": "0",  
  "vpc_id": "081c539d-ed09-43b0-81d7-a87603e528b7",  
  "deploy_mode": "container",  
  "vm_cluster_size": 50,  
  "labels": [  
    {  
      "key": "testKey",  
      "value": "testValue"  
    }  
  ],  
  "create_time": 1689564968283,  
  "update_time": 1689574181240,  
  "resources": [  
    {  
      "id": "d2917177-3436-4e53-a5bc-706c3529e0e7",  
      "type": "elb",  
      "name": "elb-pu6a"  
    }  
  ]  
}
```

----End

8.2 Creating and Managing an Application

Scenarios

An application is a service system with functions and consists of one or more components.

The following describes how to call the APIs for [Creating an Application](#), [Modifying Application Configurations Based on the Application ID](#), and [Obtaining Application Details Based on the Application ID](#).

For details about how to call APIs, see [Calling APIs](#).

Prerequisites

- You have obtained the IAM user token by calling the API for [obtaining a user token](#), and set the token to an environment variable, that is, **export Token={token}**.
- You have obtained the ServiceStage endpoint to replace **{servicestage_endpoint}** in the following procedure. For details, see [Regions and Endpoints](#).

Procedure

- Step 1** Call the API for [Creating an Application](#) to create an application. The following is a request example:

```
curl -k -H "Content-Type:application/json;charset=utf8" -H "X-Auth-Token:$Token" -X POST "https://  
{servicestage_endpoint}/v3/{project_id}/cas/applications" -d '
```

```
{
  "name": "app-test-application",
  "description": "test application",
  "enterprise_project_id": "0",
  "labels": [
    {
      "key": "testKey",
      "value": "testValue"
    }
  ]
}
```

Information similar to the following is displayed.

```
{
  "id": "2c458700-c8e0-4651-a73b-5219b70a6572",
  "name": "app-test-application",
  "description": "test application",
  "project_id": "063a3158b1a34710b36ad208b9497d00",
  "enterprise_project_id": "0",
  "creator": "test_user",
  "create_time": 1689575641705,
  "update_time": 1689575641705,
  "unified_model": null,
  "labels": [
    {
      "key": "testKey",
      "value": "testValue"
    }
  ]
}
```

Record the application ID in bold in the command output to replace **{application_id}** in subsequent examples.

Step 2 Call the API for [Modifying Application Configurations Based on the Application ID](#) to add an environment variable for the application. The following is a request example:

```
curl -k -H "Content-Type:application/json;charset=utf8" -H "X-Auth-Token:$Token" -X PUT "https://{servicestage_endpoint}/v3/{project_id}/cas/applications/{application_id}/configuration" -d '{
  "environment_id": "21a43958-f64f-4077-8ae3-3a03bf0111c7",
  "configuration": {
    "env": [
      {
        "name": "testName",
        "value": "testValue"
      }
    ]
  }
}'
```

Information similar to the following is displayed.

```
{
  "application_id": "2c458700-c8e0-4651-a73b-5219b70a6572",
  "environment_id": "21a43958-f64f-4077-8ae3-3a03bf0111c7",
  "configuration": {
    "env": [
      {
        "name": "testName",
        "value": "testValue"
      }
    ]
  }
}
```

- Step 3** Call the API for [Obtaining Application Details Based on the Application ID](#) to query application details based on the application ID. The following is a request example:

```
curl -k -H "Content-Type:application/json;charset=utf8" -H "X-Auth-Token:$token" -X GET "https://{servicestage_endpoint}/v3/{project_id}/cas/applications/{application_id}"
```

Information similar to the following is displayed.

```
{
  "id": "2c458700-c8e0-4651-a73b-5219b70a6572",
  "name": "app-test-application",
  "description": "test application",
  "project_id": "063a3158b1a34710b36ad208b9497d00",
  "enterprise_project_id": "0",
  "creator": "test_user",
  "create_time": 1689575641705,
  "update_time": 1689575641705,
  "unified_model": null,
  "labels": [
    {
      "key": "testKey",
      "value": "testValue"
    }
  ],
  "component_count": 0
}
```

----End

8.3 Creating and Managing a Component

Scenarios

A component is a service feature implementation of an application. It is carried by code or software packages and can be independently deployed and run in an environment.

The following describes how to call the APIs for [Creating an Application Component](#), [Obtaining Job Details](#), [Modifying Component Information Based on the Component ID](#), and [Delivering a Component Task Based on the Component ID](#).

For details about how to call APIs, see [Calling APIs](#).

Prerequisites

- You have obtained the IAM user token by calling the API for [obtaining a user token](#), and set the token to an environment variable, that is, **export Token={token}**.
- You have obtained the ServiceStage endpoint to replace **{servicestage_endpoint}** in the following procedure. For details, see [Regions and Endpoints](#).

Procedure

- Step 1** Call the API for [Creating an Application Component](#) to create a component for an application. The following is a request example:

```
curl -k -H "Content-Type:application/json;charset=utf8" -H "X-Auth-Token:$token" -X POST "https://{servicestage_endpoint}/v3/{project_id}/cas/applications/{application_id}/components" -d '
```

```
{
  "name": "test-component",
  "description": "",
  "labels": [],
  "version": "2023.0712.09204",
  "environment_id": "21a43958-f64f-4077-8ae3-3a03bf0111c7",
  "application_id": "2c458700-c8e0-4651-a73b-5219b70a6572",
  "enterprise_project_id": "0",
  "runtime_stack": {
    "name": "Docker",
    "version": "",
    "type": "Docker",
    "deploy_mode": "container"
  },
  "source": {
    "kind": "image",
    "storage": "swr",
    "url": "swr.ap-southeast-1.myhuaweicloud.com/servicestage-demo/test_02:v1",
    "version": "v1"
  },
  "tomcat_opts": {
    "server_xml": ""
  },
  "refer_resources": [
    {
      "id": "1d3091a2-1ecf-11ee-ae69-0255ac1001ba",
      "type": "cce",
      "parameters": {
        "namespace": "default",
        "type": "VirtualMachine"
      }
    }
  ],
  "replica": 2,
  "config_mode": "ui",
  "limit_cpu": 0.25,
  "limit_memory": 0.5,
  "request_cpu": 0.25,
  "request_memory": 0.5,
  "external_accesses": [],
  "storages": [],
  "logs": [],
  "affinity": {
    "az": [],
    "node": [],
    "component": []
  },
  "anti_affinity": {
    "az": [],
    "node": [],
    "component": []
  },
  "update_strategy": {
    "type": "RollingUpdate",
    "max_surge": 0,
    "max_unavailable": 1
  },
  "command": {
    "args": [],
    "command": []
  },
  "custom_metric": {}
}
```

Information similar to the following is displayed.

```
{
  "job_id": "JOB870e9ffe-227e-43a9-9c84-20a7ad02e77c",
  "component_id": "f3f6cb89-831d-4579-8bbc-664487889e38"
}
```

Record **job_id** and **component_id** in bold in the command output to replace **{job_id}** and **{component_id}** in subsequent examples.

Step 2 Call the API for **Obtaining Job Details** to check whether the component is created. The following is a request example:

```
curl -k -H "Content-Type:application/json;charset=utf8" -H "X-Auth-Token:$Token" -X GET "https://{servicestage_endpoint}/v3/{project_id}/cas/jobs/{job_id}"
```

Information similar to the following is displayed.

```
{
  "job": {
    "JOB_ID": "JOB870e9ffe-227e-43a9-9c84-20a7ad02e77c",
    "JOB_NAME": "provision/provMain",
    "JOB_TYPE": "PROVISION",
    "SERVICE_INSTANCE_ID": "429a35a0-dee8-4648-86f6-d1f8bc74ba52",
    "PROJECT_ID": "063a3158b1a34710b36ad208b9497d00",
    "ORDER_ID": "",
    "JOB_DESC": "provision/provMain",
    "EXECUTION_STATUS": "SUCCEEDED",
    "CREATED_BY": "585cc5c16cda4954b28d4eb460f06261"
  },
  "tasks": [
    {
      "MESSAGES": "{}",
      "TASK_ID": "Task-bc64b46c-1688-4787-b7ff-87cb53b346ea",
      "TASK_TYPE": "echo",
      "TASK_INDEX": 1,
      "TASK_NAME": "SEPARATOR_PREPARE_RESOURCES",
      "TASK_STATUS": "SUCCEEDED",
      "LAST_HEALTH_CHECK": "2020-02-04T04:01:03.344Z",
      "CREATED_AT": "2020-02-04T04:01:03.344Z",
      "OWNER_ID": "cas-mgr-69bb87d7c-wpqwb:30114"
    },
    {
      "MESSAGES": "{}",
      "TASK_ID": "Task-15ec252f-bca0-401e-ae5b-798a639df92e",
      "TASK_TYPE": "echo",
      "TASK_INDEX": 4,
      "TASK_NAME": "SEPARATOR_BUILDING",
      "TASK_STATUS": "SUCCEEDED",
      "LAST_HEALTH_CHECK": "2020-02-04T04:01:03.346Z",
      "CREATED_AT": "2020-02-04T04:01:03.345Z",
      "OWNER_ID": "cas-mgr-69bb87d7c-wpqwb:30114"
    },
    {
      "MESSAGES": "{}",
      "TASK_ID": "Task-52a5d1ae-2c54-46ec-8134-53e7b6a45fb7",
      "TASK_TYPE": "restful",
      "TASK_INDEX": 6,
      "TASK_NAME": "TN00091",
      "TASK_STATUS": "SUCCEEDED",
      "LAST_HEALTH_CHECK": "2020-02-04T04:01:08.924Z",
      "CREATED_AT": "2020-02-04T04:01:03.346Z",
      "OWNER_ID": "cas-mgr-69bb87d7c-wpqwb:30114"
    },
    {
      "MESSAGES": "{}",
      "TASK_ID": "Task-a149b585-d31b-4c72-a665-d820efcb0bf9",
      "TASK_TYPE": "restful",
      "TASK_INDEX": 7,
      "TASK_NAME": "TN00094",
      "TASK_STATUS": "SUCCEEDED",
      "LAST_HEALTH_CHECK": "2020-02-04T04:01:08.961Z",
      "CREATED_AT": "2020-02-04T04:01:08.925Z",
      "OWNER_ID": "cas-mgr-69bb87d7c-wpqwb:30114"
    },
    {
      "MESSAGES": ""
    }
  ]
}
```

```
    "TASK_ID": "Task-a72fe8d8-8cc1-42b5-a097-0cf9e29692a7",
    "TASK_TYPE": "restful",
    "TASK_INDEX": 9,
    "TASK_NAME": "TN00097",
    "TASK_STATUS": "SUCCEEDED",
    "LAST_HEALTH_CHECK": "2020-02-04T04:01:38.975Z",
    "CREATED_AT": "2020-02-04T04:01:08.962Z",
    "OWNER_ID": "cas-mgr-69bb87d7c-wpqbwb:30114"
  }
],
"task_count": 5
}
```

Step 3 Call the API for **Modifying Component Information Based on the Component ID** to upgrade the component. The following is a request example:

```
curl -k -H "Content-Type:application/json;charset=utf8" -H "X-Auth-Token:$token" -X PUT "https://{servicestage_endpoint}/v3/{project_id}/cas/applications/{application_id}/components/{component_id}" -d '{
  {
    "source": {
      "kind": "image",
      "storage": "swr",
      "url": "swr.ap-southeast-1.myhuaweicloud.com/demo/arm_tomcat8:v1"
    },
    "version": "2023.0717.17160",
    "refer_resources": [
      {
        "id": "1d3091a2-1ecf-11ee-ae69-0255ac1001ba",
        "type": "cce",
        "parameters": {
          "namespace": "default",
          "type": "ARM64"
        }
      }
    ],
    "limit_cpu": 0.25,
    "limit_memory": 0.5,
    "request_cpu": 0.25,
    "request_memory": 0.5,
    "mesher": {},
    "affinity": {},
    "anti_affinity": {},
    "liveness_probe": {},
    "readiness_probe": {},
    "command": {
      "command": [],
      "args": []
    },
    "envs": [
      {
        "name": "PAAS_PROJECT_ID",
        "inner": false,
        "value": "063a3158b1a34710b36ad208b9497d00"
      },
      {
        "name": "CAS_APP_ID",
        "inner": false,
        "value": "f3f6cb89-831d-4579-8bbc-664487889e38"
      },
      {
        "name": "PAAS_CLUSTER_ID",
        "inner": false,
        "value": "1d3091a2-1ecf-11ee-ae69-0255ac1001ba"
      },
      {
        "name": "CAS_APPLICATION_ID",
        "inner": false,
        "value": "2c458700-c8e0-4651-a73b-5219b70a6572"
      }
    ]
  }
}
```

```
"name": "CAS_INSTANCE_VERSION",
"inner": false,
"value": "2023.0717.17062"
},
{
  "name": "CAS_COMPONENT_ID",
  "inner": false,
  "value": "e1c3021b-df40-48e9-bd41-ef033fc77f2b"
},
{
  "name": "CAS_INSTANCE_NAME",
  "inner": false,
  "value": "etst1-test-arm-5xdhrd"
},
{
  "name": "CAS_COMPONENT_NAME",
  "inner": false,
  "value": "etst1"
},
{
  "name": "PAAS_NAMESPACE",
  "inner": false,
  "value": "default"
},
{
  "name": "PAAS_APP_NAME",
  "inner": false,
  "value": "etst1"
},
{
  "name": "PAAS_PROJECT_NAME",
  "inner": false,
  "value": "cn-north-7"
},
{
  "name": "CAS_INSTANCE_ID",
  "inner": false,
  "value": "f3f6cb89-831d-4579-8bbc-664487889e38"
},
{
  "name": "CAS_ENVIRONMENT_NAME",
  "inner": false,
  "value": "test_arm"
},
{
  "name": "AOM_DISCOVERY_RULE_VERSION",
  "inner": false,
  "value": "V1"
},
{
  "name": "CAS_ENVIRONMENT_ID",
  "inner": false,
  "value": "21a43958-f64f-4077-8ae3-3a03bf0111c7"
},
{
  "name": "testName",
  "inner": false,
  "value": "testValue"
},
{
  "name": "PAAS_REGION_NAME",
  "inner": false,
  "value": "cn-north-7"
},
{
  "name": "CAS_APPLICATION_NAME",
  "inner": false,
  "value": "app-test-application"
}
```



```
    },
    "deploy_strategy": {
      "type": "RollingRelease",
      "rolling_release": {
        "batches": "1"
      }
    },
    "id": "f3f6cb89-831d-4579-8bbc-664487889e38",
    "name": "etst1"
  }
}
```

Information similar to the following is displayed.

```
{
  "job_id": "JOB2209202b-617c-4591-b0c6-ef2521a943a2"
}
```

Record **job_id** in bold in the command output to replace **{job_id}** in subsequent examples.

- Step 4** Call the API for [Step 2](#), use the **job_id** returned in [Step 3](#) to query the job status, and wait until the job is complete.

- Step 5** Call the API for [Delivering a Component Task Based on the Component ID](#) to roll back the component. The following is a request example:

```
curl -k -H "Content-Type:application/json;charset=utf8" -H "X-Auth-Token:$token" -X POST "https://
{servicestage_endpoint}/v3/{project_id}/cas/applications/{application_id}/components/{component_id}/
action" -d '
{
  "action": "rollback_current"
}'
```

Information similar to the following is displayed.

```
{
  "job_id": "JOB66eb56db-062e-4462-ba95-8d3c63d5d462"
}
```

Record **job_id** in bold in the command output to replace **{job_id}** in subsequent examples.

- Step 6** Call the API for [Step 2](#), use the **job_id** returned in [Step 5](#) to query the job status, and wait until the job is complete.

- Step 7** Call the API for [Delivering a Component Task Based on the Component ID](#) to stop the component instance. The following is a request example:

```
curl -k -H "Content-Type:application/json;charset=utf8" -H "X-Auth-Token:$token" -X POST "https://
{servicestage_endpoint}/v3/{project_id}/cas/applications/{application_id}/components/{component_id}/
action" -d '
{
  "action": "stop"
}'
```

Information similar to the following is displayed.

```
{
  "job_id": "JOB2790bf9f-7d6e-473d-827f-1bb94e0cf10d"
}
```

Record **job_id** in bold in the command output to replace **{job_id}** in subsequent examples.

- Step 8** Call the API for [Step 2](#), use the **job_id** returned in [Step 7](#) to query the job status, and wait until the job is complete.

----End

8.4 Microservice Registry and Discovery

Scenarios

Microservice is defined by service. If a process provides a service, it is a microservice. Each service has its own service functions. APIs that are not restricted by languages are open to other systems (HTTP frequently used). Multiple microservices form an application.

The following describes how to call the APIs for [Creating Static Information for a Microservice](#), [Registering a Microservice Instance](#), and [Querying a Microservice Instance by Filter Criteria](#).

NOTE

The preceding microservice names are examples.

For details about how to call APIs, see [Calling APIs](#).

Prerequisites

- You have obtained the IAM token by calling the API for [obtaining a user token](#), and set the token to an environment variable, that is, **export Token={token}**.
- You have obtained the CSE endpoint to replace **{cse_endpoint}** in the following procedure. For details, see [Regions and Endpoints](#).

Procedure

- Step 1** Call the API for [Creating Static Information for a Microservice](#) to register the my-provider microservice. The following is a request example:

```
curl -k -H "x-domain-name:default" -H "x-auth-token:$Token" -XPOST "https://{cse_endpoint}/v4/default/registry/microservices" -d '{
  "service": {
    "serviceName": "my-provider",
    "appId": "default",
    "version": "1.0.0",
    "description": "test",
    "level": "MIDDLE",
    "status": "UP"
  }
}'
```

Information similar to the following is displayed.

```
{"serviceId":"918282e8562dc5fdc9a8dcd4d1baabb492190aa4"}
```

Record **serviceId** in bold to replace **{providerServiceId}** in subsequent examples.

- Step 2** Call the API for [Registering a Microservice Instance](#) to register the instance of the my-provider microservice.

The validity period of the instance is 1 hour. The instance is automatically brought offline when the validity period expires. Assume that the listening address of the my-provider microservice instance is **127.0.0.1:8080**. The following is a request example:

```
curl -k -H "x-domain-name:default" -H "x-auth-token:$Token" -XPOST "https://{cse_endpoint}/v4/default/registry/microservices/{providerServiceId}/instances" -d '{
  "instance": {
    "hostName": "test",
    "endpoints": [
      "rest:127.0.0.1:8080"
    ],
    "status": "UP",
    "healthCheck": {
      "mode": "push",
      "interval": 900,
      "times": 3
    }
  }
}'
```

Information similar to the following is displayed.

```
{"instanceId":"2be605a095ed11eabcbe0255ac100fa3"}
```

- Step 3** Call the API for [Creating Static Information for a Microservice](#) to register the my-consumer microservice. The following is a request example:

```
curl -k -H "x-domain-name:default" -H "x-auth-token:$Token" -XPOST "https://{cse_endpoint}/v4/default/registry/microservices" -d '{
  "service": {
    "serviceName": "my-consumer",
    "appld": "default",
    "version": "1.0.0",
    "description": "test",
    "level": "MIDDLE",
    "status": "UP"
  }
}'
```

Information similar to the following is displayed.

```
{"serviceId":"9db248934c31fc754d6e922b48ede4a5c004d3c1"}
```

Record **serviceId** in bold to replace **{consumerServiceId}** in subsequent examples.

- Step 4** Call the API for [Querying a Microservice Instance by Filter Criteria](#). The my-consumer microservice queries information about the my-provider microservice instance from the service center based on its own **serviceId**. The following is a request example:

```
curl -k -H "x-domain-name:default" -H "x-auth-token:$Token" -H "X-ConsumerId:{consumerServiceId}" -XGET "https://{cse_endpoint}/v4/default/registry/instances?appld=default&serviceName=my-provider&version=0.0.0%2B"
```

Information similar to the following is displayed.

```
{
  "instances": [
    {
      "instanceId": "2be605a095ed11eabcbe0255ac100fa3",
      "serviceId": "918282e8562dc5fdc9a8dcd4d1baabb492190aa4",
      "endpoints": [
        "rest:127.0.0.1:8080"
      ],
      "hostName": "test",
      "status": "UP",
      "healthCheck": {
        "mode": "push",
        "interval": 150,
        "times": 3
      },
      "timestamp": "1589465646",
      "modTimestamp": "1589465646",
      "version": "1.0.0"
    }
  ]
}
```

```
}  
]  
}
```

In actual practice, the my-consumer microservice can obtain the address of the my-provider microservice instance from the **endpoint** field in the instance query result and initiate service invoking.

Alternatively, log in to ServiceStage, click **Console** of a microservice engine, and choose **Service Catalog** to view the service registry information.

----End

9 Data Structure

9.1 Parameter Description

9.1.1 HealthCheck

Table 9-1 Parameters

Parameter	Mandatory	Type	Description
mode	Yes	String	Check mode. Value: push or pull.
port	No	Integer	Port number. The value can contain up to 65,536 characters. Regular expression: <code>^[0-9]*\$</code> .
interval	Yes	Integer	Heartbeat interval. Unit: s. If the value is less than 5s, the registration is performed at an interval of 5s. The value can contain up to 4,294,967,296 characters. Regular expression: <code>^[0-9]+\$</code> .
times	Yes	Integer	Maximum number of request attempts. The value can contain up to 4,294,967,296 characters. Regular expression: <code>^[0-9]+\$</code> .

9.1.2 MicroServiceInstance

Table 9-2 Parameters

Parameter	Mandatory	Type	Description
instanceId	No	String	Instance ID, which must be unique. The instance ID is generated by the service center.
serviceId	No	String	Microservice ID, which must be unique. The microservice ID in the URL will be used.
version	No	String	Microservice version.
hostName	Yes	String	PC name, which can be a domain name or IP address.
endpoints	No	Array of strings	Example: rest:127.0.0.1:8080.
status	Yes	String	Instance status. • UP • DOWN • STARTING • OUTOFSERVICE
properties	No	Map<String, String>	Extended attribute. You can customize a key and value.
healthCheck	No	HealthCheck	Health check mode. Value: push or pull. interval indicates the pull interval (unit: s), and times indicates the maximum request attempts. When in the pull mode, you need to define port to ensure that the registration center actively connects to the port. The default value of healthCheck is push , that of interval is 30 , and that of times is 3 .
timestamp	No	String	Time when an instance is created, which is automatically generated.
modTimestamp	No	String	Update time.
dataCenterInfo	No	DataCenterInfo	Time zone.

9.1.3 HeartbeatSetElement

Table 9-3 Parameters

Parameter	Mandatory	Type	Description
serviceld	Yes	String	Microservice ID. The value contains 1 to 64 characters. Regular expression: ^.*\$.
instanceId	Yes	String	Microservice instance ID. The value contains 1 to 64 characters. Regular expression: ^[A-Za-z0-9_-.]*\$.

9.1.4 InstanceHbRst

Table 9-4 Parameters

Parameter	Mandatory	Type	Description
serviceld	No	String	Microservice ID.
instanceId	No	String	Microservice instance ID.
errorMessage	No	String	Error information. If the operation is successful, the value is empty. If the operation fails, the error information is returned.

9.1.5 DelServicesResponse

Table 9-5 Parameter

Parameter	Mandatory	Type	Description
serviceld	No	String	Microservice ID.

9.1.6 MetricData

Table 9-6 Parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Service name.
appld	Yes	String	Application ID.

Parameter	Mandatory	Type	Description
serviceId	No	String	Service ID.
version	Yes	String	Service version.
instanceId	No	String	Instance ID.
instance	Yes	String	Instance name.
thread	No	String	Number of threads.
cpu	No	String	CPU usage.
memory	No	String	Memory.
interfaces	No	Table 9-7	API data details.

 NOTE

Parameters **serviceId** and **instanceId** must be specified or left blank at the same time.

Table 9-7 items parameters

Parameter	Mandatory	Type	Description
items	No	FunctionMetricInfo	API-level aggregation metric data.

9.1.7 FunctionMetricInfo

Table 9-8 Parameters

Parameter	Mandatory	Type	Description
AggregateMetricInfo	No	Object	Aggregation metric data.
l995	No	Integer	99.5% requests have a value smaller than the value of this parameter.
l99	No	Integer	99% requests have a value smaller than the value of this parameter.
l90	No	Integer	90% requests have a value smaller than the value of this parameter.
l75	No	Integer	75% requests have a value smaller than the value of this parameter.
l50	No	Integer	50% requests have a value smaller than the value of this parameter.

Parameter	Mandatory	Type	Description
l25	No	Integer	25% requests have a value smaller than the value of this parameter.
l5	No	Integer	5% requests have a value smaller than the value of this parameter.

9.1.8 InstanceMetricInfo

Table 9-9 Parameters

Parameter	Mandatory	Type	Description
AggregateMetricInfo	No	Object	Aggregation metric data.
thread	No	Integer	Number of threads.
cpu	No	number	CPU usage.
memory	No	Object	Memory information.

9.2 Common Request Parameters

9.2.1 MicroService

Table 9-10 Parameters

Parameter	Mandatory	Type	Description
serviceId	No	String	Microservice ID, which must be unique. The value contains 1 to 64 characters. Regular expression: $\wedge.*\$$.
environment	No	String	Microservice environment. Value: Empty value, development, testing, acceptance, or production.
appId	No	String	Application ID, which must be unique. The value contains 1 to 160 characters. Regular expression: $\wedge[a-zA-Z0-9]*\wedge[a-zA-Z0-9][a-zA-Z0-9\_\.]*[a-zA-Z0-9]\$$.

Parameter	Mandatory	Type	Description
serviceName	Yes	String	Microservice name, which must be unique in an application. The value contains 1 to 128 characters. Regular expression: <code>^[a-zA-Z0-9]*\$ ^[a-zA-Z0-9][a-zA-Z0-9_\-\.]*[a-zA-Z0-9]\$</code> .
version	No	String	Microservice version. The value contains 1 to 64 characters. Regular expression: <code>^[0-9]*\$ ^[0-9]+(\.[0-9]+)*\$</code> .
description	No	String	Description of a microservice. The value can contain up to 256 characters.
level	No	String	Microservice level. • FRONT • MIDDLE • BACK
timestamp	No	String	Microservice registration time.
schemas	No	Array of String	Foreign key ID of a microservice access schema. The array length supports up to 100 schemas.
status	No	String	Microservice status. Value: UP or DOWN. Default value: UP.
paths	No	ServicePath	Service path.
framework	No	Framework	Microservice development framework.
registerBy	No	String	Microservice registration mode. • SDK • PLATFORM • SIDECAR • UNKNOWN
modTimestamp	No	String	Latest modification time (UTC).
properties	No	Map<String, String>	Extended attribute. You can customize a key and value.
alias	No	String	Microservice alias. The value contains 1 to 128 characters. Regular expression: <code>^[a-zA-Z0-9_\-\.]*\$</code> .

Table 9-11 ServicePath parameters

Parameter	Mandatory	Type	Description
Path	No	String	Route address. The value contains 1 to 160 bytes, including digits, letters, and the following characters:.,?'\/+& %\$#=#~_-@{}
Property	No	Properties	Extended attribute.

Table 9-12 Framework parameters

Parameter	Mandatory	Type	Description
name	No	String	Microservice development framework. Default value: UNKNOWN.
version	No	String	Version of the microservice development framework. The value contains 1 to 64 bytes.

9.2.2 Properties

Table 9-13 Parameters

Parameter	Mandatory	Type	Description
< * >	No	String	Extended attribute. You can customize a key and value. The value must be at least 1 byte long.

9.2.3 DependencyMicroService

Table 9-14 Parameters

Parameter	Mandatory	Type	Description
appld	Yes	String	Application ID, which must be unique. The value contains 1 to 160 characters. Regular expression: ^[a-zA-Z0-9]*\$ ^[a-zA-Z0-9][a-zA-Z0-9_\.]*[a-zA-Z0-9]\$.

Parameter	Mandatory	Type	Description
serviceName	Yes	String	Microservice name. For a provider microservice, the value can be *, indicating all services of a tenant. If the value is *, both appld and version can be omitted. For a consumer microservice, the value cannot be *. The value contains 1 to 128 characters. Regular expression: <code>^[a-zA-Z0-9]*\$ ^[a-zA-Z0-9][a-zA-Z0-9_\-\.]*[a-zA-Z0-9]\$</code> .
environment	Yes	String	Microservice environment. • Empty value • development • testing • acceptance • production

9.2.4 Rule

Table 9-15 Parameters

Parameter	Mandatory	Type	Description
ruleId	No	String	Blacklist or whitelist ID.
ruleType	Yes	String	Type. • WHITE • BLACK
attribute	Yes	String	If the value starts with tag_XXX , the attributes are filtered by Tag . Otherwise, the attributes are filtered by ServiceId , Appld , ServiceName , Version , Description , Level , or Status .
pattern	Yes	String	Matching rule. The value is a regular expression containing 1 to 64 characters.
timestamp	No	String	Rule creation time. This parameter is used only when you obtain a rule.
description	No	String	Description of a rule. The value can contain up to 256 characters.

Parameter	Mandatory	Type	Description
modificationTime	No	String	Latest modification time (UTC).

9.2.5 AddOrUpdateRule

Table 9-16 Parameters

Parameter	Mandatory	Type	Description
ruleType	No	String	Type. • WHITE • BLACK
attribute	No	String	If the value starts with tag_XXX , the attributes are filtered by Tag . Otherwise, the attributes are filtered by ServiceId , AppId , ServiceName , Version , Description , Level , or Status .
pattern	No	String	Matching rule. The value is a regular expression containing 1 to 64 characters.
description	No	String	Description of a rule. The value can contain up to 256 characters.

9.2.6 DataCenterInfo

Table 9-17 Parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Data center name. The value must be 1 to 128 bytes long. Only digits, letters, underscores (_), hyphens (-), and periods (.) are allowed.
region	Yes	String	Region where the data center is located. The value must be 1 to 128 bytes long. Only digits, letters, underscores (_), hyphens (-), and periods (.) are allowed.

Parameter	Mandator y	Type	Description
availableZone	Yes	String	AZ where the data center is located. The value must be 1 to 128 bytes long. Only digits, letters, underscores (_), hyphens (-), and periods (.) are allowed.

9.2.7 Schema

Table 9-18 Parameters

Parameter	Mandator y	Type	Description
schemald	No	String	Microservice schema ID. The value must be 1 to 160 bytes long. Only digits, letters, underscores (_), hyphens (-), and periods (.) are allowed.
schema	No	String	Microservice schema content. The value must be 1 to 2048 bytes long.
summary	No	String	Microservice schema summary. The value can contain up to 128 bytes. Only digits and letters are allowed.

9.3 Common Response Parameters

9.3.1 WatchMicroServiceKey

Table 9-19 Parameters

Parameter	Mand atory	Type	Description
appld	No	String	Application ID, which must be unique. The value contains 1 to 160 characters. Regular expression: <code>^[a-zA-Z0-9]*\$ ^[a-zA-Z0-9][a-zA-Z0-9_\-\.]*[a-zA-Z0-9]\$</code> .

Parameter	Mandatory	Type	Description
serviceName	No	String	Microservice name. For a provider microservice, the value can be *, indicating all services of a tenant. If the value is *, both appId and version can be omitted. For a consumer microservice, the value cannot be *.
version	No	String	Microservice version. For a provider microservice, the value can be the version+, fixed version, or latest version. For example, 1.0.1+ indicates version 1.0.1 or later. For a consumer microservice, the value can only be a fixed version.

9.3.2 SuccessdResponse

Table 9-20 Parameters

Parameter	Mandatory	Type	Description
Result	Yes	String	Returned result.

9.3.3 ServiceInfo

Table 9-21 Parameters

Parameter	Mandatory	Type	Description
serviceId	Yes	String	Microservice ID, which must be unique. The value is a UUID. Regular expression: $\wedge.*\$$.
appId	Yes	String	Application ID, which must be unique. The value contains 1 to 160 characters. Regular expression: $\wedge[a-zA-Z0-9]*\$ ^\wedge[a-zA-Z0-9][a-zA-Z0-9\_\\-]*[a-zA-Z0-9]\$$.
serviceName	Yes	String	Microservice name, which must be unique in an application. The value contains 1 to 128 characters. Regular expression: $\wedge[a-zA-Z0-9]*\$ ^\wedge[a-zA-Z0-9][a-zA-Z0-9\_\\-]*[a-zA-Z0-9]\$$.

Parameter	Mandatory	Type	Description
version	Yes	String	Microservice version. The value contains 1 to 64 characters. Regular expression: <code>^[0-9]*\$ ^[0-9]+(\.[0-9]+)*\$</code> .
level	Yes	String	Microservice level. • FRONT • MIDDLE • BACK
status	Yes	String	Service status. • UP • DOWN
timestamp	Yes	String	Timestamp.

9.3.4 AggregateMetricInfo

Table 9-22 Parameters

Parameter	Mandatory	Type	Description
time	No	Integer	Report time.
name	No	String	API name.
qps	No	Number	Current Query Per Second (QPS).
latency	No	Number	Average latency.
rate	No	Integer	Success rate.
total	No	Integer	Total number of requests.
isCircuitBreakerOpen	No	Boolean	Whether the circuit breaker is enabled.
failure	No	Integer	Total number of failed requests.
shortCircuited	No	Integer	Total number of short circuits.
semaphoreRejected	No	Integer	Total number of rejected semaphores.
threadPoolRejected	No	Integer	Total number of rejected threads.
countTimeout	No	Integer	Total number of timeout requests.

9.3.5 ServiceDetail

Table 9-23 Parameters

Parameter	Mandatory	Type	Description
microService	No	MicroService	Service information.
instances	No	MicroServiceInstance	Instance information.
schemaInfos	No	Schema	Microservice API details.
rules	No	Rule	Blacklist and whitelist information.
providers	No	MicroService	All provider information.
consumers	No	MicroService	All consumer information.
tags	No	Tags	All tag information about a microservice.
microServiceVersions	No	Array. Each item in the array is a string.	All microservice versions.

Table 9-24 tags parameters

Parameter	Mandatory	Type	Description
tags	No	Properties	Request struct of an extended attribute of a microservice.

9.3.6 TenantProject

Table 9-25 TenantProject parameters

Parameter	Loca ted In	Type	Description
tenant	body	String	Tenant name.
projectId	body	String	Project ID.

9.3.7 Error

Table 9-26 Parameters

Parameter	Mandat ory	Type	Description
errorCode	Yes	String	Error code.
errorMessage	Yes	String	Error message.
detail	No	String	Location details.

10 Permissions and Supported Actions

10.1 Introduction

You can use Identity and Access Management (IAM) for fine-grained permissions management of your ServiceStage. If your Huawei Cloud account does not need individual IAM users, you can skip this section.

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

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

NOTE

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

Your account has all the permissions required to call all APIs, but IAM users under your account must be assigned the required permissions. The required permissions are determined by the actions supported by the API. Only users with the policies allowing for those actions can call the API successfully. For example, if an IAM user wants to query metrics using an API, the user must have been granted permissions that allow the **servicestage:app:get** action.

Supported Actions

ServiceStage provides system-defined policies that can be directly used in IAM. You can also create custom policies to supplement system-defined policies for more refined access control. Operations supported by policies are specific to APIs. The following are common concepts related to policies:

- Permissions: Defined by actions in a custom policy.

- APIs: REST APIs that can be called by a user who has been granted specific permissions.
- Actions: specific operations that are allowed or denied.
- IAM projects/Enterprise projects: the authorization scope of a custom policy. A custom policy can be applied to IAM projects or enterprise projects or both. Policies that contain actions for both IAM and enterprise projects can be used and applied for both IAM and Enterprise Management. Policies that contain actions only for IAM projects can be used and applied to IAM only. Administrators can check whether an action supports IAM projects or enterprise projects in the action list. For details about the differences between IAM and enterprise projects, see [What Are the Differences Between IAM and Enterprise Management?](#)

ServiceStage supports the following actions in custom policies:

Table 10-1 ServiceStage actions

Permissions	APIs	Actions	IAM Projects	Enterprise Projects
Create an application	POST /v3/{project_id}/cas/applications	servicestage:app:create	√	√
Delete an application	DELETE /v3/{project_id}/cas/applications/{application_id}	servicestage:app:delete	√	√
Update an application	PUT /v3/{project_id}/cas/applications/{application_id}	servicestage:app:modify	√	√
Query applications	GET /v3/{project_id}/cas/applications	servicestage:app:list	√	√
Query application information	GET /v3/{project_id}/cas/applications/{application_id}	servicestage:app:get	√	√
Modify a project	-	servicestage:project:modify	√	√
Create a project	-	servicestage:project:create	√	√
Approve an application	-	servicestage:app:approve	√	√

Permissions	APIs	Actions	IA M Proj ects	Enterp rise Projec ts
View pipelines	-	servicesta ge:pipelin e:list	√	√
Modify a build job	-	servicesta ge:assem bling:mod ify	√	√
Approve a pipeline	-	servicesta ge:pipelin e:review	√	√
Execute a pipeline	-	servicesta ge:pipelin e:execute	√	√
View the build information	-	servicesta ge:project :get	√	√
Delete a build job	-	servicesta ge:assem bling:dele te	√	√
Delete a pipeline	-	servicesta ge:pipelin e:delete	√	√
Delete a project	-	servicesta ge:project :delete	√	√
Modify a pipeline	-	servicesta ge:pipelin e:modify	√	√
Create a build job	-	servicesta ge:assem bling:crea te	√	√
View build jobs	-	servicesta ge:assem bling:list	√	√
View the build information	-	servicesta ge:assem bling:get	√	√

Permissions	APIs	Actions	IA M Proj ects	Enterp rise Projec ts
View projects	-	servicesta ge:project :list	√	√
View the pipeline information	-	servicesta ge:pipelin e:get	√	√
Create a pipeline	-	servicesta ge:pipelin e:create	√	√

11 Appendix

11.1 Status Codes

Table 11-1 describes the status codes.

Table 11-1 Status codes

Status Code	Message	Description
200	-	OK
304	Not Modified	The requested resource is not modified.
400	Bad Request	Invalid request. The client should not repeat the request without modifications.
401	Unauthorized	The authentication information is incorrect or invalid.
403	Forbidden	The server refused the request from the client.
404	Not Found	The requested resource cannot be found. The client should not repeat the request without modifications.
422	Unprocessable Entity	The request was well-formed but was unable to be followed due to semantic errors.
500	Internal Server Error	The server is able to receive the request but it could not understand the request.

11.2 ServiceStage Error Codes

If an error occurs after a request is sent to the system, a response containing an error code will be returned.

If an error code starting with **APIGW** is returned after you call an API, rectify the fault by referring to [Error Codes](#).

The following lists the error codes.

Status Code	Error Code	Error Message	Description	Measure
400	SVCSTG.0010 0400	Bad Request	Invalid parameter.	Enter a valid value and try again.
400	SVCSTG.0010 0402	Bad Request	The component has been deployed. Uninstall it first.	Uninstall the component and try again.
400	SVCSTG.0010 0414	Bad Request	The job ID cannot be empty.	Specify a job ID.
400	SVCSTG.0010 0418	Bad Request	The component instance operation is not defined.	Specify a defined operation.
400	SVCSTG.0010 0420	Bad Request	The runtime system is not supported.	Specify a supported runtime system.
400	SVCSTG.0010 0422	Bad Request	The basic resource cannot be empty.	Configure basic resources.
400	SVCSTG.0010 0425	Bad Request	The software package or image of the instance is empty.	The artifact parameter in the packet cannot be empty.
400	SVCSTG.0010 0449	Bad Request	The environment name already exists.	Use another environment name.

Status Code	Error Code	Error Message	Description	Measure
400	SVCSTG.REPO.0400	Bad Request	Invalid parameter.	Enter a valid value and try again.
400	SVCSTG.REPO.0401	Bad Request	The repository is not implemented.	Select another repository.
400	SVCSTG.REPO.0402	Bad Request	Invalid OAuth code or state.	Perform OAuth 2.0 authorization again to obtain a correct authentication code and random string.
400	SVCSTG.REPO.0403	Bad Request	The resource already exists.	Check whether the resource already exists.
400	SVCSTG.REPO.0450	Bad Request	No repository authorization.	Check whether the authorization name in the request header is correct.
400	SVCSTG.REPO.0451	Bad Request	Invalid repository authorization.	Perform authorization again.
400	SVCSTG.REPO.0452	Bad Request	Invalid clone URL.	Enter a correct clone URL. Example: https://user-name@github.com/user-name/demo.git .
404	SVCSTG.00100403	Not Found	Application not found.	Specify an existing application.

Status Code	Error Code	Error Message	Description	Measure
404	SVCSTG.0010 0404	Not Found	Component not found.	Specify an existing component.
404	SVCSTG.0010 0408	Not Found	Request not found.	Specify an existing request.
404	SVCSTG.0010 0411	Not Found	Environment not found.	Specify an existing environment.
404	SVCSTG.0010 0412	Not Found	Instance not found.	Specify an existing instance.
404	SVCSTG.0010 0424	Not Found	ELB instance not found.	Specify an existing ELB instance.
404	SVCSTG.0010 0429	Not Found	Build job not found.	Specify an existing build job.
404	SVCSTG.0010 0433	Not Found	Release task not found.	Specify an existing release task.
404	SVCSTG.0010 0436	Not Found	Instance snapshot not found.	Specify an existing instance snapshot.
404	SVCSTG.0010 0438	Not Found	Template image not found.	Specify an existing template image.
404	SVCSTG.0010 0441	Not Found	The component instance version number has been used in the environment.	Use another version number.
405	SVCSTG.0010 0421	Method Not Allowed	The account quota has been suspended.	Restore the account.

Status Code	Error Code	Error Message	Description	Measure
405	SVCSTG.0010 0427	Method Not Allowed	No permission.	Perform authorization again.
408	SVCSTG.0010 0428	Request Timeout	The system processing timed out.	The system is busy. Try again later.
409	SVCSTG.0010 0401	Conflict	The application is not empty.	Delete all components of the application and then delete the application.
409	SVCSTG.0010 0413	Conflict	The instance name already exists.	Use another instance name.
409	SVCSTG.0010 0419	Conflict	The operation cannot be performed in the current state.	The operation is being processed. Try again later.
409	SVCSTG.0010 0423	Conflict	The component instance cannot be deleted.	The operation is being processed. Try again later.
409	SVCSTG.0010 0426	Conflict	The environment is in use.	Delete the instances deployed in the environment and then delete the environment.
409	SVCSTG.0010 0432	Conflict	The component is already in the release task.	Wait until the current task is complete and try again.
409	SVCSTG.0010 0435	Conflict	The release task cannot be deleted.	The operation is being processed. Try again later.

Status Code	Error Code	Error Message	Description	Measure
409	SVCSTG.0010 0439	Conflict	The ELB instance has been used in the component.	Use another ELB instance.
500	SVCSTG.0010 0500	Internal Server Error	Internal server error.	The system is busy. Try again later.
500	SVCSTG.0010 0501	Internal Server Error	Remote server error.	The system is busy. Try again later.
500	SVCSTG.REPO. 0500	Internal Server Error	Internal server error.	Network error. Try again later.
500	SVCSTG.REPO. 0501	Internal Server Error	Remote repository error.	Check whether the request parameter is correct or the remote Git repository file has been modified.

11.3 CSE Error Codes

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

Table 11-2 Engine management error codes

Status Code	Error Code	Error Message	Description	Measure
400	SVCSTG.0050 1134	Bad Request	Invalid engine type.	The engine type field is the specType field in the request query. Set it to a valid value (for example, CSE2) and try again. For example, to query the specifications supported by CSE, set specType to CSE2 .
400	SVCSTG.0050 1111	Bad Request	Duplicate engine instance name.	Change the engine name.
400	SVCSTG.0050 1112	Bad Request	Insufficient engine instance quota.	Contact technical support engineers to increase the engine instance quota or delete unnecessary engine instances.
400	SVCSTG.0050 1133	Bad Request	Too many engines at the site.	Contact technical support engineers.

Status Code	Error Code	Error Message	Description	Measure
400	SVCSTG.0050 1103	Bad Request	Empty or invalid billing mode.	Input the billing mode parameter of the engine and check whether the user can create such an engine.
400	SVCSTG.0050 1104	Bad Request	Incorrect engine instance flavor format.	Invalid flavor. Enter a valid flavor.
400	SVCSTG.0050 1105	Bad Request	Incorrect AZ format.	Specify a valid AZ in the request parameter. The AZ field is a string array and cannot be empty. The number of AZ name levels ranges from 1 to 3.
400	SVCSTG.0050 1117	Bad Request	Engine cannot be deleted.	Wait until the engine state changes to Available, Unavailable, or Creation failed, and then try again.
400	SVCSTG.0050 1152	Bad Request	Incorrect task ID format.	Use the job ID returned by the APIs for creating and deleting an engine.
400	SVCSTG.0050 1153	Bad Request	Failed to find the task.	Use the job ID returned by the APIs for creating and deleting an engine.

Status Code	Error Code	Error Message	Description	Measure
400	SVCSTG.0050 1149	Bad Request	Abnormal engine node.	Try again later or contact technical support engineers.
400	SVCSTG.0050 1116	Bad Request	Failed to find the engine instance.	Check whether the input engine ID is correct. Use the username and password of the tenant to log in to CSE and check whether the current tenant has an engine with the corresponding ID.
400	SVCSTG.0050 1201	Bad Request	Incorrect format of the peer engine instance name.	The peer engine instance name must contain 3 to 24 characters, including letters, digits, and hyphens (-), and must start with a letter but cannot end with a hyphen (-). The name cannot be default and must be different from the current engine instance name.

Status Code	Error Code	Error Message	Description	Measure
400	SVCSTG.0050 1202	Bad Request	The peer engine must be an HA engine.	The engine must be an HA engine. Enter the engine flavor again.
400	SVCSTG.0050 1203	Bad Request	The peer engine must be in a single AZ.	The AZs must be single AZs. Enter the AZ list again.
500	SVCSTG.0050 0400	Internal Server Error	Incorrect internal request parameter of the service.	Check the error message. If the parameter is invalid, check whether the input argument is correct. For other errors, contact technical support engineers.

Table 11-3 Dynamic configuration error codes

Status Code	Error Code	Error Message	Description	Measure
400	SVCSTG.0040 0649	Bad Request	Too long key or value.	The configuration item contains 1 to 2048 characters, including digits, letters, and special characters (.-_[:]). The value can contain 1 to 131072 characters, including digits, letters, and special characters ('~!@#\$%^&*()_ \{} :"<>?'`-[] ,./=).
400	SVCSTG.0040 0650	Bad Request	Invalid configuration file.	Upload a valid file.
400	SVCSTG.0040 0651	Bad Request	The file is empty or in an incorrect format.	The uploaded file is empty or not in JSON format.
400	SVCSTG.0040 0652	Bad Request	The key or value is empty or is not a string.	The uploaded key or value is null or is not a string.
400	SVCSTG.0040 0653	Bad Request	Invalid label or body.	Enter a valid body or label in the query parameter.
400	SVCSTG.0040 0100	Bad Request	Invalid parameter.	Change parameters as prompted.
404	SVCSTG.0040 0106	Not Found	The resource does not exist.	Enter valid search criteria.

Status Code	Error Code	Error Message	Description	Measure
409	SVCSTG.00400107	Conflict	The resource already exists.	Do not create the same record.
500	SVCSTG.00300605	Failed to connect the configuration center to ETCD.	Internal Server Error	Contact technical support engineers.
400	SVCSTG.00300401	Invalid token	Unauthorized	Enter a correct token.

11.4 ServiceComb Error Codes

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

Cat ego ry	Status Code	Error Code	Error Message	Description	Measure
Co mm on erro r cod es	400	400001	Bad Request	Invalid parameter.	Change parameters as prompted.
	404	404001	Not Found	The resource does not exist.	Enter valid search criteria.
	409	409001	Conflict	The resource already exists.	Do not create the same record.
	500	500003	Internal Server Error	Internal server error.	Contact technical support engineers.
Mic rose rvic es	400	400002	Bad Request	The service is unhealthy.	Try again later or contact technical support engineers.

Cat ego ry	Status Code	Error Code	Error Message	Description	Measure
		400010	Bad Request	The service already exists.	Modify the service ID or microservice description in the body of the request for creating a microservice.
		400011	Bad Request	No available backend instance.	Try again later or contact technical support engineers.
		400012	Bad Request	The service does not exist.	Enter a valid service ID.
		400013	Bad Request	The microservice cannot be deleted because instances have been deployed.	Take the instance offline and then delete the microservice. Alternatively, forcibly delete the microservice by setting the query parameter force to true in the URL.
		400014	Bad Request	The schema ID does not exist.	Enter a valid schema ID.
		400015	Bad Request	The schema cannot be modified.	The schema has been registered and cannot be modified.
		400016	Bad Request	The schema does not exist.	Register the schema first.
		400017	Bad Request	The instance does not exist.	Enter a valid instance ID.

Cat ego ry	Status Code	Error Code	Error Message	Description	Measure
		400018	Bad Request	The tag does not exist.	This error code is usually generated for a query API, indicating that the tag does not exist. The service performs subsequent processing based on the returned value.
		400019	Bad Request	The rule already exists.	The rule is repeatedly created. Generally, this error can be ignored.
		400020	Bad Request	Blacklist and whitelist error.	Modify the parameter value based on the error message.
		400021	Bad Request	The rule cannot be modified.	You can modify the microservice information only after changing the version number.
		400022	Bad Request	The rule does not exist.	This error code is usually generated for a query API, indicating that the rule does not exist. The service performs subsequent processing based on the returned value.

Cat ego ry	Status Code	Error Code	Error Message	Description	Measure
		400023	Bad Request	The microservice cannot be deleted because it is the dependent service of other microservices.	You can forcibly delete microservices by setting the query parameter force to true in the URL.
		400024	Bad Request	Insufficient permissions.	Use a proper account to perform operations.
		400025	Bad Request	The port already exists.	Check whether the port is occupied by another instance.
		400026	Bad Request	The microservice instance does not exist.	Enter a correct version number or range.
		400100	Bad Request	Insufficient quota.	The quotas of resources such as microservices, instances, or schemas are insufficient. Delete some resources and create again.

Cat ego ry	Status Code	Error Code	Error Message	Description	Measure
	401	401204	Unauthorize d	Unauthorize d.	This parameter is mandatory if security authentication is enabled for the microservice engine. Otherwise, this parameter is not required. The token of the microservice engine with security authentication enabled is in the following format: Authorization:Bearer {Token} For details about how to obtain the token, see Obtaining the User Token of an Exclusive Microservice Engine.
		401201	Unauthorize d	Unauthorize d.	Enter a valid authorization.
	403	403001	Forbidden	Insufficient permissions.	Use a proper account to perform operations.
	500	500011	Internal Server Error	The registry service is unavailable.	Contact technical support engineers.
		500101	Internal Server Error	No quota.	Try again later or contact technical support engineers.

Cat ego ry	Status Code	Error Code	Error Message	Description	Measure
		500605	N/A	Failed to connect to etcd of the configuration center.	Try again later or contact technical support engineers.
Aut hen tica tion	401	401202	Unauthorize d	Incorrect account name or password.	Enter the correct account name and password.

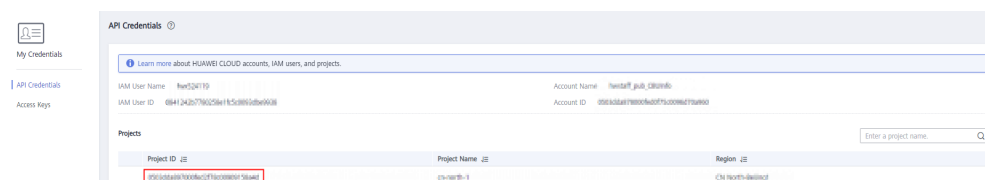
11.5 Obtaining a Project ID

Obtaining a Project ID from the Console

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

- Step 1** Log in to ServiceStage.
- Step 2** Move the cursor to the username in the upper right corner and select **My Credentials** from the drop-down list.
- Step 3** On the **Projects** area, obtain the required project ID in the project list.

Figure 11-1 Viewing project IDs



----End

Obtaining a Project ID by Calling an API

A project ID can also be obtained by calling a specific API. For details, see [Querying Project Information Based on the Specified Criteria](#).

The API used to obtain a project ID is **GET https://{Endpoint}/v3/projects/**, where *{Endpoint}* indicates the IAM endpoint. You can obtain the IAM endpoint from [Regions and Endpoints](#). For details on API calling authentication, see [Authentication](#).

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

```
{
  "projects": [
```

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

11.6 Obtaining an Account ID

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

Step 1 Log in to ServiceStage.

Step 2 Move the cursor to the username in the upper right corner and select **My Credentials** from the drop-down list.

View the account ID.

Figure 11-2 Viewing the account ID



----End

12 Historical APIs

12.1 Application Management V2 APIs

12.1.1 Meta

12.1.1.1 Obtaining All Supported Runtime Systems of Application Components

Function

This API is used to obtain all supported runtime systems of application components.

URI

GET /v2/{project_id}/cas/metadata/runtimes

Table 12-1 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .

Request

Table 12-2 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .

Response

Table 12-3 Response parameters

Parameter	Type	Description
runtimes	Array of objects	Runtime system parameters. See Table 12-4 .

Table 12-4 runtimes

Parameter	Type	Description
type_name	String	Type.
display_name	String	Display name.
container_default_port	Integer	Default container port.
type_desc	String	Type description.

Example Request

None

Example Response

```
{
  "runtimes": [
    {
      "type_name": "Java8",
      "display_name": "Java 8",
      "container_default_port": 8080,
```

```
        "type_desc": "Java 8 runtime server"
      }
    ]
  }
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

12.1.1.2 Obtaining All Supported Flavors of Application Resources

Function

This API is used to obtain all supported flavors of application resources.

URI

GET /v2/{project_id}/cas/metadata/flavors

Table 12-5 Path parameters

Parameter	Man dato ry	Type	Description
project_id	Yes	Strin g	Tenant's project ID. See Obtaining a Project ID .

Request

Table 12-6 Request header parameters

Parameter	Man dator y	Type	Description
Content-Type	Yes	Strin g	Message body type (format). Default value: application/json;charset=utf8.

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .

Response

Table 12-7 Response parameters

Parameter	Type	Description
flavors	Array of objects	Flavor parameters. See Table 12-8 .

Table 12-8 flavors

Parameter	Type	Description
flavor_id	String	Flavor ID.
storage_size	String	Storage size.
num_cpu	String	CPU limit.
num_cpu_init	String	Initial CPU value.
memory_size	String	Memory limit.
memory_size_init	String	Initial memory value.
label	String	Label.
custom	boolean	Whether resource specifications are customized.

Example

None

Example Response

```
{
  "flavors": [
    {
      "flavor_id": "MICRO-5G:0.5C:1G",
```

```
{
  "storage_size": "5G",
  "num_cpu": "500m",
  "num_cpu_init": "200m",
  "memory_size": "1Gi",
  "memory_size_init": "200Mi",
  "label": "Micro: 5G Storage, 0.5 CPU, 1G Memory",
  "custom": false
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

12.1.2 Environment

12.1.2.1 Creating an Environment

Function

This API is used to create an environment.

URI

POST /v2/{project_id}/cas/environments

Table 12-9 Path parameters

Parameter	Mandator y	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .

Request

Table 12-10 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .

Table 12-11 Request body parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Environment name. The value contains 2 to 64 characters consisting of letters, digits, hyphens (-), and underscores (_). It starts with a letter and ends with a letter or digit. Letters are case insensitive.
alias	No	String	Environment alias. The value can contain up to 64 characters.
deploy_mode	Yes	String	Environment type. • virtualmachine • container: Kubernetes • mixed: VM and Kubernetes
description	No	String	Environment description. The value can contain up to 128 characters.
enterprise_project_id	No	String	Enterprise project ID.

Parameter	Mandatory	Type	Description
charge_mode	No	String	Billing mode. Value: provided, on_demanded, and monthly. provided (default) indicates that provided resources are used and no fees are charged, on_demanded indicates the pay-per-use billing mode, and monthly indicates the monthly billing mode.
vpc_id	Yes	String	VPC ID.
base_resources	Yes	Array of objects	Basic resources. See Table 12-12 .
optional_resources	No	Array of objects	Optional resources. See Table 12-12 .

Table 12-12 resources

Parameter	Mandatory	Type	Description
id	Yes	String	Resource ID.
type	Yes	String	Basic resources: cce, as, and ecs. Optional resources: rds, dcs, elb, cse, and other services.
name	No	String	Resource name.

Response

Table 12-13 Response parameters

Parameter	Type	Description
id	String	Environment ID.
name	String	Environment name.
alias	String	Environment alias.
description	String	Environment description.
project_id	String	Project ID.
enterprise_project_id	String	Enterprise project ID.
charge_mode	String	Billing mode.

Parameter	Type	Description
deploy_mode	String	Environment type. • virtualmachine • container: Kubernetes • mixed: VM and Kubernetes
vpc_id	String	VPC ID.
base_resources	Array of objects	Basic resources. See Table 12-14 . The returned value is empty.
optional_resources	Array of objects	Optional resources. See Table 12-14 . The returned value is empty.
creator	String	Creator.
create_time	Integer	Creation time.
update_time	Integer	Update time.

Table 12-14 resources

Parameter	Type	Description
id	String	Resource ID.
name	String	Resource name.
type	String	Basic resources: cce, as, and ecs. Optional resources: rds, dcs, elb, cse, and other services.

Example Request

Create a VM environment **development-env**. The ID of the VPC where the environment is located is **29d55020-ae0e-4a18-871c-93e6976ee7bd**.

```
{
  "name": "development-env",
  "description": "",
  "charge_mode": "provided",
  "deploy_mode": "container",
  "vpc_id": "29d55020-ae0e-4a18-871c-93e6976ee7bd",
  "base_resources": [
    {
      "id": "b6862a62-d916-11e9-bdf1-0255ac101fd9",
      "type": "cce"
    }
  ],
  "optional_resources": [
    {
      "id": "default",
      "type": "cse"
    }
  ]
}
```



```
    ]  
  }  
}
```

Example Response

```
{  
  "id": "00078e9d-a61c-476e-ac63-a10c9cb2638e",  
  "name": "development-env",  
  "alias": null,  
  "description": "",  
  "project_id": "e7d2e9c589e5445e808a8ff0d1235aca",  
  "enterprise_project_id": "0",  
  "charge_mode": "provided",  
  "deploy_mode": "container",  
  "vpc_id": "29d55020-ae0e-4a18-871c-93e6976ee7bd",  
  "base_resources": [  
    {  
      "id": "b6862a62-d916-11e9-bdf1-0255ac101fd9",  
      "type": "cce"  
    }  
  ],  
  "optional_resources": [  
    {  
      "id": "default",  
      "type": "cse"  
    }  
  ],  
  "creator": "ss-test",  
  "create_time": 1610418873730,  
  "update_time": 1610418873730  
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

12.1.2.2 Obtaining All Environments

Function

This API is used to obtain all created environments.

URI

GET /v2/{project_id}/cas/environments

Table 12-15 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .

Table 12-16 Query parameters

Parameter	Mandatory	Type	Description
limit	No	Integer	Number of records to be displayed. The value is 1000 or ranges from 0 to 100. If the specified value is not within the range, value 10 is assigned. In the non-pagination scenario, the value is 1000 . In the pagination scenario, the value ranges from 0 to 100.
offset	No	Integer	Offset, which indicates the result after the Nth data record is queried.
order_by	No	String	Sorting field. By default, query results are sorted by creation time. Enumerated values: create_time, name, and update_time. If the transferred value is not within the supported enumerated value range, the default sorting field is used.
order	No	String	Sorting order. • desc (default) • asc

Request

Table 12-17 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .

Response

Table 12-18 Response parameters

Parameter	Type	Description
count	Integer	Total number of environments.
environments	Array of objects	Environment parameters. See Table 12-19 .

Table 12-19 environments

Parameter	Type	Description
id	String	Environment ID.
name	String	Environment name.
alias	String	Environment alias.
description	String	Environment description.
project_id	String	Project ID.
enterprise_project_id	String	Enterprise project ID.
charge_mode	String	Billing mode.
deploy_mode	String	Environment type. ● virtualmachine ● container: Kubernetes ● mixed: VM and Kubernetes
vpc_id	String	VPC ID.
base_resources	Array of objects	Basic resources. See Table 12-20 .

Parameter	Type	Description
optional_resources	Array of objects	Optional resources. See Table 12-20 .
creator	String	Creator.
create_time	Integer	Creation time.
update_time	Integer	Update time.

Table 12-20 resources

Parameter	Type	Description
id	String	Resource ID.
name	String	Resource name.
type	String	Basic resources: cce, as, and ecs. Optional resources: rds, dcs, elb, cse, and other services.

Example Request

None

Example Response

```
{
  "environments": [
    {
      "id": "00078e9d-a61c-476e-ac63-a10c9cb2638e",
      "name": "development-env",
      "alias": null,
      "description": "",
      "project_id": "e7d2e9c589e5445e808a8ff0d1235aca",
      "enterprise_project_id": "0",
      "charge_mode": "provided",
      "deploy_mode": "container",
      "vpc_id": "29d55020-ae0e-4a18-871c-93e6976ee7bd",
      "base_resources": [
        {
          "id": "523498f1-36c4-11eb-ae36-0255ac1000c2",
          "type": "cce",
          "name": "cce-test"
        }
      ],
      "optional_resources": [
        {
          "id": "9963306a-791e-468c-9306-cf80a9d00298",
          "type": "elb",
          "name": "elb-2dd2"
        }
      ],
      "creator": "ss-test",
      "create_time": 1610418873730,
      "update_time": 1610418873730
    }
  ]
}
```

```
}  
],  
"count": 1  
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

12.1.2.3 Modifying Environment Information

Function

This API is used to modify environment information based on the environment ID.

URI

PUT /v2/{project_id}/cas/environments/{environment_id}

Table 12-21 Path parameters

Parameter	Mandator y	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
environment_i d	Yes	String	Environment ID. See Obtaining All Environments .

Request

Table 12-22 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .

Table 12-23 Request body parameters

Parameter	Mandatory	Type	Description
name	No	String	Environment name. The value contains 2 to 64 characters consisting of letters, digits, hyphens (-), and underscores (_). It starts with a letter and ends with a letter or digit. Letters are case insensitive.
alias	No	String	Environment alias. The value can contain up to 64 characters.
description	No	String	Environment description. The value can contain up to 128 characters.

Response

Table 12-24 Response parameters

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

Parameter	Type	Description
alias	String	Environment alias.
description	String	Environment description.
project_id	String	Project ID.
enterprise_project_id	String	Enterprise project ID.
charge_mode	String	Billing mode.
deploy_mode	String	Environment type. • virtualmachine • container: Kubernetes • mixed: VM and Kubernetes
vpc_id	String	VPC ID.
base_resources	Array of objects	Basic resources. See Table 12-25 .
optional_resources	Array of objects	Optional resources. See Table 12-25 .
creator	String	Creator.
create_time	Integer	Creation time.
update_time	Integer	Update time.

Table 12-25 resources

Parameter	Type	Description
id	String	Resource ID.
name	String	Resource name.
type	String	Basic resources: cce, as, and ecs. Optional resources: rds, dcs, elb, cse, and other services.

Example Request

Add environment description for environment **development-env2**.

```
{
  "name": "development-env2",
  "description": "here is description"
}
```

Example Response

```
{
  "id": "00078e9d-a61c-476e-ac63-a10c9cb2638e",
  "name": "development-env2",
  "alias": null,
  "description": "here is description",
  "project_id": "e7d2e9c589e5445e808a8ff0d1235aca",
  "enterprise_project_id": "0",
  "charge_mode": "provided",
  "deploy_mode": "container",
  "vpc_id": "29d55020-ae0e-4a18-871c-93e6976ee7bd",
  "base_resources": [
    {
      "id": "523498f1-36c4-11eb-ae36-0255ac1000c2",
      "type": "cce",
      "name": "cce-test"
    }
  ],
  "optional_resources": [
    {
      "id": "9963306a-791e-468c-9306-cf80a9d00298",
      "type": "elb",
      "name": "elb-2dd2"
    }
  ],
  "creator": "ss-test",
  "create_time": 1610418873730,
  "update_time": 1610420992462
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

12.1.2.4 Deleting an Environment Based on the Environment ID

Function

This API is used to delete an environment based on the environment ID.

URI

DELETE /v2/{project_id}/cas/environments/{environment_id}

Table 12-26 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
environment_id	Yes	String	Environment ID. See Obtaining All Environments .

Request

Table 12-27 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .

Response

None

Example Request

None

Example Response

None

Status Code

Status Code	Description
204	OK
400	Bad Request
404	Not Found

Status Code	Description
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

12.1.2.5 Obtaining Environment Details Based on the Environment ID

Function

This API is used to obtain environment details based on the environment ID.

URI

GET /v2/{project_id}/cas/environments/{environment_id}

Table 12-28 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
environment_id	Yes	String	Environment ID. See Obtaining All Environments .

Request

Table 12-29 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .

Response

Table 12-30 Response parameters

Parameter	Type	Description
id	String	Environment ID.
name	String	Environment name.
alias	String	Environment alias.
description	String	Environment description.
project_id	String	Project ID.
enterprise_project_id	String	Enterprise project ID.
charge_mode	String	Billing mode.
deploy_mode	String	Environment type. • virtualmachine • container: Kubernetes • mixed: VM and Kubernetes
vpc_id	String	VPC ID.
base_resources	Array of objects	Basic resources. See Table 12-31 .
optional_resources	Array of objects	Optional resources. See Table 12-31 .
creator	String	Creator.
create_time	Integer	Creation time.
update_time	Integer	Update time.

Table 12-31 resources

Parameter	Type	Description
id	String	Resource ID.
name	String	Resource name.
type	String	Basic resources: cce, as, and ecs. Optional resources: rds, dcs, elb, cse, and other services.

Example Request

None

Example Response

```
{
  "id": "ea011e01-2eb5-453f-87bf-874e4a855abe",
  "name": "dev-env",
  "alias": null,
  "description": "develop environment",
  "project_id": "bf8523d898b64e4eb956e3be3555ca16",
  "enterprise_project_id": "0",
  "charge_mode": "provided",
  "deploy_mode": "container",
  "vpc_id": "234241234124xxvasf2342xxxxxxxxxx",
  "base_resources": [
    {
      "id": "211112333sd332w23322332",
      "type": "cce"
    }
  ],
  "optional_resources": null,
  "creator": "string",
  "create_time": 1578984198394,
  "update_time": 1578984198394
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

12.1.2.6 Modifying Environment Resources

Function

This API is used to modify environment resources.

URI

PATCH /v2/{project_id}/cas/environments/{environment_id}/resources

Table 12-32 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
environment_id	Yes	String	Environment ID. See Obtaining All Environments .

Request

Table 12-33 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .

Table 12-34 Request body parameters

Parameter	Mandatory	Type	Description
add_base_resources	No	Array of objects	Basic resources to be added. See Table 12-35 .
add_optional_resources	No	Array of objects	Optional resources to be added. See Table 12-36 .
remove_resources	No	Array of objects	Resources to be removed. See Table 12-37 .

Table 12-35 add_base_resources

Parameter	Mandatory	Type	Description
id	Yes	String	Resource ID. (To ensure that the API can be called properly, enter the correct resource ID.)
name	No	String	Resource name.
type	Yes	String	Basic resources: cce, as, and ecs.

Table 12-36 add_optional_resources

Parameter	Mandatory	Type	Description
id	Yes	String	Resource ID. (To ensure that the API can be called properly, enter the correct resource ID.)
name	No	String	Resource name.
type	Yes	String	Optional resources: rds, dcs, elb, cse, and other services.

Table 12-37 remove_resources

Parameter	Mandatory	Type	Description
id	Yes	String	Resource ID. (To ensure that the API can be called properly, enter the correct resource ID.)
name	No	String	Resource name.
type	Yes	String	Basic resources: cce, as, and ecs. When deploy_mode is set to virtualmachine , you can add ecs and as. When deploy_mode is set to container , you can add cce. Optional resources: rds, dcs, elb, cse, and other services.

Response

Table 12-38 Response parameters

Parameter	Type	Description
id	String	Environment ID.
name	String	Environment name.
alias	String	Environment alias.
description	String	Environment description.
project_id	String	Project ID.
enterprise_project_id	String	Enterprise project ID.
charge_mode	String	Billing mode.
deploy_mode	String	Environment type. • virtualmachine • container: Kubernetes • mixed: VM and Kubernetes
vpc_id	String	VPC ID.
base_resources	Array of objects	Basic resources. See Table 12-39 .
optional_resources	Array of objects	Optional resources. See Table 12-39 .
creator	String	Creator.
create_time	Integer	Creation time.
update_time	Integer	Update time.

Table 12-39 schemas

Parameter	Type	Description
id	String	Resource ID.
name	String	Resource name.
type	String	Basic resources: cce, as, and ecs. Optional resources: rds, dcs, elb, cse, and other services.

Example Request

Add ECS basic resource **ecs-9876** and CSE optional resource **Cloud Service Engine**, and remove ECS resource **test**.

```
{
  "add_base_resources": [
    {
      "id": "ed2f3420-7031-4d93-b92b-e360cd4acf9e",
      "type": "ecs",
      "name": "ecs-9876"
    }
  ],
  "add_optional_resources": [
    {
      "id": "default",
      "type": "cse",
      "name": "Cloud Service Engine"
    }
  ],
  "remove_resources": [
    {
      "id": "523498f1-36c4-11eb-ae36-0255ac1000c2",
      "name": "test",
      "type": "ecs"
    }
  ]
}
```

Example Response

```
{
  "id": "00078e9d-a61c-476e-ac63-a10c9cb2638e",
  "name": "development-env2",
  "alias": null,
  "description": "here is description",
  "project_id": "e7d2e9c589e5445e808a8ff0d1235aca",
  "enterprise_project_id": "0",
  "charge_mode": "provided",
  "deploy_mode": "virtualmachine",
  "vpc_id": "29d55020-ae0e-4a18-871c-93e6976ee7bd",
  "base_resources": [
    {
      "id": "ed2f3420-7031-4d93-b92b-e360cd4acf9e",
      "type": "ecs",
      "name": "ecs-9876"
    }
  ],
  "optional_resources": [
    {
      "id": "9963306a-791e-468c-9306-cf80a9d00298",
      "type": "elb",
      "name": "elb-2dd2"
    },
    {
      "id": "default",
      "type": "cse",
      "name": "Cloud Service Engine"
    }
  ],
  "creator": "ss-test",
  "create_time": 1610418873730,
  "update_time": 1610420992462
}
```


Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

12.1.3 Application

12.1.3.1 Creating an Application

Function

An application is a service system with complete functions and consists of one or more components related to features.

This API is used to create an application.

URI

POST /v2/{project_id}/cas/applications

Table 12-40 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .

Request

Table 12-41 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .

Table 12-42 Request body parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Application name. The value contains 2 to 64 characters consisting of letters, digits, hyphens (-), and underscores (_). It starts with a letter and ends with a letter or digit.
description	No	String	Application description. The value can contain up to 128 characters.
enterprise_project_id	No	String	Enterprise project ID.

Response

Table 12-43 Response parameters

Parameter	Type	Description
id	String	Application ID.
name	String	Application name.
description	String	Application description.
creator	String	Creator.
project_id	String	Project ID.
enterprise_project_id	String	Enterprise project ID.
create_time	Integer	Creation time.
update_time	Integer	Update time.

Parameter	Type	Description
unified_model	String	Whether to enable the unified model. Default value: null.

Example Request

Create application **app-xpmtii** and add application description **test**.

```
{
  "name": "app-xpmtii",
  "description": "test"
}
```

Example Response

```
{
  "id": "e5213b62-0e3c-476f-9960-3e4108787350",
  "name": "app-xpmtii",
  "description": "test",
  "project_id": "e7d2e9c589e5445e808a8ff0d1235aca",
  "enterprise_project_id": "0",
  "creator": "ss-test",
  "create_time": 1610432385245,
  "update_time": 1610432385245,
  "unified_model": null
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

12.1.3.2 Obtaining All Applications

Function

This API is used to obtain all created applications.

URI

GET /v2/{project_id}/cas/applications

Table 12-44 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID. See Obtaining a Project ID .

Table 12-45 Query parameters

Parameter	Mandatory	Type	Description
limit	No	String	Number of records to be displayed. The value is 1000 or ranges from 0 to 100. If the specified value is not within the range, value 10 is assigned. In the non-pagination scenario, the value is 1000 . In the pagination scenario, the value ranges from 0 to 100.
offset	No	String	Offset, which indicates the result after the Nth data record is queried.
order_by	No	String	Sorting field. By default, query results are sorted by creation time. Enumerated values: create_time, name, and update_time. If the transferred value is not within the supported enumerated value range, the default sorting field is used.
order	No	String	Sorting order. • desc (default) • asc

Request

Table 12-46 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .

Response

Table 12-47 Response parameters

Parameter	Type	Description
count	Integer	Total number of applications.
applications	Array of objects	Application information. See Table 12-48 .

Table 12-48 applications

Parameter	Type	Description
id	String	Application ID.
name	String	Application name.
description	String	Application description.
creator	String	Creator.
project_id	String	Project ID.
enterprise_project_id	String	Enterprise project ID.
create_time	Integer	Creation time.
update_time	Integer	Update time.
unified_model	String	Whether to enable the unified model. Default value: null.
component_count	Integer	Number of components.

Example Request

None

Example Response

```
{
  "count": 1,
  "applications": [
    {
      "id": "e5213b62-0e3c-476f-9960-3e4108787350",
      "name": "app-xpmtii",
      "description": "test",
      "project_id": "e7d2e9c589e5445e808a8ff0d1235aca",
      "enterprise_project_id": "0",
      "creator": "ss-test",
      "create_time": 1610432385245,
      "update_time": 1610432385245,
      "unified_model": null,
      "component_count": 0
    }
  ]
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

12.1.3.3 Modifying Application Information

Function

This API is used to modify application information based on the application ID.

URI

PUT /v2/{project_id}/cas/applications/{application_id}

Table 12-49 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
application_id	Yes	String	Application ID. See Obtaining All Applications .

Request

Table 12-50 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .

Table 12-51 Request body parameters

Parameter	Mandatory	Type	Description
name	No	String	Application name. The value contains 2 to 64 characters consisting of letters, digits, hyphens (-), and underscores (_). It starts with a letter and ends with a letter or digit.
description	No	String	Application description. The value can contain up to 128 characters.

Response

Table 12-52 Response parameters

Parameter	Type	Description
id	String	Application ID.
name	String	Application name.
description	String	Application description.
creator	String	Creator.
project_id	String	Project ID.
enterprise_project_id	String	Enterprise project ID.

Parameter	Type	Description
create_time	Integer	Creation time.
update_time	Integer	Update time.
unified_model	String	Whether to enable the unified model. Default value: null.

Example Request

Modify application name to **app-test**.

```
{
  "name": "app-test",
  "description": "test"
}
```

Example Response

```
{
  "id": "e5213b62-0e3c-476f-9960-3e4108787350",
  "name": "app-test",
  "description": "test",
  "project_id": "e7d2e9c589e5445e808a8ff0d1235aca",
  "enterprise_project_id": "0",
  "creator": "ss-test",
  "create_time": 1610432385245,
  "update_time": 1610433070875,
  "unified_model": null
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

12.1.3.4 Deleting an Application Based on the Application ID

Function

This API is used to delete an application based on the application ID.

URI

DELETE /v2/{project_id}/cas/applications/{application_id}

Table 12-53 Path parameters

Parameter	Mandat ory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
application_id	Yes	String	Application ID. See Obtaining All Applications .

Request

Table 12-54 Request header parameters

Parameter	Mandat ory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .

Response

None

Example Request

None

Example Response

None

Status Code

Status Code	Description
204	OK

Status Code	Description
400	Bad Request
404	Not Found
409	Conflict
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

12.1.3.5 Obtaining Application Details Based on the Application ID

Function

This API is used to obtain application details based on the application ID.

URI

GET /v2/{project_id}/cas/applications/{application_id}

Table 12-55 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
application_id	Yes	String	Application ID. See Obtaining All Applications .

Request

Table 12-56 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .

Response

Table 12-57 Response parameters

Parameter	Type	Description
id	String	Application ID.
name	String	Application name.
description	String	Application description.
creator	String	Creator.
project_id	String	Project ID.
enterprise_project_id	String	Enterprise project ID.
create_time	Integer	Creation time.
update_time	Integer	Update time.
unified_model	String	Whether to enable the unified model. Default value: null.
component_count	Integer	Number of components.

Example Request

None

Example Response

```
{
  "id": "e5213b62-0e3c-476f-9960-3e4108787350",
  "name": "app-test",
  "description": "test",
  "project_id": "e7d2e9c589e5445e808a8ff0d1235aca",
  "enterprise_project_id": "0",
  "creator": "ss-test",
  "create_time": 1610432385245,
  "update_time": 1610433070875,
  "unified_model": null,
```

```
"component_count": 0  
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

12.1.3.6 Modifying Application Configurations

Function

This API is used to modify application configurations.

Application configurations refer to common environment variables of an application. Components deployed in the application inherit these environment variables.

URI

PUT /v2/{project_id}/cas/applications/{application_id}/configuration

Table 12-58 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
application_id	Yes	String	Application ID. See Obtaining All Applications .

Request

Table 12-59 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .

Table 12-60 Request body parameters

Parameter	Mandatory	Type	Description
environment_id	Yes	String	Environment ID, which can be obtained in the corresponding environment by referring to Obtaining All Environments .
configuration	Yes	Object	Application configurations, such as public environment variables. See Table 12-61 .

Table 12-61 configuration

Parameter	Mandatory	Type	Description
env	Yes	Array of objects	Environment variables of the application. See Table 12-62 . If the names of multiple environment variables are the same, only the last environment variable takes effect.

Table 12-62 env

Parameter	Mandatory	Type	Description
name	Yes	String	Name of the environment variable. The value contains 1 to 64 characters, including letters, digits, underscores (_), hyphens (-), and dots (.), and cannot start with a digit.
value	Yes	String	Value of the environment variable.

NOTICE

The **env** variable in the request parameter **configuration** will overwrite the existing environment variable configuration of the application. Set this parameter based on the scenario where environment variables are added or replaced.

Response

Table 12-63 Response parameters

Parameter	Type	Description
application_id	String	Application ID.
environment_id	String	Environment ID.
configuration	Object	Application configurations. See Table 12-64 .

Table 12-64 configuration

Parameter	Type	Description
env	Array of objects	Environment variables of the application. See Table 12-65 .

Table 12-65 env

Parameter	Type	Description
name	String	Name of the environment variable.
value	String	Value of the environment variable.

Example Request

Modify the environment whose ID is **0fa75dfe-0d32-4bc1-848b-8008cf3f2567** by changing the name to **app-env** and the value to **env**.

```
{
  "environment_id": "0fa75dfe-0d32-4bc1-848b-8008cf3f2567",
  "configuration": {
    "env": [
      {
        "name": "app-env",
        "value": "env"
      }
    ]
  }
}
```

Example Response

```
{
  "application_id": "06cf2fda-af45-44b4-9e22-6294c4804515",
  "environment_id": "0fa75dfe-0d32-4bc1-848b-8008cf3f2567",
  "configuration": {
    "env": [
      {
        "name": "app-env",
        "value": "env"
      }
    ]
  }
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

12.1.3.7 Deleting Application Configurations

Function

This API is used to delete application configurations.

URI

DELETE /v2/{project_id}/cas/applications/{application_id}/configuration

Table 12-66 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
application_id	Yes	String	Application ID. See Obtaining All Applications .

Table 12-67 Query parameters

Parameter	Mandatory	Type	Description
environment_id	Yes	String	Environment ID. See Obtaining All Environments .

Request

Table 12-68 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .

Response

None

Example Request

None

Example Response

None

Status Code

Status Code	Description
204	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

12.1.3.8 Obtaining Application Configurations

Function

This API is used to obtain application configurations.

URI

GET /v2/{project_id}/cas/applications/{application_id}/configuration

Table 12-69 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
application_id	Yes	String	Application ID. See Obtaining All Applications .

Table 12-70 Query parameters

Parameter	Mandatory	Type	Description
environment_id	No	String	Environment ID. If this parameter is not specified, all environments are queried. See Obtaining All Environments .

Request

Table 12-71 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .

Response

Table 12-72 Response parameters

Parameter	Type	Description
configuration	Array of objects	Configuration information. See Table 12-73 .

Table 12-73 Application configuration parameters

Parameter	Type	Description
application_id	String	Application ID.
environment_id	String	Environment ID.
configuration	Object	Application configurations. See Table 12-74 .

Table 12-74 configuration

Parameter	Type	Description
env	Array of objects	Environment variables of the application component. See Table 12-75 .

Table 12-75 env

Parameter	Type	Description
name	String	Name of the environment variable.
value	String	Value of the environment variable.

Example Request

None

Example Response

```
{
  "configuration": [
    {
      "application_id": "e5213b62-0e3c-476f-9960-3e4108787350",
      "environment_id": "00078e9d-a61c-476e-ac63-a10c9cb2638e",
      "configuration": {
        "env": [
          {
            "name": "env",
            "value": "value1"
          }
        ]
      }
    }
  ]
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

12.1.4 Component

12.1.4.1 Creating an Application Component

Function

An application component implements a service feature of an application. It is in the form of code or software packages and can be deployed independently.

This API is used to create an application component.

URI

POST /v2/{project_id}/cas/applications/{application_id}/components

Table 12-76 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
application_id	Yes	String	Application ID. See Obtaining All Applications .

Request

Table 12-77 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .

Table 12-78 Request body parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Application component name. The value contains 2 to 64 characters consisting of letters, digits, hyphens (-), and underscores (_). It starts with a letter and ends with a letter or digit. Letters are case insensitive.
runtime	Yes	String	Runtime. The value can be obtained from type_name returned by the API in Obtaining All Supported Runtime Systems of Application Components .
category	Yes	String	Application component category. Example: Webapp, MicroService, or Common.
sub_category	No	String	Application component sub-category. Webapp sub-categories include Web. MicroService sub-categories include Java Chassis, and Spring Cloud. Common sub-category can be empty.
description	No	String	Description. The value can contain up to 128 characters.
source	No	Object	Source of the code or software package. See Table 12-79 .
build	No	Object	Component build. See Table 12-83 .

Table 12-79 source

Parameter	Mandatory	Type	Description
kind	Yes	String	Type. Option: source code or artifact software package.
spec	Yes	Object	For details about the source code, see Table 12-80 . For details about the artifact software package, see Table 12-81 .

Table 12-80 code spec

Parameter	Mandatory	Type	Description
repo_type	No	String	Code repository type. Value: GitHub, BitBucket, or GitLab.
repo_url	No	String	Code repository URL. Example: https://github.com/example/demo.git.
repo_ref	No	String	Code branch or tag. Default value: master.
repo_auth	No	String	Authorization name, which can be obtained from the authorization list.

Table 12-81 artifact spec

Parameter	Mandatory	Type	Description
storage	No	String	Storage mode. Value: swr (software repository), swr (image), obs, or codearts.
type	No	String	Type. • VM-based deployment supports package. • Container-based deployment supports package and image.
url	No	String	Address of the software package or source code.
auth	No	String	Authentication mode. Value: iam or none. Default value: iam.
properties	No	Object	Other attributes of the software package. You need to add these attributes only when you set storage to obs . See Table 12-82 .
webUrl	No	String	Address of a software repository.

Table 12-82 artifact spec properties

Parameter	Mandatory	Type	Description
endpoint	No	String	OBS endpoint address. Example: https://obs.region_id.external_domain_name.com.
bucket	No	String	Name of the OBS bucket where the software package is stored.

Parameter	Mandatory	Type	Description
key	No	String	Object in the OBS bucket, which is usually the name of the software package. If there is a folder, the path of the folder must be added. Example: test.jar or demo/test.jar.

Table 12-83 build

Parameter	Mandatory	Type	Description
parameters	No	Map<String, Object>	See Table 12-84 . This parameter is provided only when no ID is available during build creation.

Table 12-84 parameters

Parameter	Mandatory	Type	Description
build_cmd	No	String	Compilation command. By default: 1. When build.sh exists in the root directory, the command is ./build.sh . 2. When build.sh does not exist in the root directory, the command varies depending on the OS. Example: • Java and Tomcat: mvn clean package • Node.js: npm build
dockerfile_path	No	String	Address of the Docker file. By default, the Docker file is in the root directory (./).
artifact_namespace	No	String	Build archive organization. Default value: cas_{project_id}.
cluster_id	No	String	ID of the cluster to be built.
node_label_selector	No	Map<String, String>	key: label key. value: label value.

Response

Table 12-85 Response parameters

Parameter	Type	Description
id	String	Application component ID.
name	String	Application component name.
status	Integer	Value: 0 or 1. 0: Normal. 1: Being deleted.
runtime	String	Runtime system.
category	String	Application component category. Example: Webapp, MicroService, or Common.
sub_category	String	Application component sub-category. Webapp sub-categories include Web. MicroService sub-categories include Java Chassis, and Spring Cloud. Common sub-category can be empty.
description	String	Description.
project_id	String	Project ID.
application_id	String	Application ID.
source	Object	Source of the code or software package. See Table 12-86 .
build	Object	Build. See Table 12-90 .
create_time	Integer	Creation time.
update_time	Integer	Update time.
creator	String	Creator.

Table 12-86 source

Parameter	Type	Description
kind	String	Type. Option: source code or artifact software package.
spec	Object	For details about the artifact software package, see Table 12-88 ; For details about the source code, see Table 12-87 .

Table 12-87 code spec

Parameter	Type	Description
repo_type	String	Code repository type. Value: GitHub, BitBucket, or GitLab.
repo_url	String	Code repository URL. Example: https://github.com/example/demo.git.
repo_ref	String	Code branch or tag. Default value: master.
repo_auth	String	Authorization name, which can be obtained from the authorization list.

Table 12-88 artifact spec

Parameter	Type	Description
storage	String	Storage mode.
type	String	Type.
url	String	Address of the software package or source code.
auth	String	Authentication mode.
properties	Object	Other attributes of the software package. You need to add these attributes only when you set storage to obs . See Table 12-89 .

Table 12-89 artifact spec properties

Parameter	Type	Description
endpoint	String	OBS endpoint address. Example: https://obs.region_id.external_domain_name.com.
bucket	String	Name of the OBS bucket where the software package is stored.
key	String	Object in the OBS bucket, which is usually the name of the software package. If there is a folder, the path of the folder must be added. Example: test.jar or demo/test.jar.

Table 12-90 build

Parameter	Type	Description
id	String	Type.

Parameter	Type	Description
parameters	Map<String, Object>	See Table 12-91 .

Table 12-91 parameters

Parameter	Type	Description
build_cmd	String	Compilation command.
dockerfile_path	String	Address of the Docker file.
artifact_namespace	String	Build archive organization.
cluster_id	String	ID of the cluster to be built.
node_selector	Map<String, String>	key: label key. value: label value.

Example Request

Create an application microservice component for software package deployment. Set runtime to Java8 and the component software package source to the **demo.jar** package in the myapp bucket of OBS.

```
{
  "name": "mycomponent",
  "runtime": "Java8",
  "category": "MicroService",
  "sub_category": "Java Chassis",
  "description": "",
  "build": {
    "parameters": {
      "artifact_namespace": "ns",
      "use_public_cluster": false,
      "cluster_id": "523498f1-36c4-11eb-ae36-0255ac1000c2",
      "cluster_name": "cce-test",
      "cluster_type": "VirtualMachine"
    }
  },
  "source": {
    "kind": "artifact",
    "spec": {
      "storage": "obs",
      "type": "package",
      "url": "obs://myapp/demo.jar",
      "properties": {
        "bucket": "myapp",
        "key": "demo.jar",
        "endpoint": "https://obs.region_id.external_domain_name.com"
      }
    }
  }
}
```

Example Response

```
{
  "id": "384eb8d4-c193-4d84-9558-6fda2366b536",
  "name": "mycomponent",
  "runtime": "Java8",
  "category": "MicroService",
  "sub_category": "Java Chassis",
  "description": "",
  "project_id": "384eb8d4-c193-4d84-9558-6fda23698536",
  "application_id": "a8f7eed5-0aa0-4251-9723-c9119a6bf56d",
  "source": {
    "kind": "artifact",
    "spec": {
      "storage": "obs",
      "type": "package",
      "url": "obs://myapp/demo.jar",
      "properties": {
        "bucket": "myapp",
        "key": "demo.jar",
        "endpoint": "https://obs.region_id.external_domain_name.com"
      }
    }
  },
  "build": {
    "id": "w3dpv7p0t1vpvxy5hjb22iuwxway1vupwx0nae1",
    "parameters": {
      "artifact_namespace": "ns",
      "dockerfile_path": "./"
    }
  },
  "status": 0,
  "creator": "test_user",
  "create_time": 1610333934288,
  "update_time": 1610333934288
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

12.1.4.2 Obtaining All Components of an Application

Function

This API is used to obtain all components of an application.

URI

GET /v2/{project_id}/cas/applications/{application_id}/components

Table 12-92 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
application_id	Yes	String	Application ID. See Obtaining All Applications .

Table 12-93 Query parameters

Parameter	Mandatory	Type	Description
limit	No	String	Number of records to be displayed. The value is 1000 or ranges from 0 to 100. If the specified value is not within the range, value 10 is assigned. In the non-pagination scenario, the value is 1000 . In the pagination scenario, the value ranges from 0 to 100.
offset	No	String	Offset.
order_by	No	String	Sorting field. By default, query results are sorted by creation time. Enumerated values: create_time, name, and update_time.
order	No	String	Sorting order. • desc (default) • asc

Request

Table 12-94 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .

Response

Table 12-95 Response parameters

Parameter	Type	Description
count	Integer	Total number of components.
components	Array of objects	Component parameters. See Table 12-96 .

Table 12-96 components

Parameter	Type	Description
id	String	Application component ID.
name	String	Application component name.
status	Integer	Value: 0 or 1. 0: Normal. 1: Being deleted.
runtime	String	Runtime system.
category	String	Application component category. Example: Webapp, MicroService, or Common.
sub_category	String	Application component sub-category. Webapp sub-categories include Web. MicroService sub-categories include Java Chassis, and Spring Cloud. Common sub-category can be empty.
application_id	String	Application ID.
source	Object	Source of the code or software package. See Table 12-97 .
description	String	Description.

Parameter	Type	Description
creator	String	Creator.
create_time	Integer	Creation time.
update_time	Integer	Update time.
project_id	String	Tenant's project ID.
build	Object	Build. See Table 12-101 .

Table 12-97 source

Parameter	Type	Description
kind	String	Type. Option: source code or artifact software package.
spec	Object	For details about the source code, see Table 12-98 . For details about the artifact software package, see Table 12-99 .

Table 12-98 code spec

Parameter	Type	Description
repo_type	String	Code repository type. Value: GitHub, BitBucket, or GitLab.
repo_url	String	Code repository URL. Example: https://github.com/example/demo.git .
repo_ref	String	Code branch or tag. Default value: master.
repo_auth	String	Authorization name, which can be obtained from the authorization list.

Table 12-99 artifact spec

Parameter	Type	Description
storage	String	Storage mode.
type	String	Type.
url	String	Address of the software package or source code.
auth	String	Authentication mode.

Parameter	Type	Description
properties	Object	Other attributes of the software package. You need to add these attributes only when you set storage to obs . See Table 12-100 .

Table 12-100 artifact spec properties

Parameter	Type	Description
endpoint	String	OBS endpoint address. Example: https://obs.region_name.external_domain_name.com.
bucket	String	Name of the OBS bucket where the software package is stored.
key	String	Object in the OBS bucket, which is usually the name of the software package. If there is a folder, the path of the folder must be added. Example: test.jar or demo/test.jar.

Table 12-101 build

Parameter	Type	Description
id	String	Job ID.
parameters	Map<String,String>	See Table 12-102 .

Table 12-102 parameters

Parameter	Type	Description
build_cmd	String	Compilation command.
dockerfile_path	String	Address of the Docker file.
artifact_namespace	String	Build archive organization.
cluster_id	String	ID of the cluster to be built.
node_label_selector	Map<String, String>	key: label key. value: label value.

Example Request

None

Example Response

```
{
  "components": [
    {
      "id": "384eb8d4-c193-4d84-9558-6fda2366b536",
      "name": "mycomponent",
      "runtime": "Java8",
      "category": "MicroService",
      "sub_category": "Java Chassis",
      "description": "",
      "project_id": "384eb8d4-c193-4d84-9558-6fda23698536",
      "application_id": "a8f7eed5-0aa0-4251-9723-c9119a6bf56d",
      "source": {
        "kind": "artifact",
        "spec": {
          "storage": "obs",
          "type": "package",
          "url": "obs://myapp/demo.jar",
          "properties": {
            "bucket": "myapp",
            "key": "demo.jar",
            "endpoint": "https://obs.region_id.external_domain_name.com"
          }
        }
      },
      "build": {
        "id": "w3dpv7p0t1vpvvey5hjb22iuwxway1vupwx0nae1",
        "parameters": {
          "artifact_namespace": "ns",
          "use_public_cluster": true
        }
      },
      "status": 0,
      "creator": "test_user",
      "create_time": 1610333934288,
      "update_time": 1610333934288
    }
  ],
  "count": 1
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

12.1.4.3 Modifying Component Information Based on the Component ID

Function

This API is used to modify component information based on the component ID.

URI

PUT /v2/{project_id}/cas/applications/{application_id}/components/{component_id}

Table 12-103 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
application_id	Yes	String	Application ID. See Obtaining All Applications .
component_id	Yes	String	Component ID. See Obtaining All Components of an Application .

Request

Table 12-104 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .

Table 12-105 Request body parameters

Parameter	Mandatory	Type	Description
name	No	String	Application component name. The value contains 2 to 64 characters consisting of letters, digits, hyphens (-), and underscores (_). It starts with a letter and ends with a letter or digit. Letters are case insensitive.
description	No	String	Description. The value can contain up to 128 characters.
source	No	Object	Source of the code or software package. See Table 12-106 .
build	No	Object	Build. See Table 12-110 .

Table 12-106 source

Parameter	Mandatory	Type	Description
kind	Yes	String	Type. Option: source code or artifact software package.
spec	Yes	Object	For details about the source code, see Table 12-107 . For details about the artifact software package, see Table 12-108 .

Table 12-107 code spec

Parameter	Mandatory	Type	Description
repo_type	No	String	Code repository type. Value: GitHub, BitBucket, or GitLab.
repo_url	No	String	Code repository URL. Example: https://github.com/example/demo.git.
repo_ref	No	String	Code branch or tag. Default value: master.
repo_auth	No	String	Authorization name, which can be obtained from the authorization list.

Table 12-108 artifact spec

Parameter	Mandatory	Type	Description
storage	No	String	Storage mode. Value: swr (software repository), swr (image), obs, or codearts.
type	No	String	Type. • VM-based deployment supports package. • Container-based deployment supports package and image.
url	No	String	Address of the software package or source code.
auth	No	String	Authentication mode. Value: iam or none. Default value: iam.
properties	No	Object	Other attributes of the software package. You need to add these attributes only when you set storage to obs . See Table 12-109 .
webUrl	No	String	Address of a software repository.

Table 12-109 artifact spec properties

Parameter	Mandatory	Type	Description
endpoint	No	String	OBS endpoint address. Example: https://obs.region_id.external_domain_name.com.
bucket	No	String	Name of the OBS bucket where the software package is stored.
key	No	String	Object in the OBS bucket, which is usually the name of the software package. If there is a folder, the path of the folder must be added. Example: test.jar or demo/test.jar.

Table 12-110 build

Parameter	Mandatory	Type	Description
parameters	No	Map<String, Object>	See Table 12-111 . This parameter is provided only when no ID is available during build creation.

Table 12-111 parameters

Parameter	Mandatory	Type	Description
build_cmd	No	String	Compilation command. By default: 1. When build.sh exists in the root directory, the command is ./build.sh . 2. When build.sh does not exist in the root directory, the command varies depending on the OS. Example: • Java and Tomcat: mvn clean package • Node.js: npm build
dockerfile_path	No	String	Address of the Docker file. By default, the Docker file is in the root directory (./).
artifact_namespace	No	String	Build archive organization. Default value: cas_{project_id}.
cluster_id	Yes	String	ID of the cluster to be built.
node_label_selector	No	Map<String, String>	key: label key. value: label value.
cluster_name	No	String	Name of the cluster to be built.
use_public_cluster	No	Boolean	Whether to use the public cluster: • true • false
cluster_type	No	String	Type of the cluster to be built.

Response

Table 12-112 Response parameters

Parameter	Type	Description
id	String	Application component ID.
name	String	Application component name.
status	Integer	Value: 0 or 1. 0: Normal. 1: Being deleted.
runtime	String	Runtime system.
category	String	Application component category. Example: Webapp, MicroService, or Common.
sub_category	String	Application component sub-category. Webapp sub-categories include Web. MicroService sub-categories include Java Chassis, and Spring Cloud. Common sub-category can be empty.
description	String	Description.
project_id	String	Project ID.
application_id	String	Application ID.
source	Object	Source of the code or software package. See Table 12-113 .
build	Object	Build. See Table 12-117 .
create_time	Integer	Creation time.
update_time	Integer	Update time.
creator	String	Creator.

Table 12-113 source

Parameter	Type	Description
kind	String	Type.
spec	Object	For details about the source code, see Table 12-114 . For details about the artifact software package, see Table 12-115 .

Table 12-114 code spec

Parameter	Type	Description
repo_type	String	Code repository type. Value: github, bitbucket, or gitlab.
repo_url	String	Code repository URL. Example: https://github.com/example/demo.git.
repo_ref	String	Code branch or tag. Default value: master.
repo_auth	String	Authorization name, which can be obtained from the authorization list.

Table 12-115 spec

Parameter	Type	Description
storage	String	Storage mode.
type	String	Type.
url	String	Address of the software package or source code.
auth	String	Authentication mode.
properties	Object	Other attributes of the software package. You need to add these attributes only when you set storage to obs . See Table 12-116 .

Table 12-116 artifact spec properties

Parameter	Type	Description
endpoint	String	OBS endpoint address. Example: https://obs.region_id.external_domain_name.com.
bucket	String	Name of the OBS bucket where the software package is stored.
key	String	Object in the OBS bucket, which is usually the name of the software package. If there is a folder, the path of the folder must be added. Example: test.jar or demo/test.jar.

Table 12-117 build

Parameter	Type	Description
id	String	Type.

Parameter	Type	Description
parameters	Map<String, Object>	See Table 12-118 .

Table 12-118 parameters

Parameter	Type	Description
build_cmd	String	Compilation command.
dockerfile_path	String	Address of the Docker file.
artifact_namespace	String	Build archive organization.
cluster_id	String	ID of the cluster to be built.
node_label_selector	Map<String, String>	key: label key. value: label value.

Example Request

Modify the component information corresponding to the specified component ID as follows: Change the component name to **test-micro**, set the component software package source to the **fusionweather-1.0.0.jar** package in the test-soft bucket of OBS, and specify the cce-test cluster as the component to build a cluster.

```
{
  "name": "test-micro",
  "description": "",
  "source": {
    "kind": "artifact",
    "spec": {
      "storage": "obs",
      "type": "package",
      "url": "obs://test-soft/fusionweather-1.0.0.jar",
      "auth": "iam",
      "webUrl": "",
      "properties": {
        "bucket": "test-soft",
        "endpoint": "https://obs.region_id.development.com:443",
        "key": "fusionweather-1.0.0.jar"
      }
    }
  },
  "build": {
    "parameters": {
      "artifact_namespace": "ns",
      "use_public_cluster": false,
      "cluster_id": "523498f1-36c4-11eb-ae36-0255ac1000c2",
      "cluster_name": "cce-test",
      "cluster_type": "VirtualMachine"
    }
  }
}
```

Example Response

```
{
  "id": "1cfdda6f-84cd-4ead-8e09-628fabf662e2",
  "name": "test-micro",
  "runtime": "Java8",
  "category": "MicroService",
  "sub_category": "Java Chassis",
  "description": "",
  "project_id": "e7d2e9c589e5445e808a8ff0d1235aca",
  "application_id": "4d92833a-fa05-4fc0-a761-e67620022310",
  "source": {
    "kind": "artifact",
    "spec": {
      "storage": "obs",
      "type": "package",
      "url": "obs://test-soft/fusionweather-1.0.0.jar",
      "auth": "iam",
      "properties": {
        "bucket": "test-soft",
        "endpoint": "https://obs.region_id.development.com:443",
        "key": "fusionweather-1.0.0.jar"
      }
    }
  },
  "build": {
    "id": "zpdkj42b3rnko8bmyojzt1ng828ubychw13j8flv",
    "parameters": {
      "artifact_namespace": "ns",
      "cluster_id": "523498f1-36c4-11eb-ae36-0255ac1000c2"
    }
  },
  "status": 0,

  "creator": "ss-test",
  "create_time": 1610331760105,
  "update_time": 1610519881943
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

12.1.4.4 Deleting a Component Based on the Component ID

Function

This API is used to delete a component based on the component ID.

URI

DELETE /v2/{project_id}/cas/applications/{application_id}/components/{component_id}

Table 12-119 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
application_id	Yes	String	Application ID. See Obtaining All Applications .
component_id	Yes	String	Component ID. See Obtaining All Components of an Application .

Table 12-120 Query parameters

Parameter	Mandatory	Type	Description
force	No	boolean	Whether to forcibly delete the component.

Request

Table 12-121 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .

Response

None

Example Request

None

Example Response

None

Status Code

Status Code	Description
204	OK
400	Bad Request
404	Not Found
409	Conflict
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

12.1.4.5 Obtaining Component Information Based on the Component ID

Function

This API is used to obtain component information based on the component ID.

URI

GET /v2/{project_id}/cas/applications/{application_id}/components/{component_id}

Table 12-122 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
application_id	Yes	String	Application ID. See Obtaining All Applications .
component_id	Yes	String	Component ID. See Obtaining All Components of an Application .

Request

Table 12-123 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type or format.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .

Response

Table 12-124 Response parameters

Parameter	Type	Description
id	String	Application component ID.
name	String	Application component name.
status	Integer	Value: 0 or 1. 0: Normal. 1: Being deleted.
runtime	String	Runtime system.
category	String	Application component category. Example: Webapp, MicroService, or Common.
sub_category	String	Application component sub-category. Webapp sub-categories include Web. MicroService sub-categories include Java Chassis, and Spring Cloud. Common sub-category can be empty.
description	String	Description.
project_id	String	Project ID.
application_id	String	Application ID.
source	Object	Source of the code or software package. See Table 12-125 .
build	Object	Build. See Table 12-129 .
create_time	Integer	Creation time.

Parameter	Type	Description
update_time	Integer	Update time.
creator	String	Creator.
pipeline_ids	Array of String	Pipeline ID.

Table 12-125 source

Parameter	Type	Description
kind	String	Type.
spec	Object	For details about the source code, see Table 12-126 . See Table 12-127 .

Table 12-126 code spec

Parameter	Type	Description
repo_type	String	Code repository type. Value: GitHub, BitBucket, or GitLab.
repo_url	String	Code repository URL. Example: https://github.com/example/demo.git .
repo_ref	String	Code branch or tag. Default value: master.
repo_auth	String	Authorization name, which can be obtained from the authorization list.

Table 12-127 spec

Parameter	Type	Description
storage	String	Storage mode.
type	String	Type.
url	String	Address of the software package/source code.
auth	String	Authentication mode.
webUrl	String	Address of the software/source code repository.
properties	Object	Other attributes of the software package. You need to add these attributes only when you set storage to obs . See Table 12-128 .

Table 12-128 artifact spec properties

Parameter	Type	Description
endpoint	String	OBS endpoint address. Example: https://obs.region_id.external_domain_name.com.
bucket	String	Name of the OBS bucket where the software package is stored.
key	String	Object in the OBS bucket, which is usually the name of the software package. If there is a folder, the path of the folder must be added. Example: test.jar or demo/test.jar.

Table 12-129 bulid

Parameter	Type	Description
id	String	Job ID.
parameters	Map<String,String>	See Table 12-130 .

Table 12-130 parameters

Parameter	Type	Description
build_cmd	String	Compilation command.
dockerfile_path	String	Address of the Docker file.
artifact_name_space	String	Build archive organization.
cluster_id	String	ID of the cluster to be built.
node_label_selector	Map<String,String>	key: label key. value: label value.

Example Request

None

Example Response

```
{
  "id": "1cfdda6f-84cd-4ead-8e09-628fabf662e2",
  "name": "test-micro",
  "runtime": "Java8",
  "category": "MicroService",
  "sub_category": "Java Chassis",
  "description": ""
```

```
{
  "project_id": "e7d2e9c589e5445e808a8ff0d1235aca",
  "application_id": "4d92833a-fa05-4fc0-a761-e67620022310",
  "source": {
    "kind": "artifact",
    "spec": {
      "storage": "obs",
      "type": "package",
      "url": "obs://test-soft/fusionweather-1.0.0.jar",
      "auth": "iam",
      "webUrl": "",
      "properties": {
        "bucket": "test-soft",
        "endpoint": "https://obs.region_id.development.com:443",
        "key": "fusionweather-1.0.0.jar"
      }
    }
  },
  "build": {
    "id": "zpdkj42b3rnko8bmyojzt1ng828ubychw13j8flv",
    "parameters": {
      "artifact_namespace": "ns",
      "use_public_cluster": false,
      "cluster_id": "523498f1-36c4-11eb-ae36-0255ac1000c2",
      "cluster_name": "cce-test",
      "cluster_type": "VirtualMachine"
    }
  },
  "status": 0,
  "creator": "ss-test",
  "create_time": 1610331760105,
  "update_time": 1610519881943
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

12.1.5 Instance

12.1.5.1 Creating an Application Component Instance

Function

This API is used to create a component instance.

URI

POST /v2/{project_id}/cas/applications/{application_id}/components/{component_id}/instances

Table 12-131 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
application_id	Yes	String	Application ID. See Obtaining All Applications .
component_id	Yes	String	Component ID. See Obtaining All Components of an Application .

Request

Table 12-132 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .

Table 12-133 Request body parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Component instance name. The value contains 2 to 63 characters consisting of lowercase letters, digits, and hyphens (-). It starts with a lowercase letter and ends with a lowercase letter or digit.
environment_id	Yes	String	Environment ID.

Parameter	Mandatory	Type	Description
replica	Yes	Integer	Number of instance replicas.
flavor_id	Yes	String	Resource specifications, which can be obtained by using the API in Obtaining All Supported Flavors of Application Resources. If you need to customize resource specifications, the format is CUSTOM-xxG:xxm-xxm:xxGi-xxGi. Where: • xxG: storage capacity allocated to a component instance. It is a reserved field. You can set it to a fixed number. • xxm-xxm: the maximum and minimum number of CPU cores allocated to a component instance. • xxGi-xxGi: the maximum and minimum memory allocated to a component instance. For example, CUSTOM-10G:0.5m-0.25m:1.6Gi-0.8Gi indicates that the maximum number of CPU cores allocated to a component instance is 0.5, the minimum number of CPU cores is 0.25, the maximum memory is 1.6 Gi, and the minimum memory is 0.8 Gi.
artifacts	Yes	Map<String, Object>	Artifact. key indicates the component name. In the Docker container scenario, key indicates the container name. See Table 12-151. If the source parameters of a component specify the software package source, this parameter is optional, and the software package source of the component is inherited by default. Otherwise, this parameter is mandatory.
version	Yes	String	Application component version that meets version semantics. Example: 1.0.0.
configuration	No	Object	Configuration parameters, such as environment variables, deployment configurations, and O&M monitoring. By default, this parameter is left blank. See Table 12-134 .

Parameter	Mandatory	Type	Description
description	No	String	Description. The value can contain up to 128 characters.
external_accesses	No	Array of objects	External network access. See Table 12-152 .
refer_resources	Yes	Array of objects	Deployed resources. See Table 12-153 .

Table 12-134 configuration

Parameter	Mandatory	Type	Description
env	No	Array of objects	Environment variable. See Table 12-135 .
storage	No	Array of objects	Data storage configuration. See Table 12-136 .
strategy	No	Object	Upgrade strategy. See Table 12-137 .
lifecycle	No	Object	Lifecycle. See Table 12-138 .
scheduler	No	Object	Scheduling strategy. See Table 12-139 .
probes	No	Object	Health check. See Table 12-140 .
container_spec	No	Object	Container information. See Table 12-154 .

Table 12-135 environment

Parameter	Mandatory	Type	Description
name	Yes	String	Environment variable name. The value contains 1 to 64 characters, including letters, digits, underscores (_), hyphens (-), and dots (.), and cannot start with a digit.
value	Yes	String	Environment variable value.

Table 12-136 storage

Parameter	Mandatory	Type	Description
type	Yes	String	Storage type. Value: • HostPath: host path mounting. • EmptyDir: temporary directory mounting. • ConfigMap: configuration item mounting. • Secret: secret volume mounting. • PersistentVolumeClaim: cloud storage mounting.
parameters	Yes	Object	Storage parameter. See Table 12-149 .
mounts	Yes	Array of objects	Directory mounted to the container. See Table 12-150 .

Table 12-137 strategy

Parameter	Mandatory	Type	Description
upgrade	No	String	Upgrade policy. Value: Recreate or RollingUpdate (default). The former indicates in-place upgrade while the latter indicates rolling upgrade.

Table 12-138 lifecycle

Parameter	Mandatory	Type	Description
entrypoint	No	Object	Startup command. See Table 12-146 .
post-start	No	Object	Post-start processing. See Table 12-147 .
pre-stop	No	Object	Pre-stop processing. See Table 12-147 .

Table 12-139 scheduler

Parameter	Mandatory	Type	Description
affinity	No	Object	Affinity. See Table 12-145 .

Parameter	Mandatory	Type	Description
anti-affinity	No	Object	Anti-affinity. See Table 12-145 .

Table 12-140 probes

Parameter	Mandatory	Type	Description
livenessProbe	No	Object	Component liveness probe. See Table 12-141 .
readinessProbe	No	Object	Component service probe. See Table 12-141 .

Table 12-141 probes_object

Parameter	Mandatory	Type	Description
type	Yes	String	Value: http, tcp, or command. The check methods are HTTP request check, TCP port check, and command execution check, respectively.
parameters	Yes	Object	Parameters. • If type is set to http, see Table 12-142. • If type is set to tcp, see Table 12-144. • If type is set to command, see Table 12-143.
delay	No	Integer	Interval between the startup and detection.
timeout	No	Integer	Detection timeout interval.

Table 12-142 probes_param_http

Parameter	Mandatory	Type	Description
scheme	No	String	Value: HTTP or HTTPS.
host	No	String	Pod IP address (default). You can specify an IP address.
port	No	Integer	Port.
path	No	String	Request path.

Table 12-143 probes_param_command

Parameter	Mandatory	Type	Description
command	No	Array of String	Command list.

Table 12-144 probes_param_tcp

Parameter	Mandatory	Type	Description
port	No	Integer	Port

Table 12-145 scheduler_affinity

Parameter	Mandatory	Type	Description
az	No	Array of String	AZ list.
node	No	Array of String	Node private IP address list.
application	No	Array of String	Component instance list.

Table 12-146 lifecycle_entrypoint

Parameter	Mandatory	Type	Description
command	No	Array of String	Command that can be executed.
args	No	Array of String	Running parameters.

Table 12-147 lifecycle_process

Parameter	Mandatory	Type	Description
type	No	String	Value: command or http. The former indicates command execution and the latter indicates HTTP requests.
parameters	No	Object	Post-start or pre-stop processing parameters. See Table 12-148 .

Table 12-148 lifecycle_process_parameters

Parameter	Mandatory	Type	Description
command	No	Array of String	Command parameters, for example, ["sleep", "1"]. This parameter is applicable to the command type.
host	No	String	Pod IP address (default). You can also specify an IP address. This parameter is applicable to the http type.
port	No	Integer	Port number. This parameter is applicable to the http type.
path	No	String	Request URL. This parameter is applicable to the http type.

Table 12-149 storage_parameters

Parameter	Mandatory	Type	Description
path	No	String	Host path. This parameter is applicable to the HostPath storage type.
name	No	String	Name of a configuration item or secret. This parameter is applicable to the ConfigMap and Secret storage type.
defaultMode	No	Integer	Mounting permission. The value is in decimal format. Example: 384.
medium	No	String	This parameter is applicable to EmptyDir storage. If it is not specified, the default disk medium is used. If it is set to memory , memory storage is enabled.

Table 12-150 storage_mounts

Parameter	Mandatory	Type	Description
path	Yes	String	Path of the mounted disk.
subPath	No	String	Sub-path of the mounted disk.
readOnly	Yes	boolean	Permission of the mounted disk, which can be read-only or read/write.

Table 12-151 artifact

Parameter	Mandatory	Type	Description
storage	Yes	String	Storage mode. Value: swr (software repository), swr (image), obs, or codearts.
type	Yes	String	Type. • VM-based deployment supports package. • Container-based deployment supports package and image.
url	Yes	String	Software package or image address. For a component deployed on a VM, this parameter is the software package address. For a component deployed based on a container, this parameter is the image address or component name:v\${index} . The latter indicates that the component source code or the image automatically built using the software package will be used.
auth	Yes	String	Authentication mode. Value: iam or none. Default value: iam.
version	No	String	Version number.
properties	No	Map<String, Object>	Property information.

Table 12-152 external_accesses

Parameter	Mandatory	Type	Description
protocol	Yes	String	Protocol. Value: http or https.
address	Yes	String	Access address.
forward_port	Yes	Integer	Port number.

Table 12-153 refer_resources

Parameter	Mandatory	Type	Description
id	Yes	String	Resource ID. If type is set to ecs , the value of this parameter must be Default .
type	Yes	String	Basic resources: cce, as, and ecs. Optional resources: rds, dcs, elb, cse, and other services.
refer_alias	No	String	Application alias, which is provided only in DCS scenario. Value: "distributed_session", "distributed_cache", "distributed_session, distributed_cache". Default value: "distributed_session, distributed_cache".
parameters	No	Map<String, Object>	Reference resource parameter. NOTICE - When type is set to cce, this parameter is mandatory. You need to specify the namespace of the cluster where the component is to be deployed. Example: {"namespace": "default"}. - When type is set to ecs, this parameter is mandatory. You need to specify the hosts where the component is to be deployed. Example: {"hosts": ["04d9f887-9860-4029-91d1-7d3102903a69", "04d9f887-9860-4029-91d1-7d3102903a70"] }.

Table 12-154 container_spec

Parameter	Mandatory	Type	Description
containers	No	Array of objects	Container information. See Table 12-155 .
type	No	String	Workload type. • deployment • statefulset

Table 12-155 container

Parameter	Mandatory	Type	Description
name	No	String	Container name.
size	No	Object	Container size. See Table 12-156 .
env	No	Array of objects	Environment variable. See Table 12-135 .
storage	No	Array of objects	Data storage configuration. See Table 12-136 .
lifecycle	No	Object	Lifecycle. See Table 12-138 .
probes	No	Object	Health check. See Table 12-140 .

Table 12-156 size

Parameter	Man dato ry	Type	Description
id	No	String	Resource specifications, which can be obtained by using the API in Obtaining All Supported Flavors of Application Resources. If you need to customize resource specifications, the format is CUSTOM-xxG:xxm-xxm:xxGi-xxGi. Where: • xxG: storage capacity allocated to a component instance. It is a reserved field. You can set it to a fixed number. • xxm-xxm: the maximum and minimum number of CPU cores allocated to a component instance. • xxGi-xxGi: the maximum and minimum memory allocated to a component instance. For example, CUSTOM-10G:0.5m-0.25m:1.6Gi-0.8Gi indicates that the maximum number of CPU cores allocated to a component instance is 0.5, the minimum number of CPU cores is 0.25, the maximum memory is 1.6 Gi, and the minimum memory is 0.8 Gi.

Response

Table 12-157 Response parameters

Parameter	Type	Description
job_id	String	Job ID, which is used to query information about the created job.
instance_id	String	Component instance ID.

Example Request

Create component instance **component-instance-name**. Select the environment whose ID is **6e763000-9128-4a9d-adea-34c42cc5344d** as the running environment. Select the **demo.jar** package in the myapp bucket of OBS as the deployment component. Select the CCE resource whose ID is **b6862a62-d916-11e9-bdf1-0255ac101fd9** and the ELB resource whose ID is **8c0a45cc-626f-4d65-8257-507ee059aa9a** as the deployment resources.

```
{
  "name": "component-instance-name",
  "environment_id": "6e763000-9128-4a9d-adea-34c42cc5344d",
  "flavor_id": "MICRO-5G:0.5m:1G",
  "replica": 1,
  "artifacts": {
    "container-name": {
      "storage": "obs",
      "type": "package",
      "url": "obs://myapp/demo.jar",
      "properties": {
        "bucket": "myapp",
        "key": "demo.jar",
        "endpoint": "https://obs.region_id.external_domain_name.com"
      },
      "auth": "iam"
    }
  },
  "version": 1,
  "description": "instance desc",
  "configuration": {
    "env": [
      {
        "name": "log-level",
        "value": "warn"
      }
    ]
  },
  "refer_resources": [
    {
      "id": "b6862a62-d916-11e9-bdf1-0255ac101fd9",
      "type": "cce",
      "parameters": {
        "namespace": "default"
      }
    },
    {
      "id": "8c0a45cc-626f-4d65-8257-507ee059aa9a",
      "type": "elb"
    }
  ]
}
```

Example Response

```
{
  "instance_id": "89f5baf5-efe4-4f12-9c0d-734d2af5a184",
  "job_id": "JOB66761060-f209-407c-a093-4df6f531b9dc"
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

12.1.5.2 Obtaining All Component Instances

Function

This API is used to obtain all instances of a component.

URI

GET /v2/{project_id}/cas/applications/{application_id}/components/{component_id}/instances

Table 12-158 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
application_id	Yes	String	Application ID. See Obtaining All Applications .
component_id	Yes	String	Component ID. See Obtaining All Components of an Application .

Table 12-159 Query parameters

Parameter	Mandatory	Type	Description
maxWaitTime	No	int	User-defined overall startup time.
limit	No	Integer	Number of records to be displayed. The value is 1000 or ranges from 0 to 100. If the specified value is not within the range, value 10 is assigned. In the non-pagination scenario, the value is 1000 . In the pagination scenario, the value ranges from 0 to 100.
offset	No	Integer	Offset.
order_by	No	String	Sorting field. By default, query results are sorted by creation time. Enumerated values: create_time, name, version, and update_time.

Parameter	Mandatory	Type	Description
order	No	String	Sorting order. • desc (default) • asc

Request

Table 12-160 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .

Response

Table 12-161 Response parameters

Parameter	Type	Description
count	Integer	Total number of instances.
instances	Array of objects	Instance parameters. See Table 12-162 .

Table 12-162 instance

Parameter	Type	Description
id	String	Component instance ID.
application_id	String	Application ID.
application_name	String	Application name.
component_id	String	Component ID.

Parameter	Type	Description
component_name	String	Component name.
name	String	Component instance name.
environment_id	String	Component environment ID.
environment_name	String	Environment name.
platform_type	String	Platform type. Value: cce or vmapp.
artifacts	Map<String, Object>	Artifact. key indicates the component name. In the Docker container scenario, key indicates the container name. See Table 12-163 .
creator	String	Creator.
version	String	Component version.
create_time	Integer	Creation time.
update_time	Integer	Update time.
external_accesses	Array of objects	Access mode. See Table 12-164 .
status_detail	Object	Status details. See Table 12-165 .

Table 12-163 artifact

Parameter	Type	Description
storage	String	Storage mode. Value: swr (software repository), swr (image), obs, or codearts.
type	String	Type. • VM-based deployment supports package. • Container-based deployment supports package and image.
url	String	Software package or image address.
auth	String	Authentication mode. Value: iam or none. Default value: iam.
version	String	Version number.
properties	Map<String, String>	Property information.

Table 12-164 external_accesses

Parameter	Type	Description
id	String	Access type ID.
protocol	String	Protocol.
address	String	Access address.
forward_port	Integer	Port for listening to an application component process.
type	String	Type.
status	String	Status.
create_time	Integer	Creation time.
update_time	Integer	Update time.

Table 12-165 status_detail

Parameter	Type	Description
enterprise_project_id	String	Enterprise project ID.
status	String	Instance status.
available_replica	Integer	Number of normal instance replicas.
replica	Integer	Number of instance replicas.
fail_detail	String	Failure description.
last_job_id	String	Latest job ID.
last_job_statuses	String	Latest job status.

Example Request

None

Example Response

```
{
  "instances": [
    {
      "id": "11eddb33-140b-4e51-b1e2-6ec265373ca3",
      "application_id": "4d92833a-fa05-4fc0-a761-e67620022310",
      "application_name": "test-app",
      "component_id": "1cfdda6f-84cd-4ead-8e09-628fabf662e2",
      "component_name": "test-com",
      "name": "test-micro-test-env-7iafjk",
```

```
{
  "environment_id": "61b81021-21d5-42f3-b80e-0b6bd10dbf7d",
  "environment_name": "test-env",
  "platform_type": "cce",
  "version": "1.0.0",
  "artifacts": {
    "test-micro": {
      "storage": "obs",
      "type": "package",
      "url": "obs://myapp/demo.jar",
      "properties": {
        "bucket": "myapp",
        "key": "demo.jar",
        "endpoint": "https://obs.region_id.external_domain_name.com"
      },
      "auth": "iam",
      "version": "1.0.0",
    }
  },
  "create_time": 1610331819996,
  "update_time": 1610331830398,
  "creator": "test_user",
  "status_detail": {
    "status": "RUNNING",
    "replica": 1,
    "available_replica": 1,
    "fail_detail": null,
    "last_job_id": "JOB7bc0366c-dc43-41f9-b65f-8d6078038488",
    "last_job_status": "SUCCEEDED",
    "enterprise_project_id": "0"
  }
},
"count": 1
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

12.1.5.3 Querying the Operations Performed on a Component Instance

Function

This API is used to query the operations performed on a component instance.

URI

POST /v2/{project_id}/cas/applications/{application_id}/components/{component_id}/instances/{instance_id}/action

Table 12-166 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
application_id	Yes	String	Application ID. See Obtaining All Applications .
component_id	Yes	String	Component ID. See Obtaining All Components of an Application .
instance_id	Yes	String	Component instance ID. See Obtaining All Component Instances .

Request

Table 12-167 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .

Table 12-168 Request body parameters

Parameter	Mandatory	Type	Description
action	Yes	String	Operation. Value: start, stop, restart, scale, or rollback. - start - stop - restart - scale - rollback
parameters	No	Object	Operation parameter, which is mandatory when action is set to scale or rollback . See Table 12-169 .

Table 12-169 parameters

Parameter	Mandatory	Type	Description
replica	No	Integer	Number of instances, which is mandatory when action is set to scale .
hosts	No	Array	ECS ID list. This parameter specifies the ECS deployed during VM scaling, and is mandatory for scaling.
version	No	String	Version number, which is mandatory when action is set to rollback . You can obtain the version number by using the API in Obtaining Component Instance Snapshots .

Response

Table 12-170 Response parameters

Parameter	Type	Description
job_id	String	Job ID.

Example Request

Scale component instances and change the number of component instances to 4.

```
{  
  "action": "scale",  
}
```

```
"parameters": {  
  "replica": 4  
}
```

Example Response

```
{  
  "job_id": "JOB66761060-f209-407c-a093-4df6f531b9dc"  
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

12.1.5.4 Modifying a Component Instance

Function

This API is used to modify a component instance.

URI

PUT /v2/{project_id}/cas/applications/{application_id}/components/
{component_id}/instances/{instance_id}

Table 12-171 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
application_id	Yes	String	Application ID. See Obtaining All Applications .
component_id	Yes	String	Component ID. See Obtaining All Components of an Application .
instance_id	Yes	String	Component instance ID. See Obtaining All Component Instances .

Request

Table 12-172 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .

Table 12-173 Request body parameters

Parameter	Mandatory	Type	Description
version	Yes	String	Application component version that meets version semantics. Example: 1.0.1.
flavor_id	No	String	Resource specifications, which can be obtained by using the API in Obtaining All Supported Flavors of Application Resources. If you need to customize resource specifications, the format is CUSTOM-xxG:xxm-xxm:xxGi-xxGi. Where: • xxG: storage capacity allocated to a component instance. It is a reserved field. You can set it to a fixed number. • xxm-xxm: the maximum and minimum number of CPU cores allocated to a component instance. • xxGi-xxGi: the maximum and minimum memory allocated to a component instance. For example, CUSTOM-10G:0.5m-0.25m:1.6Gi-0.8Gi indicates that the maximum number of CPU cores allocated to a component instance is 0.5, the minimum number of CPU cores is 0.25, the maximum memory is 1.6 Gi, and the minimum memory is 0.8 Gi.

Parameter	Mandatory	Type	Description
artifacts	No	Map<String, Object>	Artifact. key indicates the component name. In the Docker container scenario, key indicates the container name. See Table 12-174 .
configuration	No	Object	Configuration parameters, such as environment variables, deployment configurations, and O&M monitoring. By default, this parameter is left blank. See Table 12-134 in Creating an Application Component Instance .
description	No	String	Description. The value can contain up to 128 characters.
external_accesses	No	Array of objects	Access mode. See Table 12-175 .
refer_resources	No	Array of objects	Deployed resources. See Table 12-176 .

Table 12-174 artifact

Parameter	Mandatory	Type	Description
storage	Yes	String	Storage mode. Value: swr (software repository), swr (image), obs, or codearts.
type	Yes	String	Type. - VM-based deployment supports package. - Container-based deployment supports package and image.
url	Yes	String	Software package or image address. For a component deployed on a VM, this parameter is the software package address. For a component deployed based on a container, this parameter is the image address or component name:v\${index} . The latter indicates that the component source code or the image automatically built using the software package will be used.
auth	Yes	String	Authentication mode. Value: iam or none. Default value: iam.

Parameter	Mandatory	Type	Description
version	No	String	Version number.
properties	No	Map<String,String>	Property information.

Table 12-175 external_accesses

Parameter	Mandatory	Type	Description
id	No	String	Access type ID.
protocol	Yes	String	Protocol. Value: http or https.
address	Yes	String	Access address. Example: www.example.com.
forward_port	Yes	Integer	Port for listening to an application component process.
type	No	String	Type.
status	No	String	Status.
create_time	No	Integer	Creation time.
update_time	No	Integer	Update time.

Table 12-176 refer_resources

Parameter	Mandatory	Type	Description
id	Yes	String	Resource ID.
type	Yes	String	Resource type. Example: rds or dcs.
refer_alias	No	String	Application alias. This parameter is available only when type is set to dcs . Value: "distributed_session", "distributed_cache", or "distributed_session, distributed_cache" (default).
parameters	No	Map<String,Object>	Reference resource parameter.

Response

Table 12-177 Response parameter

Parameter	Type	Description
job_id	String	Job ID, which is used to query information about the created job.

Example Request

Modify the application component instance. Change the artifact to the **test-micro** image package in the SWR image repository **swr.roma-dev-1.roma.development.com/ns1/examples:v2**. Change the deployment resources to the CCE resource whose ID is **523498f1-36c4-11eb-ae36-0255ac1000c2** and the professional microservice engine **cse**.

```
{
  "description": "",
  "artifacts": {
    "test-micro": {
      "storage": "swr",
      "type": "image",
      "url": "swr.roma-dev-1.roma.development.com/ns1/examples:v2",
      "auth": "iam",
      "version": "1.0.0",
      "properties": {}
    }
  },
  "refer_resources": [
    {
      "id": "523498f1-36c4-11eb-ae36-0255ac1000c2",
      "type": "cce",
      "parameters": {
        "namespace": "default"
      }
    },
    {
      "id": "default",
      "type": "cse"
    }
  ],
  "version": "1.0.2"
}
```

Example Response

```
{
  "job_id": "JOB66761060-f209-407c-a093-4df6f531b9dc"
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request

Status Code	Description
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

12.1.5.5 Deleting a Component Instance

Function

This API is used to delete a component instance.

URI

DELETE /v2/{project_id}/cas/applications/{application_id}/components/{component_id}/instances/{instance_id}

Table 12-178 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
application_id	Yes	String	Application ID. See Obtaining All Applications .
component_id	Yes	String	Component ID. See Obtaining All Components of an Application .
instance_id	Yes	String	Component instance ID. See Obtaining All Component Instances .

Table 12-179 Query parameters

Parameter	Mandatory	Type	Description
force	No	boolean	Whether to forcibly delete the component.

Request

Table 12-180 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .

Response

Table 12-181 Response parameter

Parameter	Type	Description
job_id	String	Job ID

Example Request

None

Example Response

```
{
  "job_id": "JOB66761060-f209-407c-a093-4df6f531b9dc"
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

12.1.5.6 Querying Instance Details Based on the Instance ID

Function

This API is used to query instance details based on the instance ID.

URI

GET /v2/{project_id}/cas/applications/{application_id}/components/{component_id}/instances/{instance_id}

Table 12-182 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
application_id	Yes	String	Application ID. See Obtaining All Applications .
component_id	Yes	String	Component ID. See Obtaining All Components of an Application .
instance_id	Yes	String	Component instance ID. See Obtaining All Component Instances .

Request

Table 12-183 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .

Response

Table 12-184 Response parameters

Parameter	Type	Description
id	String	Component instance ID.
name	String	Component instance name.
environment_id	String	Component environment ID.
platform_type	String	Platform type. Value: cce or vmapp.
description	String	Instance description.
flavor_id	String	Resource flavor.
artifacts	Map<String, Object>	Artifact. key indicates the component name. In the Docker container scenario, key indicates the container name. See Table 12-185 .
version	String	Component version.
configuration	Map<String, Object>	Component configurations, such as environment variables.
creator	String	Creator.
create_time	Integer	Creation time.
update_time	Integer	Update time.
external_accesses	Array of objects	Access mode. See Table 12-186 .
refer_resources	Array of objects	Deployed resources. See Table 12-187 .
status_detail	Object	Status details. See Table 12-188 .

Table 12-185 artifact

Parameter	Type	Description
storage	String	Storage mode. Value: swr (software repository), swr (image), obs, or codearts.
type	String	Type. - VM-based deployment supports package. - Container-based deployment supports package and image.

Parameter	Type	Description
url	String	Software package or image address.
auth	String	Authentication mode. Value: iam or none. Default value: iam.
version	String	Version number.
properties	Map<String,String>	Property information.

Table 12-186 external_accesses

Parameter	Type	Description
id	String	Access type ID.
protocol	String	Protocol.
address	String	Access address.
forward_port	Integer	Port for listening to an application component process.
type	String	Type.
status	String	Status.
create_time	Integer	Creation time.
update_time	Integer	Update time.

Table 12-187 refer_resources

Parameter	Type	Description
id	String	Resource ID.
type	String	Basic resources: cce, as, and ecs. Optional resources: rds, dcs, elb, cse, and other services.
refer_alias	String	Application alias, which is provided only in DCS scenario. Value: "distributed_session", "distributed_cache", or "distributed_session, distributed_cache". Default value: "distributed_session, distributed_cache".
parameters	Map<String,Object>	Reference resource parameter.

Table 12-188 status_detail

Parameter	Type	Description
enterprise_project_id	String	Enterprise project ID.
status	String	Instance status. - INITIALIZING - UPGRADING - FAILED - RUNNING - DELETING - STARTING - STOPPING - STOPPED - RESTARTING - PENDING - PARTIALLY_FAILED
available_replica	Integer	Number of normal instance replicas.
replica	Integer	Number of instance replicas.
fail_detail	String	Failure description.
last_job_id	String	Latest job ID.
last_job_statuses	String	Latest job status.

Example Request

None

Example Response

```
{
  "id": "11eddb33-140b-4e51-b1e2-6ec265373ca3",
  "name": "test-micro-test-env-7iafjk",
  "environment_id": "61b81021-21d5-42f3-b80e-0b6bd10dbf7d",
  "platform_type": "cce",
  "flavor_id": "CUSTOM-10G:250m-250m:0.5Gi-0.5Gi",
  "artifacts": {
    "test-micro": {
      "storage": "swr",
      "type": "image",
      "url": "swr.region_id.development.com/ns/examples:v2",
      "auth": "iam",
      "version": "1.0.0",
      "properties": {}
    }
  },
  "version": "1.0.2",
}
```

```
"description": "",
"configuration": {
  "env": [
    {
      "name": "PAAS_PROJECT_ID",
      "value": "e7d2e9c589e5445e808a8ff0d1235aca",
      "internal": true
    }
  ],
  "lifecycle": {
    "post-start": null,
    "pre-stop": null
  },
  "scheduler": {
    "affinity": {
      "az": [],
      "node": [],
      "application": []
    },
    "anti-affinity": {
      "az": [],
      "node": [],
      "application": []
    },
    "isSynchronized": true
  },
  "strategy": {
    "upgrade": "RollingUpdate",
    "spec": {
      "maxSurge": 0,
      "maxUnavailable": 1
    }
  }
},
"creator": null,
"create_time": 1610331819996,
"update_time": 1610522676792,
"external_accesses": null,
"refer_resources": [
  {
    "id": "523498f1-36c4-11eb-ae36-0255ac1000c2",
    "type": "cce",
    "parameters": {
      "namespace": "default"
    }
  },
  {
    "id": "default",
    "type": "cse"
  }
],
"status_detail": {
  "status": "RUNNING",
  "replica": 1,
  "available_replica": 1,
  "fail_detail": null,
  "last_job_id": "JOB16c33466-41e6-4c5b-a9ac-2059253a1e5c",
  "last_job_status": "SUCCEEDED",
  "enterprise_project_id": "0"
}
```

Status Code

Status Code	Description
200	OK

Status Code	Description
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

12.1.5.7 Obtaining Component Instance Snapshots

Function

This API is used to obtain component instance snapshots.

Instance snapshots refer to the instance information about the source version. They can be used for version rollback.

URI

GET /v2/{project_id}/cas/applications/{application_id}/components/{component_id}/instances/{instance_id}/snapshots

Table 12-189 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
application_id	Yes	String	Application ID. See Obtaining All Applications .
component_id	Yes	String	Component ID. See Obtaining All Components of an Application .
instance_id	Yes	String	Component instance ID. See Obtaining All Component Instances .

Table 12-190 Query parameters

Parameter	Mandatory	Type	Description
limit	No	Integer	Number of records to be displayed. The value is 1000 or ranges from 0 to 100. If the specified value is not within the range, value 10 is assigned. In the non-pagination scenario, the value is 1000 . In the pagination scenario, the value ranges from 0 to 100.
offset	No	Integer	Offset.
order_by	No	String	Sorting field. By default, query results are sorted by creation time.
order	No	String	Sorting order. • desc (default) • asc

Request

Table 12-191 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .

Response

Table 12-192 Response parameters

Parameter	Type	Description
count	Integer	Number of snapshots.
snapshots	Array of objects	Snapshot parameters. See Table 12-193 .

Table 12-193 snapshot

Parameter	Type	Description
create_time	Integer	Creation time.
description	String	Description.
instance_id	String	Component instance ID.
version	String	Version number.

Example Request

None

Example Response

```
{
  "snapshots": [
    {
      "version": "1.0.0",
      "description": "{\"action\":\"New Version: 1.0.0\"}",
      "instance_id": "11eddb33-140b-4e51-b1e2-6ec265373ca3",
      "create_time": 1610331830841
    }
  ],
  "count": 1
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

12.1.6 Deployment Jobs

12.1.6.1 Obtaining Job Details

Function

This API is used to obtain job details.

URI

GET /v2/{project_id}/cas/jobs/{job_id}

Table 12-194 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Tenant's project ID. See Obtaining a Project ID .
job_id	Yes	String	Deployment job ID, which is job_id in the response parameter in Creating an Application Component Instance .

Table 12-195 Query parameters

Parameter	Mandatory	Type	Description
instance_id	No	String	Component instance ID. See Obtaining All Component Instances .
limit	No	Integer	Number of specified records.
offset	No	Integer	Offset.
desc	No	String	Whether query results are displayed in descending order. Value: true (descending order) or false (ascending order).

Request

Table 12-196 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .

Response

Table 12-197 Response parameters

Parameter	Type	Description
task_count	Integer	Number of tasks.
job	Object	Job parameters. See Table 12-198 .
tasks	Array<Object>	Task parameters. See Table 12-199 .

Table 12-198 job

Parameter	Type	Description
CREATED_BY	String	Creator.
EXECUTION_STATUS	String	Execution status. • SUBMITTED • SCHEDULED • RUNNING • FAILED • SUCCEEDED • CANCELLED • PARTIALLY_FAILED • ROLLBACKING • ROLLBACKED • UNKNOWN
JOB_DESC	String	Job description.
JOB_ID	String	Job ID.
JOB_NAME	String	Job name.
JOB_TYPE	String	Type. • PROVISION • DEPROVISION • DAILY_OPERATION • SYSTEM_INNER • UNKNOWN
ORDER_ID	String	Order ID.
PROJECT_ID	String	Tenant's project ID.
SERVICE_INSTANCE_ID	String	Instance ID.

Table 12-199 task

Parameter	Type	Description
CREATED_AT	String	Creation time.
LAST_HEALTH_CHECK	String	Health check time.
MESSAGES	String	Message.
OWNER_ID	String	Creator ID.
TASK_ID	String	Task ID.
TASK_INDEX	Integer	Task index.
TASK_NAME	String	Task name.
TASK_STATUS	String	Task status. • SUBMITTED • SCHEDULED • RUNNING • FAILED • SUCCEEDED • CANCELLED • PARTIALLY_FAILED • SKIPPED • ROLLBACKING • ROLLBACKED • UNKNOWN
TASK_TYPE	String	Task type. • kube: Kubernetes task • restful: RESTful HTTP request task • sleep: sleep waiting task • echo: output task • context • sub-job

Example Request

None

Example Response

```
{
  "job": {
    "JOB_ID": "JOB474362ef-091d-429f-9b3e-c9246b37dee6",
    "JOB_NAME": "provision/provMain",
```

```
"JOB_TYPE": "PROVISION",
"SERVICE_INSTANCE_ID": "429a35a0-dee8-4648-86f6-d1f8bc74ba52",
"PROJECT_ID": "063a3158b1a34710b36ad208b9497d00",
"ORDER_ID": "",
"JOB_DESC": "provision/provMain",
"EXECUTION_STATUS": "FAILED",
"CREATED_BY": "585cc5c16cda4954b28d4eb460f06261"
},
"tasks": [
  {
    "MESSAGES": "{}",
    "TASK_ID": "Task-bc64b46c-1688-4787-b7ff-87cb53b346ea",
    "TASK_TYPE": "echo",
    "TASK_INDEX": 1,
    "TASK_NAME": "SEPARATOR_PREPARE_RESOURCES",
    "TASK_STATUS": "SUCCEEDED",
    "LAST_HEALTH_CHECK": "2020-02-04T04:01:03.344Z",
    "CREATED_AT": "2020-02-04T04:01:03.344Z",
    "OWNER_ID": "cas-mgr-69bb87d7c-wpqwb:30114"
  },
  {
    "MESSAGES": "{}",
    "TASK_ID": "Task-15ec252f-bca0-401e-ae5b-798a639df92e",
    "TASK_TYPE": "echo",
    "TASK_INDEX": 4,
    "TASK_NAME": "SEPARATOR_BUILDING",
    "TASK_STATUS": "SUCCEEDED",
    "LAST_HEALTH_CHECK": "2020-02-04T04:01:03.346Z",
    "CREATED_AT": "2020-02-04T04:01:03.345Z",
    "OWNER_ID": "cas-mgr-69bb87d7c-wpqwb:30114"
  },
  {
    "MESSAGES": "{}",
    "TASK_ID": "Task-52a5d1ae-2c54-46ec-8134-53e7b6a45fb7",
    "TASK_TYPE": "restful",
    "TASK_INDEX": 6,
    "TASK_NAME": "TN00091",
    "TASK_STATUS": "SUCCEEDED",
    "LAST_HEALTH_CHECK": "2020-02-04T04:01:08.924Z",
    "CREATED_AT": "2020-02-04T04:01:03.346Z",
    "OWNER_ID": "cas-mgr-69bb87d7c-wpqwb:30114"
  },
  {
    "MESSAGES": "{}",
    "TASK_ID": "Task-a149b585-d31b-4c72-a665-d820efcb0bf9",
    "TASK_TYPE": "restful",
    "TASK_INDEX": 7,
    "TASK_NAME": "TN00094",
    "TASK_STATUS": "SUCCEEDED",
    "LAST_HEALTH_CHECK": "2020-02-04T04:01:08.961Z",
    "CREATED_AT": "2020-02-04T04:01:08.925Z",
    "OWNER_ID": "cas-mgr-69bb87d7c-wpqwb:30114"
  },
  {
    "MESSAGES": "{
      \"Message\": \"ECC00011\",
      \"Detail\": \"Building failed.\"
    }",
    "TASK_ID": "Task-a72fe8d8-8cc1-42b5-a097-0cf9e29692a7",
    "TASK_TYPE": "restful",
    "TASK_INDEX": 9,
    "TASK_NAME": "TN00097",
    "TASK_STATUS": "FAILED",
    "LAST_HEALTH_CHECK": "2020-02-04T04:01:38.975Z",
    "CREATED_AT": "2020-02-04T04:01:08.962Z",
    "OWNER_ID": "cas-mgr-69bb87d7c-wpqwb:30114"
  }
],
```

```
"task_count": 5  
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request
404	Not Found
500	Internal Server Error

Error Code

See [ServiceStage Error Codes](#).

12.2 Build APIs

12.2.1 Obtaining the Build Job List Based on the Component Name

Function

This API is used to obtain the build job list based on the component name.

URI

GET /assembling/v1/jobs?
page={page}&per_page={per_page}&job_name={job_name}

[Table 12-200](#) describes the parameters.

Table 12-200 Path parameters

Parameter	Mandatory	Type	Description
page	Yes	Integer	Page index, indicating the page from which the query starts.
per_page	Yes	Integer	Page size, indicating the number of records on a page.
job_name	No	String	Build job name, which is the same as the custom component name. See Obtaining All Components of an Application .

Request

Table 12-201 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .

Response

Table 12-202 Response parameters

Parameter	Type	Description
list	Array of object	Build phase structure. See Table 12-203 .
total	Integer	Number of build phases.
page	Integer	Page size, indicating the number of records on a page.
per_page	Integer	Page index, indicating the page from which the query starts.

Table 12-203 list

Parameter	Type	Description
job_id	String	Build job ID.
job_name	String	Build job name, which is the same as the custom component name.

Example Request

None

Example Response

```
{
  "total": 12,
  "per_page": 6,
```

```
"page": 1,
"list": [
  {
    "job_id": "swpfk3wtzdnjlyely4rf40k52zyuq5s4lha9ch8f",
    "job_name": "test1"
  }
]
```

Status Code

Status Code	Description
200	OK
400	Bad Request

Error Code

The error code format is SVCSTG.CPE.[Error_ID], for example, SVCSTG.CPE.5004401. For details, see [ServiceStage Error Codes](#).

12.2.2 Obtaining the Build Job Instance List Based on the Build Job ID

Function

This API is used to obtain the build job instance list based on the build job ID.

URI

GET /assembling/v1/jobs/{job_id}/builds?name={job_name}

[Table 12-204](#) describes the parameters.

Table 12-204 Path parameters

Parameter	Mandatory	Type	Description
job_id	Yes	String	Build job ID. See Obtaining the Build Job List Based on the Component Name .
job_name	Yes	String	Build job name. See Obtaining the Build Job List Based on the Component Name .

Request

Table 12-205 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .

Response

Table 12-206 Response parameters

Parameter	Type	Description
list	Array of object	Build phase structure. See Table 12-207 .
total	Integer	Number of build phases.
page	Integer	Start page number.
per_page	Integer	Page size.
finished	Integer	Number of completed build instances.
successful	Integer	Number of successful build instances.

Table 12-207 list

Parameter	Type	Description
build_id	String	Build job instance ID.
build_state	String	Build instance status. • Cancel • Failed • Successful • Timeout
duration	Integer	Build duration, in seconds.

Example Request

None

Example Response

```
{
  "total": 1,
  "per_page": 20,
  "page": 1,
  "finished": 1,
  "successful": 1,
  "list": [
    {
      "build_id": "0f7fqckzmjcfi0f7yxhx91l2rwrifyzd2poac90b5",
      "build_state": "Successful",
      "duration": 1181
    }
  ]
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request

Error Code

The error code format is SVCSTG.CPE.[Error_ID], for example, SVCSTG.CPE.5004401. For details, see [ServiceStage Error Codes](#).

12.2.3 Obtaining the Build Phase List Based on the Build Job Instance ID

Function

This API is used to obtain the build phase list based on the build job instance ID.

URI

GET /assembling/v1/builds/{build_id}/phases

[Table 12-208](#) describes the parameters.

Table 12-208 Path parameters

Parameter	Mandatory	Type	Description
build_id	Yes	String	Build job instance ID. See Obtaining the Build Job Instance List Based on the Build Job ID .

Request

Table 12-209 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .

Response

Table 12-210 Response parameters

Parameter	Type	Description
list	Array of object	Build phase structure. See Table 12-211 .
total	Integer	Number of build phases.

Table 12-211 list

Parameter	Type	Description
id	String	Build phase ID.

Parameter	Type	Description
name	String	Build phase name. • Init • Download • PreProcess • Build • Artifact • PostProcess
state	String	Build phase status. • NotStarted • Queuing • Building • Pushing: The package is being pushed. • Successful • Failed • Timeout • Cancel • Deleted

Example Request

None

Example Response

```
{
  "total": 6,
  "list": [
    {
      "id": 1627984,
      "name": "Init",
      "state": "Successful"
    },
    {
      "id": 1627985,
      "name": "Download",
      "state": "Successful"
    },
    {
      "id": 1627986,
      "name": "PreProcess",
      "state": "Successful"
    },
    {
      "id": 1627987,
      "name": "Build",
      "state": "Successful"
    },
    {
      "id": 1627988,
      "name": "Artifact",

```

```
{
  "state": "Successful"
},
{
  "id": 1627989,
  "name": "PostProcess",
  "state": "Successful"
}
]
```

Status Code

Status Code	Description
200	OK
400	Bad Request

Error Code

The error code format is SVCSTG.CPE.[Error_ID], for example, SVCSTG.CPE.5004401. For details, see [ServiceStage Error Codes](#).

12.2.4 Obtaining Build Logs Based on the Build Job Instance ID

Function

This API is used to obtain build job logs based on the build job instance ID.

URI

GET /assembling/v1/builds/{build_id}/logs?from={from}

[Table 12-212](#) describes the parameters.

Table 12-212 Path parameters

Parameter	Mandatory	Type	Description
build_id	Yes	String	Build job instance ID. See Obtaining the Build Job Instance List Based on the Build Job ID .
from	Yes	Integer	Start position for obtaining logs, for example, 0.

Request

Table 12-213 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Message body type (format). Default value: application/json;charset=utf8.
X-Auth-Token	Yes	String	API calling can be authenticated using a token or AK/SK. If you use a token, this parameter is mandatory and must be set to the token. For details about how to obtain a token, see Obtaining a User Token .

Response

Table 12-214 Response parameters

Parameter	Type	Description
buildlog	String	Returned build log.
next	Integer	End position of the returned log string.
phase_states	String	Build phase status. • NotStarted • Queuing • Building • Pushing: The package is being pushed. • Successful • Failed • Timeout • Cancel • Deleted
phases	String	Build phase name. • Init • Download • PreProcess • Build • Artifact • PostProcess

Example Request

None

Example Response

```
{
  "buildlog": "Starting...",
  "next": 25524,
  "phase_states": "Successful,Successful,Successful,Successful,Successful,Successful",
  "phases": "Init,Download,PreProcess,Build,Artifact,PostProcess"
}
```

Status Code

Status Code	Description
200	OK
400	Bad Request

Error Code

The error code format is SVCSTG.CPE.[Error_ID], for example, SVCSTG.CPE.5004401. For details, see [ServiceStage Error Codes](#).

12.3 Querying Configurations

Function

This API is used to query configurations in the configuration center.

Call Method

- For a professional microservice engine, see [Calling APIs](#).
- For an exclusive microservice engine, perform the following steps:
 - a. Log in to ServiceStage and choose **Cloud Service Engine > Engines**.
 - b. Select the microservice engine of the API to be called from the **Microservice Engine** drop-down list in the upper part of the page.
 - c. In the **Service Discovery and Configuration** area, view or click  to copy the configuration center address.
 - d. Access the API by referring to [Calling APIs](#). In **Request URI**, replace **{Endpoint}** with the obtained configuration center address.

URI

GET /v3/{project_id}/configuration/items

Table 12-215 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	- For a professional microservice engine, enter the tenant project ID. The value contains 1 to 64 characters. You can also enter the fixed value default. - For an exclusive microservice engine, enter the fixed value default.

Table 12-216 Query parameters

Parameter	Mandatory	Type	Description
dimensionsInfo	Yes	String	Service information, which consists of the service name (mandatory), application (mandatory), and version (optional). The combination format is as follows (note that @ and # must be escaped when being added to the HTTP URL. @ is escaped to %40, and # is escaped to %23): {serviceName}@{appId}#{version} or {serviceName}@{appId} The value cannot contain spaces or the following special characters: \$%^&+/\
keyFilter	No	String	key filter criteria. A regular expression is supported. If special characters are contained, escape them.
revision	No	String	Version of a configuration item. This parameter is used to determine the difference between the local cache configuration and the configuration in the configuration center. - Input 0 for the first time. - If the input revision value is the same as the current revision value, status code 304 is returned, but no body is returned. If they are inconsistent, both revision values are returned. - If revision is required in the next request, use the value of version in revision in the returned body.

Request

Table 12-217 Request header parameter

Parameter	Mandatory	Type	Description
x-environment	No	String	Environment whose configurations need to be queried. Value: development, testing, acceptance, or production. Other values are regarded as null.
Authorization	Yes	String	This parameter is mandatory if security authentication is enabled for an exclusive microservice engine. Otherwise, this parameter is not required. The token of an exclusive microservice engine with security authentication enabled is in the following format: Authorization:Bearer {Token} For details about how to obtain the token, see Obtaining the User Token of an Exclusive Microservice Engine.

Response

If the input revision value is the same as the current revision value, no response message is returned.

[Table 12-218](#) describes the response parameters.

Table 12-218 Response parameters

Parameter	Type	Description
{dimension} (The parameter name and number of parameters are not fixed.)	Map<String, String>	Configuration content.
revision	Table 12-219	Revision.

Table 12-219 revision

Parameter	Type	Description
version	String	Version.

Example Request

```
GET /v3/default/configuration/items?dimensionsInfo=service%40app%231.0.0
```

Example Response

```
{
  "service@app": {
    "k1": "v1"
  },
  "service@app#1.0.0": {
    "k2": "v2"
  }
}
```

Status Code

See [Status Codes](#).

Error Code

See [CSE Error Codes](#).

12.4 Creating a Dependency Between Services

Function

This API is used to create a dependency between services. For **consumer**, **version** must be a confirmed version and **serviceName** cannot be *. **consumer** must be an existing service while **provider** can be a service that has not been created.

Call Method

1. Log in to ServiceStage and choose **Cloud Service Engine > Engines**.
2. Select the target microservice engine from the **Microservice Engine** drop-down list in the upper part of the page.
3. In the **Service Discovery and Configuration** area, view or click  to copy the service center address.
4. Call the API by referring to [Calling APIs](#). In **Request URI**, replace **{Endpoint}** with the obtained service center address.

URI

```
PUT /v4/{project_id}/registry/dependencies
```

Table 12-220 Path parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Fixed value: default.

Request

Table 12-221 Request header parameter

Parameter	Mandatory	Type	Description
Authorization	Yes	String	This parameter is mandatory if security authentication is enabled for an exclusive microservice engine. Otherwise, this parameter is not required. The token of an exclusive microservice engine with security authentication enabled is in the following format: Authorization:Bearer {Token} For details about how to obtain the token, see Obtaining the User Token of an Exclusive Microservice Engine.

Table 12-222 Request body parameters

Parameter	Mandatory	Type	Description
dependencies	Yes	Array of objects	Set of rules. See Table 12-223 . The array can contain 1–100 objects.

Table 12-223 MicroServiceDependency

Parameter	Mandatory	Type	Description
consumer	Yes	Object	Consumer microservice. See DependencyMicroService .
providers	No	Array of objects	Provider microservice. See DependencyMicroService .

Response

None

Example Request

```
{
  "dependencies": [
    {
      "consumer": {
        "environment": "",
        "appld": "default",
        "serviceName": "test1",
        "version": "1.0.0"
      },
      "providers": [
        {
          "environment": "",
          "appld": "default",
          "serviceName": "test",
          "version": "1.0.0"
        }
      ]
    }
  ]
}
```

Example Response

None

Status Code

See [Status Codes](#).

Error Code

See [CSE Error Codes](#).

12.5 Reporting Service Metrics

Function

This API is used to report service metrics to the dashboard.

Call Method

See [Calling APIs](#).

URI

POST /v2/{project_id}/csemonitor/metric

Table 12-224 Path parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which must be unique. The value contains 1 to 64 characters.

Request

Table 12-225 Request header parameters

Parameter	Mandatory	Type	Description
x-domain-name	Yes	String	Tenant account name.
X-Auth-Token	Yes	String	User token.

Table 12-226 Request body parameters

Parameter	Mandatory	Type	Description
name	No	String	Name.
appId	No	String	Application ID.
serviceId	No	String	Service ID.
version	No	String	Version.
instanceId	No	String	Instance ID.
instance	No	String	Instance information.
thread	No	Integer	Thread name.
cpu	No	Double	CPU information.
memory	No	Map<String, Long>	Memory information.
interfaces	No	Array of FunctionMetricInfo objects	Monitoring item list.

Table 12-227 FunctionMetricInfo

Parameter	Mandatory	Type	Description
functionName	No	String	Name of the monitoring item.

Parameter	Mandatory	Type	Description
functionType	No	String	Monitoring item type.
desc	No	String	Monitoring item description.
time	No	Long	Time.
appld	No	String	Application ID.
version	No	String	Version.
qps	No	Double	Throughput.
latency	No	Double	Latency.
rate	No	Double	Percentage.
failureRate	No	Double	Failure percentage.
shortCircuited	No	Integer	Number of circuits.
semaphoreRejected	No	Integer	Number of rejected semaphores.
threadPoolRejected	No	Integer	Number of rejected threads.
countTimeout	No	Integer	Timeout duration.
l995	No	Double	99.5% requests have a value smaller than the value of this parameter.
l99	No	Double	99% requests have a value smaller than the value of this parameter.
l90	No	Double	90% requests have a value smaller than the value of this parameter.
l75	No	Double	75% requests have a value smaller than the value of this parameter.
l50	No	Double	50% requests have a value smaller than the value of this parameter.
l25	No	Double	25% requests have a value smaller than the value of this parameter.
l5	No	Double	5% requests have a value smaller than the value of this parameter.

Response

Status code: 400

Table 12-228 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Status code: 500

Table 12-229 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Error message.
detail	String	Location details.

Example Request

POST https://{endpoint}/v2/{project_id}/csemonitor/metric

```
{
  "name" : "test",
  "appId" : "default",
  "version" : "1.0.0",
  "instance" : "hostNameTest",
  "thread" : 0,
  "cpu" : 0,
  "memory" : {
    "1": "1"
  },
  "interfaces" : [
    {
      "time" : 0,
      "name" : "test",
      "qps" : 0,
      "latency" : 0,
      "rate" : 0,
      "total" : 0,
      "isCircuitBreakerOpen" : true,
      "failure" : 0,
      "shortCircuited" : 0,
      "semaphoreRejected" : 0,
      "threadPoolRejected" : 0,
      "countTimeout" : 0,
      "l995" : 0,
      "l99" : 0,
      "l90" : 0,
      "l75" : 0,
      "l50" : 0,
    }
  ]
}
```

```
"l25" : 0,  
"l5" : 0  
}  
]
```

Example Response

Status code: 400

```
{  
  "errorCode" : "string",  
  "errorMessage" : "string",  
  "detail" : "string"  
}
```

Status code: 500

```
{  
  "errorCode" : "string",  
  "errorMessage" : "string",  
  "detail" : "string"  
}
```

Status Code

Status Code	Description
200	The data is reported successfully.
400	Bad Request
500	Internal Server Error

Error Code

See [CSE Error Codes](#).