应用管理与运维平台

快速入门

文档版本 01

发布日期 2024-09-25

华为云计算技术有限公司

商标声明


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

1 使用 ServiceStage 托管微服务应用

应用管理与运维平台（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。

ServiceStage提供了环境管理功能，把相同VPC下的计算资源（如云容器引擎CCE、弹性云服务器ECS等）、网络资源（如弹性负载均衡ELB、弹性IP等）和中间件（如分布式缓存DCS、云数据库RDS、微服务引擎CSE等）组合为一个环境，部署应用时选择环境会自动加载包含的资源。

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。

组件是组成应用的某个业务特性的实现，以代码或者软件包为载体，可独立部署运行。

针对组件，ServiceStage提供了启停、升级、回退、伸缩、查看日志、查看事件、设置访问方式、设置阈值告警等运维操作。

本例基于ServiceComb（SpringMVC）框架，基于源码快速创建微服务应用，供您体验ServiceStage的功能。

说明

ServiceStage基于GitHub提供了一些不同语言的demo。体验特定语言demo在ServiceStage中的源码部署功能，请参考**如何体验ServiceStage的源码部署功能?**

使用流程

使用ServiceStage的托管微服务应用的流程及说明，请参考**图1-1**。

图 1-1 ServiceStage 使用流程

前提条件

1. 已**注册华为账号并开通华为云**。
2. 当前登录账号拥有使用ServiceStage服务的权限。账号权限授权与绑定，请参考**创建用户并授权使用ServiceStage**。
3. 创建一个虚拟私有云VPC，请参考**创建虚拟私有云和子网**。
4. 创建一个CCE集群，请参考**购买集群**。
   - CCE集群所在VPC为**3**所创建的VPC。
   - 集群中至少包含一个ECS节点（为方便后续步骤的操作，节点规格最好选择4vCPUs、8GB内存）并且绑定弹性IP。为CCE集群添加节点，请参考**创建节点**。
   - CCE集群不能被其他环境绑定。

注册 GitHub 账号并复刻 Demo 源码

步骤1 **注册GitHub账号**。

步骤2 **登录GitHub**。

步骤3 导航到**Demo源码仓库**。

步骤4 复刻Demo源码仓库到个人账号下，请参考**复刻仓库**。

----结束

创建仓库授权

步骤1 使用已注册的华为云账号**登录ServiceStage控制台**。

步骤2 在区域列表选择您准备使用ServiceStage的区域（例如：华北-北京四）。

图 1-2 登录 ServiceStage 控制台

步骤3 选择“持续交付 > 仓库授权”。

步骤4 单击“新建授权”，进入创建仓库授权页面。

步骤5 “授权名称”保持默认。

步骤6 设置“仓库授权”。

1. 选择“GitHub”仓库。
2. “授权方式”选择“OAuth”。
3. 单击“使用OAuth授权”。
4. 阅读了解服务声明后，勾选“我已知晓本服务的源码构建功能收集上述信息，并同意授权对其的收集、使用行为。”。

5. 单击“确定”。
6. 输入您的GitHub账号及密码登录GitHub完成身份认证，等待授权完成。

步骤7 单击“确认”，在仓库授权列表可以查看到已经创建完成的授权。

图 1-3 仓库授权

----结束

创建组织

步骤1 选择“部署源管理 > 组织管理”。

步骤2 单击“创建组织”，在弹出的页面中填写“组织名称”（例如：ss-org）。

步骤3 单击“确定”。

----结束

创建微服务引擎专享版

须知

如果引擎创建账号的权限为创建引擎的最小权限，如**CSE细粒度权限依赖说明**中的“cse:engine:create”所示。则需要由主账号为其预置VPC默认安全组cse-engine-default-sg，请参考**创建微服务引擎**。

步骤1 选择“微服务引擎 > 引擎实例”。

步骤2 单击“购买微服务引擎”，参考下表设置引擎配置参数。

参数	参数说明
计费模式	选择“按需计费”。
企业项目	默认选择default。 企业项目是一种云资源管理方式，企业项目管理服务提供统一的云资源按项目管理，以及项目内的资源管理、成员管理。 已**开通企业项目**后可以使用。
规格	保持默认。
引擎类型	选择“集群”。
微服务引擎名称	设置微服务引擎名称（例如：cse-test）。
可用区	为微服务引擎选择1个可用区。

参数	参数说明
网络	为微服务引擎选择**前提条件**中已经创建的虚拟私有云及其子网，可以为您的引擎构建隔离的、自主配置和管理的虚拟网络环境。
安全认证	选择“关闭安全认证”。

步骤3 单击“立即购买”。

步骤4 确认引擎配置无误后，单击“提交”。

微服务引擎创建完成，大约需要31分钟。微服务引擎创建成功后，“状态”为“可用”。

图 1-4 创建微服务引擎

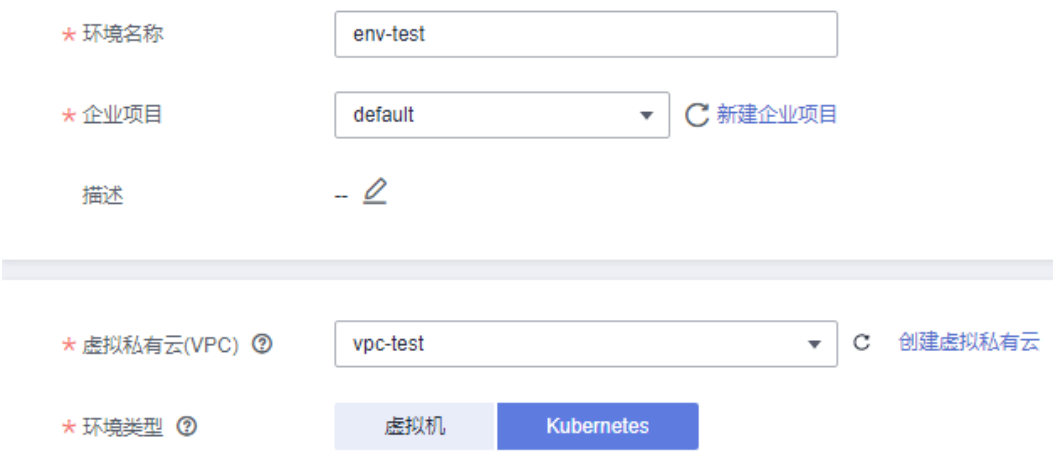
----结束

创建环境

步骤1 选择“环境管理 > 创建环境”，参照下表设置环境信息。

参数	参数说明
环境名称	输入环境名称（例如：env-test）。
企业项目	默认选择default。 企业项目是一种云资源管理方式，企业项目管理服务提供统一的云资源按项目管理，以及项目内的资源管理、成员管理。 已**开通企业项目**后可以使用。
虚拟私有云(VPC)	选择**前提条件**中已准备好的虚拟私有云VPC。 **说明** 环境创建完成后，不支持修改VPC。
环境类型	选择Kubernetes。

图 1-5 创建环境



步骤2 单击“立即创建”，进入环境详情页面。

步骤3 在“资源”下左侧列表，选择“计算”资源类型下的“云容器引擎 CCE”，单击“立即绑定”。

步骤4 在弹出的对话框中，选择**前提条件**中已创建的CCE集群资源，单击“确定”。

步骤5 在“资源”下左侧列表，选择“中间件”资源类型下的“ServiceComb引擎”，单击“纳管资源”。

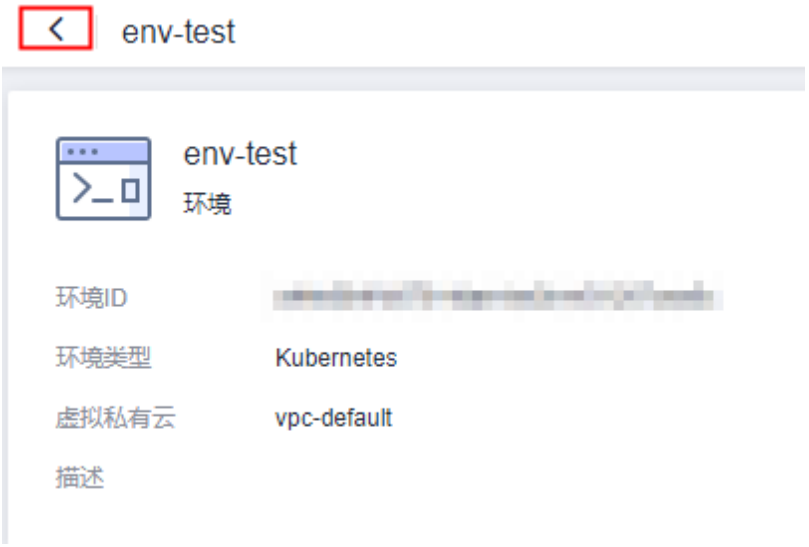
步骤6 在弹出的对话框中，选择**创建微服务引擎专享版**时创建的ServiceComb引擎资源，单击“确定”。

----结束

创建应用

步骤1 单击左上角 < ，返回“环境管理”页面。

图 1-6 返回“环境管理”页面



步骤2 选择“应用管理 > 创建应用”，参考下表设置应用基本信息，其余参数保持默认。

参数	参数说明
应用名称	输入应用名称（例如：servicecomb）。
企业项目	默认选择default。 企业项目是一种云资源管理方式，企业项目管理服务提供统一的云资源按项目管理，以及项目内的资源管理、成员管理。 已**开通企业项目**后可以使用。

步骤3 单击“确定”，完成应用创建。

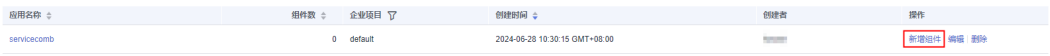
图 1-7 创建应用

----结束

创建并部署组件

步骤1 单击**创建应用**时创建的应用名称（例如：servicecomb）所在“操作”列“新增组件”，进入“创建组件”页面。

图 1-8 进入“创建组件”页面



步骤2 在“基本信息”区域，参考下表设置必填组件基本信息，其余参数保持默认。

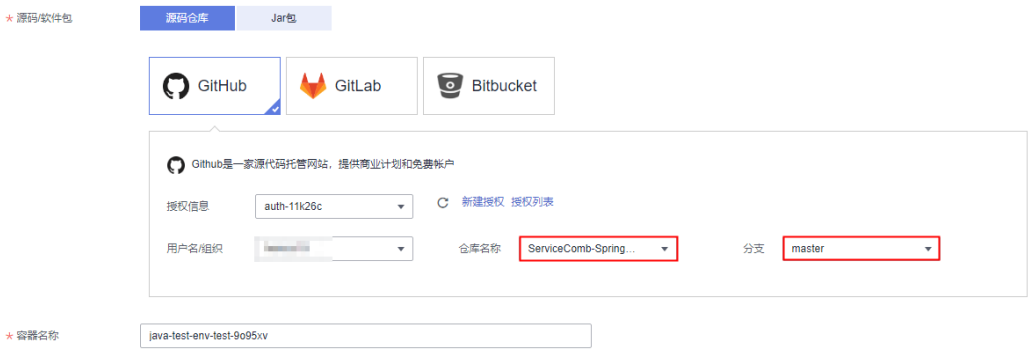
参数	说明
组件名称	输入组件名称（例如：java-test）。
组件版本	输入1.0.0。
所属环境	选择**创建环境**时创建的环境（例如：env-test）。
所属应用	选择**创建应用**时创建的应用（例如：servicecomb）。

图 1-9 设置组件基本信息

步骤3 在“组件包”区域，设置如下组件包必填参数，其余参数保持默认。

1. “技术栈”选择“Java”。
2. “源码/软件包”选择“源码仓库”。
3. 选择“GitHub”源码仓库。
4. “授权信息”选择**创建仓库授权**时创建的仓库授权名称。
5. “用户名/组织”选择**注册GitHub账号并复刻Demo源码**时创建的GitHub账号。
6. “仓库名称”选择**注册GitHub账号并复刻Demo源码**时复刻的Demo源码仓库名称“ServiceComb-SpringMVC”。
7. “分支”选择“master”。

图 1-10 设置组件来源

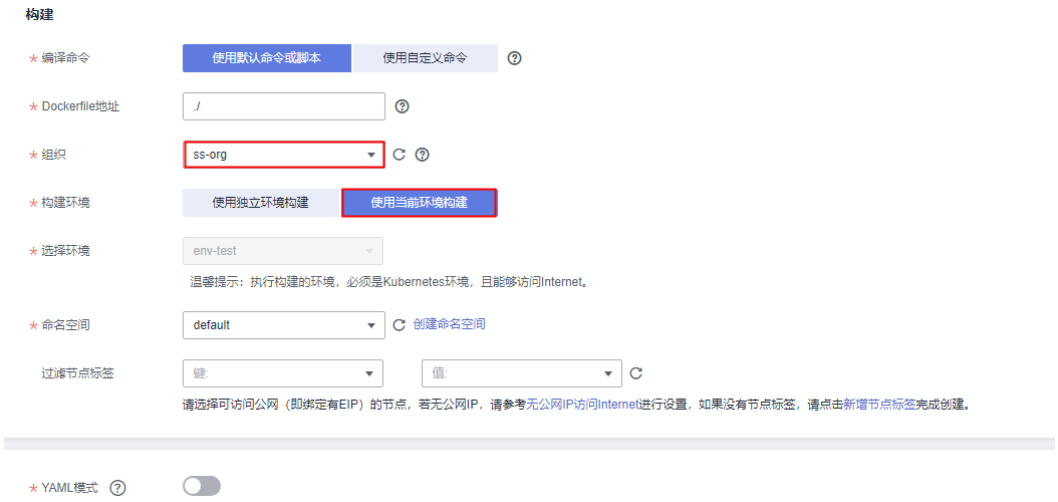


步骤4 在“构建”区域，参考下表设置必填构建参数，其余参数保持默认。

参数	说明
组织	选择**创建组织**时创建的组织的名称（例如：ss-org）。 组织用于管理组件构建生成的镜像。

参数	说明
构建环境	选择“使用当前环境构建”，使用组件所属的部署环境中的CCE集群进行镜像构建。 当前环境CCE集群的master节点和node节点的CPU架构必须保持一致，否则会导致组件构建失败。

图 1-11 设置构建参数



步骤5 单击“下一步”。

步骤6 “资源”区域，参考下表设置必填参数，其余参数保持默认。

参数	说明
资源需求	取消勾选“CPU配额”和“内存配额”，表示不限制资源需求。
实例数	设置为1。

图 1-12 设置组件实例“资源”参数

步骤7 绑定微服务引擎。

1. 选择“云服务配置 > 微服务引擎”。
2. 单击“绑定微服务引擎”。
3. 选择当前环境下已纳管的ServiceComb引擎专享版。
4. 单击“确定”。

图 1-13 绑定微服务引擎

步骤8 单击“创建并部署”，等待组件部署完成。

图 1-14 组件部署成功

----结束

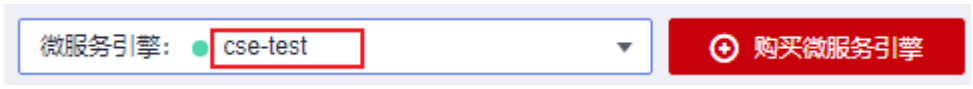
确认部署结果

步骤1 单击左上角 < ，返回“组件管理”页面。

图 1-15 返回“组件管理”页面

步骤2 选择“微服务引擎 > 微服务目录”。

步骤3 在微服务引擎下拉列表选择**创建环境**时纳管的微服务引擎。



步骤4 在“微服务列表”页签的“全部应用”下拉列表中选择springmvc应用。

如果存在已部署的servicecombspringmvc微服务，且微服务实例数为1，则表示组件实例成功接入微服务引擎。

图 1-16 确认部署结果

----结束

访问应用

步骤1 单击“应用管理”，进入应用列表。

步骤2 单击**创建应用**时创建的应用名称（例如：servicecomb），进入“应用概览”页。

步骤3 在“组件列表”区域，单击**创建并部署组件**时创建的组件名称（例如：java-test），进入组件“概览”页。

步骤4 单击“访问方式”。

步骤5 单击“TCP/UDP路由配置”区域的“添加服务”，参考下表设置参数。

参数	参数说明
服务名称	使用默认。
访问方式	选择“公网访问”。
访问类型	选择“弹性IP”。
服务亲和	保持默认。
端口映射	1. 协议：选择“TCP”。 2. 容器端口：输入8080。 3. 访问端口：选择“自动生成”。

图 1-17 添加访问方式

步骤6 单击“确定”，生成访问地址。

图 1-18 生成访问地址

步骤7 单击图1-18所示“访问地址”列下的访问地址，访问应用。

返回如下结果：

```
{"message":"Not Found"}
```

步骤8 在浏览器地址栏输入http://*步骤6生成的访问地址*/rest/helloworld?name=ServiceStage，再次访问应用。

可以返回如下结果：

```
"ServiceStage"
```

----结束

2 入门实践

您可以根据自身业务需求使用ServiceStage提供的一系列常用实践，以帮助您更好的理解和使用ServiceStage。

表 2-1 常用最佳实践

实践	描述
使用ServiceStage托管微服务应用	基于ServiceComb（SpringMVC）框架，快速创建微服务应用，供您体验ServiceStage的功能。
开启微服务引擎专享版安全认证	微服务引擎专享版支持基于RBAC（Role-Based Access Control，基于角色的访问控制）策略的安全认证，并支持开启/关闭安全认证。引擎开启了安全认证之后，要求所有连接该引擎的微服务都要配置安全认证账号和密码。否则，微服务将注册失败，导致业务受损。 本实践介绍未开启安全认证的微服务引擎专享版，开启安全认证并确保已接入引擎的微服务组件业务不受影响，即如何平滑开启安全认证。
微服务引擎仪表盘中的数据通过ServiceStage对接到AOM	部署到微服务引擎的Java Chassis应用，在微服务引擎仪表盘上的实时监控数据默认保留5分钟。如果需要持久化存储历史监控数据用于后续查询分析，可以使用ServiceStage的自定义指标监控功能，将微服务显示到微服务引擎仪表盘中的数据对接到AOM。 本实践以软件包部署应用为例，指导您完成将微服务引擎仪表盘中的数据通过ServiceStage对接到AOM。

实践	描述
使用ServiceStage零代码修改实现微服务注册引擎迁移	本实践指导您将使用Java Chassis微服务框架开发并注册在ServiceStage微服务引擎专业版上的微服务应用组件，零代码修改迁移注册到微服务引擎专享版。