

应用管理与运维平台

# API 参考

文档版本 01  
发布日期 2025-02-18



版权所有 © 华为云计算技术有限公司 2025。保留一切权利。

非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

## 商标声明



HUAWEI和其他华为商标均为华为技术有限公司的商标。

本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

## 注意

您购买的产品、服务或特性等应受华为云计算技术有限公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为云计算技术有限公司对本文档内容不做任何明示或暗示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

# 华为云计算技术有限公司

地址：贵州省贵安新区黔中大道交兴功路华为云数据中心 邮编：550029

网址：<https://www.huaweicloud.com/>

# 目录

<b>1 使用前必读</b>	<b>1</b>
<b>2 API 概览</b>	<b>3</b>
<b>3 如何调用 API</b>	<b>14</b>
3.1 构造请求	14
3.2 认证鉴权	16
3.3 返回结果	18
<b>4 应用管理 API (v3 版本)</b>	<b>20</b>
4.1 环境	20
4.1.1 创建环境	20
4.1.2 获取所有环境	24
4.1.3 根据环境 ID 删除环境	27
4.1.4 根据环境 ID 修改环境	28
4.1.5 根据环境 ID 获取环境详细信息	32
4.1.6 根据环境 ID 修改环境资源	35
4.1.7 根据环境 ID 查询环境纳管的资源	39
4.2 应用	41
4.2.1 创建应用	41
4.2.2 获取所有应用	44
4.2.3 根据应用 ID 修改应用信息	47
4.2.4 根据应用 ID 删除应用	50
4.2.5 根据应用 ID 获取应用详细信息	52
4.2.6 根据应用 ID 获取应用配置	54
4.2.7 根据应用 ID 修改应用配置	57
4.2.8 根据应用 ID 删除应用配置	60
4.3 组件	61
4.3.1 应用中创建组件	61
4.3.2 应用中导入 CCE 工作负载创建组件	78
4.3.3 获取应用所有组件	82
4.3.4 获取所有组件	88
4.3.5 根据组件 ID 修改组件信息	94
4.3.6 根据组件 ID 删除组件	115
4.3.7 根据组件 ID 获取组件信息	116

4.3.8 根据组件 ID 下发组件任务.....	132
4.3.9 根据组件 ID 获取记录.....	135
4.3.10 根据组件 ID 刷新组件信息.....	138
4.3.11 获取组件相关配置信息.....	140
4.4 技术栈.....	142
4.4.1 查询运行时栈.....	142
4.4.2 查询内置镜像.....	145
4.4.3 创建技术栈.....	147
4.4.4 根据技术栈 ID 查询技术栈.....	151
4.4.5 根据技术栈 ID 修改技术栈.....	153
4.4.6 根据技术栈 ID 删除技术栈.....	157
4.4.7 发布和取消发布技术栈.....	158
4.5 配置管理.....	160
4.5.1 创建配置分组.....	160
4.5.2 获取配置分组.....	162
4.5.3 创建配置文件.....	165
4.5.4 导入配置文件.....	168
4.5.5 获取配置文件信息.....	170
4.5.6 根据配置文件 ID 获取配置文件信息.....	173
4.5.7 根据配置文件 ID 删除配置文件.....	176
4.5.8 根据配置文件 ID 修改配置文件.....	177
4.5.9 根据配置文件 ID 获取配置文件历史.....	180
4.5.10 根据配置文件历史 ID 获取配置文件历史信息.....	184
4.5.11 根据配置文件历史 ID 删除配置文件历史.....	186
4.5.12 根据配置分组 ID 获取分组详情.....	187
4.5.13 根据配置分组 ID 删除分组.....	189
4.6 泳道组.....	190
4.6.1 创建泳道组.....	191
4.6.2 获取所有泳道组.....	195
4.6.3 根据泳道组 ID 获取泳道组信息.....	199
4.6.4 根据泳道组 ID 修改泳道组.....	205
4.6.5 根据泳道组 ID 删除泳道组.....	213
4.6.6 根据网关 ID 获取目标服务列表.....	215
4.6.7 根据泳道组 ID 修改泳道组网关路由配置.....	217
4.7 泳道.....	225
4.7.1 泳道组下创建泳道.....	225
4.7.2 根据微服务引擎 ID 获取所有泳道.....	228
4.7.3 获取泳道组下所有泳道.....	232
4.7.4 根据泳道 ID 获取泳道信息.....	235
4.7.5 根据泳道 ID 修改泳道信息.....	239
4.7.6 根据泳道 ID 删除泳道.....	242
4.7.7 根据泳道 ID 控制泳道路由开关.....	244

4.7.8 根据泳道 ID 管理泳道纳管组件.....	245
4.7.9 根据泳道 ID 批量操作泳道组件.....	248
4.7.10 根据泳道 ID 关联组件.....	250
4.8 部署任务.....	254
4.8.1 获取部署任务详细信息.....	254
4.9 插件管理.....	258
4.9.1 创建插件.....	258
4.9.2 查询插件列表.....	260
4.9.3 查询插件元数据信息.....	263
4.9.4 根据插件 ID 查询插件详情.....	265
4.9.5 根据插件 ID 下发插件任务.....	267
4.9.6 根据插件 ID 删除插件.....	269
<b>5 访问 Git 仓库 API.....</b>	<b>271</b>
5.1 获取仓库授权列表.....	271
5.2 获取授权重定向 URL.....	273
5.3 创建 OAuth 授权.....	275
5.4 创建私人令牌授权.....	279
5.5 创建口令授权.....	281
5.6 删除仓库授权.....	284
5.7 获取仓库 namespaces.....	285
5.8 通过 clone url 获取仓库信息.....	287
5.9 获取命名空间下所有项目.....	289
5.10 创建软件仓库项目.....	291
5.11 获取项目分支.....	293
5.12 获取项目 tags.....	295
5.13 创建项目 tags.....	296
5.14 删除项目 tags.....	298
5.15 获取项目 commits.....	300
5.16 获取项目 hooks.....	302
5.17 创建项目 hook.....	304
5.18 删除项目 hook.....	306
5.19 获取仓库文件目录.....	308
5.20 获取仓库文件内容.....	310
5.21 创建仓库文件.....	312
5.22 修改仓库文件内容.....	314
5.23 删除仓库文件.....	316
<b>6 CSE API.....</b>	<b>319</b>
6.1 调用说明.....	319
6.2 动态配置.....	319
6.2.1 导入配置.....	319
6.2.2 导出配置.....	324
6.3 引擎管理.....	327

6.3.1 查询微服务引擎专享版支持规格.....	328
6.3.2 查询微服务引擎专享版列表.....	330
6.3.3 创建微服务引擎专享版.....	336
6.3.4 查询微服务引擎专享版详情.....	340
6.3.5 删除微服务引擎专享版.....	347
6.3.6 查询微服务引擎专享版任务详情.....	349
<b>7 ServiceComb API.....</b>	<b>355</b>
7.1 调用说明.....	355
7.2 认证.....	355
7.2.1 获取微服务引擎专享版用户 Token.....	355
7.3 微服务.....	357
7.3.1 查询指定微服务信息.....	358
7.3.2 删除微服务定义信息.....	361
7.3.3 查询所有微服务信息.....	364
7.3.4 创建微服务静态信息.....	367
7.3.5 批量删除微服务静态信息.....	374
7.3.6 修改微服扩展属性信息.....	377
7.3.7 查询微服务唯一标识信息.....	380
7.4 契约.....	383
7.4.1 查询微服务契约.....	383
7.4.2 修改微服务契约.....	386
7.4.3 查询微服务的所有契约.....	388
7.5 微服务实例.....	391
7.5.1 注册微服务实例.....	391
7.5.2 根据 service_id 查询微服务实例.....	395
7.5.3 注销微服务实例.....	399
7.5.4 查询微服务实例详细信息.....	401
7.5.5 修改微服务实例扩展信息.....	405
7.5.6 修改微服务实例状态.....	408
7.5.7 发送心跳信息.....	410
7.5.8 按条件查询微服务实例.....	412
7.5.9 批量查询微服务实例.....	416
7.6 依赖关系.....	422
7.6.1 查询服务 providers.....	423
7.7 配置管理.....	427
7.7.1 创建配置.....	427
7.7.2 修改配置.....	430
7.7.3 查询指定配置.....	433
7.7.4 查询配置列表.....	435
7.7.5 删除指定配置.....	439
7.7.6 删除所有配置.....	441
<b>8 应用示例.....</b>	<b>444</b>

8.1 创建并管理环境.....	444
8.2 创建并管理应用.....	446
8.3 创建并管理组件.....	448
8.4 微服务注册发现.....	453
<b>9 数据结构.....</b>	<b>456</b>
9.1 参数说明.....	456
9.1.1 HealthCheck.....	456
9.1.2 MicroServiceInstance.....	456
9.1.3 HeartbeatSetElement.....	457
9.1.4 InstanceHbRst.....	458
9.1.5 DelServicesResponse.....	458
9.1.6 MetricData.....	458
9.1.7 FunctionMetricInfo.....	459
9.1.8 InstanceMetricInfo.....	459
9.2 公共请求参数.....	459
9.2.1 MicroService.....	460
9.2.2 Properties.....	461
9.2.3 DependencyMicroService.....	462
9.2.4 Rule.....	462
9.2.5 AddOrUpdateRule.....	463
9.2.6 DataCenterInfo.....	464
9.2.7 Schema.....	464
9.3 公共响应参数.....	464
9.3.1 WatchMicroServiceKey.....	464
9.3.2 SuccessdResponse.....	465
9.3.3 ServiceInfo.....	465
9.3.4 AggregateMetricInfo.....	466
9.3.5 ServiceDetail.....	467
9.3.6 TenantProject.....	467
9.3.7 Error.....	468
<b>10 权限和授权项.....</b>	<b>469</b>
10.1 权限及授权项说明.....	469
<b>11 附录.....</b>	<b>472</b>
11.1 状态码.....	472
11.2 ServiceStage 错误码.....	472
11.3 CSE 错误码.....	476
11.4 ServiceComb 错误码.....	480
11.5 获取项目 ID.....	483
11.6 获取账号 ID.....	484
<b>12 历史 API.....</b>	<b>485</b>
12.1 应用管理 (v2 版本) .....	485

12.1.1 Meta.....	485
12.1.1.1 获取所有支持的应用组件运行时类型.....	485
12.1.1.2 获取所有支持的应用资源规格.....	487
12.1.2 环境.....	489
12.1.2.1 创建环境.....	489
12.1.2.2 获取所有环境.....	493
12.1.2.3 修改环境信息.....	497
12.1.2.4 根据环境 ID 删除环境.....	500
12.1.2.5 根据环境 ID 获取环境详细信息.....	501
12.1.2.6 修改环境资源.....	504
12.1.3 应用.....	508
12.1.3.1 创建应用.....	508
12.1.3.2 获取所有应用.....	511
12.1.3.3 修改应用信息.....	514
12.1.3.4 根据应用 ID 删除应用.....	516
12.1.3.5 根据应用 ID 获取应用详细信息.....	517
12.1.3.6 修改应用配置信息.....	519
12.1.3.7 删除应用配置.....	523
12.1.3.8 获取应用配置.....	524
12.1.4 组件.....	527
12.1.4.1 应用中创建组件.....	527
12.1.4.2 获取应用所有组件.....	534
12.1.4.3 根据组件 ID 修改组件信息.....	539
12.1.4.4 根据组件 ID 删除组件.....	546
12.1.4.5 根据组件 ID 获取组件信息.....	548
12.1.5 实例.....	552
12.1.5.1 创建组件实例.....	552
12.1.5.2 获取组件实例.....	563
12.1.5.3 对组件实例的操作.....	568
12.1.5.4 修改组件实例.....	570
12.1.5.5 删除组件实例.....	574
12.1.5.6 根据实例 ID 获取实例详细信息.....	576
12.1.5.7 获取组件实例快照.....	581
12.1.6 部署任务.....	584
12.1.6.1 获取部署任务详细信息.....	584
12.2 构建 API.....	589
12.2.1 根据组件名称获取构建任务列表.....	589
12.2.2 根据构建任务 ID 获取构建任务实例列表.....	591
12.2.3 根据构建任务实例 ID 获取构建阶段列表.....	593
12.2.4 根据构建任务实例 ID 获取构建日志.....	596
12.3 查询配置.....	598
12.4 创建服务间依赖关系.....	600



---

12.5 创建上报服务指标.....	602
--------------------	-----

# 1 使用前必读

欢迎使用应用管理与运维平台（ServiceStage）。ServiceStage是面向企业的应用管理与运维平台，提供应用发布、部署、监控与运维等一站式解决方案。支持Java、Php、Python、Node.js、Docker、Tomcat技术栈。支持Apache ServiceComb Java Chassis（Java Chassis）、Spring Cloud等微服务应用，让企业应用上云更简单。

您可以使用本文档提供的API对ServiceStage进行相关操作，如管理环境、应用和组件等。支持的全部操作请参见[API概览](#)。

在您调用ServiceStage API之前，请确保已经充分了解ServiceStage的相关概念，详细信息请参见[产品介绍](#)。

另外，ServiceStage还提供了微服务引擎功能增强接口，请参见[CSE API](#)。如果您要使用微服务引擎原生接口，请参见[ServiceComb API](#)。

## 终端节点

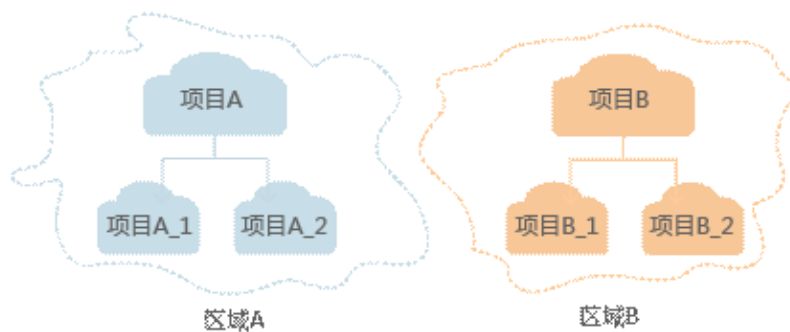
终端节点即调用API的[请求地址](#)，不同服务不同区域的终端节点不同，您可以从[地区和终端节点](#)中查询ServiceStage服务的终端节点。

## 基本概念

- 账号  
用户注册华为云时的账号，账号对其所拥有的资源及云服务具有完全的访问权限，可以重置用户密码、分配用户权限等。由于账号是付费主体，为了确保账号安全，建议您不要直接使用账号进行日常管理工作，而是创建并使用用户进行日常管理工作。
- 用户  
由账号在IAM中创建的用户，是云服务的使用人员，具有身份凭证（密码和访问密钥）。  
在[我的凭证](#)下，您可以查看账号ID和用户ID。通常在调用API的鉴权过程中，您需要用到账号、用户和密码等信息。
- 区域（Region）  
从地理位置和网络时延维度划分，同一个Region内共享弹性计算、块存储、对象存储、VPC网络、弹性公网IP、镜像等公共服务。Region分为通用Region和专属Region，通用Region指面向公共租户提供通用云服务的Region；专属Region指只承载同一类业务或只面向特定租户提供业务服务的专用Region。  
详情请参见[区域和可用区](#)。

- 可用区（AZ，Availability Zone）  
一个AZ是一个或多个物理数据中心的集合，有独立的风火水电，AZ内逻辑上再将计算、网络、存储等资源划分成多个集群。一个Region中的多个AZ间通过高速光纤相连，以满足用户跨AZ构建高可用性系统的需求。
- 项目  
华为云的区域默认对应一个项目，这个项目由系统预置，用来隔离物理区域间的资源（计算资源、存储资源和网络资源），以默认项目为单位进行授权，用户可以访问您账号中该区域的所有资源。如果您希望进行更加精细的权限控制，可以在区域默认的项目中创建子项目，并在子项目中购买资源，然后以子项目为单位进行授权，使得用户仅能访问特定子项目中资源，使得资源的权限控制更加精确。

图 1-1 项目隔离模型



同样在[我的凭证](#)下，您可以查看项目ID。

- 企业项目  
企业项目是项目的升级版，针对企业不同项目间资源的分组和管理，是逻辑隔离。企业项目中可以包含多个区域的资源，且项目中的资源可以迁入迁出。  
关于企业项目ID的获取及企业项目特性的详细信息，请参见《[企业管理服务用户指南](#)》。

# 2 API 概览

表 2-1 接口简介

类型	子类型	说明
应用管理API (v3版本)	环境接口	环境的相关接口。 环境是用于组件部署和运行的计算（如云容器引擎 CCE、弹性云服务器 ECS等）、网络（如弹性负载均衡 ELB、弹性IP EIP等）和中间件（如分布式缓存 DCS、云数据库 RDS等）等基础资源的集合。 ServiceStage把多种基础资源组合为一个环境，如：开发环境、测试环境、预生产环境和生产环境。按环境维度来管理资源、部署组件，可以减少具体基础资源运维管理的复杂性。
	应用接口	应用的相关接口。 应用是一个功能相对完备的业务系统，由一个或多个特性相关的组件组成。
	组件接口	组件的相关接口。 组件是组成应用的某个业务特性实现，以代码或者软件包为载体，可独立部署在环境下运行。

类型	子类型	说明
	<a href="#">配置管理接口</a>	配置的相关接口。 在ServiceStage中，配置就是一个文件。通过创建统一的配置文件，您可以将环境和应用的系统变量（例如环境关联的IP、端口、数据库地址，应用关联的应用名称等）填充到配置中，生成一个配置文件。组件关联配置文件进行部署时，系统变量会自动替换为实际值，通过文件挂载实现一次配置、多环境使用。
	<a href="#">技术栈接口</a>	技术栈相关接口。 技术栈包含组件运行所依赖的操作系统、框架、运行时，由技术栈名称、类型、状态、版本等属性构成。其中，版本号遵从 <a href="#">语义化版本控制规范</a> 。ServiceStage为您提供并管理技术栈生命周期，您只需聚焦具体业务开发，提升您的应用托管体验。
	<a href="#">部署任务接口</a>	部署任务的相关接口。
	<a href="#">插件管理接口</a>	管理插件的相关接口。
访问Git仓库API	<a href="#">访问Git仓库接口</a>	Git仓库授权、仓库命名空间、项目、分支、tags、commits、hooks、仓库文件目录、内容的相关接口。
CSE API	<a href="#">动态配置</a>	提供从配置管理中心导入、导出配置信息的接口。
	<a href="#">引擎管理</a>	提供创建、查询、删除微服务引擎专享版的引擎管理接口。
ServiceComb API	<a href="#">认证</a>	微服务身份认证功能的相关操作接口。
	<a href="#">微服务</a>	微服务管理的相关操作接口。
	<a href="#">契约</a>	微服务契约查询、上传等相关操作接口。
	<a href="#">微服务实例</a>	微服务实例注册、上报心跳等相关操作接口。
	<a href="#">依赖关系</a>	微服务依赖关系查询接口。
	<a href="#">配置管理</a>	提供配置中心配置管理功能的相关操作接口。

## 环境接口

表 2-2 环境接口

API	说明
<a href="#">创建环境</a>	此API用来创建环境。
<a href="#">获取所有环境</a>	此API用来获取所有环境。
<a href="#">根据环境ID删除环境</a>	此API用来通过环境ID删除环境。
<a href="#">根据环境ID修改环境</a>	此API用来通过环境ID修改环境。
<a href="#">根据环境ID获取环境详细信息</a>	此API用来通过环境ID获取环境详细信息。
<a href="#">根据环境ID修改环境资源</a>	此API用来通过环境ID修改环境资源。
<a href="#">根据环境ID查询环境纳管的资源</a>	此API用来根据环境ID查询环境纳管的资源。

## 应用接口

表 2-3 应用接口

API	说明
<a href="#">创建应用</a>	此API用来创建应用。
<a href="#">获取所有应用</a>	此API用来获取所有应用。
<a href="#">根据应用ID修改应用信息</a>	此API用来通过应用ID修改应用信息。
<a href="#">根据应用ID删除应用</a>	此API用来通过应用ID删除应用。
<a href="#">根据应用ID获取应用详细信息</a>	此API用来通过应用ID获取应用详细信息。
<a href="#">根据应用ID获取应用配置</a>	此API用来通过应用ID获取应用配置信息。
<a href="#">根据应用ID修改应用配置</a>	此API用来通过应用ID修改应用配置。
<a href="#">根据应用ID删除应用配置</a>	此API用来通过应用ID删除应用配置。

## 组件接口

表 2-4 组件接口

API	说明
<a href="#">应用中创建组件</a>	此API用来在应用中创建组件。
<a href="#">获取应用所有组件</a>	此API用来获取应用下所有组件。
<a href="#">获取所有组件</a>	此API用来获取所有应用组件。
<a href="#">根据组件ID修改组件信息</a>	此API用来通过组件ID修改组件信息。
<a href="#">根据组件ID删除组件</a>	此API用来通过组件ID删除组件。
<a href="#">根据组件ID获取组件信息</a>	此API用来通过组件ID获取组件信息。
<a href="#">根据组件ID下发组件任务</a>	此API用来通过组件ID下发组件任务。
<a href="#">根据组件ID获取记录</a>	此API用来通过组件ID获取记录。
<a href="#">根据组件ID刷新组件信息</a>	此API用来通过组件ID刷新组件状态和访问方式信息。
<a href="#">获取组件相关配置信息</a>	此API用来获取所有所有组件所属的应用信息、环境信息、企业项目信息。

## 技术栈接口

表 2-5 技术栈接口介绍

API	说明
<a href="#">查询运行时栈</a>	此API用来查询技术栈信息。
<a href="#">查询内置镜像</a>	此API用来获取ServiceStage各个技术栈构建时使用的内置基础镜像。
<a href="#">创建技术栈</a>	此API用来创建自定义技术栈。
<a href="#">根据技术栈ID查询技术栈</a>	此API用来根据技术栈ID查询技术栈。
<a href="#">根据技术栈ID修改技术栈</a>	此API用来根据技术栈ID修改技术栈。
<a href="#">根据技术栈ID删除技术栈</a>	此API用来根据技术栈ID删除技术栈。
<a href="#">发布和取消发布技术栈</a>	此API用来发布和取消发布技术栈。

## 配置管理接口

表 2-6 配置管理接口介绍

API	说明
<a href="#">创建配置分组</a>	此API用来创建配置分组，用于对配置文件分类管理。
<a href="#">获取配置分组</a>	此API用来获取所有已经创建的配置分组。
<a href="#">创建配置文件</a>	此API用来创建配置文件。
<a href="#">导入配置文件</a>	此API通过导入zip压缩包创建配置文件。
<a href="#">获取配置文件信息</a>	此API用来获取配置文件信息。
<a href="#">根据配置文件ID获取配置文件信息</a>	此API通过配置文件ID获取配置文件的详情。
<a href="#">根据配置文件ID删除配置文件</a>	此API通过配置文件ID删除配置文件。
<a href="#">根据配置文件ID修改配置文件</a>	此API通过配置文件ID修改配置文件。
<a href="#">根据配置文件ID获取配置文件历史</a>	此API通过配置文件ID获取配置文件历史。
<a href="#">根据配置文件历史ID获取配置文件历史信息</a>	此API通过配置文件历史ID获取配置文件历史信息。
<a href="#">根据配置文件历史ID删除配置文件历史</a>	此API通过配置文件历史ID删除配置文件历史。
<a href="#">根据配置分组ID获取分组详情</a>	此API通过配置分组ID获取配置分组详情。
<a href="#">根据配置分组ID删除分组</a>	此API根据配置分组ID删除分组。

## 泳道组接口

表 2-7 泳道组接口

API	说明
<a href="#">创建泳道组</a>	此API用来在创建泳道组。 泳道组是一组泳道的集合，用于区分不同的组织或场景。同一个项目下，最多支持创建300个泳道组。
<a href="#">获取所有泳道组</a>	此API用来获取所有已经创建的泳道组。
<a href="#">根据泳道组ID获取泳道组信息</a>	此API用来根据泳道组ID获取泳道组信息。



API	说明
<a href="#">根据泳道组ID修改泳道组</a>	此API用来根据泳道组ID修改泳道组。
<a href="#">根据泳道组ID删除泳道组</a>	此API用来根据泳道组ID删除泳道组。
<a href="#">根据网关ID获取目标服务列表</a>	此API用来根据网关ID获取目标服务列表。
<a href="#">根据泳道组ID修改泳道组网关路由配置</a>	此API用来根据泳道组ID修改泳道组网关路由配置。

## 泳道接口

表 2-8 泳道接口

API	说明
<a href="#">泳道组下创建泳道</a>	此API用来在泳道组下创建泳道。
<a href="#">根据微服务引擎ID获取所有泳道</a>	此API用来根据微服务引擎ID获取所有泳道。
<a href="#">获取泳道组下所有泳道</a>	此API用来获取泳道组下所有泳道。
<a href="#">根据泳道ID获取泳道信息</a>	此API用来根据泳道ID获取泳道信息。
<a href="#">根据泳道ID修改泳道信息</a>	此API用来根据泳道ID修改泳道信息。
<a href="#">根据泳道ID删除泳道</a>	此API用来根据泳道ID删除泳道。
<a href="#">根据泳道ID控制泳道路由开关</a>	此API用来根据泳道ID控制泳道路由开关。
<a href="#">根据泳道ID关联组件</a>	此API用来根据泳道ID为泳道关联组件。

## 部署任务接口

表 2-9 部署任务接口

API	说明
<a href="#">获取部署任务详细信息</a>	此API用来获取部署任务详细信息。

## 插件管理接口

表 2-10 插件管理接口

API	说明
<a href="#">创建插件</a>	此API用来创建插件。
<a href="#">查询插件列表</a>	此API用来查询环境下的插件列表。
<a href="#">查询插件元数据信息</a>	此API用来查询环境下插件的元数据信息。
<a href="#">根据插件ID查询插件详情</a>	此API用来查询环境下指定插件详情。
<a href="#">根据插件ID下发插件任务</a>	此API用来根据插件ID下发插件任务。
<a href="#">根据插件ID删除插件</a>	此API用于根据插件ID删除插件。

## 访问 Git 仓库接口

表 2-11 访问 Git 仓库接口

API	说明
<a href="#">获取仓库授权列表</a>	此API用于获取Git仓库所有授权信息。
<a href="#">获取授权重定向URL</a>	此API用于获取授权重定向URL。
<a href="#">创建OAuth授权</a>	此API用于创建OAuth授权。
<a href="#">创建私人令牌授权</a>	此API用于创建私有令牌授权。
<a href="#">创建口令授权</a>	此API用于创建Git仓库口令授权。
<a href="#">删除仓库授权</a>	此API用于通过名称删除仓库授权。
<a href="#">获取仓库namespaces</a>	此API用于获取仓库namespaces。
<a href="#">通过clone url获取仓库信息</a>	此API用于通过clone url获取仓库信息。
<a href="#">获取命名空间下所有项目</a>	此API用于获取命名空间下所有项目。
<a href="#">创建软件仓库项目</a>	此API用于创建软件仓库项目。
<a href="#">获取项目分支</a>	此API用于获取项目分支。
<a href="#">获取项目tags</a>	此API用于获取项目tags。
<a href="#">创建项目tags</a>	此API用于创建项目tags。
<a href="#">删除项目tags</a>	此API用于删除项目tags。
<a href="#">获取项目commits</a>	此API用于获取项目最近10次commits。
<a href="#">获取项目hooks</a>	此API用于获取项目hooks。

API	说明
<a href="#">创建项目hook</a>	此API用于创建项目hook。
<a href="#">删除项目hook</a>	此API用于删除项目hook。
<a href="#">获取仓库文件目录</a>	此API用于获取仓库文件目录。
<a href="#">获取仓库文件内容</a>	此API用于获取仓库文件内容。
<a href="#">创建仓库文件</a>	此API用于创建仓库文件。
<a href="#">修改仓库文件内容</a>	此API用于修改仓库文件内容。
<a href="#">删除仓库文件</a>	此API用于删除仓库文件内容。

## 动态配置

表 2-12 动态配置接口

API	说明
<a href="#">导入配置</a>	此API用于从配置管理中心导入配置信息。
<a href="#">导出配置</a>	此API用于从配置管理中心导出配置信息。

## 引擎管理

表 2-13 引擎管理接口

API	说明
<a href="#">查询微服务引擎专享版支持规格</a>	此API用于查询微服务引擎专享版支持的规格。
<a href="#">查询微服务引擎专享版列表</a>	此API用于查询微服务引擎专享版列表。
<a href="#">创建微服务引擎专享版</a>	此API用于创建微服务引擎专享版。
<a href="#">查询微服务引擎专享版详情</a>	此API用于查询微服务引擎专享版详情。
<a href="#">删除微服务引擎专享版</a>	此API用于删除微服务引擎专享版。
<a href="#">查询微服务引擎专享版任务详情</a>	此API用于查询微服务引擎专享版任务详情。

## 认证

表 2-14 认证接口

API	说明
<a href="#">获取微服务引擎专享版用户Token</a>	此API仅用于开启了安全认证的微服务引擎专享版。 在访问开启了安全认证的微服务引擎专享版的接口前，需要先调用该接口获取用户Token，该Token是访问接口的唯一凭据。

## 微服务

表 2-15 微服务接口

API	说明
<a href="#">查询指定微服务信息</a>	此API用于根据serviceld查询微服务定义信息。
<a href="#">删除微服务定义信息</a>	此API用于删除一个微服务定义及其相关信息，同时注销其所有实例信息。
<a href="#">查询所有微服务信息</a>	此API用于根据条件组合，查询满足所有条件的微服务定义信息。
<a href="#">创建微服务静态信息</a>	此API用于在注册微服务实例前创建服务静态信息，之后注册的微服务实例根据serviceld这个字段与静态信息关联，一个服务对应对多个实例。serviceld可以定制。如果定制了，系统将使用定制serviceld；如果没定制，系统会随机生成serviceld。
<a href="#">批量删除微服务静态信息</a>	此API用于批量删除指定的微服务定义及其相关信息，同时注销其所有实例信息。
<a href="#">修改微服扩展属性信息</a>	此API用于创建微服务静态信息后对服务部分字段进行更新。每次更新都需要传入完整的服务静态信息json，也就是说，即便不更新部分的字段也要作为json的属性传过去。
<a href="#">查询微服务唯一标识信息</a>	此API用于通过指定条件，查询微服务serviceld或schema的唯一标识信息。

## 契约

表 2-16 契约接口

API	说明
<a href="#">查询微服务契约</a>	此API用于根据service_id和schema_id查询微服务的契约信息。
<a href="#">修改微服务契约</a>	此API用于根据schema_id更新微服务的契约内容。
<a href="#">查询微服务的所有契约</a>	此API用于查询一个微服务的所有契约信息，包括schemaid和summary信息。

## 微服务实例

表 2-17 微服务实例接口

API	说明
<a href="#">注册微服务实例</a>	此API用于在创建微服务后注册该微服务的实例。注册微服务实例时，需提供该微服务实例相关的信息。instanceId可定制，如果定制了，再次注册就直接全内容覆盖。如果没定制，逻辑如下：系统自动生成id，如果endpoints内容重复，则使用原来的id。
<a href="#">根据service_id查询微服务实例</a>	此API用于在实例注册后根据service_id发现该微服务的所有实例。
<a href="#">注销微服务实例</a>	此API用于在实例注册后根据instance_id进行实例注销。
<a href="#">查询微服务实例详细信息</a>	此API用于在实例注册后根据service_id和instance_id获取该实例的详细信息。
<a href="#">修改微服务实例扩展信息</a>	此API用于在实例注册后根据instance_id添加/更新一个微服务实例的扩展信息。
<a href="#">修改微服务实例状态</a>	此API用于在实例注册后根据instance_id更新一个微服务实例的状态。
<a href="#">发送心跳信息</a>	此API用于在服务提供端向服务中心发送心跳信息，以保证服务中心知道服务实例是否健康。

API	说明
<a href="#">按条件查询微服务实例</a>	此API用于在实例注册后根据微服务版本规则或字段条件发现该微服务的实例。
<a href="#">批量查询微服务实例</a>	此API用于在实例注册后根据微服务版本规则或字段条件批量发现多个微服务的实例。

## 依赖关系

表 2-18 依赖关系接口

API	说明
<a href="#">查询服务providers</a>	此API用于根据consumer_id获取该服务的所有providers。比如A->B A->C, 那么根据A的服务Id, 可以查询到B、C的微服务信息。

## 配置管理

表 2-19 配置管理接口

API	说明
<a href="#">创建配置</a>	此API用于创建一条配置。
<a href="#">修改配置</a>	此API用于修改指定一条配置。
<a href="#">查询指定配置</a>	此API用于查询指定一条配置。
<a href="#">查询配置列表</a>	此API用于查询配置列表。
<a href="#">删除指定配置</a>	此API用于删除指定一条配置。
<a href="#">删除所有配置</a>	此API用于删除所有配置。

# 3 如何调用 API

## 3.1 构造请求

本节介绍REST API请求的组成，并以调用IAM的[管理员创建IAM用户](#)接口说明如何调用API。

您还可以通过这个视频教程了解如何构造请求调用API：<https://bbs.huaweicloud.com/videos/102987>。

### 请求 URI

请求URI由如下部分组成。

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

尽管请求URI包含在请求消息头中，但大多数语言或框架都要求您从请求消息中单独传递它，所以在此单独强调。

- **URI-scheme**：表示用于传输请求的协议，当前所有API均采用HTTPS协议。
- **Endpoint**：指定承载REST服务端点的服务器域名或IP，不同服务不同区域的Endpoint不同，您可以从[地区和终端节点](#)处获取。例如IAM服务在“华北-北京四”区域的Endpoint为“iam.cn-north-4.myhuaweicloud.com”。
- **resource-path**：资源路径，也即API访问路径。从具体API的URI模块获取，例如“管理员创建IAM用户”API的resource-path为“/v3.0/OS-USER/users”。
- **query-string**：查询参数，是可选部分，并不是每个API都有查询参数。查询参数前面需要带一个“？”，形式为“参数名=参数取值”。例如“limit=10”，表示查询不超过10条数据。

例如您需要获取创建IAM用户，由于IAM为全局服务。则使用任一区域的Endpoint（比如“华北-北京四”区域的Endpoint：iam.cn-north-4.myhuaweicloud.com），并在[管理员创建IAM用户](#)的URI部分找到resource-path（/v3.0/OS-USER/users），拼接起来如下所示。

```
https://iam.cn-north-4.myhuaweicloud.com/v3.0/OS-USER/users
```

图 3-1 URI 示意图



### 说明

为查看方便，在每个具体API的URI部分，只给出resource-path部分，并将请求方法写在一起。这是因为URI-scheme都是HTTPS，而Endpoint在同一个区域也相同，所以简洁起见将这两部分省略。

## 请求方法

HTTP请求方法（也称为操作或动词），表示服务正在请求什么类型的操作。

- **GET**：请求服务器返回指定资源。
- **PUT**：请求服务器更新指定资源。
- **POST**：请求服务器新增资源或执行特殊操作。
- **DELETE**：请求服务器删除指定资源，如删除对象等。
- **HEAD**：请求服务器资源头部。
- **PATCH**：请求服务器更新资源的部分内容。当资源不存在的时候，PATCH可能会去创建一个新的资源。

在[管理员创建IAM用户](#)的URI部分，您可以看到其请求方法为“POST”，则其请求为：

```
POST https://iam.cn-north-4.myhuaweicloud.com/v3.0/OS-USER/users
```

## 请求消息头

附加请求头字段，如指定的URI和HTTP方法所要求的字段。例如定义消息体类型的请求头“Content-Type”，请求鉴权信息等。

如下公共消息头需要添加到请求中。

- **Content-Type**：消息体的类型（格式），必选，默认取值为“application/json;charset=utf8”，有其他取值时会在具体接口中专门说明。
- **Authorization**：签名认证信息，可选，当使用AK/SK方式认证时，使用SDK对请求进行签名的过程中会自动填充该字段。AK/SK认证的详细说明请参见[AK/SK认证](#)。
- **X-Sdk-Date**：请求发送的时间，可选，当使用AK/SK方式认证时，使用SDK对请求进行签名的过程中会自动填充该字段。AK/SK认证的详细说明请参见[AK/SK认证](#)。
- **X-Auth-Token**：用户Token，可选，当使用Token方式认证时，必须填充该字段。用户Token也就是调用[获取用户Token](#)接口的响应值，该接口是唯一不需要认证的接口。
- **X-Project-ID**：子项目ID，可选，在多项目场景中使用。如果云服务资源创建在子项目中，AK/SK认证方式下，操作该资源的接口调用需要在请求消息头中携带X-Project-ID。



- **X-Domain-ID**: 账号ID, 可选。AK/SK认证方式下, 全局服务的接口调用时, 需在请求消息头中携带X-Domain-ID。

对于**管理员创建IAM用户**接口, 使用AK/SK方式认证时, 添加消息头后的请求如下所示。

```
POST https://iam.cn-north-4.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=*****
```

## 请求消息体

请求消息体通常以结构化格式发出, 与请求消息头中Content-type对应, 传递除请求消息头之外的内容。若请求消息体中参数支持中文, 则Content-type中需声明字符编码方式为UTF-8。例如, Content-Type: application/json; charset=utf-8。

每个接口的请求消息体内容不同, 也并不是每个接口都需要有请求消息体 (或者说消息体为空), GET、DELETE操作类型的接口就不需要消息体, 消息体具体内容需要根据具体接口而定。

对于**管理员创建IAM用户**接口, 您可以从接口的请求部分看到所需的请求参数及参数说明。将消息体加入后的请求如下所示, 加粗的字段需要根据实际值填写。

- **accountid**为IAM用户所属的账号ID。
- **username**为要创建的IAM用户名。
- **email**为IAM用户的邮箱。
- **\*\*\*\*\***为IAM用户的登录密码。

```
POST https://iam.cn-north-4.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"
  }
}
```

到此为止, 一个API请求所需要的内容已经准备完成, 您可以使用curl、Postman或直接编写代码等方式发送请求调用API。

## 3.2 认证鉴权

调用接口有如下两种认证方式, 您可以选择其中一种进行认证鉴权。

- AK/SK认证: 通过AK ( Access Key ID ) /SK ( Secret Access Key ) 加密调用请求。
- Token认证: 通过Token认证调用请求。

## AK/SK 认证

### 说明

- AK/SK签名认证方式仅支持消息体大小12MB以内，12MB以上的请求请使用Token认证。
- AK/SK既可以使用永久访问密钥中的AK/SK，也可以使用临时访问密钥中的AK/SK。但使用临时访问密钥的AK/SK时需要额外携带“X-Security-Token”字段，字段值为临时访问密钥的security\_token。

AK/SK认证就是使用AK/SK对请求进行签名，在请求时将签名信息添加到消息头，从而通过身份认证。

- AK(Access Key ID)：访问密钥ID。与私有访问密钥关联的唯一标识符；访问密钥ID和私有访问密钥一起使用，对请求进行加密签名。
- SK(Secret Access Key)：与访问密钥ID结合使用的密钥，对请求进行加密签名，可标识发送方，并防止请求被修改。

使用AK/SK认证时，您可以基于签名算法使用AK/SK对请求进行签名，也可以使用专门的签名SDK对请求进行签名。详细的签名方法和SDK使用方法请参见[API签名指南](#)。

### 须知

签名SDK只提供签名功能，与服务提供的SDK不同，使用时请注意。

## Token 认证

### 说明

- Token的有效期为24小时，需要使用一个Token鉴权时，可以先缓存起来，避免频繁调用。
- 使用Token前请确保Token离过期有足够的时间，防止调用API的过程中Token过期导致调用API失败。

Token在计算机系统中代表令牌（临时）的意思，拥有Token就代表拥有某种权限。

Token认证就是在调用API的时候将Token加到请求消息头，从而通过身份认证，获得操作API的权限。

Token可通过调用[获取用户Token](#)接口获取，调用本服务API需要project级别的Token，即调用[获取用户Token](#)接口时，请求body中auth.scope的取值需要选择project，如下所示。

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

```
    "project": {  
      "name": "xxxxxxx"  
    }  
  }  
}
```

获取Token后，再调用其他接口时，您需要在请求消息头中添加“X-Auth-Token”，其值即为Token。例如Token值为“ABCDEFGH...”，则调用接口时将“X-Auth-Token: ABCDEFGH...”加到请求消息头即可，如下所示。

```
POST https://iam.cn-north-4.myhuaweicloud.com/v3.0/OS-USER/users  
Content-Type: application/json  
X-Auth-Token: ABCDEFGH...
```

您还可以通过这个视频教程了解如何使用Token认证：<https://bbs.huaweicloud.com/videos/101333>。

## 3.3 返回结果

请求发送以后，您会收到响应，包含状态码、响应消息头和消息体。

### 状态码

状态码是一组从1xx到5xx的数字代码，状态码表示了请求响应的状态，完整的状态码列表请参见[状态码](#)。

对于[管理员创建IAM用户](#)接口，如果调用后返回状态码为“201”，则表示请求成功。

### 响应消息头

对应请求消息头，响应同样也有消息头，如“Content-type”。

对于[管理员创建IAM用户](#)接口，返回如[图3-2](#)所示的消息头。

图 3-2 管理员创建 IAM 用户响应消息头

```
"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"
```

### 响应消息体

响应消息体通常以结构化格式返回，与响应消息头中Content-type对应，传递除响应消息头之外的内容。

对于**管理员创建IAM用户**接口，返回如下消息体。为篇幅起见，这里只展示部分内容。

```
{
  "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": "",
    "default_project_id": null
  }
}
```

当接口调用出错时，会返回错误码及错误信息说明，错误响应的Body体格式如下所示。

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

其中，error\_code表示错误码，error\_msg表示错误描述信息。

# 4 应用管理 API (v3 版本)

## 4.1 环境

### 4.1.1 创建环境

#### 功能介绍

环境是用于组件部署和运行的计算（如云容器引擎 CCE、弹性云服务器 ECS等）、网络（如弹性负载均衡 ELB、弹性IP EIP等）和中间件（如分布式缓存 DCS、云数据库 RDS等）等基础资源的集合。ServiceStage把多种基础资源组合为一个环境，如：开发环境、测试环境、预生产环境和生产环境。按环境维度来管理资源、部署组件，可以减少具体基础资源运维管理的复杂性。

此API用于创建环境。

#### 调试

您可以在[API Explorer](#)中调试该接口。

#### URI

POST /v3/{project\_id}/cas/environments

表 4-1 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。

## 请求消息

表 4-2 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

表 4-3 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	环境名称。 由英文字母、数字、中划线（-）或下划线（_）组成，并以英文字母开头，英文字母或者数字结尾，英文字母区分大小写，长度为2~64个字符。
deploy_mode	否	String	环境类型。 <ul style="list-style-type: none"><li>virtualmachine：虚拟机。</li><li>container：Kubernetes。</li><li>mixed：虚拟机和Kubernetes。</li></ul>
description	否	String	环境描述。 最大长度为128个字符。
enterprise_project_id	否	String	企业项目ID。
vpc_id	是	String	虚拟私有云ID。
labels	否	Array of objects	标签，可使用标签管理服务通过标签过滤资源，请参考 <a href="#">表4-4</a> 。 同一个环境下最多可以添加20个标签。

表 4-4 labels

参数	是否必选	参数类型	描述
key	是	String	标签名称。 标签用于标识资源，当您拥有相同类型的许多资源时，可以使用标签按各种维度（例如用途、所有者或环境）对资源进行分类。 长度为1到36个字符，只能包含数字、英文字母、下划线（_）、中划线（-）或中文。
value	是	String	标签值。 长度为1到43个字符，只能包含数字、英文字母、下划线（_）、点（.）、中划线（-）或中文。

## 响应消息

表 4-5 响应参数

参数	参数类型	描述
id	String	环境ID。
name	String	环境名称。
description	String	环境描述。
project_id	String	项目ID。
enterprise_project_id	String	企业项目ID。
deploy_mode	String	环境类型。
vpc_id	String	虚拟私有云ID。
creator	String	环境创建人。
create_time	Integer	环境创建时间。
update_time	Integer	环境修改时间。
labels	Array of objects	标签，可使用标签管理服务通过标签过滤查询资源信息，请参考 <a href="#">表4-6</a> 。

表 4-6 labels

参数	参数类型	描述
key	String	标签名称。

参数	参数类型	描述
value	String	标签值。

## 请求示例

创建名称为env-test的“虚拟机”类型环境，环境所在VPC的id为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"
}
```

## 响应示例

```
{
  "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": []
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在。
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。



## 4.1.2 获取所有环境

### 功能介绍

此API用于获取所有已经创建的环境。

### 调试

您可以在[API Explorer](#)中调试该接口。

### URI

GET /v3/{project\_id}/cas/environments

表 4-7 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。

表 4-8 查询参数

参数	是否必选	参数类型	描述
limit	否	Integer	查询个数，取值[0, 100]。
offset	否	Integer	查询偏移量，表示查询第n条数据后的结果。
order_by	否	String	排序字段。 排序字段支持如下枚举值： <ul style="list-style-type: none"><li>• create_time，环境创建时间，默认排序字段。</li><li>• name，环境名称。</li><li>• update_time，环境修改时间。</li></ul> 传值不在支持的枚举值范围内按默认排序字段排序。
order	否	String	排序方式。 <ul style="list-style-type: none"><li>• desc，降序排序，默认排序方式。</li><li>• asc，升序排序。</li></ul>
name	否	String	指定环境名称，查询对应环境名称的环境信息。
environment_id	否	String	指定环境ID，查询对应环境ID的环境信息。
enterprise_project_id	否	String	指定企业项目ID，查询对应企业项目ID的环境信息。

## 请求消息

表 4-9 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

## 响应消息

表 4-10 响应参数

参数	参数类型	描述
count	Integer	环境总数。
environments	Array of objects	环境参数，请参考 <a href="#">表4-11</a> 。

表 4-11 environments

参数	参数类型	描述
id	String	环境ID。
name	String	环境名称。
description	String	环境描述。
project_id	String	项目ID。
enterprise_project_id	String	企业项目ID。
deploy_mode	String	环境类型。
vpc_id	String	虚拟私有云ID。
creator	String	环境创建人。
create_time	Integer	环境创建时间。
update_time	Integer	环境修改时间。

参数	参数类型	描述
labels	Array of objects	标签，可使用标签管理服务通过标签过滤资源，请参考 <a href="#">表4-12</a> 。

表 4-12 labels

参数	参数类型	描述
key	String	标签名称。
value	String	标签值。

## 请求示例

无

## 响应示例

```
{
  "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
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在。
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

### 4.1.3 根据环境 ID 删除环境

#### 功能介绍

此API用于通过环境ID删除环境。

#### 调试

您可以在[API Explorer](#)中调试该接口。

#### URI

DELETE /v3/{project\_id}/cas/environments/{environment\_id}

表 4-13 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
environment_id	是	String	环境ID。获取方法，请参考 <a href="#">获取所有环境</a> 。

#### 请求消息

表 4-14 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

#### 响应消息

无

#### 请求示例

无

#### 响应示例

无

## 状态码

状态码	描述
204	操作成功。
400	错误的请求。
404	请求对象不存在。
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

### 4.1.4 根据环境 ID 修改环境

#### 功能介绍

此API用于通过环境ID修改环境。

#### 调试

您可以在[API Explorer](#)中调试该接口。

#### URI

PUT /v3/{project\_id}/cas/environments/{environment\_id}

表 4-15 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
environment_id	是	String	环境ID。获取方法，请参考 <a href="#">获取所有环境</a> 。

## 请求消息

表 4-16 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

表 4-17 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	环境名称。 由英文字母、数字、中划线（-）、下划线（_）组成，并以英文字母开头，英文字母或者数字结尾，英文字母区分大小写，长度为2~64个字符。
description	否	String	环境描述。 最大长度为128个字符。
labels	否	Array of objects	标签，可使用标签管理服务通过标签过滤资源，请参考 <a href="#">表4-18</a> 。 同一个环境下最多可以添加20个标签。

表 4-18 labels

参数	是否必选	参数类型	描述
key	是	String	标签名称。 标签用于标识资源，当您拥有相同类型的许多资源时，可以使用标签按各种维度（例如用途、所有者或环境）对资源进行分类。 长度为1到36个字符，只能包含数字、英文字母、下划线（_）、中划线（-）或中文。
value	是	String	标签值。 长度为1到43个字符，只能包含数字、英文字母、下划线（_）、点（.）、中划线（-）或中文。

## 响应消息

表 4-19 响应参数

参数	参数类型	描述
id	String	环境ID。
name	String	环境名称。
description	String	环境描述。
project_id	String	项目ID。
enterprise_project_id	String	企业项目ID。
deploy_mode	String	环境类型。
vpc_id	String	虚拟私有云ID。
creator	String	环境创建人。
create_time	Integer	环境创建时间。
update_time	Integer	环境修改时间。
labels	Array of objects	标签，可使用标签管理服务通过标签过滤资源，请参考 <a href="#">表4-20</a> 。
resources	Array of objects	资源，请参考 <a href="#">表4-21</a> 。

表 4-20 labels

参数	参数类型	描述
key	String	标签名称。
value	String	标签值。

表 4-21 resources

参数	参数类型	描述
id	String	资源ID。
name	String	资源名称。

参数	参数类型	描述
type	String	<p>资源类型。</p> <ul style="list-style-type: none"> <li>• vpc, 虚拟私有云。</li> <li>• eip, 弹性公网IP。</li> <li>• elb, 弹性负载均衡。</li> <li>• cce, 云容器引擎。</li> <li>• ecs, 弹性云服务器。</li> <li>• as, 弹性伸缩组。</li> <li>• cse, ServiceComb引擎。</li> <li>• nacos, 注册配置中心。</li> <li>• ms_gateway, 应用网关。 仅“华东-上海一”、“亚太-新加坡”区域支持纳管应用网关, 用于使用全链路流量控制功能。</li> <li>• dcs, 分布式缓存服务。</li> <li>• rds, 云数据库。</li> <li>• subnet, 子网。</li> <li>• acl, 网络ACL。</li> </ul>

## 请求示例

修改环境名称为development-env, 描述设置为空, 为环境增加键值对为"test": "value"的标签。

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

## 响应示例

```
{
  "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"
  }
]
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在。
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.1.5 根据环境 ID 获取环境详细信息

### 功能介绍

此API用于通过环境ID获取环境详细信息。

### 调试

您可以在[API Explorer](#)中调试该接口。

### URI

GET /v3/{project\_id}/cas/environments/{environment\_id}

表 4-22 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
environment_id	是	String	环境ID。获取方法，请参考 <a href="#">获取所有环境</a> 。

## 请求消息

表 4-23 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

## 响应消息

表 4-24 响应参数

参数	参数类型	描述
id	String	环境ID。
name	String	环境名称。
description	String	环境描述。
project_id	String	项目ID。
enterprise_project_id	String	企业项目ID。
deploy_mode	String	环境类型。
vpc_id	String	虚拟私有云ID。
creator	String	环境创建人。
create_time	Integer	环境创建时间。
update_time	Integer	环境修改时间。
labels	Array of objects	标签，可使用标签管理服务通过标签过滤资源，请参考 <a href="#">表4-25</a> 。
resources	Array of objects	资源，请参考 <a href="#">表4-26</a> 。

表 4-25 labels

参数	参数类型	描述
key	String	标签名称。

参数	参数类型	描述
value	String	标签值。

表 4-26 resources

参数	参数类型	描述
id	String	资源ID。
name	String	资源名称。
type	String	资源类型。 <ul style="list-style-type: none"> <li>• vpc, 虚拟私有云。</li> <li>• eip, 弹性公网IP。</li> <li>• elb, 弹性负载均衡。</li> <li>• cce, 云容器引擎。</li> <li>• ecs, 弹性云服务器。</li> <li>• as, 弹性伸缩组。</li> <li>• cse, ServiceComb引擎。</li> <li>• nacos, 注册配置中心。</li> <li>• ms_gateway, 应用网关。 仅“华东-上海一”、“亚太-新加坡”区域支持纳管应用网关, 用于使用全链路流量控制功能。</li> <li>• dcs, 分布式缓存服务。</li> <li>• rds, 云数据库。</li> <li>• subnet, 子网。</li> <li>• acl, 网络ACL。</li> </ul>

## 请求示例

无

## 响应示例

```
{
  "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  
  }  
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在。
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.1.6 根据环境 ID 修改环境资源

### 功能介绍

此API用于通过环境ID修改环境资源。

### URI

PUT /v3/{project\_id}/cas/environments/{environment\_id}/resources

表 4-27 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
environment_id	是	String	环境ID。获取方法，请参考 <a href="#">获取所有环境</a> 。

## 请求消息

表 4-28 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

表 4-29 请求 Body 参数

参数	是否必选	参数类型	描述
resources	是	Array of objects	添加资源，请参考 <a href="#">表4-30</a> 。

表 4-30 resources

参数	是否必选	参数类型	描述
id	是	String	资源ID（为了保证接口调用正常，请您确保调用前输入正确的资源ID）。
name	否	String	资源名称。

参数	是否必选	参数类型	描述
type	是	String	资源类型。 <ul style="list-style-type: none"> <li>• eip, 弹性公网IP。</li> <li>• elb, 弹性负载均衡。</li> <li>• cce, 云容器引擎。</li> <li>• ecs, 弹性云服务器。</li> <li>• as, 弹性伸缩组。</li> <li>• cse, ServiceComb引擎。</li> <li>• nacos, 注册配置中心。</li> <li>• ms_gateway, 应用网关。 仅“华东-上海一”、“亚太-新加坡”区域支持纳管应用网关, 用于使用全链路流量控制功能。</li> <li>• dcs, 分布式缓存服务。</li> <li>• rds, 云数据库。</li> <li>• subnet, 子网。</li> <li>• acl, 网络ACL。</li> </ul>

## 响应消息

表 4-31 响应参数

参数	参数类型	描述
resources	Array of objects	资源, 请参考 <a href="#">表4-32</a> 。

表 4-32 schemas

参数	参数类型	描述
id	String	资源ID。
name	String	资源名称。

参数	参数类型	描述
type	String	资源类型。 <ul style="list-style-type: none"><li>• eip, 弹性公网IP。</li><li>• elb, 弹性负载均衡。</li><li>• cce, 云容器引擎。</li><li>• ecs, 弹性云服务器。</li><li>• as, 弹性伸缩组。</li><li>• cse, 微服务引擎。</li><li>• dcs, 分布式缓存服务。</li><li>• rds, 云数据库。</li></ul>

## 请求示例

修改环境ECS资源为id为67835bb3-1235-4cc9-be71-becbb2b4ca0d、名称为ecs-a5a9的弹性云服务器。

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

## 响应示例

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

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在。
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.1.7 根据环境 ID 查询环境纳管的资源

### 功能介绍

此API用于根据环境ID查询环境纳管的资源。

### 调试

您可以在[API Explorer](#)中调试该接口。

### URI

GET /v3/{project\_id}/cas/environments/{environment\_id}/resources

表 4-33 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
environment_id	是	String	环境ID。获取方法，请参考 <a href="#">获取所有环境</a> 。

### 请求消息

表 4-34 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">认证鉴权</a> 。



## 响应消息

表 4-35 响应参数

参数	参数类型	描述
resources	Array of objects	资源, 请参考 <a href="#">表4-36</a> 。

表 4-36 schemas

参数	参数类型	描述
id	String	资源ID。
name	String	资源名称。
type	String	资源类型。 <ul style="list-style-type: none"><li>• eip, 弹性公网IP。</li><li>• elb, 弹性负载均衡。</li><li>• cce, 云容器引擎。</li><li>• ecs, 弹性云服务器。</li><li>• dcs, 分布式缓存服务。</li><li>• rds, 云数据库。</li><li>• cse, ServiceComb引擎。</li><li>• nacos, 注册配置中心。</li><li>• ms_gateway, 应用网关。</li><li>• subnet, 子网。</li><li>• acl, 网络ACL。</li></ul>

## 请求示例

无

## 响应示例

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

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在。
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.2 应用

### 4.2.1 创建应用

#### 功能介绍

应用是一个功能相对完备的业务系统，由一个或多个特性相关的组件组成。  
此API用于创建应用。

#### 调试

您可以在[API Explorer](#)中调试该接口。

#### URI

POST /v3/{project\_id}/cas/applications

表 4-37 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。

## 请求消息

表 4-38 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

表 4-39 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	应用名称。 由英文字母、数字、中划线（-）或下划线（_）组成，并以英文字母开头，英文字母或者数字结尾，长度为2~64个字符。
description	否	String	应用描述。 最大长度为128个字符。
enterprise_project_id	否	String	企业项目ID。
labels	否	Array of objects	标签，可使用标签管理服务通过标签过滤资源，请参考 <a href="#">表4-40</a> 。 同一个应用下最多可以添加20个标签。

表 4-40 labels

参数	是否必选	参数类型	描述
key	是	String	标签名称。 标签用于标识资源，当您拥有相同类型的许多资源时，可以使用标签按各种维度（例如用途、所有者或环境）对资源进行分类。 长度为1到36个字符，只能包含数字、英文字母、下划线（_）、中划线（-）或中文。

参数	是否必选	参数类型	描述
value	是	String	标签值。 长度为1到43个字符，只能包含数字、英文字母、下划线（_）、点（.）、中划线（-）或中文。

## 响应消息

表 4-41 响应参数

参数	参数类型	描述
id	String	应用ID。
name	String	应用名称。
description	String	应用描述。
creator	String	应用创建人。
project_id	String	项目ID。
enterprise_project_id	String	企业项目ID。
create_time	Integer	应用创建时间。
update_time	Integer	应用修改时间。
labels	Array of objects	标签，可使用标签管理服务通过标签过滤资源，请参考 <a href="#">表4-42</a> 。

表 4-42 labels

参数	参数类型	描述
key	String	标签名称。
value	String	标签值。

## 请求示例

创建名称为app-xpmtii的应用，描述信息输入test，标签键值对设置为"key":"var"。

```
{
  "name": "app-xpmtii",
  "description": "test",
  "labels": [
    {
      "key": "key",
```

```
    "value": "var"  
  }  
]  
}
```

## 响应示例

```
{  
  "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"  
    }  
  ]  
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在。
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.2.2 获取所有应用

### 功能介绍

此API用于获取所有已经创建的应用。

### 调试

您可以在[API Explorer](#)中调试该接口。

### URI

GET /v3/{project\_id}/cas/applications

表 4-43 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。

表 4-44 查询参数

参数	是否必选	参数类型	描述
limit	否	integer	查询个数，取值[0, 100]。
offset	否	integer	查询偏移量，表示查询第n条数据后的结果。
order_by	否	String	排序字段。 排序字段支持以下枚举值： <ul style="list-style-type: none"><li>• create_time，应用创建时间，默认排序字段。</li><li>• name，应用名称。</li><li>• update_time，应用修改时间。</li></ul> 传值不在支持的枚举值范围内的话按默认排序字段排序。
order	否	String	排序方式。 <ul style="list-style-type: none"><li>• desc，降序排序，默认排序方式。</li><li>• asc，升序排序。</li></ul>

## 请求消息

表 4-45 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

## 响应消息

表 4-46 响应参数

参数	参数类型	描述
count	Integer	应用总数。
applications	Array of object	应用信息，请参考 <a href="#">表4-47</a> 。

表 4-47 applications

参数	参数类型	描述
id	String	应用ID。
name	String	应用名称。
description	String	应用描述。
creator	String	应用创建人。
project_id	String	项目ID。
enterprise_project_id	String	企业项目ID。
create_time	Integer	应用创建时间。
update_time	Integer	应用修改时间。
component_count	Integer	应用下部署的组件个数。
labels	Array of objects	标签，可使用标签管理服务通过标签过滤资源，请参考 <a href="#">表4-48</a> 。

表 4-48 labels

参数	参数类型	描述
key	String	标签名称。
value	String	标签值。

## 请求示例

无

## 响应示例

```
{
  "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
    }
  ]
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在。
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

### 4.2.3 根据应用 ID 修改应用信息

#### 功能介绍

此API用于通过应用ID修改应用信息。

#### 调试

您可以在[API Explorer](#)中调试该接口。

#### URI

PUT /v3/{project\_id}/cas/applications/{application\_id}



表 4-49 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
application_id	是	String	应用ID。获取方法，请参考 <a href="#">获取所有应用</a> 。

## 请求消息

表 4-50 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

表 4-51 请求 Body 参数

参数	是否必选	参数类型	描述
name	否	String	应用名称。 由英文字母、数字、中划线（-）或下划线（_）组成，并以英文字母开头，英文字母或者数字结尾，长度为2~64个字符。
description	否	String	应用描述。 最大长度为128个字符。
enterprise_project_id	否	String	企业项目ID。
labels	否	Array of objects	标签，可使用标签管理服务通过标签过滤资源，请参考 <a href="#">表4-52</a> 。 同一个应用下最多可以添加20个标签。

表 4-52 labels

参数	是否必选	参数类型	描述
key	是	String	标签名称。 标签用于标识资源，当您拥有相同类型的许多资源时，可以使用标签按各种维度（例如用途、所有者或环境）对资源进行分类。 长度为1到36个字符，只能包含数字、英文字母、下划线（_）、中划线（-）或中文。
value	是	String	标签值。 长度为1到43个字符，只能包含数字、英文字母、下划线（_）、点（.）、中划线（-）或中文。

## 响应消息

表 4-53 响应参数

参数	参数类型	描述
id	String	应用ID。
name	String	应用名称。
description	String	应用描述。
creator	String	应用创建人。
project_id	String	项目ID。
enterprise_project_id	String	企业项目ID。
create_time	Integer	应用创建时间。
update_time	Integer	应用修改时间。
component_count	Integer	应用下部署的组件个数。
labels	Array of objects	标签，可使用标签管理服务通过标签过滤资源，请参考 <a href="#">表4-54</a> 。

表 4-54 labels

参数	参数类型	描述
key	String	标签名称。
value	String	标签值。

## 请求示例

修改应用名称为app-xpmtii2，描述信息修改为test2，标签键值对修改为"key":"var"。

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

## 响应示例

```
{
  "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"
    }
  ]
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在。
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.2.4 根据应用 ID 删除应用

### 功能介绍

此API用于通过应用ID删除应用。

## 调试

您可以在[API Explorer](#)中调试该接口。

## URI

DELETE /v3/{project\_id}/cas/applications/{application\_id}

表 4-55 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
application_id	是	String	应用ID。获取方法，请参考 <a href="#">获取所有应用</a> 。

## 请求消息

表 4-56 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

## 响应消息

无

## 请求示例

无

## 响应示例

无

## 状态码

状态码	描述
204	操作成功。

状态码	描述
400	错误的请求。
404	请求对象不存在。
409	请求与当前对象状态冲突。
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.2.5 根据应用 ID 获取应用详细信息

### 功能介绍

此API用于通过应用ID获取应用详细信息。

### 调试

您可以在[API Explorer](#)中调试该接口。

### URI

GET /v3/{project\_id}/cas/applications/{application\_id}

表 4-57 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
application_id	是	String	应用ID。获取方法，请参考 <a href="#">获取所有应用</a> 。

### 请求消息

表 4-58 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

## 响应消息

表 4-59 响应参数

参数	参数类型	描述
id	String	应用ID。
name	String	应用名称。
description	String	应用描述。
creator	String	应用创建人。
project_id	String	项目ID。
enterprise_project_id	String	企业项目ID。
create_time	Integer	应用创建时间。
update_time	Integer	应用修改时间。
component_count	Integer	应用下部署的组件个数。
labels	Array of objects	标签，可使用标签管理服务通过标签过滤资源，请参考 <a href="#">表4-60</a> 。

表 4-60 labels

参数	参数类型	描述
key	String	标签名称。
value	String	标签值。

## 请求示例

无

## 响应示例

```
{  
  "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  
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在。
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.2.6 根据应用 ID 获取应用配置

### 功能介绍

此API用于通过应用ID获取应用配置信息。

### 调试

您可以在[API Explorer](#)中调试该接口。

### URI

GET /v3/{project\_id}/cas/applications/{application\_id}/configuration

表 4-61 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
application_id	是	String	应用ID。获取方法，请参考 <a href="#">获取所有应用</a> 。

表 4-62 查询参数

参数	是否必选	参数类型	描述
environment_id	否	String	环境ID。获取方法，请参考 <a href="#">获取所有环境</a> 。

## 请求消息

表 4-63 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

## 响应消息

表 4-64 响应参数

参数	参数类型	描述
configuration	Array of objects	配置列表，请参考 <a href="#">表4-65</a> 。

表 4-65 configuration

参数	参数类型	描述
application_id	String	应用ID。
environment_id	String	环境ID。
configuration	Object	环境变量配置，请参考 <a href="#">表4-66</a> 。



表 4-66 configurationEnv

参数	参数类型	描述
env	Array of object	环境变量，请参考 <a href="#">表4-67</a> 。

表 4-67 env

参数	参数类型	描述
name	String	变量名。
value	String	变量值。

## 请求示例

无

## 响应示例

```
{
  "configuration": [
    {
      "application_id": "fd7889d-eef8-4db1-befc-7037b02d949e",
      "environment_id": "0fa75dfe-0d32-4bc1-848b-8008cf3f2567",
      "configuration": {
        "env": [
          {
            "name": "app-env",
            "value": "env"
          }
        ]
      }
    }
  ]
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在。
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.2.7 根据应用 ID 修改应用配置

### 功能介绍

此API用于通过应用ID修改应用配置。

### 调试

您可以在[API Explorer](#)中调试该接口。

### URI

PUT /v3/{project\_id}/cas/applications/{application\_id}/configuration

表 4-68 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
application_id	是	String	应用ID。获取方法，请参考 <a href="#">获取所有应用</a> 。

### 请求消息

表 4-69 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

表 4-70 请求 Body 参数

参数	是否必选	参数类型	描述
environment_id	是	String	环境ID。获取方法，请参考 <a href="#">获取所有环境</a> 。
configuration	是	Object	环境变量配置，请参考 <a href="#">表4-71</a> 。

表 4-71 configuration

参数	是否必选	参数类型	描述
env	是	Array of objects	环境变量，请参考 <a href="#">表4-72</a> 。

表 4-72 env

参数	是否必选	参数类型	描述
name	是	String	变量名，应用环境变量的名称，在同一个应用环境下必须唯一。 长度1到64个字符，可以包含英文字母、数字、下划线(_)、中划线(-)或点(.)，以英文字母、下划线(_)或中划线(-)开头。
value	是	String	变量值，应用环境变量的值。

## 响应消息

表 4-73 响应参数

参数	参数类型	描述
application_id	String	应用ID。
environment_id	String	环境ID。
configuration	Object	环境变量配置，请参考 <a href="#">表4-74</a> 。

表 4-74 configuration

参数	参数类型	描述
env	Array of objects	环境变量，请参考 <a href="#">表4-75</a> 。

表 4-75 env

参数	参数类型	描述
name	String	变量名。
value	String	变量值。

## 请求示例

修改应用对应环境id为0fa75dfe-0d32-4bc1-848b-8008cf3f2567的环境变量配置为名为app-env、值为env的环境变量。

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

## 响应示例

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

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在。
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.2.8 根据应用 ID 删除应用配置

### 功能介绍

此API用于通过应用ID删除应用配置。

### 调试

您可以在[API Explorer](#)中调试该接口。

### URI

DELETE /v3/{project\_id}/cas/applications/{application\_id}/configuration

表 4-76 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
application_id	是	String	应用ID。获取方法，请参考 <a href="#">获取所有应用</a> 。

表 4-77 查询参数

参数	是否必选	参数类型	描述
environment_id	是	String	环境ID。获取方法，请参考 <a href="#">获取所有环境</a> 。

### 请求消息

表 4-78 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

## 响应消息

无

## 请求示例

无

## 响应示例

无

## 状态码

状态码	描述
204	操作成功。
400	错误的请求。
404	请求对象不存在。
409	请求与当前对象状态冲突。
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.3 组件

### 4.3.1 应用中创建组件

#### 功能介绍

组件是组成应用的某个业务特性实现，以代码或者软件包为载体，可独立部署在环境下运行。

此API用于在应用中创建组件。

#### 调试

您可以在[API Explorer](#)中调试该接口。

#### URI

POST /v3/{project\_id}/cas/applications/{application\_id}/components

表 4-79 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
application_id	是	String	应用ID。获取方法，请参考 <a href="#">获取所有应用</a> 。

## 请求消息

表 4-80 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

表 4-81 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	组件名称。 由英文字母、数字、中划线（-）或下划线（_）组成，并以英文字母开头，英文字母或者数字结尾，长度为2~64个字符。
workload_name	否	String	工作负载名称。
labels	否	Array of objects	标签。 容器部署方式部署的组件，可通过标签查询对应工作负载；虚机部署方式部署的组件可配置对应的微服务名称的标签来使用优雅上下线功能。请参考 <a href="#">表4-82</a> 。
runtime_stack	是	Object	运行时，请参考 <a href="#">表4-108</a> ，可通过查询 <a href="#">查询运行时栈</a> 接口返回的信息获取。
environment_id	否	String	环境ID，可通过 <a href="#">获取所有环境</a> 接口返回的信息获取。
description	否	String	组件描述。 最大长度为128个字符。

参数	是否必选	参数类型	描述
source	是	Object	代码/软件包来源, 请参考 <a href="#">表4-109</a> 。
build	否	Object	组件构建, 容器部署并没有使用镜像时必传, 请参考 <a href="#">表4-110</a> 。
limit_cpu	否	Number	最大CPU限制, 单位为Core。
limit_memory	否	Number	最大内存限制, 单位为GiB。
request_cpu	否	Number	申请CPU资源, 单位为Core。
request_memory	否	Number	申请内存资源, 单位为GiB。
version	是	String	组件版本号, 最大长度32, 规则为: $^([0-9]+)(.[0-9]+){2,3}$$
envs	否	Array of objects	环境变量列表, 请参考 <a href="#">表4-106</a> 。
replica	是	Integer	实例数。
storages	否	Array of objects	存储, 请参考 <a href="#">表4-83</a> 。
deploy_strategy	否	Object	组件部署, 请参考 <a href="#">表4-86</a> 。
command	否	Object	启动命令, 请参考 <a href="#">表4-90</a> 。
post_start	否	Object	启动后处理, 请参考 <a href="#">表4-91</a> 。
pre_stop	否	Object	停止前处理, 请参考 <a href="#">表4-91</a> 。
mesher	否	Object	多语言接入服务网格, 当技术栈选择Node.js且绑定CSE引擎时为必选, 请参考 <a href="#">表4-92</a> 。
timezone	否	String	指定组件运行的时区, 比如Asia/Shanghai。
jvm_opts	否	String	jvm参数, 例如-Xms256m -Xmx1024m, 多个参数以空格间隔, 不填则使用默认值。



参数	是否必选	参数类型	描述
tomcat_opts	否	Object	tomcat参数，技术栈选择tomcat时生效，请参考 <a href="#">表4-93</a> 。
host_aliases	否	Array of objects	主机别名，请参考 <a href="#">表4-94</a> 。
dns_policy	否	String	DNS策略。 <ul style="list-style-type: none"><li>• Default，继承pod所在节点域名解析配置。</li><li>• ClusterFirst，追加域名解析配置。</li><li>• ClusterFirstWithHostNet，仅对以hostNetWork方式运行的pod生效。</li><li>• None，替换域名解析策略。</li></ul>
dns_config	否	Object	DNS配置，请参考 <a href="#">表4-95</a> 。
workload_kind	否	String	工作负载类型。 <ul style="list-style-type: none"><li>• deployment，无状态工作负载。</li><li>• statefulset，有状态工作负载。</li><li>• daemonset，守护进程集。</li></ul>
security_context	否	Object	安全策略，请参考 <a href="#">表4-97</a> 。
logs	否	Array of objects	日志采集，请参考 <a href="#">表4-99</a> 。
custom_metrics	否	Object	自定义指标监控，请参考 <a href="#">表4-100</a> 。
affinity	否	Array of objects	亲和性，请参考 <a href="#">表4-101</a> 。
anti_affinity	否	Array of objects	反亲和性，请参考 <a href="#">表4-101</a> 。
liveness_probe	否	Object	组件存活探针，请参考 <a href="#">表4-103</a> 。
readiness_probe	否	Object	组件业务探针，请参考 <a href="#">表4-103</a> 。

参数	是否必选	参数类型	描述
refer_resources	是	Array of objects	关联资源, 请参考 <a href="#">表4-104</a> 。
external_accesses	否	Array of Objects	外部访问, 请参考 <a href="#">表4-112</a> 。

表 4-82 labels

参数	是否必选	参数类型	描述
key	是	String	标签名称。 1到36个字符, 只能包含数字、英文字母、下划线(_)、中划线(-)或中文。 <b>须知</b> Key的取值不能是系统内置的app、casid或version, 否则会导致创建并部署组件时报错。
value	是	String	标签值。 0到43个字符, 只能包含数字、英文字母、下划线(_)、点(.)、中划线(-)或中文。

表 4-83 storages

参数	是否必选	参数类型	描述
type	是	String	数据存储类型。 <ul style="list-style-type: none"><li>HostPath, 本地磁盘主机路径挂载。</li><li>EmptyDir, 本地磁盘临时路径挂载。</li><li>ConfigMap, 本地磁盘配置项挂载。</li><li>Secret, 本地磁盘密钥挂载。</li><li>PersistentVolumeClaim, 云存储挂载。</li></ul>
name	是	String	存储盘名称。
parameters	是	Object	对应各种数据存储类型的信息参数, 请参考 <a href="#">表4-84</a> 。

参数	是否必选	参数类型	描述
mounts	是	array of objects	数据存储挂载路径，请参考表4-85。

表 4-84 component\_storage\_parameters

参数	是否必选	参数类型	描述
path	否	String	主机路径，适用于HostPath的存储类型。当数据存储类型为HostPath为必选参数。
name	否	String	配置项、密钥或者PVC的名字，适用于ConfigMap、Secret和PersistentVolumeClaim的存储类型。当数据存储类型为ConfigMap、Secret和PersistentVolumeClaim为必选参数。
default_mode	否	Integer	挂载的权限，十进制格式，例如：384。适用于ConfigMap和Secret的存储类型。
medium	否	String	适用于EmptyDir类型的存储。不传参数为默认的磁盘介质，传参为memory则开启内存存储。

表 4-85 mounts

参数	是否必选	参数类型	描述
path	是	String	挂载路径，数据存储挂载到应用上的路径。例如，/tmp。
sub_path	是	String	挂载路径的子路径。
read_only	是	Boolean	是否只读。 <ul style="list-style-type: none"> <li>• true: 只读，只能读应用路径中的数据存储。</li> <li>• false: 读写，可修改应用路径中的数据存储，应用迁移时新写入的数据不会随之迁移，会造成数据丢失。</li> </ul>

表 4-86 deploy\_strategy

参数	是否必选	参数类型	描述
type	是	String	部署类型。 <ul style="list-style-type: none"><li>• OneBatchRelease, 单批升级。</li><li>• RollingRelease, 滚动部署升级。</li><li>• GrayRelease, 灰度发布升级。</li></ul>
rolling_release	否	Object	滚动部署参数, 部署类型为RollingRelease时为必选, 请参考表4-87。
gray_release	否	Object	灰度发布升级, 部署类型为GrayRelease时为必选, 请参考表4-88。

表 4-87 rolling\_release

参数	是否必选	参数类型	描述
batches	是	Integer	分批部署批次。

表 4-88 gray\_release

参数	是否必选	参数类型	描述
type	是	String	灰度策略。 <ul style="list-style-type: none"><li>• WEIGHT, 按灰度流量进行升级。</li><li>• CONTENT, 按内容进行灰度。</li></ul>
first_batch_weight	是	Integer	首批灰度流量比例, 灰度策略为WEIGHT时为必选。
first_batch_replica	是	Integer	首批灰度实例数量, 灰度策略为WEIGHT时为必选。
remaining_batch	是	Integer	剩余实例部署批次, 首批灰度发布成功之后, 剩余实例滚动升级分多少批次完成。 例如: 剩余实例数是5, 剩余部署批次是3, 那么升级剩余实例会按照2:2:1个实例分批升级。 灰度类型为WEIGHT时为必选。
deployment_mode	否	Integer	部署模型。 <ul style="list-style-type: none"><li>• 1, 上游是微服务网关。</li><li>• 3, 上游是微服务。</li><li>• 4, 上游是ELB。</li></ul>

参数	是否必选	参数类型	描述
replica_surge_mode	否	String	灰度实例新增模式，灰度策略为CONTENT时为必选。 <ul style="list-style-type: none"> <li>• MIRROR, 蓝绿。</li> <li>• EXTRA, 金丝雀（先增后减）。</li> <li>• NOSURGE, 金丝雀（先减后增）。</li> </ul>
rule_match_mode	否	String	灰度规则生效方式，灰度策略为CONTENT时为必选。 <ul style="list-style-type: none"> <li>• ALL, 满足所有条件。</li> <li>• ANY, 满足任一条件。</li> </ul>
rules	否	Object	灰度规则，参数表4-89，灰度策略为CONTENT时为必选。

表 4-89 gray\_rules

参数	是否必选	参数类型	描述
type	是	String	匹配类型。 当前仅支持：header，请求头。
key	是	String	参数名称。
value	是	String	条件值。
condition	是	String	条件类型。 <ul style="list-style-type: none"> <li>• equal, 相等。</li> <li>• match, 匹配。</li> <li>• in, 枚举。</li> </ul>

表 4-90 command

参数	是否必选	参数类型	描述
command	否	Array of String	运行命令，控制容器运行的输入命令。
args	否	Array of String	运行参数，控制容器运行命令的输入参数。 例如：-port=8080，多条参数以换行分隔。

表 4-91 component\_lifecycle

参数	是否必选	参数类型	描述
type	是	String	处理方式。 <ul style="list-style-type: none"><li>• http, http请求方式。</li><li>• command, 命令行方式。</li></ul>
scheme	否	String	http请求类型参数: HTTP、HTTPS。 type为http类型时生效。
host	否	String	默认为POD的IP, 可以指定自定义的IP。 type为http类型时生效。
port	否	Integer	端口号, type为http类型时生效。
path	否	String	请求路径, type为http类型时生效。
command	否	Array of String	命令列表, type为command类型时生效。

表 4-92 mesher

参数	是否必选	参数类型	描述
port	是	Integer	进程监听端口。

表 4-93 tomcat\_opt

参数	是否必选	参数类型	描述
server_xml	是	String	<ol style="list-style-type: none"><li>1. 监听端口以server.xml配置的为准, 如果之前在公网访问配置了监听端口, 请保持一致。</li><li>2. 自定义的Tomcat配置可能存在端口冲突, 请检查自定义的端口是否被占用。</li></ol>

表 4-94 host\_aliases

参数	是否必选	参数类型	描述
ip	是	String	ip地址。

参数	是否必选	参数类型	描述
hostnames	是	Array of String	主机别名。

表 4-95 dns\_config

参数	是否必选	参数类型	描述
nameservers	否	Array of String	Pod的DNS服务器的IP地址列表。
searches	否	Array of String	可选，用于在Pod中查找主机名的DNS搜索域的列表。
options	否	Array of objects	可选，对象列表。其中每个对象可能具有name属性（必需）和value属性（可选）。此属性中的内容将合并到从指定的DNS策略生成的选项，重复的条目将被删除，请参考表4-96。

表 4-96 options

参数	是否必选	参数类型	描述
name	是	String	操作名称。
value	否	String	操作值。

表 4-97 security\_context

参数	是否必选	参数类型	描述
run_as_user	否	Integer	运行用户，容器以当前用户权限运行，如以root权限运行则填写root用户ID 0。
run_as_group	否	Integer	运行属组，指定所有容器中的进程都以属组运行。
capabilities	否	Object	能力集，请参考表4-98。

表 4-98 capabilities

参数	是否必选	参数类型	描述
add	否	Array of String	为运行用户添加Linux权限。
drop	否	Array of String	移除运行用户Linux权限。

表 4-99 logs

参数	是否必选	参数类型	描述
log_path	是	String	容器中日志路径。
rotate	是	String	日志转储周期。
host_path	是	String	挂载的主机路径。
host_extend_path	是	String	主机扩展路径，通过扩展主机路径，实现同一个主机路径下区分来自不同容器的挂载： <ul style="list-style-type: none"> <li>• None，不使用扩展路径。</li> <li>• PodUID，通过POD的ID扩展主机路径。</li> <li>• PodName，通过POD的名称扩展主机路径。</li> <li>• PodUID/ContainerName，通过POD的ID和容器名称扩展主机路径。</li> <li>• PodName/ContainerName，通过POD的名称和容器名称扩展主机路径。</li> </ul>

表 4-100 custom\_metric

参数	是否必选	参数类型	描述
path	是	String	采集路径，例如：/metrics。
port	是	Integer	采集端口，例如：9090。
dimensions	是	String	监控维度，例如："cpu_usage,mem_usage"。



表 4-101 component\_affinity

参数	是否必选	参数类型	描述
condition	是	String	是否必须满足。 <ul style="list-style-type: none"><li>• required, 必须满足。</li><li>• preferred, 尽量满足。</li></ul>
kind	是	String	亲和类型。 <ul style="list-style-type: none"><li>• node, 节点亲和。</li><li>• pod, pod亲和。</li></ul>
weight	否	Integer	亲和权重, 取值范围0~100。
match_expressions	是	Array of objects	匹配条件, 请参考 <a href="#">表4-102</a> 。

表 4-102 match\_expressions

参数	是否必选	参数类型	描述
key	是	String	匹配标签key值。
operation	是	String	匹配条件。
value	是	String	匹配标签value值。

表 4-103 component\_probe

参数	是否必选	参数类型	描述
type	是	String	类型: http、tcp、command。
delay	是	Integer	启动后多久开始探测。
timeout	是	Integer	探测超时时间。
scheme	否	String	请求类型: HTTP、HTTPS, type为http类型时生效。
host	否	String	默认为POD的IP, 可以指定自定义的IP。type为http类型时生效。
port	否	Integer	端口号, type为http和tcp类型时生效。
path	否	String	请求路径, type为http类型时生效。

参数	是否必选	参数类型	描述
command	否	Array of String	命令列表，type为command类型时生效。

表 4-104 refer\_resources

参数	是否必选	参数类型	描述
id	是	String	资源ID。
type	是	String	资源类型： <ul style="list-style-type: none"> <li>• vpc: 虚拟私有云</li> <li>• eip: 弹性公网IP</li> <li>• elb: 弹性负载均衡</li> <li>• cce: 云容器引擎</li> <li>• ecs: 弹性云服务器</li> <li>• as: 弹性伸缩组</li> <li>• cse: ServiceComb引擎</li> <li>• nacos: 注册配置中心</li> <li>• dcs: 分布式缓存服务</li> <li>• rds: 云数据库</li> </ul>
parameters	否	Object	资源参数，请参考 <a href="#">表4-105</a> 。

表 4-105 refer\_resource\_parameter

参数	是否必选	参数类型	描述
namespace	否	String	命名空间。

表 4-106 env

参数	是否必选	参数类型	描述
name	是	String	变量名。
value	否	String	变量值。
value_from	否	Object	变量引用，请参考 <a href="#">表4-107</a> 。

表 4-107 value\_from

参数	是否必选	参数类型	描述
reference_type	是	String	引用类型。 <ul style="list-style-type: none"><li>• configMapKey, 配置项导入。</li><li>• secretKey, 密钥导入。</li></ul>
name	是	String	配置项或密钥的名称。
key	否	String	配置项或密钥的键值。
optional	否	Boolean	配置项或密钥的key是否必须存在。

表 4-108 runtime\_stack

参数	是否必选	参数类型	描述
name	是	String	技术栈名称。
type	是	String	技术栈类型, 支持Java、Tomcat、Nodejs、Php、Docker、Python。 当部署模式为虚拟机部署时, 仅支持Java、Tomcat、Nodejs; 容器部署上述类型都支持。
version	是	String	技术栈版本。
deploy_mode	是	String	部署模式。 <ul style="list-style-type: none"><li>• container, 容器部署。</li><li>• virtualmachine, 虚拟机部署。</li></ul>

表 4-109 source

参数	是否必选	参数类型	描述
kind	是	String	组件来源类型。 <ul style="list-style-type: none"><li>• code, 源码。</li><li>• package, 软件包。</li><li>• image, 镜像包。</li></ul>
version	否	String	版本号。
url	否	String	包地址, 类型为package和image时需要添加。

参数	是否必选	参数类型	描述
storage	否	String	存储方式，支持镜像仓库swr、软件仓库swr、软件开发生产线codearts和对象存储obs，类型为package和image时需要添加。
auth	否	String	认证方式，支持iam、none，默认是iam。
repo_auth	否	String	授权名称，在授权列表获取，请参考 <a href="#">获取仓库授权列表</a> 。
repo_namepace	否	String	代码仓库命名空间。
repo_ref	否	String	代码分支或者Tag，默认是master。
repo_type	否	String	代码仓类型，支持CodeArts、Gitee、GitHub、BitBucket、GitLab。
web_url	否	String	代码跳转链接，例如： <a href="https://github.com/example/demo.git">https://github.com/example/demo.git</a> 。
repo_url	否	String	代码仓url，例如： <a href="https://github.com/example/demo.git">https://github.com/example/demo.git</a> 。
commit_id	否	String	代码仓库中，commit id是每次提交的唯一标识符，用于识别和引用特定的提交。 长度为8到40个字符，以小写字母或数字开头。

表 4-110 build

参数	是否必选	参数类型	描述
parameters	否	Map<String, Object>	只有在没有ID，新创建构建时提供，请参考 <a href="#">表 4-111</a> 。

表 4-111 parameters

参数	是否必选	参数类型	描述
build_cmd	否	String	编译命令。默认： 1. 根目录存在build.sh: ./build.sh 2. 根据运行系统，示例如下： <ul style="list-style-type: none"> <li>• Java和Tomcat: <b>mvn clean package</b></li> <li>• Nodejs: <b>npm build</b></li> </ul>

参数	是否必选	参数类型	描述
dockerfile_path	否	String	dockerfile地址，默认是根目录./。
artifact_namespace	否	String	构建归档组织，默认cas_{project_id}。
cluster_id	是	String	指定构建集群的id。
node_label_selector	否	Map<String, String>	key是标签的键，value是标签的值。
environment_id	否	String	环境ID。获取方法，请参考 <a href="#">获取所有环境</a> 。

表 4-112 external\_accesses

参数	是否必选	参数类型	描述
protocol	否	String	外部访问类型，取值范围为：http、https。
address	否	String	外部访问地址。
forward_port	否	Integer	外部访问端口。

## 响应消息

表 4-113 响应参数

参数	参数类型	描述
job_id	String	任务ID。
component_id	String	应用组件ID。

## 请求示例

- 场景一：CCE容器环境下创建组件

创建名为mycomponent的组件。组件部署使用1.2.0版本的Java技术栈，组件来源选择存储在OBS下bucket0001桶里的weather-1.0.0.jar包，部署使用的资源为id为fd92bfab-73b3-11ee-90f0-0255ac1001b3的云容器引擎。

```
{
  "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
}
```

- 场景二：ECS虚拟机环境下创建组件

创建名为mycomponent的组件。组件部署使用1.2.0版本的Java技术栈，组件来源选择存储在OBS下bucket0001桶里的weather-1.0.0.jar包，部署使用的资源为id为67835bb3-1235-4cc9-be71-becbb2b4ca0d的弹性云服务器。

```
{
    "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  
    }  
  ]  
}
```

## 响应示例

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

## 状态码

状态码	描述
200	操作成功
400	错误的请求
404	请求对象不存在
500	内部错误

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.3.2 应用中导入 CCE 工作负载创建组件

### 功能介绍

在CCE中，工作负载（Workload）是指一组以某种方式协同工作的Pod集合，它们为了完成一个特定的应用程序或服务而存在。

此API用于在应用中导入CCE中已经创建的工作负载用于创建组件。

### URI

POST /v3/{project\_id}/cas/applications/{application\_id}/components

表 4-114 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。

参数	是否必选	参数类型	描述
application_id	是	String	应用ID。获取方法，请参考 <a href="#">获取所有应用</a> 。

## 请求消息

表 4-115 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">认证鉴权</a> 。

表 4-116 请求 Body 参数

参数	是否必选	参数类型	描述
create_mode	是	String	组件创建方式。 导入CCE中已经创建的工作负载用于创建组件，该参数需要设置为import。
name	是	String	应用组件名称。 由英文字母、数字、中划线(-)、下划线(_)组成，并以英文字母开头，英文字母或者数字结尾，长度为2~64个字符。
workload_name	是	String	CCE上已经创建的工作负载的名称。
labels	否	Array of objects	标签。 容器部署可通过标签查询对应工作负载，请参考 <a href="#">表4-117</a> 。
runtime_stack	是	Object	运行时，固定为Docker。
environment_id	是	String	环境ID。
description	否	String	描述。 最大长度为128个字符。



参数	是否必选	参数类型	描述
version	是	String	组件版本号，最大长度32，规则为： ^([0-9]+)(.[0-9]+){2,3}\$
workload_kind	是	String	工作负载类型。 <ul style="list-style-type: none"> <li>• deployment，无状态工作负载。</li> <li>• statefulset，有状态工作负载。</li> </ul>
refer_resources	是	Array of objects	关联资源，请参考表4-118。 导入的工作负载需要进行微服务灰度发布时，需关联ServiceComb引擎或者Nacos引擎。
external_accesses	否	Array of Objects	外部访问，请参考表4-120。

表 4-117 labels

参数	是否必选	参数类型	描述
key	是	String	标签名称。
value	是	String	标签值。

表 4-118 refer\_resources

参数	是否必选	参数类型	描述
id	是	String	资源id。
type	是	String	资源类型： <ul style="list-style-type: none"> <li>• vpc：虚拟私有云</li> <li>• eip：弹性公网IP</li> <li>• elb：弹性负载均衡</li> <li>• cce：云容器引擎</li> <li>• dcs：分布式缓存服务</li> <li>• rds：云数据库</li> <li>• cse：ServiceComb引擎</li> <li>• nacos：注册配置中心</li> </ul>
parameters	否	Object	资源参数，请参考表4-119。

表 4-119 refer\_resource\_parameter

参数	是否必选	参数类型	描述
namespace	否	String	命名空间。

表 4-120 external\_accesses

参数	是否必选	参数类型	描述
protocol	否	String	外部访问类型，取值范围为：http、https。
address	否	String	外部访问地址。
forward_port	否	Integer	外部访问端口。

## 响应消息

表 4-121 响应参数

参数	参数类型	描述
instance_id	String	组件实例ID。

## 请求示例

通过从id为1d23eab7-da00-11ee-a1e1-0255ac1001ba的CCE中导入test-cyf-deployment工作负载，创建名为testcyf-import-deployment的无状态组件。

```
{
  "name": "testcyf-import-deployment",
  "description": "",
  "labels": [],
  "version": "2024.0506.16306",
  "environment_id": "9cff7557-4ebb-481b-bc1e-6a6573b5b8d2",
  "application_id": "563cab6a-53fb-4f5d-8921-de1bfe36db00",
  "enterprise_project_id": "0",
  "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"  
}
```

## 响应示例

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

## 状态码

状态码	描述
200	操作成功
400	错误的请求
404	请求对象不存在
500	内部错误

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.3.3 获取应用所有组件

### 功能介绍

此API用于获取应用下所有应用组件。

### 调试

您可以在[API Explorer](#)中调试该接口。

### URI

GET /v3/{project\_id}/cas/applications/{application\_id}/components

表 4-122 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
application_id	是	String	应用ID。获取方法，请参考 <a href="#">获取所有应用</a> 。

表 4-123 查询参数

参数	是否必选	参数类型	描述
limit	否	integer	查询个数，取值[0, 100]。
offset	否	integer	查询偏移量，表示查询第n条数据后的结果。
order_by	否	String	排序字段。 排序字段支持以下枚举值： <ul style="list-style-type: none"><li>• create_time，组件创建时间，默认排序字段。</li><li>• name，组件名称。</li><li>• update_time，组件修改时间。</li></ul> 传值不在支持的枚举值范围内的话按默认排序字段排序。
order	否	String	排序方式。 <ul style="list-style-type: none"><li>• desc，降序排序，默认排序方式。</li><li>• asc，升序排序。</li></ul>

## 请求消息

表 4-124 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

## 响应消息

表 4-125 响应参数

参数	参数类型	描述
count	Integer	组件总数。
components	Array of objects	组件参数，请参考 <a href="#">表4-126</a> 。

表 4-126 components

参数	参数类型	描述
id	String	组件ID。
name	String	组件名称。
labels	Array of objects	标签。容器部署方式部署的组件可通过标签查询对应工作负载；虚拟机部署方式部署的组件可配置对应的微服务名称的标签来使用优雅上下线功能，请参考表4-131。
status	Object	组件状态，请参考表4-129。
runtime_stack	Object	技术栈，请参考表4-128。
application_name	String	应用名称。
application_id	String	应用ID。
environment_name	String	环境名称。
environment_id	String	环境ID。
version	String	组件版本号。
source	Object	代码/软件包来源，请参考表4-127。
creator	String	组件创建人。
platform_type	String	组件部署类型。
external_accesses	Array of objects	外部访问，请参考表4-132。

表 4-127 source

参数	参数类型	描述
kind	String	组件来源类型。 <ul style="list-style-type: none"><li>code, 源码。</li><li>package, 软件包。</li><li>image, 镜像包。</li></ul>
version	String	版本号。
url	String	包地址，类型为package和image时需要添加。
storage	String	存储方式，支持镜像仓库swr、软件仓库swr、软件开发生产线codearts和对象存储obs，类型为package和image时需要添加。

参数	参数类型	描述
auth	String	认证方式，支持iam、none，默认是iam。
repo_auth	String	授权名称，在授权列表获取。
repo_namespace	String	代码仓库命名空间。
repo_ref	String	代码分支或者Tag，默认是master。
repo_type	String	代码仓类型，支持CodeArts、Gitee、GitHub、BitBucket、GitLab。
web_url	String	代码跳转链接，例如：https://github.com/example/demo.git。
repo_url	String	代码仓url，例如：https://github.com/example/demo.git。

表 4-128 runtime\_stack

参数	参数类型	描述
name	String	技术栈名称，例如：OpenJDK8。
type	String	技术栈类型，支持Java、Tomcat、Nodejs、Php、Docker、Python。 当部署模式为虚拟机部署时，仅支持Java、Tomcat、Nodejs；容器部署上述类型都支持。
version	String	技术栈版本，例如：1.1.4。
deploy_mode	String	部署模式。 <ul style="list-style-type: none"><li>容器部署，container。</li><li>虚拟机部署，virtualmachine。</li></ul>

表 4-129 status

参数	参数类型	描述
component_status	String	组件状态。 <ul style="list-style-type: none"><li>INITIALIZING, 初始化。</li><li>UPGRADING, 升级中。</li><li>FAILED, 失败。</li><li>RUNNING, 运行中。</li><li>DOWN, 下线。</li><li>DELETING, 删除中。</li><li>DELETED, 已删除。</li><li>RESERVED, 保留。</li><li>STARTING, 启动中。</li><li>STOPPING, 停止中。</li><li>STOPPED, 停止。</li><li>RESTARTING, 重启中。</li><li>PENDING, 未就绪。</li><li>UNKNOWN, 未知。</li><li>PARTIALLY_FAILED, 部分失败。</li></ul>
available_replica	Integer	可用实例。
replica	Integer	总实例。
fail_detail	String	失败原因。 <ul style="list-style-type: none"><li>cluster_deleted, 集群被删除。</li><li>cluster_unavailable, 集群不可用。</li><li>cluster_inaccessible, 集群无法访问。</li><li>namespace_deleted, 命名空间被删除。</li><li>namespace_unavailable, 命名空间不可用。</li><li>namespace_inaccessible, 命名空间无法访问。</li><li>resource_deleted, 资源已删除。</li><li>create_failed, 创建组件失败。</li><li>delete_failed, 删除组件失败。</li></ul>
last_job_id	String	最后执行任务ID。
create_time	Integer	创建时间。
update_time	Integer	更新时间。
creator	String	创建人。

参数	参数类型	描述
artifact	Object	组件构建信息，请参考 <a href="#">表4-130</a> 。

表 4-130 artifact

参数	参数类型	描述
type	String	组件来源类型。 <ul style="list-style-type: none"> <li>虚拟机部署支持软件包package。</li> <li>容器部署支持软件包package、镜像包image。</li> </ul>
url	String	软件包/镜像地址。

表 4-131 labels

参数	参数类型	描述
key	String	标签名称。
value	String	标签值。

表 4-132 external\_accesses

参数	参数类型	描述
protocol	String	外部访问类型，取值范围为：http、https。
address	String	外部访问地址。
forward_port	Integer	外部访问端口。

## 请求示例

无

## 响应示例

```
{
  "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
}
```

## 状态码

状态码	描述
200	操作成功
400	错误的请求
404	请求对象不存在
500	内部错误

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

### 4.3.4 获取所有组件

#### 功能介绍

此API用于获取所有应用组件。

#### 调试

您可以在[API Explorer](#)中调试该接口。

#### URI

GET /v3/{project\_id}/cas/components

表 4-133 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。

表 4-134 查询参数

参数	是否必选	参数类型	描述
limit	否	integer	查询个数，取值[0, 100]。
offset	否	integer	查询偏移量，表示查询第n条数据后的结果。
order_by	否	String	排序字段。 排序字段支持以下枚举值： <ul style="list-style-type: none"><li>• create_time，组件创建时间，默认排序字段。</li><li>• name，组件名称。</li><li>• update_time，组件修改时间。</li></ul> 传值不在支持的枚举值范围内的话按默认排序字段排序。
order	否	String	排序方式。 <ul style="list-style-type: none"><li>• desc，降序排序，默认排序方式。</li><li>• asc，升序排序。</li></ul>
application_name	否	String	应用名称。
component_name	否	String	组件名称。

## 请求消息

表 4-135 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

## 响应消息

表 4-136 响应参数

参数	参数类型	描述
count	Integer	组件总数。
components	Array of objects	组件参数，请参考 <a href="#">表4-137</a> 。

表 4-137 components

参数	参数类型	描述
id	String	组件ID。
name	String	组件名称。
labels	Array of objects	标签。容器部署可通过标签查询对应工作负载，虚拟机部署可配置对应的微服务名称的标签来使用优雅上下线功能，请参考 <a href="#">表4-142</a> 。
status	Object	组件状态，请参考 <a href="#">表4-140</a> 。
runtime_stack	Object	技术栈，请参考 <a href="#">表4-139</a> 。
application_name	String	应用名称。
application_id	String	应用ID。
environment_name	String	环境名称。
environment_id	String	环境ID。
version	String	版本号。
source	Object	代码/软件包来源，请参考 <a href="#">表4-138</a> 。
creator	String	创建人。
platform_type	String	部署类型。
external_accesses	Array of Objects	外部访问，请参考 <a href="#">表4-143</a> 。

表 4-138 source

参数	参数类型	描述
kind	String	组件来源类型。 <ul style="list-style-type: none"> <li>code, 源码。</li> <li>package, 软件包。</li> <li>image, 镜像包。</li> </ul>
version	String	版本号。
url	String	包地址, 类型为package和image时需要添加。
storage	String	存储方式, 例如: obs。
auth	String	认证方式, 例如: iam。
repo_auth	String	授权名称。
repo_namespace	String	代码仓库命名空间。
repo_ref	String	代码分支或者Tag, 例如: master。
repo_type	String	代码仓类型, 例如: CodeArts、Gitee、GitHub、BitBucket、GitLab。
web_url	String	代码跳转链接, 例如: https://github.com/example/demo.git。
repo_url	String	代码仓url, 例如: https://github.com/example/demo.git。

表 4-139 runtime\_stack

参数	参数类型	描述
name	String	技术栈名称。
type	String	技术栈类型, 支持Java、Tomcat、Nodejs、Php、Docker、Python。 当部署模式为虚拟机部署时, 仅支持Java、Tomcat、Nodejs; 容器部署上述类型都支持。
version	String	技术栈版本。
deploy_mode	String	部署模式。

表 4-140 status 参数

参数	参数类型	描述
component_status	String	组件状态。 <ul style="list-style-type: none"><li>INITIALIZING, 初始化。</li><li>UPGRADING, 升级中。</li><li>FAILED, 失败。</li><li>RUNNING, 运行中。</li><li>DOWN, 下线。</li><li>DELETING, 删除中。</li><li>DELETED, 已删除。</li><li>RESERVED, 保留。</li><li>STARTING, 启动中。</li><li>STOPPING, 停止中。</li><li>STOPPED, 停止。</li><li>RESTARTING, 重启中。</li><li>PENDING, 未就绪。</li><li>UNKNOWN, 未知。</li><li>PARTIALLY_FAILED, 部分失败。</li></ul>
available_replica	Integer	可用实例。
replica	Integer	总实例。
fail_detail	String	失败原因。 <ul style="list-style-type: none"><li>cluster_deleted, 集群被删除。</li><li>cluster_unavailable, 集群不可用。</li><li>cluster_inaccessible, 集群无法访问。</li><li>namespace_deleted, 命名空间被删除。</li><li>namespace_unavailable, 命名空间不可用。</li><li>namespace_inaccessible, 命名空间无法访问。</li><li>resource_deleted, 资源已删除。</li><li>create_failed, 创建组件失败。</li><li>delete_failed, 删除组件失败。</li></ul>
last_job_id	String	最后执行任务ID。
create_time	Integer	创建时间。
update_time	Integer	更新时间。
creator	String	创建人。

参数	参数类型	描述
artifact	Object	组件构建信息，请参考 <a href="#">表4-141</a> 。

表 4-141 artifact

参数	参数类型	描述
type	String	组件来源类型。 <ul style="list-style-type: none"> <li>虚拟机部署支持软件包package。</li> <li>容器部署支持软件包package、镜像包image。</li> </ul>
url	String	软件包/镜像地址。

表 4-142 labels

参数	参数类型	描述
key	String	标签名称
value	String	标签值

表 4-143 external\_accesses

参数	参数类型	描述
protocol	String	外部访问类型，取值范围为：http、https。
address	String	外部访问地址。
forward_port	Integer	外部访问端口。

## 请求示例

无

## 响应示例

```
{
  "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
}
```

## 状态码

状态码	描述
200	操作成功
400	错误的请求
404	请求对象不存在
500	内部错误

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

### 4.3.5 根据组件 ID 修改组件信息

#### 功能介绍

此API用于通过组件ID修改组件信息。

#### 调试

您可以在[API Explorer](#)中调试该接口。

## URI

PUT /v3/{project\_id}/cas/applications/{application\_id}/components/  
{component\_id}

表 4-144 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
application_id	是	String	应用ID。获取方法，请参考 <a href="#">获取所有应用</a> 。
component_id	是	String	组件ID。获取方法，请参考 <a href="#">获取应用所有组件</a> 。

## 请求消息

表 4-145 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

表 4-146 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	应用组件名称。不可修改。 由英文字母、数字、中划线（-）、下划线（_）组成，并以英文字母开头，英文字母或者数字结尾，长度为2~64个字符。
labels	否	Array of objects	标签。容器部署可通过标签查询对应工作负载，虚拟机部署可配置对应的微服务名称的标签来使用优雅上下线功能，请参考 <a href="#">表4-147</a> 。



参数	是否必选	参数类型	描述
runtime_stack	是	Object	运行时，不可修改，请参考 <a href="#">表4-173</a> ，可通过查询 <a href="#">查询运行时栈</a> 接口返回的信息获取。
description	否	String	描述。 最大长度为128个字符。
source	是	Object	代码/软件包来源，请参考 <a href="#">表4-174</a> 。
build	否	Object	组件构建，请参考 <a href="#">表4-175</a> 。
limit_cpu	否	Number	最大CPU限制，单位为Core。
limit_memory	否	Number	最大内存限制，单位为GiB。
request_cpu	否	Number	申请CPU资源，单位为Core。
request_memory	否	Number	申请内存资源，单位为GiB。
version	是	String	组件版本号，最大长度32个字符，规则为： ^([0-9]+)(.[0-9]+){2,3}\$
envs	否	Array of objects	环境变量列表，请参考 <a href="#">表4-171</a> 。
replica	是	Integer	实例数，仅在使用扩缩容功能时支持进行修改。
storages	否	Array of objects	存储，请参考 <a href="#">表4-148</a> 。
deploy_strategy	否	Object	组件部署，请参考 <a href="#">表4-151</a> 。
command	否	Object	启动命令，请参考 <a href="#">表4-155</a>
post_start	否	Object	启动后处理，请参考 <a href="#">表4-156</a> 。
pre_stop	否	Object	停止前处理，请参考 <a href="#">表4-156</a> 。
mesher	否	Object	多语言接入服务网格，当技术栈选择Nodejs且绑定微服务引擎CSE时为必选，请参考 <a href="#">表4-157</a> 。

参数	是否必选	参数类型	描述
timezone	否	String	指定组件运行的时区，比如Asia/Shanghai。
jvm_opts	否	String	jvm参数，例如-Xms256m -Xmx1024m，多个参数以空格间隔，不填则使用默认值。
tomcat_opts	否	Object	tomcat参数，技术栈选择tomcat时生效，请参考 <a href="#">表4-158</a> 。
host_aliases	否	Array of objects	主机别名，请参考 <a href="#">表4-159</a> 。
dns_policy	否	String	DNS策略。 <ul style="list-style-type: none"><li>• Default，继承pod所在节点域名解析配置。</li><li>• ClusterFirst，追加域名解析配置。</li><li>• ClusterFirstWithHostNet，仅对以hostNetWork方式运行的pod生效。</li><li>• None，替换域名解析策略。</li></ul>
dns_config	否	Object	DNS配置，请参考 <a href="#">表4-160</a> 。
workload_kind	否	String	工作负载类型。 <ul style="list-style-type: none"><li>• deployment，无状态工作负载。</li><li>• statefulset，有状态工作负载。</li><li>• daemonset，守护进程集。</li></ul>
security_context	否	Object	安全策略，请参考 <a href="#">表4-162</a> 。
logs	否	Array of objects	日志采集，请参考 <a href="#">表4-164</a> 。
custom_metric	否	Object	自定义指标监控，请参考 <a href="#">表4-165</a> 。
affinity	否	Array of objects	亲和性，请参考 <a href="#">表4-166</a> 。
anti_affinity	否	Array of objects	反亲和性，请参考 <a href="#">表4-166</a> 。

参数	是否必选	参数类型	描述
liveness_probe	否	Object	组件存活探针, 请参考 <a href="#">表4-168</a> 。
readiness_probe	否	Object	组件业务探针, 请参考 <a href="#">表4-168</a> 。
refer_resources	否	Array of objects	关联资源, 请参考 <a href="#">表4-169</a> 。
external_accesses	否	Array of objects	外部访问, 请参考 <a href="#">表4-177</a> 。
tolerations	否	Array of Objects	容忍度策略, 请参考 <a href="#">表4-178</a> 。
container_name	否	String	容器名称。
update_strategy	否	Object	组件升级策略, 请参考 <a href="#">表4-179</a> 。
multi_containers	否	Array of Objects	多容器配置, 请参考 <a href="#">表4-180</a> 。
workload_content	否	String	工作负载内容。
configs	否	Array of Objects	组件统一配置, 请参考 <a href="#">表4-181</a> 。
config_mode	否	String	工作负载配置模式。 <ul style="list-style-type: none"> <li>ui: 基于ui页面配置选项部署。</li> <li>yaml: 基于yaml文件部署。</li> </ul>
enable_serment_injection	否	Boolean	是否开启Serment。
enable_serment_spring_register	否	Boolean	Spring Cloud Huawei框架的微服务是否需要Serment帮助注册到注册中心。
enable_serment_config	否	Boolean	是否开启Serment动态配置插件能力。

参数	是否必选	参数类型	描述
cell_id	否	String	部署单元ID。
swimlane_id	否	String	泳道ID。
sermantAgentInjectActionFlag	否	Boolean	JDK17以及JDK21是否自动添加以下JVM环境变量。 JDK17: --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 JDK21: --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	否	Map<String, String>	使用压缩包在虚拟机环境下部署组件配置项。
image_pull_secrets	否	Array of String	镜像拉取密钥列表。

表 4-147 labels

参数	是否必选	参数类型	描述
key	是	String	标签名称。
value	是	String	标签值。

表 4-148 storages

参数	是否必选	参数类型	描述
type	是	String	数据存储类型。 <ul style="list-style-type: none"> <li>HostPath, 本地磁盘主机路径挂载。</li> <li>EmptyDir, 本地磁盘临时路径挂载。</li> <li>ConfigMap, 本地磁盘配置项挂载。</li> <li>Secret, 本地磁盘密钥挂载。</li> <li>PersistentVolumeClaim, 云存储挂载。</li> </ul>
name	是	String	存储盘名称。
parameters	是	Object	对应各种数据存储类型的信息参数, 请参考表 4-149。
mounts	是	Array of objects	数据存储挂载路径, 请参考表 4-150。

表 4-149 component\_storage\_parameters

参数	是否必选	参数类型	描述
path	否	String	主机路径, 适用于HostPath的存储类型。当数据存储类型为HostPath为必选参数。
name	否	String	配置项、密钥或者PVC的名字, 适用于ConfigMap、Secret和PersistentVolumeClaim的存储类型。当数据存储类型为ConfigMap、Secret和PersistentVolumeClaim为必选参数。
default_mode	否	Integer	挂载的权限, 十进制格式, 例如: 384。适用于ConfigMap和Secret的存储类型。
medium	否	String	适用于EmptyDir类型的存储。不传参数为默认的磁盘介质, 传参为memory则开启内存存储。

表 4-150 mounts

参数	是否必选	参数类型	描述
path	是	String	挂载路径
sub_path	是	String	挂载路径的子路径

参数	是否必选	参数类型	描述
read_only	是	Boolean	是否只读

表 4-151 deploy\_strategy

参数	是否必选	参数类型	描述
type	是	String	部署类型。 <ul style="list-style-type: none"><li>• OneBatchRelease, 单批升级。</li><li>• RollingRelease, 滚动部署升级。</li><li>• GrayRelease, 灰度发布升级。</li></ul>
rolling_release	否	Object	滚动部署参数, 部署类型为RollingRelease时为必选, 请参考表4-152。
gray_release	否	Object	灰度发布升级, 部署类型为GrayRelease时为必选, 请参考表4-153。

表 4-152 rolling\_release

参数	是否必选	参数类型	描述
batches	是	Integer	分批部署批次。

表 4-153 gray\_release

参数	是否必选	参数类型	描述
type	是	String	灰度策略。 <ul style="list-style-type: none"><li>• WEIGHT, 按灰度流量进行升级</li><li>• CONTENT, 按内容进行灰度</li></ul>
first_batch_weight	是	Integer	首批灰度流量比例, 灰度策略为WEIGHT时为必选。
first_batch_replica	是	Integer	首批灰度实例数量, 灰度策略为WEIGHT时为必选。

参数	是否必选	参数类型	描述
remaining_batch	是	Integer	剩余实例部署批次，首批灰度发布成功之后，剩余实例滚动升级分多少批次完成。 例如：剩余实例数是5，剩余部署批次是3，那么升级剩余实例会按照2:2:1个实例分批升级。 灰度类型为WEIGHT时为必选。
deployment_mode	否	Integer	部署模型。 <ul style="list-style-type: none"><li>1，上游是微服务网关。</li><li>3，上游是微服务。</li><li>4，上游是ELB。</li></ul>
replica_surge_mode	否	String	灰度实例新增模式，灰度策略为CONTENT时为必选。 <ul style="list-style-type: none"><li>MIRROR，蓝绿。</li><li>EXTRA，金丝雀（先增后减）。</li><li>NOSURGE，金丝雀（先减后增）。</li></ul>
rule_match_mode	否	String	灰度规则生效方式，灰度策略为CONTENT时为必选。 <ul style="list-style-type: none"><li>ALL，满足所有条件。</li><li>ANY，满足任一条件。</li></ul>
rules	否	Object	灰度规则，灰度类型为CONTENT时为必选。请参考 <a href="#">表4-154</a> 。

表 4-154 gray\_rules

参数	是否必选	参数类型	描述
type	是	String	匹配类型。 当前仅支持：HEADER，请求头。
key	是	String	参数名称。
value	是	String	条件值。
condition	是	String	条件类型。 <ul style="list-style-type: none"><li>EQUAL，相等。</li><li>MATCH，匹配。</li><li>IN，枚举。</li></ul>

表 4-155 command

参数	是否必选	参数类型	描述
command	否	Array of String	运行命令，控制容器运行的输入命令。
args	否	Array of String	运行参数，控制容器运行命令的输入参数。例如：-port=8080，多条参数以换行分隔。

表 4-156 component\_lifecycle

参数	是否必选	参数类型	描述
type	是	String	处理方式。 <ul style="list-style-type: none"><li>• http, http请求方式。</li><li>• command, 命令行方式。</li></ul>
scheme	否	String	http请求类型参数: HTTP、HTTPS, type为http类型时生效。
host	否	String	默认为POD的IP, 可以指定自定义的IP。type为http类型时生效。
port	否	Integer	端口, type为http类型时生效。
path	否	String	请求路径, type为http类型时生效。
command	否	Array of String	命令列表, type为command类型时生效。

表 4-157 mesher

参数	是否必选	参数类型	描述
port	是	Integer	进程监听端口。



表 4-158 tomcat\_opt

参数	是否必选	参数类型	描述
server_xml	是	String	1. 监听端口以server.xml配置的为准，如果之前在公网访问配置了监听端口，请保持一致。 2. 自定义的Tomcat配置可能存在端口冲突，请检查自定义的端口是否被占用。

表 4-159 host\_aliases

参数	是否必选	参数类型	描述
ip	是	String	IP地址
hostnames	是	Array of String	主机别名

表 4-160 dns\_config

参数	是否必选	参数类型	描述
nameservers	否	Array of String	Pod的DNS服务器的IP地址列表。
searches	否	Array of String	可选，用于在Pod中查找主机名的DNS搜索域的列表。
options	否	Array of objects	可选的对象列表，其中每个对象可能具有name属性（必需）和value属性（可选）。此属性中的内容将合并到从指定的DNS策略生成的选项。重复的条目将被删除。请参考 <a href="#">表4-161</a> 。

表 4-161 options

参数	是否必选	参数类型	描述
name	是	String	操作名称
value	否	String	操作值

表 4-162 security\_context

参数	是否必选	参数类型	描述
run_as_user	否	Integer	运行用户，容器以当前用户权限运行，如以root权限运行则填写root用户ID 0。
run_as_group	否	Integer	运行属组，指定所有容器中的进程都以属组运行。
capabilities	否	Object	能力集，请参考表4-163。

表 4-163 capabilities

参数	是否必选	参数类型	描述
add	否	Array of String	为运行用户添加Linux权限。
drop	否	Array of String	移除运行用户Linux权限。

表 4-164 logs

参数	是否必选	参数类型	描述
log_path	是	String	容器中日志路径。
rotate	是	String	日志转储周期。
host_path	是	String	挂载的主机路径。
host_extend_path	是	String	主机扩展路径，通过扩展主机路径，实现同一个主机路径下区分来自不同容器的挂载： <ul style="list-style-type: none"><li>• None，不使用扩展路径。</li><li>• PodUID，通过POD的ID扩展主机路径。</li><li>• PodName，通过POD的名称扩展主机路径。</li><li>• PodUID/ContainerName，通过POD的ID和容器名称扩展主机路径。</li><li>• PodName/ContainerName，通过POD的名称和容器名称扩展主机路径。</li></ul>

表 4-165 custom\_metric

参数	是否必选	参数类型	描述
path	是	String	采集路径，例如：/metrics。
port	是	Integer	采集端口，例如：9090。
dimensions	是	String	监控维度，例如："cpu_usage,mem_usage"。

表 4-166 component\_affinity

参数	是否必选	参数类型	描述
condition	是	String	是否必须满足。 <ul style="list-style-type: none"> <li>required, 必须满足。</li> <li>preferred, 尽量满足。</li> </ul>
kind	是	String	亲和类型。 <ul style="list-style-type: none"> <li>node, 节点亲和。</li> <li>pod, pod亲和。</li> </ul>
weight	否	Integer	亲和权重，取值范围0~100。
match_expressions	是	Array of objects	匹配条件，请参考 <a href="#">表4-167</a> 。

表 4-167 match\_expressions

参数	是否必选	参数类型	描述
key	是	String	匹配标签key值。
operation	是	String	匹配条件。
value	是	String	匹配标签value值。

表 4-168 component\_probe

参数	是否必选	参数类型	描述
type	是	String	类型：http、tcp、command。

参数	是否必选	参数类型	描述
delay	是	Integer	启动后多久开始探测。
timeout	是	Integer	探测超时时间。
scheme	否	String	请求类型: HTTP、HTTPS, type为http类型时生效。
host	否	String	默认为POD的IP, 可以指定自定义的IP。type为http类型时生效。
port	否	Integer	端口号, type为http和tcp类型时生效
path	否	String	请求路径, type为http类型时生效。
command	否	Array of String	命令列表, type为command类型时生效。

表 4-169 refer\_resources

参数	是否必选	参数类型	描述
id	是	String	资源id。
type	是	String	资源类型。 <ul style="list-style-type: none"><li>• vpc: 虚拟私有云</li><li>• eip: 弹性公网IP</li><li>• elb: 弹性负载均衡</li><li>• cce: 云容器引擎</li><li>• ecs: 弹性云服务器</li><li>• as: 弹性伸缩组</li><li>• cse: ServiceComb引擎</li><li>• nacos: 注册配置中心</li><li>• dcs: 分布式缓存服务</li><li>• rds: 云数据库</li></ul>
parameters	否	Object	资源参数, 当资源类型为cce时必选, 请参考 <a href="#">表 4-170</a> 。

表 4-170 refer\_resource\_parameter

参数	是否必选	参数类型	描述
namespace	否	String	命名空间

表 4-171 env

参数	是否必选	参数类型	描述
name	是	String	变量名。
value	否	String	变量值。
value_from	否	Object	变量引用, 请参考 <a href="#">表4-172</a> 。

表 4-172 value\_from

参数	是否必选	参数类型	描述
reference_type	是	String	引用类型。 <ul style="list-style-type: none"> <li>configMapKey, 配置项导入。</li> <li>secretKey, 密钥导入。</li> </ul>
name	是	String	配置项或密钥的名称。
key	否	String	配置项或密钥的键值。
optional	否	Boolean	配置项或密钥的key是否必须存在: true、false。

表 4-173 runtime\_stack

参数	是否必选	参数类型	描述
name	是	String	技术栈名称。
type	是	String	技术栈类型, 支持Java、Tomcat、Nodejs、Php、Docker、Python。 当部署模式为虚拟机部署时, 仅支持Java、Tomcat、Nodejs; 容器部署上述类型都支持。
version	是	String	技术栈版本。

参数	是否必选	参数类型	描述
deploy_mode	是	String	部署模式。 <ul style="list-style-type: none"> <li>• container, 容器部署。</li> <li>• virtualmachine, 虚拟机部署。</li> </ul>

表 4-174 source

参数	是否必选	参数类型	描述
kind	是	String	组件来源类型。 <ul style="list-style-type: none"> <li>• code, 源码。</li> <li>• package, 软件包。</li> <li>• image, 镜像包。</li> </ul>
version	否	String	版本号。
url	否	String	包地址, 类型为package和image时需要添加。
storage	否	String	存储方式, 支持镜像仓库swr、软件仓库swr、软件开发生产线codearts和对象存储obs, 类型为package和image时需要添加。
auth	否	String	认证方式, 支持iam、none, 默认是iam。
repo_auth	否	String	授权名称, 在授权列表获取。
repo_namespace	否	String	代码仓库命名空间。
repo_ref	否	String	代码分支或者Tag, 默认是master。
repo_type	否	String	代码仓类型, 支持CodeArts、Gitee、GitHub、BitBucket、GitLab。
web_url	否	String	码跳转链接, 例如: <a href="https://github.com/example/demo.git">https://github.com/example/demo.git</a> 。
repo_url	否	String	代码仓url, 例如: <a href="https://github.com/example/demo.git">https://github.com/example/demo.git</a> 。
commit_id	否	String	代码仓库中, commit id是每次提交的唯一标识符, 用于识别和引用特定的提交。 长度为8到40个字符, 以小写字母或数字开头。

表 4-175 build

参数	是否必选	参数类型	描述
parameters	否	Map<String, Object>	只有在没有ID, 新创建构建时提供。请参考 <a href="#">表 4-176</a> 。

表 4-176 parameters

参数	是否必选	参数类型	描述
build_cmd	否	String	编译命令。默认： 1. 根目录存在build.sh: ./build.sh 2. 根据运行系统，示例如下： <ul style="list-style-type: none"><li>• Java和Tomcat: <b>mvn clean package</b></li><li>• Nodejs: <b>npm build</b></li></ul>
dockerfile_path	否	String	dockerfile地址。默认是根目录./。
artifact_namespace	否	String	构建归档组织，默认为cas_{project_id}。
cluster_id	是	String	指定构建集群的id。
node_label_selector	否	Map<String, String>	key是标签的键，value是标签的值。
environment_id	否	String	环境ID。

表 4-177 external\_accesses

参数	是否必选	参数类型	描述
protocol	否	String	外部访问类型，取值范围为：http、https。
address	否	String	外部访问地址。
forward_port	否	Integer	外部访问端口。

表 4-178 tolerations

参数	是否必选	参数类型	描述
key	否	String	污点键。
operator	否	String	操作符。 <ul style="list-style-type: none"><li>Equal: 设置此操作符表示准确匹配指定污点键（必填）和污点值的节点。如果不填写污点值，则表示可以与所有污点键相同的污点匹配。</li><li>Exists: 设置此操作符表示匹配存在指定污点键的节点，此时容忍度不能指定污点值。若不填写污点键则可以容忍全部污点。</li></ul>
value	否	String	污点值。“operator”为“Equal”时，需要填写污点值。
effect	否	String	污点策略。 <ul style="list-style-type: none"><li>NoScheduler: 表示匹配污点效果为NoSchedule的污点。</li><li>PreferNoSchedule: 表示匹配污点效果为PreferNoSchedule的污点。</li><li>NoExecute: 表示匹配污点效果为NoExecute的污点。</li><li>不设置该参数，表示匹配所有污点效果。</li></ul>
toleration_seconds	否	Integer	容忍时间窗。 当“effect”为“NoExecute”时支持配置。 在容忍时间窗内，Pod还会在拥有污点的节点上运行，超出时间后会被驱逐。

表 4-179 update\_strategy

参数	是否必须	参数类型	描述
type	否	String	组件升级类型： <ul style="list-style-type: none"><li>RollingUpdate, 滚动升级，即逐步创建新实例再删除旧实例，为默认策略。</li><li>Recreate, 替换升级，即先把当前实例删掉再重新创建实例。</li></ul>



参数	是否必须	参数类型	描述
max_unavailable	否	<ul style="list-style-type: none"> <li>String</li> <li>Integer</li> </ul>	<p>最大无效实例数。与replica相比，可以有多少个实例失效，也就是删除的比例，默认值是25%。workload_kind为deployment时支持设置。</p> <ul style="list-style-type: none"> <li>若下发百分比则为String类型，即与实例数相比无效实例数在实例数中的最大百分比。</li> <li>若下发整数则为Integer类型，即与实例数相比最多可以有多少个无效实例数。</li> </ul>
max_surge	否	<ul style="list-style-type: none"> <li>String</li> <li>Integer</li> </ul>	<p>最大浪涌。与replica相比，可以有多少个实例存在，默认值是25%。仅workload_kind为deployment时支持设置。</p> <ul style="list-style-type: none"> <li>若下发百分比则为String类型。</li> <li>若下发整数则为Integer类型。</li> </ul>
termination_period_seconds	否	Integer	<p>缩容时间窗。工作负载停止前命令的执行时间窗，默认值为30秒。</p>
min_ready_seconds	否	Integer	<p>实例可用最短时间。指定新创建的实例在没有任意容器崩溃情况下的最小就绪时间，只有超出这个时间实例才被视为可用。 默认值为 0。</p>
progress_deadline_seconds	否	Integer	<p>升级最大时长。指定系统在报告组件升级失败之前等待组件取得进展的秒数。 如果指定，则此字段值需要大于min_ready_seconds取值。</p>
revision_history_limit	否	Integer	<p>最大保留版本数。用来设定出于回滚目的所要保留的旧实例数量。默认情况下，系统保留10个旧实例。</p>

表 4-180 multi\_containers

参数	是否必选	参数类型	描述
container_name	是	String	容器名称。

参数	是否必选	参数类型	描述
source	是	Object	代码/软件包来源, 请参考 <a href="#">表4-174</a> , 多容器模式下仅支持镜像。
limit_cpu	否	Number	最大CPU限制, 单位为Core。
limit_memory	否	Number	最大内存限制, 单位为GiB。
request_cpu	否	Number	申请CPU资源, 单位为Core。
request_memory	否	Number	申请内存资源, 单位为GiB。
envs	否	Array of objects	环境变量列表, 请参考 <a href="#">表4-171</a> 。
liveness_probe	否	Object	容器存活探针, 请参考 <a href="#">表4-168</a> 。
readiness_probe	否	Object	容器业务探针, 请参考 <a href="#">表4-168</a> 。
startup_probe	否	Object	容器启动探针, 请参考 <a href="#">表4-168</a> 。
command	否	Object	启动命令, 请参考 <a href="#">表4-155</a> 。
post_start	否	Object	启动后处理, 请参考 <a href="#">表4-156</a> 。
pre_stop	否	Object	停止前处理, 请参考 <a href="#">表4-156</a> 。
logs	否	Array of objects	日志采集, 请参考 <a href="#">表4-164</a> 。
storages	否	Array of objects	存储, 请参考 <a href="#">表4-148</a> 。
security_context	否	Object	安全策略, 请参考 <a href="#">表4-162</a> 。

表 4-181 configs

参数	是否必须	参数类型	描述
config_id	是	String	统一配置ID。
mount_path	是	String	挂载路径。

## 响应消息

表 4-182 响应参数

参数	参数类型	描述
job_id	String	任务ID。

## 请求示例

修改名为mycomponet的组件，组件来源使用OBS上bucket0001桶里面的weather-1.0.0.jar包。

```
{
  "name": "mycomponet",
  "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": []
}
```

## 响应示例

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

## 状态码

状态码	描述
200	操作成功。

状态码	描述
400	错误的请求。
404	请求对象不存在。
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.3.6 根据组件 ID 删除组件

### 功能介绍

此API用于通过组件ID删除组件。

### 调试

您可以在[API Explorer](#)中调试该接口。

### URI

DELETE /v3/{project\_id}/cas/applications/{application\_id}/components/{component\_id}

表 4-183 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
application_id	是	String	应用ID。获取方法，请参考 <a href="#">获取所有应用</a> 。
component_id	是	String	组件ID。获取方法，请参考 <a href="#">获取应用所有组件</a> 。

### 请求消息

表 4-184 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

## 响应消息

无

## 请求示例

无

## 响应示例

无

## 状态码

状态码	描述
200	操作成功
400	错误的请求
404	请求对象不存在
409	请求与当前对象状态冲突
500	内部错误

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.3.7 根据组件 ID 获取组件信息

### 功能介绍

此API用于通过组件ID获取组件信息。

### 调试

您可以在[API Explorer](#)中调试该接口。

## URI

GET /v3/{project\_id}/cas/applications/{application\_id}/components/  
{component\_id}

表 4-185 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
application_id	是	String	应用ID。获取方法，请参考 <a href="#">获取所有应用</a> 。
component_id	是	String	组件ID。获取方法，请参考 <a href="#">获取应用所有组件</a> 。

表 4-186 查询参数

参数	是否必选	参数类型	描述
version	否	String	组件版本号。获取方法，请参考 <a href="#">获取应用所有组件</a> 。

## 请求消息

表 4-187 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

## 响应消息

表 4-188 响应参数

参数	参数类型	描述
name	String	应用组件名称。 由英文字母、数字、中划线 (-)、下划线 (_) 组成，并以英文字母开头，英文字母或者数字结尾，不区分英文字母大小写，长度为2~64个字符。
workload_name	String	工作负载名称。
labels	Array of objects	标签。容器部署可通过标签查询对应工作负载，虚拟机部署可配置对应的微服务名称的标签来使用优雅上下线功能，请参考 <a href="#">表4-189</a> 。
runtime_stack	Object	运行时，请参考 <a href="#">表4-215</a> ，可通过查询 <a href="#">查询运行时栈</a> 接口返回的信息获取。
environment_id	String	环境ID。
application_id	String	应用ID。
description	String	描述。 最大长度为128个字符。
source	Object	代码/软件包来源，请参考 <a href="#">表4-216</a> 。
build	Object	组件构建，请参考 <a href="#">表4-217</a> 。
limit_cpu	Number	最大CPU限制，单位为Core。
limit_memory	Number	最大内存限制，单位为GiB。
request_cpu	Number	申请CPU资源，单位为Core。
request_memory	Number	申请内存资源，单位为GiB。
version	String	组件版本号，最大长度32个字符，规则为： ^([0-9]+)(.[0-9]+){2,3}\$
envs	Array of objects	环境变量列表，请参考 <a href="#">表4-213</a> 。
replica	Integer	实例数。
storages	Array of objects	存储，请参考 <a href="#">表4-190</a> 。
deploy_strategy	Object	组件部署，请参考 <a href="#">表4-193</a> 。

参数	参数类型	描述
command	Object	启动命令, 请参考 <a href="#">表4-197</a>
post_start	Object	启动后处理, 请参考 <a href="#">表4-198</a> 。
pre_stop	Object	停止前处理, 请参考 <a href="#">表4-198</a> 。
mesher	Object	多语言接入服务网格, 当技术栈选择Nodejs且绑定cse引擎时为必选, 请参考 <a href="#">表4-199</a> 。
timezone	String	指定组件运行的时区, 比如Asia/Shanghai。
jvm_opts	String	jvm参数, 例如-Xms256m -Xmx1024m, 多个参数以空格间隔, 不填则使用默认值。
tomcat_opts	Object	tomcat参数, 技术栈选择tomcat时生效, 请参考 <a href="#">表4-200</a> 。
host_aliases	Array of objects	主机别名, 请参考 <a href="#">表4-201</a> 。
dns_policy	String	DNS策略。 <ul style="list-style-type: none"><li>• Default, 继承pod所在节点域名解析配置。</li><li>• ClusterFirst, 追加域名解析配置。</li><li>• ClusterFirstWithHostNet, 仅对以hostNetWork方式运行的pod生效。</li><li>• None, 替换域名解析策略。</li></ul>
dns_config	Object	DNS配置, 请参考 <a href="#">表4-202</a> 。
workload_kind	String	工作负载类型。 <ul style="list-style-type: none"><li>• deployment, 无状态工作负载。</li><li>• statefulset, 有状态工作负载。</li><li>• daemonset, 守护进程集。</li></ul>
security_context	Object	安全策略, 请参考 <a href="#">表4-204</a> 。
logs	Array of objects	日志采集, 请参考 <a href="#">表4-206</a> 。
custom_metrics	Object	自定义指标监控, 请参考 <a href="#">表4-207</a> 。
affinity	Array of objects	亲和性, 请参考 <a href="#">表4-208</a> 。
anti_affinity	Array of objects	反亲和性, 请参考 <a href="#">表4-208</a> 。
liveness_probe	Array of objects	组件存活探针, 请参考 <a href="#">表4-210</a> 。



参数	参数类型	描述
readiness_probe	Object	组件业务探针, 请参考 <a href="#">表4-210</a> 。
refer_resources	Array of objects	关联资源, 请参考 <a href="#">表4-211</a> 。
status	Object	组件状态, 请参考 <a href="#">表4-219</a> 。

表 4-189 labels

参数	参数类型	描述
key	String	标签名称
value	String	标签值

表 4-190 storages

参数	参数类型	描述
type	String	数据存储类型。 <ul style="list-style-type: none"><li>• HostPath, 本地磁盘主机路径挂载。</li><li>• EmptyDir, 本地磁盘临时路径挂载。</li><li>• ConfigMap, 本地磁盘配置项挂载。</li><li>• Secret, 本地磁盘密钥挂载。</li><li>• PersistentVolumeClaim, 云存储挂载。</li></ul>
name	String	存储盘名称。
parameters	Object	对应各种数据存储类型的信息参数, 请参考 <a href="#">表 4-191</a> 。
mounts	Array of objects	数据存储挂载路径, 请参考 <a href="#">表4-192</a> 。

表 4-191 component\_storage\_parameters

参数	参数类型	描述
path	String	主机路径, 适用于HostPath的存储类型。
name	String	配置项、密钥或者PVC的名字, 适用于ConfigMap、Secret和PersistentVolumeClaim的存储类型。

参数	参数类型	描述
default_mode	Integer	挂载的权限，十进制格式，例如：384。适用于 ConfigMap 和 Secret 的存储类型。
medium	String	适用于 EmptyDir 类型的存储，例如：memory。

表 4-192 mounts

参数	参数类型	描述
path	String	挂载路径
sub_path	String	挂载路径的子路径
read_only	Boolean	是否只读

表 4-193 deploy\_strategy

参数	参数类型	描述
type	String	部署类型。 <ul style="list-style-type: none"> <li>OneBatchRelease，单批升级。</li> <li>RollingRelease，滚动部署升级。</li> <li>GrayRelease，灰度发布升级。</li> </ul>
rolling_release	Object	滚动部署参数，请参考表4-194。
gray_release	Object	灰度发布升级，请参考表4-195。

表 4-194 rolling\_release

参数	参数类型	描述
batches	Integer	分批部署批次。

表 4-195 gray\_release

参数	参数类型	描述
type	String	灰度策略。 <ul style="list-style-type: none"> <li>WEIGHT，按灰度流量进行升级。</li> <li>CONTENT，按内容进行灰度。</li> </ul>

参数	参数类型	描述
first_batch_weight	Integer	首批灰度流量比例，灰度策略为WEIGHT时有效。
first_batch_replica	Integer	首批灰度实例数量，灰度策略为WEIGHT时有效。
remaining_batch	Integer	剩余实例部署批次，首批灰度发布成功之后，剩余实例滚动升级分多少批次完成。 例如：剩余实例数是5，剩余部署批次是3，那么升级剩余实例会按照2:2:1个实例分批升级。 灰度策略为WEIGHT时有效。
deployment_mode	Integer	部署模型。 <ul style="list-style-type: none"><li>1，上游是微服务网关。</li><li>3，上游是微服务。</li><li>4，上游是ELB。</li></ul>
replica_surge_mode	String	灰度实例新增模式，灰度策略为CONTENT时有效。 <ul style="list-style-type: none"><li>MIRROR，蓝绿。</li><li>EXTRA，金丝雀（先增后减）。</li><li>NOSURGE，金丝雀（先减后增）。</li></ul>
rule_match_mode	String	灰度规则生效方式，灰度策略为CONTENT时有效。 <ul style="list-style-type: none"><li>ALL，满足所有条件。</li><li>ANY，满足任一条件。</li></ul>
rules	Object	灰度规则，灰度策略为CONTENT时有效，请参考 <a href="#">表 4-196</a> 。

表 4-196 gray\_rules

参数	参数类型	描述
type	String	匹配类型。 当前仅支持：header，请求头。
key	String	参数名称。
value	String	条件值。
condition	String	条件类型。 <ul style="list-style-type: none"><li>equal，相等。</li><li>match，匹配。</li><li>in，枚举。</li></ul>

表 4-197 command

参数	参数类型	描述
command	Array of String	运行命令，控制容器运行的输入命令。
args	Array of String	运行参数，控制容器运行命令的输入参数。例如：-port=8080，多条参数以换行分隔。

表 4-198 component\_lifecycle

参数	参数类型	描述
type	String	处理方式。 <ul style="list-style-type: none"><li>• http，http请求方式。</li><li>• command，命令行方式。</li></ul>
scheme	String	http请求类型参数：HTTP、HTTPS。type为http类型时生效。
host	String	默认为POD的IP，可以指定自定义的IP。type为http类型时生效。
port	Integer	端口号，type为http类型时生效。
path	String	请求路径，type为http类型时生效。
command	Array of String	命令列表，type为command类型时生效。

表 4-199 mesher

参数	参数类型	描述
port	Integer	进程监听端口。

表 4-200 tomcat\_opt

参数	参数类型	描述
server_xml	String	<ol style="list-style-type: none"><li>1. 监听端口以server.xml配置的为准，如果之前在公网访问配置了监听端口，请保持一致。</li><li>2. 自定义的Tomcat配置可能存在端口冲突，请检查自定义的端口是否被占用。</li></ol>

表 4-201 host\_aliases

参数	参数类型	描述
ip	String	IP地址
hostname	Array of String	主机别名

表 4-202 dns\_config

参数	参数类型	描述
nameservers	Array of String	Pod的DNS服务器的IP地址列表。
searches	Array of String	用于在Pod中查找主机名的DNS搜索域的列表。
options	Array of objects	可选的对象列表，其中每个对象可能具有name属性（必需）和value属性（可选）。此属性中的内容将合并到从指定的DNS策略生成的选项。重复的条目将被删除。请参考 <a href="#">表4-203</a> 。

表 4-203 options

参数	参数类型	描述
name	String	操作名称
value	String	操作值

表 4-204 security\_context

参数	参数类型	描述
run_as_user	Integer	运行用户，容器以当前用户权限运行，如以root权限运行则填写root用户ID 0。
run_as_group	Integer	运行属组，指定所有容器中的进程都以属组运行。
capabilities	Object	能力集，请参考 <a href="#">表4-205</a> 。

表 4-205 capabilities

参数	参数类型	描述
add	Array of String	为运行用户添加Linux权限。
drop	Array of String	移除运行用户Linux权限。

表 4-206 logs

参数	参数类型	描述
log_path	String	容器中日志路径。
rotate	String	日志转储周期。
host_path	String	挂载的主机路径。
host_extend_path	String	主机扩展路径，通过扩展主机路径，实现同一个主机路径下区分来自不同容器的挂载： <ul style="list-style-type: none"><li>• None，不使用扩展路径。</li><li>• PodUID，通过POD的ID扩展主机路径。</li><li>• PodName，通过POD的名称扩展主机路径。</li><li>• PodUID/ContainerName，通过POD的ID和容器名称扩展主机路径。</li><li>• PodName/ContainerName，通过POD的名称和容器名称扩展主机路径。</li></ul>

表 4-207 custom\_metric

参数	参数类型	描述
path	String	采集路径，例如：/metrics。
port	Integer	采集端口，例如：9090。
dimensions	String	监控维度，例如："cpu_usage,mem_usage"。

表 4-208 component\_affinity

参数	参数类型	描述
condition	String	是否必须满足。 <ul style="list-style-type: none"><li>• required，必须满足。</li><li>• preferred，尽量满足。</li></ul>

参数	参数类型	描述
kind	String	亲和类型。 <ul style="list-style-type: none"><li>• node, 节点亲和。</li><li>• pod, pod亲和。</li></ul>
weight	Integer	亲和权重, 取值范围0~100。
match_expressions	String	匹配条件, 请参考 <a href="#">表4-209</a> 。

表 4-209 match\_expressions

参数	参数类型	描述
key	String	匹配标签key值。
operation	String	匹配条件。
value	String	匹配标签value值。

表 4-210 component\_probe

参数	参数类型	描述
type	String	类型: http、tcp、command。
delay	Integer	启动后多久开始探测。
timeout	Integer	探测超时时间
scheme	String	请求类型: HTTP、HTTPS, type为http类型时生效。
host	String	默认为POD的IP, 可以指定自定义的IP。type为http类型时生效。
port	Integer	端口号, type为http和tcp类型时生效。
path	String	请求路径, type为http类型时生效。
command	Array of String	命令列表, type为command类型时生效。

表 4-211 refer\_resources

参数	参数类型	描述
id	String	资源id

参数	参数类型	描述
type	String	资源类型。 <ul style="list-style-type: none"> <li>• vpc: 虚拟私有云</li> <li>• eip: 弹性公网IP</li> <li>• elb: 弹性负载均衡</li> <li>• cce: 云容器引擎</li> <li>• ecs: 弹性云服务器</li> <li>• as: 弹性伸缩组</li> <li>• cse: ServiceComb引擎</li> <li>• nacos: 注册配置中心</li> <li>• dcs: 分布式缓存服务</li> <li>• rds: 云数据库</li> </ul>
parameters	Object	资源参数, 请参考 <a href="#">表4-212</a> 。

表 4-212 refer\_resource\_parameter

参数	参数类型	描述
namespace	String	命名空间

表 4-213 env

参数	参数类型	描述
name	String	变量名。
value	String	变量值。
value_from	Object	变量引用, 请参考 <a href="#">表4-214</a> 。

表 4-214 value\_from

参数	参数类型	描述
reference_type	String	引用类型。 <ul style="list-style-type: none"> <li>• configMapKey, 配置项导入。</li> <li>• secretKey, 密钥导入。</li> </ul>
name	String	配置项或密钥的名称。
key	String	配置项或密钥的键值。
optional	Boolean	配置项或密钥的key是否必须存在。



表 4-215 runtime\_stack

参数	参数类型	描述
name	String	技术栈名称
type	String	技术栈类型，支持Java、Tomcat、Nodejs、Php、Docker、Python。 当部署模式为虚拟机部署时，仅支持Java、Tomcat、Nodejs；容器部署上述类型都支持。
version	String	技术栈版本。
deploy_mode	String	部署模式。 <ul style="list-style-type: none"><li>• container，容器部署。</li><li>• virtualmachine，虚拟机部署。</li></ul>

表 4-216 source

参数	参数类型	描述
kind	String	组件来源类型。 <ul style="list-style-type: none"><li>• code，源码。</li><li>• package，软件包。</li><li>• image，镜像包。</li></ul>
version	String	版本号。
url	String	包地址，类型为package和image时需要添加。
storage	String	存储方式，支持镜像仓库swr、软件仓库swr、软件开发生产线codearts和对象存储obs。类型为package和image时需要添加。
auth	String	认证方式，支持iam、none，默认是iam。
repo_auth	String	授权名称，在授权列表获取。
repo_namespace	String	代码仓库命名空间。
repo_ref	String	代码分支或者Tag，默认是master。
repo_type	String	代码仓类型，支持CodeArts、Gitee、GitHub、BitBucket、GitLab。
web_url	String	代码跳转链接，例如： <a href="https://github.com/example/demo.git">https://github.com/example/demo.git</a> 。
repo_url	String	代码仓url，例如： <a href="https://github.com/example/demo.git">https://github.com/example/demo.git</a> 。

参数	参数类型	描述
commit_id	String	代码仓库中，commit id是每次提交的唯一标识符，用于识别和引用特定的提交。 长度为8到40个字符，以小写字母或数字开头。 <b>请求消息</b> 包含 <b>表4-186</b> 时才返回该字段。

表 4-217 build

参数	参数类型	描述
parameters	Map<String, Object>	只有在没有ID，新创建构建时提供。请参考 <b>表 4-218</b> 。

表 4-218 parameters

参数	参数类型	描述
build_cmd	String	编译命令。默认： 1. 根目录存在build.sh: ./build.sh 2. 根据运行系统，示例如下： <ul style="list-style-type: none"> <li>• Java和Tomcat: <b>mvn clean package</b></li> <li>• Nodejs: <b>npm build</b></li> </ul>
dockerfile_path	String	dockerfile地址，默认是根目录./。
artifact_name_space	String	构建归档组织，默认cas_{project_id}。
cluster_id	String	指定构建集群的id。
node_label_selector	Map<String, String>	key是标签的键，value是标签的值。
environment_id	String	环境ID。

表 4-219 status

参数	参数类型	描述
component_status	String	组件状态。 <ul style="list-style-type: none"><li>INITIALIZING, 初始化。</li><li>UPGRADING, 升级中。</li><li>FAILED, 失败。</li><li>RUNNING, 运行中。</li><li>DOWN, 下线。</li><li>DELETING, 删除中。</li><li>DELETED, 已删除。</li><li>RESERVED, 保留。</li><li>STARTING, 启动中。</li><li>STOPPING, 停止中。</li><li>STOPPED, 停止。</li><li>RESTARTING, 重启中。</li><li>PENDING, 未就绪</li><li>UNKNOWN, 未知。</li><li>PARTIALLY_FAILED, 部分失败。</li></ul>
available_replica	Integer	可用实例。
replica	Integer	总实例。
fail_detail	String	失败原因。 <ul style="list-style-type: none"><li>cluster_deleted, 集群被删除。</li><li>cluster_unavailable, 集群不可用。</li><li>cluster_inaccessible, 集群无法访问。</li><li>namespace_deleted, 命名空间被删除。</li><li>namespace_unavailable, 命名空间不可用。</li><li>namespace_inaccessible, 命名空间无法访问。</li><li>resource_deleted, 资源已删除。</li><li>create_failed, 创建组件失败。</li><li>delete_failed, 删除组件失败。</li></ul>
last_job_id	String	最后执行任务ID。
create_time	Integer	创建时间。
update_time	Integer	更新时间。
creator	String	创建人。

参数	参数类型	描述
artifact	Object	组件构建信息，请参考 <a href="#">表4-220</a> 。

表 4-220 artifact

参数	参数类型	描述
type	String	组件来源类型。 <ul style="list-style-type: none"><li>虚拟机部署支持软件包package。</li><li>容器部署支持软件包package、镜像包image。</li></ul>
url	String	软件包/镜像地址。

## 请求示例

无

## 响应示例

```
{
  "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"
}
```

## 状态码

状态码	描述
200	操作成功
400	错误的请求
404	请求对象不存在
500	内部错误

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.3.8 根据组件 ID 下发组件任务

### 功能介绍

此API用于通过组件ID下发组件任务。

### 调试

您可以在[API Explorer](#)中调试该接口。

### URI

POST /v3/{project\_id}/cas/applications/{application\_id}/components/  
{component\_id}/action

表 4-221 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
application_id	是	String	应用ID。获取方法，请参考 <a href="#">获取所有应用</a> 。

参数	是否必选	参数类型	描述
component_id	是	String	组件ID。获取方法，请参考 <a href="#">获取应用所有组件</a> 。

## 请求消息

表 4-222 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

表 4-223 请求 Body 参数

参数	是否必选	参数类型	描述
action	是	String	组件任务类型。 <ul style="list-style-type: none"><li>• start, 启动。</li><li>• stop, 停止。</li><li>• restart, 重启。</li><li>• scale, 伸缩。</li><li>• rollback, 回滚。</li><li>• rollback_current, 回滚到上一个版本。</li><li>• continue_deploy, 继续部署。</li><li>• check_gray_release, 检查灰度规则。</li><li>• modify_gray_rule, 修改灰度规则。</li><li>• sync_workload, 同步工作负载。</li></ul>
parameters	否	Object	任务参数，请参考 <a href="#">表4-224</a> 。

表 4-224 parameters

参数	是否必选	参数类型	描述
replica	否	Integer	组件实例数量。
version	否	String	版本号。
hosts	否	Array of String	hostID列表，当虚拟机部署组件伸缩时必选。

## 响应消息

表 4-225 响应参数

参数	参数类型	描述
job_id	String	任务ID。
result	String	任务状态。

## 请求示例

启动组件。

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

## 响应示例

```
{
  "job_id": "string",
  "result": "succeeded"
}
```

## 状态码

状态码	描述
200	操作成功
400	错误的请求
404	请求对象不存在

状态码	描述
500	内部错误

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.3.9 根据组件 ID 获取记录

### 功能介绍

此API用于通过组件ID获取记录。

### 调试

您可以在[API Explorer](#)中调试该接口。

### URI

GET /v3/{project\_id}/cas/applications/{application\_id}/components/{component\_id}/records

表 4-226 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
application_id	是	String	应用ID。获取方法，请参考 <a href="#">获取所有应用</a> 。
component_id	是	String	组件ID。获取方法，请参考 <a href="#">获取应用所有组件</a> 。

表 4-227 查询参数

参数	是否必选	参数类型	描述
limit	否	String	指定个数，取值[0, 100]。
offset	否	String	指定偏移量。
order_by	否	String	排序字段，默认按创建时间排序。 排序字段支持枚举值：create_time、name、update_time。



参数	是否必选	参数类型	描述
order	否	String	排序方式。 <ul style="list-style-type: none"><li>• desc, 降序排序, 默认为desc。</li><li>• asc, 升序排序。</li></ul>

## 请求消息

表 4-228 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

## 响应消息

表 4-229 响应参数

参数	参数类型	描述
count	Integer	记录总数。
records	Array of objects	记录列表，请参考 <a href="#">表4-230</a> 。

表 4-230 records 参数

参数	参数类型	描述
begin_time	String	开始时间。
end_time	String	结束时间。
description	Object	描述。
instance_id	String	实例ID。
version	String	版本号。
current_used	Boolean	当前使用。

参数	参数类型	描述
status	String	状态。
deploy_type	String	部署类型。 <ul style="list-style-type: none"> <li>• OneBatchRelease, 单批部署。</li> <li>• RollingRelease, 滚动部署。</li> <li>• GrayRelease, 灰度发布。</li> </ul>
jobs	Array of objects	任务列表, 请参考 <a href="#">表4-231</a> 。

表 4-231 jobs

参数	参数类型	描述
sequence	Integer	执行顺序。
deploy_type	String	部署类型。 <ul style="list-style-type: none"> <li>• OneBatchRelease, 单批部署。</li> <li>• RollingRelease, 滚动部署。</li> <li>• GrayRelease, 灰度发布。</li> </ul>
job_id	String	任务ID。
job_info	Object	任务信息, 请参考 <a href="#">表4-232</a> 。

表 4-232 job\_info

参数	参数类型	描述
deploy_type	String	部署类型
source_url	String	组件来源
first_batch_weight	Integer	首次执行权重
first_batch_replica	Integer	首次部署组件实例
replica	Integer	总实例
remaining_batch	Integer	剩余批次

## 请求示例

无

## 响应示例

```
{
  "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
          }
        }
      ]
    }
  ]
}
```

## 状态码

状态码	描述
200	操作成功
400	错误的请求
404	请求对象不存在
500	内部错误

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

### 4.3.10 根据组件 ID 刷新组件信息

#### 功能介绍

此API用于通过组件ID刷新组件状态和访问方式信息。

#### URI

PUT /v3/{project\_id}/cas/applications/{application\_id}/components/  
{component\_id}/refresh

表 4-233 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
application_id	是	String	应用ID。获取方法，请参考 <a href="#">获取所有应用</a> 。
component_id	是	String	组件ID。获取方法，请参考 <a href="#">获取应用所有组件</a> 。

## 请求消息

表 4-234 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

## 响应消息

表 4-235 响应参数

参数	参数类型	描述
message	String	表示刷新成功。 固定值：OK。

## 请求示例

无

## 响应示例

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

## 状态码

状态码	描述
200	操作成功
400	错误的请求
404	请求对象不存在
500	内部错误

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

### 4.3.11 获取组件相关配置信息

#### 功能介绍

此API用于获取所有所有组件所属的应用信息、环境信息、企业项目信息。

#### URI

GET /v3/{project\_id}/cas/components/filterOptions

表 4-236 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。

#### 请求消息

表 4-237 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

## 响应消息

表 4-238 响应参数

参数	参数类型	描述
filter_options	Array of objects	组件相关信息参数，请参考 <a href="#">表4-239</a> 。

表 4-239 组件相关参数

参数	参数类型	描述
applications	Array of objects	应用信息，请参考 <a href="#">表4-240</a> 。
enterprise_projects	Array of String	企业项目信息，组件使用的企业项目ID列表。
environments	Array of objects	环境信息，请参考 <a href="#">表4-241</a> 。

表 4-240 应用信息

参数	参数类型	描述
id	String	应用ID。
label	String	应用名称。

表 4-241 环境信息

参数	参数类型	描述
id	String	环境ID。
label	String	环境名称。

## 请求示例

无

## 响应示例

```
{
  "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"  
  ]  
}
```

## 状态码

状态码	描述
200	操作成功
400	错误的请求
404	请求对象不存在
500	内部错误

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.4 技术栈

### 4.4.1 查询运行时栈

#### 功能介绍

此API用于查询运行时栈信息。

#### 调试

您可以在[API Explorer](#)中调试该接口。

#### URI

GET /v3/{project\_id}/cas/runtimestacks

表 4-242 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。

## 请求消息

表 4-243 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">认证鉴权</a> 。

## 响应消息

表 4-244 响应参数

参数	参数类型	描述
runtime_stacks	Array of objects	技术栈，请参考 <a href="#">表4-245</a> 。
total	Integer	技术栈总数。

表 4-245 runtimestack

参数	参数类型	描述
id	String	技术栈的ID。
name	String	运行时名称。
project_id	String	租户项目ID。
deploy_mode	String	部署类型。 <ul style="list-style-type: none"><li>• container，容器部署。</li><li>• virtualmachine，虚拟机部署。</li></ul>
type	String	技术栈类型：Nodejs、Java、Tomcat、Python、Docker、Php。



参数	参数类型	描述
version	String	技术栈版本。
spec	Object	技术栈相关参数，请参考表4-246。
system	Integer	是否系统技术栈。 <ul style="list-style-type: none"> <li>• 1: 系统技术栈。</li> <li>• 0: 自定义技术栈。</li> </ul>
status	String	技术栈状态。 <ul style="list-style-type: none"> <li>• Supported, 启用。</li> <li>• Deprecated, 已弃用。</li> <li>• Disable, 未启用</li> </ul>
create_time	Integer	创建时间。
update_time	Integer	修改时间。
component_count	Integer	技术栈关联的组件个数。

表 4-246 spec

参数	参数类型	描述
os	String	技术栈操作系统。
sdk	String	技术栈工具包。
parameters	Map<String,String>	技术栈相关镜像地址。

## 请求示例

无

## 响应示例

```
{
  "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
}
```

## 状态码

状态码	描述
200	操作成功
400	错误的请求
404	请求对象不存在
409	请求与当前对象状态冲突
500	内部错误

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.4.2 查询内置镜像

### 功能介绍

此API用于获取ServiceStage各个技术栈构建时使用的内置基础镜像。

### URI

GET /v3/{project\_id}/cas/innerimages

表 4-247 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。

## 请求消息

表 4-248 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

## 响应消息

表 4-249 响应参数

参数	参数类型	描述
runtime_stacks	Array of objects	运行时栈，请参考 <a href="#">表4-250</a> 。

表 4-250 runtime\_stacks

参数	参数类型	描述
type	String	技术栈类型。 <ul style="list-style-type: none"> <li>• Nodejs</li> <li>• Java</li> <li>• Tomcat</li> <li>• Python</li> <li>• Php</li> </ul>
url	String	镜像地址。

## 请求示例

无

## 响应示例

```
{
  "runtimestacks": [
    {
      "url": "swr.ap-southeast-1.myhuaweicloud.com/op_svc_cse/openjdk-x86_64:17-1.3.6",
      "type": "Java"
    }
  ]
}
```

## 状态码

状态码	描述
200	操作成功
400	错误的请求
404	请求对象不存在
409	请求与当前对象状态冲突
500	内部错误

## 错误码

错误码格式为: SVCSTG.00100.[Error\_ID], 例如: SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.4.3 创建技术栈

### 功能介绍

此API用于创建自定义技术栈。

### URI

POST /v3/{project\_id}/cas/runtimestacks

表 4-251 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法, 请参考 <a href="#">获取项目ID</a> 。

## 请求消息

表 4-252 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">认证鉴权</a> 。

表 4-253 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	技术栈名称。
deploy_mode	是	String	部署类型。 目前仅支持virtualmachine（虚拟机部署）类型。
type	是	String	技术栈类型。 目前仅支持Java、Tomcat。
version	是	String	版本号。
spec	否	Object	技术栈相关参数，请参考 <a href="#">表4-254</a> 。
description	否	String	技术栈描述。

表 4-254 spec

参数	是否必选	参数类型	描述
sdk	否	String	技术栈工具包。
os	否	String	技术栈操作系统。
parameters	否	Map<String, String>	技术栈相关镜像参数。

## 响应消息

表 4-255 响应参数

参数	参数类型	描述
id	String	技术栈的ID。
name	String	运行时名称。
project_id	String	租户项目ID。
deploy_mode	String	部署类型。 <ul style="list-style-type: none"><li>• container, 容器部署。</li><li>• virtualmachine, 虚拟机部署。</li></ul>
version	String	版本号。
type	String	技术栈类型: Nodejs、Java、Tomcat、Python、Docker、Php。
version	String	技术栈版本。
spec	Object	技术栈相关参数, 请参考 <a href="#">表4-256</a> 。
system	Integer	是否系统技术栈。 <ul style="list-style-type: none"><li>• 1: 系统技术栈。</li><li>• 0: 自定义技术栈。</li></ul>
status	String	技术栈状态。 <ul style="list-style-type: none"><li>• Supported, 启用。</li><li>• Deprecated, 已弃用。</li><li>• Disable, 未启用</li></ul>
creator	String	创建人。
create_time	Integer	创建时间。
update_time	Integer	修改时间。
component_count	Integer	技术栈关联的组件个数。

表 4-256 spec

参数	参数类型	描述
os	String	技术栈操作系统。
sdk	String	技术栈工具包。
parameters	Map<String,String>	技术栈相关镜像地址。

## 请求示例

```
{
  "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"
    }
  }
}
```

## 响应示例

```
{
  "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
}
```

## 状态码

状态码	描述
200	操作成功
400	错误的请求
404	请求对象不存在
409	请求与当前对象状态冲突
500	内部错误

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.4.4 根据技术栈 ID 查询技术栈

### 功能介绍

此API用于根据技术栈ID查询技术栈。

### URI

GET /v3/{project\_id}/cas/runtimestacks/{runtimestack\_id}

表 4-257 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
runtimestack_id	是	String	技术栈ID。获取方法，请参考 <a href="#">查询运行时栈</a> 。

### 请求消息

表 4-258 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">认证鉴权</a> 。

### 响应消息

表 4-259 响应参数

参数	参数类型	描述
id	String	技术栈的ID。
name	String	运行时名称。
project_id	String	租户项目ID。
deploy_mode	String	部署类型。 <ul style="list-style-type: none"><li>container，容器部署。</li><li>virtualmachine，虚拟机部署。</li></ul>



参数	参数类型	描述
version	String	版本号。
type	String	技术栈类型：Nodejs、Java、Tomcat、Python、Docker、Php。
version	String	技术栈版本。
spec	Object	技术栈相关参数，请参考表4-260。
system	Integer	是否是系统技术栈。 <ul style="list-style-type: none"> <li>• 1：系统技术栈。</li> <li>• 0：自定义技术栈。</li> </ul>
status	String	技术栈状态。 <ul style="list-style-type: none"> <li>• Supported，启用。</li> <li>• Deprecated，已弃用。</li> <li>• Disable，未启用</li> </ul>
creator	String	创建人。
create_time	Integer	创建时间。
update_time	Integer	修改时间。
component_count	Integer	技术栈关联的组件个数。

表 4-260 spec

参数	参数类型	描述
os	String	技术栈操作系统。
sdk	String	技术栈工具包。
parameters	Map<String,String>	技术栈相关镜像地址。

## 请求示例

无

## 响应示例

```
{
  "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
}
```

## 状态码

状态码	描述
200	操作成功
400	错误的请求
404	请求对象不存在
409	请求与当前对象状态冲突
500	内部错误

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.4.5 根据技术栈 ID 修改技术栈

### 功能介绍

此API用于根据技术栈ID修改技术栈。

### URI

PUT /v3/{project\_id}/cas/runtimestacks/{runtimestack\_id}

表 4-261 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
runtimestack_id	是	String	技术栈ID。获取方法，请参考 <a href="#">查询运行时栈</a> 。

## 请求消息

表 4-262 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">认证鉴权</a> 。

表 4-263 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	技术栈名称。
deploy_mode	是	String	部署类型。 目前仅支持virtualmachine（虚拟机部署）类型。
type	是	String	技术栈类型。 目前仅支持Java、Tomcat。
version	是	String	版本号。
spec	否	Object	技术栈相关参数，请参考 <a href="#">表4-264</a> 。
description	否	String	技术栈描述。

表 4-264 spec

参数	是否必选	参数类型	描述
sdk	否	String	技术栈工具包。
os	否	String	技术栈操作系统。
parameters	否	Map<String, String>	技术栈相关镜像参数。

## 响应消息

表 4-265 响应参数

参数	参数类型	描述
id	String	技术栈的ID。
name	String	运行时名称。
project_id	String	租户项目ID。
deploy_mode	String	部署类型。 <ul style="list-style-type: none"><li>• container, 容器部署。</li><li>• virtualmachine, 虚拟机部署。</li></ul>
version	String	版本号。
type	String	技术栈类型: Nodejs、Java、Tomcat、Python、Docker、Php。
version	String	技术栈版本。
spec	Object	技术栈相关参数, 请参考 <a href="#">表4-266</a> 。
system	Integer	是否是系统技术栈。 <ul style="list-style-type: none"><li>• 1: 系统技术栈。</li><li>• 0: 自定义技术栈。</li></ul>
status	String	技术栈状态。 <ul style="list-style-type: none"><li>• Supported, 启用。</li><li>• Deprecated, 已弃用。</li><li>• Disable, 未启用</li></ul>
creator	String	创建人。
create_time	Integer	创建时间。
update_time	Integer	修改时间。
component_count	Integer	技术栈关联的组件个数。

表 4-266 spec

参数	参数类型	描述
os	String	技术栈操作系统。
sdk	String	技术栈工具包。
parameters	Map<String,String>	技术栈相关镜像地址。

## 请求示例

```
{
  "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"
    }
  }
}
```

## 响应示例

```
{
  "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
}
```

## 状态码

状态码	描述
200	操作成功
400	错误的请求
404	请求对象不存在
409	请求与当前对象状态冲突
500	内部错误

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.4.6 根据技术栈 ID 删除技术栈

### 功能介绍

从API用于根据技术栈ID删除技术栈。

### URI

DELETE /v3/{project\_id}/cas/runtimestacks/{runtimestack\_id}

表 4-267 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
runtimestack_id	是	String	技术栈ID。获取方法，请参考 <a href="#">查询运行时栈</a> 。

### 请求消息

表 4-268 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">认证鉴权</a> 。

### 响应消息

无

### 请求示例

无

### 响应示例

无

## 状态码

状态码	描述
204	操作成功
400	错误的请求
404	请求对象不存在
409	请求与当前对象状态冲突
500	内部错误

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.4.7 发布和取消发布技术栈

### 功能介绍

此API用于发布和取消发布技术栈。

### URI

POST /v3/{project\_id}/cas/runtimestacks

表 4-269 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。

### 请求消息

表 4-270 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">认证鉴权</a> 。

表 4-271 请求 Body 参数

参数	是否必选	参数类型	描述
action	是	String	对技术栈进行的操作。 <ul style="list-style-type: none"> <li>Supported, 发布技术栈。</li> <li>Disable, 取消发布技术栈。</li> </ul>
parameters	是	Map<String, Object>	请参考 <a href="#">表4-272</a> 。

表 4-272 parameters

参数	是否必选	参数类型	描述
runtimestacks	是	Array of objects	对技术栈进行发布/取消发布操作。 参数格式请参考 <a href="#">表4-273</a> 。

表 4-273 runtimestack\_item

参数	是否必选	参数类型	描述
id	是	String	技术栈ID。

## 响应消息

无

## 请求示例

```
{
  "action": "Supported",
  "parameters": {
    "runtimestacks": [
      {
        "id": "b153f6b8-9335-46a1-913e-c2d8f966d4b2"
      }
    ]
  }
}
```

## 响应示例

无



## 状态码

状态码	描述
204	操作成功
400	错误的请求
404	请求对象不存在
409	请求与当前对象状态冲突
500	内部错误

## 错误码

错误码格式为: SVCSTG.00100.[Error\_ID], 例如: SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.5 配置管理

### 4.5.1 创建配置分组

#### 功能介绍

此API用于创建配置分组，用于对配置文件分类管理。

#### URI

POST /v3/{project\_id}/cas/config-groups

表 4-274 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。

## 请求消息

表 4-275 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">认证鉴权</a> 。

表 4-276 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	配置分组名称。 由大小写英文字母、数字、中划线（-）、下划线（_）组成，并以大小写英文字母开头，大小写英文字母或者数字结尾，长度为2~64个字符。
description	否	String	配置分组描述。 最大长度为256个字符。

## 响应消息

表 4-277 响应参数

参数	参数类型	描述
id	String	配置分组ID。
name	String	配置分组名称。
description	String	配置分组描述。
creator	String	创建人。
project_id	String	项目ID。
create_time	Integer	创建时间。

## 请求示例

创建名为test-group的配置分组。

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

## 响应示例

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

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在。
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.5.2 获取配置分组

### 功能介绍

此API用于获取已经创建的配置分组。

### URI

GET /v3/{project\_id}/cas/config-groups

表 4-278 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。

表 4-279 查询参数

参数	是否必选	参数类型	描述
name	否	String	配置分组名称。
limit	否	integer	指定个数，取值[0, 100]。
offset	否	integer	指定偏移量。
order_by	否	String	排序字段，默认按创建时间排序。 排序字段支持枚举值： <ul style="list-style-type: none"><li>• create_time，配置分组创建时间。</li><li>• name，配置分组名称。</li><li>• update_time，配置分组修改时间。</li></ul>
order	否	String	排序方式。 <ul style="list-style-type: none"><li>• desc，降序排序，默认排序方式。</li><li>• asc，升序排序。</li></ul>

## 请求消息

表 4-280 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">认证鉴权</a> 。

## 响应消息

表 4-281 响应参数

参数	参数类型	描述
count	Integer	配置分组总数。
config_groups	Array of objects	分组列表，请参考 <a href="#">表4-282</a> 。

表 4-282 config\_groups

参数	参数类型	描述
id	String	配置分组ID。
project_id	String	项目ID。
name	String	配置分组名称。
description	String	配置分组描述。
creator	String	创建人。
create_time	Integer	创建时间。

### 请求示例

无

### 响应示例

```
{
  "config_groups": [
    {
      "id": "0f5722ea-ee97-42bf-8583-404caa0a46e5",
      "project_id": "2cfeefc3e4c54a5aa7548b8350e638d7",
      "name": "test1",
      "description": "",
      "create_time": 1717482692680,
      "creator": "ss-test"
    },
    {
      "id": "49cb9636-9f89-44c3-af68-95282887b377",
      "project_id": "2cfeefc3e4c54a5aa7548b8350e638d7",
      "name": "test2",
      "description": "",
      "create_time": 1717481328004,
      "creator": "ss-test"
    }
  ],
  "count": 2
}
```

### 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在。
500	内部错误。

### 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.5.3 创建配置文件

### 功能介绍

此API用于创建配置文件。

### URI

POST /v3/{project\_id}/cas/configs

表 4-283 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。

### 请求消息

表 4-284 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">认证鉴权</a> 。

表 4-285 请求 Body 参数

参数	是否必选	参数类型	描述
config_group_id	是	String	配置分组ID。
name	是	String	配置文件名称。 为长度2到64个字符的字符串，可以包含英文字母、数字、下划线（_）或中划线（-），以英文字母开头、英文字母或者数字结尾。
content	是	String	配置文件内容。

参数	是否必选	参数类型	描述
description	否	String	配置文件描述，为不超过128个字符的字符串。
type	是	String	配置文件格式： <ul style="list-style-type: none"> <li>• yaml</li> <li>• properties</li> </ul>
sensitive	是	Boolean	配置文件内容是否加密： <ul style="list-style-type: none"> <li>• true，加密。</li> <li>• false，不加密。</li> </ul>

## 响应消息

表 4-286 响应参数

参数	参数类型	描述
id	String	配置文件ID。
name	String	配置文件名称。
description	String	配置描述。
creator	String	创建人。
project_id	String	项目ID。
type	String	配置文件格式： <ul style="list-style-type: none"> <li>• yaml</li> <li>• properties</li> </ul>
sensitive	Boolean	配置文件内容是否加密： <ul style="list-style-type: none"> <li>• true，加密。</li> <li>• false，不加密。</li> </ul>
components	Array of objects	配置文件绑定的组件，请参考 <a href="#">表4-287</a> 。
create_time	Integer	创建时间。
update_time	Integer	修改时间。

表 4-287 components

参数	参数类型	描述
application_id	String	应用ID。
component_id	String	组件ID。
component_name	String	组件名称。
environment_id	String	环境ID。

## 请求示例

创建名为test-config的配置文件，不加密，文件格式为properties。

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

## 响应示例

```
{
  "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"
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在。
500	内部错误。



## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.5.4 导入配置文件

### 功能介绍

此API用于通过导入zip压缩包创建配置文件。

### URI

POST /v3/{project\_id}/cas/configs/import

表 4-288 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。

### 请求消息

表 4-289 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），设置为：multipart/form-data
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">认证鉴权</a> 。

表 4-290 请求 Body 参数

参数	是否必选	参数类型	描述
upload_file	是	File	导入的配置文件需要压缩成zip格式，且压缩包文件大小不能超过500KB。 导入的压缩包中的配置文件数不能超过500个。 压缩包中的文件夹名称为配置分组名称，文件名称为配置名称，文件类型支持yaml、properties格式。
repetition_policy	否	String	导入配置文件中同名配置文件处理策略： <ul style="list-style-type: none"><li>covered: 覆盖，用导入的配置文件替换分组下已有同名配置文件。</li><li>skip: 跳过，跳过导入同名配置文件。</li></ul>

## 响应消息

表 4-291 响应参数

参数	参数类型	描述
covered_num	Integer	导入配置文件中被覆盖的配置文件数量。
skip_num	Integer	导入配置文件中跳过的配置文件数量。
total_num	Integer	导入配置文件的数量。

## 请求示例

```
POST https://{endpoint}/v3/{project_id}/cas/configs/import
{repetition_policy=covered }
```

## 响应示例

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

## 状态码

状态码	描述
204	操作成功。
400	错误的请求。
404	请求对象不存在。

状态码	描述
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.5.5 获取配置文件信息

### 功能介绍

此API用于获取配置文件信息。

### URI

GET /v3/{project\_id}/cas/configs

表 4-292 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。

表 4-293 查询参数

参数	是否必选	参数类型	描述
name	否	String	配置文件名称。
config_group_id	否	String	配置分组ID。
component_id	否	String	组件ID。
limit	否	integer	指定个数，取值[0, 100]。
offset	否	integer	指定偏移量。
order_by	否	String	排序字段，默认按创建时间排序。 <ul style="list-style-type: none"><li>• create_time，配置文件创建时间。</li><li>• name，配置文件名称。</li><li>• update_time，配置文件修改时间。</li></ul>

参数	是否必选	参数类型	描述
order	否	String	排序方式。 <ul style="list-style-type: none"> <li>• desc, 降序排序, 默认排序方式。</li> <li>• asc, 升序排序。</li> </ul>

## 请求消息

表 4-294 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型 (格式), 默认取值为 “application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种, 如果您使用的Token方式, 此参数为必填, 请填写Token的值。Token获取方式, 请参考 <a href="#">ServiceStage错误码</a> 。

## 响应消息

表 4-295 响应参数

参数	参数类型	描述
count	Integer	配置文件总数。
configs	Array of objects	配置文件列表, 请参考 <a href="#">表4-296</a> 。

表 4-296 configs

参数	参数类型	描述
config_group_id	String	配置分组ID。
id	String	配置文件ID。
name	String	配置文件名称。
description	String	配置文件描述。
creator	String	创建人。

参数	参数类型	描述
type	String	配置文件类型： <ul style="list-style-type: none"><li>• yaml</li><li>• properties</li></ul>
content	String	配置文件内容。
version	String	配置文件版本号。
project_id	String	项目ID。
sensitive	Boolean	配置文件内容是否加密： <ul style="list-style-type: none"><li>• true, 加密。</li><li>• false, 不加密。</li></ul>
components	Array of objects	配置文件绑定的组件列表，请参考 <a href="#">表4-297</a> 。
create_time	Integer	创建时间。
update_time	Integer	修改时间。

表 4-297 components

参数	参数类型	描述
application_id	String	应用ID。
component_id	String	组件ID。
component_name	String	组件名称。
environment_id	String	环境ID。

## 请求示例

无

## 响应示例

```
{
  "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
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在。
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.5.6 根据配置文件 ID 获取配置文件信息

### 功能介绍

此API用于通过配置文件ID获取配置文件的详情。

### URI

GET /v3/{project\_id}/cas/configs/{config\_id}

表 4-298 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
config_id	是	String	配置文件ID。获取方法，请参考 <a href="#">获取配置文件信息</a> 。

## 请求消息

表 4-299 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">认证鉴权</a> 。

## 响应消息

表 4-300 响应参数

参数	参数类型	描述
config_group_id	String	配置分组ID。
id	String	配置文件ID。
name	String	配置文件名称。
description	String	配置文件描述。
creator	String	创建人。
type	String	配置文件类型： <ul style="list-style-type: none"><li>• yaml</li><li>• properties</li></ul>
content	String	配置文件内容。
version	String	配置文件版本号。

参数	参数类型	描述
project_id	String	项目ID。
sensitive	Boolean	配置文件内容是否加密： <ul style="list-style-type: none"><li>• true, 加密。</li><li>• false, 不加密。</li></ul>
components	Array of objects	配置文件绑定的组件列表，请参考 <a href="#">表4-301</a> 。
create_time	Integer	创建时间。
update_time	Integer	修改时间。

表 4-301 components

参数	参数类型	描述
application_id	String	应用ID。
component_id	String	组件ID。
component_name	String	组件名称。
environment_id	String	环境ID。

## 请求示例

无

## 响应示例

```
{
  "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"
}
```



## 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在。
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.5.7 根据配置文件 ID 删除配置文件

### 功能介绍

此API用于通过配置文件ID删除配置文件。

### URI

DELETE /v3/{project\_id}/cas/configs/{config\_id}

表 4-302 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
config_id	是	String	配置文件ID。获取方法，请参考 <a href="#">获取配置文件信息</a> 。

### 请求消息

表 4-303 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">认证鉴权</a> 。

## 响应消息

无

## 请求示例

无

## 响应示例

无

## 状态码

状态码	描述
204	操作成功。
400	错误的请求。
404	请求对象不存在。
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.5.8 根据配置文件 ID 修改配置文件

### 功能介绍

此API用于通过配置文件ID修改配置文件。

### URI

PUT /v3/{project\_id}/cas/configs/{config\_id}

表 4-304 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
config_id	是	String	配置文件ID。获取方法，请参考 <a href="#">获取配置文件信息</a> 。

## 请求消息

表 4-305 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">认证鉴权</a> 。

表 4-306 请求 Body 参数

参数	是否必选	参数类型	描述
config_group_id	是	String	配置分组ID。
content	是	String	配置文件内容。
type	是	String	配置文件格式。
description	否	String	配置文件描述。
sensitive	否	Boolean	配置文件内容是否加密： <ul style="list-style-type: none"><li>• true，加密。</li><li>• false，不加密。</li></ul>

## 响应消息

表 4-307 响应参数

参数	参数类型	描述
config_group_id	String	配置分组ID。
id	String	配置文件ID。
name	String	配置文件名称。
description	String	配置文件描述。
creator	String	创建人。
type	String	配置文件类型： <ul style="list-style-type: none"><li>• yaml</li><li>• properties</li></ul>
content	String	配置文件内容。
version	String	配置文件版本号。
project_id	String	项目ID。
sensitive	Boolean	配置文件内容是否加密： <ul style="list-style-type: none"><li>• true, 加密。</li><li>• false, 不加密。</li></ul>
components	Array of objects	配置文件绑定的组件列表，请参考 <a href="#">表4-308</a> 。
create_time	Integer	创建时间。
update_time	Integer	修改时间。

表 4-308 components

参数	参数类型	描述
application_id	String	应用ID。
component_id	String	组件ID。
component_name	String	组件名称。
environment_id	String	环境ID。

## 请求示例

修改配置文件的分组ID为4808b50d-d7f3-401b-9175-968cb4f9892b、配置文件类型为properties、文件内容为testKey = testModify、不加密文件内容。

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

## 响应示例

```
{
  "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"
}
```

## 状态码

状态码	描述
204	操作成功。
400	错误的请求。
404	请求对象不存在。
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.5.9 根据配置文件 ID 获取配置文件历史

### 功能介绍

此API用于通过配置文件ID获取配置文件历史。

## URI

GET /v3/{project\_id}/cas/configs/{config\_id}/histories

表 4-309 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
config_id	是	String	配置文件ID。获取方法，请参考 <a href="#">获取配置文件信息</a> 。

表 4-310 查询参数

参数	是否必选	参数类型	描述
limit	否	integer	指定个数，取值[0, 100]。
offset	否	integer	指定偏移量。
order_by	否	String	排序字段，默认按创建时间排序。 排序字段支持枚举值： <ul style="list-style-type: none"><li>• create_time，配置文件创建时间。</li><li>• name，配置文件名称。</li><li>• update_time，配置文件修改时间。</li></ul>
order	否	String	排序方式。 <ul style="list-style-type: none"><li>• desc，降序排序，默认排序方式。</li><li>• asc，升序排序。</li></ul>

## 请求消息

表 4-311 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">认证鉴权</a> 。

## 响应消息

表 4-312 响应参数

参数	参数类型	描述
count	Integer	配置文件历史总数。
histories	Array of objects	配置文件历史列表, 请参考 <a href="#">表4-313</a> 。

表 4-313 histories

参数	参数类型	描述
config_history_id	String	配置文件历史ID。
config_id	String	配置文件ID。
content	String	配置文件历史内容。
version	String	配置文件历史版本。
create_time	Integer	创建时间。
components	Array of objects	配置文件绑定的组件, 请参考 <a href="#">表4-314</a> 。
creator	String	创建人。
group_id	String	配置分组ID。
config_name	String	配置文件名称。
description	String	配置文件描述。
sensitive	Boolean	配置文件内容是否加密: <ul style="list-style-type: none"><li>• true, 加密。</li><li>• false, 不加密。</li></ul>

表 4-314 components

参数	参数类型	描述
application_id	String	应用ID。
component_id	String	组件ID。
component_name	String	组件名称。
environment_id	String	环境ID。

## 请求示例

无

## 响应示例

```
{
  "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
    }
  ]
}
```

## 状态码

状态码	描述
204	操作成功。
400	错误的请求。
404	请求对象不存在。
500	内部错误。



## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.5.10 根据配置文件历史 ID 获取配置文件历史信息

### 功能介绍

此API用于通过配置文件历史ID获取配置文件历史信息。

### URI

GET /v3/{project\_id}/cas//configs/{config\_id}/histories/{config\_history\_id}

表 4-315 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
config_id	是	String	配置文件ID。获取方法，请参考 <a href="#">获取配置文件信息</a> 。
config_history_id	是	String	配置文件历史ID。获取方法，请参考 <a href="#">根据配置文件ID获取配置文件历史</a> 。

### 请求消息

表 4-316 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">认证鉴权</a> 。

## 响应消息

表 4-317 响应参数

参数	参数类型	描述
config_history_id	String	配置文件历史ID。
config_id	String	配置文件ID。
content	String	配置文件历史内容。
version	String	配置文件历史版本。
type	String	配置文件。
create_time	Integer	创建时间。
components	Array of objects	配置文件绑定的组件列表，请参考表4-318。
creator	String	创建人。
group_id	String	配置分组ID。
config_name	String	配置文件名称。
description	String	配置文件描述。
sensitive	Boolean	配置文件内容是否加密； <ul style="list-style-type: none"><li>• true，加密。</li><li>• false，不加密。</li></ul>

表 4-318 components

参数	参数类型	描述
application_id	String	应用ID。
component_id	String	组件ID。
component_name	String	组件名称。
environment_id	String	环境ID。

## 请求示例

无

## 响应示例

```
{  
  "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"
}
```

## 状态码

状态码	描述
204	操作成功。
400	错误的请求。
404	请求对象不存在。
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.5.11 根据配置文件历史 ID 删除配置文件历史

### 功能介绍

此API用于通过配置文件历史ID删除配置文件历史。

### URI

DELETE /v3/{project\_id}/cas/configs/{config\_id}/histories/{config\_history\_id}

表 4-319 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
config_id	是	String	配置文件ID。获取方法，请参考 <a href="#">获取配置文件信息</a> 。
config_history_id	是	String	配置文件历史ID。获取方法，请参考 <a href="#">根据配置文件ID获取配置文件历史</a> 。

## 请求消息

表 4-320 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">认证鉴权</a> 。

## 响应消息

无

## 请求示例

无

## 响应示例

无

## 状态码

状态码	描述
204	操作成功。
400	错误的请求。
404	请求对象不存在。
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.5.12 根据配置分组 ID 获取分组详情

### 功能介绍

此API用于通过配置分组ID获取配置分组详情。

## URI

GET /v3/{project\_id}/cas/config-groups/{config\_group\_id}

表 4-321 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
config_group_id	是	String	配置分组ID。获取方法，请参考 <a href="#">获取配置分组</a> 。

## 请求消息

表 4-322 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">认证鉴权</a> 。

## 响应消息

表 4-323 响应参数

参数	参数类型	描述
id	String	配置分组ID。
project_id	String	项目ID。
name	String	配置分组名称。
description	String	配置分组描述。
creator	String	创建人。
create_time	Integer	创建时间。

## 请求示例

无

## 响应示例

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

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在。
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.5.13 根据配置分组 ID 删除分组

### 功能介绍

此API用于根据配置分组ID删除分组。

### URI

DELETE /v3/{project\_id}/cas/config-groups/{config\_group\_id}

表 4-324 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
config_group_id	是	String	配置分组ID。获取方法，请参考 <a href="#">获取配置分组</a> 。

## 请求消息

表 4-325 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">认证鉴权</a> 。

## 响应消息

无

## 请求示例

无

## 响应示例

无

## 状态码

状态码	描述
204	操作成功。
400	错误的请求。
404	请求对象不存在。
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.6 泳道组

## 4.6.1 创建泳道组

### 功能介绍

此API用于创建泳道组。

泳道组是一组泳道的集合，用于区分不同的组织或场景。同一个项目下，最多支持创建300个泳道组。

### URI

POST /v3/{project\_id}/cas/swimlane-group

表 4-326 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。

### 请求消息

表 4-327 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

表 4-328 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	泳道组的名称。 由大小写英文字母、数字、中划线（-）、下划线（_）组成，并以大小写英文字母开头，大小写英文字母或者数字结尾，长度为2~64个字符。



参数	是否必选	参数类型	描述
gateways	否	Array of objects	泳道组关联网关列表，全链路流量的入口。请参考 <a href="#">表4-329</a> 。
description	否	String	泳道组的描述信息。 最大长度为128个字符。

表 4-329 gateways

参数	是否必选	参数类型	描述
engine_id	是	String	目标服务引擎ID，对应于目标服务对应的服务来源所用的引擎ID。
id	是	String	网关ID。
name	否	String	网关名称。
target_services	是	Array of objects	目标服务列表，对应于网关服务管理中的服务。请参考 <a href="#">表4-330</a> 。

表 4-330 target\_services

参数	是否必选	参数类型	描述
host	是	String	目标服务host信息。 host信息由目标服务的接入引擎后的微服务名、微服务所在环境名、微服务所属应用名称已经微服务接入的引擎类型组成，以点号(.)分隔。 例如，unit-controller.testing.test-application.servicecomb。unit-controller为接入引擎的微服务名称，testing为unit-controller微服务所在环境名，test-application为unit-controller微服务所属应用名称，servicecomb表示unit-controller微服务接入的是ServiceComb引擎。
name	否	String	目标服务名称。
id	是	String	目标服务ID。

参数	是否必选	参数类型	描述
engine_id	是	String	引擎ID。
engine_type	是	String	引擎类型。 <ul style="list-style-type: none"> <li>• servicecomb: ServiceComb引擎。</li> <li>• nacos: 注册配置中心。</li> </ul>

## 响应消息

表 4-331 响应参数

参数	参数类型	描述
name	String	泳道组的名称。
gateways	Array of objects	流量网关入口, 请参考 <a href="#">表4-332</a> 。
description	String	泳道组的描述。
gray_release_type	String	灰度发布类型。 <ul style="list-style-type: none"> <li>• WEIGHT: 基于流量比例进行全链路灰度。</li> <li>• CONTENT: 基于内容进行全链路灰度。</li> </ul>
id	String	泳道组的ID。
gateway_engine_ids	Array of String	网关ID列表。
creator	String	创建者。
create_time	String	创建时间。
update_time	String	更新时间
project_id	String	租户项目ID。

表 4-332 gateways

参数	参数类型	描述
engine_id	String	目标服务引擎ID, 对应于目标服务对应的服务来源所用的引擎ID。
id	String	网关ID。
name	String	网关名称。

参数	参数类型	描述
target_services	Array of objects	目标服务列表，对应于网关服务管理中的服务。请参考表4-333。

表 4-333 target\_services

参数	参数类型	描述
host	String	目标服务host信息。 host信息由目标服务的接入引擎后的微服务名、微服务所在环境名、微服务所属应用名称已经微服务接入的引擎类型组成，以点号(.)分隔。 例如，unit-controller.testing.test-application.servicecomb。unit-controller为接入引擎的微服务名称，testing为unit-controller微服务所在环境名，test-application为unit-controller微服务所属应用名称，servicecomb表示unit-controller微服务接入的是ServiceComb引擎。
name	String	目标服务名称。
id	String	目标服务ID。
engine_id	String	引擎ID。
engine_type	String	引擎类型。 <ul style="list-style-type: none"> <li>• servicecomb: ServiceComb引擎。</li> <li>• nacos: 注册配置中心。</li> </ul>

## 请求示例

创建一个名为test、描述信息为test的泳道组。该泳道组绑定了名为microGateway-cy8bhr的应用网关作为流量入口网关，通过该应用网关转发流量的目标服务为接入了ServiceComb引擎的所在环境为空、所属应用名称为demo-java-chassis-cse-v2的名为provider的微服务。

```
{
  "name": "test",
  "description": "test",
  "gateways": [
    {
      "id": "71949725-9c8f-483a-a05f-74d33ff1613e",
      "name": "microGateway-cy8bhr",
      "engine_id": "d68a5ca7-b5b6-4917-a464-0adef94b8067",
      "target_services": [
        {
          "id": "8177d237-b912-4167-a90d-33c603610d0b",
          "host": "provider.demo-java-chassis-cse-v2.servicecomb",
          "name": "provider",
          "engine_type": "servicecomb",
          "engine_id": "d68a5ca7-b5b6-4917-a464-0adef94b8067"
        }
      ]
    }
  ]
}
```

```
}  
]  
}
```

## 响应示例

```
{  
  "gateways": [  
    {  
      "id": "71949725-9c8f-483a-a05f-74d33ff1613e",  
      "name": "microGateway-cy8bhr",  
      "target_services": [  
        {  
          "id": "8177d237-b912-4167-a90d-33c603610d0b",  
          "host": "provider.demo-java-chassis-cse-v2.servicecomb",  
          "name": "provider",  
          "engine_id": "d68a5ca7-b5b6-4917-a464-0adef94b8067",  
          "engine_type": "servicecomb"  
        }  
      ],  
      "engine_id": "d68a5ca7-b5b6-4917-a464-0adef94b8067"  
    }  
  ],  
  "id": "ba1ca687-1895-4fc3-88ca-9db02ce2a14e",  
  "name": "test",  
  "gray_release_type": "WEIGHT",  
  "description": "test",  
  "project_id": "578ac30b81034b89a7255b3af26db9c9",  
  "creator": "test_user",  
  "create_time": 1726623793113,  
  "update_time": 1726623793113,  
  "gateway_engine_ids": [  
    "d68a5ca7-b5b6-4917-a464-0adef94b8067"  
  ]  
}
```

## 状态码

状态码	描述
200	操作成功
400	错误的请求
404	请求对象不存在
500	内部错误

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.6.2 获取所有泳道组

### 功能介绍

此API用于获取所有已经创建的泳道组。

## URI

GET /v3/{project\_id}/cas/swimlane-group

表 4-334 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。

## 请求消息

表 4-335 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

## 响应消息

表 4-336 响应参数

参数	参数类型	描述
count	Integer	泳道组总数。
swimlaneGroups	Array of objects	泳道组列表，请参考 <a href="#">表4-337</a> 。

表 4-337 swimlaneGroups

参数	参数类型	描述
description	String	泳道组的描述。
gray_release_type	String	灰度发布类型。 <ul style="list-style-type: none"><li>WEIGHT：基于流量比例进行全链路灰度。</li><li>CONTENT：基于内容进行全链路灰度。</li></ul>
gateways	Array of objects	流量网关入口，请参考 <a href="#">表4-338</a> 。

参数	参数类型	描述
name	String	泳道组的名称。
id	String	泳道组的ID。
gateway_engine_ids	Array of String	网关ID列表。
creator	String	创建者。
create_time	String	创建时间。
update_time	String	更新时间
project_id	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。

表 4-338 gateways

参数	参数类型	描述
engine_id	String	引擎ID。
id	String	网关ID。
name	String	网关名称。
target_services	Array of objects	目标服务列表。请参考 <a href="#">表4-339</a> 。

表 4-339 target\_services

参数	参数类型	描述
host	String	目标服务host信息。 host信息由目标服务的接入引擎后的微服务名、微服务所在环境名、微服务所属应用名称已经微服务接入的引擎类型组成，以点号 (.) 分隔。 例如，unit-controller.testing.test-application.servicecomb。unit-controller为接入引擎的微服务名称，testing为unit-controller微服务所在环境名，test-application为unit-controller微服务所属应用名称，servicecomb表示unit-controller微服务接入的是ServiceComb引擎。
name	String	目标服务名称。
id	String	目标服务ID。
engine_id	String	引擎ID。

参数	参数类型	描述
engine_type	String	引擎类型。 <ul style="list-style-type: none"><li>• servicecomb: ServiceComb引擎。</li><li>• nacos: 注册配置中心。</li></ul>

### 请求示例

无

### 响应示例

```
{
  "gateways": [
    {
      "id": "71949725-9c8f-483a-a05f-74d33ff1613e",
      "name": "microGateway-cy8bhr",
      "target_services": [
        {
          "id": "8177d237-b912-4167-a90d-33c603610d0b",
          "host": "provider.demo-java-chassis-cse-v2.servicecomb",
          "name": "provider",
          "engine_id": "d68a5ca7-b5b6-4917-a464-0adef94b8067",
          "engine_type": "servicecomb"
        }
      ],
      "engine_id": "d68a5ca7-b5b6-4917-a464-0adef94b8067"
    }
  ],
  "id": "ba1ca687-1895-4fc3-88ca-9db02ce2a14e",
  "name": "test",
  "gray_release_type": "WEIGHT",
  "description": "test",
  "project_id": "578ac30b81034b89a7255b3af26db9c9",
  "creator": "test_user",
  "create_time": 1726623793113,
  "update_time": 1726623793113,
  "gateway_engine_ids": [
    "d68a5ca7-b5b6-4917-a464-0adef94b8067"
  ]
}
```

### 状态码

状态码	描述
200	操作成功
400	错误的请求
404	请求对象不存在
500	内部错误

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.6.3 根据泳道组 ID 获取泳道组信息

### 功能介绍

此API用于根据泳道组ID获取泳道组信息。

### URI

GET /v3/{project\_id}/cas/swimlane-group/{lane\_group\_id}

表 4-340 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
lane_group_id	是	String	泳道组ID。获取方法，请参考 <a href="#">6.4.2-获取所有泳道组</a> 。

### 请求消息

表 4-341 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

### 响应消息

表 4-342 响应参数

参数	参数类型	描述
description	String	泳道组的描述。



参数	参数类型	描述
gateways	Array of objects	流量网关入口, 请参考 <a href="#">表4-343</a> 。
gray_release_type	String	灰度发布类型。 <ul style="list-style-type: none"> <li>WEIGHT: 基于流量比例进行全链路灰度。</li> <li>CONTENT: 基于内容进行全链路灰度。</li> </ul>
id	String	泳道组的ID。
name	String	泳道组的名称。
swimlane_count	Integer	泳道组下的泳道数量。
swimlanes	Array of objects	泳道组下的泳道列表, 请参考 <a href="#">表4-345</a> 。
creator	String	创建者。
create_time	String	创建时间。
update_time	String	更新时间
project_id	String	租户项目ID。获取方法, 请参考 <a href="#">获取项目ID</a> 。
target_services	Array of objects	目标服务列表。请参考 <a href="#">表4-344</a> 。

表 4-343 gateways

参数	参数类型	描述
engine_id	String	网关引擎ID。
id	String	网关ID。
name	String	网关名称。
target_services	Array of objects	目标服务列表。请参考 <a href="#">表4-344</a> 。

表 4-344 target\_services

参数	是否必选	参数类型	描述
host	是	String	目标服务host信息。 host信息由目标服务的接入引擎后的微服务名、微服务所在环境名、微服务所属应用名称已经微服务接入的引擎类型组成，以点号(.)分隔。 例如，unit-controller.testing.test-application.servicecomb。unit-controller为接入引擎的微服务名称，testing为unit-controller微服务所在环境名，test-application为unit-controller微服务所属应用名称，servicecomb表示unit-controller微服务接入的是ServiceComb引擎。
name	否	String	目标服务名称。
id	是	String	目标服务ID。
engine_id	是	String	引擎ID。
engine_type	是	String	引擎类型。 <ul style="list-style-type: none"><li>• servicecomb: ServiceComb引擎。</li><li>• nacos: 注册配置中心。</li></ul>

表 4-345 swimlanes

参数	参数类型	描述
id	String	泳道ID。
name	String	泳道名称。
swimlane_group_id	String	所属的泳道组ID。
type	String	泳道类型。 <ul style="list-style-type: none"><li>• BASE: 基线泳道。</li><li>• GRAY: 灰度泳道。</li></ul>
tag	String	泳道标签，用于根据标签来控制流量的走向。
route_status	String	泳道路由状态。 <ul style="list-style-type: none"><li>• ENABLE: 开启路由。</li><li>• DISABLED: 关闭路由。</li></ul>

参数	参数类型	描述
rule_match_mode	String	路由规则匹配模式，当泳道组灰度类型为基于内容即gray_release_type为CONTENT时生效。 <ul style="list-style-type: none"><li>● ALL：匹配所有的规则。</li><li>● ANY：匹配任一规则。</li></ul>
rules	Array of objects	路由规则，当泳道组灰度类型为基于内容即gray_release_type为CONTENT时生效。请参考表4-346。
weight	Integer	泳道流量权重，范围在0-100之间。当泳道组灰度类型为基于流量即gray_release_type为WEIGHT时生效。
instances	Array of objects	泳道中纳管的组件列表，请参考表4-347。
instance_count	Integer	泳道中纳管的组件个数。
latest_opt	String	泳道上一步的操作。 <ul style="list-style-type: none"><li>● CLONE：克隆。</li><li>● UPGRADE：升级。</li><li>● ROLLBACK：回滚。</li><li>● CREATE：创建。</li></ul>
release_plan_id	String	泳道上一步操作关联的发布单ID。
project_id	String	项目ID。
create_time	Integer	创建时间。
update_time	Integer	更新时间。
creator	String	创建人。

表 4-346 rules

参数	参数类型	描述
type	String	灰度匹配规则类型。当前只支持HEADER，即基于Header头进行匹配。
key	String	键。
value	String	值。

参数	参数类型	描述
condition	String	匹配条件。 <ul style="list-style-type: none"> <li>EXACT: 精确匹配。</li> <li>PREFIX: 前缀匹配。</li> <li>REGEX: 正则匹配。</li> </ul>

表 4-347 instances

参数	参数类型	描述
id	String	应用组件实例ID。
name	String	应用组件实例名称。
version	String	应用组件版本号。
application_id	String	应用ID。
component_id	String	组件ID。
application_name	String	应用名称。
status	String	实例状态
engine_id	String	组件关联引擎ID。
runtime_stack	Object	运行时, 请参考 <a href="#">表4-348</a> 。
replica	Integer	实例副本数。

表 4-348 runtime\_stack

参数	参数类型	描述
name	String	技术栈名称。
type	String	技术栈类型, 支持Java、Tomcat、Nodejs、Php、Docker、Python。 当部署模式为虚拟机部署时, 仅支持Java、Tomcat、Nodejs; 容器部署上述类型都支持。
version	String	技术栈版本。
deploy_mode	String	部署模式。 <ul style="list-style-type: none"> <li>container, 容器部署。</li> <li>virtualmachine, 虚拟机部署。</li> </ul>

## 请求示例

无

## 响应示例

```
{
  "gateways": [
    {
      "id": "71949725-9c8f-483a-a05f-74d33ff1613e",
      "name": "microGateway-cy8bhr",
      "target_services": [
        {
          "id": "8177d237-b912-4167-a90d-33c603610d0b",
          "host": "provider.demo-java-chassis-cse-v2.servicecomb",
          "name": "provider",
          "engine_id": "d68a5ca7-b5b6-4917-a464-0adef94b8067",
          "engine_type": "servicecomb"
        }
      ]
    },
    {
      "engine_id": "d68a5ca7-b5b6-4917-a464-0adef94b8067"
    }
  ],
  "id": "ba1ca687-1895-4fc3-88ca-9db02ce2a14e",
  "name": "test",
  "gray_release_type": "WEIGHT",
  "description": "test",
  "project_id": "578ac30b81034b89a7255b3af26db9c9",
  "creator": "test_user",
  "create_time": 1726623793113,
  "update_time": 1726623793113,
  "swimlanes": [
    {
      "id": "1cd0cb71-5d5d-41a0-8b05-f7ed53400cf7",
      "name": "base",
      "swimlane_group_id": "ba1ca687-1895-4fc3-88ca-9db02ce2a14e",
      "type": "BASE",
      "tag": "base",
      "route_status": "ENABLE",
      "rule_match_mode": null,
      "project_id": "578ac30b81034b89a7255b3af26db9c9",
      "rules": null,
      "creator": "test_user",
      "create_time": 1726626094346,
      "update_time": 1726626094346,
      "instances": [
        {
          "componentId": "c14f17b7-ac44-4738-8b57-4be7eff7a956",
          "environmentId": "d1b623d9-3350-4777-8979-5b4b571ca518",
          "id": "5fd78836-301e-4750-bdc9-be8a9e43b19a",
          "name": "weater-beta",
          "version": "2024.0914.17114",
          "application_id": "69993ce9-e468-3349-acb8-a30b85b17de5",
          "application_name": "weathermap",
          "status": "RUNNING",
          "engine_id": "d68a5ca7-b5b6-4917-a464-0adef94b8067",
          "runtime_stack": {
            "id": "656d9392-658f-4c2c-bb4f-3702a1169267",
            "url": "openjdk-{arch}:8-1.3.8",
            "type": "Java",
            "name": "OpenJDK8",
            "deploy_mode": "container",
            "app_spec_name": "",
            "version": "1.3.8",
            "status": "Supported",
            "release_note": "Fixed some security issues and bugs.",
            "spec": {
              "os": "EulerOS 2.9.8",
              "sdk": "OpenJDK-8u411",
            }
          }
        }
      ]
    }
  ]
}
```

```
        "digest": null
      },
      "parameters": null,
      "system": null
    },
    "replica": 1
  }
],
"instance_count": 1,
"weight": 100,
"batch_status": null,
"latest_opt": null,
"release_plan_id": null
}
],
"swimlane_count": 1,
"engine_ids": [
  "d68a5ca7-b5b6-4917-a464-0adef94b8067"
]
}
```

## 状态码

状态码	描述
200	操作成功
400	错误的请求
404	请求对象不存在
500	内部错误

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.6.4 根据泳道组 ID 修改泳道组

### 功能介绍

此API用于根据泳道组ID修改泳道组。

### URI

PUT /v3/{project\_id}/cas/swimlane-group/{lane\_group\_id}

表 4-349 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。

参数	是否必选	参数类型	描述
lane_group_id	是	String	泳道组ID。获取方法，请参考 <a href="#">6.4.2-获取所有泳道组</a> 。

## 请求消息

表 4-350 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

表 4-351 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	泳道组的名称。 由大小写英文字母、数字、中划线（-）、下划线（_）组成，并以大小写英文字母开头，大小写英文字母或者数字结尾，长度为2~64个字符。
gateways	是	Array of objects	泳道组关联网关列表，全链路流量的入口。请参考 <a href="#">表4-352</a> 。
description	否	String	泳道组的描述。 最大长度为128个字符。

表 4-352 gateways

参数	是否必选	参数类型	描述
engine_id	是	String	目标服务引擎ID，对应于目标服务对应的服务来源所用的引擎ID。
id	是	String	网关ID。

参数	是否必选	参数类型	描述
name	是	String	网关名称。
target_services	是	Array of objects	目标服务列表，对应于网关服务管理中的服务。请参考 <a href="#">表4-353</a> 。

表 4-353 target\_services

参数	是否必选	参数类型	描述
host	是	String	目标服务host信息。 host信息由目标服务的接入引擎后的微服务名、微服务所在环境名、微服务所属应用名称已经微服务接入的引擎类型组成，以点号(.)分隔。 例如，unit-controller.testing.test-application.servicecomb。unit-controller为接入引擎的微服务名称，testing为unit-controller微服务所在环境名，test-application为unit-controller微服务所属应用名称，servicecomb表示unit-controller微服务接入的是ServiceComb引擎。
name	否	String	目标服务名称。
id	是	String	目标服务ID。
engine_id	是	String	引擎ID。
engine_type	是	String	引擎类型。 <ul style="list-style-type: none"><li>• servicecomb: ServiceComb引擎。</li><li>• nacos: 注册配置中心。</li></ul>

## 响应消息

表 4-354 响应参数

参数	参数类型	描述
id	String	泳道组ID。
name	String	泳道组名称。
description	String	泳道组描述。



参数	参数类型	描述
creator	String	创建人。
project_id	String	租户项目ID。
gray_release_type	String	灰度发布类型。 <ul style="list-style-type: none"> <li>WEIGHT: 基于流量比例进行全链路灰度。</li> <li>CONTENT: 基于内容进行全链路灰度。</li> </ul>
gateway_engine_ids	Array of String	网关目标服务引擎ID。
gateways	Array of objects	目标服务列表, 请参考 <a href="#">表 gateways</a> 。
create_time	Integer	创建时间。
update_time	Integer	更新时间。
swimlanes	Array of objects	泳道列表, 请参考 <a href="#">表4-357</a> 。

表 4-355 gateways

参数	是否必选	参数类型	描述
engine_id	是	String	目标服务引擎ID, 对应于目标服务对应的服务来源所用的引擎ID。
id	是	String	网关ID。
name	是	String	网关名称。
target_services	是	Array of objects	目标服务列表, 对应于网关服务管理中的服务。请参考 <a href="#">表4-356</a> 。

表 4-356 target\_services

参数	参数类型	描述
host	String	目标服务host信息。 host信息由目标服务的接入引擎后的微服务名、微服务所在环境名、微服务所属应用名称已经微服务接入的引擎类型组成，以点号(.)分隔。 例如，unit-controller.testing.test-application.servicecomb。unit-controller为接入引擎的微服务名称，testing为unit-controller微服务所在环境名，test-application为unit-controller微服务所属应用名称，servicecomb表示unit-controller微服务接入的是ServiceComb引擎。
name	String	目标服务名称。
id	String	目标服务ID。
engine_id	String	引擎ID。
engine_type	String	引擎类型。 <ul style="list-style-type: none"><li>• servicecomb: ServiceComb引擎。</li><li>• nacos: 注册配置中心。</li></ul>

表 4-357 swimlanes

参数	参数类型	描述
id	String	泳道ID。
name	String	泳道名称。
swimlane_group_id	String	所属的泳道组ID。
type	String	泳道类型。 <ul style="list-style-type: none"><li>• BASE: 基线泳道。</li><li>• GRAY: 灰度泳道。</li></ul>
tag	String	泳道标签，用于根据标签来控制流量的走向。
route_status	String	泳道路由状态。 <ul style="list-style-type: none"><li>• ENABLE: 开启路由。</li><li>• DISABLED: 关闭路由。</li></ul>
rule_match_mode	String	路由规则匹配模式，当泳道组灰度类型为基于内容即gray_release_type为CONTENT时生效。 <ul style="list-style-type: none"><li>• ALL: 匹配所有的规则。</li><li>• ANY: 匹配任一规则。</li></ul>

参数	参数类型	描述
rules	Array of objects	路由规则，当泳道组灰度类型为基于内容即 gray_release_type 为 CONTENT 时生效。请参考表 4-358。
weight	Integer	泳道流量权重，范围在 0-100 之间。 当泳道组灰度类型为基于流量即 gray_release_type 为 WEIGHT 时生效。
instances	Array of objects	泳道中纳管的组件列表，请参考表 4-359。
instance_count	Integer	泳道中纳管的组件个数。
latest_opt	String	泳道上一步的操作。 <ul style="list-style-type: none"><li>● CLONE: 克隆。</li><li>● UPGRADE: 升级。</li><li>● ROLLBACK: 回滚。</li><li>● CREATE: 创建。</li></ul>
release_plan_id	String	泳道上一步操作关联的发布单 ID。
project_id	String	项目 ID。
create_time	Integer	创建时间。
update_time	Integer	更新时间。
creator	String	创建人。

表 4-358 rules

参数	参数类型	描述
type	String	灰度匹配规则类型。当前只支持 HEADER，即基于 Header 头进行匹配。
key	String	键。
value	String	值。
condition	String	匹配条件。 <ul style="list-style-type: none"><li>● EXACT: 精确匹配。</li><li>● PREFIX: 前缀匹配。</li><li>● REGEX: 正则匹配。</li></ul>

表 4-359 instances

参数	参数类型	描述
id	String	应用组件实例ID。
name	String	应用组件实例名称。
version	String	应用组件版本号。
application_id	String	应用ID。
component_id	String	组件ID。
application_name	String	应用名称。
status	String	实例状态
engine_id	String	组件关联引擎ID。
runtime_stack	Object	运行时，请参考表4-360。
replica	Integer	实例副本数。

表 4-360 runtime\_stack

参数	参数类型	描述
name	String	技术栈名称。
type	String	技术栈类型，支持Java、Tomcat、Nodejs、Php、Docker、Python。 当部署模式为虚拟机部署时，仅支持Java、Tomcat、Nodejs；容器部署上述类型都支持。
version	String	技术栈版本。
deploy_mode	String	部署模式。 <ul style="list-style-type: none"><li>• container，容器部署。</li><li>• virtualmachine，虚拟机部署。</li></ul>

## 请求示例

修改id为ba1ca687-1895-4fc3-88ca-9db02ce2a14e的泳道组的名称、描述、流量入口网关以及目标服务。

```
{
  "name": "test",
  "description": "test-descption",
  "gateways": [
    {
      "id": "71949725-9c8f-483a-a05f-74d33ff1613e",
      "name": "microGateway-cy8bhr",
      "engine_id": "d68a5ca7-b5b6-4917-a464-0adef94b8067",
    }
  ]
}
```

```
    "target_services": [
      {
        "id": "8177d237-b912-4167-a90d-33c603610d0b",
        "host": "provider..demo-java-chassis-cse-v2.servicecomb",
        "name": "provider",
        "engine_type": "servicecomb",
        "engine_id": "d68a5ca7-b5b6-4917-a464-0adef94b8067"
      }
    ]
  }
}
```

## 响应示例

```
{
  "gateways": [
    {
      "id": "71949725-9c8f-483a-a05f-74d33ff1613e",
      "name": "microGateway-cy8bhr",
      "target_services": [
        {
          "id": "8177d237-b912-4167-a90d-33c603610d0b",
          "host": "provider..demo-java-chassis-cse-v2.servicecomb",
          "name": "provider",
          "engine_id": "d68a5ca7-b5b6-4917-a464-0adef94b8067",
          "engine_type": "servicecomb"
        }
      ],
      "engine_id": "d68a5ca7-b5b6-4917-a464-0adef94b8067",
    }
  ],
  "id": "ba1ca687-1895-4fc3-88ca-9db02ce2a14e",
  "name": "test",
  "gray_release_type": "WEIGHT",
  "description": "test-description",
  "project_id": "578ac30b81034b89a7255b3af26db9c9",
  "creator": "test_user",
  "create_time": 1726623793113,
  "update_time": 1726623793113,
  "swimlanes": [
    {
      "id": "1cd0cb71-5d5d-41a0-8b05-f7ed53400cf7",
      "name": "base",
      "swimlane_group_id": "ba1ca687-1895-4fc3-88ca-9db02ce2a14e",
      "type": "BASE",
      "tag": "base",
      "route_status": "ENABLE",
      "rule_match_mode": null,
      "project_id": "578ac30b81034b89a7255b3af26db9c9",
      "rules": null,
      "creator": "test_user",
      "create_time": 1726626094346,
      "update_time": 1726626094346,
      "instances": [
        {
          "componentId": "c14f17b7-ac44-4738-8b57-4be7eff7a956",
          "environmentId": "d1b623d9-3350-4777-8979-5b4b571ca518",
          "id": "5fd78836-301e-4750-bdc9-be8a9e43b19a",
          "name": "weater-beta",
          "version": "2024.0914.17114",
          "application_id": "69993ce9-e468-3349-acb8-a30b85b17de5",
          "application_name": "weathermap",
          "status": "RUNNING",
          "engine_id": "d68a5ca7-b5b6-4917-a464-0adef94b8067",
          "runtime_stack": {
            "id": "656d9392-658f-4c2c-bb4f-3702a1169267",
            "url": "openjdk-{arch}:8-1.3.8",
            "type": "Java",
          }
        }
      ]
    }
  ]
}
```

```
    "name": "OpenJDK8",
    "deploy_mode": "container",
    "app_spec_name": "",
    "version": "1.3.8",
    "status": "Supported",
    "release_note": "Fixed some security issues and bugs.",
    "spec": {
      "os": "EulerOS 2.9.8",
      "sdk": "OpenJDK-8u411",
      "digest": null
    },
    "parameters": null,
    "system": null
  },
  "replica": 1
}
],
"instance_count": 1,
"weight": 100,
"batch_status": null,
"latest_opt": null,
"release_plan_id": null
}
],
"swimlane_count": 1,
"engine_ids": [
  "d68a5ca7-b5b6-4917-a464-0adef94b8067"
]
}
```

## 状态码

状态码	描述
200	操作成功
400	错误的请求
404	请求对象不存在
500	内部错误

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.6.5 根据泳道组 ID 删除泳道组

### 功能介绍

此API用于根据泳道组ID删除泳道组。

### URI

DELETE /v3/{project\_id}/cas/swimlane-group/{lane\_group\_id}

表 4-361 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
lane_group_id	是	String	泳道组ID。获取方法，请参考 <a href="#">6.4.2-获取所有泳道组</a> 。

## 请求消息

表 4-362 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

## 响应消息

无

## 请求示例

无

## 响应示例

无

## 状态码

状态码	描述
200	操作成功
400	错误的请求
404	请求对象不存在
500	内部错误

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.6.6 根据网关 ID 获取目标服务列表

### 功能介绍

此API用于根据网关ID获取目标服务列表。

### URI

GET /v3/{project\_id}/cas/swimlane-group/target-services

表 4-363 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。

表 4-364 查询参数

参数	是否必选	参数类型	描述
gateway_id	是	String	网关ID。

### 请求消息

表 4-365 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。



## 响应消息

表 4-366 响应参数

参数	参数类型	描述
id	String	目标服务ID。
host	String	目标服务host信息。 host信息由目标服务的接入引擎后的微服务名、微服务所在环境名、微服务所属应用名称已经微服务接入的引擎类型组成，以点号(.)分隔。 例如，unit-controller.testing.test-application.servicecomb。unit-controller为接入引擎的微服务名称，testing为unit-controller微服务所在环境名，test-application为unit-controller微服务所属应用名称，servicecomb表示unit-controller微服务接入的是ServiceComb引擎。
name	String	目标服务名称。
service_source_name	String	目标服务对应的服务来源名称。
engine_id	String	目标服务对应的服务来源引擎ID。
engine_type	String	目标服务对应的服务来源引擎类型。
swimlane_group_id	String	泳道组ID。
swimlane_group_name	String	泳道组名称。
status	String	目标服务状态。 <ul style="list-style-type: none"><li>• TARGET_SERVICE_MANAGED: 目标服务已经被纳管。</li><li>• MICROSERVICE_OF_TARGET_SERVICE_UNROUTING: 目标服务对应的微服务不支持路由。</li><li>• MICROSERVICE_OF_TARGET_SERVICE_MANAGED: 目标服务对应的微服务被纳管。</li></ul>

## 请求示例

无

## 响应示例

[  
{

```
[{"id": "8177d237-b912-4167-a90d-33c603610d0b", "host": "provider..demo-java-chassis-cse-v2.servicecomb", "name": "provider", "service_source_name": "cscs", "engine_id": "d68a5ca7-b5b6-4917-a464-0adef94b8067", "engine_type": "servicecomb", "swimlane_group_id": null, "swimlane_group_name": null, "status": null}, {"id": "8d8d5d85-2ae6-463b-b2e9-0fa9d9a0aa7d", "host": "test.static", "name": "test", "service_source_name": "static", "engine_id": null, "engine_type": null, "swimlane_group_id": null, "swimlane_group_name": null, "status": "MICROSERVICE_OF_TARGET_SERVICE_UNROUTING"}]
```

## 状态码

状态码	描述
200	操作成功
400	错误的请求
404	请求对象不存在
500	内部错误

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.6.7 根据泳道组 ID 修改泳道组网关路由配置

### 功能介绍

此API用于根据泳道组ID修改泳道组网关路由配置。

### URI

PUT /v3/{project\_id}/cas/swimlane-group/{lane\_group\_id}/route

表 4-367 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。

参数	是否必选	参数类型	描述
lane_group_id	是	String	泳道组ID。获取方法，请参考 <a href="#">6.4.2-获取所有泳道组</a> 。

## 请求消息

表 4-368 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

表 4-369 请求 Body 参数

参数	是否必选	参数类型	描述
gray_release_type	是	String	泳道组灰度类型。 <ul style="list-style-type: none"><li>WEIGHT：基于流量比例进行全链路灰度。</li><li>CONTENT：基于内容进行全链路灰度。</li></ul>
swimlanes	是	Array of objects	泳道列表，请参考 <a href="#">表4-370</a> 。

表 4-370 swimlanes

参数	是否必选	参数类型	描述
rule_match_mode	是	String	路由规则匹配模式。当泳道组灰度类型为基于内容，即gray_release_type为CONTENT时生效。 <ul style="list-style-type: none"><li>ALL：匹配所有的规则。</li><li>ANY：匹配任一规则。</li></ul>

参数	是否必选	参数类型	描述
rules	是	Array of objects	路由规则。当泳道组灰度类型为基于内容，即gray_release_type为CONTENT时生效。请参考表4-371。
weight	是	Integer	泳道流量权重，范围在0-100之间。当泳道组灰度类型为基于流量，即gray_release_type为WEIGHT时生效。

表 4-371 rules

参数	是否必选	参数类型	描述
type	是	String	灰度匹配规则类型。当前只支持HEADER，即基于Header头进行匹配。
key	是	String	键。
value	是	String	值。
condition	是	String	匹配条件。 <ul style="list-style-type: none"><li>● EXACT: 精确匹配</li><li>● PREFIX: 前缀匹配。</li><li>● REGEX: 正则匹配。</li></ul>

## 响应消息

表 4-372 响应参数

参数	参数类型	描述
id	String	泳道组ID。
name	String	泳道组名称。
description	String	泳道组描述。
creator	String	创建人。
project_id	String	项目ID。
gray_release_type	String	泳道组灰度类型。 <ul style="list-style-type: none"><li>● WEIGHT: 基于流量比例进行全链路灰度。</li><li>● CONTENT: 基于内容进行全链路灰度。</li></ul>
engine_ids	Array of String	网关目标服务引擎ID。

参数	参数类型	描述
gateways	Array of objects	目标服务列表, 请参考 <a href="#">表4-373</a> 。
swimlane_count	Integer	泳道数量。
swimlanes	Array of objects	泳道列表, 请参考 <a href="#">表4-375</a> 。
create_time	Integer	创建时间。
update_time	Integer	更新时间。

表 4-373 gateways

参数	参数类型	描述
id	String	网关ID。
name	String	网关名称。
engine_id	String	目标服务引擎ID, 对应于目标服务对应的服务来源所用的引擎ID。
target_services	Array of objects	目标服务列表, 对应于网关服务管理中的服务。请参考 <a href="#">表4-374</a> 。

表 4-374 target\_services

参数	参数类型	描述
id	String	目标服务ID。
name	String	目标服务名称。
host	String	目标服务host信息。 host信息由目标服务的接入引擎后的微服务名、微服务所在环境名、微服务所属应用名称已经微服务接入的引擎类型组成, 以点号(.)分隔。 例如, unit-controller.testing.test-application.servicecomb。unit-controller为接入引擎的微服务名称, testing为unit-controller微服务所在环境名, test-application为unit-controller微服务所属应用名称, servicecomb表示unit-controller微服务接入的是ServiceComb引擎。

表 4-375 swimlanes

参数	参数类型	描述
id	String	泳道ID。
name	String	泳道名称。
swimlane_group_id	String	所属的泳道组ID。
type	String	泳道类型。 <ul style="list-style-type: none"><li>• BASE: 基线泳道。</li><li>• GRAY: 灰度泳道。</li></ul>
tag	String	泳道标签, 根据标签来控制流量的走向。
route_status	String	泳道路由状态。 <ul style="list-style-type: none"><li>• ENABLE: 开启路由。</li><li>• DISABLED: 关闭路由。</li></ul>
rule_match_mode	String	路由规则匹配模式。包括ALL和ANY。 ALL表示匹配所有的规则、ANY表示匹配任一规则。 当泳道组灰度类型为基于内容, 即 gray_release_type为CONTENT时生效。
rules	Array of objects	路由规则。 当泳道组灰度类型为基于内容, 即 gray_release_type为CONTENT时生效。 请参考 <a href="#">表4-376</a> 。
weight	Integer	泳道流量权重。范围在0-100之间。 当泳道组灰度类型为基于流量, 即 gray_release_type为WEIGHT时生效。
instances	Array of objects	泳道中纳管的组件列表。 请参考 <a href="#">表4-377</a> 。
instance_count	Integer	泳道中纳管的组件个数。
latest_opt	String	泳道上一步的操作。 <ul style="list-style-type: none"><li>• CLONE: 克隆。</li><li>• UPGRADE: 升级。</li><li>• ROLLBACK: 回滚。</li><li>• CREATE: 创建。</li></ul>
release_plan_id	String	泳道上一步操作关联的发布单ID。
project_id	String	项目ID。

参数	参数类型	描述
create_time	Integer	创建时间。
update_time	Integer	更新时间。
creator	String	创建人。

表 4-376 rules

参数	参数类型	描述
type	String	灰度匹配规则类型。当前只支持HEADER，即基于Header头进行匹配。
key	String	键。
value	String	值。
condition	String	匹配条件。 <ul style="list-style-type: none"><li>● EXACT: 精确匹配。</li><li>● PREFIX: 前缀匹配。</li><li>● REGEX: 正则匹配。</li></ul>

表 4-377 instances

参数	参数类型	描述
id	String	应用组件实例ID。
name	String	应用组件实例名称。
version	String	应用组件版本号。
application_id	String	应用ID。
component_id	String	组件ID。
application_name	String	应用名称。
status	String	实例状态。
engine_id	String	组件关联引擎ID。
runtime_stack	Object	运行时，请参考 <a href="#">表4-378</a> 。
replica	Integer	实例副本数。

表 4-378 runtime\_stack

参数	参数类型	描述
name	String	技术栈名称。
type	String	技术栈类型，支持Java、Tomcat、Nodejs、Php、 Docker、Python。 当部署模式为虚拟机部署时，仅支持Java、Tomcat、 Nodejs；容器部署上述类型都支持。
version	String	技术栈版本。
deploy_mode	String	部署模式。 <ul style="list-style-type: none"><li>• container，容器部署。</li><li>• virtualmachine，虚拟机部署。</li></ul>

## 请求示例

修改ID为ba1ca687-1895-4fc3-88ca-9db02ce2a14e的泳道组网关路由配置。

```
{
  "gray_release_type": "WEIGHT",
  "swimlanes": [
    {
      "id": "1cd0cb71-5d5d-41a0-8b05-f7ed53400cf7",
      "weight": 50
    },
    {
      "id": "414dccf2-3c15-47c2-8479-0337a5ccc4d9",
      "weight": 50
    }
  ]
}
```

## 响应示例

```
{
  "gateways": [
    {
      "id": "71949725-9c8f-483a-a05f-74d33ff1613e",
      "name": "microGateway-cy8bhr",
      "target_services": [
        {
          "id": "8177d237-b912-4167-a90d-33c603610d0b",
          "host": "provider.demo-java-chassis-cse-v2.servicecomb",
          "name": "provider",
          "engine_id": "d68a5ca7-b5b6-4917-a464-0adef94b8067",
          "engine_type": "servicecomb"
        }
      ]
    },
    {
      "engine_id": "d68a5ca7-b5b6-4917-a464-0adef94b8067"
    }
  ],
  "id": "ba1ca687-1895-4fc3-88ca-9db02ce2a14e",
  "name": "test",
  "gray_release_type": "WEIGHT",
  "description": "test-descption",
  "project_id": "578ac30b81034b89a7255b3af26db9c9",
  "creator": "test_user",
  "create_time": 1726623793113,
}
```



```
"update_time": 1726623793113,
"swimlanes": [
  {
    "id": "1cd0cb71-5d5d-41a0-8b05-f7ed53400cf7",
    "name": "base",
    "swimlane_group_id": "ba1ca687-1895-4fc3-88ca-9db02ce2a14e",
    "type": "BASE",
    "tag": "base",
    "route_status": "ENABLE",
    "rule_match_mode": null,
    "project_id": "578ac30b81034b89a7255b3af26db9c9",
    "rules": null,
    "creator": "test_user",
    "create_time": 1726626094346,
    "update_time": 1726626094346,
    "instances": [
      {
        "componentId": "c14f17b7-ac44-4738-8b57-4be7eff7a956",
        "environmentId": "d1b623d9-3350-4777-8979-5b4b571ca518",
        "id": "5fd78836-301e-4750-bdc9-be8a9e43b19a",
        "name": "weater-beta",
        "version": "2024.0914.17114",
        "application_id": "69993ce9-e468-3349-acb8-a30b85b17de5",
        "application_name": "weathermap",
        "status": "RUNNING",
        "engine_id": "d68a5ca7-b5b6-4917-a464-0adef94b8067",
        "runtime_stack": {
          "id": "656d9392-658f-4c2c-bb4f-3702a1169267",
          "url": "openjdk-{arch}:8-1.3.8",
          "type": "Java",
          "name": "OpenJDK8",
          "deploy_mode": "container",
          "app_spec_name": "",
          "version": "1.3.8",
          "status": "Supported",
          "release_note": "Fixed some security issues and bugs.",
          "spec": {
            "os": "EulerOS 2.9.8",
            "sdk": "OpenJDK-8u411",
            "digest": null
          },
          "parameters": null,
          "system": null
        },
        "replica": 1
      }
    ],
    "instance_count": 1,
    "weight": 50,
    "batch_status": null,
    "latest_opt": null,
    "release_plan_id": null
  },
  {
    "id": "414dccf2-3c15-47c2-8479-0337a5ccc4d9",
    "name": "gray",
    "swimlane_group_id": "ba1ca687-1895-4fc3-88ca-9db02ce2a14e",
    "type": "GRAY",
    "tag": "gray",
    "route_status": "ENABLE",
    "rule_match_mode": null,
    "project_id": "578ac30b81034b89a7255b3af26db9c9",
    "rules": null,
    "creator": "test_user",
    "create_time": 1726627610327,
    "update_time": 1726627610327,
    "instances": null,
    "instance_count": 0,
    "weight": 50,
  }
]
```

```
    "batch_status": null,  
    "latest_opt": null,  
    "release_plan_id": null  
  },  
],  
"swimlane_count": 2,  
"engine_ids": [  
  "d68a5ca7-b5b6-4917-a464-0adef94b8067"  
]  
}
```

## 状态码

状态码	描述
200	操作成功
400	错误的请求
404	请求对象不存在
500	内部错误

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.7 泳道

### 4.7.1 泳道组下创建泳道

#### 功能介绍

此API用于在泳道组下创建泳道。

#### URI

POST /v3/{project\_id}/cas/swimlane-group/{lane\_group\_id}/swimlane

表 4-379 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
lane_group_id	是	String	泳道组ID。获取方法，请参考 <a href="#">6.4.2-获取所有泳道组</a> 。

## 请求消息

表 4-380 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

表 4-381 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	泳道的名称。 由大小写英文字母、数字、中划线(-)、下划线(_)组成，并以大小写英文字母开头，大小写英文字母或者数字结尾，长度为2~64个字符。
tag	是	String	泳道的标签。
type	是	String	泳道的类型。当泳道组下没有泳道时，必须先创建基线泳道；当泳道组下已创建基线泳道时，只能创建灰度泳道。 <ul style="list-style-type: none"><li>• BASE，基线泳道。</li><li>• GRAY，灰度泳道。</li></ul>

## 响应消息

表 4-382 响应参数

参数	参数类型	描述
id	String	泳道ID。
name	String	泳道名称。
swimlane_group_id	String	所属的泳道组ID。

参数	参数类型	描述
type	String	泳道类型。 <ul style="list-style-type: none"><li>● BASE: 基线泳道。</li><li>● GRAY: 灰度泳道。</li></ul>
tag	String	泳道标签, 根据标签来控制流量的走向。
route_status	String	泳道路由状态。 <ul style="list-style-type: none"><li>● ENABLE: 开启路由。</li><li>● DISABLE: 关闭路由。</li></ul>
rule_match_mode	String	路由规则匹配模式。 当泳道组灰度类型为基于内容, 即 gray_release_type 为 CONTENT 时生效。 <ul style="list-style-type: none"><li>● ALL: 匹配所有的规则。</li><li>● ANY: 匹配任一规则。</li></ul>
rules	Array of objects	路由规则。当泳道组灰度类型为基于内容, 即 gray_release_type 为 CONTENT 时生效。请参考表 4-383。
weight	Integer	泳道流量权重, 范围在 0-100 之间。当泳道组灰度类型为基于流量, 即 gray_release_type 为 WEIGHT 时生效。
project_id	String	项目 ID。
create_time	Integer	创建时间。
update_time	Integer	更新时间。
creator	String	创建人。

表 4-383 rules

参数	参数类型	描述
type	String	灰度匹配规则类型。当前只支持 HEADER, 即基于 Header 头进行匹配。
key	String	键。
value	String	值。
condition	String	匹配条件。 <ul style="list-style-type: none"><li>● EXACT: 精确匹配。</li><li>● PREFIX: 前缀匹配。</li><li>● REGEX: 正则匹配。</li></ul>

## 请求示例

在泳道组中创建名为gray的灰度泳道。

```
{
  "name": "gray",
  "type": "GRAY",
  "tag": "gray"
}
```

## 响应示例

```
{
  "id": "9974a8b3-cbd7-4084-bac9-194f7d5ba3da",
  "name": "gray",
  "swimlane_group_id": "b76dad96-66cf-4ae7-aa55-11b6d630e777",
  "type": "GRAY",
  "tag": "gray",
  "weight": 0,
  "route_status": "DISABLED",
  "rule_match_mode": null,
  "project_id": "4008a952b3f44b5a919c9a48d90811f3",
  "rules": null,
  "creator": "test_user",
  "create_time": 1722928561358,
  "update_time": 1722928561358
}
```

## 状态码

状态码	描述
200	操作成功
400	错误的请求
404	请求对象不存在
500	内部错误

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.7.2 根据微服务引擎 ID 获取所有泳道

### 功能介绍

此API用于根据微服务引擎ID获取所有泳道。

### URI

GET /v3/{project\_id}/cas/swimlane

表 4-384 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。

表 4-385 查询参数

参数	是否必选	参数类型	描述
engine_id	是	String	微服务引擎ID。获取方法，请参考 <a href="#">8.2.2.2-查询微服务引擎专享版列表</a> 。

## 请求消息

表 4-386 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

## 响应消息

表 4-387 响应参数

参数	参数类型	描述
count	Integer	泳道总数。
swimlanes	Array of objects	泳道列表，请参考 <a href="#">表4-388</a> 。

表 4-388 swimlanes

参数	参数类型	描述
id	String	泳道ID。
name	String	泳道名称。

参数	参数类型	描述
swimlane_group_id	String	所属的泳道组ID。
type	String	泳道类型。 <ul style="list-style-type: none"><li>• BASE: 基线泳道。</li><li>• GRAY: 灰度泳道。</li></ul>
tag	String	泳道标签, 根据标签来控制流量的走向。
route_status	String	泳道路由状态。 <ul style="list-style-type: none"><li>• ENABLE: 开启路由。</li><li>• DISABLED: 关闭路由。</li></ul>
rule_match_mode	String	路由规则匹配模式。当泳道组灰度类型为基于内容, 即gray_release_type为CONTENT时生效。 <ul style="list-style-type: none"><li>• ALL: 匹配所有的规则。</li><li>• ANY: 匹配任一规则。</li></ul>
rules	Array of objects	路由规则。当泳道组灰度类型为基于内容, 即gray_release_type为CONTENT时生效。请参考 <a href="#">表 4-389</a> 。
weight	Integer	泳道流量权重。范围在0-100之间。当泳道组灰度类型为基于流量, 即gray_release_type为WEIGHT时生效。
project_id	String	项目ID。
create_time	Integer	创建时间。
update_time	Integer	更新时间。
creator	String	创建人。

表 4-389 rules

参数	参数类型	描述
type	String	灰度匹配规则类型。当前只支持HEADER, 即基于Header头进行匹配。
key	String	键。
value	String	值。
condition	String	匹配条件。 <ul style="list-style-type: none"><li>• EXACT: 精确匹配。</li><li>• PREFIX: 前缀匹配。</li><li>• REGEX: 正则匹配。</li></ul>

## 请求示例

无

## 响应示例

```
{
  "swimlanes": [
    {
      "id": "8c6274b9-113e-4066-9b1c-320fe5ae327b",
      "name": "baseline",
      "swimlane_group_id": "13c73b95-ec0f-4e11-8901-d1e3c9142d75",
      "type": "BASE",
      "tag": "base",
      "weight": 100,
      "route_status": "ENABLE",
      "swimlane_group": {
        "routeInGroupTags": null,
        "gateways": [
          {
            "id": "6b44bcf9-99e9-46f4-9ab5-7670456073fc",
            "name": "microGateway-pyy-0127-1",
            "target_services": [
              {
                "id": "e882070c-7d02-4db1-8b84-619ee88f1a25",
                "host": "unit-provider-all.wulumuqi.servicecomb",
                "name": "unit-provider-all"
              }
            ]
          }
        ],
        "engine_id": "b4032a08-b09d-47e4-a01f-596906df6f55"
      }
    },
    {
      "id": "13c73b95-ec0f-4e11-8901-d1e3c9142d75",
      "name": "test-wqd",
      "gray_release_type": "WEIGHT",
      "description": "111"
    }
  ],
  "count": 1
}
```

## 状态码

状态码	描述
200	操作成功
400	错误的请求
404	请求对象不存在
500	内部错误

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。



## 4.7.3 获取泳道组下所有泳道

### 功能介绍

此API用于获取泳道组下所有泳道。

### URI

GET /v3/{project\_id}/cas/swimlane-group/{lane\_group\_id}/swimlane

表 4-390 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
lane_group_id	是	String	泳道组ID。获取方法，请参考 <a href="#">6.4.2-获取所有泳道组</a> 。

### 请求消息

表 4-391 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

### 响应消息

表 4-392 响应参数

参数	参数类型	描述
swimlanes	Array of objects	泳道列表，请参考 <a href="#">表4-393</a> 。
count	Integer	泳道列表数量。

表 4-393 swimlanes

参数	参数类型	描述
id	String	泳道ID。
name	String	泳道名称。
swimlane_group_id	String	所属的泳道组ID。
type	String	泳道类型。 <ul style="list-style-type: none"><li>• BASE: 基线泳道。</li><li>• GRAY: 灰度泳道。</li></ul>
tag	String	泳道标签, 根据标签来控制流量的走向。
route_status	String	泳道路由状态。 <ul style="list-style-type: none"><li>• ENABLE: 开启路由。</li><li>• DISABLED: 关闭路由。</li></ul>
rule_match_mode	String	路由规则匹配模式。 当泳道组灰度类型为基于内容, 即 gray_release_type 为 CONTENT 时生效。 <ul style="list-style-type: none"><li>• ALL: 匹配所有的规则。</li><li>• ANY: 匹配任一规则。</li></ul>
rules	Array of objects	路由规则。当泳道组灰度类型为基于内容, 即 gray_release_type 为 CONTENT 时生效。请参考表 4-394。
weight	Integer	泳道流量权重。范围在0-100之间。 当泳道组灰度类型为基于流量, 即 gray_release_type 为 WEIGHT 时生效。
project_id	String	项目ID。
create_time	Integer	创建时间。
update_time	Integer	更新时间。
creator	String	创建人。

表 4-394 rules

参数	参数类型	描述
type	String	灰度匹配规则类型。当前只支持HEADER, 即基于Header头进行匹配。
key	String	键。
value	String	值。

参数	参数类型	描述
condition	String	匹配条件。 <ul style="list-style-type: none"><li>● EXACT: 精确匹配。</li><li>● PREFIX: 前缀匹配。</li><li>● REGEX: 正则匹配。</li></ul>

## 请求示例

无

## 响应示例

```
{
  "swimlanes": [
    {
      "id": "9974a8b3-cbd7-4084-bac9-194f7d5ba3da",
      "name": "gray",
      "swimlane_group_id": "b76dad96-66cf-4ae7-aa55-11b6d630e777",
      "type": "GRAY",
      "tag": "gray",
      "weight": 0,
      "route_status": "DISABLED",
      "rule_match_mode": null,
      "project_id": "4008a952b3f44b5a919c9a48d90811f3",
      "rules": null,
      "creator": "test_user",
      "create_time": 1722928561358,
      "update_time": 1722928561358
    }
  ],
  "count": 1
}
```

## 状态码

状态码	描述
200	操作成功
400	错误的请求
404	请求对象不存在
500	内部错误

## 错误码

错误码格式为: SVCSTG.00100.[Error\_ID], 例如: SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.7.4 根据泳道 ID 获取泳道信息

### 功能介绍

此API用于根据泳道ID获取泳道信息。

### URI

GET /v3/{project\_id}/cas/swimlane-group/{lane\_group\_id}/swimlane/{lane\_id}

表 4-395 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
lane_group_id	是	String	泳道组ID。获取方法，请参考 <a href="#">6.4.2-获取所有泳道组</a> 。
lane_id	是	String	泳道ID。获取方法，请参考 <a href="#">6.5.2-获取泳道组下所有泳道</a> 。

### 请求消息

表 4-396 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

### 响应消息

表 4-397 响应参数

参数	参数类型	描述
id	String	泳道ID。
name	String	泳道名称。
swimlane_group_id	String	所属的泳道组ID。

参数	参数类型	描述
type	String	泳道类型。 <ul style="list-style-type: none"><li>• BASE: 基线泳道。</li><li>• GRAY: 灰度泳道。</li></ul>
tag	String	泳道标签, 根据标签来控制流量的走向。
route_status	String	泳道路由状态。 <ul style="list-style-type: none"><li>• ENABLE: 开启路由。</li><li>• DISABLED: 关闭路由。</li></ul>
rule_match_mode	String	路由规则匹配模式。 当泳道组灰度类型为基于内容, 即 gray_release_type 为 CONTENT 时生效。 <ul style="list-style-type: none"><li>• ALL: 匹配所有的规则。</li><li>• ANY: 匹配任一规则。</li></ul>
rules	Array of objects	路由规则。当泳道组灰度类型为基于内容, 即 gray_release_type 为 CONTENT 时生效。请参考表4-398。
weight	Integer	泳道流量权重, 范围在0-100之间。当泳道组灰度类型为基于流量, 即 gray_release_type 为 WEIGHT 时生效。
instances	Array of objects	泳道中纳管的组件列表, 请参考表4-399。
instance_count	Integer	泳道中纳管的组件个数。
latest_opt	String	泳道上一步的操作。 <ul style="list-style-type: none"><li>• CLONE: 克隆。</li><li>• UPGRADE: 升级。</li><li>• ROLLBACK: 回滚。</li><li>• CREATE: 创建。</li></ul>
release_plan_id	String	泳道上一步操作关联的发布单ID。
project_id	String	项目ID。
create_time	Integer	创建时间。
update_time	Integer	更新时间。
creator	String	创建人。

表 4-398 rules

参数	参数类型	描述
type	String	灰度匹配规则类型。当前只支持 HEADER，即基于Header头进行匹配。
key	String	键。
value	String	值。
condition	String	匹配条件。 <ul style="list-style-type: none"><li>• EXACT：精确匹配。</li><li>• PREFIX：前缀匹配。</li><li>• REGEX：正则匹配。</li></ul>

表 4-399 instances

参数	参数类型	描述
id	String	应用组件实例ID。
name	String	应用组件实例名称。
version	String	应用组件版本号。
application_id	String	应用ID。
component_id	String	组件ID。
application_name	String	应用名称。
status	String	实例状态。
engine_id	String	组件关联引擎ID。
runtime_stack	Object	运行时，请参考 <a href="#">表4-400</a> 。
replica	Integer	实例副本数。

表 4-400 runtime\_stack

参数	参数类型	描述
name	String	技术栈名称。
type	String	技术栈类型，支持Java、Tomcat、Nodejs、Php、Docker、Python。 当部署模式为虚拟机部署时，仅支持Java、Tomcat、Nodejs；容器部署上述类型都支持。
version	String	技术栈版本。

参数	参数类型	描述
deploy_mode	String	部署模式。 <ul style="list-style-type: none"><li>container, 容器部署。</li><li>virtualmachine, 虚拟机部署。</li></ul>

### 请求示例

无

### 响应示例

```
{
  "id": "8f8a5b33-ee1f-4a70-b3a9-c1b179a69fd2",
  "name": "base",
  "swimlane_group_id": "5447f910-ab74-4d15-b2ab-dd0945900ce9",
  "type": "BASE",
  "tag": "base",
  "route_status": "ENABLE",
  "rule_match_mode": null,
  "project_id": "86a1f7b663b349e89071a3090d9d81f3",
  "rules": null,
  "creator": "test_user",
  "create_time": 1722848532908,
  "update_time": 1722848532908,
  "instances": [
    {
      "componentId": "0acb85e6-4d17-4aed-9184-9c418ebf50c1",
      "id": "8d2665d9-2420-42a4-8c30-8f4f10b75edd",
      "name": "comp-qor4hy12",
      "version": "1.0.1",
      "application_id": "520dc8fd-a333-402f-bef7-4412ced5a452",
      "application_name": "app-apitest-cgWi",
      "status": "RUNNING",
      "runtime_stack": {
        "type": "Tomcat",
        "name": "Tomcat8/OpenJDK8",
        "version": null,
        "status": null,
        "release_note": null,
        "spec": null,
        "parameters": null
      },
      "replica": 1
    },
    {
      "componentId": "0b582884-1ad0-4183-8d6d-16998f54818f",
      "id": "887f99d7-4ce5-431d-924f-6a328cb05572",
      "name": "comp-uh59l7s2",
      "version": "1.0.0",
      "application_id": "70502782-4f93-4eda-af19-415bac2d12ae",
      "application_name": "app-apitest-cgWi",
      "status": "RUNNING",
      "replica": 1
    }
  ],
  "instance_count": 2,
  "weight": 100,
  "batch_status": null,
  "latest_opt": null,
  "release_plan_id": null
}
```

## 状态码

状态码	描述
200	操作成功
400	错误的请求
404	请求对象不存在
500	内部错误

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.7.5 根据泳道 ID 修改泳道信息

### 功能介绍

此API用于根据泳道ID修改泳道信息。

泳道需未执行过[根据泳道ID关联组件](#)操作。

### URI

PUT /v3/{project\_id}/cas/swimlane-group/{lane\_group\_id}/swimlane/{lane\_id}

表 4-401 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
lane_group_id	是	String	泳道组ID。获取方法，请参考 <a href="#">6.4.2-获取所有泳道组</a> 。
lane_id	是	String	泳道ID。获取方法，请参考 <a href="#">6.5.2-获取泳道组下所有泳道</a> 。



表 4-402 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	泳道的名称。 由大小写英文字母、数字、中划线 (-)、下划线 (_) 组成, 并以大小写英文字母开头, 大小写英文字母或者数字结尾, 长度为2~64个字符。
tag	是	String	泳道的标签。
type	是	String	泳道的类型。当泳道组下没有泳道时, 必须先创建基线泳道; 当泳道组下已创建基线泳道时, 只能创建灰度泳道。 <ul style="list-style-type: none"><li>• BASE, 基线泳道。</li><li>• GRAY, 灰度泳道。</li></ul>

## 请求消息

表 4-403 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型 (格式), 默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种, 如果您使用的Token方式, 此参数为必填, 请填写Token的值。Token获取方式, 请参考 <a href="#">获取用户Token</a> 。

## 响应消息

表 4-404 响应参数

参数	参数类型	描述
id	String	泳道ID。
name	String	泳道名称。
swimlane_group_id	String	所属的泳道组ID。

参数	参数类型	描述
type	String	泳道类型。 <ul style="list-style-type: none"><li>● BASE: 基线泳道。</li><li>● GRAY: 灰度泳道。</li></ul>
tag	String	泳道标签, 根据标签来控制流量的走向。
route_status	String	泳道路由状态。 <ul style="list-style-type: none"><li>● ENABLE: 开启路由。</li><li>● DISABLED: 关闭路由。</li></ul>
rule_match_mode	String	路由规则匹配模式。 当泳道组灰度类型为基于内容, 即 gray_release_type 为 CONTENT 时生效。 <ul style="list-style-type: none"><li>● ALL: 匹配所有的规则。</li><li>● ANY: 匹配任一规则。</li></ul>
rules	Array of objects	路由规则。当泳道组灰度类型为基于内容, 即 gray_release_type 为 CONTENT 时生效。请参考表 4-405。
weight	Integer	泳道流量权重, 范围在 0-100 之间。当泳道组灰度类型为基于流量, 即 gray_release_type 为 WEIGHT 时生效。
project_id	String	项目 ID。
create_time	Integer	创建时间。
update_time	Integer	更新时间。
creator	String	创建人。

表 4-405 rules

参数	参数类型	描述
type	String	灰度匹配规则类型。当前只支持 HEADER, 即基于 Header 头进行匹配。
key	String	键。
value	String	值。
condition	String	匹配条件。 <ul style="list-style-type: none"><li>● EXACT: 精确匹配。</li><li>● PREFIX: 前缀匹配。</li><li>● REGEX: 正则匹配。</li></ul>

## 请求示例

修改Id为9974a8b3-cbd7-4084-bac9-194f7d5ba3da的灰度泳道名为gray。

```
{
  "name": "gray",
  "type": "GRAY",
  "tag": "base-lane"
}
```

## 响应示例

```
{
  "id": "9974a8b3-cbd7-4084-bac9-194f7d5ba3da",
  "name": "gray",
  "swimlane_group_id": "b76dad96-66cf-4ae7-aa55-11b6d630e777",
  "type": "GRAY",
  "tag": "gray",
  "weight": 0,
  "route_status": "DISABLED",
  "rule_match_mode": null,
  "project_id": "4008a952b3f44b5a919c9a48d90811f3",
  "rules": null,
  "creator": "test_user",
  "create_time": 1722928561358,
  "update_time": 1722928561358
}
```

## 状态码

状态码	描述
200	操作成功
400	错误的请求
404	请求对象不存在
500	内部错误

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.7.6 根据泳道 ID 删除泳道

### 功能介绍

此API用于根据泳道ID删除泳道。

泳道需未执行过[根据泳道ID关联组件](#)操作。

### URI

DELETE /v3/{project\_id}/cas/swimlane-group/{lane\_group\_id}/swimlane/{lane\_id}

表 4-406 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
lane_group_id	是	String	泳道组ID。获取方法，请参考 <a href="#">6.4.2-获取所有泳道组</a> 。
lane_id	是	String	泳道ID。获取方法，请参考 <a href="#">6.5.2-获取泳道组下所有泳道</a> 。

## 请求消息

表 4-407 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

## 响应消息

无

## 请求示例

无

## 响应示例

无

## 状态码

状态码	描述
200	操作成功
400	错误的请求
404	请求对象不存在
500	内部错误

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.7.7 根据泳道 ID 控制泳道路由开关

### 功能介绍

此API用于根据泳道ID控制泳道路由开关。

### URI

POST /v3/{project\_id}/cas/swimlane-group/{lane\_group\_id}/swimlane/{lane\_id}/action

表 4-408 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
lane_group_id	是	String	泳道组ID。获取方法，请参考 <a href="#">6.4.2-获取所有泳道组</a> 。
lane_id	是	String	泳道ID。获取方法，请参考 <a href="#">6.5.2-获取泳道组下所有泳道</a> 。

### 请求消息

表 4-409 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

表 4-410 请求 Body 参数

参数	是否必选	参数类型	描述
route_status	是	String	泳道路由状态。 <ul style="list-style-type: none"><li>• ENABLE: 开启路由。</li><li>• DISABLED: 关闭路由。</li></ul>

## 响应消息

无

## 请求示例

关闭泳道路由。

```
{  
  "route_status": "DISABLED"  
}
```

## 响应示例

无

## 状态码

状态码	描述
200	操作成功
400	错误的请求
404	请求对象不存在
500	内部错误

## 错误码

错误码格式为: SVCSTG.00100.[Error\_ID], 例如: SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.7.8 根据泳道 ID 管理泳道纳管组件

### 功能介绍

此API用于根据泳道ID管理泳道纳管组件。

## URI

POST /v3/{project\_id}/cas/swimlane-group/{lane\_group\_id}/swimlane/{lane\_id}/components-action

表 4-411 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
lane_group_id	是	String	泳道组ID。获取方法，请参考 <a href="#">获取所有泳道组</a> 。
lane_id	是	String	泳道ID。获取方法，请参考 <a href="#">获取泳道组下所有泳道</a> 。

## 请求消息

表 4-412 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">认证鉴权</a> 。

表 4-413 请求 Body 参数

参数	是否必选	参数类型	描述
action	是	String	泳道组件实例纳管移除操作。 <ul style="list-style-type: none"><li>• ADD：纳管组件。</li><li>• REMOVE：移除组件。</li></ul>
instance_ids	是	Array of String	组件实例列表。

## 响应消息

无

## 请求示例

将ID为887f99d7-4ce5-431d-924f-6a328cb05572的组件实例纳管到泳道中。

```
{
  "instance_ids": [
    "887f99d7-4ce5-431d-924f-6a328cb05572"
  ],
  "action": "ADD"
}
```

## 响应示例

```
{
  "id": "8f8a5b33-ee1f-4a70-b3a9-c1b179a69fd2",
  "name": "base",
  "swimlane_group_id": "5447f910-ab74-4d15-b2ab-dd0945900ce9",
  "type": "BASE",
  "tag": "base",
  "route_status": "ENABLE",
  "rule_match_mode": null,
  "project_id": "86a1f7b663b349e89071a3090d9d81f3",
  "rules": null,
  "creator": "w00842292",
  "create_time": 1722848532908,
  "update_time": 1722848532908,
  "instances": [
    {
      "componentId": "0b582884-1ad0-4183-8d6d-16998f54818f",
      "id": "887f99d7-4ce5-431d-924f-6a328cb05572",
      "name": "comp-uh59l7s2",
      "version": "1.0.0",
      "application_id": "70502782-4f93-4eda-af19-415bac2d12ae",
      "application_name": "app-apitest-1o6l",
      "replica": 0
    }
  ],
  "instance_count": 1,
  "weight": 100,
  "batch_status": null,
  "latest_opt": null,
  "release_plan_id": null
}
```

## 状态码

状态码	描述
200	操作成功
400	错误的请求
404	请求对象不存在
500	内部错误

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。



## 4.7.9 根据泳道 ID 批量操作泳道组件

### 功能介绍

此API用于泳道ID批量操作泳道组件。

### URI

POST /v3/{project\_id}/cas/swimlane-group/{lane\_group\_id}/swimlane/{lane\_id}/components-instances

表 4-414 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
lane_group_id	是	String	泳道组ID。获取方法，请参考 <a href="#">获取所有泳道组</a> 。
lane_id	是	String	泳道ID。获取方法，请参考 <a href="#">获取泳道组下所有泳道</a> 。

### 请求消息

表 4-415 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">认证鉴权</a> 。

表 4-416 请求 Body 参数

参数	是否必选	参数类型	描述
action	是	String	泳道组件批量操作，当前仅支持delete操作。
parameters	是	Map<String, Object>	配置参数。

表 4-417 parameters

参数	是否必选	参数类型	描述
components	是	Array of Objects	组件列表, 请参考 <a href="#">表4-418</a> 。

表 4-418 components

参数	是否必选	参数类型	描述
id	是	String	组件实例ID。
application_id	是	String	组件应用ID。

## 响应消息

无

## 请求示例

将ID为30f88521-fa43-42ff-aaec-9cac4259fb16的组件实例从泳道中删除。

```
{
  "parameters": {
    "components": [
      {
        "id": "30f88521-fa43-42ff-aaec-9cac4259fb16",
        "application_id": "f397be2a-4c49-4ed8-8144-ce13417e179d"
      }
    ]
  },
  "action": "delete"
}
```

## 响应示例

无

## 状态码

状态码	描述
200	操作成功
400	错误的请求
404	请求对象不存在

状态码	描述
500	内部错误

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.7.10 根据泳道 ID 关联组件

### 功能介绍

此API用于根据泳道ID为泳道关联组件。

### URI

PUT /v3/{project\_id}/cas/swimlane-group/{lane\_group\_id}/swimlane/{lane\_id}/instances

表 4-419 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
lane_group_id	是	String	泳道组ID。获取方法，请参考 <a href="#">6.4.2-获取所有泳道组</a> 。
lane_id	是	String	泳道ID。获取方法，请参考 <a href="#">6.5.2-获取泳道组下所有泳道</a> 。

### 请求消息

表 4-420 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

表 4-421 请求 Body 参数

参数	是否必选	参数类型	描述
action	是	String	为泳道关联移除组件操作。 <ul style="list-style-type: none"><li>• ADD: 关联组件。</li><li>• REMOVE: 移除组件。</li></ul>
instance_ids	是	Array of String	组件实例列表。

## 响应消息

表 4-422 响应参数

参数	参数类型	描述
id	String	泳道ID。
name	String	泳道名称。
swimlane_group_id	String	所属的泳道组ID。
type	String	泳道类型。 <ul style="list-style-type: none"><li>• BASE: 基线泳道。</li><li>• GRAY: 灰度泳道。</li></ul>
tag	String	泳道标签, 根据标签来控制流量的走向。
route_status	String	泳道路由状态。 <ul style="list-style-type: none"><li>• ENABLE: 开启路由。</li><li>• DISABLED: 关闭路由。</li></ul>
rule_match_mode	String	路由规则匹配模式。 当泳道组灰度类型为基于内容, 即 gray_release_type 为 CONTENT 时生效。 <ul style="list-style-type: none"><li>• ALL: 匹配所有的规则。</li><li>• ANY: 匹配任一规则。</li></ul>
rules	Array of objects	路由规则。当泳道组灰度类型为基于内容, 即 gray_release_type 为 CONTENT 时生效。请参考表 4-423。
weight	Integer	泳道流量权重, 范围在0-100之间。当泳道组灰度类型为基于流量, 即 gray_release_type 为 WEIGHT 时生效。
instances	Array of objects	泳道中纳管的组件列表, 请参考表4-424。

参数	参数类型	描述
instance_count	Integer	泳道中纳管的组件个数。
latest_opt	String	泳道上一步的操作。 <ul style="list-style-type: none"> <li>• CLONE: 克隆。</li> <li>• UPGRADE: 升级。</li> <li>• ROLLBACK: 回滚。</li> <li>• CREATE: 创建。</li> </ul>
release_plan_id	String	泳道上一步操作关联的发布单ID。
project_id	String	项目ID。
create_time	Integer	创建时间。
update_time	Integer	更新时间。
creator	String	创建人。

表 4-423 rules

参数	参数类型	描述
type	String	灰度匹配规则类型。当前只支持HEADER，即基于Header头进行匹配。
key	String	键。
value	String	值。
condition	String	匹配条件。 <ul style="list-style-type: none"> <li>• EXACT: 精确匹配。</li> <li>• PREFIX: 前缀匹配。</li> <li>• REGEX: 正则匹配。</li> </ul>

表 4-424 instances

参数	参数类型	描述
id	String	应用组件实例ID。
name	String	应用组件实例名称。
version	String	应用组件版本号。
application_id	String	应用ID。
component_id	String	组件ID。

参数	参数类型	描述
application_name	String	应用名称。
replica	Integer	实例副本数。

## 请求示例

为id为8f8a5b33-ee1f-4a70-b3a9-c1b179a69fd2的泳道关联id为887f99d7-4ce5-431d-924f-6a328cb05572的组件实例。

```
{
  "instance_ids": [
    "887f99d7-4ce5-431d-924f-6a328cb05572"
  ],
  "action": "ADD"
}
```

## 响应示例

```
{
  "id": "8f8a5b33-ee1f-4a70-b3a9-c1b179a69fd2",
  "name": "base",
  "swimlane_group_id": "5447f910-ab74-4d15-b2ab-dd0945900ce9",
  "type": "BASE",
  "tag": "base",
  "route_status": "ENABLE",
  "rule_match_mode": null,
  "project_id": "86a1f7b663b349e89071a3090d9d81f3",
  "rules": null,
  "creator": "test_user",
  "create_time": 1722848532908,
  "update_time": 1722848532908,
  "instances": [
    {
      "componentId": "0b582884-1ad0-4183-8d6d-16998f54818f",
      "id": "887f99d7-4ce5-431d-924f-6a328cb05572",
      "name": "comp-uh59l7s2",
      "version": "1.0.0",
      "application_id": "70502782-4f93-4eda-af19-415bac2d12ae",
      "application_name": "app-apitest-1o6l",
      "replica": 0
    }
  ],
  "instance_count": 1,
  "weight": 100,
  "batch_status": null,
  "latest_opt": null,
  "release_plan_id": null
}
```

## 状态码

状态码	描述
200	操作成功
400	错误的请求
404	请求对象不存在

状态码	描述
500	内部错误

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.8 部署任务

### 4.8.1 获取部署任务详细信息

#### 功能介绍

此API用于获取部署任务详细信息。

#### 调试

您可以在[API Explorer](#)中调试该接口。

#### URI

GET /v3/{project\_id}/cas/jobs/{job\_id}

表 4-425 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
job_id	是	String	部署任务ID，为 <a href="#">应用中创建组件</a> 、 <a href="#">根据组件ID修改组件信息</a> 、 <a href="#">根据组件ID下发组件任务</a> 接口响应参数的job_id。

表 4-426 查询参数

参数	是否必选	参数类型	描述
limit	否	Integer	指定个数。
offset	否	Integer	指定偏移量。
desc	否	String	是否降序： true=desc, false=asc。

## 请求消息

表 4-427 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">认证鉴权</a> 。

## 响应消息

表 4-428 响应参数

参数	参数类型	描述
task_count	Integer	部署任务数量。
job	Object	构建工程参数，请参考 <a href="#">表 4-429</a> 。
tasks	Array<Object>	部署任务参数，请参考 <a href="#">表 4-430</a> 。



表 4-429 job

参数	参数类型	描述
EXECUTION_STATUS	String	执行状态。 <ul style="list-style-type: none"> <li>• SUBMITTED: 已提交但尚未处理。</li> <li>• SCHEDULED: 计划在未来执行的。</li> <li>• RUNNING: 当前正在处理。</li> <li>• FAILED: 处于失败状态。</li> <li>• SUCCEEDED: 处于成功状态。</li> <li>• CANCELLED: 已取消。</li> <li>• PARTIALLY_FAILED: 部分失败。</li> <li>• ROLLBACKING: 正在回滚。</li> <li>• ROLLBACKED: 已回滚。</li> <li>• UNKNOWN: 未知状态。</li> </ul>
JOB_ID	String	工作ID。
JOB_TYPE	String	类别。 <ul style="list-style-type: none"> <li>• PROVISION: 提供服务任务。</li> <li>• DEPROVISION: 取消提交服务任务。</li> <li>• DAILY_OPERATION: 日常操作任务。</li> <li>• SYSTEM_INNER: 系统内部任务。</li> <li>• UNKNOWN: 未知任务。</li> </ul>
ORDER_ID	String	排序ID。
PROJECT_ID	String	创建租户Project ID。
SERVICE_INSTANCE_ID	String	实例ID。

表 4-430 task

参数	参数类型	描述
CREATED_AT	String	创建时间。
MESSAGES	String	消息。

参数	参数类型	描述
TASK_ID	String	任务ID。
TASK_INDEX	Integer	任务序号。
TASK_NAME	String	任务名称。
TASK_STATUS	String	任务状态。 <ul style="list-style-type: none"><li>• SUBMITTED: 已提交但尚未处理。</li><li>• SCHEDULED: 计划在未来执行的。</li><li>• RUNNING: 当前正在处理。</li><li>• FAILED: 处于失败状态。</li><li>• SUCCEEDED: 处于成功状态。</li><li>• CANCELLED: 已取消。</li><li>• PARTIALLY_FAILED: 部分失败。</li><li>• SKIPPED: 已跳过。</li><li>• ROLLBACKING: 正在回滚。</li><li>• ROLLBACKED: 已回滚。</li><li>• UNKNOWN: 未知状态。</li></ul>
UPDATE_AT	String	更新时间。

### 请求示例

无

### 响应示例

```
{
  "job": {
    "execution_status": "SUCCEEDED",
    "job_id": "JOB17cb1778-07b1-4c8f-a183-c86cffbe8f9e",
    "job_type": "PROVISION"
  },
  "tasks": [
    {
      "task_id": "Task-af75dd60-0b04-4f4b-b669-0bad38fcd97",
      "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  
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在
500	内部错误

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

# 4.9 插件管理

## 4.9.1 创建插件

### 功能介绍

此API用于创建插件。

### URI

POST /v3/{project\_id}/cas/environments/{environment\_id}/addons

表 4-431 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
environment_id	是	String	环境ID。获取方法，请参考 <a href="#">获取所有环境</a> 。

## 请求消息

表 4-432 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">认证鉴权</a> 。

表 4-433 请求 Body 参数

参数	是否必选	参数类型	描述
cce_id	是	String	环境所绑定的CCE集群ID。获取方法，请参考 <a href="#">根据环境ID查询环境纳管的资源</a> 。
cse_id	否	String	环境绑定的微服务引擎ID。获取方法，请参考 <a href="#">根据环境ID查询环境纳管的资源</a> 。 当name为“sermantInjector”时必选。
name	是	String	插件名称。 sermantInjector: Sermant Injector插件。用于自动挂载Sermant Agent，通过Sermant Agent接入未开启安全认证的ServiceComb引擎。无需修改应用代码即可接入引擎并使用应用注册发现、全链路灰度发布、优雅上下线、标签路由等功能，但是不支持使用微服务治理功能。

## 响应消息

表 4-434 响应参数

参数	参数类型	描述
job_id	String	任务ID。
id	String	插件ID。

## 请求示例

创建Sermant Injector插件，并为插件管理id为b4e49ad5-5142-437d-8a6b-1de66b1af0db的微服务引擎和id为02f9a523-cefe-11ee-afc1-0255ac100045的CCE集群。

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

## 响应示例

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

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在
500	内部错误

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.9.2 查询插件列表

### 功能介绍

此API用于查询环境下的插件列表。

### URI

GET /v3/{project\_id}/cas/environments/{environment\_id}/addons

表 4-435 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
environment_id	是	String	环境ID，为 <a href="#">创建环境</a> 接口响应参数的id。

## 请求消息

表 4-436 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">认证鉴权</a> 。

## 响应消息

表 4-437 响应参数

参数	参数类型	描述
addons	Arrays of objects	插件列表，请参考 <a href="#">表4-438</a> 。

表 4-438 addons

参数	参数类型	描述
cce_id	String	插件所绑定的CCE集群ID。
cse_id	String	插件所绑定的CSE微服务引擎ID。
environment_id	String	插件所在的环境ID。
id	String	插件ID
job_id	String	插件执行的最后一个任务的ID。
name	String	插件名称。 sermantInjector: Sermant Injector插件。用于自动挂载 Sermant Agent，通过Sermant Agent接入未开启安全认证的 ServiceComb引擎。无需修改应用代码即可接入引擎并使用应用注册发现、全链路灰度发布、优雅上下线、标签路由等功能，但是不支持使用微服务治理功能。

参数	参数类型	描述
status	String	插件状态。 <ul style="list-style-type: none"><li>INITIALIZING, 初始化。</li><li>UPGRADING, 升级中。</li><li>FAILED, 失败。</li><li>SUCCEEDED, 成功。</li><li>DELETING, 删除中。</li></ul>
version	String	插件版本。

### 请求示例

无

### 响应示例

```
{  "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"    }  ]}
```

### 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在
500	内部错误

### 错误码

错误码格式为: SVCSTG.00100.[Error\_ID], 例如: SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.9.3 查询插件元数据信息

### 功能介绍

此API用于查询环境下插件的元数据信息。

### URI

GET /v3/{project\_id}/cas/environments/{environment\_id}/addons-metadata

表 4-439 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
environment_id	是	String	环境ID。获取方法，请参考 <a href="#">获取所有环境</a> 。

### 请求消息

表 4-440 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">认证鉴权</a> 。

### 响应消息

表 4-441 响应参数

参数	参数类型	描述
addons	Arrays of objects	插件版本信息，请参考 <a href="#">表 4-442</a> 。



表 4-442 addons

参数	参数类型	描述
name	String	插件名称。 sermantInjector: Sermant Injector插件。用于自动挂载 Sermant Agent，通过Sermant Agent接入未开启安全认证的 ServiceComb引擎。无需修改应用代码即可接入引擎并使用应用注册发现、全链路灰度发布、优雅上下线、标签路由等功能，但是不支持使用微服务治理功能。
latest_version	String	插件最新版本。

### 请求示例

无

### 响应示例

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

### 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在
500	内部错误

### 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.9.4 根据插件 ID 查询插件详情

### 功能介绍

此API用于查询环境下指定插件详情。

### URI

GET /v3/{project\_id}/cas/environments/{environment\_id}/addons/{addon\_id}

表 4-443 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
environm ent_id	是	String	环境ID。获取方法，请参考 <a href="#">获取所有环境</a> 。
addon_id	是	String	插件ID。获取方法，请参考 <a href="#">查询插件列表</a> 。

### 请求消息

表 4-444 请求 Header 参数

参数	是否必选	参数类型	描述
Content- Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth- Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">认证鉴权</a> 。

### 响应消息

表 4-445 响应参数

参数	参数类型	描述
addons	Arrays of objects	插件版本信息，请参考 <a href="#">表 4-446</a> 。

表 4-446 addons

参数	参数类型	描述
id	String	插件ID。
name	String	插件名称。 sermantInjector: Sermant Injector插件。用于自动挂载Sermant Agent, 通过Sermant Agent接入未开启安全认证的ServiceComb引擎。无需修改应用代码即可接入引擎并使用应用注册发现、全链路灰度发布、优雅上下线、标签路由等功能, 但是不支持使用微服务治理功能。
version	String	插件版本。
cce_id	String	插件安装所在的CCE集群ID。
cse_id	String	插件安装所绑定的CSE微服务引擎ID。
job_id	String	插件执行的任务ID。
environment_id	String	插件所在的环境ID。
status	String	插件状态。 <ul style="list-style-type: none"><li>INITIALIZING: 初始化。</li><li>UPGRADING: 升级中。</li><li>FAILED: 失败。</li><li>SUCCEEDED: 成功。</li><li>DELETING: 删除中。</li></ul>

## 请求示例

无

## 响应示例

```
{
  "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"  
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在
500	内部错误

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.9.5 根据插件 ID 下发插件任务

### 功能介绍

此API用于根据插件ID下发插件任务。

### URI

POST /v3/{project\_id}/cas/environments/{environment\_id}/addons/{addon\_id}/  
action

表 4-447 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
environment_id	是	String	环境ID。获取方法，请参考 <a href="#">获取所有环境</a> 。
addon_id	是	String	插件ID。获取方法，请参考 <a href="#">查询插件列表</a> 。

### 请求消息

表 4-448 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">认证鉴权</a> 。

表 4-449 请求 Body 参数

参数	是否必选	参数类型	描述
action	是	String	插件任务类型。 <ul style="list-style-type: none"> <li>restart: 重新启动。</li> <li>redeploy: 重新部署。</li> </ul>

## 响应消息

表 4-450 响应参数

参数	参数类型	描述
job_id	String	任务ID。

## 请求示例

下发重新部署插件任务。

```
{
  "action": "redeploy"
}
```

## 响应示例

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

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在
500	内部错误

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 4.9.6 根据插件 ID 删除插件

### 功能介绍

此API用于根据插件ID删除插件。

### URI

DELETE /v3/{project\_id}/cas/environments/{environment\_id}/addons/{addon\_id}

表 4-451 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
environment_id	是	String	环境ID。获取方法，请参考 <a href="#">获取所有环境</a> 。
addon_id	是	String	插件ID。获取方法，请参考 <a href="#">查询插件列表</a> 。

### 请求消息

表 4-452 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">认证鉴权</a> 。

### 响应消息

表 4-453 响应参数

参数	参数类型	描述
job_id	String	任务ID。

## 请求示例

无

## 响应示例

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

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在
500	内部错误

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

# 5 访问 Git 仓库 API

## 5.1 获取仓库授权列表

### 功能介绍

获取Git仓库所有授权信息。

### 调试

您可以在[API Explorer](#)中调试该接口。

### URI

GET /v1/{project\_id}/git/auths

参数说明见[表5-1](#)。

表 5-1 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。

### 请求消息

表 5-2 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。



参数	是否必选	参数类型	描述
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

## 响应消息

表 5-3 响应参数

参数	参数类型	描述
authorizations	Array<Object>	授权结构体，请参考 <a href="#">表5-4</a> 。
count	Integer	仓库授权数量。

表 5-4 authorizations

参数	参数类型	描述
name	String	仓库授权的名称。
repo_type	String	仓库类型。 取值范围：CodeArts、Gitee、GitHub、BitBucket、GitLab。
repo_host	String	仓库地址。
repo_home	String	您的仓库主页。
repo_user	String	您的仓库用户名。
avatar	String	您的仓库用户头像。
token_type	String	仓库授权方式。
create_time	Long	仓库授权创建时间戳。
update_time	Long	仓库授权修改时间戳。
status	Integer	仓库授权状态。 0：正常 1：过期 2：未认证
tag	String	授权环境的类型，默认为null。

## 请求示例

无

## 响应示例

```
{
  "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
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。

## 错误码

错误码格式为：SVCSTG.REPO.[Error\_ID]，例如：SVCSTG.REPO.0401。错误码说明请参考[ServiceStage错误码](#)。

## 5.2 获取授权重定向 URL

### 功能介绍

获取授权重定向URL。

### 调试

您可以在[API Explorer](#)中调试该接口。

### URI

GET /v1/{project\_id}/git/auths/{repo\_type}/redirect

参数说明见[表5-5](#)。

表 5-5 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
repo_type	是	String	仓库类型。 取值范围：CodeArts、Gitee、GitHub、BitBucket、GitLab。

表 5-6 查询参数

参数	是否必选	参数类型	描述
tag	否	String	站点标签。

## 请求消息

表 5-7 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

## 响应消息

表 5-8 响应参数

参数	参数类型	描述
url	String	授权重定向URL。

## 请求示例

无

## 响应示例

```
{
  "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%2FM0yng0euuGH%2FJNWJl3FeSVZhTUVj%2FF7r52mqpUKcYXz7f6.****&scope=read%3Auser%20repo%20write%3Arepo_hook"
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。

## 错误码

错误码格式为：SVCSTG.REPO.[Error\_ID]，例如：SVCSTG.REPO.0401。错误码说明请参考[ServiceStage错误码](#)。

## 5.3 创建 OAuth 授权

### 功能介绍

创建OAuth授权。

### 调试

您可以在[API Explorer](#)中调试该接口。

### URI

POST /v1/{project\_id}/git/auths/{repo\_type}/oauth

参数说明见[表5-9](#)。

表 5-9 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
repo_type	是	String	仓库类型。 取值范围：CodeArts、Gitee、GitHub、BitBucket、GitLab。

表 5-10 查询参数

参数	是否必选	参数类型	描述
tag	否	String	站点标签。

## 请求消息

表 5-11 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

表 5-12 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	授权名称。
code	是	String	git仓库授权后，重定向返回的url里面的code参数。
state	是	String	git仓库授权后，一次性的认证编码和随机串。

## 说明

code和state参数值获取方法：

1. 在浏览器地址栏中输入访问[获取授权重定向URL](#)接口获取到的URL并回车。
2. 在登录页面输入您的代码仓库账号和密码。  
如果您已经登录代码仓库，则无需执行这一步。
3. 在华为云控制台登录页面输入账号和密码，单击“登录”。  
如果您已经登录华为云控制台，则无需执行这一步。
4. 等待页面显示新建授权失败后，复制地址栏的URL。

例如：

```
https://console.huaweicloud.com/servicestage/oauth?  
repo_type=github&code=04bed2df009f3fc9***c&state=%242a  
%2410%24.uQYThj5yKFs51w9s9ajeuQ0NSGshPH7IUfXoZLPlz7J%2FoAJ8B***
```

5. 对获取到的URL进行URL解码，获得解码后的URL。

示例如下：

```
https://console.huaweicloud.com/servicestage/oauth?  
repo_type=github&code=04bed2df009f3fc9***c&state=$2a  
$10$.uQYThj5yKFs51w9s9ajeuQ0NSGshPH7IUfXoZLPlz7J/oAJ8B***
```

获取其中code和state对应的参数值。

获取到的state参数值为一次性的认证编码和随机串，仅支持使用一次。如需再次调用本接口创建OAuth授权，请再次执行[获取授权重定向URL](#)获取URL并参考本方法获取code和state参数值。

## 响应消息

表 5-13 响应参数

参数	参数类型	描述
authorization	Object	授权结构体，请参考 <a href="#">表5-14</a> 。

表 5-14 authorization

参数	参数类型	描述
name	String	授权名称。
repo_type	String	仓库类型。 取值范围：github、gitlab、gitee、bitbucket。
repo_host	String	仓库地址。
repo_home	String	您的仓库主页。
repo_user	String	您的仓库用户名。
avartar	String	您的仓库头像。
token_type	String	仓库授权方式。

参数	参数类型	描述
create_time	Long	仓库授权创建时间戳。
update_time	Long	仓库授权修改时间戳。
status	Integer	仓库授权状态。 0: 正常 1: 过期 2: 未认证。

## 请求示例

创建名称为auth1的OAuth授权。

```
{
  "name": "auth1",
  "code": "04bed2df009f3fc9***c",
  "state": "$2a$10$.uQYThj5yKFs51w9s9ajeuQ0NSGshPH7IUfXoZLPlz7J/oAJ8B****"
}
```

## 响应示例

```
{
  "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
  }
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。

## 错误码

错误码格式为：SVCSTG.REPO.[Error\_ID]，例如：SVCSTG.REPO.0401。错误码说明请参考[ServiceStage错误码](#)。

## 5.4 创建私人令牌授权

### 功能介绍

创建私有令牌授权。

### 调试

您可以在[API Explorer](#)中调试该接口。

### URI

POST /v1/{project\_id}/git/auths/{repo\_type}/personal

参数说明见[表5-15](#)。

表 5-15 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
repo_type	是	String	仓库类型。 取值范围：github、gitlab、gitee。

### 请求消息

表 5-16 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。



表 5-17 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	授权名称。
token	是	String	仓库Token。
host	否	String	仓库地址。

## 响应消息

表 5-18 响应参数

参数	参数类型	描述
authorization	Object	授权结构体，请参考 <a href="#">表5-19</a> 。

表 5-19 authorization

参数	参数类型	描述
name	String	授权名称。
repo_type	String	仓库类型。 取值范围：github、gitlab、gitee。
repo_host	String	仓库地址。
repo_home	String	您的仓库主页。
repo_user	String	您的仓库用户名。
avatar	String	您的仓库用户头像。
token_type	String	仓库授权方式。
create_time	Long	仓库授权创建时间戳。
update_time	Long	仓库授权修改时间戳。
status	Integer	仓库授权状态。 0: 正常 1: 过期 2: 未认证

## 请求示例

创建名称为token1的私人令牌授权。

```
{  
  "name": "token1",  
  "token": "ghp_7QoYcNLfahSXSbhhuT8R5xHLKe*****6",  
  "host": "https://github.com/tom-repo"  
}
```

## 响应示例

```
{  
  "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  
  }  
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。

## 错误码

错误码格式为：SVCSTG.REPO.[Error\_ID]，例如：SVCSTG.REPO.0401。错误码说明请参考[ServiceStage错误码](#)。

## 5.5 创建口令授权

### 功能介绍

创建Git仓库口令授权。

### 调试

您可以在[API Explorer](#)中调试该接口。

### URI

POST /v1/{project\_id}/git/auths/{repo\_type}/password

参数说明见[表5-20](#)。

表 5-20 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
repo_type	是	String	仓库类型。 取值范围：codearts、bitbucket。

## 请求消息

表 5-21 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

表 5-22 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	授权名称。
user	是	String	仓库用户名。
password	是	String	仓库密码。

## 响应消息

表 5-23 响应参数

参数	参数类型	描述
authorization	Object	授权结构体，请参考 <a href="#">表5-24</a> 。

表 5-24 authorization

参数	参数类型	描述
name	String	授权名称。
repo_type	String	仓库类型。 取值范围：codearts、bitbucket。
repo_host	String	仓库地址。
repo_home	String	仓库主页。
repo_user	String	仓库用户名。
avartar	String	头像。
token_type	String	授权方式。
create_time	Long	创建时间。
update_time	Long	修改时间。
status	Integer	仓库授权状态。 0: 正常 1: 过期 2: 未认证

## 请求示例

创建名称为pass1的口令授权。

```
{  
  "name": "pass1",  
  "user": "bucket-ljn",  
  "password": "*****"  
}
```

## 响应示例

```
{  
  "authorization": {  
    "name": "pass1",  
    "repo_type": "bitbucket",  
    "repo_host": "https://api.bitbucket.org",  
    "repo_home": "https://bitbucket.org/%7B63cc78de-*****-9505c0335f96%7D/",  
    "repo_user": "bucket-ljn",  
    "avartar": "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  
  }  
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。

## 错误码

错误码格式为：SVCSTG.REPO.[Error\_ID]，例如：SVCSTG.REPO.0401。错误码说明请参考[ServiceStage错误码](#)。

# 5.6 删除仓库授权

## 功能介绍

通过名称删除仓库授权。

## 调试

您可以在[API Explorer](#)中调试该接口。

## URI

DELETE /v1/{project\_id}/git/auths/{name}

参数说明见[表5-25](#)。

表 5-25 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
name	是	String	授权名称。

## 请求消息

表 5-26 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

## 响应消息

无

## 请求示例

无

## 响应示例

无

## 状态码

状态码	描述
204	操作成功。
500	内部服务错误。

## 错误码

错误码格式为：SVCSTG.REPO.[Error\_ID]，例如：SVCSTG.REPO.0401。错误码说明请参考[ServiceStage错误码](#)。

## 5.7 获取仓库 namespaces

### 功能介绍

获取仓库namespaces。

### 调试

您可以在[API Explorer](#)中调试该接口。

### URI

GET /v1/{project\_id}/git/repos/namespaces

参数说明见[表5-27](#)。

表 5-27 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。

## 请求消息

表 5-28 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。
X-Repo-Auth	是	String	授权名称。通过 <a href="#">获取仓库授权列表</a> 获取对应的授权名称。

## 响应消息

表 5-29 响应参数

参数	参数类型	描述
namespaces	Array<Object>	命名空间结构体，请参考 <a href="#">表5-30</a> 。

表 5-30 namespaces

参数	参数类型	描述
id	String	命名空间ID。
name	String	命名空间名称。

## 请求示例

无

## 响应示例

```
{
  "namespaces": [
    {
      "id": "zmg",
      "name": "zmg"
    }
  ]
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。

## 错误码

错误码格式为：SVCSTG.REPO.[Error\_ID]，例如：SVCSTG.REPO.0401。错误码说明请参考[ServiceStage错误码](#)。

# 5.8 通过 clone url 获取仓库信息

## 功能介绍

通过clone url获取仓库信息。

## 调试

您可以在[API Explorer](#)中调试该接口。

## URI

GET /v1/{project\_id}/git/repos/project-info

参数说明见[表5-31](#)。

表 5-31 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。



表 5-32 查询参数

参数	是否必选	参数类型	描述
clone_url	是	String	仓库克隆URL。获取方法，请参考 <a href="#">获取命名空间下所有项目</a> 。

## 请求消息

表 5-33 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。
X-Repo-Auth	是	String	授权名称，通过 <a href="#">获取仓库授权列表</a> 获取对应的授权名称。

## 响应消息

表 5-34 响应参数

参数	参数类型	描述
namespace_id	String	命名空间ID。
namespace	String	命名空间。
project_id	String	仓库项目ID。
project	String	仓库项目。

## 请求示例

通过clone url获取仓库信息，clone url值为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
```

## 响应示例

```
{  
  "project": "gradle_custom_gradle_dir",  
}
```

```
"namespace_id": "zmg",  
"namespace": "zmg",  
"project_id": "79",  
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。

## 错误码

错误码格式为：SVCSTG.REPO.[Error\_ID]，例如：SVCSTG.REPO.0401。错误码说明请参考[ServiceStage错误码](#)。

# 5.9 获取命名空间下所有项目

## 功能介绍

获取命名空间下所有项目。

## 调试

您可以在[API Explorer](#)中调试该接口。

## URI

GET /v1/{project\_id}/git/repos/{namespace}/projects

参数说明见[表5-35](#)。

表 5-35 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
namespace	是	String	命名空间ID。获取方法，请参考 <a href="#">获取仓库namespaces</a> 。

## 请求消息

表 5-36 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。
X-Repo-Auth	是	String	授权名称。通过 <a href="#">获取仓库授权列表</a> 获取对应的授权名称。

## 响应消息

表 5-37 响应参数

参数	参数类型	描述
projects	Array<Object>	项目结构体，请参考 <a href="#">表5-38</a> 。

表 5-38 projects

参数	参数类型	描述
id	String	项目ID。
name	String	项目名称。
clone_url	String	克隆URL。

## 请求示例

无

## 响应示例

```
{
  "projects": [
    {
      "id": "79",
      "name": "gradle_custom_gradle_dir",
      "clone_url": "http://***@cpe.gitlab.com:8090/zmg/gradle_custom_gradle_dir.git"
    }
  ]
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。

## 错误码

错误码格式为：SVCSTG.REPO.[Error\_ID]，例如：SVCSTG.REPO.0401。错误码说明请参考[ServiceStage错误码](#)。

# 5.10 创建软件仓库项目

## 功能介绍

创建软件仓库项目。

## 调试

您可以在[API Explorer](#)中调试该接口。

## URI

POST /v1/{project\_id}/git/repos/{namespace}/projects

参数说明见[表5-39](#)。

表 5-39 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
namespace	是	String	命名空间ID。获取方法，请参考 <a href="#">获取仓库 namespaces</a> 。

## 请求消息

表 5-40 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。
X-Repo-Auth	是	String	授权名称。通过 <a href="#">获取仓库授权列表</a> 获取对应的授权名称。

表 5-41 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	项目名称。

## 响应消息

表 5-42 响应参数

参数	参数类型	描述
id	String	项目ID。
name	String	项目名称。
clone_url	String	克隆URL。

## 请求示例

创建名称为string的软件仓库项目。

```
{  
  "name": "string"  
}
```

## 响应示例

```
{  
  "id": "79",  
  "name": "gradle_custom_gradle_dir",  
  "clone_url": "http://***@cpe.gitlab.com:8090/zmg/gradle_custom_gradle_dir.git"  
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。

## 错误码

错误码格式为：SVCSTG.REPO.[Error\_ID]，例如：SVCSTG.REPO.0401。错误码说明请参考[ServiceStage错误码](#)。

# 5.11 获取项目分支

## 功能介绍

获取项目分支。

## 调试

您可以在[API Explorer](#)中调试该接口。

## URI

GET /v1/{project\_id}/git/repos/{namespace}/{project}/branches

参数说明见[表5-43](#)。

表 5-43 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
namespace	是	String	命名空间ID。获取方法，请参考 <a href="#">获取仓库 namespaces</a> 。
project	是	String	仓库项目ID，如果含有“/”，需要将“/”替换为“:”。获取方法，请参考 <a href="#">获取命名空间下所有项目</a> 。

## 请求消息

表 5-44 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。
X-Repo-Auth	是	String	授权名称。通过 <a href="#">获取仓库授权列表</a> 获取对应的授权名称。

## 响应消息

表 5-45 响应参数

参数	参数类型	描述
branches	Array<String>	项目分支。

## 请求示例

无

## 响应示例

```
{  
  "branches": [  
    "main"  
  ]  
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。

## 错误码

错误码格式为：SVCSTG.REPO.[Error\_ID]，例如：SVCSTG.REPO.0401。错误码说明请参考[ServiceStage错误码](#)。

## 5.12 获取项目 tags

### 功能介绍

获取项目tags。

### 调试

您可以在[API Explorer](#)中调试该接口。

### URI

GET /v1/{project\_id}/git/repos/{namespace}/{project}/tags

参数说明见[表5-46](#)。

表 5-46 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
namespace	是	String	命名空间ID或者URL编码名称。获取方法，请参考 <a href="#">获取仓库namespaces</a> 。
project	是	String	仓库项目ID或者URL编码名称，如果含有“/”，需要将“/”替换为“.”。获取方法，请参考 <a href="#">获取命名空间下所有项目</a> 。

### 请求消息

表 5-47 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。
X-Repo-Auth	是	String	授权名称。通过 <a href="#">获取仓库授权列表</a> 获取对应的授权名称。



## 响应消息

表 5-48 响应参数

参数	参数类型	描述
tags	Array<String>	项目tags。

## 请求示例

无

## 响应示例

```
{
  "tags": [
    "test1"
  ]
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。

## 错误码

错误码格式为：SVCSTG.REPO.[Error\_ID]，例如：SVCSTG.REPO.0401。错误码说明请参考[ServiceStage错误码](#)。

## 5.13 创建项目 tags

### 功能介绍

创建项目tags。

### 调试

您可以在[API Explorer](#)中调试该接口。

### URI

POST /v1/{project\_id}/git/repos/{namespace}/{project}/tags

参数说明见[表5-49](#)。

表 5-49 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
namespace	是	String	命名空间ID或者URL编码名称。获取方法，请参考 <a href="#">获取仓库namespaces</a> 。
project	是	String	仓库项目ID或者URL编码名称，如果含有“/”，需要将“/”替换为“.”。获取方法，请参考 <a href="#">获取命名空间下所有项目</a> 。

表 5-50 查询参数

参数	是否必选	参数类型	描述
ref	是	String	分支名称或者tag名称或者commit sha。

## 请求消息

表 5-51 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。
X-Repo-Auth	是	String	授权名称，通过 <a href="#">获取仓库授权列表</a> 获取对应的授权名称。

表 5-52 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	tags名称。
description	是	String	描述。

## 响应消息

表 5-53 响应参数

参数	参数类型	描述
name	String	项目tags名称。

## 请求示例

创建名称为test1的项目tags，tags描述信息设置为test version 1。

```
{  
  "name": "test1",  
  "description": "test version 1"  
}
```

## 响应示例

```
{  
  "name": "test1"  
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。

## 错误码

错误码格式为：SVCSTG.REPO.[Error\_ID]，例如：SVCSTG.REPO.0401。错误码说明请参考[ServiceStage错误码](#)。

## 5.14 删除项目 tags

### 功能介绍

删除项目tags。

### 调试

您可以在[API Explorer](#)中调试该接口。

### URI

DELETE /v1/{project\_id}/git/repos/{namespace}/{project}/tags/{tag\_name}

参数说明见[表5-54](#)。

表 5-54 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
namespace	是	String	命名空间ID或者URL编码名称。获取方法，请参考 <a href="#">获取仓库namespaces</a> 。
project	是	String	仓库项目ID或者URL编码名称，如果含有“/”，需要将“/”替换为“:”。获取方法，请参考 <a href="#">获取命名空间下所有项目</a> 。
tag_name	是	String	tag名称。获取方法，请参考 <a href="#">获取项目tags</a> 。

## 请求消息

表 5-55 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。
X-Repo-Auth	是	String	授权名称。通过 <a href="#">获取仓库授权列表</a> 获取对应的授权名称。

## 响应消息

无

## 请求示例

无

## 响应示例

无

## 状态码

状态码	描述
204	操作成功。
400	错误的请求。

## 错误码

错误码格式为：SVCSTG.REPO.[Error\_ID]，例如：SVCSTG.REPO.0401。错误码说明请参考[ServiceStage错误码](#)。

## 5.15 获取项目 commits

### 功能介绍

获取项目最近10次commits。

### 调试

您可以在[API Explorer](#)中调试该接口。

### URI

GET /v1/{project\_id}/git/repos/{namespace}/{project}/commits

参数说明见[表5-56](#)。

表 5-56 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
namespace	是	String	命名空间ID或者URL编码名称。获取方法，请参考 <a href="#">获取仓库namespaces</a> 。
project	是	String	仓库项目ID或者URL编码名称，如果含有“/”，需要将“/”替换为“:”。获取方法，请参考 <a href="#">获取命名空间下所有项目</a> 。

表 5-57 查询参数

参数	是否必选	参数类型	描述
ref	否	String	分支名称或者tag名称，如果没有提供，使用默认分支。

## 请求消息

表 5-58 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。
X-Repo-Auth	是	String	授权名称，通过 <a href="#">获取仓库授权列表</a> 获取对应的授权名称。

## 响应消息

表 5-59 响应参数

参数	参数类型	描述
commits	Array<Object>	项目commits结构体，请参考 <a href="#">表5-60</a> 。

表 5-60 commits

参数	参数类型	描述
sha	String	commit哈希。
message	String	commit描述。
authored_date	String	合入时间。

## 请求示例

无

## 响应示例

```
{
  "commits": [
    {
      "sha": "6dcb09b5b57875f334f61aebd695e2e4193db5e",
      "message": "Fix all the bugs",
      "authored_date": "Fix all the bugs"
    }
  ]
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。

## 错误码

错误码格式为：SVCSTG.REPO.[Error\_ID]，例如：SVCSTG.REPO.0401。错误码说明请参考[ServiceStage错误码](#)。

# 5.16 获取项目 hooks

## 功能介绍

获取项目hooks。

## 调试

您可以在[API Explorer](#)中调试该接口。

## URI

GET /v1/{project\_id}/git/repos/{namespace}/{project}/hooks

参数说明见[表5-61](#)。

表 5-61 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
namespace	是	String	命名空间ID或者URL编码名称。获取方法，请参考 <a href="#">获取仓库namespaces</a> 。

参数	是否必选	参数类型	描述
project	是	String	仓库项目ID或者URL编码名称，如果含有“/”，需要将“/”替换为“:”。获取方法，请参考 <a href="#">获取命名空间下所有项目</a> 。

## 请求消息

表 5-62 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。
X-Repo-Auth	是	String	授权名称，通过 <a href="#">获取仓库授权列表</a> 获取对应的授权名称。

## 响应消息

表 5-63 响应参数

参数	参数类型	描述
hooks	Array of objects	项目hooks，请参考 <a href="#">表5-64</a> 。

表 5-64 hooks

参数	参数类型	描述
id	String	hook ID。
type	String	hook类型。
callback_url	String	回调url。

## 请求示例

无



## 响应示例

```
{
  "hooks": [
    {
      "id": "1753",
      "type": "github",
      "callback_url": "https://example.com/webhook"
    }
  ]
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。

## 错误码

错误码格式为：SVCSTG.REPO.[Error\_ID]，例如：SVCSTG.REPO.0401。错误码说明请参考[ServiceStage错误码](#)。

# 5.17 创建项目 hook

## 功能介绍

创建项目hook。

## 调试

您可以在[API Explorer](#)中调试该接口。

## URI

POST /v1/{project\_id}/git/repos/{namespace}/{project}/hooks

参数说明见[表5-65](#)。

表 5-65 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
namespace	是	String	命名空间ID或者URL编码名称。获取方法，请参考 <a href="#">获取仓库namespaces</a> 。

参数	是否必选	参数类型	描述
project	是	String	仓库项目ID或者URL编码名称，如果含有“/”，需要将“/”替换为“:”。获取方法，请参考 <a href="#">获取命名空间下所有项目</a> 。

## 请求消息

表 5-66 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。
X-Repo-Auth	是	String	授权名称，通过 <a href="#">获取仓库授权列表</a> 获取对应的授权名称。

表 5-67 请求 Body 参数

参数	是否必选	参数类型	描述
secret	是	String	用于验证接收到的payloads的字符串。
url	是	String	hook触发时回调的url。

## 响应消息

表 5-68 响应参数

参数	参数类型	描述
id	String	hook ID。
type	String	hook类型。
callback_url	String	回调url。

## 请求示例

创建项目hook，验证接收到的payloads的字符串的secrete参数，hook触发时回调的url设置为https://example.com/webhook。

```
{
  "secret": "*****",
  "url": "https://example.com/webhook"
}
```

## 响应示例

```
{
  "id": "1573",
  "type": "github",
  "callback_url": "https://example.com/webhook"
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。

## 错误码

错误码格式为：SVCSTG.REPO.[Error\_ID]，例如：SVCSTG.REPO.0401。错误码说明请参考[ServiceStage错误码](#)。

# 5.18 删除项目 hook

## 功能介绍

删除项目hook。

## 调试

您可以在[API Explorer](#)中调试该接口。

## URI

DELETE /v1/{project\_id}/git/repos/{namespace}/{project}/hooks/{hook\_id}

参数说明见[表5-69](#)。

表 5-69 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。

参数	是否必选	参数类型	描述
namespace	是	String	命名空间ID或者URL编码名称。获取方法，请参考 <a href="#">获取仓库namespaces</a> 。
project	是	String	仓库项目ID或者URL编码名称，如果含有“/”，需要将“/”替换为“:”。获取方法，请参考 <a href="#">获取命名空间下所有项目</a> 。
hook_id	是	String	hook ID。获取方法，请参考 <a href="#">获取项目hooks</a> 。

## 请求消息

表 5-70 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种。如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。
X-Repo-Auth	是	String	授权名称。通过 <a href="#">获取仓库授权列表</a> 获取对应的授权名称。

## 响应消息

无

## 请求示例

无

## 响应示例

无

## 状态码

状态码	描述
204	操作成功。
400	错误的请求。

## 错误码

错误码格式为：SVCSTG.REPO.[Error\_ID]，例如：SVCSTG.REPO.0401。错误码说明请参考[ServiceStage错误码](#)。

## 5.19 获取仓库文件目录

### 功能介绍

获取仓库文件目录。

### 调试

您可以在[API Explorer](#)中调试该接口。

### URI

GET /v1/{project\_id}/git/files/{namespace}/{project}/trees

参数说明见[表5-71](#)。

表 5-71 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
namespace	是	String	命名空间ID或者URL编码名称。获取方法，请参考 <a href="#">获取仓库namespaces</a> 。
project	是	String	仓库项目ID或者URL编码名称，如果含有“/”，需要将“/”替换为“:”。获取方法，请参考 <a href="#">获取命名空间下所有项目</a> 。

表 5-72 查询参数

参数	是否必选	参数类型	描述
ref	是	String	分支名称或者tag名称或者commit sha。

## 请求消息

表 5-73 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种。如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式,请参考 <a href="#">获取用户Token</a> 。
X-Repo-Auth	是	String	授权名称。通过 <a href="#">获取仓库授权列表</a> 获取对应的授权名称。

## 响应消息

表 5-74 响应参数

参数	参数类型	描述
paths	Array<String>	文件路径。

## 请求示例

无

## 响应示例

```
{
  "paths": [
    "files/whitespace"
  ]
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。

## 错误码

错误码格式为：SVCSTG.REPO.[Error\_ID]，例如：SVCSTG.REPO.0401。错误码说明请参考[ServiceStage错误码](#)。

## 5.20 获取仓库文件内容

### 功能介绍

获取仓库文件内容。

### 调试

您可以在[API Explorer](#)中调试该接口。

### URI

GET /v1/{project\_id}/git/files/{namespace}/{project}/{path}

参数说明见[表5-75](#)。

表 5-75 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
namespace	是	String	命名空间ID或者URL编码名称。获取方法，请参考 <a href="#">获取仓库namespaces</a> 。
project	是	String	仓库项目ID或者URL编码名称，如果含有“/”，需要将“/”替换为“:”。获取方法，请参考 <a href="#">获取命名空间下所有项目</a> 。
path	是	String	文件路径，需要将“/”替换为“:”。获取方法，请参考 <a href="#">获取仓库文件目录</a> 。

表 5-76 查询参数

参数	是否必选	参数类型	描述
ref	是	String	分支名称或者tag名称或者commit sha。

## 请求消息

表 5-77 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种。如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。
X-Repo-Auth	是	String	授权名称，通过 <a href="#">获取仓库授权列表</a> 获取对应的授权名称。

## 响应消息

表 5-78 响应参数

参数	参数类型	描述
path	String	文件路径。
sha	String	commit哈希。
encoding	String	编码方式：Base64或者text/plain。
content	String	文件内容。

## 请求示例

无

## 响应示例

```
{
  "path": "app/models/key.rb",
  "sha": "4c294617b60715c1d218e61164a3abd4808a4284cbc30e6728a01ad9aada4481",
  "encoding": "base64",
  "content": "lyA9PSBTY2hlbWEgSW5mb3..."
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。



## 错误码

错误码格式为：SVCSTG.REPO.[Error\_ID]，例如：SVCSTG.REPO.0401。错误码说明请参考[ServiceStage错误码](#)。

## 5.21 创建仓库文件

### 功能介绍

创建仓库文件。

### 调试

您可以在[API Explorer](#)中调试该接口。

### URI

POST /v1/{project\_id}/git/files/{namespace}/{project}/{path}

参数说明见[表5-79](#)。

表 5-79 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
namespace	是	String	命名空间ID或者URL编码名称。获取方法，请参考 <a href="#">获取仓库namespaces</a> 。
project	是	String	仓库项目ID或者URL编码名称，如果含有“/”，需要将“/”替换为“:”。获取方法，请参考 <a href="#">获取命名空间下所有项目</a> 。
path	是	String	文件路径，需要将“/”替换为“:”。获取方法，请参考 <a href="#">获取仓库文件目录</a> 。

表 5-80 查询参数

参数	是否必选	参数类型	描述
ref	是	String	分支名称或者tag名称或者commit哈希。

## 请求消息

表 5-81 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种。如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式,请参考 <a href="#">获取用户Token</a> 。
X-Repo-Auth	是	String	授权名称，通过 <a href="#">获取仓库授权列表</a> 获取对应的授权名称。

表 5-82 请求 Body 参数

参数	是否必选	参数类型	描述
message	是	String	描述。
content	是	String	Base64编码后的文件内容。

## 响应消息

表 5-83 响应参数

参数	参数类型	描述
path	String	文件路径。

## 请求示例

创建仓库文件，文件描述信息参数message设置为create a new file，文件内容参数content设置为Base64编码后的内容。

```
{
  "message": "create a new file",
  "content": "lyA9PSBTY2hnbWEgSW5mb3..."
}
```

## 响应示例

```
{
  "path": "app/project.rb"
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。

## 错误码

错误码格式为：SVCSTG.REPO.[Error\_ID]，例如：SVCSTG.REPO.0401。错误码说明请参考[ServiceStage错误码](#)。

## 5.22 修改仓库文件内容

### 功能介绍

修改仓库文件内容。

### 调试

您可以在[API Explorer](#)中调试该接口。

### URI

PUT /v1/{project\_id}/git/files/{namespace}/{project}/{path}

参数说明见[表5-84](#)。

表 5-84 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
namespace	是	String	命名空间ID或者URL编码名称。获取方法，请参考 <a href="#">获取仓库namespaces</a> 。
project	是	String	仓库项目ID或者URL编码名称，如果含有“/”，需要将“/”替换为“:”。获取方法，请参考 <a href="#">获取命名空间下所有项目</a> 。
path	是	String	文件路径，需要将“/”替换为“:”。获取方法，请参考 <a href="#">获取仓库文件目录</a> 。

表 5-85 查询参数

参数	是否必选	参数类型	描述
ref	是	String	分支名称或者tag名称或者commit哈希。

## 请求消息

表 5-86 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种。如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。
X-Repo-Auth	是	String	授权名称，通过 <a href="#">获取仓库授权列表</a> 获取对应的授权名称。

表 5-87 请求 Body 参数

参数	是否必选	参数类型	描述
message	是	String	描述。
content	是	String	Base64编码后的文件内容。
sha	是	String	文件的Blob SHA，可通过 <a href="#">获取仓库文件内容</a> API获取。

## 响应消息

表 5-88 响应参数

参数	参数类型	描述
path	String	文件路径。

## 请求示例

修改仓库文件内容。

```
{  
  "message": "update file",  
  "content": "lyA9PSBTY2hbwEgSW5mb3...",  
  "sha": "4c294617b60715c1d218e61164a3abd4808a4284cbc30e6728a01ad9aada4481"  
}
```

## 响应示例

```
{  
  "path": "app/project.rb"  
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。

## 错误码

错误码格式为：SVCSTG.REPO.[Error\_ID]，例如：SVCSTG.REPO.0401。错误码说明请参考[ServiceStage错误码](#)。

## 5.23 删除仓库文件

### 功能介绍

删除仓库文件内容。

### 调试

您可以在[API Explorer](#)中调试该接口。

### URI

DELETE /v1/{project\_id}/git/files/{namespace}/{project}/{path}

参数说明见[表5-89](#)。

表 5-89 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
namespace	是	String	命名空间ID或者URL编码名称。获取方法，请参考 <a href="#">获取仓库namespaces</a> 。

参数	是否必选	参数类型	描述
project	是	String	仓库项目ID或者URL编码名称，如果含有“/”，需要将“/”替换为“.”。获取方法，请参考 <a href="#">获取命名空间下所有项目</a> 。
path	是	String	文件路径，需要将“/”替换为“:”。获取方法，请参考 <a href="#">获取仓库文件目录</a> 。

表 5-90 查询参数

参数	是否必选	参数类型	描述
ref	是	String	分支名称或者tag名称或者commit哈希。
message	是	String	commit信息。
sha	是	String	文件的Blob SHA，可通过 <a href="#">获取仓库文件内容API</a> 获取。

## 请求消息

表 5-91 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种。如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式请参考 <a href="#">获取用户Token</a> 。
X-Repo-Auth	是	String	授权名称，通过 <a href="#">获取仓库授权列表</a> 获取对应的授权名称。

## 响应消息

无

## 请求示例

无

## 响应示例

无

## 状态码

状态码	描述
204	操作成功。
400	错误的请求。

## 错误码

错误码格式为：SVCSTG.REPO.[Error\_ID]，例如：SVCSTG.REPO.0401。错误码说明请参考[ServiceStage错误码](#)。

# 6 CSE API

## 6.1 调用说明

CSE提供了REST ( Representational State Transfer ) 风格API，支持您通过HTTPS请求调用。

调用微服务引擎专享版的CSE API的方法，请参考[如何调用API](#)。

## 6.2 动态配置

### 6.2.1 导入配置

#### 功能介绍

从配置管理中心导入配置信息。

#### 📖 说明

- 此接口适用于微服务引擎为2.x版本。
- 参考[如何调用API](#)调用该接口，在[请求URI](#)中，替换{Endpoint}为[终端节点](#)。

#### 调试

您可以在[API Explorer](#)中调试该接口，支持自动认证鉴权。API Explorer可以自动生成SDK代码示例，并提供SDK代码示例调试功能。

#### URI

POST /v1/{project\_id}/kie/file



表 6-1 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户子项目的唯一标识，字符长度为1~64位，获取方法请参考 <a href="#">获取项目ID</a> 。

表 6-2 Query 参数

参数	是否必选	参数类型	描述
override	是	String	force: 强制导入，会覆盖重复项。 skip: 跳过导入，会跳过重复项。 abort: 终止导入，遇到重复项会停止导入操作。
label	否	String	指定label导入，格式为：{标签key}:{标签value}，如果不填则按body的label导入。

## 请求参数

表 6-3 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。
x-engine-id	是	String	待上传配置的引擎id，获取方法请参考 <a href="#">查询微服务引擎专享版列表</a> 。
X-Enterprise-Project-ID	否	String	企业项目id。 如果不配置此参数，则默认企业项目为“default”，ID为0。

表 6-4 请求 Body 参数

参数	是否必选	参数类型	描述
upload_file	是	File	导入的文件，文件名格式为： fileName.json 文件格式为： <pre>{   "data": [     {       "key": "keyItem1",       "value": "valueItem1",       "labels": {         "1": "1"       },       "status": "enabled",       "value_type": "text"     }   ] }</pre>

## 响应参数

状态码： 200

表 6-5 响应 Body 参数

参数	参数类型	描述
success	List<KVDoc>	导入成功的配置项列表
failure	List<DocFailedOfUpload>	导入失败的配置项列表

表 6-6 KVDoc

参数	参数类型	描述
id	String	配置id
key	String	配置的key
value	String	配置的值
value_type	String	配置值的类型
status	String	配置的状态
create_time	Integer	配置的创建时间
update_time	Integer	配置的更新时间
create_revision	Integer	创建配置版本号

参数	参数类型	描述
update_revision	Integer	修改配置版本号
labels	Map<String,String>	配置的标签信息

表 6-7 DocFailedOfUpload

参数	参数类型	描述
key	String	导入失败配置项的key
labels	Map<String,String>	导入失败配置项的labels信息
error_code	String	错误代码
error_message	String	错误信息

状态码： 400

表 6-8 响应 Body 参数

参数	参数类型	描述
error_code	String	错误代码
error_message	String	错误信息
detail	String	详细定位信息

状态码： 500

表 6-9 响应 Body 参数

参数	参数类型	描述
error_code	String	错误代码
error_message	String	错误信息
detail	String	详细定位信息

## 请求示例

导入配置文件，当存在相同配置时，会强制导入覆盖重复项，按body的label导入。

```
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
```

## 响应示例

查询成功

```
{
  "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": []
}
```

## 状态码

状态码	描述
200	导入成功。
400	错误的请求。
500	服务内部错误。

## 错误码

请参见[CSE错误码](#)。

## 6.2.2 导出配置

### 功能介绍

从配置管理中心导出配置信息。

#### 说明

- 此接口适用于微服务引擎为2.x版本。
- 参考[如何调用API](#)调用该接口，在[请求URI](#)中，替换{Endpoint}为[终端节点](#)。

### 调试

您可以在[API Explorer](#)中调试该接口，支持自动认证鉴权。API Explorer可以自动生成SDK代码示例，并提供SDK代码示例调试功能。

### URI

POST /v1/{project\_id}/kie/download

表 6-10 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户子项目的唯一标识，字符长度为1~64位，获取方法请参考 <a href="#">获取项目ID</a> 。

表 6-11 Query 参数

参数	是否必选	参数类型	描述
label	否	String	按label过滤项导出，格式为： {标签key}:{标签value}。
match	否	String	对label过滤项的匹配选项。 如果值为exact表示严格匹配， 包括label个数和内容相等。 不填表示包含匹配。

## 请求参数

表 6-12 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。
x-engine-id	是	String	待下载配置的引擎id，获取方法请参考 <a href="#">查询微服务引擎专享版列表</a> 。
X-Enterprise-Project-ID	否	String	企业项目id。 如果不配置此参数，则默认企业项目为“default”，ID为0。

表 6-13 请求 Body 参数

参数	是否必选	参数类型	描述
ids	是	List<String>	待导出的配置项的id列表，获取方法请参考 <a href="#">查询配置列表</a> 。当ids取空列表时，调用该接口表示导出所有配置项。

## 响应参数

状态码： 200

表 6-14 响应 Body 参数

参数	是否必选	参数类型	描述
metadata	否	<b>Object</b>	配置项的额外信息。
data	是	List< <b>KVCreateBody</b> >	待创建的配置项列表。

表 6-15 KVCreateBody

参数	是否必选	参数类型	描述
id	是	String	配置id。
key	是	String	配置key。
value	否	String	配置value。

参数	是否必选	参数类型	描述
labels	否	Map<String,String>	配置的labels。
status	否	String	配置的状态。
value_type	否	String	配置value的类型。

表 6-16 Metadata

参数	是否必选	参数类型	描述
version	否	String	版本信息。
annotations	否	Object	配置项的额外信息。

状态码： 400

表 6-17 响应 Body 参数

参数	参数类型	描述
error_code	String	错误代码。
error_message	String	错误信息。
detail	String	详细定位信息。

状态码： 500

表 6-18 响应 Body 参数

参数	参数类型	描述
error_code	String	错误代码。
error_message	String	错误信息。
detail	String	详细定位信息。

## 请求示例

导出配置ID为“0bdfddb8-c4f9-4a88-8360-07a8a616a804”的配置信息。

```
POST https://{endpoint}/v1/{project_id}/kie/download
```

```
{
  "ids":[
    "0bdfddb8-c4f9-4a88-8360-07a8a616a804"
  ]
}
```

## 响应示例

查询成功

```
{
  "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"
    }
  ]
}
```

## 状态码

状态码	描述
200	导出成功。
400	错误的请求。
500	服务内部错误。

## 错误码

请参见[CSE错误码](#)。

## 6.3 引擎管理



## 6.3.1 查询微服务引擎专享版支持规格

### 功能介绍

查询当前支持的微服务引擎专享版的规格列表。

### 调试

您可以在 [API Explorer](#) 中调试该接口，支持自动认证鉴权。API Explorer 可以自动生成 SDK 代码示例，并提供 SDK 代码示例调试功能。

### URI

GET /v2/{project\_id}/enginemgr/flavors

表 6-19 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户子项目的唯一标识，字符长度为1~64位，获取方法请参考 <a href="#">获取项目ID</a> 。

表 6-20 query 参数

参数	是否必选	参数类型	描述
specType	否	String	<ul style="list-style-type: none"><li>查询1.x版本CSE引擎支持规格时，将specType设置为CSE。</li><li>查询2.x版本CSE引擎支持规格时，将specType设置为CSE2。</li></ul>

### 请求参数

表 6-21 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。
X-Enterprise-Project-ID	否	String	企业项目id。 如果不配置此参数，则默认企业项目为“default”，ID为0。

## 响应参数

状态码： 200

表 6-22 响应 Body 参数

参数	参数类型	描述
total	Integer	当前微服务引擎专享版规格总个数。
data	Array of Flavor objects	微服务引擎专享版规格详情。

表 6-23 Flavor

参数	参数类型	描述
flavor	String	微服务引擎专享版规格。
description	String	微服务引擎专享版规格描述。

状态码： 400

表 6-24 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

状态码： 500

表 6-25 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

## 请求示例

查询当前支持的微服务引擎专享版的规格列表。

GET /v2/{project\_id}/enginemgr/flavors

## 响应示例

```
{
  "total": 0,
  "data": [
    {
      "flavor": "string",
      "description": "string"
    }
  ]
}
```

## 状态码

状态码	描述
200	查询当前支持的微服务引擎专享版的规格列表成功。
400	错误的请求。
500	服务内部错误。

## 错误码

请参见[CSE错误码](#)。

## 6.3.2 查询微服务引擎专享版列表

### 功能介绍

查询微服务引擎专享版列表。

### 调试

您可以在[API Explorer](#)中调试该接口，支持自动认证鉴权。API Explorer可以自动生成SDK代码示例，并提供SDK代码示例调试功能。

### URI

GET /v2/{project\_id}/enginemgr/engines

表 6-26 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户子项目的唯一标识，字符长度为1~64位，获取方法请参考 <a href="#">获取项目ID</a> 。

表 6-27 Query 参数

参数	是否必选	参数类型	描述
offset	否	Integer	偏移量
limit	否	Integer	每页数据大小
type	否	String	查询的引擎类型：CSE、CSE_SHARE。

## 请求参数

表 6-28 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。

## 响应参数

状态码： 200

表 6-29 响应 Body 参数

参数	参数类型	描述
total	Integer	当前微服务引擎专享版总个数。
data	Array of <a href="#">EngineInfo</a> objects	微服务引擎专享版详情。

表 6-30 EngineInfo

参数	参数类型	描述
id	String	微服务引擎专享版ID。
name	String	微服务引擎专享版名称。
enterpriseProjectId	String	微服务引擎专享版所属企业项目ID。
enterpriseProjectName	String	微服务引擎专享版所属企业项目名称。
type	String	查询的引擎类型：CSE、CSE_SHARE。
description	String	微服务引擎专享版描述。

参数	参数类型	描述
flavor	String	微服务引擎专享版规格。
payment	String	微服务引擎专享版计费方式，0表示包周期，1表示按需，2表示免费。
authType	String	微服务引擎专享版认证方式，RBAC为安全认证，NONE为无认证。
status	String	微服务引擎专享版当前状态。
externalAddress	String	微服务引擎专享版租户侧VPC内访问地址。
serviceEndpoint	Object	微服务引擎专享版组件租户侧VPC内访问地址。
publicAddress	String	微服务引擎专享版公网访问地址，需要开启公网访问。
publicServiceEndpoint	Object	微服务引擎专享版组件公网访问地址，需要开启公网访问。
totalInstance	Integer	微服务引擎专享版可支持的实例总数。
usedInstance	Integer	已使用的实例总数。
availableInstance	Integer	可用的实例总数。
version	String	微服务引擎专享版当前版本。
latestVersion	String	微服务引擎专享版最新版本。
createTime	Integer	微服务引擎专享版创建时间。
dueTo	Integer	微服务引擎专享版到期时间。
latestJobId	Integer	微服务引擎专享版最近的任务ID。
engineAdditionalActions	Array of String	微服务引擎专享版允许的附加操作。 Delete: 删除。 ForceDelete: 强制删除。 Upgrade: 升级。 Retry: 重试。 Configure: 变更引擎配置。
specType	String	微服务引擎专享版部署类型。 <ul style="list-style-type: none"><li>• 1.x引擎时，固定值CSE。</li><li>• 2.x引擎时，固定值为CSE2。</li></ul>
reference	Object	微服务引擎专享版附加信息。

表 6-31 Endpoint

参数	参数类型	描述
kie	Object	配置中心Endpoint信息。
serviceCenter	Object	服务中心Endpoint信息。

表 6-32 EntryPoint

参数	参数类型	描述
masterEntryPoint	String	微服务引擎专享版组件VPC内ipv4主访问地址。
masterEntryPointIpv6	String	微服务引擎专享版组件VPC内ipv6主访问地址。
slaveEntryPoint	String	微服务引擎专享版组件VPC内ipv4备访问地址。
slaveEntryPointIpv6	String	微服务引擎专享版组件VPC内ipv6备访问地址。
type	String	微服务引擎专享版组件类型。 CSE：微服务引擎专享版。

表 6-33 EngineReference

参数	参数类型	描述
vpc	String	vpc名称。
vpcId	String	vpc的id，长度不超过64位。
azList	Array of String	微服务引擎专享版部署可用区列表。
networkId	String	微服务引擎专享版子网网络ID。
subnetCidr	String	微服务引擎专享版ipv4子网划分。
subnetCidrV6	String	微服务引擎专享版ipv6子网划分。
subnetGateway	String	微服务引擎专享版子网网关。
publicIpId	String	微服务引擎专享版公网地址ID，需要开启公网访问。
serviceLimit	Integer	套餐可支持的微服务总数。
instanceLimit	Integer	套餐可支持的实例总数。

参数	参数类型	描述
inputs	Map <String,String >	微服务引擎专享版附加参数。

状态码： 400

表 6-34 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

状态码： 500

表 6-35 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

## 请求示例

查询当前的微服务引擎列表。

```
GET /v2/{project_id}/enginemgr/engines
```

## 响应示例

```
{
  "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"
  }
}
}
}
}

```

## 状态码

状态码	描述
200	查询微服务引擎专享版列表成功。



状态码	描述
400	错误的请求。
500	服务内部错误。

## 错误码

请参见[CSE错误码](#)。

## 6.3.3 创建微服务引擎专享版

### 功能介绍

创建微服务引擎专享版。

### 调试

您可以在[API Explorer](#)中调试该接口，支持自动认证鉴权。API Explorer可以自动生成SDK代码示例，并提供SDK代码示例调试功能。

### URI

POST /v2/{project\_id}/enginemgr/engines

表 6-36 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户子项目的唯一标识，字符长度为1~64位，获取方法请参考 <a href="#">获取项目ID</a> 。

### 请求参数

表 6-37 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。
X-Enterprise-Project-ID	否	String	企业项目id。 如果不配置此参数，则默认企业项目为“default”，ID为0。

表 6-38 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	微服务引擎专享版名称，名称为字母开头，字母、数字、-组成，且不能以-结尾，长度3~24。 <b>须知</b> 微服务引擎名称不允许命名为“default”。
description	否	String	微服务引擎专享版描述，长度0~255。
payment	是	String	微服务引擎专享版计费方式，通过接口只支持创建计费方式为按需的引擎。 设置为固定值1。
flavor	是	String	微服务引擎专享版规格。 <ul style="list-style-type: none"> <li>• cse.s1.small2：高可用100实例引擎。</li> <li>• cse.s1.medium2：高可用200实例引擎。</li> <li>• cse.s1.large2：高可用500实例引擎。</li> <li>• cse.s1.xlarge2：高可用2000实例引擎。</li> </ul>
azList	是	Array of String	当前局点可用区列表。
authType	是	String	微服务引擎专享版认证方式，RBAC为安全认证，NONE为无认证。
authConfig	否	<a href="#">表6-39</a>	微服务引擎的RBAC设置信息。若不传，默认为开启编程接口安全认证。
vpc	是	String	vpc名称。
vpcId	是	String	vpc的id，长度不超过64位。
networkId	是	String	微服务引擎专享版子网ID。
subnetCidr	是	String	微服务引擎专享版子网划分。
publicIpId	否	String	微服务引擎专享版公网访问地址ID。
specType	是	String	微服务引擎专享版部署类型。 <ul style="list-style-type: none"> <li>• 1.x引擎时，固定值CSE。</li> <li>• 2.x引擎时，固定值为CSE2。</li> </ul>

参数	是否必选	参数类型	描述
auth_cred	否	object	微服务引擎专享版选择安全认证时必传，包含引擎的认证信息。
inputs	否	Map<String,String>	微服务引擎专享版附加参数。

表 6-39 authConfig

参数	是否必选	参数类型	描述
consoleAuthEnabled	是	Boolean	是否开启控制台安全认证。
sdkAuthEnabled	是	Boolean	是否开启编程接口安全认证。 若编程接口认证参数 sdkAuthEnabled 值为 true 时，则控制台安全认证参数 consoleAuthEnabled 也需要设置为 true。

表 6-40 object

参数	是否必传	参数类型	描述
pwd	是	String	开启安全认证引擎的 root 账号密码。

## 响应参数

状态码： 200

表 6-41 响应 Body 参数

参数	参数类型	描述
id	String	微服务引擎专享版专享版ID。
name	String	微服务引擎专享版名称。
jobId	Integer	微服务引擎专享版执行任务ID。

状态码： 400

表 6-42 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

状态码： 500

表 6-43 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

## 请求示例

创建一个名称为“test”微服务引擎专享版，其计费模式为按需计费，规格为cse.s1.small2，可用区为test，关闭安全认证，微服务引擎的部署类型为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"
  }
}
```

## 响应示例

```
{
  "id": "891bf21a-4024-4f47-b38c-bd259ca8f10a",
  "name": "test",
  "jobId": 17655
}
```

## 状态码

状态码	描述
200	查询当前支持的微服务引擎专享版的规格列表成功。
400	错误的请求。
500	服务内部错误。

## 错误码

请参见[CSE错误码](#)。

### 6.3.4 查询微服务引擎专享版详情

#### 功能介绍

查询微服务引擎专享版详情。

#### 调试

您可以在[API Explorer](#)中调试该接口，支持自动认证鉴权。API Explorer可以自动生成SDK代码示例，并提供SDK代码示例调试功能。

#### URI

GET /v2/{project\_id}/enginemgr/engines/{engine\_id}

表 6-44 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户子项目的唯一标识，字符长度为1~64位，获取方法请参考 <a href="#">获取项目ID</a> 。
engine_id	是	String	微服务引擎专享版ID，获取方法请参考 <a href="#">查询微服务引擎专享版列表</a> 。

#### 请求参数

表 6-45 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。

参数	是否必选	参数类型	描述
X-Enterprise-Project-ID	否	String	企业项目id。 如果不配置此参数，则默认企业项目为“default”，ID为0。

## 响应参数

状态码： 200

表 6-46 响应 Body 体

参数	参数类型	描述
id	String	微服务引擎专享版ID。
name	String	微服务引擎专享版名称。
projectId	String	微服务引擎专享版所属项目ID。
enterpriseProjectId	String	微服务引擎专享版所属企业项目ID。
enterpriseProjectName	String	微服务引擎专享版所属企业项目名称。
type	String	引擎的类型。CSE为微服务引擎专享版。
description	String	微服务引擎专享版描述。
beDefault	Boolean	微服务引擎专享版是否是默认引擎。
flavor	String	微服务引擎专享版规格。 <ul style="list-style-type: none"><li>● cse.s1.small2：高可用100实例引擎。</li><li>● cse.s1.medium2：高可用200实例引擎。</li><li>● cse.s1.large2：高可用500实例引擎。</li><li>● cse.s1.xlarge2：高可用2000实例引擎。</li></ul>
payment	String	微服务引擎专享版计费方式，0表示包周期，1表示按需，2表示免费。
authType	String	微服务引擎专享版认证方式，RBAC为安全认证，NONE为无认证。
authConfig	<a href="#">表6-47</a>	微服务引擎的RBAC设置信息。
status	String	微服务引擎专享版当前状态。
cceSpec	<a href="#">Object</a>	微服务引擎专享版CCE规格。
externalEntryPoint	<a href="#">Object</a>	微服务引擎专享版接入地址。

参数	参数类型	描述
publicAddress	String	微服务引擎专享版公网访问地址，需要开启公网访问。
version	String	微服务引擎专享版当前版本。
latestVersion	String	微服务引擎专享版最新版本。
createTime	Integer	微服务引擎专享版创建时间。
createUser	String	微服务引擎专享版创建者。
latestJobId	Integer	微服务引擎专享版最近的任务ID。
engineAdditionalActions	Array of String	微服务引擎专享版允许的附加操作。 Delete: 删除。 ForceDelete: 强制删除。 Upgrade: 升级。 Retry: 重试。
specType	String	微服务引擎专享版部署类型。 <ul style="list-style-type: none"> <li>1.x引擎时，固定值CSE。</li> <li>2.x引擎时，固定值为CSE2。</li> </ul>
reference	<b>Object</b>	微服务引擎专享版附加信息。
vmlIds	Array of String	当前微服务引擎专享版在资源租户侧使用的虚拟机id列表。

表 6-47 authConfig

参数	参数类型	描述
consoleAuthEnabled	Boolean	是否开启控制台安全认证。
sdkAuthEnabled	Boolean	是否开启编程接口安全认证。

表 6-48 EngineReference

参数	参数类型	描述
vpc	String	vpc名称。
vpcId	String	vpc的id，长度不超过64位。
azList	Array of String	微服务引擎专享版部署可用区列表。

参数	参数类型	描述
networkId	String	微服务引擎专享版子网网络ID。
subnetCidr	String	微服务引擎专享版ipv4子网划分。
subnetCidrV6	String	微服务引擎专享版pv6子网划分。
subnetGateway	String	微服务引擎专享版子网网关。
publicIpId	String	微服务引擎专享版公网地址ID，需要开启公网访问。
serviceLimit	Integer	套餐可支持的微服务总数。
instanceLimit	Integer	套餐可支持的实例总数。
inputs	Map <String,String >	微服务引擎专享版附加参数。

表 6-49 Spec

参数	参数类型	描述
id	Integer	微服务引擎专享版CCE规格ID。
engineId	String	微服务引擎专享版ID。
specType	String	微服务引擎专享版CCE集群部署类型
cluster	String	微服务引擎专享版CCE集群信息，目前为null。
clusterId	String	微服务引擎专享版CCE集群ID。
clusterNodes	<b>Object</b>	微服务引擎专享版CCE节点列表。
flavor	String	微服务引擎专享版CCE集群规格。
region	String	微服务引擎专享版CCE集群所在局点。
version	String	微服务引擎专享版CCE集群版本。
extendParam	String	微服务引擎专享版CCE集群附加参数。

表 6-50 SpecClusterNode

参数	参数类型	描述
clusterNodes	Array of <b>ClusterNode</b> objects	微服务引擎专享版CCE节点信息。



表 6-51 ClusterNode

参数	参数类型	描述
id	String	微服务引擎专享版CCE节点ID。
az	String	微服务引擎专享版CCE节点所属可用区。
ip	String	微服务引擎专享版CCE节点IP。
label	String	微服务引擎专享版CCE节点标签。
status	String	微服务引擎专享版CCE节点状态。

表 6-52 EngineExternalEntrypoint

参数	参数类型	描述
externalAddress	String	微服务引擎专享版租户侧VPC内访问地址。
publicAddress	String	微服务引擎专享版公网访问地址，需要开启公网访问。
serviceEndpoint	Object	微服务引擎专享版组件租户侧VPC内访问地址。
publicServiceEndpoint	Object	微服务引擎专享版组件公网访问地址，需要开启公网访问。

表 6-53 Endpoint

参数	参数类型	描述
kie	Object	配置中心Endpoint信息
serviceCenter	Object	服务中心Endpoint信息

表 6-54 EntryPoint

参数	参数类型	描述
masterEndpoint	String	微服务引擎专享版组件VPC内ipv4主访问地址。
masterEndpointIpv6	String	微服务引擎专享版组件VPC内ipv6主访问地址。
slaveEndpoint	String	微服务引擎专享版组件VPC内ipv4备访问地址。

参数	参数类型	描述
slaveEntrypointIpv6	String	微服务引擎专享版组件VPC内ipv6备访问地址。
type	String	微服务引擎专享版组件类型。 CSE：微服务引擎专享版。

状态码： 400

表 6-55 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

状态码： 500

表 6-56 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

## 请求示例

查询指定engine\_id的专享版引擎的详细信息。

```
GET /v2/{project_id}/enginemgr/engines/{engine_id}
```

## 响应示例

```
{
  "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",
  "vpclId": "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": [""]
}
```

## 状态码

状态码	描述
200	查询微服务引擎专享版详情成功。
400	错误的请求。
500	服务内部错误。

## 错误码

请参见[CSE错误码](#)。

## 6.3.5 删除微服务引擎专享版

### 功能介绍

删除微服务引擎专享版。

### 调试

您可以在[API Explorer](#)中调试该接口，支持自动认证鉴权。API Explorer可以自动生成SDK代码示例，并提供SDK代码示例调试功能。

### URI

DELETE /v2/{project\_id}/enginemgr/engines/{engine\_id}

表 6-57 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户子项目的唯一标识，字符长度为1~64位，获取方法请参考 <a href="#">获取项目ID</a> 。
engine_id	是	String	微服务引擎专享版ID。

## 请求参数

表 6-58 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。
X-Enterprise-Project-ID	否	String	企业项目id。 如果不配置此参数，则默认企业项目为“default”，ID为0。

## 响应参数

状态码： 200

表 6-59 响应 Body 参数

参数	参数类型	描述
id	String	微服务引擎专享版ID。
name	String	微服务引擎专享版名称。
jobId	Integer	微服务引擎专享版执行任务ID。

状态码： 400

表 6-60 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

状态码： 500

表 6-61 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

## 请求示例

删除指定engine\_id的引擎。

```
DELETE /v2/{project_id}/enginemgr/engines/{engine_id}
```

## 响应示例

```
{  
  "id": "891bf21a-4024-4f47-b38c-bd259ca8f10a",  
  "name": "test",  
  "jobId": 7256  
}
```

## 状态码

状态码	描述
200	删除微服务引擎专享版成功。
400	错误的请求。
500	服务内部错误。

## 错误码

请参见[CSE错误码](#)。

## 6.3.6 查询微服务引擎专享版任务详情

### 功能介绍

查询微服务引擎专享版任务详情。

### 调试

您可以在[API Explorer](#)中调试该接口，支持自动认证鉴权。API Explorer可以自动生成SDK代码示例，并提供SDK代码示例调试功能。

## URI

```
GET /v2/{project_id}/enginemgr/engines/{engine_id}/jobs/{job_id}
```

表 6-62 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户子项目的唯一标识，字符长度为1~64位，获取方法请参考 <a href="#">获取项目ID</a> 。
engine_id	是	String	微服务引擎专享版ID，获取方法请参考 <a href="#">查询微服务引擎专享版列表</a> 。

参数	是否必选	参数类型	描述
job_id	是	String	微服务引擎专享版ID，获取方法请参考 <a href="#">查询微服务引擎专享版详情</a> 。

## 请求参数

表 6-63 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。
X-Enterprise-Project-ID	否	String	企业项目id。 如果不配置此参数，则默认企业项目为“default”，ID为0。

## 响应参数

状态码： 200

表 6-64 响应 Body 体

参数	参数类型	描述
id	Integer	任务ID。
engineId	String	微服务引擎专享版ID。
type	String	任务类型。 Create：创建引擎。 Delete：删除引擎。 Upgrade：升级引擎。 Configure：变更引擎配置。
description	String	任务描述。
status	String	任务当前执行状态。 Init：初始化。 Executing：执行中。 Error：执行失败。 Timeout：执行超时。 Finished：结束。
scheduling	Integer	任务是否正在执行，0表示不在执行，1表示执行中。

参数	参数类型	描述
createUser	String	任务创建者。
startTime	Integer	任务开始执行时间。
endTime	Integer	任务结束时间。
context	String	任务执行上下文。
tasks	Array of <b>TaskSteps</b> objects	任务包含处理阶段。

表 6-65 TaskSteps

参数	参数类型	描述
taskName	String	处理阶段名称。
taskNames	Array of String	当前处理阶段包含的处理步骤名称列表。
status	String	任务处理阶段状态。
startTime	Integer	任务处理阶段开始时间。
endTime	Integer	任务处理阶段结束时间。
taskExecutorBrief	<b>Object</b>	任务元数据。
tasks	Array of <b>Task</b> objects	处理阶段包含的子任务。

表 6-66 TaskExecutorBrief

参数	参数类型	描述
duration	Integer	子任务持续时长。
description	String	子任务描述。

表 6-67 Task

参数	参数类型	描述
jobId	Integer	子任务所属任务ID。
id	Integer	子任务ID, 使用uuid。
type	String	子任务的类型。



参数	参数类型	描述
assigned	String	子任务的执行者。
taskName	String	子任务名称。
engineName	String	子任务所属引擎名称。
taskOrder	Integer	子任务执行的顺序，从小到大。
status	String	子任务状态。
startTime	Integer	子任务开始时间。
endTime	Integer	子任务结束时间。
createTime	Integer	子任务创建时间。
updateTime	Integer	子任务更新时间。
timeout	Integer	子任务是否超时。
log	String	子任务详细信息，执行过程中产生的辅助信息。
output	String	子任务输出信息。
taskExecutorBrief	<b>Object</b>	子任务元数据。

状态码： 400

表 6-68 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

状态码： 500

表 6-69 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

## 请求示例

查询指定engine\_id的专享版引擎下job\_id对应的任务的详情。

```
GET /v2/{project_id}/enginemgr/engines/{engine_id}/jobs/{job_id}
```

## 响应示例

```
{
  "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": "资源准备"
    }
  },
  "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": "租户处理"
    }
  }
  ]
}
```

## 状态码

状态码	描述
200	查询微服务引擎专享版任务详情成功。
400	错误的请求。
500	服务内部错误。

## 错误码

请参见[CSE错误码](#)。

# 7 ServiceComb API

## 7.1 调用说明

CSE提供了REST ( Representational State Transfer ) 风格API，支持您通过HTTPS请求调用。

### 须知

[微服务](#)、[契约](#)、[微服务实例](#)、[依赖关系](#)接口只在华北-北京一、华北-北京四、华北-乌兰察布一、华东-上海一、华东-上海二、华南-广州、西南-贵阳一、中国-香港、亚太-新加坡、拉美-墨西哥城二、中东-利雅得局点支持。

调用微服务引擎专享版的ServiceComb API的方法如下：

1. 登录ServiceStage控制台，选择“微服务引擎 > 引擎列表”。
2. 在页面上方“微服务引擎”下拉列表，选择待调用接口的微服务引擎。  
调用[认证](#)、[微服务](#)、[契约](#)、[微服务实例](#)、[依赖关系](#)接口时，在“服务发现 & 配置”区域，查看或单击  复制“服务注册发现地址”。  
调用[配置管理](#)接口时，在“服务发现 & 配置”区域，查看或单击  复制“配置中心地址”。
3. 参考[如何调用API](#)调用该接口，在[请求URI](#)中，替换{Endpoint}为已获取到的服务注册发现地址。

## 7.2 认证

### 7.2.1 获取微服务引擎专享版用户 Token

#### 功能介绍

仅用于开启了安全认证的微服务引擎专享版。

在访问开启了安全认证的微服务引擎专享版的接口前，需要先调用该接口获取用户Token，该Token是访问接口的唯一凭据。

发送访问请求时需要在请求头中带上该Token，格式为：

```
Authorization:Bearer {Token}
```

## 接口约束

无

## URI

POST /v4/token

## 请求参数

表 7-1 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	用户账号名称。
password	是	String	用户账号密码。

## 响应参数

状态码： 200

表 7-2 响应 Body 参数

参数	参数类型	描述
token	String	获取的用户Token，有效期为12小时。

状态码： 401

表 7-3 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

状态码： 500

表 7-4 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

## 请求示例

POST https://{服务注册发现地址}/v4/token

```
{
  "name": "root",
  "password": "*****"
}
```

## 响应示例

**状态码： 200**

用于获取用户Token的响应结构体。

```
{
  "token" :
  "****bGciOiJSUzUxMiIsInR5cCI6IkpXVCJ9.eyJhY2NvdW50Ijoicm9vdCIsImV4cCI6MTY1MDU5MTcwMSwicm9sZXMiOiSiYWRTaW4iXX0.WKwNAjaYMMCSjNX0qCGCeyh13FJRzLousxoXlThdkMwKH-
  pXEmG51_SguH0LlHOZolc8gNjq-ilQg4bxTo1s0pnQZIS3wma0qvE-
  MzaYnFguTuHM7rxD7eZdwnbUe3dhnw9xRqR1hcd-lTuBbLoL9fbED4U_63loEDyBCJI9D_l0F86uGzpUysCvC-
  t6MrJHgi7miUaO7ZZQmSAUNhmbEoN8IIVp-QtP_cWNWtWaFO-
  eoQrmCT2FdlYiB9MCuELr9-5EGM_mFLPgs6E4fyIGiGHY7IwoGUKOCW5w6Jb0L-2JxeUe3eOI5Md5kzOIAE_EYU
  ATxCbJ5GmgpSSJf*****"
}
```

## 状态码

状态码	描述
200	用于获取用户Token的响应结构体。
401	错误的请求，鉴权失败。
500	服务内部错误

## 错误码

请参见[ServiceComb错误码](#)。

## 7.3 微服务

## 7.3.1 查询指定微服务信息

### 功能介绍

根据service\_id查询微服务定义信息。

### 接口约束

无

### URI

GET /v4/{project\_id}/registry/microservices/{service\_id}

表 7-5 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	<ul style="list-style-type: none"><li>微服务引擎专业版，请填租户项目id，长度为1~64，获取方法请参考<a href="#">获取项目ID</a>。也可以填固定值default。</li><li>微服务引擎专享版，请填固定值：default。</li></ul>
service_id	是	String	微服务唯一标识。字符长度为1~64，正则表达式为 <code>^[.*\$]</code> ，获取方法请参考 <a href="#">查询所有微服务信息</a> 。

### 请求参数

表 7-6 请求 Header 参数

参数	是否必选	参数类型	描述
Authorization	否	String	若微服务引擎专享版开启了安全认证，此参数必填。否则，无此参数。 开启了安全认证的微服务引擎专享版Token，格式为： Authorization:Bearer {Token} Token获取方法，请参考 <a href="#">获取微服务引擎专享版用户Token</a> 。

### 响应参数

状态码： 200

表 7-7 响应 Body 参数

参数	参数类型	描述
service	<b>MicroService</b> object	微服务信息。

表 7-8 MicroService

参数	参数类型	描述
serviceId	String	微服务唯一标识。字符长度为1~64。正则表达式为 $^{\wedge}.*\$$ 。
environment	String	用于区分微服务环境，取值为development、testing、acceptance、production。
appId	String	应用App唯一标识。字符长度为1~160。正则表达式为 $^{\wedge}[a-zA-Z0-9]^{\wedge}[a-zA-Z0-9][a-zA-Z0-9_-.][a-zA-Z0-9]^{\wedge}.$ 。
serviceName	String	微服务名称，同一个App要保证唯一。字符长度为1~128。正则表达式为 $^{\wedge}[a-zA-Z0-9]^{\wedge}[a-zA-Z0-9][a-zA-Z0-9_-.][a-zA-Z0-9]^{\wedge}.$ 。
version	String	微服务版本号。字符长度为1~64。正则表达式为 $^{\wedge}[0-9]^{\wedge}[0-9]+.(.[0-9]+)^{\wedge}.$ 。
description	String	微服务描述信息。字符长度不超过256。
level	String	微服务层级：FRONT、MIDDLE、BACK。
registerBy	String	微服务注册方式：SDK、PLATFORM、SIDECAR、UNKNOWN。
schemas	Array of strings	微服务访问的契约内容。支持数字、字母，支持使用括号内字符做连接符(_-.)，长度1-160字节，数组长度最大100个契约。
status	String	微服务状态，UP表示上线，DOWN表示下线，默认值UP。 枚举值： <ul style="list-style-type: none"><li>• UP</li><li>• DOWN</li></ul>
timestamp	String	微服务注册时间。
modTimestamp	String	最后修改UTC时间。
framework	<b>Framework</b> object	开发框架信息。



参数	参数类型	描述
paths	Array of <a href="#">ServicePath</a> objects	服务路由信息。

表 7-9 Framework

参数	参数类型	描述
name	String	微服务开发框架，默认值为UNKNOWN。
version	String	微服务开发框架版本号。

表 7-10 ServicePath

参数	参数类型	描述
Path	String	路由地址。
Property	Object	微服务扩展属性，可以自定义KEY和相应的Value。长度最小1字节。

状态码： 400

表 7-11 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

状态码： 500

表 7-12 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

## 请求示例

查询ID为819706e21b7173306797d19922ce4231441c17c5的微服务的详细信息。

```
GET https://{endpoint}/v4/{project_id}/registry/microservices/  
819706e21b7173306797d19922ce4231441c17c5
```

## 响应示例

**状态码： 200**

查询成功。

```
{  
  "service": {  
    "serviceId": "819706e21b7173306797d19922ce4231441c17c5",  
    "appId": "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"  
  }  
}
```

## 状态码

状态码	描述
200	查询成功。
400	错误的请求。
500	服务内部错误

## 错误码

请参见[ServiceComb错误码](#)。

## 7.3.2 删除微服务定义信息

### 功能介绍

删除一个微服务定义及其相关信息，同时注销其所有实例信息。

### 接口约束

无

### URI

```
DELETE /v4/{project_id}/registry/microservices/{service_id}
```

表 7-13 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	<ul style="list-style-type: none"> <li>微服务引擎专业版，请填写租户项目id，长度为1~64，获取方法请参考<a href="#">获取项目ID</a>。也可以填固定值default。</li> <li>微服务引擎专享版，请填写固定值：default。</li> </ul>
service_id	是	String	微服务唯一标识。字符长度为1~64。正则表达式为 <code>^[a-z0-9]*\$</code> ，获取方法请参考 <a href="#">查询所有微服务信息</a> 。

表 7-14 Query 参数

参数	是否必选	参数类型	描述
force	否	Boolean	<p>是否强制删除。</p> <p>true表示强制删除，false表示非强制删除。</p> <p>选择强制删除会自动注销所有服务实例，并且删除相关的服务依赖关系。如果未传此参数，那么在服务拥有实例时，无法被删除。</p> <p>缺省值：false</p>

## 请求参数

表 7-15 请求 Header 参数

参数	是否必选	参数类型	描述
Authorization	否	String	<p>若微服务引擎专享版开启了安全认证，此参数必填。否则，无此参数。</p> <p>开启了安全认证的微服务引擎专享版Token，格式为： Authorization:Bearer {Token}</p> <p>Token获取方法，请参考<a href="#">获取微服务引擎专享版用户Token</a>。</p>

## 响应参数

状态码： 400

表 7-16 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

状态码： 500

表 7-17 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

## 请求示例

删除一个ID为e0f0da073f2c91e8979a89ff2d7c69t6的微服务。

```
DELETE https://{endpoint}/v4/d9f4da085f2c11e8959a00ff2d7c69b7/registry/microservices/  
e0f0da073f2c91e8979a89ff2d7c69t6
```

## 响应示例

无

## 状态码

状态码	描述
200	删除成功。
400	错误的请求。
500	服务内部错误。

## 错误码

请参见[ServiceComb错误码](#)。

### 7.3.3 查询所有微服务信息

#### 功能介绍

根据条件组合，查询满足所有条件的微服务定义信息。

#### 接口约束

无

#### URI

GET /v4/{project\_id}/registry/microservices

表 7-18 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	<ul style="list-style-type: none"><li>微服务引擎专业版，请填写租户项目id，长度为1~64，获取方法请参考<a href="#">获取项目ID</a>。也可以填固定值default。</li><li>微服务引擎专享版，请填写固定值：default。</li></ul>

#### 请求参数

表 7-19 请求 Header 参数

参数	是否必选	参数类型	描述
Authorization	否	String	<p>若微服务引擎专享版开启了安全认证，此参数必填。否则，无此参数。</p> <p>开启了安全认证的微服务引擎专享版Token，格式为： Authorization:Bearer {Token}</p> <p>Token获取方法，请参考<a href="#">获取微服务引擎专享版用户Token</a>。</p>

#### 响应参数

状态码： 200

表 7-20 响应 Body 参数

参数	参数类型	描述
services	Array of <b>MicroService</b> objects	微服务列表。

表 7-21 MicroService

参数	参数类型	描述
serviceId	String	微服务唯一标识。字符长度为1~64。正则表达式为 $^{\wedge}.*\$$ 。
environment	String	用于区分微服务环境，取值为development、testing、acceptance、production。当配置为development、testing或acceptance时，可以通过批量上传schemas接口新增或者修改已存在的Schema；当配置为production时，则不可以新增或者修改Schema。默认值development。
appId	String	应用App唯一标识。字符长度为1~160。正则表达式为 $^{\wedge}[a-zA-Z0-9]\$ ^{\wedge}[a-zA-Z0-9][a-zA-Z0-9_-.][a-zA-Z0-9]\$$ 。
serviceName	String	微服务名称，同一个App要保证唯一。字符长度为1~128。正则表达式为 $^{\wedge}[a-zA-Z0-9]\$ ^{\wedge}[a-zA-Z0-9][a-zA-Z0-9_-.][a-zA-Z0-9]\$$ 。
version	String	微服务版本号。字符长度为1~64。正则表达式为 $^{\wedge}[0-9]\$ ^{\wedge}[0-9]+(\.[0-9]+)\$$ 。
description	String	微服务描述信息。字符长度不超过256。
level	String	微服务层级：FRONT、MIDDLE、BACK。
registerBy	String	微服务注册方式：SDK、PLATFORM、SIDECAR、UNKNOWN。
schemas	Array of strings	微服务访问的契约内容。支持数字、字母，支持使用括号内字符做连接符(_.)，长度1-160字节，数组长度最大100个契约。
status	String	微服务状态，UP表示上线，DOWN表示下线，默认值UP。
timestamp	String	微服务注册时间。
modTimestamp	String	最后修改UTC时间。
framework	<b>Framework</b> object	开发框架信息。

参数	参数类型	描述
paths	Array of <a href="#">ServicePath</a> objects	服务路由信息。

表 7-22 Framework

参数	参数类型	描述
name	String	微服务开发框架，默认值为UNKNOWN。
version	String	微服务开发框架版本号。

表 7-23 ServicePath

参数	参数类型	描述
Path	String	路由地址。
Property	Object	微服务扩展属性，可以自定义KEY和相应的Value。长度最小1字节。

状态码： 400

表 7-24 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

状态码： 500

表 7-25 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

## 请求示例

查询当前注册的所有微服务信息。

```
GET https://{endpoint}/v4/{project_id}/registry/microservices
```

## 响应示例

**状态码： 200**

查询成功。

```
{
  "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"
    }
  ]
}
```

## 状态码

状态码	描述
200	查询成功。
400	错误的请求。
500	服务内部错误

## 错误码

请参见[ServiceComb错误码](#)。

## 7.3.4 创建微服务静态信息

### 功能介绍

在注册微服务实例前需要创建服务静态信息，之后注册的微服务实例根据serviceId这个字段与静态信息关联，一个服务对应对多个实例。



serviceld可以定制。如果定制了，系统将使用定制的serviceld；如果没有定制，系统会随机生成serviceld。

## 接口约束

无

## URI

POST /v4/{project\_id}/registry/microservices

表 7-26 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	<ul style="list-style-type: none"><li>微服务引擎专业版，请填写租户项目id，长度为1~64，获取方法请参考<a href="#">获取项目ID</a>。也可以填固定值default。</li><li>微服务引擎专享版，请填写固定值：default。</li></ul>

## 请求参数

表 7-27 请求 Header 参数

参数	是否必选	参数类型	描述
Authorization	否	String	若微服务引擎专享版开启了安全认证，此参数必填。否则，无此参数。 开启了安全认证的微服务引擎专享版Token，格式为： Authorization:Bearer {Token} Token获取方法，请参考 <a href="#">获取微服务引擎专享版用户Token</a> 。

表 7-28 请求 Body 参数

参数	是否必选	参数类型	描述
service	是	<b>MicroService</b> object	微服务信息。
rules	否	Array of <b>Rule</b> objects	黑白名单信息。

参数	是否必选	参数类型	描述
instances	否	Array of <b>MicroServiceInstance</b> objects	实例信息。
tags	否	Object	微服务扩展属性，可以自定义KEY和相应的Value。长度最小1字节。

表 7-29 MicroService

参数	是否必选	参数类型	描述
serviceId	否	String	微服务唯一标识。字符长度为1~64。正则表达式为 $^.*\$$ 。
environment	否	String	用于区分微服务环境，取值为development、testing、acceptance、production。当配置为development、testing或acceptance时，可以通过批量上传schemas接口新增或者修改已存在的Schema；当配置为production时，则不可以新增或者修改Schema。默认值development。
appId	否	String	应用App唯一标识。字符长度为1~160。正则表达式为 $^[a-zA-Z0-9]^\{a-zA-Z0-9\}[a-zA-Z0-9_\.][a-zA-Z0-9]^\$$ 。
serviceName	是	String	微服务名称，同一个App要保证唯一。字符长度为1~128。正则表达式为 $^[a-zA-Z0-9]^\{a-zA-Z0-9\}[a-zA-Z0-9_\.][a-zA-Z0-9]^\$$ 。
version	否	String	微服务版本号。字符长度为1~64。正则表达式为 $^[0-9]^\{0-9\}^\{0-9\}^\{0-9\}^\$$ 。
description	否	String	微服务描述信息。字符长度不超过256。
level	否	String	微服务层级：FRONT、MIDDLE、BACK。
registerBy	否	String	微服务注册方式：SDK、PLATFORM、SIDECAR、UNKNOWN。

参数	是否必选	参数类型	描述
schemas	否	Array of strings	微服务访问契约内容的外键ID，数组长度最大100个契约。
status	否	String	微服务状态，UP表示上线，DOWN表示下线，默认值UP。
timestamp	否	String	微服务注册时间。
modTimestamp	否	String	最后修改UTC时间。
framework	否	Framework object	开发框架信息。
paths	否	Array of ServicePath objects	服务路由信息。

表 7-30 Framework

参数	是否必选	参数类型	描述
name	否	String	微服务开发框架，默认值为UNKNOWN。
version	否	String	微服务开发框架版本号。

表 7-31 ServicePath

参数	是否必选	参数类型	描述
Path	否	String	路由地址。
Property	否	Object	微服务扩展属性，可以自定义KEY和相应的Value。长度最小1字节。

表 7-32 Rule

参数	是否必选	参数类型	描述
ruleId	否	String	自定义ruleId。
ruleType	否	String	rule类型：WHITE或者BLACK。

参数	是否必选	参数类型	描述
attribute	否	String	如果是tag_xxx开头，则按Tag过滤attribute属性，否则，则按"serviceld", "Appld", "ServiceName", "Version", "Description", "Level", "Status"过滤。
pattern	否	String	匹配规则，正则表达式，长度1到64。
description	否	String	rule描述。
timestamp	否	String	只有获取rule时返回使用，创建rule的时间。
modTimestamp	否	String	更新时间。

表 7-33 MicroServiceInstance

参数	是否必选	参数类型	描述
instanceId	否	String	实例id，唯一标识。创建实例，instanceId由service-center产生。
serviceld	否	String	微服务唯一标识。创建实例时，以url里面的为准，不用这里的serviceld。
version	否	String	微服务版本号。
hostName	是	String	主机信息。
endpoints	是	Array of strings	访问地址信息。
status	否	String	实例状态：UP、DOWN、STARTING、OUTOFSERVICE，默认值UP。
properties	否	Object	微服务扩展属性，可以自定义KEY和相应的Value。长度最小1字节。
healthCheck	否	HealthCheck object	健康检查信息。
dataCenterInfo	否	DataCenterInfo object	数据中心信息。
timestamp	否	String	实例创建时间戳，自动生成。

参数	是否必选	参数类型	描述
modTimestamp	否	String	更新时间。

表 7-34 HealthCheck

参数	是否必选	参数类型	描述
mode	是	String	心跳模式： push或者pull。
port	否	Integer	端口。
interval	是	Integer	心跳间隔（秒），当值小于5秒时，按5秒注册。
times	是	Integer	最大尝试请求次数。

表 7-35 DataCenterInfo

参数	是否必选	参数类型	描述
name	是	String	区域名称。
region	是	String	区域。
availableZone	是	String	可用区。

## 响应参数

状态码： 200

表 7-36 响应 Body 参数

参数	参数类型	描述
serviceId	String	微服务唯一标识。

状态码： 400

表 7-37 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。

参数	参数类型	描述
detail	String	详细定位信息。

状态码： 500

表 7-38 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

## 请求示例

创建一个微服务的静态信息，其微服务名为test，微服务版本为1.0.0，rule的类型为白名单，实例的主机信息为instanceTest，访问地址为rest:127.0.0.1:8080。

POST https://{endpoint}/v4/{project\_id}/registry/microservices

```
{
  "service": {
    "appId": "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"
  }
}
```

## 响应示例

状态码： 200

创建成功。serviceId为该微服务生成的ID。后续的操作都是针对该微服务ID进行操作。

```
{
  "serviceId" : "8aed80ea052ac04a64dfc79c24f2170224d074f5"
}
```

## 状态码

状态码	描述
200	创建成功。
400	错误的请求。
500	服务内部错误。

## 错误码

请参见[ServiceComb错误码](#)。

## 7.3.5 批量删除微服务静态信息

### 功能介绍

批量删除指定的微服务定义及其相关信息，同时注销其所有实例信息。

### 接口约束

无

### URI

DELETE /v4/{project\_id}/registry/microservices

表 7-39 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	<ul style="list-style-type: none"><li>微服务引擎专业版，请填租户项目id，长度为1~64，获取方法请参考<a href="#">获取项目ID</a>。也可以填固定值default。</li><li>微服务引擎专享版，请填固定值：default。</li></ul>

表 7-40 Query 参数

参数	是否必选	参数类型	描述
force	否	Boolean	是否强制删除。 true表示强制删除，false表示非强制删除。 选择强制删除会自动注销所有服务实例，并且删除相关的服务依赖关系。如果未传此参数，那么在服务拥有实例时，无法被删除。 缺省值：false。

## 请求参数

表 7-41 请求 Header 参数

参数	是否必选	参数类型	描述
Authorization	否	String	若微服务引擎专享版开启了安全认证，此参数必填。否则，无此参数。 开启了安全认证的微服务引擎专享版Token，格式为： Authorization:Bearer {Token} Token获取方法，请参考 <a href="#">获取微服务引擎专享版用户Token</a> 。

表 7-42 请求 Body 参数

参数	是否必选	参数类型	描述
serviceIds	是	Array of strings	服务ID列表，获取方法请参考 <a href="#">查询所有微服务信息</a> 。

## 响应参数

状态码： 200



表 7-43 响应 Body 参数

参数	参数类型	描述
services	Array of <a href="#">DelServicesRsplInfo</a> objects	需要删除的服务信息列表。

表 7-44 DelServicesRsplInfo

参数	参数类型	描述
serviceId	String	微服务id。

状态码： 400

表 7-45 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

状态码： 500

表 7-46 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

## 请求示例

删除微服务ID为id1和id2的微服务的静态信息。

```
DELETE https://{endpoint}/v4/{project_id}/registry/microservices
{
  "serviceIds" : [ "id1", "id2" ]
}
```

## 响应示例

**状态码： 200**

删除成功。

```
{
  "services": [ {
    "serviceId": "id1"
  },
  {
    "serviceId": "id2"
  }
]
}
```

## 状态码

状态码	描述
200	删除成功。
400	错误的请求。
500	服务内部错误。

## 错误码

请参见[ServiceComb错误码](#)。

## 7.3.6 修改微服扩展属性信息

### 功能介绍

创建微服务静态信息后可对服务部分字段进行更新，每次更新都需要传入完整的服务静态信息json，即便不更新部分的字段也要作为json的属性传入。

### 接口约束

无

### URI

PUT /v4/{project\_id}/registry/microservices/{service\_id}/properties

表 7-47 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	<ul style="list-style-type: none"><li>微服务引擎专业版，请填写租户项目id，长度为1~64，获取方法请参考<a href="#">获取项目ID</a>。也可以填固定值 default。</li><li>微服务引擎专享版，请填写固定值：default。</li></ul>
service_id	是	String	微服务唯一标识。字符长度为1~64，正则表达式为 $^{\wedge}.*\$$ ，获取方法请参考 <a href="#">查询所有微服务信息</a> 。

## 请求参数

表 7-48 请求 Header 参数

参数	是否必选	参数类型	描述
Authorization	否	String	若微服务引擎专享版开启了安全认证，此参数必填。否则，无此参数。 开启了安全认证的微服务引擎专享版Token，格式为： Authorization:Bearer {Token} Token获取方法，请参考 <a href="#">获取微服务引擎专享版用户Token</a> 。

表 7-49 请求 Body 参数

参数	是否必选	参数类型	描述
properties	是	Object	微服务扩展属性，可以自定义KEY和相应的Value。长度最小1字节。

表 7-50 Framework

参数	是否必选	参数类型	描述
name	否	String	微服务开发框架，默认值为UNKNOWN。
version	否	String	微服务开发框架版本号。

## 响应参数

状态码： 400

表 7-51 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

状态码： 500

表 7-52 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

## 请求示例

修改指定service\_id的微服务的扩展信息。

```
PUT https://{endpoint}/v4/{project_id}/registry/microservices/{service_id}/properties
```

```
{  
  "properties": {  
    "a": "test"  
  }  
}
```

## 响应示例

无

## 状态码

状态码	描述
200	修改成功。
400	错误的请求。

状态码	描述
500	服务内部错误。

## 错误码

请参见[ServiceComb错误码](#)。

## 7.3.7 查询微服务唯一标识信息

### 功能介绍

可通过指定条件，查询微服务serviceId或schema的唯一标识信息。

### URI

GET /v4/{project\_id}/registry/existence

表 7-53 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	<ul style="list-style-type: none"><li>微服务引擎专业版，请填租户项目id，长度为1~64，获取方法请参考<a href="#">获取项目ID</a>。也可以填固定值default。</li><li>微服务引擎专享版，请填固定值：default。</li></ul>

表 7-54 Query 参数

参数	是否必选	参数类型	描述
type	是	String	请求类型： <ul style="list-style-type: none"><li>microservice</li><li>schema</li></ul> <b>说明</b> <ul style="list-style-type: none"><li>若type为microservice，参数env、appId、serviceName、version必填。</li><li>若type为schema，参数serviceId、schemaId必填。</li></ul>
env	否	String	微服务的环境信息： development、testing、 acceptance、production。

参数	是否必选	参数类型	描述
appId	是	String	资源类型为microservice时需传入应用app唯一标识。字符长度为1~160。正则表达式为^[a-zA-Z0-9]\$^[a-zA-Z0-9][a-zA-Z0-9_-.][a-zA-Z0-9]\$。
serviceName	是	String	资源类型为microservice时需传入微服务名称。字符长度为1~128。正则表达式为^[a-zA-Z0-9]\$^[a-zA-Z0-9][a-zA-Z0-9_-.][a-zA-Z0-9]\$。
version	是	String	资源类型为microservice时需传入微服务版本。字符长度为1~64。正则表达式为^[0-9]\$^[0-9]+([0-9]+)\$。
serviceId	是	String	资源类型为schema时需传入微服务唯一标识。字符长度为1~64。正则表达式为^.*\$。
schemald	是	String	资源类型为schema时需传入schema唯一标识。字符长度为1~160。正则表达式为^[a-zA-Z0-9]{1,160}\$^[a-zA-Z0-9][a-zA-Z0-9_-.]{0,158}[a-zA-Z0-9]\$。

## 请求参数

表 7-55 请求 Header 参数

参数	是否必选	参数类型	描述
Authorization	否	String	<p>若微服务引擎专享版开启了安全认证，此参数必填。否则，无此参数。</p> <p>开启了安全认证的微服务引擎专享版Token，格式为： Authorization:Bearer {Token}</p> <p>Token获取方法，请参考<a href="#">获取微服务引擎专享版用户Token</a>。</p>

## 响应参数

状态码： 200

表 7-56 响应 Header 参数

参数	参数类型	描述
X-Schema-Summary	String	契约的摘要。

表 7-57 响应 Body 参数

参数	参数类型	描述
serviceId	String	查询实体为服务时，返回服务ID。
schemald	String	查询实体为契约时，返回契约ID。

状态码： 400

表 7-58 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

状态码： 500

表 7-59 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

## 请求示例

查询微服务serviceId，指定的查询条件是请求类型为microservice，应用id为default，微服务名称为service，微服务版本为1.0.0。

```
GET https://{endpoint}/v4/{project_id}/registry/existence?  
type=microservice&appId=default&serviceName=service&version=1.0.0
```

## 响应示例

状态码： 200

查询成功，则返回serviceId或者schemald。

```
{  
  "serviceId" : "8aed80ea052ac04a64dfc79c24f2170224d074f5"  
}
```

## 状态码

状态码	描述
200	查询成功。
400	错误的请求。
500	服务内部错误。

## 错误码

请参见[ServiceComb错误码](#)。

## 7.4 契约

### 7.4.1 查询微服务契约

#### 功能介绍

根据service\_id和schema\_id查询微服务的契约信息。

#### 接口约束

无

#### URI

GET /v4/{project\_id}/registry/microservices/{service\_id}/schemas/{schema\_id}

表 7-60 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	<ul style="list-style-type: none"><li>微服务引擎专业版，请填租户项目id，长度为1~64，获取方法请参考<a href="#">获取项目ID</a>。也可以填固定值default。</li><li>微服务引擎专享版，请填固定值：default。</li></ul>



参数	是否必选	参数类型	描述
service_id	是	String	微服务唯一标识。字符长度为1~64，正则表达式为 $\wedge.*\$$ 。获取方法请参考 <a href="#">查询所有微服务信息</a> 。
schema_id	是	String	契约的id。获取方法请参考 <a href="#">查询微服务的所有契约</a> 。

## 请求参数

表 7-61 请求 Header 参数

参数	是否必选	参数类型	描述
Authorization	否	String	若微服务引擎专享版开启了安全认证，此参数必填。否则，无此参数。 开启了安全认证的微服务引擎专享版Token，格式为： Authorization:Bearer {Token} Token获取方法，请参考 <a href="#">获取微服务引擎专享版用户Token</a> 。

## 响应参数

状态码： 200

表 7-62 响应 Header 参数

参数	参数类型	描述
X-Schema-Summary	String	微服务契约的摘要。

表 7-63 响应 Body 参数

参数	参数类型	描述
schema	String	契约内容。

状态码： 400

表 7-64 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

状态码： 500

表 7-65 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

## 请求示例

查询指定service\_id和schema\_id的契约的详细信息。

```
GET https://{endpoint}/v4/{project_id}/registry/microservices/{service_id}/schemas/{schema_id}
```

## 响应示例

状态码： 200

查询成功，如果summary存在，则header里面的X-Schema-Summary的value为该schema对应的摘要。

```
{
  "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\"\nbasePath: \"/provider\"\nconsumes:\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}
```

## 状态码

状态码	描述
200	查询成功。
400	错误的请求。
500	服务内部错误。

## 错误码

请参见[ServiceComb错误码](#)。

## 7.4.2 修改微服务契约

### 功能介绍

根据schema\_id更新微服务的契约内容。

### 接口约束

服务中心老版本如果微服务环境定为production，不支持契约覆盖；从版本2.3.33开始，支持使用环境变量SCHEMA\_EDITABLE支持契约覆盖。

### URI

PUT /v4/{project\_id}/registry/microservices/{service\_id}/schemas/{schema\_id}

表 7-66 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	<ul style="list-style-type: none"><li>微服务引擎专业版，请填写租户项目id，长度为1~64，获取方法请参考<a href="#">获取项目ID</a>。也可以填固定值default。</li><li>微服务引擎专享版，请填写固定值：default。</li></ul>
service_id	是	String	微服务唯一标识。字符长度为1~64，正则表达式为 $^.*\$$ 。获取方法请参考 <a href="#">查询所有微服务信息</a> 。
schema_id	是	String	微服务契约唯一标识。字符长度为1~160，正则表达式为 $^[a-zA-Z0-9]{1,160}\$ ^[a-zA-Z0-9][a-zA-Z0-9_-]{0,158}[a-zA-Z0-9]\$$ 。获取方法请参考 <a href="#">查询微服务的所有契约</a> 。

## 请求参数

表 7-67 请求 Header 参数

参数	是否必选	参数类型	描述
Authorization	否	String	若微服务引擎专享版开启了安全认证，此参数必填。否则，无此参数。 开启了安全认证的微服务引擎专享版Token，格式为： Authorization:Bearer {Token} Token获取方法，请参考 <a href="#">获取微服务引擎专享版用户Token</a> 。

表 7-68 请求 Body 参数

参数	是否必选	参数类型	描述
schema	是	String	契约内容。
summary	否	String	新加入参数，后面创建schema，请尽量提供schema的摘要。

## 响应参数

状态码： 400

表 7-69 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

状态码： 500

表 7-70 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。

参数	参数类型	描述
detail	String	详细定位信息。

## 请求示例

修改指定service\_id和schema\_id的契约内容和摘要。

```
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        200:\n        description:\n        \"response of 200\"\n        schema:\n        type: \"string\"\n        ,\n        \"summary\": \"test\"\n}
```

## 响应示例

无

## 状态码

状态码	描述
200	修改成功。
400	错误的请求。
500	服务内部错误。

## 错误码

请参见[ServiceComb错误码](#)。

## 7.4.3 查询微服务的所有契约

### 功能介绍

查询一个微服务的所有契约信息，包括schemaId和summary信息。

### URI

GET /v4/{project\_id}/registry/microservices/{service\_id}/schemas

表 7-71 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	<ul style="list-style-type: none"><li>• 微服务引擎专业版，请填写租户项目id，长度为1~64，获取方法请参考<a href="#">获取项目ID</a>。也可以填固定值default。</li><li>• 微服务引擎专享版，请填写固定值：default。</li></ul>
service_id	是	String	微服务唯一标识。字符长度为1~64，正则表达式为 <code>^[.*\$]</code> 。获取方法请参考 <a href="#">查询所有微服务信息</a> 。

表 7-72 Query 参数

参数	是否必选	参数类型	描述
withSchema	否	Integer	是否查询schema内容，默认为0。 0表示不显示schema内容，只有schema_id和summary。 1表示显示schema_id、summary和schema。

## 请求参数

表 7-73 请求 Header 参数

参数	是否必选	参数类型	描述
Authorization	否	String	若微服务引擎专享版开启了安全认证，此参数必填。否则，无此参数。 开启了安全认证的微服务引擎专享版Token，格式为： Authorization:Bearer {Token} Token获取方法，请参考 <a href="#">获取微服务引擎专享版用户Token</a> 。

## 响应参数

状态码： 200

表 7-74 响应 Body 参数

参数	参数类型	描述
schemas	Array of <a href="#">Schema</a> objects	契约列表。

表 7-75 Schema

参数	参数类型	描述
schemald	String	契约id。
schema	String	契约内容。
summary	String	契约的Summary。

状态码： 400

表 7-76 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

状态码： 500

表 7-77 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

## 请求示例

查询指定service\_id的微服务的契约信息，显示schema\_id、summary和schema信息。

```
GET /v4/{project_id}/registry/microservices/{service_id}/schemas?withSchema=1
```

## 响应示例

状态码： 200

查询成功。

```
{
  "schemas": [{
    "schemaId": "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\"\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            \"xxxx\",
          \"summary\": \"abcda7b4072ef2d7a5fc9aefccf03e5548029ae31c6cd5fc29da7685d6d9e14adea3\"
        }
  ]
}
```

## 状态码

状态码	描述
200	查询成功。
400	错误的请求。
500	服务内部错误。

## 错误码

请参见[ServiceComb错误码](#)。

## 7.5 微服务实例

### 7.5.1 注册微服务实例

#### 功能介绍

创建微服务后就可以注册该微服务的实例。

注册微服务实例时，需提供该微服务实例相关的信息。

instanceId可定制，如果定制了，再次注册就直接全内容覆盖；如果没有定制，系统自动生成id，如果endpoints内容重复，则使用原来的id。

#### URI

POST /v4/{project\_id}/registry/microservices/{service\_id}/instances



表 7-78 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	<ul style="list-style-type: none"><li>微服务引擎专业版，请填写租户项目id，长度为1~64，获取方法请参考<a href="#">获取项目ID</a>。也可以填固定值default。</li><li>微服务引擎专享版，请填写固定值：default。</li></ul>
service_id	是	String	微服务唯一标识。字符长度为1~64，正则表达式为 <code>^[.*\$]</code> 。获取方法请参考 <a href="#">查询所有微服务信息</a> 。

## 请求参数

表 7-79 请求 Header 参数

参数	是否必选	参数类型	描述
Authorization	否	String	若微服务引擎专享版开启了安全认证，此参数必填。否则，无此参数。 开启了安全认证的微服务引擎专享版Token，格式为： Authorization:Bearer {Token} Token获取方法，请参考 <a href="#">获取微服务引擎专享版用户Token</a> 。

表 7-80 请求 Body 参数

参数	是否必选	参数类型	描述
instance	是	<a href="#">MicroServiceInstance</a> object	微服务实例信息。

表 7-81 MicroServiceInstance

参数	是否必选	参数类型	描述
instanceId	否	String	实例id，唯一标识。创建实例，instanceId由service-center产生。

参数	是否必选	参数类型	描述
serviceId	否	String	微服务唯一标识，创建实例时，以url里面的为准，不用这里的serviceId。
version	否	String	微服务版本号。
hostName	是	String	主机信息。
endpoints	否	Array of strings	访问地址信息。
status	否	String	实例状态：UP、DOWN、STARTING、OUTOFSERVICE。 默认值UP。
properties	否	Object	微服务扩展属性，可以自定义KEY和相应的Value。长度最小1字节。
healthCheck	否	HealthCheck object	健康检查信息。
dataCenterInfo	否	DataCenterInfo object	数据中心信息。
timestamp	否	String	实例创建时间戳，自动生成。
modTimestamp	否	String	更新时间。

表 7-82 HealthCheck

参数	是否必选	参数类型	描述
mode	是	String	心跳模式push/pull。
port	否	Integer	端口。
interval	是	Integer	心跳间隔（秒），当值小于5秒时，按5秒注册。
times	是	Integer	最大尝试请求次数。

表 7-83 DataCenterInfo

参数	是否必选	参数类型	描述
name	是	String	区域名字。
region	是	String	区域。

参数	是否必选	参数类型	描述
availableZone	是	String	可用区。

## 响应参数

状态码： 200

表 7-84 响应 Body 参数

参数	参数类型	描述
instanceId	String	实例ID。

状态码： 400

表 7-85 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

状态码： 500

表 7-86 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

## 请求示例

注册一个微服务实例，其主机信息为localhost，健康检查的心跳模式为push，心跳间隔为30s，所在region为r1，名字为dc，可用区为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
}
}
```

## 响应示例

**状态码： 200**

注册成功。

```
{
  "instanceId": "8540bb8b693c4ad1a7fb6a756c415244"
}
```

## 状态码

状态码	描述
200	注册成功。
400	错误的请求。
500	服务内部错误。

## 错误码

请参见[ServiceComb错误码](#)。

## 7.5.2 根据 service\_id 查询微服务实例

### 功能介绍

实例注册后可以根据service\_id发现该微服务的所有实例。

### URI

GET /v4/{project\_id}/registry/microservices/{service\_id}/instances

表 7-87 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	<ul style="list-style-type: none"><li>微服务引擎专业版，请填写租户项目id，长度为1~64，获取方法请参考<a href="#">获取项目ID</a>。也可以填固定值default。</li><li>微服务引擎专享版，请填写固定值：default。</li></ul>
service_id	是	String	微服务唯一标识。字符长度为1~64，正则表达式为 <code>^.*\$</code> 。获取方法请参考 <a href="#">查询所有微服务信息</a> 。

表 7-88 Query 参数

参数	是否必选	参数类型	描述
tags	否	String	Tag标签过滤，多个时逗号分隔。正则表达式为 <code>^[a-zA-Z][a-zA-Z0-9_-]{0,63}\$</code> 。

## 请求参数

表 7-89 请求 Header 参数

参数	是否必选	参数类型	描述
X-ConsumerId	否	String	微服务消费者的微服务唯一标识。
Authorization	否	String	若微服务引擎专享版开启了安全认证，此参数必填。否则，无此参数。 开启了安全认证的微服务引擎专享版Token，格式为： Authorization:Bearer {Token} Token获取方法，请参考 <a href="#">获取微服务引擎专享版用户Token</a> 。

## 响应参数

状态码： 200

表 7-90 响应 Body 参数

参数	参数类型	描述
instances	Array of <b>MicroServiceInstance</b> objects	实例列表。

表 7-91 MicroServiceInstance

参数	参数类型	描述
instanceId	String	实例id, 唯一标识。创建实例, instanceId由 service-center产生。
serviceId	String	微服务唯一标识, 创建实例时, 以url里面的为准, 不用这里的serviceId。
version	String	微服务版本号。
hostName	String	主机信息。
endpoints	Array of strings	访问地址信息。
status	String	实例状态: UP、DOWN、STARTING、OUTOFSERVICE。默认值UP。
properties	Object	微服务扩展属性, 可以自定义KEY和相应的Value。长度最小1字节。
healthCheck	<b>HealthCheck</b> object	健康检查信息。
dataCenterInfo	<b>DataCenterInfo</b> object	数据中心信息。
timestamp	String	实例创建时间戳, 自动生成。
modTimestamp	String	更新时间。

表 7-92 HealthCheck

参数	参数类型	描述
mode	String	心跳模式push/pull。
port	Integer	端口。
interval	Integer	心跳间隔(秒), 当值小于5秒时, 按5秒注册。
times	Integer	最大尝试请求次数。

表 7-93 DataCenterInfo

参数	参数类型	描述
name	String	区域名字。
region	String	区域。
availableZone	String	可用区。

状态码： 400

表 7-94 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

状态码： 500

表 7-95 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

## 请求示例

查询指定service\_id的微服务的实例信息。

```
GET https://{endpoint}/v4/{project_id}/registry/microservices/{service_id}/instances
```

## 响应示例

状态码： 200

查询成功。

```
{
  "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"  
  }  
]  
}
```

## 状态码

状态码	描述
200	查询成功。
400	错误的请求。
500	服务内部错误。

## 错误码

请参见[ServiceComb错误码](#)。

## 7.5.3 注销微服务实例

### 功能介绍

实例注册后可以根据instance\_id进行实例注销。

### URI

DELETE /v4/{project\_id}/registry/microservices/{service\_id}/instances/{instance\_id}

表 7-96 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	<ul style="list-style-type: none"><li>微服务引擎专业版，请填写租户项目id，长度为1~64，获取方法请参考<a href="#">获取项目ID</a>。也可以填固定值default。</li><li>微服务引擎专享版，请填写固定值：default。</li></ul>



参数	是否必选	参数类型	描述
service_id	是	String	微服务实例唯一标识。字符长度不超过64位，正则表达式为 <code>^[A-Za-z0-9_-]*\$</code> 。获取方法请参考 <a href="#">根据service_id查询微服务实例</a> 。
instance_id	是	String	微服务实例唯一标识。字符长度不超过64位，正则表达式为 <code>^[A-Za-z0-9_-]*\$</code> 。获取方法请参考 <a href="#">根据service_id查询微服务实例</a> 。

## 请求参数

表 7-97 请求 Header 参数

参数	是否必选	参数类型	描述
Authorization	否	String	若微服务引擎专享版开启了安全认证，此参数必填。否则，无此参数。 开启了安全认证的微服务引擎专享版Token，格式为： Authorization:Bearer {Token} Token获取方法，请参考 <a href="#">获取微服务引擎专享版用户Token</a> 。

## 响应参数

状态码： 400

表 7-98 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

状态码： 500

表 7-99 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

## 请求示例

注销指定instance\_id的实例。

```
DELETE https://{endpoint}/v4/{project_id}/registry/microservices/{service_id}/instances/{instance_id}
```

## 响应示例

无

## 状态码

状态码	描述
200	注销成功。
400	错误的请求。
500	服务内部错误。

## 错误码

请参见[ServiceComb错误码](#)。

## 7.5.4 查询微服务实例详细信息

### 功能介绍

实例注册后可以根据service\_id和instance\_id获取该实例的详细信息。

### URI

```
GET /v4/{project_id}/registry/microservices/{service_id}/instances/{instance_id}
```

表 7-100 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	<ul style="list-style-type: none"><li>微服务引擎专业版，请填写租户项目id，长度为1~64，获取方法请参考<a href="#">获取项目ID</a>。也可以填固定值default。</li><li>微服务引擎专享版，请填写固定值：default。</li></ul>
service_id	是	String	微服务唯一标识。字符长度为1~64，正则表达式为 <code>^.*\$</code> 。获取方法请参考 <a href="#">查询所有微服务信息</a> 。
instance_id	是	String	微服务实例唯一标识。字符长度不超过64位，正则表达式为 <code>^[A-Za-z0-9_-]*\$</code> 。获取方法请参考 <a href="#">根据service_id查询微服务实例</a> 。

表 7-101 Query 参数

参数	是否必选	参数类型	描述
tags	否	String	Tag标签过滤，多个时逗号分隔。正则表达式为 <code>^[a-zA-Z][a-zA-Z0-9_-]{0,63}\$</code> 。

## 请求参数

表 7-102 请求 Header 参数

参数	是否必选	参数类型	描述
X-consumerId	否	String	微服务消费者的微服务唯一标识。
Authorization	否	String	<p>若微服务引擎专享版开启了安全认证，此参数必填。否则，无此参数。</p> <p>开启了安全认证的微服务引擎专享版Token，格式为： Authorization:Bearer {Token}</p> <p>Token获取方法，请参考<a href="#">获取微服务引擎专享版用户Token</a>。</p>

## 响应参数

状态码： 200

表 7-103 响应 Body 参数

参数	参数类型	描述
instance	<b>MicroServiceInstance</b> object	微服务实例信息。

表 7-104 MicroServiceInstance

参数	参数类型	描述
instanceId	String	实例id, 唯一标识。创建实例, instanceId由 service-center产生。
serviceId	String	微服务唯一标识, 创建实例时, 以url里面的为准, 不用这里的serviceId。
version	String	微服务版本号。
hostName	String	主机信息。
endpoints	Array of strings	访问地址信息。
status	String	实例状态: UP、DOWN、STARTING、OUTOFSERVICE。默认值UP。
properties	Object	微服务扩展属性, 可以自定义KEY和相应的 Value。长度最小1字节。
healthCheck	<b>HealthCheck</b> object	健康检查信息。
dataCenterInfo	<b>DataCenterInfo</b> object	数据中心信息。
timestamp	String	实例创建时间戳, 自动生成。
modTimestamp	String	更新时间。

表 7-105 HealthCheck

参数	参数类型	描述
mode	String	心跳模式push/pull。
port	Integer	端口。

参数	参数类型	描述
interval	Integer	心跳间隔（秒），当值小于5秒时，按5秒注册。
times	Integer	最大尝试请求次数。

表 7-106 DataCenterInfo

参数	参数类型	描述
name	String	区域名称
region	String	区域
availableZone	String	可用区

状态码： 400

表 7-107 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

状态码： 500

表 7-108 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

## 请求示例

查询指定service\_id和instance\_id的该实例的详细信息。

```
GET https://{endpoint}/v4/{project_id}/registry/microservices/{service_id}/instances/{instance_id}
```

## 响应示例

状态码： 200

查询成功。

```
{
  "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"
  }
}
```

## 状态码

状态码	描述
200	查询成功。
400	错误的请求。
500	服务内部错误。

## 错误码

请参见[ServiceComb错误码](#)。

## 7.5.5 修改微服务实例扩展信息

### 功能介绍

实例注册后，可以根据instance\_id添加或更新一个微服务实例扩展信息。

### 接口约束

无

### URI

PUT /v4/{project\_id}/registry/microservices/{service\_id}/instances/{instance\_id}/properties

表 7-109 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	<ul style="list-style-type: none"> <li>微服务引擎专业版，请填写租户项目id，长度为1~64，获取方法请参考<a href="#">获取项目ID</a>。也可以填固定值default。</li> <li>微服务引擎专享版，请填写固定值：default。</li> </ul>
service_id	是	String	微服务唯一标识。字符长度为1~64，正则表达式为 <code>^.*\$</code> 。获取方法请参考 <a href="#">查询所有微服务信息</a> 。
instance_id	是	String	微服务实例唯一标识。字符长度不超过64位，正则表达式为 <code>^[A-Za-z0-9_-]*\$</code> 。获取方法请参考 <a href="#">根据service_id查询微服务实例</a> 。

## 请求参数

表 7-110 请求 Header 参数

参数	是否必选	参数类型	描述
Authorization	否	String	<p>若微服务引擎专享版开启了安全认证，此参数必填。否则，无此参数。</p> <p>开启了安全认证的微服务引擎专享版Token，格式为： Authorization:Bearer {Token}</p> <p>Token获取方法，请参考<a href="#">获取微服务引擎专享版用户Token</a>。</p>
X-consumerId	否	String	微服务消费者的微服务唯一标识。

表 7-111 请求 Body 参数

参数	是否必选	参数类型	描述
properties	否	Object	微服务扩展属性，可以自定义KEY和相应的Value。长度最小1字节。

## 响应参数

状态码： 400

表 7-112 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

状态码： 500

表 7-113 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

## 请求示例

更新指定instance\_id的实例的扩展信息为"attr1": "b"。

```
PUT https://{endpoint}/v4/{project_id}/registry/microservices/{service_id}/instances/{instance_id}/properties
{
  "properties": {
    "attr1": "b"
  }
}
```

## 响应示例

无

## 状态码

状态码	描述
200	修改成功。
400	错误的请求。
500	服务内部错误。



## 错误码

请参见[ServiceComb错误码](#)。

## 7.5.6 修改微服务实例状态

### 功能介绍

实例注册后可以根据instance\_id进行更新一个微服务实例状态。

### 接口约束

TESTING状态是2.3.X版本才支持的，老版本不支持。

### URI

PUT /v4/{project\_id}/registry/microservices/{service\_id}/instances/{instance\_id}/status

表 7-114 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	<ul style="list-style-type: none"><li>微服务引擎专业版，请填写租户项目id，长度为1~64，获取方法请参考<a href="#">获取项目ID</a>。也可以填固定值default。</li><li>微服务引擎专享版，请填写固定值：default。</li></ul>
service_id	是	String	微服务唯一标识。字符长度为1~64，正则表达式为 $^{\wedge}.*$$ 。获取方法请参考 <a href="#">查询所有微服务信息</a> 。
instance_id	是	String	微服务实例唯一标识。字符长度不超过64位，正则表达式为 $^{\wedge}[A-Za-z0-9_-]*$$ 。获取方法请参考 <a href="#">根据service_id查询微服务实例</a> 。

表 7-115 Query 参数

参数	是否必选	参数类型	描述
value	是	String	实例状态。UP表示在线、OUTOFSERVICE表示摘机、STARTING表示正在启动、DOWN表示下线、TESTING表示拨测。

## 请求参数

表 7-116 请求 Header 参数

参数	是否必选	参数类型	描述
Authorization	否	String	若微服务引擎专享版开启了安全认证，此参数必填。否则，无此参数。 开启了安全认证的微服务引擎专享版Token，格式为： Authorization:Bearer {Token} Token获取方法，请参考 <a href="#">获取微服务引擎专享版用户Token</a> 。

## 响应参数

状态码： 400

表 7-117 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

状态码： 500

表 7-118 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

## 请求示例

更新指定instance\_id的实例的状态为UP。

```
PUT https://{endpoint}/v4/{project_id}/registry/microservices/{service_id}/instances/{instance_id}/status?value=UP
```

## 响应示例

无

## 状态码

状态码	描述
200	修改成功。
400	错误的请求。
500	服务内部错误。

## 错误码

请参见[ServiceComb错误码](#)。

## 7.5.7 发送心跳信息

### 功能介绍

服务提供端需要向服务中心发送心跳信息，以保证服务中心感知服务实例是否健康。

### 接口约束

无

### URI

PUT /v4/{project\_id}/registry/microservices/{service\_id}/instances/{instance\_id}/heartbeat

表 7-119 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	<ul style="list-style-type: none"><li>微服务引擎专业版，请填租户项目id，长度为1~64，获取方法请参考<a href="#">获取项目ID</a>。也可以填固定值default。</li><li>微服务引擎专享版，请填固定值：default。</li></ul>
service_id	是	String	微服务唯一标识。字符长度为1~64，正则表达式为 $^.*$$ 。获取方法请参考 <a href="#">查询所有微服务信息</a> 。

参数	是否必选	参数类型	描述
instance_id	是	String	微服务实例唯一标识。字符长度不超过64位，正则表达式为 <code>^[A-Za-z0-9_-]*\$</code> 。获取方法请参考 <a href="#">根据service_id查询微服务实例</a> 。

## 请求参数

表 7-120 请求 Header 参数

参数	是否必选	参数类型	描述
Authorization	否	String	若微服务引擎专享版开启了安全认证，此参数必填。否则，无此参数。 开启了安全认证的微服务引擎专享版Token，格式为： Authorization:Bearer {Token} Token获取方法，请参考 <a href="#">获取微服务引擎专享版用户Token</a> 。

## 响应参数

状态码： 400

表 7-121 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

状态码： 500

表 7-122 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

## 请求示例

发送心跳信息，感知instance\_id的实例是否健康。

```
PUT https://{endpoint}/v4/{project_id}/registry/microservices/{service_id}/instances/{instance_id}/heartbeat
```

## 响应示例

无

## 状态码

状态码	描述
200	更新成功。
400	错误的请求。
500	服务内部错误。

## 错误码

请参见[ServiceComb错误码](#)。

## 7.5.8 按条件查询微服务实例

### 功能介绍

实例注册后可以根据微服务字段条件发现该微服务的实例。

### URI

```
GET /v4/{project_id}/registry/instances
```

表 7-123 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	<ul style="list-style-type: none"><li>微服务引擎专业版，请填租户项目id，长度为1~64，获取方法请参考<a href="#">获取项目ID</a>。也可以填固定值default。</li><li>微服务引擎专享版，请填固定值：default。</li></ul>

表 7-124 Query 参数

参数	是否必选	参数类型	描述
appld	是	String	应用app唯一标识。字符长度为1~160。正则表达式为 <code>^[a-zA-Z0-9]\$ ^([a-zA-Z0-9][a-zA-Z0-9_-])[a-zA-Z0-9]\$</code> 。
serviceName	是	String	微服务名称。字符长度为1~128。正则表达式为 <code>^[a-zA-Z0-9]\$ ^([a-zA-Z0-9][a-zA-Z0-9_-])[a-zA-Z0-9]\$</code> 。
tags	否	String	Tag标签过滤，多个时逗号分隔。正则表达式为 <code>^[a-zA-Z][a-zA-Z0-9_-]{0,63}\$</code> 。
env	否	String	实例的环境信息。

## 请求参数

表 7-125 请求 Header 参数

参数	是否必选	参数类型	描述
X-ConsumerId	否	String	微服务消费者的微服务唯一标识。
Authorization	否	String	若微服务引擎专享版开启了安全认证，此参数必填。否则，无此参数。 开启了安全认证的微服务引擎专享版Token，格式为： Authorization:Bearer {Token} Token获取方法，请参考 <a href="#">获取微服务引擎专享版用户Token</a> 。

## 响应参数

状态码： 200

表 7-126 响应 Body 参数

参数	参数类型	描述
instances	Array of <a href="#">MicroServiceInstance</a> objects	实例列表。

表 7-127 MicroServiceInstance

参数	参数类型	描述
instanceId	String	实例id, 唯一标识。创建实例, instanceId 由service-center产生。
serviceId	String	微服务唯一标识, 创建实例时, 以url里面的为准, 不用这里的serviceId。
version	String	微服务版本号。
hostName	String	主机信息。
endpoints	Array of strings	访问地址信息。
status	String	实例状态: UP、DOWN、STARTING、OUTOFSERVICE。默认值UP。
properties	Object	微服务扩展属性, 可以自定义KEY和相应的Value。长度最小1字节。
healthCheck	<b>HealthCheck</b> object	健康检查信息。
dataCenterInfo	<b>DataCenterInfo</b> object	数据中心信息。
timestamp	String	实例创建时间戳, 自动生成。
modTimestamp	String	更新时间。

表 7-128 HealthCheck

参数	参数类型	描述
mode	String	心跳模式push/pull。
port	Integer	端口。
interval	Integer	心跳间隔(秒), 当值小于5秒时, 按5秒注册。
times	Integer	最大尝试请求次数。

表 7-129 DataCenterInfo

参数	参数类型	描述
name	String	区域名称。
region	String	区域。
availableZone	String	可用区。

**状态码： 400**

**表 7-130 响应 Body 参数**

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

**状态码： 500**

**表 7-131 响应 Body 参数**

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

## 请求示例

查询符合应用ID为default，微服务名称为test的微服务的实例。

```
GET https://{endpoint}/v4/{project_id}/registry/instances?appId=default&serviceName=test
```

## 响应示例

**状态码： 200**

查询成功。

```
{
  "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"  
  }  
]  
}
```

## 状态码

状态码	描述
200	查询成功。
400	错误的请求。
500	服务内部错误。

## 错误码

请参见[ServiceComb错误码](#)。

## 7.5.9 批量查询微服务实例

### 功能介绍

实例注册后可以根据微服务字段条件批量发现多个微服务的实例。

### URI

POST /v4/{project\_id}/registry/instances/action

表 7-132 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	<ul style="list-style-type: none"><li>微服务引擎专业版，请填租户项目id，长度为1~64，获取方法请参考<a href="#">获取项目ID</a>。也可以填固定值default。</li><li>微服务引擎专享版，请填固定值：default。</li></ul>

表 7-133 Query 参数

参数	是否必选	参数类型	描述
type	是	String	操作类型，目前仅有“query”，表示查询。

## 请求参数

表 7-134 请求 Header 参数

参数	是否必选	参数类型	描述
X-consumerId	否	String	微服务消费者的微服务唯一标识。
Authorization	否	String	若微服务引擎专享版开启了安全认证，此参数必填。否则，无此参数。 开启了安全认证的微服务引擎专享版Token，格式为： Authorization:Bearer {Token} Token获取方法，请参考 <a href="#">获取微服务引擎专享版用户Token</a> 。

表 7-135 请求 Body 参数

参数	是否必选	参数类型	描述
services	两个参数至少必须选择一个。	Array of <a href="#">FindService</a> objects	查询服务信息结构体。
instances		Array of <a href="#">FindInstance</a> objects	查询实例信息结构体。

表 7-136 FindService

参数	是否必选	参数类型	描述
service	是	<a href="#">Dependency Key</a> object	依赖关系条目。
rev	否	String	客户端缓存的版本号，用于判断本地缓存和服务中心微服务的差异。 <ul style="list-style-type: none"><li>首次传值0。</li><li>若传入的rev值与当前的微服务的rev一致，则返回notModified；若不一致，则返回updated信息。</li><li>下次请求若需要传入rev，则该值为当前返回body体中的updated中的rev的值。</li></ul>

表 7-137 DependencyKey

参数	是否必选	参数类型	描述
environment	是	String	环境信息：development、testing、acceptance、production。
appId	是	String	应用app唯一标识。
serviceName	是	String	微服务名称。

表 7-138 FindInstance

参数	是否必选	参数类型	描述
instance	否	<b>HeartbeatSet Element</b> object	微服务实例请求结构体。
rev	否	String	客户端缓存的版本号， 客户端缓存的版本号，用于判断本地缓存和服务中心微服务实例的差异。 <ul style="list-style-type: none"><li>首次传值0。</li><li>若传入的rev值与当前的微服务实例的rev一致，则返回notModified；若不一致，则返回updated信息。</li><li>下次请求若需要传入rev，则该值为当前返回body体中的updated中的rev的值。</li></ul>

表 7-139 HeartbeatSetElement

参数	是否必选	参数类型	描述
serviceId	否	String	微服务id。
instanceId	否	String	微服务实例id。

## 响应参数

状态码： 200

表 7-140 响应 Body 参数

参数	参数类型	描述
services	<b>BatchFindResult</b> object	批量查询结果结构体。
instances	<b>BatchFindResult</b> object	批量查询结果结构体。

表 7-141 BatchFindResult

参数	参数类型	描述
failed	Array of <b>FindFailedResult</b> objects	查询失败结果结构体。
notModified	Array of integers	与请求数组对应的索引集合。
updated	Array of <b>FindResult</b> objects	查询结果结构体列表。

表 7-142 FindFailedResult

参数	参数类型	描述
indexes	Array of integers	与请求数组对应的索引集合。
error	<b>Error</b> object	异常错误结构体。

表 7-143 Error

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

表 7-144 FindResult

参数	参数类型	描述
index	Integer	与请求数组对应的索引。
rev	String	服务端返回集合版本，如跟客户端缓存版本号一致，则instances为空。

参数	参数类型	描述
instances	Array of <a href="#">MicroServiceInstance</a> objects	实例列表。

表 7-145 MicroServiceInstance

参数	参数类型	描述
instanceId	String	实例id, 唯一标识。创建实例, instanceId由 service-center产生。
serviceId	String	微服务唯一标识, 创建实例时, 以url里面的为准, 不用这里的serviceId。
version	String	微服务版本号。
hostName	String	主机信息。
endpoints	Array of strings	访问地址信息。
status	String	实例状态: UP、DOWN、STARTING、OUTOFSERVICE。默认值UP。
properties	Object	微服务扩展属性, 可以自定义KEY和相应的 Value。长度最小为1字节。
healthCheck	<a href="#">HealthCheck</a> object	健康检查信息。
dataCenterInfo	<a href="#">DataCenterInfo</a> object	数据中心信息。
timestamp	String	实例创建时间戳, 自动生成。
modTimestamp	String	更新时间。

表 7-146 HealthCheck

参数	参数类型	描述
mode	String	心跳模式push/pull。
port	Integer	端口。
interval	Integer	心跳间隔(秒), 当值小于5秒时, 按5秒注册。
times	Integer	最大尝试请求次数。

表 7-147 DataCenterInfo

参数	参数类型	描述
name	String	区域名称。
region	String	区域。
availableZone	String	可用区。

状态码： 400

表 7-148 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

状态码： 500

表 7-149 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

## 请求示例

查询应用ID为default，微服务名称为test的微服务的所有实例。

```
POST https://{endpoint}/v4/{project_id}/registry/instances/action?type=query
{
  "services": [
    {
      "service": {
        "environment": "",
        "appld": "default",
        "serviceName": "test"
      },
      "rev": "0"
    }
  ]
}
```

## 响应示例

**状态码： 200**

查询成功。

```
{
  "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"
          }
        ]
      }
    ]
  }
}
```

## 状态码

状态码	描述
200	查询成功。
400	错误的请求。
500	服务内部错误。

## 错误码

请参见[ServiceComb错误码](#)。

## 7.6 依赖关系

## 7.6.1 查询服务 providers

### 功能介绍

根据consumer\_id获取该服务的所有providers，比如A->B、A->C。那么根据A的服务id，可以查询到B、C的微服务信息。

### URI

GET /v4/{project\_id}/registry/microservices/{consumer\_id}/providers

表 7-150 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	<ul style="list-style-type: none"><li>微服务引擎专业版，请填写租户项目id，长度为1~64，获取方法请参考<a href="#">获取项目ID</a>。也可以填固定值default。</li><li>微服务引擎专享版，请填写固定值：default。</li></ul>
consumer_id	是	String	消费者的服务id。

表 7-151 Query 参数

参数	是否必选	参数类型	描述
noSelf	否	Integer	是否取消返回自依赖的关系，0表示不返回自依赖关系，1表示返回自依赖关系。
sameDomain	否	Integer	是否取消返回共享服务的关系，0表示不返回共享服务，1表示返回。



## 请求参数

表 7-152 请求 Header 参数

参数	是否必选	参数类型	描述
Authorization	否	String	<p>若微服务引擎专享版开启了安全认证，此参数必填。否则，无此参数。</p> <p>开启了安全认证的微服务引擎专享版Token，格式为： Authorization:Bearer {Token}</p> <p>Token获取方法，请参考<a href="#">获取微服务引擎专享版用户Token</a>。</p>

## 响应参数

状态码： 200

表 7-153 响应 Body 参数

参数	参数类型	描述
providers	MicroService object	微服务信息。

表 7-154 MicroService

参数	参数类型	描述
serviceld	String	微服务唯一标识。字符长度为1~64。正则表达式为 $^.*\$$ 。
environment	String	用于区分微服务环境，取值为：development、testing、acceptance、production。当配置为development、testing或acceptance时，可以通过批量上传schemas接口新增或者修改已存在的Schema；当配置为production时，则不可以新增或者修改Schema。默认值development。
appId	String	应用App唯一标识。字符长度为1~160。正则表达式为 $^[a-zA-Z0-9]^\{a-zA-Z0-9\}[a-zA-Z0-9_\.][a-zA-Z0-9]^\{a-zA-Z0-9\}$ 。
serviceName	String	微服务名称，同一个App要保证唯一。字符长度为1~128。正则表达式为 $^[a-zA-Z0-9]^\{a-zA-Z0-9\}[a-zA-Z0-9_\.][a-zA-Z0-9]^\{a-zA-Z0-9\}$ 。
version	String	微服务版本号。字符长度为1~64。正则表达式为 $^[0-9]^\{0-9\}+(\.[0-9]^\{0-9\})^\{0-9\}$ 。

参数	参数类型	描述
description	String	微服务描述信息。字符长度不超过256。
level	String	微服务层级：FRONT、MIDDLE、BACK。
registerBy	String	微服务注册方式：SDK、PLATFORM、SIDECAR、UNKNOWN。
schemas	Array of strings	微服务访问的契约内容。支持数字、字母，支持使用括号内字符（_-.）做连接符，长度1-160字节，数组长度最大100个契约。
status	String	微服务状态：UP表示上线，DOWN表示下线。默认值UP。
timestamp	String	微服务注册时间。
modTimestamp	String	最后修改UTC时间。
framework	Framework object	开发框架信息。
paths	Array of ServicePath objects	服务路由信息。
properties	Object	扩展属性。可以自定义KEY和相应的Value。

表 7-155 Framework

参数	参数类型	描述
name	String	微服务开发框架，默认值为UNKNOWN。
version	String	微服务开发框架版本号。

表 7-156 ServicePath

参数	参数类型	描述
Path	String	路由地址。
Property	Object	微服务扩展属性，可以自定义KEY和相应的Value。长度最小1字节。

状态码： 400

表 7-157 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

状态码： 500

表 7-158 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

## 请求示例

查询指定服务ID为consumer\_id的消费者服务的providers信息。

```
GET https://{endpoint}/v4/{project_id}/registry/microservices/{consumer_id}/providers
```

## 响应示例

状态码： 200

查询成功，则返回serviceld或者schemald。

```
{
  "providers": [
    {
      "serviceld": "8aed80ea052ac04a64dfc79c24f2170224d074f5",
      "appld": "default",
      "serviceName": "test",
      "version": "1.0.0",
      "description": "this is a test",
      "level": "BACK",
      "status": "UP",
      "properties": {
        "test": "test"
      },
      "timestamp": "1650543950",
      "modTimestamp": "1650544411"
    }
  ]
}
```

## 状态码

状态码	描述
200	查询成功。
400	错误的请求。
500	服务内部错误。

## 错误码

请参见[ServiceComb错误码](#)。

## 7.7 配置管理

### 7.7.1 创建配置

#### 功能介绍

创建一条配置。

#### URI

POST /v1/{project\_id}/kie/kv

表 7-159 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	请填固定值：default。

#### 请求参数

表 7-160 请求 Header 参数

参数	是否必选	参数类型	描述
Authorization	否	String	若微服务引擎专享版开启了安全认证，此参数必填。否则，无此参数。 开启了安全认证的微服务引擎专享版Token，格式为： Authorization:Bearer {Token} Token获取方法，请参考 <a href="#">获取微服务引擎专享版用户Token</a> 。

表 7-161 请求 Body 参数

参数	是否必选	参数类型	描述
key	是	String	配置的key。
value	否	String	配置的value。
labels	否	Map<String,String>	配置的labels。
status	否	String	配置的状态。
value_type	否	String	配置value的类型。支持的类型有text、yaml、json、properties、ini、xml。

## 响应参数

状态码： 200

表 7-162 响应 Body 参数

参数	参数类型	描述
kie	<b>SingleKieInfo</b>	配置信息。

表 7-163 SingleKieInfo

参数	参数类型	描述
id	String	配置id。
key	String	配置的key。
value	String	配置的value。
value_type	String	配置value的类型。
status	String	配置的状态。
create_time	String	配置的创建时间。
update_time	String	配置的更新时间。
labels	Map<String,String>	配置的标签信息。
create_version	Integer	配置的创建版本号。
update_version	Integer	配置的修改版本号。

**状态码： 400**

**表 7-164** 响应 Body 参数

参数	参数类型	描述
error_code	String	错误代码。
error_message	String	错误信息。

**状态码： 500**

**表 7-165** 响应 Body 参数

参数	参数类型	描述
error_code	String	错误代码。
error_message	String	错误信息。

## 请求示例

创建一条配置，其标签信息为："key1": "value1", "key2": "value2"。

```
POST https://{endpoint}/v1/{project_id}/kie/kv
{
  "key": "String",
  "value": "String",
  "labels": {
    "key1": "value1",
    "key2": "value2",
  },
  "status": "String",
  "value_type": "String"
}
```

## 响应示例

**状态码： 200**

创建成功，返回配置信息。

```
{
  "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,  
}
```

## 状态码

状态码	描述
200	创建成功。
400	错误的请求。
409	创建的配置已存在。
500	服务内部错误。

## 错误码

请参见[ServiceComb错误码](#)。

## 7.7.2 修改配置

### 功能介绍

修改指定一条配置。

### URI

PUT /v1/{project\_id}/kie/kv/{kv\_id}

表 7-166 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	请填写固定值：default。
kv_id	是	String	修改的指定配置的id，获取方法请参考 <a href="#">查询配置列表</a> 。

## 请求参数

表 7-167 请求 Header 参数

参数	是否必选	参数类型	描述
Authorization	否	String	若微服务引擎专享版开启了安全认证，此参数必填。否则，无此参数。 开启了安全认证的微服务引擎专享版Token，格式为： Authorization:Bearer {Token} Token获取方法，请参考 <a href="#">获取微服务引擎专享版用户Token</a> 。

表 7-168 请求 Body 参数

参数	是否必选	参数类型	描述
value	是	String	配置的value。
status	否	String	配置的状态。

## 响应参数

状态码： 200

表 7-169 响应 Body 参数

参数	参数类型	描述
kie	<a href="#">SingleKieInfo</a>	一条配置信息。

状态码： 400

表 7-170 响应 Body 参数

参数	参数类型	描述
error_code	String	错误代码。
error_message	String	错误信息。

状态码： 500



表 7-171 响应 Body 参数

参数	参数类型	描述
error_code	String	错误代码。
error_message	String	错误信息。

## 请求示例

修改配置ID为8a9e6a5d-8d65-48fb-a40c-5150c8479da8的配置值为this is a test，设置其配置状态为enabled。

```
PUT https://{endpoint}/v1/{project_id}/kie/kv/8a9e6a5d-8d65-48fb-a40c-5150c8479da8
{
  "value": "this is a test",
  "status": "enabled"
}
```

## 响应示例

**状态码： 200**

修改成功，返回配置信息。

```
{
  "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,
}
```

## 状态码

状态码	描述
200	修改成功。
400	错误的请求。
500	服务内部错误。

## 错误码

请参见[ServiceComb错误码](#)。

## 7.7.3 查询指定配置

### 功能介绍

从配置管理中心查询配置信息。

### URI

GET /v1/{project\_id}/kie/kv/{kv\_id}

表 7-172 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	请填写固定值：default。
kv_id	是	String	查询的配置id，获取方法请参考 <a href="#">查询配置列表</a> 。

### 请求参数

表 7-173 请求 Header 参数

参数	是否必选	参数类型	描述
Authorization	否	String	若微服务引擎专享版开启了安全认证，此参数必填。否则，无此参数。 开启了安全认证的微服务引擎专享版Token，格式为： Authorization:Bearer {Token} Token获取方法，请参考 <a href="#">获取微服务引擎专享版用户Token</a> 。

### 响应参数

状态码： 200

表 7-174 响应 Body 参数

参数	参数类型	描述
id	String	配置id。
key	String	配置的key。
value	String	配置的值。
value_type	String	配置value的类型。

参数	参数类型	描述
status	String	配置的状态。
create_time	String	配置的创建时间。
update_time	String	配置的更新时间。
labels	Map<String,String>	配置的标签信息。
create_version	Integer	配置的创建版本号。
update_version	Integer	配置的修改版本号。

状态码： 400

表 7-175 响应 Body 参数

参数	参数类型	描述
error_code	String	错误代码。
error_message	String	错误信息。

状态码： 404

表 7-176 响应 Body 参数

参数	参数类型	描述
error_code	String	错误代码。
error_message	String	错误信息。

状态码： 500

表 7-177 响应 Body 参数

参数	参数类型	描述
error_code	String	错误代码。
error_message	String	错误信息。

## 请求示例

查询配置ID为8a9e6a5d-8d65-48fb-a40c-5150c847的配置信息。

```
GET https://{endpoint}/v1/{project_id}/kie/kv/8a9e6a5d-8d65-48fb-a40c-5150c8479da8
```

## 响应示例

**状态码： 200**

查询成功

```
{
  "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
}
```

## 状态码

状态码	描述
200	查询成功。
400	错误的请求。
404	所请求的资源不存在。
500	服务内部错误。

## 错误码

请参见[ServiceComb错误码](#)。

## 7.7.4 查询配置列表

### 功能介绍

从配置管理中心查询配置列表信息。

### URI

```
GET /v1/{project_id}/kie/kv
```

表 7-178 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	请填写固定值：default。

## 请求参数

表 7-179 请求 Header 参数

参数	是否必选	参数类型	描述
Authorization	否	String	若微服务引擎专享版开启了安全认证，此参数必填。否则，无此参数。 开启了安全认证的微服务引擎专享版Token，格式为： Authorization:Bearer {Token} Token获取方法，请参考 <a href="#">获取微服务引擎专享版用户Token</a> 。

表 7-180 Query 参数

参数	是否必选	参数类型	描述
label	否	String	按label过滤项导出，格式为： {标签key}:{标签value}。
match	否	String	对label过滤项的匹配选项，如果值为exact，表示严格匹配，包括label个数和内容相等；不填表示包含匹配。
revision	否	Integer	配置版本号。 用于判断本地缓存配置和配置管理中心的配置差异。 <ul style="list-style-type: none"><li>首次传值0。</li><li>若传入的revision值与当前配置的revision一致，则返回304状态码，无body体返回；若不一致，则返回当前配置和revision信息。</li><li>下次请求若需要传入revision，则该值为当前返回body体中的update_revision。</li></ul>

## 响应参数

状态码： 200

表 7-181 响应 Body 参数

参数	参数类型	描述
total	Integer	查询结果个数。
data	List<SingleKielInfo>	查询的kie列表。

表 7-182 SingleKielInfo

参数	参数类型	描述
id	String	配置id。
key	String	配置的key。
value	String	配置的value。
value_type	String	配置value的类型。
status	String	配置的状态。
create_time	String	配置的创建时间。
update_time	String	配置的更新时间。
labels	Map<String,String>	配置的标签信息。
create_version	Integer	配置的创建版本号。
update_version	Integer	配置的修改版本号。

状态码： 304

无

状态码： 400

表 7-183 响应 Body 参数

参数	参数类型	描述
error_code	String	错误代码。

参数	参数类型	描述
error_message	String	错误信息。

状态码： 500

表 7-184 响应 Body 参数

参数	参数类型	描述
error_code	String	错误代码。
error_message	String	错误信息。

## 请求示例

查询配置列表。

GET https://{endpoint}/v1/{project\_id}/kie/kv

## 响应示例

状态码： 200

查询成功。

```
{
  "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
  }]
}
```

## 状态码

状态码	描述
200	查询成功。
304	查询的配置未改变。
400	错误的请求。

状态码	描述
500	服务内部错误。

## 错误码

请参见[ServiceComb错误码](#)。

## 7.7.5 删除指定配置

### 功能介绍

删除一条配置。

### URI

DELETE /v1/{project\_id}/kie/kv/{kv\_id}

表 7-185 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	请填固定值：default。
kv_id	是	String	删除的配置id，获取方法请参考 <a href="#">查询配置列表</a> 。

### 请求参数

表 7-186 请求 Header 参数

参数	是否必选	参数类型	描述
Authorization	否	String	若微服务引擎专享版开启了安全认证，此参数必填。否则，无此参数。 开启了安全认证的微服务引擎专享版Token，格式为： Authorization:Bearer {Token} Token获取方法，请参考 <a href="#">获取微服务引擎专享版用户Token</a> 。

### 响应参数

状态码： 200

删除成功。



**状态码： 404**

**表 7-187 响应 Body 参数**

参数	参数类型	描述
error_code	String	错误代码。
error_message	String	错误信息。

**状态码： 400**

**表 7-188 响应 Body 参数**

参数	参数类型	描述
error_code	String	错误代码。
error_message	String	错误信息。

**状态码： 500**

**表 7-189 响应 Body 参数**

参数	参数类型	描述
error_code	String	错误代码。
error_message	String	错误信息。

## 请求示例

删除配置ID为8a9e6a5d-8d65-48fb-a40c-5150c8479da8的配置。

```
DELETE https://{endpoint}/v1/{project_id}/kie/kv/8a9e6a5d-8d65-48fb-a40c-5150c8479da8
```

## 响应示例

**状态码： 204**

删除成功。

## 状态码

状态码	描述
200	删除成功。
400	错误的请求。
404	未找到指定资源。
500	服务内部错误。

## 错误码

请参见[ServiceComb错误码](#)。

### 7.7.6 删除所有配置

#### 功能介绍

删除所有配置。

#### URI

DELETE /v1/{project\_id}/kie/kv

表 7-190 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	请填固定值：default。

#### 请求参数

表 7-191 请求 Header 参数

参数	是否必选	参数类型	描述
Authorization	否	String	若微服务引擎专享版开启了安全认证，此参数必填。否则，无此参数。 开启了安全认证的微服务引擎专享版Token，格式为： Authorization:Bearer {Token} Token获取方法，请参考 <a href="#">获取微服务引擎专享版用户Token</a> 。

表 7-192 请求 Body 参数

参数	是否必选	参数类型	描述
ids	是	List<String>	待删除配置的id列表，获取方法请参考 <a href="#">查询配置列表</a> 。

## 响应参数

状态码： 204

删除成功。

状态码： 404

表 7-193 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

状态码： 500

表 7-194 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

## 请求示例

删除所有配置。

```
DELETE https://{endpoint}/v1/{project_id}/kie/kv
{
  "ids":[
    "id1",
    "id2"
  ]
}
```

## 响应示例

**状态码： 204**

删除成功。

## 状态码

状态码	描述
204	删除成功。
400	错误的请求。
500	服务内部错误。

## 错误码

请参见[ServiceComb错误码](#)。

# 8 应用示例

## 8.1 创建并管理环境

### 操作场景

环境是用于组件部署和运行的计算（如云容器引擎 CCE、弹性云服务器 ECS等）、网络（如弹性负载均衡 ELB、弹性IP EIP等）和中间件（如分布式缓存 DCS、云数据库 RDS等）等基础资源的集合。ServiceStage把多种基础资源组合为一个环境，如：开发环境、测试环境、预生产环境和生产环境。按环境维度来管理资源、部署组件，可以减少具体基础资源运维管理的复杂性。

下面介绍如何调用[创建环境](#)API创建环境，再调用[根据环境ID修改环境资源](#)API为环境添加其他需要的资源，最后调用[根据环境ID获取环境详细信息](#)API根据环境ID查询环境详细信息。

API的调用方法请参见[如何调用API](#)。

### 前提条件

- 已调用[获取用户Token](#)获取IAM用户的Token，并将token值设置为环境变量：  
export Token={token}。
- 已获取ServiceStage的终端节点，详细信息请参见[地区和终端节点](#)，并用于替换以下操作步骤示例中的{servicestage\_endpoint}。

### 操作步骤

**步骤1** 调用[创建环境](#)API创建环境，请求示例如下：

```
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"  
  }  
}
```

返回结果:

```
{  
  "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": []  
}
```

记录返回结果中加粗显示的环境ID，用于替换后续示例中的{environment\_id}。

**步骤2** 调用[根据环境ID修改环境资源](#)API为环境添加资源。

根据您的已有的资源和需求，可以将和环境相同VPC下的计算、网络、中间件等多种类型的资源添加到环境中。请求示例如下：

```
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"  
    }  
  ]  
}'
```

返回结果:

```
{  
  "resources": [  
    {  
      "id": "d2917177-3436-4e53-a5bc-706c3529e0e7",  
      "type": "elb",  
      "name": "elb-pu6a"  
    }  
  ]  
}
```

**步骤3** 调用[根据环境ID获取环境详细信息](#)API根据环境ID查询环境详细信息，请求示例如下：

```
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}"
```

返回结果:

```
{  
  "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": []  
}
```

```
"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"
  }
]
}
```

----结束

## 8.2 创建并管理应用

### 操作场景

应用是一个功能相对完备的业务系统，由一个或多个特性相关的组件组成。

下面介绍如何调用[创建应用](#)API创建应用，再调用[根据应用ID修改应用配置](#)API为应用添加应用环境变量，最后调用[根据应用ID获取应用详细信息](#)API查询应用详细信息。

API的调用方法请参见[如何调用API](#)。

### 前提条件

- 已调用[获取用户Token](#)获取IAM用户的Token，并将token值设置为环境变量：  
export Token={token}。
- 已获取ServiceStage的终端节点，详细信息请参见[地区和终端节点](#)，并用于替换以下操作步骤示例中的{servicestage\_endpoint}。

### 操作步骤

**步骤1** 调用[创建应用](#)API创建应用，请求示例如下：

```
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"
      }
    ]
  }
}'
```

返回结果：

```
{
  "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"
  }
]
```

记录返回结果中加粗显示的应用ID，用于替换后续示例中的{application\_id}。

**步骤2** 调用**根据应用ID修改应用配置**API为应用添加应用环境变量，请求示例如下：

```
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"
      }
    ]
  }
}'
```

返回结果：

```
{
  "application_id": "2c458700-c8e0-4651-a73b-5219b70a6572",
  "environment_id": "21a43958-f64f-4077-8ae3-3a03bf0111c7",
  "configuration": {
    "env": [
      {
        "name": "testName",
        "value": "testValue"
      }
    ]
  }
}
```

**步骤3** 调用**根据应用ID获取应用详细信息**API根据应用ID查询应用详细信息，请求示例如下：

```
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}"
```

返回结果：

```
{
  "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  
}
```

----结束

## 8.3 创建并管理组件

### 操作场景

组件是组成应用的某个业务特性实现，以代码或者软件包为载体，可独立部署在环境下运行。

下面介绍如何调用[应用中创建组件](#)API创建组件后，先调用[获取部署任务详细信息](#)API查询创建组件任务的状态，再调用[根据组件ID修改组件信息](#)API升级组件，然后调用[根据组件ID下发组件任务](#)API回滚组件版本，最后再次调用[根据组件ID下发组件任务](#)API停止组件实例运行。

API的调用方法请参见[如何调用API](#)。

### 前提条件

- 已调用[获取用户Token](#)获取IAM用户的Token，并将token值设置为环境变量：  
export Token={token}。
- 已获取ServiceStage的终端节点，详细信息请参见[地区和终端节点](#)，并用于替换以下操作步骤示例中的{servicestage\_endpoint}。

### 操作步骤

**步骤1** 调用[应用中创建组件](#)API在某个应用下创建组件，请求示例如下：

```
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.cn-north-4.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": {}
}'
```

返回结果:

```
{
  "job_id": "JOB870e9ffe-227e-43a9-9c84-20a7ad02e77c",
  "component_id": "f3f6cb89-831d-4579-8bbc-664487889e38"
}
```

记录返回结果中加粗显示的job\_id、component\_id的值，用于替换后续示例中的{job\_id}、{component\_id}。

**步骤2** 调用**获取部署任务详细信息**API查询创建组件任务是否成功，请求示例如下：

```
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}"
```

返回结果:

```
{
  "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-wpqwb:30114"
}
],
"task_count": 5
}
```

### 步骤3 调用[根据组件ID修改组件信息](#)API升级组件，请求示例如下：

```
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.cn-north-4.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"
}'
```

返回结果:

```
{
  "job_id": "JOB2209202b-617c-4591-b0c6-ef2521a943a2"
}
```

记录返回结果中加粗显示的job\_id的值，用于替换后续示例中的{job\_id}。

**步骤4** 调用**步骤2**的接口，用**步骤3**返回的job\_id查询任务状态，等待任务完成。

**步骤5** 调用**根据组件ID下发组件任务**API回滚组件，请求示例如下：

```
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"
}'
```

返回结果:

```
{
  "job_id": "JOB66eb56db-062e-4462-ba95-8d3c63d5d462"
}
```

记录返回结果中加粗显示的job\_id的值，用于替换后续示例中的{job\_id}。

**步骤6** 调用**步骤2**的接口，用**步骤5**返回的job\_id查询任务状态，等待任务完成。

**步骤7** 调用**根据组件ID下发组件任务**API停止组件实例，请求示例如下：

```
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"
}'
```

返回结果：

```
{
  "job_id": "JOB2790bf9f-7d6e-473d-827f-1bb94e0cf10d"
}
```

记录返回结果中加粗显示的job\_id的值，用于替换后续示例中的{job\_id}。

**步骤8** 调用**步骤2**的接口，用**步骤7**返回的job\_id查询任务状态，等待任务完成。

----结束

## 8.4 微服务注册发现

### 操作场景

微服务是业务概念，某个进程提供某种服务，那它就是个微服务。每一个服务都具有自主运行的业务功能，对外开放不受语言限制的API（最常用的是HTTP）。多个微服务组成应用程序。

下面介绍如何调用**创建微服务静态信息**API注册微服务my-provider后，先调用**注册微服务实例**API注册微服务my-provider的实例，再次调用**创建微服务静态信息**API注册微服务my-consumer，然后调用**按条件查询微服务实例**API，my-consumer带着自身的serviceld去服务中心查询my-provider的实例信息。

#### 📖 说明

以上微服务名称均为示例。

API的调用方法请参见[如何调用API](#)。

### 前提条件

- 获取IAM的Token，调用**获取用户Token**，并将token值设置为环境变量：export Token={token}。
- 获取微服务引擎的终端节点，请参考[地区和终端节点](#)，以下示例中以{cse\_endpoint}代替。

### 操作步骤

**步骤1** 调用**创建微服务静态信息**API注册微服务my-provider，请求示例如下：

```
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",
  "appld": "default",
  "version": "1.0.0",
  "description": "test",
  "level": "MIDDLE",
  "status": "UP"
}
```

返回结果:

```
{"serviceld": "918282e8562dc5fdc9a8dcd4d1baabb492190aa4"}
```

记录加粗显示的serviceld值，用于替换后续示例中的{providerServiceld}。

**步骤2** 调用[注册微服务实例](#)API注册微服务my-provider的实例。

实例有效期1小时，到期自动下线。假设provider实例监听的地址为127.0.0.1:8080，请求示例如下：

```
curl -k -H "x-domain-name:default" -H "x-auth-token:$Token" -XPOST "https://{cse_endpoint}/v4/default/registry/microservices/{providerServiceld}/instances" -d '{
  "instance": {
    "hostName": "test",
    "endpoints": [
      "rest:127.0.0.1:8080"
    ],
    "status": "UP",
    "healthCheck": {
      "mode": "push",
      "interval": 900,
      "times": 3
    }
  }
}
```

返回结果:

```
{"instanceld": "2be605a095ed11eabcbe0255ac100fa3"}
```

**步骤3** 调用[创建微服务静态信息](#)API注册微服务my-consumer，请求示例如下：

```
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"
  }
}
```

返回结果:

```
{"serviceld": "9db248934c31fc754d6e922b48ede4a5c004d3c1"}
```

记录加粗显示的serviceld值，用于替换后续示例中的{consumerServiceld}代替。

**步骤4** 调用[按条件查询微服务实例](#)接口，my-consumer带着自身的serviceld去服务中心查询my-provider的实例信息。请求示例如下：

```
curl -k -H "x-domain-name:default" -H "x-auth-token:$Token" -H "X-ConsumerId:{consumerServiceld}" -XGET "https://{cse_endpoint}/v4/default/registry/instances?appld=default&serviceName=my-provider&version=0.0.0%2B"
```

返回结果:

```
{
  "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"
    }
  ]
}
```

在实际业务中，my-consumer可从实例查询结果中的"endpoint"字段获取my-provider实例的地址，发起业务调用。

您还可以进入ServiceStage控制台，单击微服务引擎的“查看控制台”，在“服务目录”查看服务注册信息。

----结束



# 9 数据结构

## 9.1 参数说明

### 9.1.1 HealthCheck

表 9-1 参数说明

参数	是否必选	参数类型	描述
mode	是	String	check模式, push/pull。
port	否	Integer	端口, 字符长度不超过65536。正则表达式为 $^{[0-9]^*}$ 。
interval	是	Integer	心跳间隔(秒), 当值小于5秒时, 按5秒注册。字符长度不超过4294967296。正则表达式为 $^{[0-9]^+}$ 。
times	是	Integer	最大尝试请求次数, 字符长度不超过4294967296。正则表达式为 $^{[0-9]^+}$ 。

### 9.1.2 MicroServiceInstance

表 9-2 参数说明

参数	是否必选	参数类型	描述
instanceId	否	String	实例id, 唯一标识。创建实例, instanceId由service-center产生。
serviceId	否	String	微服务唯一标识, 创建实例时, 以url里面的为准。
version	否	String	微服务版本号。

参数	是否必选	参数类型	描述
hostName	是	String	计算机名称，可以为本机的域名，或IP地址等。
endpoints	否	Array of strings	例：rest:127.0.0.1:8080。
status	是	String	实例状态。包括： <ul style="list-style-type: none"> <li>• UP</li> <li>• DOWN</li> <li>• STARTING</li> <li>• OUTFSERVICE</li> </ul>
properties	否	Map<String, String>	扩展属性，可以自定义KEY和相应的Value。
healthCheck	否	<b>HealthCheck</b>	服务健康模式检查。定义微服务的健康检查模式，可选的模式为主动推送（push）和被动拉取（pull）。interval定义拉取时长间隔（秒）、times定义最大尝试请求次数。在被动拉取模式下需定义port，由注册中心主动向该port进行链接。默认为push模式，interval默认为30，times默认为3。
timestamp	否	String	实例创建时间戳，自动生成。
modTimestamp	否	String	更新时间。
dataCenterInfo	否	<b>DataCenterInfo</b>	时区。

### 9.1.3 HeartbeatSetElement

表 9-3 参数说明

参数	是否必选	参数类型	描述
serviceId	是	String	微服务id，字符长度为1~64位。正则表达式为 $^.*\$$ 。
instanceId	是	String	微服务实例id，字符长度为1~64位。正则表达式为 $^[A-Za-z0-9_-.]*\$$ 。

## 9.1.4 InstanceHbRst

表 9-4 参数说明

参数	是否必选	参数类型	描述
serviceld	否	String	微服务id。
instanceld	否	String	微服务实例id。
errMessage	否	String	错误信息，成功为空，不成功，则为具体的错误信息。

## 9.1.5 DelServicesResponse

表 9-5 参数说明

参数	是否必选	参数类型	描述
serviceld	否	String	微服务id。

## 9.1.6 MetricData

表 9-6 参数说明

参数	是否必选	参数类型	描述
name	是	String	服务名称。
appld	是	String	应用Id。
serviceld	否	String	服务Id。
version	是	String	服务版本号。
instanceld	否	String	实例Id。
instance	是	String	实例名称。
thread	否	String	线程数。
cpu	否	String	CPU使用率。
memory	否	String	内存。
interfaces	否	<a href="#">表9-7</a>	实例包含的详细接口数据。

### 说明

serviceld和instanceld两个参数必须同时存在或者同时不存在。

表 9-7 items 参数说明

参数	是否必选	参数类型	描述
items	否	FunctionMetricInfo	接口级别聚合指标数据。

## 9.1.7 FunctionMetricInfo

表 9-8 参数说明

参数	是否必选	参数类型	描述
<b>AggregateMetricInfo</b>	否	Object	聚合指标数据。
l995	否	Integer	99.5%的请求低于此值。
l99	否	Integer	99%的请求低于此值。
l90	否	Integer	90%的请求低于此值。
l75	否	Integer	75%的请求低于此值。
l50	否	Integer	50%的请求低于此值。
l25	否	Integer	25%的请求低于此值。
l5	否	Integer	5%的请求低于此值。

## 9.1.8 InstanceMetricInfo

表 9-9 参数说明

参数	是否必选	参数类型	描述
<b>AggregateMetricInfo</b>	否	Object	聚合指标数据。
thread	否	Integer	线程数。
cpu	否	number	CPU使用率。
memory	否	Object	内存信息。

## 9.2 公共请求参数

## 9.2.1 MicroService

表 9-10 参数说明

参数	是否必选	参数类型	描述
serviceId	否	String	微服务唯一标识, 字符长度为1~64。正则表达式为 <code>^.*\$</code> 。
environment	否	String	微服务的环境信息, 取值可以为空、development、testing、acceptance或production。
appId	否	String	应用App唯一标识, 字符长度为1~160。正则表达式为 <code>^[a-zA-Z0-9]*\$ ^[a-zA-Z0-9][a-zA-Z0-9_\.]*[a-zA-Z0-9]\$</code> 。
serviceName	是	String	微服务名称, 同一个App要保证唯一。字符长度为1~128。正则表达式为 <code>^[a-zA-Z0-9]*\$ ^[a-zA-Z0-9][a-zA-Z0-9_\.]*[a-zA-Z0-9]\$</code> 。
version	否	String	微服务版本号, 字符长度为1~64。正则表达式为 <code>^[0-9]*\$ ^[0-9]+(\.[0-9]+)*\$</code> 。
description	否	String	微服务描述信息, 字符长度不超过256。
level	否	String	微服务层级。包括: <ul style="list-style-type: none"><li>• FRONT</li><li>• MIDDLE</li><li>• BACK</li></ul>
timestamp	否	String	微服务注册时间。
schemas	否	Array of String	微服务访问契约内容的外键ID, 数组长度最大100个契约。
status	否	String	微服务状态, UP表示上线, DOWN表示下线。默认值UP。
paths	否	ServicePath	服务路由信息。
framework	否	Framework	微服务开发框架。
registerBy	否	String	微服务注册方式。包括: <ul style="list-style-type: none"><li>• SDK</li><li>• PLATFORM</li><li>• SIDECAR</li><li>• UNKNOWN</li></ul>

参数	是否必选	参数类型	描述
modTimestamp	否	String	最后修改UTC时间。
properties	否	Map<String, String>	微服务扩展信息，可以自定义Key和相应的Value。
alias	否	String	微服务别名，字符长度为1-128。正则表达式为^[a-zA-Z0-9_\-.]*\$。

表 9-11 ServicePath 参数说明

参数	是否必选	参数类型	描述
Path	否	String	路由地址。支持数字、字母和括号内字符(.,?'\ / +&%\$#=#~_-@{}), 长度1-160字节。
Property	否	<b>Properties</b>	扩展属性。

表 9-12 Framework 参数说明

参数	是否必选	参数类型	描述
name	否	String	微服务开发框架，默认值为UNKNOWN。
version	否	String	微服务开发框架版本号，支持任意字符，长度1-64字节。

## 9.2.2 Properties

表 9-13 参数说明

参数	是否必选	参数类型	描述
< * >	否	String	微服务扩展属性，可以自定义KEY和相应的Value。长度最小1字节。

## 9.2.3 DependencyMicroService

表 9-14 参数说明

参数	是否必选	参数类型	描述
appId	是	String	应用app唯一标识，字符长度为1~160。 正则表达式为 <code>^[a-zA-Z0-9]*\$ ^([a-zA-Z0-9][a-zA-Z0-9_\-\.]*[a-zA-Z0-9])\$</code> 。
serviceName	是	String	微服务名称。作为provider支持为*，表示依赖同一租户下的所有服务。当服务名称为*时，appId和version可以省略。consumer不支持*。字符长度为1~128。 正则表达式为 <code>^[a-zA-Z0-9]*\$ ^([a-zA-Z0-9][a-zA-Z0-9_\-\.]*[a-zA-Z0-9])\$</code> 。
version	是	String	版本规则，若为微服务引擎专业版，此参数必填；否则，无此参数。 <ul style="list-style-type: none"><li>表示精确版本匹配，如：0.0.1</li><li>表示后续版本匹配，如：0.0.1+</li><li>表示最新版本，如：latest</li><li>表示版本范围，如：0.1.0-0.2.0</li></ul>
environment	是	String	微服务的环境信息。包括： <ul style="list-style-type: none"><li>空</li><li>development</li><li>testing</li><li>acceptance</li><li>production</li></ul>

## 9.2.4 Rule

表 9-15 参数说明

参数	是否必选	参数类型	描述
ruleId	否	String	黑白名单ID。
ruleType	是	String	类型。 <ul style="list-style-type: none"><li>WHITE表示白名单。</li><li>BLACK表示黑名单。</li></ul>

参数	是否必选	参数类型	描述
attribute	是	String	如果是tag_xxx开头，则按Tag过滤attribute属性。 否则，则按ServiceId、Appld、ServiceName、Version、Description、Level、Status过滤。
pattern	是	String	匹配规则，正则表达式，长度1到64个字符。
timestamp	否	String	创建rule的时间，只有获取rule时返回使用。
description	否	String	Rule的描述信息，字符长度不超过256。
modTimestamp	否	String	最后修改UTC时间。

## 9.2.5 AddOrUpdateRule

表 9-16 参数说明

参数	是否必选	参数类型	描述
ruleType	否	String	类型。 <ul style="list-style-type: none"><li>• WHITE表示白名单。</li><li>• BLACK表示黑名单。</li></ul>
attribute	否	String	如果是tag_xxx开头，则按Tag过滤attribute属性。 否则，则按ServiceId、Appld、ServiceName、Version、Description、Level、Status过滤。
pattern	否	String	匹配规则，正则表达式，长度1到64个字符。
description	否	String	rule的描述。 字符长度不超过256个字符。



## 9.2.6 DataCenterInfo

表 9-17 参数说明

参数	是否必选	参数类型	描述
name	是	String	数据中心名称，支持数字、字母，支持使用字符_-.做连接符，长度1-128字节。
region	是	String	数据中心所属区域名称，支持数字、字母，支持使用字符_-.做连接符，长度1-128字节。
availableZone	是	String	数据中心所属可用区名称，支持数字、字母，支持使用字符_-.做连接符，长度1-128字节。

## 9.2.7 Schema

表 9-18 参数说明

参数	是否必选	参数类型	描述
schemald	否	String	微服务契约ID。支持数字、字母和_-.字符，长度1-160字节。
schema	否	String	微服务契约内容。任意字符、长度1-2048字节。
summary	否	String	微服务契约摘要。支持数字、字母，长度0-128字节。

## 9.3 公共响应参数

### 9.3.1 WatchMicroServiceKey

表 9-19 参数说明

参数	是否必选	参数类型	描述
appId	否	String	应用app唯一标识，字符长度为1~160。正则表达式为^[a-zA-Z0-9]*\$^[a-zA-Z0-9][a-zA-Z0-9_\.]*[a-zA-Z0-9]\$。

参数	是否必选	参数类型	描述
serviceName	否	String	微服务名称。作为provider支持为*，表示依赖同一租户下的所有服务。当服务名称为*时，appld和version可以省略。consumer不支持*。
version	否	String	微服务版本。作为provider支持+、固定版本和latest（当前最新版本），如1.0.1+，表示1.0.1以上的版本，包括1.0.1。作为consumer只能为固定版本。

### 9.3.2 SuccessdResponse

表 9-20 参数说明

参数	是否必选	参数类型	描述
Result	是	String	返回结果。

### 9.3.3 ServiceInfo

表 9-21 参数说明

参数	是否必选	参数类型	描述
serviceId	是	String	微服务唯一标识，字符为uuid。正则表达式为^.*\$。
appld	是	String	应用app唯一标识，字符长度为1~160。正则表达式为^[a-zA-Z0-9]*\$ ^[a-zA-Z0-9][a-zA-Z0-9_\.]*[a-zA-Z0-9]\$。
serviceName	是	String	微服务名称，同一个App要保证唯一。字符长度为1~128。正则表达式为^[a-zA-Z0-9]*\$ ^[a-zA-Z0-9][a-zA-Z0-9_\.]*[a-zA-Z0-9]\$。
version	是	String	微服务版本号，字符长度为1~64。正则表达式为^[0-9]*\$ ^[0-9]+(\.[0-9]+)*\$。
level	是	String	微服务层级。包括： <ul style="list-style-type: none"> <li>• FRONT</li> <li>• MIDDLE</li> <li>• BACK</li> </ul>

参数	是否必选	参数类型	描述
status	是	String	服务状态。 <ul style="list-style-type: none"><li>• UP</li><li>• DOWN</li></ul>
timestamp	是	String	时间戳。

### 9.3.4 AggregateMetricInfo

表 9-22 参数说明

参数	是否必选	参数类型	描述
time	否	Integer	上报时间。
name	否	String	接口名称。
qps	否	Number	当前qps。
latency	否	Number	平均时延。
rate	否	Integer	成功率。
total	否	Integer	总请求数。
isCircuitBreakerOpen	否	Boolean	断路器是否打开。
failure	否	Integer	总失败数。
shortCircuited	否	Integer	总短路个数。
semaphoreRejected	否	Integer	总信号量拒绝个数。
threadPoolRejected	否	Integer	总线程拒绝个数。
countTimeout	否	Integer	总超时个数。

## 9.3.5 ServiceDetail

表 9-23 参数说明

参数	是否必选	参数类型	描述
microService	否	MicroService	服务信息。
instances	否	MicroServiceInstance	实例信息。
schemaInfos	否	Schema	微服务的API详细信息。
rules	否	Rule	黑白名单信息。
providers	否	MicroService	所有的provider信息。
consumers	否	MicroService	所有的consumer信息。
tags	否	Tags	微服务所有的标签信息。
microServiceVersions	否	Array, 每一项为String	微服务的所有版本信息。

表 9-24 Tags 参数说明

参数	是否必选	参数类型	描述
tags	否	Properties	微服务扩展属性请求结构体。

## 9.3.6 TenantProject

表 9-25 TenantProject 参数

参数	位于	参数类型	描述
tenant	body	String	租户名称
projectId	body	String	id

## 9.3.7 Error

表 9-26 参数说明

参数	是否必选	参数类型	描述
errorCode	是	String	错误代码。
errorMessage	是	String	错误信息。
detail	否	String	详细定位信息。

# 10 权限和授权项

## 10.1 权限及授权项说明

如果您需要对您所拥有的应用管理与运维平台（ServiceStage）进行精细的权限管理，您可以使用统一身份认证服务（Identity and Access Management，简称IAM），如果华为云账号已经能满足您的要求，不需要创建独立的IAM用户，您可以跳过本章节，不影响您使用ServiceStage服务的其他功能。

默认情况下，新建的IAM用户没有任何权限，您需要将其加入用户组，并给用户组授予策略或角色，才能使用户组中的用户获得相应的权限，这一过程称为授权。授权后，用户就可以基于已有权限对云服务进行操作。

权限根据授权的精细程度，分为**角色**和**策略**。角色以服务为粒度，是IAM最初提供的一种根据用户的工作职能定义权限的粗粒度授权机制。策略以API接口为粒度进行权限拆分，授权更加精细，可以精确到某个操作、资源和条件，能够满足企业对权限最小化的安全管控要求。

### 📖 说明

如果您要允许或是禁止某个接口的操作权限，请使用策略。

账号具备所有接口的调用权限，如果使用账号下的IAM用户发起API请求时，该IAM用户必须具备调用该接口所需的权限，否则，API请求将调用失败。每个接口所需要的权限，与各个接口所对应的授权项相对应，只有发起请求的用户被授予授权项所对应的策略，该用户才能成功调用该接口。例如，用户要调用接口来查询ServiceStage应用信息，那么这个IAM用户被授予的策略中必须包含允许“servicestage:app:get”的授权项，该接口才能调用成功。

## 支持的授权项

策略包含系统策略和自定义策略，如果系统策略不满足授权要求，管理员可以创建自定义策略，并通过给用户组授予自定义策略来进行精细的访问控制。策略支持的操作与API相对应，授权项列表说明如下：

- 权限：自定义策略中授权项定义的内容即为权限。
- 对应API接口：自定义策略实际调用的API接口。
- 授权项：自定义策略中支持的Action，在自定义策略中的Action中写入授权项，可以实现授权项对应的权限功能。

- IAM项目(Project)/企业项目(Enterprise Project)：自定义策略的授权范围，包括IAM项目与企业项目。授权范围如果同时支持IAM项目和企业项目，表示此授权项对应的自定义策略，可以在IAM和企业管理两个服务中给用户组授权并生效。如果仅支持IAM项目，不支持企业项目，表示仅能在IAM中给用户组授权并生效，如果在企业管理中授权，则该自定义策略不生效。管理员可以在授权项列表中查看授权项是否支持IAM项目或企业项目，“√”表示支持，“×”表示暂不支持。关于IAM项目与企业项目的区别，请参考[IAM与企业管理的区别](#)。

ServiceStage的支持自定义策略授权项，请参考[表10-1](#)。

**表 10-1** ServiceStage 授权项明细

权限	对应API接口	授权项	IAM项目	企业项目
创建应用	POST /v3/{project_id}/cas/applications	servicestage:app:create	√	√
删除应用	DELETE /v3/{project_id}/cas/applications/{application_id}	servicestage:app:delete	√	√
更新应用	PUT /v3/{project_id}/cas/applications/{application_id}	servicestage:app:modify	√	√
查看应用列表	GET /v3/{project_id}/cas/applications	servicestage:app:list	√	√
查询应用信息	GET /v3/{project_id}/cas/applications/{application_id}	servicestage:app:get	√	√
修改工程	-	servicestage:project:modify	√	√
创建工程	-	servicestage:project:create	√	√
审批应用	-	servicestage:app:approve	√	√
查看流水线列表	-	servicestage:pipeline:list	√	√
修改构建	-	servicestage:assembling:modify	√	√
审批流水线	-	servicestage:pipeline:review	√	√

权限	对应API接口	授权项	IAM项目	企业项目
执行流水线	-	servicestage:pipeline:execute	√	√
查看构建信息	-	servicestage:project:get	√	√
删除构建	-	servicestage:assembling:delete	√	√
删除流水线	-	servicestage:pipeline:delete	√	√
删除工程	-	servicestage:project:delete	√	√
修改流水线	-	servicestage:pipeline:modify	√	√
创建构建	-	servicestage:assembling:create	√	√
查看构建列表	-	servicestage:assembling:list	√	√
查看构建信息	-	servicestage:assembling:get	√	√
查看工程列表	-	servicestage:project:list	√	√
查看流水线信息	-	servicestage:pipeline:get	√	√
创建流水线	-	servicestage:pipeline:create	√	√



# 11 附录

## 11.1 状态码

状态码如表11-1所示。

表 11-1 状态码

状态码	编码	错误码说明
200	-	操作成功。
304	Not Modified	未修改，所请求的资源未修改返回此状态码。
400	BadRequest	非法请求。 建议直接修改该请求，不要重试该请求。
401	Unauthorized	在客户端提供认证信息后，返回该状态码，表明服务端指出客户端所提供的认证信息不正确或非法。
403	Forbidden	服务器拒绝执行客户端的请求。
404	NotFound	所请求的资源不存在。 建议直接修改该请求，不要重试该请求。
422	UnprocessableEntity	请求格式正确，但是由于含有语义错误，无法响应。
500	InternalServerError	表明服务端能被请求访问到，但是不能理解用户的请求。

## 11.2 ServiceStage 错误码

在向系统发出请求后，如果遇到错误，会在响应中包含响应的错误码描述错误信息。

当您调用API时，如果遇到“APIGW”开头的错误码，请参见[API网关错误码](#)进行处理。

错误码如下所示。

状态码	错误码	错误信息	描述	处理措施
400	SVCSTG.0010 0400	parameter invalid	错误请求	检查参数是否正确，稍后重试
400	SVCSTG.0010 0402	component was deployed, uninstall first	组件已部署， 请先卸载	卸载组件后重试
400	SVCSTG.0010 0414	job id can not empty	job id不能为 空	请指定job id 进行查询
400	SVCSTG.0010 0418	component instance action not defined	未定义组件实 例操作	请指定合法的 组件实例操作
400	SVCSTG.0010 0420	runtime not supported	不支持运行时	请指定支持的 运行时
400	SVCSTG.0010 0422	base resource can not empty	基础资源不能 为空	请设置基础资源
400	SVCSTG.0010 0425	instance artifacts can not be empty	实例的软件包 或者镜像为空	报文中的 artifact参数不 能为空
400	SVCSTG.0010 0449	environment name is already used	环境名已被占 用	重新提供一个 新的环境名
400	SVCSTG.REPO. 0400	parameter invalid	错误请求	检查参数是否 正确，稍后重 试
400	SVCSTG.REPO. 0401	repository not implemented	错误请求	请选择其他仓 库类型
400	SVCSTG.REPO. 0402	oauth code or state invalid	错误请求	请重新进行 OAuth2授权， 获取正确的认 证码和随机串
400	SVCSTG.REPO. 0403	resource already exist	错误请求	调用查询接 口，检查是否 存在重复的资 源

状态码	错误码	错误信息	描述	处理措施
400	SVCSTG.REPO.0450	no repository authorization	错误请求	检查请求header是否输入正确授权名称
400	SVCSTG.REPO.0451	repository authorization invalid	错误请求	重新授权
400	SVCSTG.REPO.0452	clone url invalid	错误请求	检查查询参数clone_url是否正确. 示例: https://user-name@github.com/user-name/demo.git
404	SVCSTG.00100403	application not found	应用未找到	请指定已经存在的应用
404	SVCSTG.00100404	component not found	组件未找到	请指定已经存在的组件
404	SVCSTG.00100408	request not found	请求未找到	请指定已经存在的请求
404	SVCSTG.00100411	environment not found	环境未找到	请指定存在的环境
404	SVCSTG.00100412	instance not found	实例未找到	请对存在的实例进行操作
404	SVCSTG.00100424	elb instance not found	elb实例未找到	请指定已经存在的elb实例
404	SVCSTG.00100429	job not found	构建工程未找到	请指定已经存在的构建工程
404	SVCSTG.00100433	application release not found	发布任务未找到	请选择已经存在的发布任务
404	SVCSTG.00100436	snapshot not found	实例快照未找到	请指定已经存在的实例快照
404	SVCSTG.00100438	template image not found	模板镜像未找到	请指定已经存在的模板镜像

状态码	错误码	错误信息	描述	处理措施
404	SVCSTG.0010 0441	instance version is already used by other instances in the same environment	组件实例的版本号已被相同环境的其他实例所占用	重新提供一个新的版本号
405	SVCSTG.0010 0421	account is limit suspended	账户已被暂停限额	恢复账号至正常状态
405	SVCSTG.0010 0427	permission denied	没有权限	重新授权
408	SVCSTG.0010 0428	request timeout	系统处理超时	系统繁忙，请稍后再试
409	SVCSTG.0010 0401	application not empty	应用不为空	删除应用下的组件再删除应用
409	SVCSTG.0010 0413	instance name exists	实例名已经存在	更换实例名
409	SVCSTG.0010 0419	can not operate in current status	当前状态不能操作	当前操作未完成，请稍后再试
409	SVCSTG.0010 0423	component instance is not deletable	组件实例当前状态不可删除	当前操作未完成，请稍后再试
409	SVCSTG.0010 0426	environment is in use	环境已经被使用	删除部署在环境上的实例后再删除环境
409	SVCSTG.0010 0432	component in releasing	组件当前已在发布任务中	请等待当前发布任务执行结束后再重试
409	SVCSTG.0010 0435	release can not delete	发布任务当前状态不能删除	当前操作未完成，请稍后再试
409	SVCSTG.0010 0439	elb is already used in other instance of component	elb实例已经在组件中被使用	请使用其他elb实例重试
500	SVCSTG.0010 0500	service internal error	服务内部错误	系统繁忙，请稍后重试

状态码	错误码	错误信息	描述	处理措施
500	SVCSTG.0010 0501	remote service error	远程服务错误	系统繁忙，请 稍后重试
500	SVCSTG.REPO. 0500	service internal error	服务内部错误	网络异常等原 因，稍后重 试。
500	SVCSTG.REPO. 0501	remote repository error	服务内部错误	检查请求参数 是否正确，或 者远端GIT的 仓库文件已经 被修改。

## 11.3 CSE 错误码

当您调用API时，如果遇到“APIGW”开头的错误码，请参见[API网关错误码](#)进行处理。

表 11-2 引擎管理类错误码

状态码	错误码	错误信息	描述	处理措施
400	SVCSTG.0050 1134	The engine type is left blank.	引擎类型不合 法	引擎类型字段 为请求query 的specType字 段，将其设置 为一个合法值 （比如CSE2） 后重试请求。 例如，查询 CSE引擎支持 规格时将 specType设置 为CSE2。
400	SVCSTG.0050 1111	Duplicate engine instance name.	引擎实例名称 重复	更换引擎名 称。
400	SVCSTG.0050 1112	Insufficient engine instance quota.	引擎实例配额 不足	联系技术支持 工程师扩大对 应引擎实例配 额，或者删除 不用的引擎实 例。

状态码	错误码	错误信息	描述	处理措施
400	SVCSTG.0050 1133	The number of engines has reached the upper limit of the site.	引擎数量达到局点上限	联系技术支持工程师处理。
400	SVCSTG.0050 1103	Engine payment is empty or is invalid.	引擎的计费方式为空或无效	传入引擎的计费方式参数，同时确认用户能否创建该计费方式的引擎。
400	SVCSTG.0050 1104	Incorrect format of the engine instance specifications.	引擎实例的规格格式错误	规格无效，请填写符合要求的规格。
400	SVCSTG.0050 1105	Incorrect AZ format.	可用区格式错误	请求参数中指定合法有效的可用区，可用区字段的类型为字符串数组，不能为空，且可用区名称层级数量范围是1-3。
400	SVCSTG.0050 1117	The engine in the current state cannot be deleted.	无法删除此状态的引擎	等待引擎状态变为可用、不可用、创建失败等状态时再重试。
400	SVCSTG.0050 1152	Incorrect task ID format.	任务ID格式错误	传入创建引擎，删除引擎等接口返回的job_id。
400	SVCSTG.0050 1153	No task found.	找不到任务	传入创建引擎，删除引擎等接口返回的job_id。
400	SVCSTG.0050 1149	Abnormal node status.	引擎节点状态异常	请稍后重试或联系技术支持工程师。

状态码	错误码	错误信息	描述	处理措施
400	SVCSTG.0050 1116	Failed to find the engine instance.	找不到引擎实例	无需特别处理，只需要检查传入的引擎ID是否正确。可以使用租户的用户名密码登录到微服务引擎控制台页面，查看当前租户下面是否有对应引擎ID的引擎。
400	SVCSTG.0050 1201	The name must contain 3 to 24 characters, including letters, digits, and hyphens (-). It must start with a letter but not end with a hyphen (-). The name cannot be default or the current engine name.	同步引擎实例名称格式错误须为字母开头，字母、数字、-组成且不能以-结尾，3-24个字符，不能命名 default，不能与引擎名称相同。	对等引擎实例名称须为字母开头，字母、数字、-组成且不能以-结尾，3-24个字符，不能命名为 default，同时不能与当前引擎实例名称相同。
400	SVCSTG.0050 1202	An HA engine can be the peer engine.	同步引擎必须为高可用引擎。	同步引擎的规格必须是高可用引擎，请重新传入引擎规格。
400	SVCSTG.0050 1203	The peer engine must be in a single AZ.	同步引擎必须在单可用区下。	同步引擎的可用区必须为单可用区，请重新传入可用区列表。
500	SVCSTG.0050 0400	Internal service request parameter error.	服务内部请求参数错误	检查报错 error_message 内容，如参数不合法，则检查输入参数是否正确；其他错误可联系技术支持工程师协助解决。

表 11-3 动态配置类错误码

状态码	错误码	错误信息	描述	处理措施
400	SVCSTG.00400649	Key/value length exceeds the upper limit.	配置项键/值长度超出限制。	请填写符合格式要求：配置项支持数字、字母和-_:[]六个字符，支持长度1-2048位，支持中文、数字、字母和括号内的特殊字符（'~!@#\$\$%^&*()_ \{} :"<>?'-[],/=），支持长度1-131072位。
400	SVCSTG.00400650	Invalid configuration file.	无效的配置文件。	上传的文件非法。
400	SVCSTG.00400651	The file is empty or in an incorrect format.	文件为空或者格式错误。	上传的文件内容为空或者文件名不是以json结尾。
400	SVCSTG.00400652	Configuration item key/value is null or not a string.	配置项键/值为空或者不是字符串。	上传的配置项键/值为空或者不是字符串，请输入正确的键/值重试
400	SVCSTG.00400653	Invalid label or body.	无效的label或者body。	下载文件时传入的body体或者query参数label不符合规范。
400	SVCSTG.00400100	Invalid parameter(s)	非法参数。	根据错误提示中的规则，修改提示中的参数。
404	SVCSTG.00400106	Record does not exist	该资源不存在。	输入正确的查找条件。
409	SVCSTG.00400107	Record already exist	该资源已存在。	请勿创建相同的记录。



状态码	错误码	错误信息	描述	处理措施
500	SVCSTG.00300605	Failed to connect the configuration center to ETCD.	服务内部错误。	内部错误请联系运维支持。
400	SVCSTG.00300401	Invalid token	认证不通过。	输入正确的 token。

## 11.4 ServiceComb 错误码

当您调用API时，如果遇到“APIGW”开头的错误码，请参见[API网关错误码](#)进行处理。

分类	状态码	错误码	错误信息	描述	处理措施
公共错误码	400	400001	Invalid parameter(s)	非法参数	根据错误提示中的规则，修改提示中的参数。
	404	404001	ErrRecoudNotExist	该资源不存在	输入正确的查找条件。
	409	409001	ErrRecordAlreadyExists	该资源已存在	请勿创建相同的记录。
	500	500003	Internal server error	内部错误	内部错误请联系运维支持。
微服务	400	400002	ErrUnhealthy	服务处于不健康状态	请稍后重试或联系技术支持工程师。
		400010	Micro-service already exists	服务已存在	修改创建微服务body中的serviceld或微服务描述信息。
		400011	ErrUnavailableBackend	没有可提供的后台实例	请稍后重试或联系技术支持工程师。
		400012	Micro-service does not exist	服务不存在	请输入有效的serviceld。

分类	状态码	错误码	错误信息	描述	处理措施
		400013	Micro-service has deployed instance(s)	无法删除该微服务，该微服务已部署实例	请先将实例下线，再删除微服务；或强制删除微服务（url中添加query参数“force=true”）。
		400014	Undefined schema id	schemald不存在	请输入有效的schemald。
		400015	Not allowed to modify schema	schema不允许修改	该schema已注册，不支持修改。
		400016	Schema does not exist	schema不存在	请先注册schema再查询。
		400017	Instance does not exist	实例不存在	请输入有效的instanceld。
		400018	ErrTagNotExists	标签不存在	通常出现在查询接口，表明标签不存在，业务根据返回值做恰当后续处理。
		400019	ErrRuleAlreadyExists	规则已经存在	重复创建规则，通常可以忽略该错误。
		400020	ErrBlackAndWhiteRule	错误的黑白名单	根据错误提示修改参数。
		400021	ErrModifyRuleNotAllow	不允许更改规则	必须修改版本号才允许更改微服务信息。
		400022	ErrRuleNotExists	规则不存在	通常出现在查询接口，表明规则不存在，业务根据返回值做恰当后续处理。
		400023	Consumer(s) depends on this micro-service	无法删除该微服务，该微服务被其他微服务依赖	可选择强制删除微服务（url中添加query参数“force=true”）。

分类	状态码	错误码	错误信息	描述	处理措施
		400024	ErrPermissionDeny	权限不允许	使用合理的账号进行操作。
		400025	ErrEndpointAlreadyExists	端口已存在	建议排查端口是否被其他实例占用。
		400026	Micro-service version does not exist	微服务版本不存在	请输入正确的版本号或版本号范围。
		400100	Not enough quota	配额不足	对应的资源（如微服务、实例、schema）配额不足，请删除部分资源再创建。
401		401204	No authorization header	认证不通过	若微服务引擎专享版开启了安全认证，此参数必填。否则，无此参数。 开启了安全认证的微服务引擎专享版Token，格式为： Authorization:Bearer {Token} Token获取方法，请参考 <a href="#">获取微服务引擎专享版用户Token</a> 。
		401201	Request unauthorized	认证不通过	输入的Authorization不合法。
403		403001	ErrForbidden	操作受限	使用合理的账号进行操作。
500		500011	Registry service is unavailable	后端错误	内部错误请联系运维支持。
		500101	ErrUnavailableQuota	没有提供配额	请稍后重试或联系技术支持工程师。
		500605	NA	配置中心Etcid连接失败	请稍后重试或联系技术支持工程师。

分类	状态码	错误码	错误信息	描述	处理措施
认证	401	401202	User name or password is wrong	账号名称或密码错误	输入正确的账号名称和密码。

## 11.5 获取项目 ID

### 从控制台获取 ID

在调用接口的时候，部分URL中需要填入ID，所以需要获取到ID。ID获取步骤如下：

- 步骤1** 登录ServiceStage控制台。
- 步骤2** 鼠标移动到右上角登录用户名上，在下拉菜单选择“我的凭证”。
- 步骤3** 在“项目列表”中查看项目ID。

图 11-1 查看项目 ID



----结束

### 调用 API 获取项目 ID

项目ID还可通过调用[查询指定条件下的项目信息](#)API获取。

获取项目ID的接口为“GET https://{Endpoint}/v3/projects”，其中{Endpoint}为IAM的终端节点，可以从[地区和终端节点](#)获取。接口的认证鉴权请参见[认证鉴权](#)。

响应示例如下，其中projects下的“id”即为项目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 获取账号 ID

在调用接口的时候，部分URL中需要填入账号ID（domain-id），所以需要先获取到账号ID。获取步骤如下：

**步骤1** 登录ServiceStage控制台。

**步骤2** 鼠标移动到右上角登录用户名上，在下拉菜单选择“我的凭证”。

查看账号ID。

图 11-2 查看账号 ID



----结束

# 12 历史 API

## 12.1 应用管理（v2 版本）

### 12.1.1 Meta

#### 12.1.1.1 获取所有支持的应用组件运行时类型

##### 功能介绍

此API用来获取所有支持应用组件运行时类型。

##### 调试

您可以在[API Explorer](#)中调试该接口。

##### URI

GET /v2/{project\_id}/cas/metadata/runtimes

表 12-1 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。

## 请求消息

表 12-2 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

## 响应消息

表 12-3 响应参数

参数	参数类型	描述
runtimes	Array of objects	运行时参数，请参考 <a href="#">表 12-4</a> 。

表 12-4 runtimes

参数	参数类型	描述
type_name	String	类型名称
display_name	String	显示名称
container_default_port	Integer	容器默认端口
type_desc	String	类型描述

## 请求示例

无

## 响应示例

```
{
  "runtimes": [
    {
      "type_name": "Java8",
      "display_name": "Java 8",
      "container_default_port": 8080,
      "type_desc": "Java 8 runtime server"
    }
  ]
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在。
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

### 12.1.1.2 获取所有支持的应用资源规格

## 功能介绍

通过此API获取所用支持的应用资源规格。

## 调试

您可以在[API Explorer](#)中调试该接口。

## URI

GET /v2/{project\_id}/cas/metadata/flavors

表 12-5 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。

## 请求消息

表 12-6 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。



参数	是否必选	参数类型	描述
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种。如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

## 响应消息

表 12-7 响应参数

参数	参数类型	描述
flavors	Array of objects	规格参数，请参考表 <a href="#">12-8</a> 。

表 12-8 flavors

参数	参数类型	描述
flavor_id	String	规格ID
storage_size	String	存储大小
num_cpu	String	CPU限制
num_cpu_init	String	CPU初始
memory_size	String	内存限制
memory_size_init	String	内存初始
label	String	展示标签
custom	boolean	是否是自定义资源规格

## 示例

无

## 响应示例

```
{
  "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  
  }  
]  
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在。
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 12.1.2 环境

### 12.1.2.1 创建环境

#### 功能介绍

此API用来创建环境。

#### 调试

您可以在[API Explorer](#)中调试该接口。

#### URI

POST /v2/{project\_id}/cas/environments

表 12-9 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。

## 请求消息

表 12-10 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种。如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

表 12-11 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	环境名称。 由英文字母、数字、中划线（-）、下划线（_）组成，并以英文字母开头，英文字母或者数字结尾，英文字母不区分大小写，长度为2~64个字符。
alias	否	String	环境别名。 最大长度为64个字符。
deploy_mode	是	String	指定环境类型。 <ul style="list-style-type: none"><li>● virtualmachine：虚拟机。</li><li>● container：Kubernetes。</li><li>● mixed：虚拟机和Kubernetes。</li></ul>
description	否	String	环境描述。 最大长度为128个字符。
enterprise_project_id	否	String	企业项目ID。
charge_mode	否	String	收费模式，支持provided、on_demanded、monthly。默认provided，表示使用用户提供的已有资源，无需收费。on_demanded表示按需收费，monthly表示包月收费。
vpc_id	是	String	虚拟私有云ID。
base_resources	是	Array of objects	基础资源，请参考 <a href="#">表12-12</a> 。

参数	是否必选	参数类型	描述
optional_resources	否	Array of objects	可选资源，请参考 <a href="#">表12-12</a> 。

表 12-12 resources

参数	是否必选	参数类型	描述
id	是	String	资源ID。
type	是	String	基础资源：cce、ecs、as。 可选资源：rds、dcs、elb、cse等其他类型。
name	否	String	资源名称。

## 响应消息

表 12-13 响应参数

参数	参数类型	描述
id	String	环境ID。
name	String	环境名称。
alias	String	环境别名。
description	String	环境描述。
project_id	String	项目ID。
enterprise_project_id	String	企业项目ID。
charge_mode	String	收费模式。
deploy_mode	String	环境类型。 <ul style="list-style-type: none"> <li>virtualmachine：虚拟机。</li> <li>container：Kubernetes。</li> <li>mixed：虚拟机和Kubernetes。</li> </ul>
vpc_id	String	虚拟私有云ID。
base_resources	Array of objects	基础资源，请参考 <a href="#">表12-14</a> ，返回为空。

参数	参数类型	描述
optional_resources	Array of objects	可选资源，请参考表12-14，返回为空。
creator	String	创建人。
create_time	Integer	创建时间。
update_time	Integer	修改时间。

表 12-14 resources

参数	参数类型	描述
id	String	资源ID。
name	String	资源名称。
type	String	基础资源：cce、ecs、as。 可选资源：rds、dcs、elb、cse等其他类型。

## 请求示例

创建一个名称为development-env的“虚拟机”类型的环境，环境所在VPC的id为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"
    }
  ]
}
```

## 响应示例

```
{
  "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
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在。
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

### 12.1.2.2 获取所有环境

#### 功能介绍

此API用来获取所有已经创建的环境。

#### 调试

您可以在[API Explorer](#)中调试该接口。

#### URI

GET /v2/{project\_id}/cas/environments

表 12-15 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。

表 12-16 查询参数

参数	是否必选	参数类型	描述
limit	否	Integer	指定个数，取值[0, 100]或者1000，指定的值不在该范围内的话相当于赋值为10。默认1000，用于不分页的场景。分页的场景取值请用[0, 100]。
offset	否	Integer	指定偏移量，表示查询第n条数据后的结果。
order_by	否	String	排序字段，默认按创建时间排序。 排序字段支持枚举值：create_time、name、update_time。传的值不在支持的枚举值范围内的话按默认排序字段排序。
order	否	String	排序方式。 <ul style="list-style-type: none"><li>• desc, 降序排序，默认为desc。</li><li>• asc, 升序排序。</li></ul>

## 请求消息

表 12-17 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种。如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

## 响应消息

表 12-18 响应参数

参数	参数类型	描述
count	Integer	环境总数。
environments	Array of objects	环境参数，请参考 <a href="#">表12-19</a> 。

表 12-19 environments

参数	参数类型	描述
id	String	环境ID。
name	String	环境名称。
alias	String	环境别名。
description	String	环境描述。
project_id	String	项目ID。
enterprise_project_id	String	企业项目ID。
charge_mode	String	收费模式。
deploy_mode	String	环境类型。 <ul style="list-style-type: none"><li>virtualmachine: 虚拟机。</li><li>container: Kubernetes。</li><li>mixed: 虚拟机和Kubernetes。</li></ul>
vpc_id	String	虚拟私有云ID。
base_resources	Array of objects	基础资源，请参考 <a href="#">表12-20</a> 。
optional_resources	Array of objects	可选资源，请参考 <a href="#">表12-20</a> 。
creator	String	创建人。
create_time	Integer	创建时间。
update_time	Integer	修改时间。



表 12-20 resources

参数	参数类型	描述
id	String	资源ID。
name	String	资源名称。
type	String	基础资源：cce、ecs、as。 可选资源：rds、dcs、elb、cse等其他类型。

## 请求示例

无

## 响应示例

```
{
  "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
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在。

状态码	描述
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

### 12.1.2.3 修改环境信息

## 功能介绍

此API通过环境ID修改环境信息。

## 调试

您可以在[API Explorer](#)中调试该接口。

## URI

PUT /v2/{project\_id}/cas/environments/{environment\_id}

表 12-21 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
environment_id	是	String	环境ID。获取方法，请参考 <a href="#">获取所有环境</a> 。

## 请求消息

表 12-22 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种。如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

表 12-23 请求 Body 参数

参数	是否必选	参数类型	描述
name	否	String	环境名称。 由英文字母、数字、中划线 (-)、下划线 (_) 组成，并以英文字母开头，英文字母或者数字结尾，英文字母不区分大小写，长度为2~64个字符。
alias	否	String	环境别名。 最大长度为64个字符。
description	否	String	环境描述。 最大长度为128个字符。

## 响应消息

表 12-24 响应参数

参数	参数类型	描述
id	String	环境ID。
name	String	环境名称。
alias	String	环境别名。
description	String	环境描述。
project_id	String	项目ID。
enterprise_project_id	String	企业项目ID。
charge_mode	String	收费模式。
deploy_mode	String	环境类型。 <ul style="list-style-type: none"><li>virtualmachine: 虚拟机。</li><li>container: Kubernetes。</li><li>mixed: 虚拟机和Kubernetes。</li></ul>
vpc_id	String	虚拟私有云ID。
base_resources	Array of objects	基础资源，请参考 <a href="#">表12-25</a> 。
optional_resources	Array of objects	可选资源，请参考 <a href="#">表12-25</a> 。
creator	String	创建人。

参数	参数类型	描述
create_time	Integer	创建时间。
update_time	Integer	修改时间。

表 12-25 resources

参数	参数类型	描述
id	String	资源ID。
name	String	资源名称。
type	String	基础资源：cce、ecs、as。 可选资源：rds、dcs、elb、cse等其他类型。

## 请求示例

修改名称为development-env2的环境，增加环境描述信息。

```
{
  "name": "development-env2",
  "description": "here is description"
}
```

## 响应示例

```
{
  "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
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在。
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

### 12.1.2.4 根据环境 ID 删除环境

#### 功能介绍

此API通过环境ID删除环境。

#### 调试

您可以在[API Explorer](#)中调试该接口。

#### URI

DELETE /v2/{project\_id}/cas/environments/{environment\_id}

表 12-26 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
environment_id	是	String	环境ID。获取方法，请参考 <a href="#">获取所有环境</a> 。

## 请求消息

表 12-27 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种。如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

## 响应消息

无

## 请求示例

无

## 响应示例

无

## 状态码

状态码	描述
204	操作成功。
400	错误的请求。
404	请求对象不存在。
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

### 12.1.2.5 根据环境 ID 获取环境详细信息

## 功能介绍

此API通过环境ID获取环境详细信息。

## 调试

您可以在[API Explorer](#)中调试该接口。

## URI

GET /v2/{project\_id}/cas/environments/{environment\_id}

表 12-28 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
environment_id	是	String	环境ID。获取方法，请参考 <a href="#">获取所有环境</a> 。

## 请求消息

表 12-29 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种。如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

## 响应消息

表 12-30 响应参数

参数	参数类型	描述
id	String	环境ID。
name	String	环境名称。
alias	String	环境别名。
description	String	环境描述。
project_id	String	项目ID。
enterprise_project_id	String	企业项目ID。

参数	参数类型	描述
charge_mode	String	收费模式。
deploy_mode	String	环境类型。 <ul style="list-style-type: none"><li>• virtualmachine: 虚拟机。</li><li>• container: Kubernetes。</li><li>• mixed : 虚拟机和Kubernetes。</li></ul>
vpc_id	String	虚拟私有云ID。
base_resources	Array of objects	基础资源, 请参考表12-31。
optional_resources	Array of objects	可选资源, 请参考表12-31。
creator	String	创建人。
create_time	Integer	创建时间。
update_time	Integer	修改时间。

表 12-31 resources

参数	参数类型	描述
id	String	资源ID。
name	String	资源名称。
type	String	基础资源: cce、ecs、as。 可选资源: rds、dcs、elb、cse等其他类型。

## 请求示例

无

## 响应示例

```
{
  "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  
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在。
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

### 12.1.2.6 修改环境资源

#### 功能介绍

此API用来修改环境资源。

#### 调试

您可以在[API Explorer](#)中调试该接口。

#### URI

PATCH /v2/{project\_id}/cas/environments/{environment\_id}/resources

表 12-32 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
environment_id	是	String	环境ID。获取方法，请参考 <a href="#">获取所有环境</a> 。

## 请求消息

表 12-33 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种。如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

表 12-34 请求 Body 参数

参数	是否必选	参数类型	描述
add_base_resources	否	Array of objects	添加基础资源，请参考 <a href="#">表12-35</a> 。
add_optional_resources	否	Array of objects	添加可选资源，请参考 <a href="#">表12-36</a> 。
remove_resources	否	Array of objects	移除资源，请参考 <a href="#">表12-37</a> 。

表 12-35 add\_base\_resources

参数	是否必选	参数类型	描述
id	是	String	资源ID（为了保证接口调用正常，请您确保调用前输入正确的资源ID）。
name	否	String	资源名称。
type	是	String	基础资源：cce、ecs、as。

表 12-36 add\_optional\_resources

参数	是否必选	参数类型	描述
id	是	String	资源ID（为了保证接口调用正常，请您确保调用前输入正确的资源ID）。
name	否	String	资源名称。

参数	是否必选	参数类型	描述
type	是	String	可选资源：rds、dcs、elb、cse等其他类型。

表 12-37 remove\_resources

参数	是否必选	参数类型	描述
id	是	String	资源ID（为了保证接口调用正常，请您确保调用前输入正确的资源ID）。
name	否	String	资源名称。
type	是	String	基础资源：cce、ecs、as。当 deploy_mode为virtualmachine时可添加ecs、as，为container时可添加cce资源。 可选资源：rds、dcs、elb、cse等其他类型。

## 响应消息

表 12-38 响应参数

参数	参数类型	描述
id	String	环境ID。
name	String	环境名称。
alias	String	环境别名。
description	String	环境描述。
project_id	String	项目ID。
enterprise_project_id	String	企业项目ID。
charge_mode	String	收费模式。
deploy_mode	String	环境类型。 <ul style="list-style-type: none"> <li>virtualmachine：虚拟机。</li> <li>container：Kubernetes。</li> <li>mixed：虚拟机和Kubernetes。</li> </ul>
vpc_id	String	虚拟私有云ID。

参数	参数类型	描述
base_resources	Array of objects	基础资源。请参考表12-39。
optional_resources	Array of objects	可选资源。请参考表12-39。
creator	String	创建人。
create_time	Integer	创建时间。
update_time	Integer	修改时间。

表 12-39 schemas

参数	参数类型	描述
id	String	资源ID。
name	String	资源名称。
type	String	基础资源：cce、ecs、as。 可选资源：rds、dcs、elb、cse等其他类型。

## 请求示例

修改环境资源，添加名称为ecs-9876的ECS基础资源、专业版CSE可选资源，移除名称为test的ECS资源。

```
{
  "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"
    }
  ]
}
```

## 响应示例

```
{
  "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
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在。
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 12.1.3 应用

### 12.1.3.1 创建应用

#### 功能介绍

应用是一个功能相对完备的业务系统，由一个或多个特性相关的组件组成。

此API用来创建应用。

## 调试

您可以在[API Explorer](#)中调试该接口。

## URI

POST /v2/{project\_id}/cas/applications

表 12-40 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。

## 请求消息

表 12-41 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种。如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

表 12-42 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	应用名称。 由大小写英文字母、数字、中划线(-)、下划线(_)组成，并以大小写英文字母开头，大小写英文字母或者数字结尾，长度为2~64个字符。
description	否	String	应用描述。 最大长度为128个字符。
enterprise_project_id	否	String	企业项目ID。

## 响应消息

表 12-43 响应参数

参数	参数类型	描述
id	String	应用ID。
name	String	应用名称。
description	String	应用描述。
creator	String	创建人。
project_id	String	项目ID。
enterprise_project_id	String	企业项目ID。
create_time	Integer	创建时间。
update_time	Integer	修改时间。
unified_model	String	是否开启统一模型，返回固定值null。

## 请求示例

创建名称为app-xpmtii的应用，输入应用描述信息为test。

```
{  
  "name": "app-xpmtii",  
  "description": "test"  
}
```

## 响应示例

```
{  
  "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  
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在。

状态码	描述
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

### 12.1.3.2 获取所有应用

## 功能介绍

通过此API可以获取所有已经创建的应用。

## 调试

您可以在[API Explorer](#)中调试该接口。

## URI

GET /v2/{project\_id}/cas/applications

表 12-44 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。

表 12-45 查询参数

参数	是否必选	参数类型	描述
limit	否	String	指定个数，取值[0, 100]或者1000，指定的值不在该范围内的话相当于赋值为10。默认1000，用于不分页的场景。分页的场景取值请用[0, 100]。
offset	否	String	指定偏移量，表示查询第n条数据后的结果。
order_by	否	String	排序字段，默认按创建时间排序。 排序字段支持枚举值：create_time, name, update_time。传的值不在支持的枚举值范围内的话按默认排序字段排序。



参数	是否必选	参数类型	描述
order	否	String	排序方式。 <ul style="list-style-type: none"> <li>• desc, 降序排序, 默认为desc。</li> <li>• asc, 升序排序。</li> </ul>

## 请求消息

表 12-46 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种。如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

## 响应消息

表 12-47 响应参数

参数	参数类型	描述
count	Integer	应用总数。
applications	Array of objects	应用信息，请参考 <a href="#">表12-48</a> 。

表 12-48 applications

参数	参数类型	描述
id	String	应用ID。
name	String	应用名称。
description	String	应用描述。
creator	String	创建人。
project_id	String	项目ID。

参数	参数类型	描述
enterprise_project_id	String	企业项目ID。
create_time	Integer	创建时间。
update_time	Integer	修改时间。
unified_model	String	是否开启统一模型，返回固定值null。
component_count	Integer	组件个数。

## 请求示例

无

## 响应示例

```
{
  "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
    }
  ]
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在。
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

### 12.1.3.3 修改应用信息

#### 功能介绍

此API通过应用ID修改应用信息。

#### 调试

您可以在[API Explorer](#)中调试该接口。

#### URI

PUT /v2/{project\_id}/cas/applications/{application\_id}

表 12-49 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
application_id	是	String	应用ID。获取方法，请参考 <a href="#">获取所有应用</a> 。

#### 请求消息

表 12-50 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

表 12-51 请求 Body 参数

参数	是否必选	参数类型	描述
name	否	String	应用名称。 由大小写英文字母、数字、中划线(-)、下划线(_)组成,并以大小写英文字母开头,大小写英文字母或者数字结尾,长度为2~64个字符。
description	否	String	应用描述。 最大长度为128个字符。

## 响应消息

表 12-52 响应参数

参数	参数类型	描述
id	String	应用ID。
name	String	应用名称。
description	String	应用描述。
creator	String	创建人。
project_id	String	项目ID。
enterprise_project_id	String	企业项目ID。
create_time	Integer	创建时间。
update_time	Integer	修改时间。
unified_model	String	是否开启统一模型,返回固定值null。

## 请求示例

修改应用名称为app-test。

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

## 响应示例

```
{
  "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  
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在。
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

### 12.1.3.4 根据应用 ID 删除应用

#### 功能介绍

此API通过应用ID删除应用。

#### 调试

您可以在[API Explorer](#)中调试该接口。

#### URI

DELETE /v2/{project\_id}/cas/applications/{application\_id}

表 12-53 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
application_id	是	String	应用ID。获取方法，请参考 <a href="#">获取所有应用</a> 。

## 请求消息

表 12-54 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

## 响应消息

无

## 请求示例

无

## 响应示例

无

## 状态码

状态码	描述
204	操作成功。
400	错误的请求。
404	请求对象不存在。
409	请求与当前对象状态冲突。
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

### 12.1.3.5 根据应用 ID 获取应用详细信息

## 功能介绍

此API通过应用ID获取应用详细信息。

## 调试

您可以在[API Explorer](#)中调试该接口。

## URI

GET /v2/{project\_id}/cas/applications/{application\_id}

表 12-55 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
application_id	是	String	应用ID。获取方法，请参考 <a href="#">获取所有应用</a> 。

## 请求消息

表 12-56 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

## 响应消息

表 12-57 响应参数

参数	参数类型	描述
id	String	应用ID。
name	String	应用名称。
description	String	应用描述。
creator	String	创建人。
project_id	String	项目ID。
enterprise_project_id	String	企业项目ID。

参数	参数类型	描述
create_time	Integer	创建时间。
update_time	Integer	修改时间。
unified_model	String	是否开启统一模型，返回固定值null。
component_count	Integer	包含组件个数。

## 请求示例

无

## 响应示例

```
{
  "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
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在。
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

### 12.1.3.6 修改应用配置信息

## 功能介绍

通过此API修改应用配置信息。

应用配置主要是指应用的公共环境变量，该应用下部署的组件可以继承。



## 调试

您可以在[API Explorer](#)中调试该接口。

## URI

PUT /v2/{project\_id}/cas/applications/{application\_id}/configuration

表 12-58 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
application_id	是	String	应用ID。获取方法，请参考 <a href="#">获取所有应用</a> 。

## 请求消息

表 12-59 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

表 12-60 请求 Body 参数

参数	是否必选	参数类型	描述
environment_id	是	String	环境ID，可以通过调用 <a href="#">获取所有环境</a> 找到对应的环境获取。
configuration	是	Object	应用配置信息，如公共环境变量，请参考 <a href="#">表12-61</a> 。

表 12-61 configuration

参数	是否必选	参数类型	描述
env	是	Array of objects	应用环境变量，请参考 <a href="#">表12-62</a> 。 如果多个环境变量的名称重复，则只有最后一个会生效。

表 12-62 env

参数	是否必选	参数类型	描述
name	是	String	环境变量名称。 由字母、数字、下划线、中划线或者点号组成，不能以数字开头，且长度为1-64个字符。
value	是	String	环境变量取值。

#### 须知

此接口的请求参数configuration中env变量会全量覆盖应用本身已有的环境变量配置，使用时请根据是新增还是替换环境变量的场景设置此参数。

## 响应消息

表 12-63 响应参数

参数	参数类型	描述
application_id	String	应用ID。
environment_id	String	环境ID。
configuration	Object	应用配置信息，请参考 <a href="#">表12-64</a> 。

表 12-64 configuration

参数	参数类型	描述
env	Array of objects	应用环境变量，请参考 <a href="#">表12-65</a> 。

表 12-65 env

参数	参数类型	描述
name	String	环境变量名称
value	String	环境变量取值

## 请求示例

修改应用对应环境id为0fa75dfe-0d32-4bc1-848b-8008cf3f2567的环境变量配置：名为app-env、值为env的环境变量。

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

## 响应示例

```
{
  "application_id": "06cf2fda-af45-44b4-9e22-6294c4804515",
  "environment_id": "0fa75dfe-0d32-4bc1-848b-8008cf3f2567",
  "configuration": {
    "env": [
      {
        "name": "app-env",
        "value": "env"
      }
    ]
  }
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在。
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

### 12.1.3.7 删除应用配置

#### 功能介绍

通过此API删除应用配置信息。

#### 调试

您可以在[API Explorer](#)中调试该接口。

#### URI

DELETE /v2/{project\_id}/cas/applications/{application\_id}/configuration

表 12-66 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
application_id	是	String	应用ID。获取方法，请参考 <a href="#">获取所有应用</a> 。

表 12-67 查询参数

参数	是否必选	参数类型	描述
environment_id	是	String	环境ID。获取方法，请参考 <a href="#">获取所有环境</a> 。

#### 请求消息

表 12-68 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

## 响应消息

无

## 请求示例

无

## 响应示例

无

## 状态码

状态码	描述
204	操作成功。
400	错误的请求。
404	请求对象不存在。
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

### 12.1.3.8 获取应用配置

## 功能介绍

通过此API获取应用配置信息。

## 调试

您可以在[API Explorer](#)中调试该接口。

## URI

GET /v2/{project\_id}/cas/applications/{application\_id}/configuration

表 12-69 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
application_id	是	String	应用ID。获取方法，请参考 <a href="#">获取所有应用</a> 。

表 12-70 查询参数

参数	是否必选	参数类型	描述
environment_id	否	String	环境ID，如果未提供，查询所有环境。获取方法，请参考 <a href="#">获取所有环境</a> 。

## 请求消息

表 12-71 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

## 响应消息

表 12-72 响应参数

参数	参数类型	描述
configuration	Array of objects	配置信息，请参考 <a href="#">表12-73</a> 。

表 12-73 应用配置信息

参数	参数类型	描述
application_id	String	应用ID。
environment_id	String	环境ID。
configuration	Object	应用配置信息，请参考 <a href="#">表12-74</a> 。

表 12-74 configuration

参数	参数类型	描述
env	Array of objects	应用组件环境变量，请参考 <a href="#">表12-75</a> 。

表 12-75 env

参数	参数类型	描述
name	String	环境变量名称
value	String	环境变量取值

## 请求示例

无

## 响应示例

```
{
  "configuration": [
    {
      "application_id": "e5213b62-0e3c-476f-9960-3e4108787350",
      "environment_id": "00078e9d-a61c-476e-ac63-a10c9cb2638e",
      "configuration": {
        "env": [
          {
            "name": "env",
            "value": "value1"
          }
        ]
      }
    }
  ]
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在。
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 12.1.4 组件

### 12.1.4.1 应用中创建组件

#### 功能介绍

应用组件是组成应用的某个业务特性实现，以代码或者软件包为载体，可独立部署运行。

此API用来在应用中创建组件。

#### 调试

您可以在[API Explorer](#)中调试该接口。

#### URI

POST /v2/{project\_id}/cas/applications/{application\_id}/components

表 12-76 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
application_id	是	String	应用ID。获取方法，请参考 <a href="#">获取所有应用</a> 。

#### 请求消息

表 12-77 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。



表 12-78 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	应用组件名称。 由英文字母、数字、中划线(-)、下划线(_)组成,并以英文字母开头,英文字母或者数字结尾,不区分英文字母大小写,长度为2~64个字符。
runtime	是	String	运行时,可查询 <a href="#">获取所有支持的应用组件运行时类型</a> 接口返回的type_name获取。
category	是	String	应用组件类型如: Webapp、MicroService、Common。
sub_category	否	String	应用组件子类型。 Webapp的子类型有Web。 MicroService的子类型有Java Chassis、Spring Cloud。 Common的子类型可以为空。
description	否	String	描述。 最大长度为128个字符。
source	否	Object	代码/软件包来源,请参考 <a href="#">表12-79</a> 。
build	否	Object	组件构建,请参考 <a href="#">表12-83</a> 。

表 12-79 source

参数	是否必选	参数类型	描述
kind	是	String	类型,支持源码code和artifact软件包。
spec	是	Object	对应code请参考 <a href="#">表12-80</a> ;对应artifact请参考 <a href="#">表12-81</a> 。

表 12-80 code spec

参数	是否必选	参数类型	描述
repo_type	否	String	代码仓类型,支持CodeArts、Gitee、GitHub、BitBucket、GitLab。
repo_url	否	String	代码仓url,如: https://github.com/example/demo.git。

参数	是否必选	参数类型	描述
repo_ref	否	String	代码分支或者Tag，默认是master。
repo_auth	否	String	授权名称，在授权列表获取。

表 12-81 artifact spec

参数	是否必选	参数类型	描述
storage	否	String	存储方式，支持软件仓库swr、镜像仓库swr、对象存储obs、软件发布库codearts。
type	否	String	类别。 <ul style="list-style-type: none"><li>虚拟机部署支持软件包package。</li><li>容器部署支持软件包package、镜像包image。</li></ul>
url	否	String	软件包/源码地址。
auth	否	String	认证方式，支持iam，none，默认是iam。
properties	否	Object	软件包的其他属性，只有在选择对象存储obs的时候才需要添加。请参考表12-82。
webUrl	否	String	软件仓库地址。

表 12-82 artifact spec properties

参数	是否必选	参数类型	描述
endpoint	否	String	obs的终端地址，如：https://obs.region_id.external_domain_name.com。
bucket	否	String	软件包在obs的桶名。
key	否	String	obs桶中的对象，一般是软件包名，有文件夹的话要加上文件夹的路径。比如test.jar或者demo/test.jar。

表 12-83 build

参数	是否必选	参数类型	描述
parameters	否	Map<String, Object>	请参考 <a href="#">表12-84</a> 。 只在没有ID，新创建构建时提供。

表 12-84 parameters

参数	是否必选	参数类型	描述
build_cmd	否	String	编译命令。默认： 1. 根目录存在build.sh: ./build.sh 2. 根据运行系统，示例如下： <ul style="list-style-type: none"><li>• Java和Tomcat: <b>mvn clean package</b></li><li>• Nodejs: <b>npm build</b></li></ul>
dockerfile_path	否	String	dockerfile地址。默认是根目录./。
artifact_namespace	否	String	构建归档组织，默认cas_{project_id}。
cluster_id	否	String	指定构建集群的id。
node_label_selector	否	Map<String, String>	key是标签的键，value是标签的值。

## 响应消息

表 12-85 响应参数

参数	参数类型	描述
id	String	应用组件ID。
name	String	应用组件名称。
status	Integer	取值0或1。 0: 表示正常状态。 1: 表示正在删除。
runtime	String	运行时。
category	String	应用组件类型如: Webapp、MicroService、Common。

参数	参数类型	描述
sub_category	String	应用组件子类型。 Webapp的子类型有Web。 MicroService的子类型有Java Chassis、Spring Cloud。 Common的子类型可以为空。
description	String	描述。
project_id	String	项目ID。
application_id	String	应用ID。
source	Object	代码/软件包来源，请参考表12-86。
build	Object	构建工程，请参考表12-90。
create_time	Integer	创建时间。
update_time	Integer	修改时间。
creator	String	创建者。

表 12-86 source

参数	参数类型	描述
kind	String	类型，支持源码code和artifact软件包。
spec	Object	对应artifact请参考表12-88；对应code请参考表12-87。

表 12-87 code spec

参数	参数类型	描述
repo_type	String	代码仓类型，支持CodeArts、Gitee、GitHub、BitBucket、GitLab。
repo_url	String	代码仓url，如：https://github.com/example/demo.git。
repo_ref	String	代码分支或者Tag，默认是master。
repo_auth	String	授权名称，在授权列表获取。

表 12-88 artifact spec

参数	参数类型	描述
storage	String	存储方式。
type	String	类别。
url	String	软件包/源码地址。
auth	String	认证方式。
properties	Object	软件包的其他属性，只有在选择对象存储obs的时候才需要添加。请参考 <a href="#">表12-89</a> 。

表 12-89 artifact spec properties

参数	参数类型	描述
endpoint	String	obs的终端地址，如：https://obs.region_id.external_domain_name.com。
bucket	String	软件包在obs的桶名。
key	String	obs桶中的对象，一般是软件包名，有文件夹的话要加上文件夹的路径。比如test.jar或者demo/test.jar。

表 12-90 build

参数	参数类型	描述
id	String	类型。
parameters	Map<String, Object>	请参考 <a href="#">表12-91</a> 。

表 12-91 parameters

参数	参数类型	描述
build_cmd	String	编译命令。
dockerfile_path	String	dockerfile地址。
artifact_names pace	String	构建归档组织。
cluster_id	String	指定构建集群的id。
node_label_sel ector	Map<String, String>	key是标签的键，value是标签的值。

## 请求示例

创建软件包部署的应用微服务组件，运行时设置为Java8，组件软件包来源选择存储在obs下myapp桶中的demo.jar包。

```
{
  "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"
      }
    }
  }
}
```

## 响应示例

```
{
  "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": "w3dpv7p0t1vpxvey5hjb22iuwxway1vupwx0nae1",
    "parameters": {
      "artifact_namespace": "ns",
      "dockerfile_path": "./"
    }
  }
}
```

```
},  
"status": 0,  
"creator": "test_user",  
"create_time": 1610333934288,  
"update_time": 1610333934288  
}
```

## 状态码

状态码	描述
200	操作成功
400	错误的请求
404	请求对象不存在
500	内部错误

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

### 12.1.4.2 获取应用所有组件

## 功能介绍

通过此API获取应用下所有应用组件。

## 调试

您可以在[API Explorer](#)中调试该接口。

## URI

GET /v2/{project\_id}/cas/applications/{application\_id}/components

表 12-92 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
application_id	是	String	应用ID。获取方法，请参考 <a href="#">获取所有应用</a> 。

表 12-93 查询参数

参数	是否必选	参数类型	描述
limit	否	String	指定个数，取值[0, 100]或者1000，指定的值不在该范围内的话相当于赋值为10。默认1000，用于不分页的场景。分页的场景取值请用[0, 100]。
offset	否	String	指定偏移量。
order_by	否	String	排序字段，默认按创建时间排序。 排序字段支持枚举值：create_time、name、update_time。
order	否	String	排序方式。 <ul style="list-style-type: none"><li>• desc，降序排序，默认为desc。</li><li>• asc，升序排序。</li></ul>

## 请求消息

表 12-94 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

## 响应消息

表 12-95 响应参数

参数	参数类型	描述
count	Integer	组件总数。
components	Array of objects	组件参数，请参考 <a href="#">表12-96</a> 。



表 12-96 components

参数	参数类型	描述
id	String	应用组件ID。
name	String	应用组件名称。
status	integer	取值0或1。 0: 表示正常状态。 1: 表示正在删除。
runtime	String	运行时。
category	String	应用组件类型如: Webapp、MicroService、Common。
sub_category	String	应用组件子类型。 Webapp的子类型有Web。 MicroService的子类型有Java Chassis、Spring Cloud。 Common的子类型可以为空。
application_id	String	应用ID。
source	Object	代码/软件包来源, 请参考 <a href="#">表12-97</a> 。
description	String	描述。
creator	String	创建人。
create_time	Integer	创建时间。
update_time	Integer	修改时间。
project_id	String	租户项目ID。
build	Object	构建工程, 请参考 <a href="#">表12-101</a> 。

表 12-97 source

参数	参数类型	描述
kind	String	类型, 支持源码code和artifact软件包。
spec	Object	对应code请参考 <a href="#">表12-98</a> ; 对应artifact请参考 <a href="#">表12-99</a> 。

表 12-98 code spec

参数	参数类型	描述
repo_type	String	代码仓类型，支持CodeArts、Gitee、GitHub、BitBucket、GitLab。
repo_url	String	代码仓url，如：https://github.com/example/demo.git。
repo_ref	String	代码分支或者Tag，默认是master。
repo_auth	String	授权名称，在授权列表获取。

表 12-99 artifact spec

参数	参数类型	描述
storage	String	存储方式。
type	String	类别。
url	String	软件包/源码地址。
auth	String	认证方式。
properties	Object	软件包的其他属性，只有在选择对象存储obs的时候才需要添加。请参考表12-100。

表 12-100 artifact spec properties

参数	参数类型	描述
endpoint	String	obs的终端地址，比如：https://obs.region_name.external_domain_name.com。
bucket	String	软件包在obs的桶名。
key	String	obs桶中的对象，一般是软件包名，有文件夹的话要加上文件夹的路径。比如test.jar或者demo/test.jar。

表 12-101 build

参数	参数类型	描述
id	String	构建工程id。
parameters	Map<String,String>	请参考表12-102。

表 12-102 parameters

参数	参数类型	描述
build_cmd	String	编译命令。
dockerfile_path	String	dockerfile地址。
artifact_namespace	String	构建归档组织。
cluster_id	String	指定构建集群的id。
node_label_selector	Map<String, String>	key是标签的键，value是标签的值。

### 请求示例

无

### 响应示例

```
{
  "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": "w3dpv7p0t1vpxvey5hjb22iuwxway1vupwx0nae1",
    "parameters": {
      "artifact_namespace": "ns",
      "use_public_cluster": true
    }
  },
  "status": 0,
  "creator": "test_user",
  "create_time": 1610333934288,
  "update_time": 1610333934288
},
"count": 1
}
```

## 状态码

状态码	描述
200	操作成功
400	错误的请求
404	请求对象不存在
500	内部错误

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

### 12.1.4.3 根据组件 ID 修改组件信息

#### 功能介绍

此API通过组件ID修改组件信息。

#### 调试

您可以在[API Explorer](#)中调试该接口。

#### URI

PUT /v2/{project\_id}/cas/applications/{application\_id}/components/{component\_id}

表 12-103 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
application_id	是	String	应用ID。获取方法，请参考 <a href="#">获取所有应用</a> 。
component_id	是	String	组件ID。获取方法，请参考 <a href="#">获取应用所有组件</a> 。

## 请求消息

表 12-104 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

表 12-105 请求 Body 参数

参数	是否必选	参数类型	描述
name	否	String	应用组件名称。 由英文字母、数字、中划线（-）、下划线（_）组成，并以英文字母开头，英文字母或者数字结尾，不区分英文字母大小写，长度为2~64个字符。
description	否	String	描述。 最大长度为128个字符。
source	否	Object	代码/软件包来源，请参考 <a href="#">表12-106</a> 。
build	否	Object	构建工程，请参考 <a href="#">表12-110</a> 。

表 12-106 source

参数	是否必选	参数类型	描述
kind	是	String	类型，支持源码code和artifact软件包。
spec	是	Object	对应code请参考 <a href="#">表12-107</a> ；对应artifact请参考 <a href="#">表12-108</a> 。

表 12-107 code spec

参数	是否必选	参数类型	描述
repo_type	否	String	代码仓类型，支持CodeArts、Gitee、GitHub、BitBucket、GitLab。
repo_url	否	String	代码仓url，如： <a href="https://github.com/example/demo.git">https://github.com/example/demo.git</a> 。

参数	是否必选	参数类型	描述
repo_ref	否	String	代码分支或者Tag，默认是master。
repo_auth	否	String	授权名称，在授权列表获取。

表 12-108 artifact spec

参数	是否必选	参数类型	描述
storage	否	String	存储方式，支持软件仓库swr、镜像仓库swr、对象存储obs、软件发布库codearts。
type	否	String	类别。 <ul style="list-style-type: none"><li>虚拟机部署支持软件包package。</li><li>容器部署支持软件包package、镜像包image。</li></ul>
url	否	String	软件包/源码地址。
auth	否	String	认证方式：支持iam、none。默认是iam。
properties	否	Object	软件包的其他属性，只有在选择对象存储obs的时候才需要添加。 <a href="#">表12-109</a> 。
webUrl	否	String	软件仓库地址。

表 12-109 artifact spec properties

参数	是否必选	参数类型	描述
endpoint	否	String	obs的终端地址，比如： <a href="https://obs.region_id.external_domain_name.com">https://obs.region_id.external_domain_name.com</a> 。
bucket	否	String	软件包在obs的桶名。
key	否	String	obs桶中的对象，一般是软件包名，有文件夹的话要加上文件夹的路径。比如test.jar或者demo/test.jar。

表 12-110 build

参数	是否必选	参数类型	描述
parameters	否	Map<String, Object>	请参考表12-111。只有在没有ID，新创建构建时提供。

表 12-111 parameters

参数	是否必选	参数类型	描述
build_cmd	否	String	编译命令。默认： 1. 根目录存在build.sh: ./build.sh 2. 根据运行系统，示例如下： <ul style="list-style-type: none"><li>• Java和Tomcat: <b>mvn clean package</b></li><li>• Nodejs: <b>npm build</b></li></ul>
dockerfile_path	否	String	dockerfile地址。默认是根目录./。
artifact_namespace	否	String	构建归档组织。默认cas_{project_id}。
cluster_id	是	String	指定构建集群的id。
node_label_selector	否	Map<String, String>	key是标签的键，value是标签的值。
cluster_name	否	String	指定构建集群的名称。
use_public_cluster	否	Boolean	是否使用公共集群： <ul style="list-style-type: none"><li>• true: 是</li><li>• false: 否</li></ul>
cluster_type	否	String	指定构建集群的类型。

## 响应消息

表 12-112 响应参数

参数	参数类型	描述
id	String	应用组件ID。
name	String	应用组件名称。

参数	参数类型	描述
status	integer	取值0或1。 0: 表示正常状态。 1: 表示正在删除。
runtime	String	运行时。
category	String	应用组件类型如: Webapp、MicroService、Common。
sub_category	String	应用组件子类型。 Webapp的子类型有Web。 MicroService的子类型有Java Chassis、Spring Cloud。 Common的子类型可以为空。
description	String	描述。
project_id	String	项目ID。
application_id	String	应用ID。
source	Object	代码/软件包来源, 请参考表12-113。
build	Object	构建工程, 请参考表12-117。
create_time	Integer	创建时间。
update_time	Integer	修改时间。
creator	String	创建者。

表 12-113 source

参数	参数类型	描述
kind	String	类型。
spec	Object	对应code请参考表12-114; 对应artifact请参考表12-115。

表 12-114 code spec

参数	参数类型	描述
repo_type	String	代码仓类型, 支持devcloud、gitee、github、bitbucket、gitlab。
repo_url	String	代码仓url, 如: https://github.com/example/demo.git。



参数	参数类型	描述
repo_ref	String	代码分支或者Tag，默认是master。
repo_auth	String	授权名称，在授权列表获取。

表 12-115 spec

参数	参数类型	描述
storage	String	存储方式。
type	String	类别。
url	String	软件包/源码地址。
auth	String	认证方式。
properties	Object	软件包的其他属性，只有在选择对象存储obs的时候才需要添加，请参考 <a href="#">表12-116</a> 。

表 12-116 artifact spec properties

参数	参数类型	描述
endpoint	String	obs的终端地址，比如：https://obs.region_id.external_domain_name.com。
bucket	String	软件包在obs的桶名。
key	String	obs桶中的对象，一般是软件包名，有文件夹的话要加上文件夹的路径。比如test.jar或者demo/test.jar。

表 12-117 build

参数	参数类型	描述
id	String	类型。
parameters	Map<String, Object>	请参考 <a href="#">表12-118</a> 。

表 12-118 parameters

参数	参数类型	描述
build_cmd	String	编译命令。

参数	参数类型	描述
dockerfile_path	String	dockerfile地址。
artifact_name_space	String	构建归档组织。
cluster_id	String	指定构建集群的id。
node_label_selector	Map<String, String>	key是标签的键，value是标签的值。

## 请求示例

修改指定组件ID对应的组件信息，修改组件名称为test-micro，设置组件软件包来源为存储在obs下test-soft桶中的fusionweather-1.0.0.jar包，指定cce-test集群为组件构建集群。

```
{
  "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"
    }
  }
}
```

## 响应示例

```
{
  "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
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在。
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

### 12.1.4.4 根据组件 ID 删除组件

## 功能介绍

此API通过组件ID删除组件。

## 调试

您可以在[API Explorer](#)中调试该接口。

## URI

```
DELETE /v2/{project_id}/cas/applications/{application_id}/components/
{component_id}
```

表 12-119 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
application_id	是	String	应用ID。获取方法，请参考 <a href="#">获取所有应用</a> 。
component_id	是	String	组件ID。获取方法，请参考 <a href="#">获取应用所有组件</a> 。

表 12-120 查询参数

参数	是否必选	参数类型	描述
force	否	boolean	是否强制删除。

## 请求消息

表 12-121 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

## 响应消息

无

## 请求示例

无

## 响应示例

无

## 状态码

状态码	描述
204	操作成功
400	错误的请求
404	请求对象不存在
409	请求与当前对象状态冲突
500	内部错误

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

### 12.1.4.5 根据组件 ID 获取组件信息

#### 功能介绍

通过组件ID获取组件信息。

#### 调试

您可以在[API Explorer](#)中调试该接口。

#### URI

GET /v2/{project\_id}/cas/applications/{application\_id}/components/{component\_id}

表 12-122 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
application_id	是	String	应用ID。获取方法，请参考 <a href="#">获取所有应用</a> 。
component_id	是	String	组件ID。获取方法，请参考 <a href="#">获取应用所有组件</a> 。

## 请求消息

表 12-123 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式）
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

## 响应消息

表 12-124 响应参数

参数	参数类型	描述
id	String	应用组件ID。
name	String	应用组件名称。
status	integer	取值0或1。 0：表示正常状态。 1：表示正在删除。
runtime	String	运行时。
category	String	应用组件类型如：Webapp、MicroService、Common。
sub_category	String	应用组件子类型。 Webapp的子类型有Web。 MicroService的子类型有Java Chassis、Spring Cloud。 Common的子类型可以为空。
description	String	描述。
project_id	String	项目ID。
application_id	String	应用ID。
source	Object	代码/软件包来源，请参考 <a href="#">表12-125</a> 。
build	Object	构建工程，请参考 <a href="#">表12-129</a> 。
create_time	Integer	创建时间。
update_time	Integer	修改时间。
creator	String	创建者。

参数	参数类型	描述
pipeline_ids	Array of String	流水线id。

表 12-125 source

参数	参数类型	描述
kind	String	类型。
spec	Object	对应code请参考 <a href="#">表12-126</a> ；请参考 <a href="#">表12-127</a> 。

表 12-126 code spec

参数	参数类型	描述
repo_type	String	代码仓类型，支持CodeArts、Gitee、GitHub、BitBucket、GitLab。
repo_url	String	代码仓url，如：https://github.com/example/demo.git。
repo_ref	String	代码分支或者Tag，默认是master。
repo_auth	String	授权名称，在授权列表获取。

表 12-127 spec

参数	参数类型	描述
storage	String	存储方式。
type	String	类别。
url	String	软件包/源码地址。
auth	String	认证方式。
webUrl	String	软件/源码仓库地址。
properties	Object	软件包的其他属性，只有在选择对象存储obs的时候才需要添加。请参考 <a href="#">表12-128</a> 。

表 12-128 artifact spec properties

参数	参数类型	描述
endpoint	String	obs的终端地址，比如：https://obs.region_id.external_domain_name.com。
bucket	String	软件包在obs的桶名。
key	String	obs桶中的对象，一般是软件包名，有文件夹的话要加上文件夹的路径。比如test.jar或者demo/test.jar。

表 12-129 bulid

参数	参数类型	描述
id	String	构建工程id。
parameters	Map<String,String>	请参考表12-130。

表 12-130 parameters

参数	参数类型	描述
build_cmd	String	编译命令。
dockerfile_path	String	dockerfile地址。
artifact_namespace	String	构建归档组织。
cluster_id	String	指定构建集群的id。
node_label_selector	Map<String,String>	key是标签的键，value是标签的值。

## 请求示例

无

## 响应示例

```
{
  "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
}
```

## 状态码

状态码	描述
200	操作成功
400	错误的请求
404	请求对象不存在
500	内部错误

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 12.1.5 实例

### 12.1.5.1 创建组件实例

#### 功能介绍

此API用来创建组件实例。

## 调试

您可以在[API Explorer](#)中调试该接口。

## URI

POST /v2/{project\_id}/cas/applications/{application\_id}/components/{component\_id}/instances

表 12-131 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
application_id	是	String	应用ID。获取方法，请参考 <a href="#">获取所有应用</a> 。
component_id	是	String	组件ID。获取方法，请参考 <a href="#">获取应用所有组件</a> 。

## 请求消息

表 12-132 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

表 12-133 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	应用组件实例名称。 由小写英文字母、数字、中划线（-）组成，并以小写英文字母开头，小写英文字母或者数字结尾，长度为2~63个字符。
environment_id	是	String	环境ID。
replica	是	Integer	实例副本数。

参数	是否必选	参数类型	描述
flavor_id	是	String	<p>资源规格，可查询<a href="#">获取所有支持的应用资源规格</a>接口获取系统预定义好的资源规格。</p> <p>如需自定义资源规格，格式为：CUSTOM-xxG:xxm-xxm:xxGi-xxGi。其中：</p> <ul style="list-style-type: none"> <li>• xxG表示给组件实例分配的存储的大小，为保留字段，当前未生效，可以指定为一个固定的数字。</li> <li>• xxm-xxm用于指定给组件实例分配的cpu最大核数和最小核数。</li> <li>• xxGi-xxGi用于指定给组件实例分配的最大内存和最小内存。</li> </ul> <p>例如，CUSTOM-10G:0.5m-0.25m:1.6Gi-0.8Gi，表示给组件实例分配的CPU核数最大为0.5核，最小为0.25核，分配的最大内存为1.6Gi，最小为0.8Gi。</p>
artifacts	是	Map<String, Object>	<p>组件部署件。key为组件名，对于Docker多容器场景，key为容器名称。请参考<a href="#">表12-151</a>。</p> <p>当组件的source参数指定软件包来源时，该参数可以不选，默认继承组件的软件包来源。否则，该参数是必选项。</p>
version	是	String	应用组件版本号，满足版本语义，如1.0.0。
configuration	否	Object	配置参数，如设置环境变量，部署配置，运维监控等，默认空。请参考 <a href="#">表12-134</a> 。
description	否	String	<p>描述。</p> <p>最大长度为128个字符。</p>
external_accesses	否	Array of objects	外网访问，请参考 <a href="#">表12-152</a> 。
refer_resources	是	Array of objects	部署资源，请参考 <a href="#">表12-153</a> 。

表 12-134 configuration

参数	是否必选	参数类型	描述
env	否	Array of objects	环境变量，请参考 <a href="#">表12-135</a> 。
storage	否	Array of objects	数据存储配置，请参考 <a href="#">表12-136</a> 。
strategy	否	Object	升级策略，请参考 <a href="#">表12-137</a> 。
lifecycle	否	Object	生命周期，请参考 <a href="#">表12-138</a> 。
scheduler	否	Object	调度策略，请参考 <a href="#">表12-139</a> 。
probes	否	Object	健康检查，请参考 <a href="#">表12-140</a> 。
container_spec	否	Object	容器信息，请看考 <a href="#">表12-154</a> 。

表 12-135 environment

参数	是否必选	参数类型	描述
name	是	String	环境变量名。 由字母、数字、下划线、中划线或者点号组成，不能以数字开头，且长度为1~64个字符。
value	是	String	环境变量值。

表 12-136 storage

参数	是否必选	参数类型	描述
type	是	String	存储类型，取值为： <ul style="list-style-type: none"> <li>HostPath：主机路径挂载</li> <li>EmptyDir：临时路径挂载</li> <li>ConfigMap：配置项挂载</li> <li>Secret：密钥挂载</li> <li>PersistentVolumeClaim：云存储挂载</li> </ul>
parameters	是	Object	存储的参数，请参考 <a href="#">表12-149</a> 。
mounts	是	Array of objects	挂载到容器的目录，请参考 <a href="#">表12-150</a> 。

表 12-137 strategy

参数	是否必选	参数类型	描述
upgrade	否	String	升级策略，默认为RollingUpdate。取值为Recreate和RollingUpdate，分别为替换升级和滚动升级。

表 12-138 lifecycle

参数	是否必选	参数类型	描述
entrypoint	否	Object	启动命令，请参考 <a href="#">表12-146</a> 。
post-start	否	Object	启动后处理，请参考 <a href="#">表12-147</a> 。
pre-stop	否	Object	停止前处理，请参考 <a href="#">表12-147</a> 。

表 12-139 scheduler

参数	是否必选	参数类型	描述
affinity	否	Object	亲和性，请参考 <a href="#">表12-145</a> 。
anti-affinity	否	Object	反亲和性，请参考 <a href="#">表12-145</a> 。

表 12-140 probes

参数	是否必选	参数类型	描述
livenessProbe	否	Object	组件存活探针，请参考 <a href="#">表12-141</a> 。
readinessProbe	否	Object	组件业务探针，请参考 <a href="#">表12-141</a> 。

表 12-141 probes\_object

参数	是否必选	参数类型	描述
type	是	String	取值可以为http、tcp或command。对应的检查方式分别为HTTP请求检查，TCP端口检查和执行命令检查。

参数	是否必选	参数类型	描述
parameters	是	Object	检查参数。 <ul style="list-style-type: none"> <li>• type为http, 请参考<a href="#">表12-142</a>。</li> <li>• type为tcp, 请参考<a href="#">表12-144</a>。</li> <li>• type为command, 请参考<a href="#">表12-143</a>。</li> </ul>
delay	否	integer	表示启动后多久开始探测。
timeout	否	integer	表示探测超时时间。

表 12-142 probes\_param\_http

参数	是否必选	参数类型	描述
scheme	否	String	HTTP或者HTTPS。
host	否	String	默认为POD的IP。可以指定自定义的IP。
port	否	integer	端口。
path	否	String	请求路径。

表 12-143 probes\_param\_command

参数	是否必选	参数类型	描述
command	否	Array of String	命令列表

表 12-144 probes\_param\_tcp

参数	是否必选	参数类型	描述
port	否	integer	端口

表 12-145 scheduler\_affinity

参数	是否必选	参数类型	描述
az	否	Array of String	az名称列表
node	否	Array of String	节点私有IP列表
application	否	Array of String	组件实例名列表

表 12-146 lifecycle\_entrypoint

参数	是否必选	参数类型	描述
command	否	Array of String	可执行的命令。
args	否	Array of String	运行参数。

表 12-147 lifecycle\_process

参数	是否必选	参数类型	描述
type	否	String	取值为command或者http。command为执行命令行，http为发送http请求。
parameters	否	Object	启动后处理或者停止前处理参数。请参考 <a href="#">表 12-148</a> 。

表 12-148 lifecycle\_process\_parameters

参数	是否必选	参数类型	描述
command	否	Array of String	命令参数，比如["sleep", "1"]。适用于command类型。
host	否	String	默认为POD实例的IP地址。也可以自己指定。适用于http类型。
port	否	integer	端口号。适用于http类型。
path	否	String	请求url。适用于http类型。

表 12-149 storage\_parameters

参数	是否必选	参数类型	描述
path	否	String	主机路径，适用于HostPath的存储类型。
name	否	String	配置项或者密钥的名字，适用于ConfigMap和Secret的存储类型。
defaultMode	否	Integer	挂载的权限，十进制格式，如384。
medium	否	String	适用于EmptyDir类型的存储。不传参数为默认的磁盘介质，传参为memory则开启内存存储。

表 12-150 storage\_mounts

参数	是否必选	参数类型	描述
path	是	String	挂载路径，记录挂载的磁盘路径。
subPath	否	String	记录挂载磁盘的子路径。
readOnly	是	boolean	挂载的磁盘权限为只读或者读写。

表 12-151 artifact

参数	是否必选	参数类型	描述
storage	是	String	存储方式，支持软件仓库swr、镜像仓库swr、对象存储obs、软件发布库codearts。
type	是	String	类别。 <ul style="list-style-type: none"><li>虚拟机部署支持软件包package。</li><li>容器部署支持软件包package、镜像包image。</li></ul>
url	是	String	软件包/镜像地址。对于基于虚机部署的组件，值为软件包的地址。对于基于容器部署的组件，值为镜像的地址或者“组件名:v\${index}”，后者表示使用组件来源中的源码或者软件包自动构建生成的镜像。
auth	是	String	认证方式，支持iam和none，默认为iam。
version	否	String	版本号。
properties	否	Map<String, Object>	属性。



表 12-152 external\_accesses

参数	是否必选	参数类型	描述
protocol	是	String	协议，支持http、https。
address	是	String	访问地址。
forward_port	是	integer	端口号。

表 12-153 refer\_resources

参数	是否必选	参数类型	描述
id	是	String	资源ID。 如果type为ecs的时候，该字段的值固定为“Default”。
type	是	String	基础资源：cce、ecs、as。 可选资源：rds、dcs、elb、cse等其他类型。
refer_alias	否	String	应用别名，dcs时才提供，支持“distributed_session”、“distributed_cache”、“distributed_session, distributed_cache”。默认值是“distributed_session, distributed_cache”。
parameters	否	Map<String, Object>	引用资源参数。 <b>须知</b> <ul style="list-style-type: none"><li>当type是cce的时候，该参数必填，需要指定部署组件所在的集群的命名空间，比如{"namespace": "default"}</li><li>当type为ecs的时候，该参数必填，需要指定组件部署在哪些主机上，比如{"hosts": ["04d9f887-9860-4029-91d1-7d3102903a69", "04d9f887-9860-4029-91d1-7d3102903a70"]}</li></ul>

表 12-154 container\_spec

参数	是否必选	参数类型	描述
containers	否	Array of Objects	容器信息，参考 <a href="#">表12-155</a> 。

参数	是否必选	参数类型	描述
type	否	String	工作负载类型： <ul style="list-style-type: none"><li>• deployment，无状态工作负载。</li><li>• statefulset，有状态工作负载。</li></ul>

表 12-155 container

参数	是否必选	参数类型	描述
name	否	String	容器名称。
size	否	Object	资源规格，请参考 <a href="#">表12-156</a> 。
env	否	Array of Object	环境变量，请参考 <a href="#">表12-135</a> 。
storage	否	Array of objects	数据存储配置，请参考 <a href="#">表12-136</a> 。
lifecycle	否	Object	生命周期，请参考 <a href="#">表12-138</a> 。
probes	否	Object	健康检查，请参考 <a href="#">表12-140</a> 。

表 12-156 size

参数	是否必选	参数类型	描述
id	否	String	资源规格，可查询 <a href="#">获取所有支持的应用资源规格</a> 接口获取系统预定义好的资源规格。 如需自定义资源规格，格式为：CUSTOM-xxG:xxm-xxm:xxGi-xxGi。其中： <ul style="list-style-type: none"><li>• xxG表示给组件实例分配的存储的大小，为保留字段，当前未生效，可以指定为一个固定的数字。</li><li>• xxm-xxm用于指定给组件实例分配的cpu最大核数和最小核数。</li><li>• xxGi-xxGi用于指定给组件实例分配的最大内存和最小内存。</li></ul> 例如， CUSTOM-10G:0.5m-0.25m:1.6Gi-0.8Gi，表示给组件实例分配的CPU核数最大为0.5核，最小为0.25核，分配的最大内存为1.6Gi，最小为0.8Gi。

## 响应消息

表 12-157 响应参数

参数	参数类型	描述
job_id	String	Job ID，用于查询创建任务信息。
instance_id	String	应用组件实例ID。

## 请求示例

创建名称为component-instance-name的组件实例，组件实例运行环境选择ID为6e763000-9128-4a9d-adea-34c42cc5344d的环境，组件部署件选择存储在obs下myapp桶中的demo.jar包，部署资源选择id为b6862a62-d916-11e9-bdf1-0255ac101fd9的CCE和id为8c0a45cc-626f-4d65-8257-507ee059aa9a的ELB。

```
{
  "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"
  }
]
}
```

## 响应示例

```
{
  "instance_id": "89f5baf5-efe4-4f12-9c0d-734d2af5a184",
}
```

```
"job_id": "JOB66761060-f209-407c-a093-4df6f531b9dc"  
}
```

## 状态码

状态码	描述
200	操作成功
400	错误的请求
404	请求对象不存在
500	内部错误

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

### 12.1.5.2 获取组件实例

## 功能介绍

通过此API获取组件下的所有组件实例。

## 调试

您可以在[API Explorer](#)中调试该接口。

## URI

GET /v2/{project\_id}/cas/applications/{application\_id}/components/  
{component\_id}/instances

表 12-158 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
application_id	是	String	应用ID。获取方法，请参考 <a href="#">获取所有应用</a> 。
component_id	是	String	组件ID。获取方法，请参考 <a href="#">获取应用所有组件</a> 。

表 12-159 查询参数

参数	是否必选	参数类型	描述
maxWaitTime	否	int	用户自定义整体启动时间。
limit	否	Integer	指定个数，取值[0, 100]或者1000，指定的值不在该范围内的话相当于赋值为10。默认1000，用于不分页的场景。分页的场景取值请用[0, 100]。
offset	否	Integer	指定偏移量。
order_by	否	String	排序字段，默认按创建时间排序。 排序字段支持枚举值：create_time、name、version、update_time。
order	否	String	排序方式。 <ul style="list-style-type: none"><li>• desc，降序排序，默认为desc。</li><li>• asc，升序排序。</li></ul>

## 请求消息

表 12-160 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

## 响应消息

表 12-161 响应参数

参数	参数类型	描述
count	Integer	实例总数。
instances	Array of objects	实例参数，请参考 <a href="#">表12-162</a> 。

表 12-162 instance

参数	参数类型	描述
id	String	应用组件实例ID。
application_id	String	应用ID。
application_name	String	应用名称。
component_id	String	组件ID。
component_name	String	组件名称。
name	String	应用组件实例名称。
environment_id	String	应用组件环境ID。
environment_name	String	环境名称。
platform_type	String	运行平台类型。 应用可以在不同的平台上运行，可选用的平台的类型有以下几种：cce、vmapp。
artifacts	Map<String, Object>	组件部署件。key为组件component_name，对于Docker多容器场景，key为容器名称,请参考表12-163。
creator	String	创建人。
version	String	应用组件版本号。
create_time	Integer	创建时间。
update_time	Integer	修改时间。
external_accesses	Array of objects	访问方式，请参考表12-164。
status_detail	Object	状态详情，请参考表12-165。

表 12-163 artifact

参数	参数类型	描述
storage	String	存储方式，支持软件仓库swr、镜像仓库swr、对象存储obs、软件发布库codearts。
type	String	类别。 <ul style="list-style-type: none"><li>虚拟机部署支持软件包package。</li><li>容器部署支持软件包package、镜像包image。</li></ul>

参数	参数类型	描述
url	String	软件包/镜像地址。
auth	String	认证方式，支持iam和none，默认为iam。
version	String	版本号。
properties	Map<String,String>	属性。

表 12-164 external\_accesses

参数	参数类型	描述
id	String	访问方式ID
protocol	String	协议
address	String	访问地址
forward_port	Integer	应用组件进程监听端口
type	String	类型
status	String	状态
create_time	Integer	创建时间
update_time	Integer	修改时间

表 12-165 status\_detail

参数	参数类型	描述
enterprise_project_id	String	企业项目ID
status	String	实例状态
available_replica	Integer	正常实例副本数
replica	Integer	实例副本数
fail_detail	String	失败描述
last_job_id	String	最近Job ID
last_job_statuses	String	最近Job的状态

## 请求示例

无

## 响应示例

```
{
  "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
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在。
500	内部错误。



## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

### 12.1.5.3 对组件实例的操作

## 功能介绍

通过此API操作组件实例。

## 调试

您可以在[API Explorer](#)中调试该接口。

## URI

POST /v2/{project\_id}/cas/applications/{application\_id}/components/{component\_id}/instances/{instance\_id}/action

表 12-166 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
application_id	是	String	应用ID。获取方法，请参考 <a href="#">获取所有应用</a> 。
component_id	是	String	组件ID。获取方法，请参考 <a href="#">获取应用所有组件</a> 。
instance_id	是	String	组件实例ID。获取方法，请参考 <a href="#">获取组件实例</a> 。

## 请求消息

表 12-167 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

表 12-168 请求 Body 参数

参数	是否必选	参数类型	描述
action	是	String	操作，支持start、stop、restart、scale、rollback。 <ul style="list-style-type: none"><li>• start：启动组件实例。</li><li>• stop：停止组件实例。</li><li>• restart：重启组件实例。</li><li>• scale：伸缩组件实例。</li><li>• rollback：回滚组件实例到上个版本。</li></ul>
parameters	否	Object	操作参数，action为scale和rollback时必选，请参考 <a href="#">表12-169</a> 。

表 12-169 parameters

参数	是否必选	参数类型	描述
replica	否	Integer	实例数，在scale操作时必选。
hosts	否	Array	ECS id列表，指定虚拟机伸缩的时候部署的ECS主机，在scale操作时必选。
version	否	String	版本号，在rollback操作时必选，通过 <a href="#">获取组件实例快照</a> 获取。

## 响应消息

表 12-170 响应参数

参数	参数类型	描述
job_id	String	Job ID。

## 请求示例

伸缩组件实例，修改组件实例数到4个。

```
{
  "action": "scale",
  "parameters": {
    "replica": 4
  }
}
```

## 响应示例

```
{
  "job_id": "JOB66761060-f209-407c-a093-4df6f531b9dc"
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在。
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

### 12.1.5.4 修改组件实例

## 功能介绍

通过此API修改组件实例。

## 调试

您可以在[API Explorer](#)中调试该接口。

## URI

PUT /v2/{project\_id}/cas/applications/{application\_id}/components/{component\_id}/instances/{instance\_id}

表 12-171 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
application_id	是	String	应用ID。获取方法，请参考 <a href="#">获取所有应用</a> 。
component_id	是	String	组件ID。获取方法，请参考 <a href="#">获取应用所有组件</a> 。
instance_id	是	String	组件实例ID。获取方法，请参考 <a href="#">获取组件实例</a> 。

## 请求消息

表 12-172 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

表 12-173 请求 Body 参数

参数	是否必选	参数类型	描述
version	是	String	应用组件版本号，满足版本语义，如1.0.1。
flavor_id	否	String	资源规格，可查询 <a href="#">获取所有支持的应用资源规格</a> 接口获取系统预定义好的资源规格。如果需要自定义资源规格的话，自定义资源规格的格式为：CUSTOM-xxG:xxm-xxm:xxGi-xxGi。其中： <ul style="list-style-type: none"><li>xxG表示给组件实例分配的存储的大小，为保留字段，当前未生效，可以指定为一个固定的数字。</li><li>xxm-xxm用于指定给组件实例分配的cpu最大核数和最小核数。</li><li>xxGi-xxGi用于指定给组件实例分配的最大内存和最小内存。</li></ul> 例如，CUSTOM-10G:0.5m-0.25m:1.6Gi-0.8Gi，表示给组件实例分配的cpu核数最大为0.5核，最小为0.25核，分配的最大内存为1.6Gi，最小为0.8Gi。
artifacts	否	Map<String, Object>	组件部署件，key为组件名；对于Docker多容器场景，key为容器名称。请参考 <a href="#">表12-174</a> 。
configuration	否	Object	配置参数，如设置环境变量、部署配置、运维监控等，默认为空。请参考 <a href="#">创建组件实例的表12-134</a> 。
description	否	String	描述。 最大长度为128个字符。

参数	是否必选	参数类型	描述
external_accesses	否	Array of objects	访问方式，请参考 <a href="#">表12-175</a> 。
refer_resources	否	Array of objects	部署资源，请参考 <a href="#">表12-176</a> 。

表 12-174 artifact

参数	是否必选	参数类型	描述
storage	是	String	存储方式，支持软件仓库swr、镜像仓库swr、对象存储obs、软件发布库codearts。
type	是	String	类别。 <ul style="list-style-type: none"><li>虚拟机部署支持软件包package。</li><li>容器部署支持软件包package、镜像包image。</li></ul>
url	是	String	软件包/镜像地址。对于基于虚拟机部署的组件，值为软件包的地址。对于基于容器部署的组件，值为镜像的地址或者“组件名:v\${index}”，后者表示使用组件来源中的源码或者软件包自动构建生成的镜像。
auth	是	String	认证方式，支持iam和none，默认为iam。
version	否	String	版本号。
properties	否	Map<String,String>	属性。

表 12-175 external\_accesses

参数	是否必选	参数类型	描述
id	否	String	访问方式ID。
protocol	是	String	协议，支持http和https。
address	是	String	访问地址，例如：www.example.com。
forward_port	是	Integer	应用组件进程监听端口。
type	否	String	类型。
status	否	String	状态。

参数	是否必选	参数类型	描述
create_time	否	Integer	创建时间。
update_time	否	Integer	修改时间。

表 12-176 refer\_resources

参数	是否必选	参数类型	描述
id	是	String	资源ID。
type	是	String	资源类型，含rds、dcs等类型。
refer_alias	否	String	应用别名，dcs时才提供，支持“distributed_session”、“distributed_cache”、“distributed_session, distributed_cache”，默认值是“distributed_session, distributed_cache”。
parameters	否	Map<String, Object>	引用资源参数。

## 响应消息

表 12-177 响应参数

参数	参数类型	描述
job_id	String	Job ID，用于查询创建任务信息。

## 请求示例

修改应用组件实例，修改组件部署件为存储在swr镜像仓库swr.roma-dev-1.roma.development.com/ns1/examples:v2下名称为test-micro的镜像包，部署资源修改为id为523498f1-36c4-11eb-ae36-0255ac1000c2的CCE和专业版微服务引擎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"  
}
```

## 响应示例

```
{  
  "job_id": "JOB66761060-f209-407c-a093-4df6f531b9dc"  
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在
500	内部错误

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

### 12.1.5.5 删除组件实例

## 功能介绍

通过此API删除组件实例。

## 调试

您可以在[API Explorer](#)中调试该接口。

## URI

DELETE /v2/{project\_id}/cas/applications/{application\_id}/components/  
{component\_id}/instances/{instance\_id}

表 12-178 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
application_id	是	String	应用ID。获取方法，请参考 <a href="#">获取所有应用</a> 。
component_id	是	String	组件ID。获取方法，请参考 <a href="#">获取应用所有组件</a> 。
instance_id	是	String	组件实例ID。获取方法，请参考 <a href="#">获取组件实例</a> 。

表 12-179 查询参数

参数	是否必选	参数类型	描述
force	否	boolean	是否强制删除。

## 请求消息

表 12-180 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

## 响应消息

表 12-181 响应参数

参数	参数类型	描述
job_id	String	Job ID



## 请求示例

无

## 响应示例

```
{
  "job_id": "JOB66761060-f209-407c-a093-4df6f531b9dc"
}
```

## 状态码

状态码	描述
200	操作成功
400	错误的请求
404	请求对象不存在
500	内部错误

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

### 12.1.5.6 根据实例 ID 获取实例详细信息

## 功能介绍

此API通过实例ID获取实例详细信息。

## 调试

您可以在[API Explorer](#)中调试该接口。

## URI

GET /v2/{project\_id}/cas/applications/{application\_id}/components/{component\_id}/instances/{instance\_id}

表 12-182 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
application_id	是	String	应用ID。获取方法，请参考 <a href="#">获取所有应用</a> 。

参数	是否必选	参数类型	描述
component_id	是	String	组件ID。获取方法，请参考 <a href="#">获取应用所有组件</a> 。
instance_id	是	String	组件实例ID。获取方法，请参考 <a href="#">获取组件实例</a> 。

## 请求消息

表 12-183 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

## 响应消息

表 12-184 响应参数

参数	参数类型	描述
id	String	应用组件实例ID。
name	String	应用组件实例名称。
environment_id	String	应用组件环境ID。
platform_type	String	运行平台类型。 应用可以在不同的平台上运行，可选用的平台的类型有以下几种：cce、vmapp。
description	String	实例描述。
flavor_id	String	资源规格。
artifacts	Map<String, Object>	组件部署件，key为组件component_name；对于Docker多容器场景，key为容器名称。请参考 <a href="#">表12-185</a> 。
version	String	应用组件版本号。
configuration	Map<String, Object>	应用组件配置，如环境变量。

参数	参数类型	描述
creator	String	创建人。
create_time	Integer	创建时间。
update_time	Integer	修改时间。
external_accesses	Array of objects	访问方式，请参考表12-186。
refer_resources	Array of objects	部署资源，请参考表12-187。
status_detail	Object	状态详情，请参考表12-188。

表 12-185 artifact

参数	参数类型	描述
storage	String	存储方式，支持软件仓库swr、镜像仓库swr、对象存储obs、软件发布库codearts。
type	String	类别。 <ul style="list-style-type: none"><li>虚拟机部署支持软件包package。</li><li>容器部署支持软件包package、镜像包image。</li></ul>
url	String	软件包/镜像地址。
auth	String	认证方式，支持iam和none，默认为iam。
version	String	版本号。
properties	Map<String,String>	属性。

表 12-186 external\_accesses

参数	参数类型	描述
id	String	访问方式ID
protocol	String	协议
address	String	访问地址
forward_port	Integer	应用组件进程监听端口
type	String	类型
status	String	状态
create_time	Integer	创建时间

参数	参数类型	描述
update_time	Integer	修改时间

表 12-187 refer\_resources

参数	参数类型	描述
id	String	资源ID。
type	String	基础资源：cce、ecs、as。 可选资源：rds、dcs、elb、cse等其他类型。
refer_alias	String	应用别名，可选资源选择dcs时才提供，支持“distributed_session”、“distributed_cache”、“distributed_session, distributed_cache”。 默认值是“distributed_session, distributed_cache”。
parameters	Map<String, Object>	引用资源参数。

表 12-188 status\_detail

参数	参数类型	描述
enterprise_project_id	String	企业项目ID。
status	String	实例状态。 <ul style="list-style-type: none"><li>● INITIALIZING：创建中。</li><li>● UPGRADING：升级中。</li><li>● FAILED：失败。</li><li>● RUNNING：运行中。</li><li>● DELETING：删除中。</li><li>● STARTING：启动中。</li><li>● STOPPING：停止中。</li><li>● STOPPED：已停止。</li><li>● RESTARTING：重启中。</li><li>● PENDING：未就绪。</li><li>● PARTIALLY_FAILED：异常。</li></ul>
available_replicas	Integer	正常实例副本数。

参数	参数类型	描述
replica	Integer	实例副本数。
fail_detail	String	失败描述。
last_job_id	String	最近Job ID。
last_job_statuses	String	最近Job的状态。

## 请求示例

无

## 响应示例

```
{
  "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"
}
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在。
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

### 12.1.5.7 获取组件实例快照

#### 功能介绍

通过此API获取应用组件实例的快照信息。

实例快照是指升级后，之前版本的实例信息，可以用于回滚。

#### 调试

您可以在[API Explorer](#)中调试该接口。

## URI

GET /v2/{project\_id}/cas/applications/{application\_id}/components/{component\_id}/instances/{instance\_id}/snapshots

表 12-189 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
application_id	是	String	应用ID。获取方法，请参考 <a href="#">获取所有应用</a> 。
component_id	是	String	组件ID。获取方法，请参考 <a href="#">获取应用所有组件</a> 。
instance_id	是	String	组件实例ID。获取方法，请参考 <a href="#">获取组件实例</a> 。

表 12-190 查询参数

参数	是否必选	参数类型	描述
limit	否	Integer	指定个数，取值[0, 100]或者1000，指定的值不在该范围内的话相当于赋值为10。默认1000，用于不分页的场景。分页的场景取值请用[0, 100]。
offset	否	Integer	指定偏移量。
order_by	否	String	排序字段，默认按创建时间排序。
order	否	String	排序方式。 <ul style="list-style-type: none"><li>• desc，降序排序，默认为desc。</li><li>• asc，升序排序。</li></ul>

## 请求消息

表 12-191 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

## 响应消息

表 12-192 响应参数

参数	参数类型	描述
count	Integer	快照总数。
snapshots	Array of Objects	快照参数，请参考 <a href="#">表12-193</a> 。

表 12-193 snapshot

参数	参数类型	描述
create_time	Integer	创建时间
description	String	描述
instance_id	String	应用组件实例ID
version	String	版本号

## 请求示例

无

## 响应示例

```
{
  "snapshots": [
    {
      "version": "1.0.0",
      "description": "{\"action\": \"New Version: 1.0.0\"}",
      "instance_id": "11eddb33-140b-4e51-b1e2-6ec265373ca3",
      "create_time": 1610331830841
    }
  ],
  "count": 1
}
```



## 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在。
500	内部错误。

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 12.1.6 部署任务

### 12.1.6.1 获取部署任务详细信息

#### 功能介绍

通过此API获取部署任务详细信息。

#### 调试

您可以在[API Explorer](#)中调试该接口。

#### URI

GET /v2/{project\_id}/cas/jobs/{job\_id}

表 12-194 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户项目ID。获取方法，请参考 <a href="#">获取项目ID</a> 。
job_id	是	String	部署任务ID，为 <a href="#">创建组件实例</a> 接口响应参数的job_id。

表 12-195 查询参数

参数	是否必选	参数类型	描述
instance_id	否	String	应用组件实例ID。获取方法，请参考 <a href="#">获取组件实例</a> 。
limit	否	Integer	指定个数。
offset	否	Integer	指定偏移量。
desc	否	String	是否降序： true=desc, false=asc。

## 请求消息

表 12-196 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

## 响应消息

表 12-197 响应参数

参数	参数类型	描述
task_count	Integer	部署任务数量。
job	Object	构建工程参数，请参考 <a href="#">表12-198</a> 。
tasks	Array<Object>	部署任务参数，请参考 <a href="#">表12-199</a> 。

表 12-198 job

参数	参数类型	描述
CREATED_BY	String	创建者。

参数	参数类型	描述
EXECUTION_STATUS	String	执行状态。 <ul style="list-style-type: none"> <li>• SUBMITTED: 已提交但尚未处理。</li> <li>• SCHEDULED: 计划在未来执行的。</li> <li>• RUNNING: 当前正在处理。</li> <li>• FAILED: 处于失败状态。</li> <li>• SUCCEEDED: 处于成功状态。</li> <li>• CANCELLED: 已取消。</li> <li>• PARTIALLY_FAILED: 部分失败。</li> <li>• ROLLBACKING: 正在回滚。</li> <li>• ROLLBACKED: 已回滚。</li> <li>• UNKNOWN: 未知状态。</li> </ul>
JOB_DESC	String	工作描述。
JOB_ID	String	工作ID。
JOB_NAME	String	工作名称。
JOB_TYPE	String	类别: <ul style="list-style-type: none"> <li>• PROVISION: 提供服务任务。</li> <li>• DEPROVISION: 取消提交服务任务。</li> <li>• DAILY_OPERATION: 日常操作任务。</li> <li>• SYSTEM_INNER: 系统内部任务。</li> <li>• UNKNOWN: 未知任务。</li> </ul>
ORDER_ID	String	排序ID。
PROJECT_ID	String	创建租户Project ID。
SERVICE_INSTANCE_ID	String	实例ID。

表 12-199 task

参数	参数类型	描述
CREATED_AT	String	创建时间。
LAST_HEALTH_CHECK	String	健康检查时间。
MESSAGES	String	消息。
OWNER_ID	String	创建用户ID。
TASK_ID	String	任务ID。

参数	参数类型	描述
TASK_INDEX	Integer	任务序号。
TASK_NAME	String	任务名称。
TASK_STATUS	String	任务状态： <ul style="list-style-type: none"><li>• SUBMITTED: 已提交但尚未处理。</li><li>• SCHEDULED: 计划在未来执行的。</li><li>• RUNNING: 当前正在处理。</li><li>• FAILED: 处于失败状态。</li><li>• SUCCEEDED: 处于成功状态。</li><li>• CANCELLED: 已取消。</li><li>• PARTIALLY_FAILED: 部分失败。</li><li>• SKIPPED: 已跳过。</li><li>• ROLLBACKING: 正在回滚。</li><li>• ROLLBACKED: 已回滚。</li><li>• UNKNOWN: 未知状态。</li></ul>
TASK_TYPE	String	任务类型： <ul style="list-style-type: none"><li>• kube: K8s任务。</li><li>• restful: RESTful HTTP请求任务。</li><li>• sleep: sleep等待任务。</li><li>• echo: 输出任务。</li><li>• context: 上下文任务。</li><li>• sub-job: 子任务。</li></ul>

### 请求示例

无

### 响应示例

```
{
  "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
}

```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。
404	请求对象不存在

状态码	描述
500	内部错误

## 错误码

错误码格式为：SVCSTG.00100.[Error\_ID]，例如：SVCSTG.00100400。错误码说明请参考[ServiceStage错误码](#)。

## 12.2 构建 API

### 12.2.1 根据组件名称获取构建任务列表

#### 功能介绍

此API用于根据组件名称获取构建任务列表。

#### URI

GET /assembling/v1/jobs?  
page={page}&per\_page={per\_page}&job\_name={job\_name}

参数说明见[表12-200](#)。

表 12-200 路径参数

参数	是否必选	参数类型	描述
page	是	Integer	分页索引，表示从第几页开始。
per_page	是	Integer	分页大小，表示一页包含几条记录。
job_name	否	String	构建任务名称，和用户定义的组件名称相同。获取方法，请参考 <a href="#">获取应用所有组件</a> 。

#### 请求消息

表 12-201 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

## 响应消息

表 12-202 响应参数

参数	参数类型	描述
list	Array of object	构建阶段结构体，请参考 <a href="#">表12-203</a> 。
total	Integer	构建阶段数量。
page	Integer	分页大小，表示一页包含几条记录。
per_page	Integer	分页索引，表示从第几页开始。

表 12-203 list

参数	参数类型	描述
job_id	String	构建任务标识。
job_name	String	构建任务名称，和用户定义的组件名称相同。

## 请求示例

无

## 响应示例

```
{
  "total": 12,
  "per_page": 6,
  "page": 1,
  "list": [
    {
      "job_id": "swpfk3wtzdjnllyely4rf40k52zyuq5s4lha9ch8f",
      "job_name": "test1"
    }
  ]
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。

## 错误码

错误码格式为：SVCSTG.CPE.[Error\_ID]，例如：SVCSTG.CPE.5004401。错误码说明请参考[ServiceStage错误码](#)。

## 12.2.2 根据构建任务 ID 获取构建任务实例列表

### 功能介绍

此API用于根据构建任务ID获取构建任务实例列表。

### URI

GET /assembling/v1/jobs/{job\_id}/builds?name={job\_name}

参数说明见[表12-204](#)。

表 12-204 路径参数

参数	是否必选	参数类型	描述
job_id	是	String	构建任务ID。获取方法，请参考 <a href="#">根据组件名称获取构建任务列表</a> 。
job_name	是	String	构建任务名称。获取方法，请参考 <a href="#">根据组件名称获取构建任务列表</a> 。

### 请求消息

表 12-205 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。



参数	是否必选	参数类型	描述
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

## 响应消息

表 12-206 响应参数

参数	参数类型	描述
list	Array of object	构建阶段结构体，请参考 <a href="#">表12-207</a> 。
total	Integer	构建阶段数量。
page	Integer	分页开始的位置。
per_page	Integer	分页大小。
finished	Integer	完成的构建实例个数。
successful	Integer	成功的构建实例个数。

表 12-207 list

参数	参数类型	描述
build_id	String	构建任务实例ID。
build_state	String	构建实例状态。 <ul style="list-style-type: none"><li>• Cancel: 取消</li><li>• Failed: 失败</li><li>• Successful: 成功</li><li>• Timeout: 超时。</li></ul>
duration	Integer	构建耗时，单位为秒。

## 请求示例

无

## 响应示例

```
{  
  "total": 1,  
  "per_page": 20,  
}
```

```
"page": 1,  
"finished": 1,  
"successful": 1,  
"list": [  
  {  
    "build_id": "0f7fqckzmjcfi0f7yxhx91l2rwrifyzd2poac90b5",  
    "build_state": "Successful",  
    "duration": 1181  
  }  
]
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。

## 错误码

错误码格式为：SVCSTG.CPE.[Error\_ID]，例如：SVCSTG.CPE.5004401。错误码说明请参考[ServiceStage错误码](#)。

## 12.2.3 根据构建任务实例 ID 获取构建阶段列表

### 功能介绍

此API用于根据构建任务实例ID获取构建阶段列表。

### URI

GET /assembling/v1/builds/{build\_id}/phases

参数说明见[表12-208](#)。

表 12-208 路径参数

参数	是否必选	参数类型	描述
build_id	是	String	构建任务实例ID。获取方法，请参考 <a href="#">根据构建任务ID获取构建任务实例列表</a> 。

## 请求消息

表 12-209 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

## 响应消息

表 12-210 响应参数

参数	参数类型	描述
list	Array of object	构建阶段结构体，请参考 <a href="#">表12-211</a> 。
total	Integer	构建阶段数量。

表 12-211 list

参数	参数类型	描述
id	String	构建阶段标识。
name	String	构建阶段名称。 <ul style="list-style-type: none"><li>• Init：初始化阶段</li><li>• Download：下载阶段</li><li>• PreProcess：预处理阶段</li><li>• Build：构建阶段</li><li>• Artifact：归档阶段</li><li>• PostProcess：后置处理阶段</li></ul>

参数	参数类型	描述
state	String	构建阶段状态。 <ul style="list-style-type: none"><li>• NotStarted: 未开始</li><li>• Queuing: 排队中</li><li>• Building: 构建中</li><li>• Pushing: 正在推送包</li><li>• Successful: 成功</li><li>• Failed: 失败</li><li>• Timeout: 超时</li><li>• Cancel: 取消</li><li>• Deleted: 已删除</li></ul>

## 请求示例

无

## 响应示例

```
{
  "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"
    }
  ]
}
```

## 状态码

状态码	描述
200	操作成功。
400	错误的请求。

## 错误码

错误码格式为：SVCSTG.CPE.[Error\_ID]，例如：SVCSTG.CPE.5004401。错误码说明请参考[ServiceStage错误码](#)。

## 12.2.4 根据构建任务实例 ID 获取构建日志

### 功能介绍

此API用于根据构建任务实例ID获取构建任务的日志。

### URI

GET /assembling/v1/builds/{build\_id}/logs?from={from}

参数说明见表12-212。

表 12-212 路径参数

参数	是否必选	参数类型	描述
build_id	是	String	构建任务实例ID。获取方法，请参考 <a href="#">根据构建任务ID获取构建任务实例列表</a> 。
from	是	Integer	获取日志的开始位置，例如从0开始。

### 请求消息

表 12-213 请求 Header 参数

参数	是否必选	参数类型	描述
Content-Type	是	String	消息体的类型（格式），默认取值为“application/json;charset=utf8”。
X-Auth-Token	是	String	调用接口的认证方式分为Token和AK/SK两种，如果您使用的Token方式，此参数为必填，请填写Token的值。Token获取方式，请参考 <a href="#">获取用户Token</a> 。

## 响应消息

表 12-214 响应参数

参数	参数类型	描述
buildlog	String	返回的构建日志。
next	Integer	返回的日志字符串结束位置。
phase_states	String	构建阶段状态。 <ul style="list-style-type: none"><li>• NotStarted: 未开始</li><li>• Queuing: 排队中</li><li>• Building: 构建中</li><li>• Pushing: 正在推送包</li><li>• Successful: 成功</li><li>• Failed: 失败</li><li>• Timeout: 超时</li><li>• Cancel: 取消</li><li>• Deleted: 已删除</li></ul>
phases	String	构建阶段名称。 <ul style="list-style-type: none"><li>• Init: 初始化阶段</li><li>• Download: 下载阶段</li><li>• PreProcess: 预处理阶段</li><li>• Build: 构建阶段</li><li>• Artifact: 归档阶段</li><li>• PostProcess: 后置处理阶段</li></ul>

## 请求示例

无

## 响应示例

```
{  
  "buildlog": "Starting...",  
  "next": 25524,  
  "phase_states": "Successful,Successful,Successful,Successful,Successful,Successful",  
  "phases": "Init,Download,PreProcess,Build,Artifact,PostProcess"  
}
```

## 状态码

状态码	描述
200	操作成功。

状态码	描述
400	错误的请求。

## 错误码


错误码格式为：SVCSTG.CPE.[Error\_ID]，例如：SVCSTG.CPE.5004401。错误码说明请参考[ServiceStage错误码](#)。

## 12.3 查询配置

### 功能介绍

从配置管理中心查询配置信息。

### 调用方法

- 微服务引擎专业版，请参考[如何调用API](#)。
- 微服务引擎专享版，请参考如下方法调用：
  - a. 登录ServiceStage控制台，选择“微服务引擎 > 引擎实例”。
  - b. 选择“微服务引擎”下拉列表中待调用接口的微服务引擎。
  - c. 在“服务发现 & 配置”区域，查看或单击复制“配置中心地址”。
  - d. 参考[如何调用API](#)访问该接口，在[请求URI](#)中，替换{Endpoint}为已获取到的配置中心地址。

### URI

GET /v3/{project\_id}/configuration/items

表 12-215 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	<ul style="list-style-type: none"><li>• 微服务引擎专业版，请填租户项目id，长度为1~64。也可以填固定值 default。</li><li>• 微服务引擎专享版，请填固定值： default。</li></ul>

表 12-216 Query 参数

参数	是否必选	参数类型	描述
dimensionsInfo	是	String	服务信息，为服务名、应用、版本的拼接（服务名、应用必填，版本选填），拼接格式为（请注意“@”和“#”添加到http url时需转义，“@”转义为“%40”，“#”转义为“%23”）： {serviceName}@{appId}#{version} 或 {serviceName}@{appId} 不能包含\$%^&+/\特殊字符和空格。
keyFilter	否	String	key的筛选条件，支持正则表达式。涉及特殊字符的正则表达式需要自行转义。
revision	否	String	配置项版本。用于判断本地缓存配置和配置管理中心的配置差异。 <ul style="list-style-type: none"> <li>首次传值0。</li> <li>若传入的revision值与当前配置的revision一致，则返回304状态码，无body体返回；若不一致，则返回当前配置和revision信息。</li> <li>下次请求若需要传入revision，则该值为当前返回body体中的revision中的version的值。</li> </ul>

## 请求消息

表 12-217 请求 Header 参数

参数	是否必选	参数类型	描述
x-environment	否	String	指定查询配置的environment，取值包括：development、testing、acceptance、production。其他值均被作为空值处理。
Authorization	是	String	若微服务引擎专享版开启了安全认证，此参数必填。否则，无此参数。 开启了安全认证的微服务引擎专享版Token，格式为： Authorization:Bearer {Token} Token获取方法，请参考 <a href="#">获取微服务引擎专享版用户Token</a> 。



## 响应消息

若传入的revision值与当前配置的revision一致，则无响应消息。

响应参数说明请参见[表12-218](#)。

表 12-218 响应参数

参数	参数类型	描述
{dimension} (参数名称和 参数个数不固 定)	Map<String, String>	配置内容
revision	<a href="#">表12-219</a>	revision

表 12-219 revision

参数	参数类型	描述
version	String	配置版本

## 请求示例

```
GET /v3/default/configuration/items?dimensionsInfo=service%40app%231.0.0
```

## 响应示例

```
{  
  "service@app": {  
    "k1": "v1"  
  },  
  "service@app#1.0.0": {  
    "k2": "v2"  
  }  
}
```

## 状态码

状态码请参见[状态码](#)。

## 错误码


错误码请参见[CSE错误码](#)。

## 12.4 创建服务间依赖关系

### 功能介绍

创建服务间的依赖关系，consumer的version必须是确定的版本，consumer的serviceName不能为\*，consumer必须是已存在的服务。provider可以是还未创建的。

## 调用方法

1. 登录ServiceStage控制台，选择“微服务引擎 > 引擎列表”。
2. 在页面上方“微服务引擎”下拉列表，选择待调用接口的微服务引擎。
3. 在“服务发现 & 配置”区域，查看或单击复制“服务注册发现地址”。
4. 参考[如何调用API](#)调用该接口，在[请求URI](#)中，替换{Endpoint}为已获取到的服务注册发现地址。

## URI

PUT /v4/{project\_id}/registry/dependencies

表 12-220 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	请填固定值：default。

## 请求消息

表 12-221 请求 Header 参数

参数	是否必选	参数类型	描述
Authorization	是	String	若微服务引擎专享版开启了安全认证，此参数必填。否则，无此参数。 开启了安全认证的微服务引擎专享版Token，格式为： Authorization:Bearer {Token} Token获取方法，请参考 <a href="#">获取微服务引擎专享版用户Token</a> 。

表 12-222 请求 Body 参数

参数	是否必选	参数类型	描述
dependencies	是	Array of objects	规则的集合，请参考 <a href="#">表12-223</a> 。列表长度为1-100。

表 12-223 MicroServiceDependency

参数	是否必选	参数类型	描述
consumer	是	Object	消费端微服务，具体参考 <a href="#">DependencyMicroService</a> 。
providers	否	Array of objects	服务端微服务，具体参考 <a href="#">DependencyMicroService</a> 。

## 响应消息

无

## 请求示例

```
{
  "dependencies": [
    {
      "consumer": {
        "environment": "",
        "appld": "default",
        "serviceName": "test1",
        "version": "1.0.0"
      },
      "providers": [
        {
          "environment": "",
          "appld": "default",
          "serviceName": "test",
          "version": "1.0.0"
        }
      ]
    }
  ]
}
```

## 响应示例

无

## 状态码

状态码请参见[状态码](#)。

## 错误码

错误码请参见[CSE错误码](#)。

# 12.5 创建上报服务指标

## 功能介绍

创建上报到仪表盘的服务指标。

## 调用方法

请参见[如何调用API](#)。

## URI

POST /v2/{project\_id}/csemonitor/metric

表 12-224 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	租户子项目的唯一标识。字符长度为1~64。

## 请求参数

表 12-225 请求 Header 参数

参数	是否必选	参数类型	描述
x-domain-name	是	String	租户账号名称。
X-Auth-Token	是	String	用户Token。

表 12-226 请求 Body 参数

参数	是否必选	参数类型	描述
name	否	String	名称
appld	否	String	应用ID
serviceId	否	String	服务ID
version	否	String	版本
instanceId	否	String	实例ID
instance	否	String	实例信息
thread	否	Integer	线程名称
cpu	否	Double	CPU信息
memory	否	Map<String, Long>	内存信息
interfaces	否	Array of <a href="#">FunctionMetricInfo</a> objects	监控项信息列表

表 12-227 FunctionMetricInfo

参数	是否必选	参数类型	描述
functionName	否	String	监控项名称
functionType	否	String	监控项类型
desc	否	String	监控项描述
time	否	Long	时间
appld	否	String	应用ID
version	否	String	版本
qps	否	Double	吞吐量
latency	否	Double	时延
rate	否	Double	占比
failureRate	否	Double	失败占比
shortCircuited	否	Integer	熔断次数
semaphoreRejected	否	Integer	信号量拒绝次数
threadPoolRejected	否	Integer	线程池拒绝次数
countTimeout	否	Integer	超时时间
l995	否	Double	99.5%的请求低于此值。
l99	否	Double	99%的请求低于此值。
l90	否	Double	90%的请求低于此值。
l75	否	Double	75%的请求低于此值。
l50	否	Double	50%的请求低于此值。
l25	否	Double	25%的请求低于此值。
l5	否	Double	5%的请求低于此值。

## 响应参数

状态码： 400

表 12-228 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。

参数	参数类型	描述
errorMessage	String	错误信息。
detail	String	详细定位信息。

状态码： 500

表 12-229 响应 Body 参数

参数	参数类型	描述
errorCode	String	错误代码。
errorMessage	String	错误信息。
detail	String	详细定位信息。

## 请求示例

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
    }
  ]
}
```

## 响应示例

状态码： 400

```
{  
  "errorCode": "string",  
  "errorMessage": "string",  
  "detail": "string"  
}
```

**状态码: 500**

```
{  
  "errorCode": "string",  
  "errorMessage": "string",  
  "detail": "string"  
}
```

## 状态码

状态码	描述
200	数据上报成功。
400	错误的请求。
500	服务内部错误

## 错误码

请参见[CSE错误码](#)。