阿里云 企业级分布式应用服务 EDAS

K8s 集群用户指南

法律声明

阿里云提醒您在阅读或使用本文档之前仔细阅读、充分理解本法律声明各条款的内容。如果您阅读或 使用本文档,您的阅读或使用行为将被视为对本声明全部内容的认可。

- **1.** 您应当通过阿里云网站或阿里云提供的其他授权通道下载、获取本文档,且仅能用于自身的合法合规的业务活动。本文档的内容视为阿里云的保密信息,您应当严格遵守保密义务;未经阿里云事先书面同意,您不得向任何第三方披露本手册内容或提供给任何第三方使用。
- **2.** 未经阿里云事先书面许可,任何单位、公司或个人不得擅自摘抄、翻译、复制本文档内容的部分或全部,不得以任何方式或途径进行传播和宣传。
- 3. 由于产品版本升级、调整或其他原因,本文档内容有可能变更。阿里云保留在没有任何通知或者提示下对本文档的内容进行修改的权利,并在阿里云授权通道中不时发布更新后的用户文档。您应当实时关注用户文档的版本变更并通过阿里云授权渠道下载、获取最新版的用户文档。
- **4.** 本文档仅作为用户使用阿里云产品及服务的参考性指引,阿里云以产品及服务的"现状"、"有缺陷"和"当前功能"的状态提供本文档。阿里云在现有技术的基础上尽最大努力提供相应的介绍及操作指引,但阿里云在此明确声明对本文档内容的准确性、完整性、适用性、可靠性等不作任何明示或暗示的保证。任何单位、公司或个人因为下载、使用或信赖本文档而发生任何差错或经济损失的,阿里云不承担任何法律责任。在任何情况下,阿里云均不对任何间接性、后果性、惩戒性、偶然性、特殊性或刑罚性的损害,包括用户使用或信赖本文档而遭受的利润损失,承担责任(即使阿里云已被告知该等损失的可能性)。
- 5. 阿里云文档中所有内容,包括但不限于图片、架构设计、页面布局、文字描述,均由阿里云和/或 其关联公司依法拥有其知识产权,包括但不限于商标权、专利权、著作权、商业秘密等。非经阿 里云和/或其关联公司书面同意,任何人不得擅自使用、修改、复制、公开传播、改变、散布、发 行或公开发表阿里云网站、产品程序或内容。此外,未经阿里云事先书面同意,任何人不得为了 任何营销、广告、促销或其他目的使用、公布或复制阿里云的名称(包括但不限于单独为或以组 合形式包含"阿里云"、"Aliyun"、"万网"等阿里云和/或其关联公司品牌,上述品牌的附属 标志及图案或任何类似公司名称、商号、商标、产品或服务名称、域名、图案标示、标志、标识 或通过特定描述使第三方能够识别阿里云和/或其关联公司)。
- 6. 如若发现本文档存在任何错误,请与阿里云取得直接联系。

通用约定

格式	说明	样例
	该类警示信息将导致系统重大变更甚至 故障,或者导致人身伤害等结果。	禁止: 重置操作将丢失用户配置数据。
A	该类警示信息可能会导致系统重大变更 甚至故障,或者导致人身伤害等结果。	警告: 重启操作将导致业务中断,恢复业务时间约十分钟。
•	用于警示信息、补充说明等 <i>,</i> 是用户必须了解的内容。	(! 注意: 权重设置为0,该服务器不会再接受新请求。
	用于补充说明、最佳实践、窍门等 <i>,</i> 不是用户必须了解的内容。	说明: 您也可以通过按Ctrl + A选中全部文件。
>	多级菜单递进。	单击 设置 > 网络 > 设置网络类型 。
粗体	表示按键、菜单、页面名称等UI元素。	在 结果确认 页面 <i>,</i> 单击 确定 。
Courier字体	命令。	执行cd /d C:/window命令,进 入Windows系统文件夹。
斜体	表示参数、变量。	bae log listinstanceid Instance_ID
[]或者[a b]	表示可选项,至多选择一个。	ipconfig [-all -t]
{}或者{a b}	表示必选项,至多选择一个。	switch {active stand}

目录

泛	&律声明	I
诵	通用约定	I
	Kubernetes集群概述	
	新版/旧版用户指南导读	
3	应用管理升级	6
4	命名空间	7
	创建并导入容器服务K8s集群	
	应用部署	
0	—···	
	6.1 制作应用容器 Docker 镜像	
	6.3 使用控制台部署	
	6.3.1 在容器服务K8s集群中使用镜像方式部署应用	
	6.3.2 在容器服务K8s集群中使用JAR包或WAR包部署应用	
	6.4 使用工具部署	
	6.4.1 使用toolkit-maven-plugin插件部署应用到容器服务K8s集群	
	6.4.2 使用 Cloud Toolkit 插件快速部署应用到容器服务 K8s 集群	
	6.4.3 使用云效部署Java应用至EDAS(K8s集群)	47
	6.5 金丝雀发布(K8s 集群)	63
	6.6 分批发布(K8s 集群)	66
7	6.6 分批发布 (K8s 集群)	
7		70
7	用户指南(旧版)	70
7	用户指南(旧版)	70 70
7	用户指南(旧版) 7.1 容器服务K8s集群的应用生命周期管理	70 70 70
7	用户指南(旧版) 7.1 容器服务K8s集群的应用生命周期管理	
7	用户指南(旧版) 7.1 容器服务K8s集群的应用生命周期管理	707070717274
7	用户指南(旧版) 7.1 容器服务K8s集群的应用生命周期管理	707071727475
7	用户指南(旧版) 7.1 容器服务K8s集群的应用生命周期管理 7.2 应用设置 7.2.1 为 Kubernetes 集群中的应用添加负载均衡 SLB 7.2.2 容器服务 K8s 集群中的应用如何复用 SLB 7.3 应用监控 7.3.1 应用总览 7.3.2 应用详情 7.3.2.1 JVM 监控	707071727475
7	用户指南(旧版) 7.1 容器服务K8s集群的应用生命周期管理 7.2 应用设置 7.2.1 为 Kubernetes 集群中的应用添加负载均衡 SLB 7.2.2 容器服务 K8s 集群中的应用如何复用 SLB 7.3 应用监控 7.3.1 应用总览 7.3.2 应用详情 7.3.2.2 主机监控	707071747575
7	用户指南(旧版) 7.1 容器服务K8s集群的应用生命周期管理 7.2 应用设置 7.2.1 为 Kubernetes 集群中的应用添加负载均衡 SLB 7.2.2 容器服务 K8s 集群中的应用如何复用 SLB 7.3 应用监控 7.3.1 应用总览 7.3.2 应用详情 7.3.2.2 主机监控 7.3.3 应用接口调用监控	7070717274757678
7	用户指南(旧版) 7.1 容器服务K8s集群的应用生命周期管理 7.2 应用设置 7.2.1 为 Kubernetes 集群中的应用添加负载均衡 SLB 7.2.2 容器服务 K8s 集群中的应用如何复用 SLB 7.3 应用监控 7.3.1 应用总览 7.3.2 应用详情 7.3.2 应用详情 7.3.2.1 JVM 监控 7.3.3 应用接口调用监控 7.3.4 高级监控	707070717275757678
7	用户指南 (旧版) 7.1 容器服务K8s集群的应用生命周期管理 7.2 应用设置 7.2.1 为 Kubernetes 集群中的应用添加负载均衡 SLB 7.2.2 容器服务 K8s 集群中的应用如何复用 SLB 7.3 应用监控 7.3.1 应用总览 7.3.2 应用详情 7.3.2.1 JVM 监控 7.3.2.2 主机监控 7.3.3 应用接口调用监控 7.3.4 高级监控 7.3.5 报警管理	70707172757576788083
7	用户指南(旧版) 7.1 容器服务K8s集群的应用生命周期管理 7.2 应用设置 7.2.1 为 Kubernetes 集群中的应用添加负载均衡 SLB. 7.2.2 容器服务 K8s 集群中的应用如何复用 SLB. 7.3 应用监控 7.3.1 应用总览. 7.3.2 应用详情. 7.3.2.1 JVM 监控. 7.3.2.2 主机监控. 7.3.3 应用接口调用监控. 7.3.4 高级监控. 7.3.5 报警管理. 7.3.5.1 创建报警.	7070707172757576788083
7	用户指南 (旧版) 7.1 容器服务K8s集群的应用生命周期管理 7.2 应用设置 7.2.1 为 Kubernetes 集群中的应用添加负载均衡 SLB 7.2.2 容器服务 K8s 集群中的应用如何复用 SLB 7.3 应用监控 7.3.1 应用总览 7.3.2 应用详情 7.3.2.1 JVM 监控 7.3.2.2 主机监控 7.3.3 应用接口调用监控 7.3.4 高级监控 7.3.5 报警管理	70707172757676808383
7	用户指南(旧版) 7.1 容器服务K8s集群的应用生命周期管理 7.2 应用设置 7.2.1 为 Kubernetes 集群中的应用添加负载均衡 SLB 7.2.2 容器服务 K8s 集群中的应用如何复用 SLB 7.3 应用监控 7.3.1 应用总览 7.3.2 应用详情 7.3.2 应用详情 7.3.2.1 JVM 监控 7.3.2.2 主机监控 7.3.3 应用接口调用监控 7.3.4 高级监控 7.3.5 报警管理 7.3.5.1 创建报警 7.3.5.2 管理报警	707070717275767680838383
7	用户指南(旧版) 7.1 容器服务K8s集群的应用生命周期管理 7.2 应用设置 7.2.1 为 Kubernetes 集群中的应用添加负载均衡 SLB 7.2.2 容器服务 K8s 集群中的应用如何复用 SLB 7.3 应用监控 7.3.1 应用总览 7.3.2 应用详情 7.3.2 应用详情 7.3.2.2 主机监控 7.3.3 应用接口调用监控 7.3.4 高级监控 7.3.5 报警管理 7.3.5.1 创建报警 7.3.5.3 创建联系人	707070717475767680838383

8	上传本地镜像	112
	7.8.3 查看应用的文件日志和容器标准输出日志	110
	7.8.2 日志目录	
	7.8.1 实时日志	
	7.8 日志诊断	
	7.7.1 弹性伸缩(K8s)	
	7.7 弹性伸缩	
	7.6 限流降级	101
	7.5.2 常见Pod问题	100
	7.5.1 查看应用事件	99
	7.5 应用事件	99

IV 文档版本: 20200707

1 Kubernetes集群概述

为了让用户可以方便地在阿里云上使用Kubernetes管理容器应用,EDAS集成了容器服务Kubernetes集群。您可以在容器服务Kubernetes版控制台创建Kubernetes集群,然后将集群导入EDAS用于应用部署,EDAS可提供高可用的容器应用管理能力,提高运维效率,让您专注于容器化的应用的开发与管理。

Kubernetes集群概述

现EDAS支持导入在容器服务Kubernetes版控制台创建的专有版Kubernetes集群、托管版Kubernetes集群和Serverless Kubernetes集群,3种集群的形态的对比详情请参见#unique 4。

专有版 Kubernetes

需要创建 3 个 Master(高可用)节点及若干 Worker 节点,可对集群基础设施进行更细粒度的控制,需要自行规划、维护、升级服务器集群。

• 托管版 Kubernetes

只需创建 Worker 节点,Master 节点由容器服务创建并托管。具备简单、低成本、高可用、无需运维管理 Kubernetes 集群 Master 节点的特点,您可以更多关注业务本身。托管版集群适合大多数业务场景,是一种最通用的集群。

Serverless Kubernetes

无需创建和管理 Master 节点及 Worker 节点,即可通过控制台或者命令配置容器实例的资源、指明应用容器镜像以及对外服务的方式,直接启动应用程序。Serverless Kubernetes则适合快速伸缩的敏捷业务场景以及单个或多个任务处理的场景。

在EDAS中部署应用至Kubernetes集群概述

EDAS无缝集成容器服务Kubernetes,并提供Kubernetes集群应用生命周期管理和轻量化运维的全新体验。除了提供Kubernetes集群的基础托管服务,还支持应用实例打散一键多AZ高可用部署、应用发布版本管理、发布变更跟踪和应用层极致弹性等全流程管控治理能力,提供一站式PaaS体验。

图 1-1: 部署应用至Kubernetes集群流程图



针对容器服务Kubernetes集群:

在容器服务Kubernetes版控制台创建Kubernetes托管版集群的详细步骤请参见#unique 7。

文档版本: 20200707 1

- 在EDAS控制台可使用镜像部署应用至容器服务K8s集群,详情请参见在容器服务K8s集群中使用 镜像方式部署应用。
- 在EDAS控制台可使用JAR包或WAR包部署应用至容器服务K8s集群,详情请参见在容器服务K8s集群中使用JAR包或WAR包部署应用。

针对Serverless Kubernetes集群:

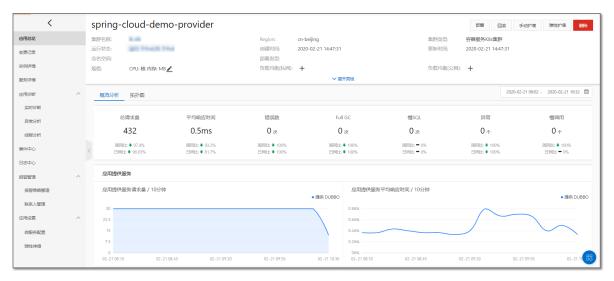
- 在容器服务Kubernetes版控制台创建Serverless Kubernetes集群的详细步骤请参见#unique_10。
- 在EDAS控制台可使用镜像部署应用至Serverless K8s集群,详情请参见#unique_11。
- 在EDAS控制台可使用JAR包或WAR包部署应用至Serverless K8s集群,详情请参见#unique_12。

2 新版/旧版用户指南导读

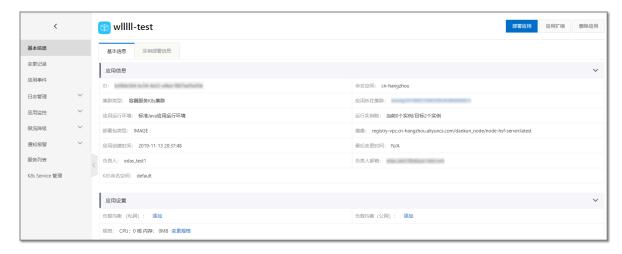
2020-02-20 24: 00 EDAS 底层架构进行了升级,该时间后导入的 K8s 集群,其上的应用具备轻量化运维能力;该时间前导入的存量 K8s 集群,其上应用采用原底层架构,功能无变更。在进行应用管理时请阅读本文。

使用场景

• 新版: 2020-02-20 24: 00 后在 EDAS 中导入的 K8s 集群,其应用管理页面为新版页面。文档请使用用户指南(新版)。



旧版: 2020-02-20 24: 00 前在 EDAS 中导入的 K8s 集群,其应用管理页面任继续使用原应用详情页面。文档请使用用户指南(旧版)。





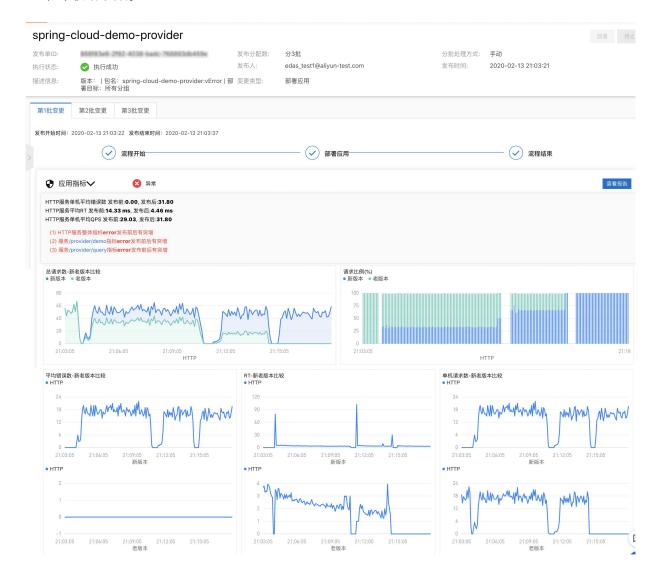
说明:

2020年2月20日24:00 EDAS K8s 集群自动升级后,其上新创建的应用管理页面为新版页面;该时间点前创建的存量应用后续将以一键升级方式升级为新版页面,敬请期待。

优势

新版的应用管理功能提供了一站式轻量化运维能力。K8s 集群导入 EDAS 时,EDAS 便对集群及其上的应用进行实时监控,如服务详情、实例详情、变更记录,并实时提供运行报告以及相应的解决方案。

例如查看变更记录功能。在新版中,EDAS展示了变更过程中应用指标的实时信息,如平均错误数、RT和单机请求数。



功能对比

功能	新版	旧版	差异
应用总览	增强	支持	相对于旧版,新版应页面用总览除包含基本信息外,还展示了该应用的诊断报告以及解决方案、该应用的概览信息和拓扑图,详情请参见#unique_14。
生命周期管理	支持	支持	新版与旧版相同。

功能	新版	旧版	差异
扩缩容	增强	支持	相对于旧版,新版手动扩容缩和自动弹性处于相邻位置,便于操作,详情请参见#unique_16。
负载均衡	增强	支持	相对于旧版,支持 SLB 配置 HTTPS 监听,详情请参见#unique_17。
应用变更	增强	支持	相对于旧版,新版增加了变更过程指标监控,如应用指标监控、异常监控、系统监控和 Pod 执行日志,具体请参见#unique_18
实例详情	支持	支持	与旧版的应用详情监控功能相同。
服务详情	支持	支持	与旧版的应用接口调用功能相同。
应用诊断	新增	不支持	相对于旧版,新版中增加了实时诊断和线程分析 功能,具体请参见应用诊断。
应用监控	支持	支持	相对于旧版,新版更侧重于一站式实时监控,简化运维。
报警管理	支持	支持	新版与旧版相同。
日志管理	支持	支持	新版与旧版相同。
事件管理	增强	支持	相对于旧版,新版展示了更多事件信息,如应用报警、诊断报告等,具体请查看应用事件。
微服务治理	支持	支持	新版与旧版相同。

文档版本: 20200707 5

3 应用管理升级

如果您想使用新版的一站式轻量化应用管理,需要对旧版应用进行升级。本文介绍如将旧版应用使用一键升级方式或者静默方式升级到新版。

一键升级

为了方便您便捷使用一站式轻量化应用管理,EDAS提供了应用管理的一键升级功能。

- 1. 登录EDAS控制台。
- 2. 执行以下操作之一:
 - 在左侧导航栏选择**资源管理 > 集群**,在**集群**页面的**容器服务K8s集群**页签上单击集群ID或名称,然后在**集群详情**页面的**应用列表**区域单击具体应用名称。
 - 在左侧导航栏选择**应用管理 > 应用列表**,在**应用列表**页面单击具体容器服务K8s集群应用的应用名称。
- 3. 在基本信息页面右上角单击体验新版。
- 4. 在升级新版应用管理对话框中勾选重新部署应用(单批发布,滚动升级),立即升级新版应用管理,并单击立即升级新版(重新部署)。

部署完成后,您的应用管理页面升级为新版的应用管理页面,新版应用管理页面详情请参见用户 指南(新版)。

如果您习惯于旧版,那么升级后可以在应用的**基本信息**页右上角单击**回到旧版**。



说明:

- 应用管理页面升级后,如果需要对应用进行 SLB 配置,须在升级后的应用管理页面进行配置,请勿返回旧版进行配置。
- 如果在升级前已经配置了 SLB, 升级后, 原有的 SLB 配置将会同步至新版。

静默升级

EDAS 除了提供一键升级方式将旧版应用管理升级至新版,还支持以静默的方式(单批或者分批部署)进行升级,具体操作请参见分批发布(K8s 集群)。应用完成部署后,便拥有了新版的应用管理页面。

4命名空间

命名空间为应用提供互相隔离的运行环境,如开发、测试和生产环境等,您可以使用命名空间实现资源和服务的隔离。在同一个地域内不允许创建两个同名的命名空间。

应用场景

例如,您在同一个VPC内有三套环境,分别用于应用开发、测试和生产,可以为这三套环境分别创建一个命名空间Dev、Test和Prod。在这三个命名空间中分别创建集群并部署应用,则不同命名空间下的资源、应用和服务也是相互隔离的,不能进行服务调用和配置推送。

默认命名空间

在EDAS创建应用时,可以选择**默认**命名空间,默认命名空间即为没有命名空间的状态,表示该应用不使用任何命名空间来实现资源和服务的隔离。

在应用列表内选择某地域下的默认命名空间时,即表示筛选出来的应用都没有归属于任何命名空间。

创建命名空间

- 1. 登录EDAS控制台。
- 2. 在左侧导航栏中选择应用管理 > 命名空间。
- 3. 在命名空间页面右上角单击创建命名空间。
- **4.** 在**创建命名空间**对话框配置命名空间参数,然后单击**创建**。



参数	描述
命名空间名称	请输入您创建的命名空间的名称。
命名空间ID	请输入自定义的字符来形成命名空间的ID,仅允许输入英文字母或数字。

文档版本: 20200707 7

参数	描述
归属地域	当前命名空间所归属的地域,不可更改。
允许远程调试	当您想对应用进行端云互联时,您在该应用所在的命名空间的编辑页面手动开启 允许远程调试 ,端云调试的相关操作请参见#unique_24。
描述	请输入一段文字来描述命名空间。

编辑命名空间

- 1. 在命名空间列表的操作列单击编辑。
- **2.** 在**编辑命名空间**对话框,您可修改**命名空间名称**及**描述**,也可开启或关闭远程调试功能。在编辑完成后单击**确定**。



主意:

命名空间ID和命名空间类型在编辑时不可更改。

删除命名空间

删除命名空间需要满足以下前提条件:

- 命名空间下无任何集群。
- 命名空间下无任何ECS实例。
- 1. 在命名空间列表的操作列中单击删除。
- 2. 在对话框中单击删除。

20200707

5 创建并导入容器服务K8s集群

容器服务Kubernetes(K8s)版提供高性能可伸缩的容器应用管理能力,支持企业级Kubernetes容器化应用的全生命周期管理。容器服务Kubernetes版简化集群的搭建和扩容等工作,整合阿里云虚拟化、存储、网络和安全能力,打造云端最佳的Kubernetes容器化应用运行环境。

前提条件

- 阿里云账号已开通EDAS服务,详情请参见开通EDAS服务。
- 该阿里云账号已开通容器服务Kubernetes版并完成角色授权,详情请参见#unique_26。

步骤一:在容器服务Kubernetes版控制台中创建容器服务Kubernetes集群

- 1. 登录容器服务K8s版控制台。
- 2. 创建Kubernetes集群。

容器服务中可以创建Kubernetes、Kubernetes托管版和多可用区Kubernetes三种类型的集群。您可以根据您的实际需求选择具体类型。

- #unique_27: 您购买、添加的实例中有3个实例需要作为管控(Master)节点,这3个实例是不能部署应用的。您只能在另外购买的实例(Worker)上部署应用。
- #unique_28: 您购买、添加的实例中都是Worker节点,即可以用于部署应用的实例。

步骤二:在EDAS控制台中导入容器服务的Kubernetes集群

- 1. 登录EDAS控制台。
- 2. 在左侧导航栏中选择资源管理 > 集群。
- **3.** 在**集群**页面单击**容器服务K8s集群**页签。在集群列表中选择已创建的容器服务Kubernetes集群,在**操作**列单击**导入**,并在**导入Kubernetes集群**对话框中单击**导入**。



当创建的容器服务Kubernetes集群的**操作**列的按钮变为**删除**,且集群状态为**运行中**,则表示容器服务的Kubernetes版集群已成功导入到EDAS。

查看并管理容器服务K8s集群

单击集群ID即可进入**集群详情**页面。在该页面您可查看该集群的详细信息、ECS实例、Deployment、应用列表和服务网格等信息。

- 集群信息:该区域展示了集群基本信息,如集群ID、csClusterId、集群名称、命名空间、集群类型、VPC ID、网络类型、集群状态和描述信息等信息。
- **ECS实例**:该区域展示了该集群包含的ECS实例列表和ECS实例概要信息。您可以在此批量移 除ECS实例,或将按量付费改为包年包月。
 - 您可在应用列表左侧勾选要进行移除的ECS实例,单击**批量移除**,然后根据指示完成后续操作。
 - 您可在应用列表左侧勾选要进行按量付费转包年包月的ECS实例,单击**按量付费转包年包 月**,然后根据指示完成后续操作。
- **Deployment**: 该区域展示您转换和托管到EDAS的第三方平台自建Deployment。
- **应用列表**:该区域展示了集群内包含的应用,您可以查看应用名称、JDK版本、应用运行环境、实例总数、运行中的实例数和应用负责人等信息。您可以单击应用名称进入应用详情页面。
- 服务网格:该区域展示服务网格信息。您可以在此安装和删除服务网格。

6 应用部署

6.1 制作应用容器 Docker 镜像

您可以在本地的开发工具中使用命令将应用程序打成 WAR 包或 JAR 包,直接部署;也可以基于 WAR 包或 JAR 包制作镜像,再将镜像上传到阿里云镜像仓库来部署。本文介绍如何制作不同框架应 用镜像的 Dockfile 以及如何将镜像上传到阿里云镜像仓库。

前提条件

在制作应用镜像前,请查阅附录:运行时环境变量信息并按照规约,同时参考以下内容制作 EDAS 应用镜像。

创建标准 Dockerfile

Dockerfile 是一个文本格式的配置文件,您可以使用 Dockerfile 快速创建镜像。

您可以根据实际应用框架,通过 Dockerfile 分别制作 HSF、Spring Cloud 或 Dubbo 应用镜像。下面将通过示例分别介绍如何创建这些不同框架应用的 Dockerfile。

一个 EDAS 标准 Dockerfile 描述了 EDAS 创建应用运行环境的所有指令,包括下载、安装和启动 OpenJDK、Tomcat、WAR 包和 JAR 包等。 您还可以通过改变 Dockerfile 内容,完成如 OpenJDK 版本替换、Tomcat 配置修改、改变运行时环境等操作。详情请参见自定义镜像:使用新版本的 OpenJDK。

- HSF 应用的 Dockerfile 示例
- Spring Cloud 或 Dubbo 应用的 Dockerfile 示例

HSF 应用的 Dockerfile 示例

Using the centos7 + openjdk8 + ali tomcat7 + pandora 3.5.9 for the base FROM apaas/edas-centos-openjdk8-alitomcat7:latest MAINTAINER Alibaba Cloud EDAS Team<edas-dev@list.alibaba-inc.com>

ENV TZ="Asia/Shanghai"

Default put your application package into /home/admin/app/ ENV APP_HOME /home/admin/app/

#####----> First case: deploy a fat-jar file ARG APP_LOCATION=https://edas-hz.oss-cn-hangzhou.aliyuncs.com/prod/demo/ HSF_PROVIDER.jar

#####----> Second case: deploy a war file, simply change the file name your wanted. #ARG APP_LOCATION=https://edas-hz.oss-cn-hangzhou.aliyuncs.com/prod/demo/HSF_PROVIDER.war

#####----> Third case: deploy a local storage file.
#ARG APP_LOCATION=/Users/yanliang.lyl/workspace/java/demo/HSF_PROVIDER.war

```
# Then download to package into /home/admin/app/
ADD ${APP_LOCATION} ${APP_HOME}/

# Default working dir is set to /home/admin

# WORKDIR /home/admin

# Entry point set to /home/admin/bin/start.sh,

# which including inside basic image (apaas/edas-centos-openjdk8-alitomcat8)

# CMD ["/home/admin/bin/start.sh"]
```

Spring Cloud 或 Dubbo 应用的 Dockerfile 示例

```
# Using centos7 + openjdk8 + tomcat8 for the base
FROM apaas/edas:latest
# Default put your application package into /home/admin/app/
ENV APP HOME /home/admin/app/
##### SIMPLE REPLACE YOUR PACKAGE FILE
#####----> First case: deploy a fat-jar file
ARG APP_LOCATION=https://edas-hz.oss-cn-hangzhou.aliyuncs.com/prod/demo/
DUBBO PROVIDER.jar
#####----> Second case: deploy a war file, simply change the file name your wanted.
#ARG APP LOCATION=https://edas-hz.oss-cn-hangzhou.aliyuncs.com/prod/demo/
DUBBO PROVIDER.war
#####----> Third case: deploy a local storage file.
#ARG APP LOCATION=/Users/yanliang.lyl/workspace/java/demo/SPRINT_CLO
UD PROVIDER.war
# Then download to package into /home/admin/app/
ADD ${APP_LOCATION} ${APP_HOME}/
```

EDAS 基础镜像说明

您可以根据您应用的运行环境,自由选择以下的基础镜像:

镜像名称	OS版本	Java 版本	Tomcat 版本	Pandora 版本	备注
apaas/edas	CentOS 7	OpenJDK 1.8	8.5.42	N/A	与 apaas/ edas-centos -openjdk8- tomcat8 相 同,最新版本 号为: latest
apaas/edas -centos- openjdk8- tomcat8	CentOS 7	OpenJDK 1.8	8.5.42	N/A	普通 SpringCloud / Dubbo 应用 推荐使用此版 本,最新版本 号为: latest

镜像名称	OS版本	Java 版本	Tomcat 版本	Pandora 版本	备注
apaas/edas -centos- openjdk8- tomcat7	CentOS 7	OpenJDK 1.8	7.0.93	N/A	基于 Tomcat 7.x 的普通 SpringCloud / Dubbo 应用 推荐使用此版 本,最新版本 号为: latest
apaas/edas -centos- openjdk8- alitomcat8	CentOS 7	OpenJDK 1.8	8.5.37	3.5.9	基于 Tomcat8 .x 的 HSF 应用 推荐使用此版 本,最新版本 号为:latest
apaas/edas -centos- openjdk8- alitomcat7	CentOS 7	OpenJDK 1.8	7.0.92	3.5.9	基于 Tomcat7 .x 的 HSF 应用 推荐使用此版 本,最新版本 号为: latest

更多镜像请访问 aPaaS Docker Hub 官方镜像仓库;同时,如果您需要额外的基础镜像,请提交工单给我们;如果您对我们的基础镜像有任何疑问,请参考基础镜像 Dockerfile 的 GIT HUB 地址。

自定义镜像:使用新版本的 OpenJDK

您可以在创建的标准 Dockerfile 上根据实际需求进行自定义设置。

升级 OpenJDK:在 Dockerfile中,可下载和安装新版本的 OpenJDK,以下示例为下载安装 OpenJDK 9。

先卸载之前的 JDK,然后再下载安装 OpenJDK 9 RUN yum erase -y java; yum -y install java-1.9.0-openjdk-devel

自定义镜像: 升级 EDAS Container 版本 (HSF 应用)

对于部署到容器服务 K8s 集群中的 HSF 应用,为使用中间件的新特性或修复原有版本的问题,可以升级应用容器版本。升级步骤如下。

- **1.** 登录 EDAS 控制台。在左侧导航栏选择**应用管理 > 应用列表**,在**应用列表**页面右上角单击**创建应 用**,在**应用基本信息**页面查看**应用运行环境**,查看最新的 EDAS-Container 版本号(3.X.X)。
- 2. 参考容器版本说明,获取 Pandora 容器版本。
- 3. 替换 Dockerfile 中的版本号信息, 如 3.5.4。

设置 EDAS-Container/Pandora 应用容器版本 ENV UPGRADED_CONTAINER_VERSION 3.5.4

RUN /home/admin/bin/upgradecontainer.sh

4. 重新制作并发布应用镜像。

自定义镜像:在镜像中调整 JVM 启动参数

基于基础镜像中的 JVM 参数,我们将通过**JAVA_OPTS**环境变量进行传递,下例中说明了如何修改 JVM 启动参数的方式:

```
FROM apaas/edas:latest
#设置 JVM 参数
ENV JAVA_OPTS="\
-Xmx3550m \
-Xms3550m \
-Xmn2g \
-Xss128k"
```

自定义镜像: SpringBoot 类型的应用修改应用配置

基于 EDAS 基础镜像中的部署 FatJar 类型的应用,如果需要修改 Tomcat 的启动配置如:Context Path、启动端口、参数等,将借助APP ARGS 环境变量进行传递,举例如下

```
# 基础镜像中的默认配置为:
# - context path 为 "/" ,
# - uri 编码为 ISO-8859-1
# - 应用启动端口为 8080 ,
# - tomcat 处理线程池数量为 400

# 您也可以通过复写 APP_ARGS 这个环境变量来修改默认配置,如下:
ENV APP_ARGS="--server.context-path=/\
--server.tomcat.uri-encoding=ISO-8859-1\
--server.port=8080\
--server.tomcat.max-threads=400"
```

在本地构建镜像

从本地命令行进入 Dockerfile 所在的目录,执行 docker build 命令构建镜像:

```
docker build -t [标签名称,最好取应用名]:[版本号] .
或
docker build -t [标签名称,最好取应用名]:[版本号] -f /path/to/custom_dockerfile_name .
#假如您创建好的 Dockerfile 在其他位置或名称不为 Dockerfile 时适用。
```

例如:

```
docker build -t hsf-provider:1.0.0 .
```

然后使用 docker images | grep <镜像标签名称> 命令查看本地打包好的镜像。

上传镜像到镜像仓库

您在本地生成构建好的应用镜像就可以上传到阿里云提供的容器镜像仓库(需要您提前访问阿里云镜像仓库控制台,并根据您 EDAS 中已经创建或尚未创建的应用所在的 Region,在阿里云镜像仓库控

制台切换到对应的 Region,查看镜像仓库列表,选择一个已经存在或者新创建的镜像仓库来存放您打包好的应用镜像,仓库名称最好是应用名称,便于识别)。

执行指定镜像仓库基本信息页面提供的 "将镜像推送到 Registry" 相关命令,将本地镜像上传到该镜像仓库。

docker login --username=[当前登录的账号名] registry.[region_id].aliyuncs.com #这一步输入的密码为阿里云容器镜像服务默认实例 -> 访问凭证页面设置的固定或者临时密码,并非阿里云账号密码。

docker tag [本地应用镜像ID] registry.[region_id].aliyuncs.com/[命令空间名称]/[镜像仓库名称]:[镜像版本号]

docker push registry.[region_id].aliyuncs.com/[命令空间名称]/[镜像仓库名称]:[镜像版本号]

例如:

docker login --username=tdy218@gmail.com registry.cn-hangzhou.aliyuncs.com docker tag 2b64f63601a7 registry.cn-hangzhou.aliyuncs.com/webapps/hsf-provider:1.0 .0 docker push registry.cn-hangzhou.aliyuncs.com/webapps/hsf-provider:1.0.0

(一)阿里云 〇 排索 费用 工单 备案 企业 支持与服务 🖂 🐧 🖢 🕢 3010316 < 华东1(杭州) | 私有 | 本地仓库 | ● 正常 基本信息 基本信息 仓库授权 仓库名称 触发器 仓库地域 华东1 (杭州) 专有网络 @ registry-vpc.cn-hangzhou.aliyuncs.com 复制 仓库类型 私有 经典网络 @ registry-internal.cn-hangzhou.aliyuncs.com/地域上 复制 镜像版本 代码仓库 无 摘要 @ for tdy218 using. 镜像同步 镜像描述 1. 登录阿里云Docker Registry \$ sudo docker login --username= 用于登录的用户名为阿里云账号全名,密码为开通服务时设置的密码。 您可以在产品控制台首页修改登录密码。 2. 从Registry中拉取镜像 \$ sudo docker pull registry.cn-hangzhou.aliyuncs.com/配面 Light :[镜像版本号] 3. 将镜像推送到Registry

阿里云镜像仓库网络加速: https://cr.console.aliyun.com/#/accelerator

\$ sudo docker login --username=tdy218fedsstemt1 registry.cn-hangzhou.aliyuncs.com \$ sudo docker tag [ImageId] registry.cn-hangzhou.aliyuncs.com/edss_mesti/tdy218:[镜像版本号] \$ sudo docker push registry.cn-hangzhou.aliyuncs.com/edss_mesti/tdy218:[镜像版本号]

附录:运行时环境变量信息

通过 Dockerfile 制作自定义镜像时,有一些环境变量是 EDAS 在运行时会自动注入的环境信息,这些信息如下表:

环境变量 Key	描述
POD_IP	POD IP

文档版本: 20200707 15

环境变量 Key	描述
EDAS_APP_ID	EDAS 应用 ID
EDAS_ECC_ID	EDAS ECC ID
EDAS_PROJECT_NAME	同 EDAS_APP_ID,用于调用链解析
EDAS_JM_CONTAINER_ID	同 EDAS_ECC_ID,用于调用链解析
EDAS_CATALINA_OPTS	中间件运行时所需 CATALINA_OPTS 参数
CATALINA_OPTS	同 EDAS_CATALINA_OPTS,默认 TOMCAT 启动 参数
CATALINA_HOME	所安装的 TOMCAT 路径
PANDORA_LOCATION	所安装的 Pandora 路径,HSF 应用中可见

同时,请勿使用 /home/admin/.spas_key/default 文件,此文件会在 POD 运行时被覆盖。

6.2 应用部署概述 (K8s集群)

Spring Cloud、Dubbo或HSF框架应用开发完成后,您可使用WAR包或JAR包的方式将应用部署至K8s集群。除可在EDAS控制台完成部署外,您还可以在控制台或使用插件、云效等工具将应用部署至K8s集群。

应用运行环境

Spring Cloud、Dubbo和HSF应用部署在K8s集群中时,选择不同部署方式时所选择的应用运行环境会有所不同,如下所示:

- Spring Cloud、Dubbo、普通Java Web或Spring MVC应用,且未使用过Pandora或Pandora Boot开发,通过WAR包部署时,需要选择**Apache Tomcat** 相关版本的运行环境。
- Spring Cloud、Dubbo、普通Java Web或Spring MVC应用,且未使用过Pandora或Pandora Boot开发,通过JAR包部署时,需要选择标准Java应用运行环境。
- HSF应用,包括使用Pandora Boot开发的Spring Cloud应用和由Dubbo转换为HSF的应用,通
 过WAR或JAR包部署时,需要选择EDAS-Container相关版本的运行环境。

Demo概述

为了方便您快速开始使用EDAS,EDAS为您准备了基于不同应用框架(Spring Cloud、Dubbo和HSF)的微服务应用Demo。Demo内包含了简单的服务调用关系(服务提供与服务消费),您只需要在部署时选择Demo即可,然后完成应用部署操作,具体操作请参见#unique_32。部署成功后,为应用绑定SLB,您可以体验服务调用和以及EDAS微服务管理功能如限流降级、服务监控和日志查看等。

• 容器端口: Consumer容器端口默认为18082、Provider的容器端口默认为18081。

您还可以在应用详情页面的**日志中心 > 实时日志(标准输出)**页签中,查询应用的容器端口,如下图所示:

```
test-doc-consumer
 实时日志 (标准输出) 日志目录 (在线查看) 文件日志 (SLS 采集)
      Pod名称:
              Picked up JAVA_TOOL_OPTIONS: -javaagent:/home/admin/.opt/ArmsAgent/arms-bootstrap-1.7.0-SNAFSHOT.jar -Darms.licenseKey-
               2020-04-10 13:52:33 [INFO ](com.navercorp.pinpoint.bootstrap.PinpointBootStrap) agent custom args: edasAppId:%
               2020-04-10 13:52:33 [WARN ](com.navercorp.pinpoint.bootstrap.AgentDirBaseClassPathResolver) JDK version=1.8.0_191 SLF4J: Failed to load class "org.slf4j.impl.StaticLoggerBinder".
               SIF4]: Befaulting to no-operation (MDP) logger implementation
SIF4J: See http://www.slf4j.org/oodes.html#StaticLoggerBinder for further details.
               2020-04-10 13:52:36 [WARN ](com.navercorp.pinpoint.bootstrap.PinpointBootStrap) arms agent start cost: 3943ms 2020-04-10 13:52:43.117 INFO 6 — [ main] trationDelegate$BeanPostProcessorChecker : Bean 'org.sp.
                                                                             main] trationDelegate$BeanFostProcessorChecker : Bean 'org. springframework. cloud. autoconfigure. ConfigurattiesF
                           main] o.s. cloud context. scope. GenericScope : BeanFactory id=barb7034-
main] trationDelegate$BeanFostProcessorChecker : Bean 'com. aliware. edas. EchoService' of type [org. springtsors
main] trationDelegate$BeanFostProcessorChecker : Bean 'org. springframework. cloud autoconfigure. ConfigurattiesR
               2020-04-10 13:52:47.222 INFO 6 --- [ 2020-04-10 13:52:47.317 INFO 6 --- [
               2020-04-10 13:52:47.611 INFO 6 --- [
                                                                             main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port(s): 18082 (http)
main] o.apache.catalina.core.StandardService : Starting service [Tomcat]
               2020-04-10 13:52:48.738 INFO 6 --- [
               2020-04-10 13:52:48.828 INFO 6 --- [
                                                                             main] org.apache.oatalina.core.StandardEngine : Starting Servlet engine: [Apache Tomoat/9.0.17]
main] o.a.c.c.C.[Tomoat].[localhost].[/] : Initializing Spring embedded WebApplicationContext
                                                                                                                                              Root WebApplicationContext: initialization completed in No URLs will be polled as dynamic configuration sources.
               2020-04-10 13:52:49.294 INFO 6 ---
               2020-04-10 13:52:50.512 WARN 6 ---
                                                                              main] c.n.c. sources. URLConfigurationSource
               2020-04-10 13:52:50.513 INFO 6 --- [
```

• 访问方式:为Consumer绑定SLB,并通过其IP地址进行访问。

在K8s集群中部署应用

使用以下资源来查找部署应用的相关操作教程,以便在EDAS中开始开发应用。建议您使用Chrome浏览器进行控制台操作。

6.3 使用控制台部署

6.3.1 在容器服务K8s集群中使用镜像方式部署应用

EDAS在云原生Kubernetes之上,全面集成了容器服务Kubernetes版,支持K8s容器化应用的全生命周期管理。容器服务K8s集群集成了阿里云虚拟机、存储、网络和安全等能力,拥有云端极佳的K8s容器化应用运行环境。本教程介绍了如何在容器服务K8s集群中使用自定义镜像来部署应用。

前提条件

- 您的阿里云账号已同时开通EDAS和容器服务Kubernetes版,详情请参见:
 - #unique_34
 - #unique 35

文档版本: 20200707 17

- 创建命名空间
- 在容器服务Kubernetes版完成角色授权,详情请参见#unique 26。
- 制作应用容器 Docker 镜像

步骤一: 创建容器服务Kubernetes集群

登录容器服务Kubernetes版控制台,创建容器服务Kubernetes集群,详情请参见#unique_7。

步骤二:在EDAS控制台中导入容器服务K8s集群

- 1. 在左侧导航栏选择**资源管理 > 容器服务K8s集群**。
- 2. 在顶部菜单栏选择地域,选择要导入集群的目标命名空间,然后单击**同步容器服务Kubernetes集**群。
- **3.** 在导入的容器服务Kubernetes集群的**操作**列单击**导入**。 当容器服务Kubernetes集群状态显示为**运行中**,并且**导入状态**显示为**导入成功**时,表示容器服务Kubernetes集群已成功导入到EDAS。

步骤三: 在容器服务Kubernetes版集群中创建应用

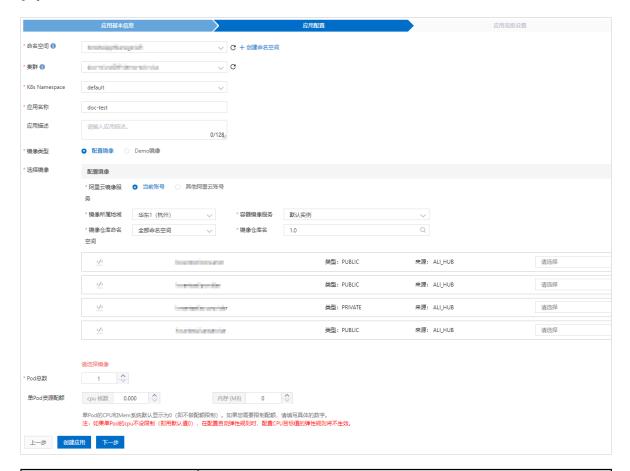
- 1. 登录EDAS控制台。
- **2.** 在左侧导航栏中选择**应用管理 > 应用列表**,在**应用列表**页面左上角单击**创建应用**。
- 3. 在**应用基本信息**页面中设置应用的集群类型和应用运行环境,然后单击**下一步**。



参数	描述
集群类型	选择Kubernetes集群。

参数	描述	
应用运行环境	在本篇文档中,选择 自定义 作为示例。	
	 自定义:适用于在K8s集群中使用自定义的镜像部署应用。 Java:支持通用的JAR包部署,适用于Dubbo和Spring Boot应用。在选择后可更改Java环境。 Tomcat:支持通用的WAR包部署,适用于Dubbo和Spring应用。在选择后可更改Java环境和容器版本。 EDAS-Container (HSF):适用于使用WAR或者FatJar部署HSF应用。在选择后可更改Java环境、Pandora版本和Ali- 	
	Tomcat版本。	

4. 在**应用配置**页签中设置应用的环境信息、基本信息、部署方式和资源参数,设置完成后单击**下一** 步。



参数	描述
命名空间	选择您创建的命名空间,如果您未创建命名空间或不做选择,命名空间则设置为 默认 。
集群	在右侧下拉选择框内选择导入的容器服务K8s集群。

参数	描述	
K8s Namespace	K8s Namespace通过将系统内部的对象分配到不同的Namespace中,形成逻辑上分组的不同项目、小组或用户组,便于不同的分组在共享使用整个集群的资源的同时还能被分别管理。 default:没有其他命名空间的对象的默认命名空间。 kube-system:系统创建的对象的命名空间。 kube-public:此命名空间是自动创建的,并且可供所有用户(包括未经过身份验证的用户)读取。	
应用名称	输入应用名称,必须以字母开头,允许数字、字母、短划 线(-)组合。最多支持输入36个字符。	
应用描述	输入应用描述 ,最大长度为128个字符。	
镜像类型	 配置镜像 阿里云镜像服务选择当前账号。 选择镜像所属地域、容器镜像服务、镜像仓库命名空间、镜像仓库名和镜像版本。 阿里云镜像服务选择为其他阿里云账号。 如您的镜像存放在公开仓库中,那么您配置完整镜像地址即可。 如您的镜像存放在私有仓库中,那么您需要使用免密插件拉取容器镜像,详情请参见#unique_36。 Demo镜像 选择EDAS提供的Demo镜像和镜像版本。 	
Pod总数	设置该应用要部署的Pod个数。	
单Pod资源配额	设置单个Pod的CPU和内存,如果您需要限额,请填写具体的数字,使用默认值0则代表不限额。	

5. (可选)配置应用高级设置。

a) 配置应用实例的调度规则。

参数	描述
默认调度	实例默认按照集群资源情况调度,所有实例有可能部署于单可用区,可用性一般。

参数	描述	
尽量多可用区部署	实例会利用反亲和性,尽可能打散部署到多可用区,可用性高。	

b) 设置启动命令和启动参数。



说明:

若不了解原Dockerfile镜像的CMD和ENTRYPOINT内容,不建议修改自定义启动命令和启动参数,错误的自定义命令将导致应用创建失败。

参数	描述
启动命令	输入启动命令,如命令CMD ["/usr/sbin/sshd","-D"],则在输入框内填写/usr/sbin/sshd -D。

参数	描述
启动参数	每个参数占据一行。如args:["-c"; "while sleep 2"; "do echo date"; "done"]中包含4个参数,需要通过单击 添加 来增加启动参数的输入框将参数分为4行来填写。

c) 设置环境变量。

在创建应用过程中,将所填环境变量注入到即将生成的容器中,这样可以避免常用环境变量的重复添加。

- 如果您需要设置JVM堆内存、JVM属性参数、Java Agent等参数,可以在设置环境变量这一步添加相关参数。
 - 变量名: CATALINA_OPTS。
 - 变量值: [您需要添加的参数] \$(EDAS_CATALINA_OPTS)。



- 如您使用MySQL镜像时,可以参考以下环境变量:
 - MYSQL_ROOT_PASSWORD(必选项):用于设置MySQL的root密码。
 - MYSQL_USER和MYSQL_PASSWORD(可选项):用于添加除root之外的账号并设置密码。
 - MYSQL_DATABASE(可选项):用于设置生成容器时需要新建的数据库。
- 如使用其它类型的镜像,请根据实际需求进行配置。
- d) 设置持久化存储。

在阿里云容器服务Kubernetes集群中,原生的Volume对象所对应的物理储存为非持久化的,它的生命周期与Kubernetes Pod一致,是一个暂态的储存对象。使用持久化的储存服务——文件储存NAS,可以永久保存实例数据,应用在升级或迁移后不丢失。



说明:

设置持久化存储前,请确保您的EDAS账号已开通文件存储NAS。使用NAS请保证当前账户拥有足够的余额或计费方式为后付费,NAS服务的计费规则请参见#unique_37。

参数	描述
存储类型	默认为NAS文件存储,不可配置。
存储服务类型	目前仅支持SSD性能型,不可配置。

参数	描述	
选择NAS	 新购一个NAS:选择NAS挂载目录和本地挂载目录。单一Region内NAS文件系统上限为10个,若已超过10个会创建失败。如有特殊需求,请提交工单。 使用已有NAS:选择一个已有的NAS。您最多可以创建两个挂载点。不符合要求的NAS将不会显示在下拉列表中。 	
挂载目录	设置挂载目录命令。	

e) 设置本地存储。

您可以将宿主机的文件系统的一部分映射到容器内部,以满足您的需求。在使用该功能之前,建议阅读hostpath,并考虑该方案的合理性。

文件类型	Value	含义
默认	空字符串	不执行类型检查,直接挂载。
(新建) 文件目录	DirectoryOrCreate	文件目录,如果不存在,则新建。
文件目录	Directory	文件目录,不存在则容器启动 失败。
(新建) 文件	FileOrCreate	文件,如果不存在,则新建。
文件	File	文件,不存在则容器启动失 败。
Socket	Socket	标准Unix Socket文件,不存 在则容器启动失败。
CharDevice	CharDevice	字符设备文件,不存在则容器 启动失败。
BlockDevice	BlockDevice	块储存设备文件,不存在则容 器启动失败。



说明:

此步骤中,您无需关注Value列的内容。但在应用创建完成后,可能会在API中使用。

f) 设置应用生命周期管理。

参数	描述
PostStart配置	一种容器钩子。该钩子在容器被创建后立刻触发,通知容器它已经被创建。该钩子不需要向其所对应的hook handler传入任何参数。如果该钩子对应的hook handler执行失败,则该容器会终止运行,并根据该容器的重启策略决定是否要重启该容器。详情请参见Container Lifecycle Hooks。
PreStop配置	一种容器钩子。该钩子在容器被删除前触发,其所对应的hook handler必须在删除该容器的请求发送给Docker daemon之前完成。在该钩子对应的hook handler完成后不论执行的结果如何,Docker daemon会发送一个SGTERN信号量给Docker daemon来删除该容器。详情请参见Container Lifecycle Hooks
Liveness配置	一种探测容器状态的探针,探测应用是否处于健康状态。如果不健康,则删除重建容器。详情请参见Pod Lifecycle。
Readiness配置	一种探测容器状态的探针,探测应用是否启动完成并且处于正常服务状态。如果不正常,则更新容器的状态。详情请参见Pod Lifecycle。

g) 设置日志收集。

您可开通日志服务SLS,能将业务文件日志、容器标准输出stdout、stderr日志输出到SLS,方便您无限制行数的查看日志、自聚合分析日志。日志服务按量计费。



参数	描述
日志库名称	输入日志库的名称,如果不存在将新建日志库。
采集日志类型	您可选择 文件日志(容器内日志路径) 或 容器标准输出日志 。
采集路径	输入容器内的日志路径。

6. 完成设置后单击创建应用。

应用创建可能需要几分钟,创建过程中,可以通过变更记录#unique_14跟踪应用的创建流程。创建完成后,返回**应用总览**页面查看实例Pod的运行状态若为**运行中**则说明应用发布成功,单击Pod的运行状态可以查看应用实例的**工作负载、容器组(Pod)**和**启动命令**等高级配置信息。

后续步骤

应用创建完毕后,可以通过添加公网SLB实现公网访问,添加私网SLB实现同VPC内所有节点够能通过私网负载均衡访问您的应用。相关操作请参见为 Kubernetes 集群中的应用添加负载均衡 SLB。

联系我们

如果在使用容器服务Kubernetes集群过程中有任何疑问,欢迎您扫描下面的二维码加入钉钉群进行 反馈。



6.3.2 在容器服务K8s集群中使用JAR包或WAR包部署应用

EDAS在云原生Kubernetes之上,全面集成了容器服务Kubernetes版,支持K8s容器化应用的全生命周期管理。容器服务K8s集群集成了阿里云虚拟机、存储、网络和安全等能力,拥有云端极佳的K8s容器化应用运行环境。本教程介绍了如何在容器服务K8s集群中使用官方Demo的JAR包来部署应用。

背景信息

在容器服务K8s集群中部署应用,您需在容器服务Kubernetes版控制台中创建容器服务Kubernetes版集群,并将该集群导入到EDAS控制台,然后在导入的容器服务K8s集群中使用部署包或镜像来部署应用。



前提条件

- 您的阿里云账号已同时开通EDAS和容器服务Kubernetes版,详情请参见:
 - #unique_34
 - #unique_35
- 创建命名空间

- 在容器服务Kubernetes版完成角色授权,详情请参见#unique_26。
- 制作应用容器 Docker 镜像

步骤一: 创建容器服务Kubernetes集群

登录容器服务Kubernetes版控制台,创建容器服务Kubernetes集群,详情请参见#unique 7。

步骤二:在EDAS控制台中导入容器服务K8s集群

- 1. 登录EDAS控制台。
- 2. 在左侧导航栏选择**资源管理 > 容器服务K8s集群**。
- 3. 在顶部菜单栏选择地域,选择要导入集群的目标命名空间,然后单击**同步容器服务Kubernetes集**群。
- **4.** 在导入的容器服务Kubernetes集群的**操作**列单击**导入**。 当容器服务Kubernetes集群状态显示为**运行中**,并且**导入状态**显示为**导入成功**时,表示容器服 务Kubernetes集群已成功导入到EDAS。

步骤三:在容器服务Kubernetes版集群中部署应用

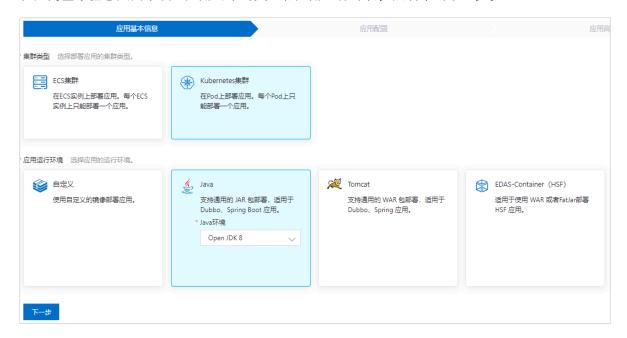


说明:

WAR和JAR的部署步骤一致,本文以通过JAR为例介绍如何部署应用。

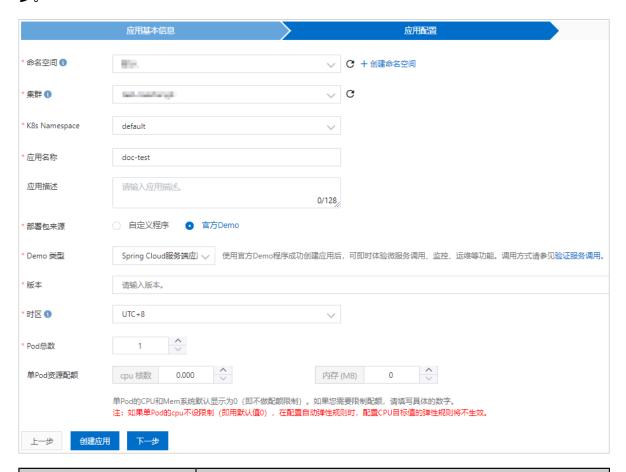
- 1. 登录EDAS控制台。
- 2. 在左侧导航栏中选择**应用管理 > 应用列表**,在**应用列表**页面左上角单击**创建应用**。

3. 在**应用基本信息**页面中设置应用的集群类型和应用运行环境,然后单击**下一步**。



参数	描述
集群类型	选择Kubernetes集群。
应用运行环境	在本篇文档中,选择 Java 作为示例。
	 自定义:适用于在K8s集群中使用自定义的镜像部署应用。 Java:支持通用的JAR包部署,适用于Dubbo和Spring Boot应用。在选择后可更改Java环境。 Tomcat:支持通用的WAR包部署,适用于Dubbo和Spring应用。在选择后可更改Java环境和容器版本。 EDAS-Container (HSF):适用于使用WAR或者FatJar部署HSF应用。在选择后可更改Java环境、Pandora版本和AliTomcat版本。

4. 在**应用配置**页签中设置应用的环境信息、基本信息、部署方式和资源参数,设置完成后单击**下一 步**。



参数	描述
命名空间	选择您创建的命名空间,如果您未创建命名空间或不做选择,命名空间则设置为 默认 。
集群	在右侧下拉选择框内选择导入的容器服务K8s集群。
K8s Namespace	K8s Namespace通过将系统内部的对象分配到不同的Namespace中,形成逻辑上分组的不同项目、小组或用户组,便于不同的分组在共享使用整个集群的资源的同时还能被分别管理。 default:没有其他命名空间的对象的默认命名空间。 kube-system:系统创建的对象的命名空间。 kube-public:此命名空间是自动创建的,并且可供所有用户(包括未经过身份验证的用户)读取。 此处以选择default为例。
应用名称	输入应用名称,必须以字母开头,允许数字、字母、短划 线(-)组合。最多可输入36个字符。
应用描述	输入应用描述 , 最大长度为128个字符。

参数	描述
部署包来源	• 自定义程序
	需选择 文件上传方式 :
	- 上传JAR包 :选择下载好的JAR包并上传。
	- JAR包地址 :输入Demo包地址。
	• 官方Demo
	EDAS提供的Demo类型有 Spring Cloud服务端应用、Spring
	Cloud客户端应用、Dubbo服务端应用和Dubbo客户端应
	用 ,请依据实际需求选择。
版本	请输入版本,您可以自定义版本号也可以使用时间戳作为版本号。
时区	设置应用的时区信息。
Pod总数	设置该应用要部署的Pod个数。
单Pod资源配额	设置单个Pod的CPU和内存,如果您需要限额,请填写具体的数字,使用默认值0则代表不限额。

- 5. (可选)配置应用高级设置。
 - a) 配置应用实例的调度规则。

参数	描述
默认调度	实例默认按照集群资源情况调度,所有实例有可能部署于单可用区,可用性一般。
尽量多可用区部署	实例会利用反亲和性,尽可能打散部署到多可用区,可用性高。

b) 设置启动命令和启动参数。



说明

若不了解原Dockerfile镜像的CMD和ENTRYPOINT内容,不建议修改自定义启动命令和启动参数,错误的自定义命令将导致应用创建失败。

参数	描述
启动命令	输入启动命令,如命令CMD ["/usr/sbin/sshd","-D"],则在输入 框内填写/usr/sbin/sshd -D。

参数	描述
启动参数	每个参数占据一行。如args:["-c"; "while sleep 2"; "do echo date"; "done"]中包含4个参数,需要通过单击 添加 来增加启动参数的输入框将参数分为4行来填写。

c) 设置环境变量。

在创建应用过程中,将所填环境变量注入到即将生成的容器中,这样可以避免常用环境变量的重复添加。

- 如果您需要设置JVM堆内存、JVM属性参数、Java Agent等参数,可以在设置环境变量这一步添加相关参数。
 - 变量名: CATALINA_OPTS。
 - 变量值: [您需要添加的参数] \$(EDAS_CATALINA_OPTS)。



- 如您使用MySQL镜像时,可以参考以下环境变量:
 - MYSQL_ROOT_PASSWORD(必选项):用于设置MySQL的root密码。
 - MYSQL_USER和MYSQL_PASSWORD(可选项):用于添加除root之外的账号并设置密码。
 - MYSQL_DATABASE(可选项):用于设置生成容器时需要新建的数据库。
- 如使用其它类型的镜像,请根据实际需求进行配置。
- d) 设置持久化存储。

在阿里云容器服务Kubernetes集群中,原生的Volume对象所对应的物理储存为非持久化的,它的生命周期与Kubernetes Pod一致,是一个暂态的储存对象。使用持久化的储存服务——文件储存NAS,可以永久保存实例数据,应用在升级或迁移后不丢失。



说明:

设置持久化存储前,请确保您的EDAS账号已开通文件存储NAS。使用NAS请保证当前账户拥有足够的余额或计费方式为后付费,NAS服务的计费规则请参见#unique_37。

参数	描述
存储类型	默认为NAS文件存储,不可配置。
存储服务类型	目前仅支持SSD性能型,不可配置。

参数	描述
选择NAS	 新购一个NAS:选择NAS挂载目录和本地挂载目录。单一Region内NAS文件系统上限为10个,若已超过10个会创建失败。如有特殊需求,请提交工单。 使用已有NAS:选择一个已有的NAS。您最多可以创建两个挂载点。不符合要求的NAS将不会显示在下拉列表中。
挂载目录	设置挂载目录命令。

e) 设置本地存储。

您可以将宿主机的文件系统的一部分映射到容器内部,以满足您的需求。在使用该功能之前,建议阅读hostpath,并考虑该方案的合理性。

文件类型	Value	含义
默认	空字符串	不执行类型检查,直接挂载。
(新建) 文件目录	DirectoryOrCreate	文件目录,如果不存在,则新建。
文件目录	Directory	文件目录,不存在则容器启动 失败。
(新建) 文件	FileOrCreate	文件,如果不存在,则新建。
文件	File	文件,不存在则容器启动失 败。
Socket	Socket	标准Unix Socket文件,不存 在则容器启动失败。
CharDevice	CharDevice	字符设备文件,不存在则容器 启动失败。
BlockDevice	BlockDevice	块储存设备文件,不存在则容 器启动失败。



说明:

此步骤中,您无需关注Value列的内容。但在应用创建完成后,可能会在API中使用。

f) 设置应用生命周期管理。

参数	描述
PostStart配置	一种容器钩子。该钩子在容器被创建后立刻触发,通知容器它已经被创建。该钩子不需要向其所对应的hook handler传入任何参数。如果该钩子对应的hook handler执行失败,则该容器会终止运行,并根据该容器的重启策略决定是否要重启该容器。详情请参见Container Lifecycle Hooks。
PreStop配置	一种容器钩子。该钩子在容器被删除前触发,其所对应的hook handler必须在删除该容器的请求发送给Docker daemon之前完成。在该钩子对应的hook handler完成后不论执行的结果如何,Docker daemon会发送一个SGTERN信号量给Docker daemon来删除该容器。详情请参见Container Lifecycle Hooks
Liveness配置	一种探测容器状态的探针,探测应用是否处于健康状态。如果不健康,则删除重建容器。详情请参见Pod Lifecycle。
Readiness配置	一种探测容器状态的探针,探测应用是否启动完成并且处于正常服务状态。如果不正常,则更新容器的状态。详情请参见Pod Lifecycle。

g) 设置日志收集。

您可开通日志服务SLS,能将业务文件日志、容器标准输出stdout、stderr日志输出到SLS,方便您无限制行数的查看日志、自聚合分析日志。日志服务按量计费。



参数	描述
日志库名称	输入日志库的名称,如果不存在将新建日志库。
采集日志类型	您可选择 文件日志(容器内日志路径) 或 容器标准输出日志 。
采集路径	输入容器内的日志路径。

6. 完成设置后单击创建应用。

应用创建可能需要几分钟,创建过程中,可以通过变更记录#unique_14跟踪应用的创建流程。创建完成后,返回**应用总览**页面查看实例Pod的运行状态若为**运行中**则说明应用发布成功,单击Pod的运行状态可以查看应用实例的**工作负载、容器组(Pod)**和**启动命令**等高级配置信息。

后续步骤

应用创建完毕后,可以通过添加公网SLB实现公网访问,添加私网SLB实现同VPC内所有节点够能通过私网负载均衡访问您的应用。相关操作请参见为 Kubernetes 集群中的应用添加负载均衡 SLB。

联系我们

如果在使用容器服务Kubernetes集群过程中有任何疑问,欢迎您扫描下面的二维码加入钉钉群进行 反馈。



6.4 使用工具部署

6.4.1 使用toolkit-maven-plugin插件部署应用到容器服务K8s集群

在容器服务K8s集群中,基于HSF、Dubbo或Spring Cloud框架开发的应用均可以通过toolkitmaven-plugin插件来自动化部署。

自动化部署

通过toolkit-maven-plugin插件自动化部署应用的流程:添加插件依赖,配置插件,构建部署。

1. 添加插件依赖

在pom.xml文件中增加如下所示的插件依赖。

</build>



说明:

toolkit-maven-plugin建议使用最近版本。

2. 配置插件

配置插件主要包含账号配置,打包配置及部署配置。如果需要更多自定义配置项可参考打包参数和部署参数进行设置。

a) 账号配置

在打包工程的根目录下创建文件格式为YAML的账号配置文件,命名为toolkit_profile.yaml并填入如下信息:

regionId: #应用所在区域,如北京为`cn-beijing`,上海为`cn-shanghai`,杭州为`cn-hangzhou`。

jarPath: #部署包路径,配置该参数可忽略Maven打包流程从而直接使用配置路径内的

包部署,您可配置绝对路径和相对路径。

accessKeyId: #访问阿里云资源的AK,建议使用子账号的AK降低安全风险。 accessKeySecret: #访问阿里云资源的SK,建议使用子账号的SK降低安全风险。

b) 打包配置

在打包工程的根目录下创建文件格式为YAML的打包配置文件。如果打包工程为Maven的子模块,则需要在子模块的目录下创建该文件,并命名为toolkit_package.yaml,填入如下信息:

apiVersion: V1 kind: AppPackage

spec:

packageType: #应用部署包类型,支持War、FatJar、Image、url; 您只有在该处配置了url, 那么packageUrl才能生效。packageUrl: #如果应用部署包类型为War或FatJar, 可填入此字段,不填则使用当前maven构建的包进行部署。imageUrl: #如果部署包类型为Image,可填入此字段; Image类型也可以在本地构建Docker镜像进行部署,请参考打包文件参数章节设置相关参数。

c) 部署配置

在打包工程的根目录下创建文件格式为YAML的部署文件,命名为toolkit_deploy.yaml,并填入如下信息:

apiVersion: V1

kind: AppDeployment

spec:

type: kubernetes

target:

appld: #部署应用的ID,如果配置了appld则无需配置namespaceld和appName namespaceld: #所属区域,如不清楚appld,可使用此所属区域及应用名称进行部署 appName: #应用名称,如不清楚appld,可使用此应用名称及命名空间进行部署 und staffrate with

updateStrategy: type: GrayBatchUpdate #灰度分批发布

grayUpdate: #灰度设置 gray: x #灰度机器数量 batchUpdate: #分批设置

batch: x #批次数量

releaseType: xxx #发布方式: auto表示自动发布, manual表示手动发布

batchWaitTime: x #批次间隔时间

3. 构建部署

进入pom.xml所在的目录(如果部署Maven子模块,则进入子模块pom.xml所在的目录),执行如下命令:

mvn clean package toolkit:deploy -Dtoolkit_profile=toolkit_profile.yaml -Dtoolkit_p ackage=toolkit_package.yaml -Dtoolkit_deploy=toolkit_deploy.yaml

命令参数含义为:

• toolkit:deploy: 在打包完成后进行应用部署。

- -Dtoolkit_profile:指定账号配置文件。如果账号文件跟pom.xml在同一个目录下,且名字为. toolkit_profile.yaml(注意:文件名最前面有个小数点),可不填此参数,插件会自动获取。
- -Dtoolkit_package: 指定打包文件。如果打包文件跟pom.xml在同一个目录下,且名字为. toolkit_package.yaml(注意:文件名最前面有个小数点),可不填此参数,插件会自动获取。
- Dtoolkit_deploy: 指定部署文件。如果部署文件跟pom.xml在同一个目录下,且名字为.
 toolkit_deploy.yaml(注意:文件名最前面有个小数点),可不填此参数,插件会自动获取取。
- -Ddeploy version: 指定部署的版本号,优先级高于部署配置文件中的version配置。



说明:

toolkit-maven-plugin插件1.0.6及以后版本支持配置该参数。

执行该打包命令后,系统显示如下结果,当回显信息中显示"BUDILD SUCCESS"表示部署成功。

```
[INFO] Sending deploy request to EDAS...
[INFO] Deploy request sent, changeOrderId is: 71548877-5bb3-4503-88e5-04761c0ac688
[INFO] Begin to trace change order: 71548877-5bb3-4503-88e5-04761c0ac688
[INFO] PipelineName:Batch: 1, PipelineId:9d7b8038-534c-4891-8f70-e63db6606297
[INFO] StageName:Workflow Start, StageId:cda21377-260d-43f0-9c82-98b55c3fd1ec
[INFO] ServiceStageName:Workflow Start, ServiceStageId:cda21377-260d-43f0-9c82-98b55c3fd1ec
[INFO] StageName:SLB Offline, StageId:1060b156-38b3-47aa-990d-5b5925256a57
[INFO] ServiceStageName:SLB Offline, ServiceStageId:1060b156-38b3-47aa-990d-5b5925256a57
[INFO] StageName:Deploy, StageId:035c7222-b84a-4c3a-bec2-3aeb3a5ac90e@co_virtual_stage
[INFO] InstanceName:nahai-20180920, InstanceIp:47.93.157.199(Public)<br/>
<br/>
br>10.30.137.182(Inner)
[INFO] InstanceStageName:RPC Offline, InstanceStageId:3a0fbf0b-1dda-4368-8b67-b2e8bf7c7de7
[INFO] Waiting...
[INFO] Waiting...
[INFO] Waiting...
[INFO] Waiting...
[INFO] Waiting...
[INFO] Waiting...
[INFO] InstanceStageName:Stop Application, InstanceStageId:f8df6de7-adea-47c0-a2cd-6a18b49ebb2f
[INFO] InstanceStageName:Environment initial, InstanceStageId:033ca419-7d0e-4996-ba91-0b4e67877548
[INFO] InstanceStageName:Start Application, InstanceStageId:634e45bc-a411-429d-a549-b693794a41ac
[INFO] Waiting...
[INFO] Waiting...
[INFO] InstanceStageName:Health Check, InstanceStageId:77f179ee-f8ac-40ad-85a5-86cad41b1e4a
[INFO] Waiting...
[INFO] Waiting...
[INFO] Waiting...
[INFO] Waiting...
[INFO] StageName:SLB Online, StageId:035c7222-b84a-4c3a-bec2-3aeb3a5ac90e
[INFO] ServiceStageName:SLB Online, ServiceStageId:035c7222-b84a-4c3a-bec2-3aeb3a5ac90e
[INFO] StageName: Workflow Complete, StageId:510d5110-d430-4508-93c4-864dc506b9ae
[INFO] ServiceStageName:Workflow Complete, ServiceStageId:510d5110-d430-4508-93c4-864dc506b9ae
[INFO] Deploy application successfully!
[INFO] -----
[INFO] BUILD SUCCESS
```

更多配置项

1. 打包参数配置

打包文件支持的参数如下所示。

```
apiVersion: V1 kind: AppPackage spec: packageType: #应用部署包类型,支持War、FatJar、Image、url; 您只有在该处配置 了url, 那么packageUrl才能生效。imageUrl: #镜像地址,Image包类型应用可填入。packageUrl: #包地址,War、FatJar类型应用可填入。build: docker: dockerfile: #Docker镜像构建文件。如您希望在本地构建镜像部署,需填入此字段。imageRepoAddress: #阿里云镜像仓库地址。如您希望在本地构建镜像部署,需填入此字段。imageRepoUser: #镜像Tag。如您希望在本地构建镜像部署,需填入此字段。imageRepoUser: #阿里云镜像仓库用户名。如您希望在本地构建镜像部署,需填入此字段。imageRepoPassword: #阿里云镜像仓库密码。如您希望在本地构建镜像部署,需填入此字段。imageRepoPassword: #阿里云镜像仓库密码。如您希望在本地构建镜像部署,需填入此字段。oss: bucket: #目标存储桶名称。如您希望使用自定义的oss仓库存储部署包,需填入此字段。key: #oss自定义路径。如您希望使用自定义的oss仓库存储部署包,需填入此字段。key: #oss自定义路径。如您希望使用自定义的oss仓库存储部署包,需填入此
```

字段。accessKeyId: #oss账号。如您希望使用自定义的oss仓库存储包,需填入此字段。accessKeySecret: #oss密码。如您希望使用自定义的oss仓库存储包,可填入此字段。

2. 部署参数

部署文件支持的参数如下所示。

```
apiVersion: V1
kind: AppDeployment
type: kubernetes
target:
 appName: #应用名称namespaceld: #应用所在命名空间appld:
                                                #应用ID。插件
会使用此应用进行部署,如未填入则使用namespaceld和appName来查找应用进行部署。
version: #部署版本号,默认使用日时分秒格式jdk: #部署的包依赖的JDK版本,JDK支持版
本为Open JDK 7和Open JDK 8。镜像不支持。webContainer: #部署的包依赖的Tomcat版
本, WebContainer支持apache-tomcat-7.0.91。镜像不支持。batchWaitTime: #分批等
待时间。command: #镜像启动命令。该命令必须为容器内存在的可执行的对象。例如:
sleep。设置该命令将导致镜像原本的启动命令失效。commandArgs: #镜像启动命令参数。
上述启动命令所需参数。
 - 1d
envs: #容器环境变量参数
 - name: envtmp0
  value: '0'
 - name: envtmp1
  value: '1'
liveness: #容器健康检查,健康检查失败的容器将被杀死并恢复。exec:
  command:
   - sleep
   - 1s
 initialDelaySeconds: 5
 timeoutSeconds: 11
readiness: #应用启动状态检查, 多次健康检查失败的容器将被杀死并重启。不通过健康检
查的容器将不会有SLB流量进入。exec:
  command:
   - sleep
   - 1s
 initialDelaySeconds: 5
 timeoutSeconds: 11
```

典型场景示例

典型部署场景及相关配置示例。

• 场景一: 本地构建War (或FatJar) 包进行部署

假设您在北京环境有WAR(或FatJar)类型的EDAS应用,期望本地构建WAR(或FatJar)进行部署。打包配置和部署配置如下所示。

- 打包文件:

apiVersion: V1 kind: AppPackage spec:

packageType: War

- 部署文件:

apiVersion: V1

kind: AppDeployment

spec:

type: kubernetes

target:

appld: #应用ID。插件会使用此应用进行部署,如未填入则使用namespaceId和appName来查找应用进行部署。namespaceId: #【可选】命名空间,如不清楚appld,可使用此命名空间及应用名称进行部署appName: #【可选】应用名称,如不清楚

appld,可使用此命名空间及应用名称进行部署

• 场景二:使用已有镜像地址部署镜像类型应用

假设您在北京环境有一个镜像类型应用,期望使用已有的镜像(registry.cn-beijing.aliyuncs.com/test/gateway:latest) 部署应用。打包配置和部署配置如下所示。

- 打包文件:

apiVersion: V1 kind: AppPackage

spec:

packageType: Image

imageUrl: registry.cn-beijing.aliyuncs.com/test/gateway:latest

- 部署文件:

apiVersion: V1

kind: AppDeployment

spec:

type: kubernetes

target:

appld: #应用ID。插件会使用此应用进行部署,如未填入则使用namespaceld和appName来查找应用进行部署。namespaceld: #【可选】命名空间,如不清楚appld,可使用此命名空间及应用名称进行部署appName: #【可选】应用名称,如不清楚

appld,可使用此命名空间及应用名称进行部署

• 场景三: 本地构建镜像上传至仓库并部署应用

假设您在北京环境有镜像类型应用,期望在本地编译并构建为镜像,并上传到阿里云镜像仓库进行部署,打包配置和部署配置如下所示。

- 打包文件:

apiVersion: V1 kind: AppPackage

spec:

packageType: Image

build: docker:

dockerfile: Dockerfile #指定Dockerfile

imageRepoAddress: #镜像仓库地址imageTag: #镜像Tag

imageRepoUser: #镜像仓库用户名imageRepoPassword: #镜像仓库密码

- 部署文件:

apiVersion: V1

kind: AppDeployment

spec:

type: kubernetes

target:

appld: #应用ID。插件会使用此应用进行部署,如未填入则使用namespaceld和appName来查找应用进行部署。namespaceld: #【可选】命名空间,如不清楚appld,可使用此命名空间及应用名称进行部署appName: #【可选】应用名称,如不清楚appld,可使用此命名空间及应用名称进行部署

问题反馈

如果您在使用工具部署应用的过程中有任何疑问,欢迎您扫描下面的二维码加入钉钉群进行反馈。



6.4.2 使用 Cloud Toolkit 插件快速部署应用到容器服务 K8s 集群

您在本地完成应用的开发、调试和测试后,可以在 IntelliJ IDEA 中安装 Cloud Toolkit 插件实现 EDAS 应用的快速部署。本文档将介绍如何在 IntelliJ IDEA 中安装 Cloud Toolkit,并快速部署应用到容器服务 K8s 集群。

前提条件

- 下载并安装 JDK 1.8 或更高版本。
- 下载并安装 Intellij IDEA (2018.3 或更高版本)。



说明:

因 JetBrains 插件市场官方服务器在海外,如遇访问缓慢无法下载安装的,请加入文末交流群,向 Cloud Toolkit 技术支持人员获取离线包安装。

• 在容器服务 K8S 集群中创建了应用并完成了首次部署,相关操作请参见在容器服务K8S集群中使用镜像方式部署应用。

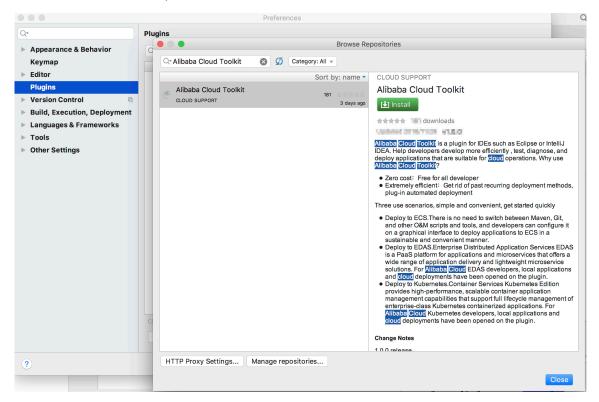
步骤一:安装 Cloud Toolkit

1. 启动 IntelliJ IDEA。

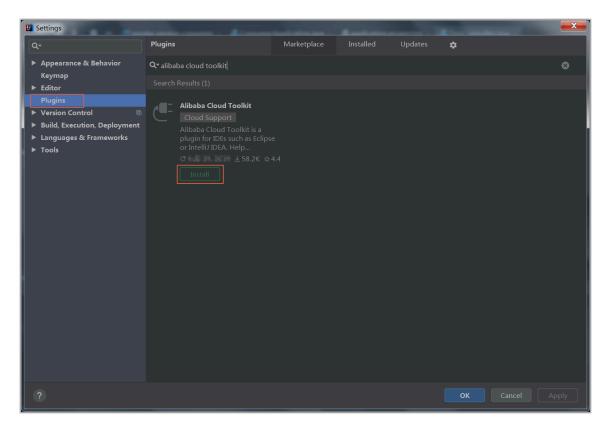
2. 在 Intellij IDEA 中安装最新版本的插件。

如您之前已安装过旧版安装包,请升级至最新版本。

Mac 系统: 进入 Preference 配置页面,选择左边的 Plugins,在右边的搜索框里输入
 Alibaba Cloud Toolkit,并单击 Install安装。



Windows 系统: 在左侧导航栏选择 Plugins, 搜索 Alibaba Cloud Toolkit, 并单击 Install 安装。



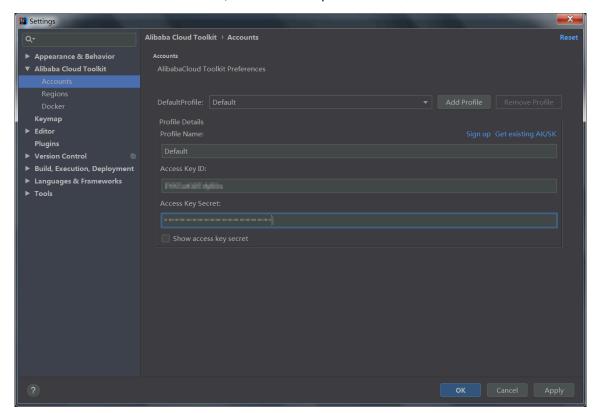
3. 在 Intellij IDEA 中插件安装成功后,重启 Intellij IDEA,您可以在工具栏看到 Alibaba Cloud Toolkit 的图标([一])。

步骤二:配置 Cloud Toolkit 账号

在安装完 Alibaba Cloud Toolkit 后,您需使用 Access Key ID 和 Access Key Secret 来配置 Cloud Toolkit 的账号。

- 1. 启动 IntelliJ IDEA。
- **2.** 单击 Alibaba Cloud Toolkit 的图标(____),在下拉列表中单击 **Preference...**,进入设置页面,在左侧导航栏选择 **Alibaba Cloud Toolkit** > **Accounts**。

- 3. 在 Accounts 界面中设置 Access Key ID和 Access Key Secret, 然后单击 OK。
 - 如果您使用**子账号**的 Access Key ID 和 Access Key Secret,请确认该子账号至少拥有**部署应 用**的权限,具体操作方法请参见常见问题中的应用列表获取不到应用。
 - 如果您是 EDAS 专有云企业版用户,还需配置 Endpoint 才能正常使用 Cloud Toolkit 功能。



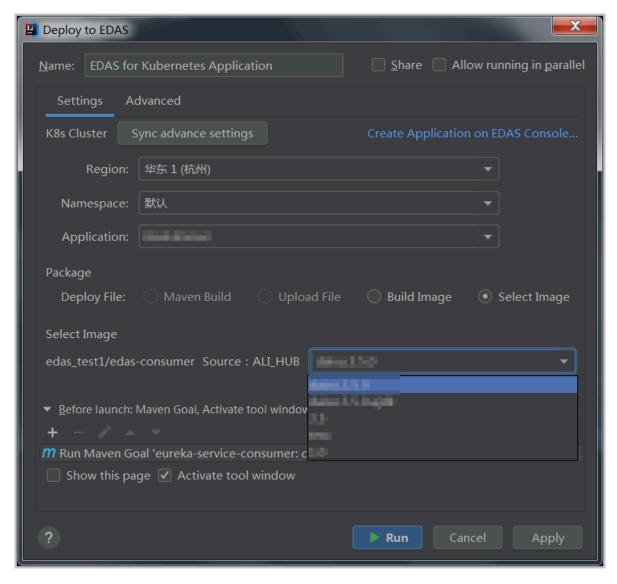
- 如果您已经注册过阿里云账号,在 Accounts 界面中单击 Get existing AK/SK, 进入阿里云登录页面。用已有账号登录后,跳转至安全信息管理页面,获取 Access Key ID 和 Access Key Secret。
- 如果您还没有阿里云账号,在 Accounts 界面中单击 Sign up,进入阿里云账号注册页面,注册账号。注册完成后按照上述方式获取 Access Key ID 和 Access Key Secret。

步骤三: 部署应用到容器服务 K8s 集群

1. 在 IntelliJ IDEA 上单击 Cloud Toolkit 的图标(____) ,在下拉列表中选择**Deploy to EDAS** > EDAS for Kubernetes Application 。

您也可以在 Intellij IDEA 上打开 Run/Debug configurations 对话框,展开 Templates选项,选择 EDAS on Alibaba Cloud > EDAS for Kubernetes Application。

2. 在 Deploy to EDAS 对话框配置应用部署参数。



a) 在配置页面中根据您的实际需求选择应用的 Region、Namespace 和 Application。



注意:

如果在应用列表中获取不到应用,请参见常见问题中的应用列表获取不到应用进行操作排查。

• Region:应用所在地域。

• Namespace: 应用所在命名空间。

• Application: 应用名称。

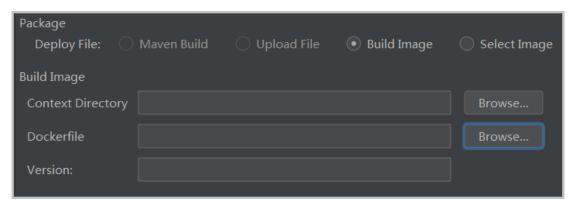
- b) 设置部署方式,可选的部署方式由您在控制台第一次部署的方式决定。
 - Maven Build: 选择 Maven Build 方式来构建应用时,系统会默认添加一个 Maven 任务来构建部署包。



设品。

如果您想部署多模块工程,请参见部署多模块工程。

- Upload File: 选择 Upload File 方式来构建应用时,选择上传您的 WAR 包或者 JAR 包,然后进行部署。
- Build Image: 选择 Build Image 方式来构建应用时,需要依次输入工作目录 Content
 Directory,文件路径 Dockerfile 和镜像标签 Version。



- Select Image: 选择 Build Image 方式来构建应用时,可在 Select Image 的下拉列表选择需要部署的镜像。
- c) 单击 Advanced 进入高级部署参数设置页签,并配置 Application Environment、Startup Command、Environments、Persistence、Storage 和 Appcalition Management 等参数。
- 3. 单击 Run, Intellij IDEA 的 Console 区域会打印部署日志。您可以根据日志信息检查部署结果。

部署多模块工程

实际工作中碰到的大部分 Maven 工程都是多模块的,各个项目模块可以独立开发,其中某些模块又可能会使用到其他的一些模块的功能,这样的项目工程就是多模块工程。

如果您的工程项目为 Maven 多模块工程并且想部署工程中的某子模块,那么需要保证 **EDAS Deployment Configurations** 页面中的 **Before launch** 中的 Maven 构建任务中最后一个任务为该
子模块的构建任务。管理 Maven 构建任务的具体操作请参见Maven 构建任务的管理。

例如:一个 CarShop 工程存在以下示例的子模块。

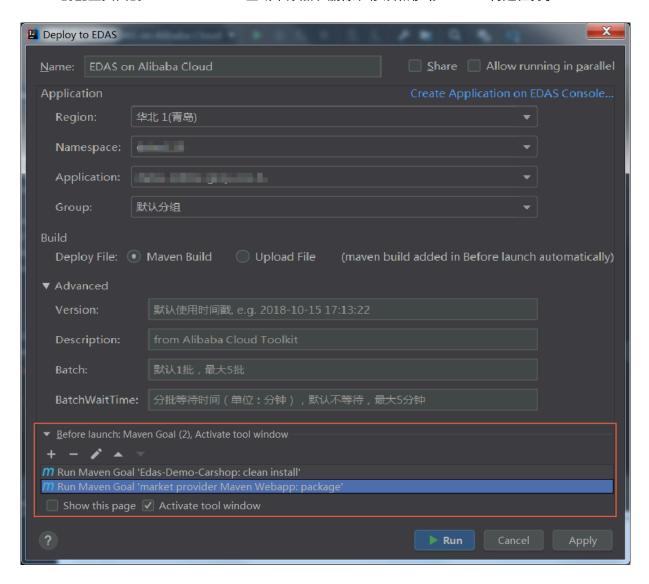
```
carshop
|---- itemcenter-api
|---- itemcenter
|---- detail
```

其中 itemcenter 和 detail 为子模块,且都依赖于 itemcenter-api 模块,现在想部署 itemcenter 模块,应该怎么做?只需要在配置页面中的 **Before launch** 中增加如下两个 Maven 任务。

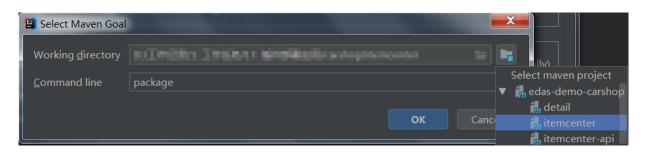
- 增加一个在父工程 carshop 中执行 mvn clean install 的 Maven 任务。
- 增加一个在子模块 itemcenter 中执行 mvn clean package Maven 任务。

Maven 构建任务的管理

在 IntelliJ IDEA 中安装的 Cloud Toolkit 内可以部署 Maven 的构建任务。您也可以在 **Deploy to EDAS** 的配置页面的 **Before launch** 区域来添加、删除、修改和移动 Maven 构建任务。



在添加 Maven 构建任务编辑框中,您可以单击右侧的文件夹按钮选择当前工程的所有可用模块,并在 Command line 中编辑构建命令。



常见问题

• 应用列表获取不到应用

通常出现这种情况为使用子账号来部署应用,且子账号没有同步到 EDAS 系统或者没有进行正确授权,从而导致在应用列表下拉框中看不到应用。您可以通过 RAM 授权或 EDAS 子账号授权来确保子账号已经同步到 EDAS 并且得到授权。

- RAM 授权

该授权方式可使子账号访问 EDAS 的所有资源。

- 1. 在 RAM 控制台左侧导航栏中选择人员管理 > 用户。
- 2. 在用户页面上找到需要授权的子用户,单击操作列中的添加权限。
- 3. 在添加权限面板的选择权限区域中,搜索 AliyunEDASFullAccess 权限,单击权限策略将 其添加至右侧的已选择列表中,然后单击确定。
- 4. 在添加权限的授权结果页面上,查看授权信息摘要,并单击完成。
- **5.** 使用主账号登录 EDAS 控制台,在左侧导航栏选择**系统管理 > 子账号**,单击**子账号**页面右上角的**同步子账号**。
- EDAS 子账号授权

该授权方式可使子账号细粒度授权访问 EDAS 的资源。

- 1. 使用主账号登录 EDAS 控制台。
- 2. 在左侧导航栏选择系统管理 > 角色 , 单击角色页面右上角的创建角色。
- 3. 输入一个角色名称,在可选权限区域框中,选择应用管理 > 应用列表 > 基本信息 > 部署应用,单击添加将部署应用角色添加到已选权限,然后单击确定。
- 4. 在左侧导航栏选择系统管理 > 子账号, 单击子账号页面右上角的同步子账号。
- **5.** 选择需进行授权的子账号,在**操作**列单击**管理角色**,在左侧穿梭狂中搜索并选择上面创建的 角色,将该角色添加到右侧已选角色列表中,然后单击**确定**。
- **6.** 选择需进行授权的子账号,在**操作**列单击**授权应用**,选择应用添加到右侧列表进行授权,然后单击**确定**。
- 配置 Endpoint



说明:

如果使用 EDAS 专有云企业版,还需要按以下步骤在 Cloud Toolkit 中配置 Endpoint。Endpoint 请联系 EDAS 技术支持获取。

- 1. 在 Preference (Filtered) 对话框的左侧导航栏中选择 Appearance & BehaviorEndpoint。
- 2. 在 Endpoint 界面中设置 Endpoint,配置完成后,单击 Apply and Close。

问题反馈

如果您在使用工具部署应用的过程中有任何疑问,欢迎您扫描下面的二维码加入钉钉群进行反馈。



6.4.3 使用云效部署Java应用至EDAS(K8s集群)

本文以Java应用为例,指导您如何使用云效将应用以镜像的方式部署至EDAS。

前提条件

- 注册阿里云账号。
- 将业务代码上传至阿里云Code。



说明:

业务代码中须包含应用的Dockerfile文件,具体制作步骤请参见制作应用容器 Docker 镜像。

- 将镜像文件上传至阿里镜像库。
- 开通云效。

部署流程



- **1.** 如果您的应用已经部署在EDAS上,则无需创建新的应用。如果第一次部署应用到EDAS,则需要在EDAS控制台创建应用,以便云效将业务代码推送到该应用中。
- **2.** 在EDAS创建应用完成后,需要在云效控制台创建对应的应用,应用的部署方式(例如JAR、WAR和镜像)两边需要一致。

- **3.** 在云效控制台完成应用创建后,需要使用云效提供的流水线模板创建出应用集成发布的基础流水 线。
- **4.** 基础流水线创建完成后,需要更改流水线中的构建任务。例如:如果您在EDAS控制台使用镜像方式将应用部署式,那么在该环节您需要将原流水线**构建**任务删掉,添加为供镜像方式部署使用的**Java镜像构建**。整个流水线运行后,本环节的构建产物将供部署环节发布使用。
- 5. 构建任务配置完成后,需要指定上环节(即构建任务环节)构建产物和应用的部署目的地,即应用在EDAS上的区域和应用名称。整个流水线运行成功后,应用成功部署到EDAS上。

步骤一:在EDAS上创建应用

如果您第一次使用EDAS托管应用,需要预先在EDAS上使用Demo应用创建相应的应用。

本文以镜像部署为例,具体操作请参见在容器服务K8s集群中使用镜像方式部署应用。



注意:

在EDAS控制台创建应用时使用的部署方式(JAR、WAR和镜像),必须与在云效的流水线设置保持一致。简而言之,在EDAS上使用镜像方式部署,那么在云效设置流水线时,构建环节必须是镜像相关配置。



步骤二:在云效配置应用基本信息

1. 登录云效控制台应用列表页面。

2. 在菜单栏中选择**项目 > 项目列表**,并在**项目列表**页面单击右上角的**新建项目**,然后在弹出的页面中设置项目信息,并单击**确定**。

项目信息有项目类型、项目名称、权限和项目背景等,请依据实际情况设置。



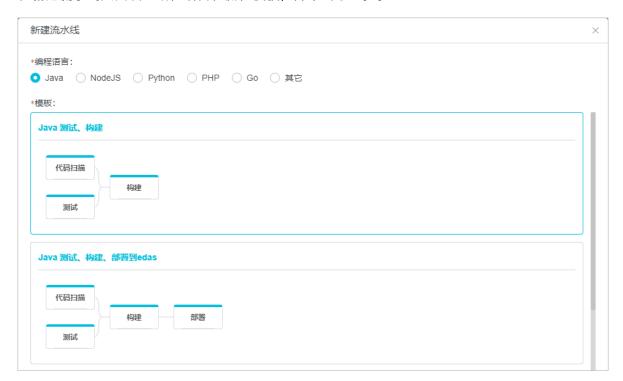
3. 在菜单栏中选择**研发 > 应用**,并在**我的应用**页面中,单击左上角的**创建新应用**,然后在**创建新应 用**设置应用基本信息并单击**创建**。



步骤三: 在云效使用模板构建基础流水线

1. 在菜单栏中选择研发 > 流水线,在流水线页面单击右上角的新建流水线。

2. 在新建流水线页面设置编程语言和流程模板,并单击下一步。



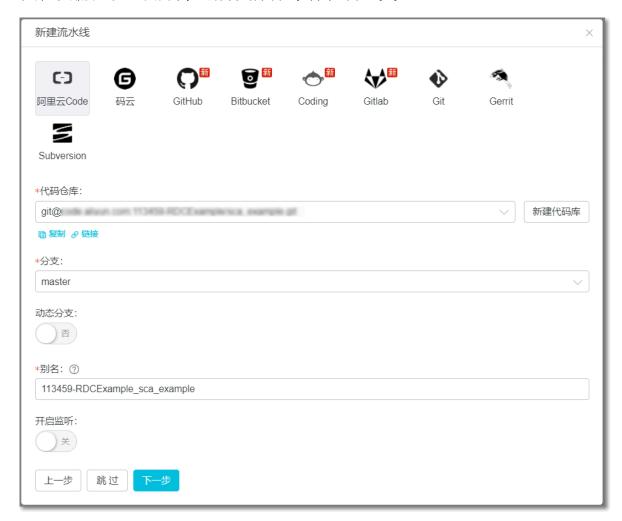
- 编程语言: 选择Java。
- 模板: 选择Java测试、构建、部署EDAS。



说明:

如果您的应用编程语言非Java,请根据实际情况选择。

3. 在弹出的流水线配置页面中,进行代码库设置,并单击**下一步**。



• 代码库类型:选择**阿里云Code**。

• 代码仓库:选择您的代码仓库地址。

• 分支: 选择代码分支。

• 别名:代码的别名,后续用于执行组件代码克隆路径,使用数字、字母或下划线。

4. 在弹出的流水线配置页面中,设置流水线的基本信息如流水线名称、管理员等,并单击**创建**。



步骤四: 在云效中配置构建任务

1. 在编辑流水线页面删除流水线中得构建和部署。



分别单击**构建**和**部署**,在**阶段:构建**页面单击 📊 图标。



2. 在编辑流水线页面单击 图标,并在弹出的阶段模板中选择Java镜像构建。





对于JAVA应用,EDAS支持镜像、JAR包和WAR包三种方式部署。本例以镜像方式为例。如果您需要采用JAR包或者WAR包方式,须在步骤一中以JAR和WAR包方式创建应用,并且本步骤中构建模板选择**Java JAR/WAR包构建**。



- 3. 配置构建任务的基本信息。
 - a) 在弹出的**阶段: 新增阶段**面板中设置**阶段名称**和**流转配置**。



- 4. 配置构建任务的执行步骤。
 - a) 单击**任务列表**中具体任务名称。
 - b) 在**任务: Java构建Docker镜像并推送镜像仓**页面的**Docker镜像构建上传**区域,设置镜像信息。

56



- 步骤名称:保持默认,也可以自定义。
- 区域: Docker镜像文件所在的地域
- 仓库: Docker镜像文件所在地址。
- 标签: Docker镜像Tag,支持固定参数例如1.0,或者动态参数\${DATETIME}。
- **Dockerfile路径**: Dockerfile相对于代码库根目录所在路径,如META/config/Dockerfile或Dockerfile。该文件云效会自动为您创建,无需自建。
- **ContextPath**:填写相对于代码根目录的路径,如target,如果不填则为Dockerfile文件所在目录。
- 组件出参: 为本构建环节产生的结果文件, 供部署环节使用。

步骤五:在云效部署应用至EDAS

步骤一所创建应用不包含您的任何业务代码,在本环节将应用代码推送至EDAS上。即将EDAS上的应用升级为含有您业务代码的应用。

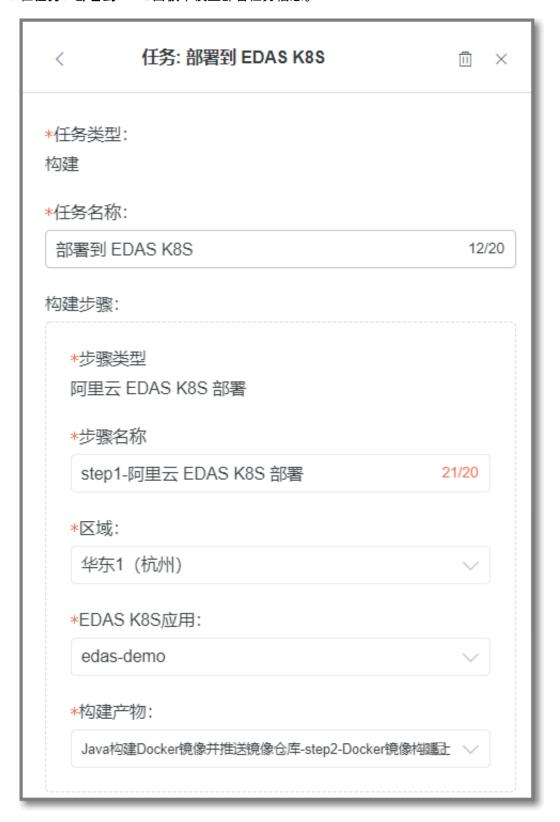
- 1. 在编辑流水线页面的阶段区域中,单击 图标。
- 2. 在弹出的阶段模板面中选择部署到EDAS K8s。



3. 单击新增的流水线任务,并设置基本信息,设置完成后单击部署到EDAS。



4. 在任务: 部署到EDAS面板中设置部署任务信息。



• 区域:选择步骤一中所创建的应用所在区域。

• EDAS K8s应用:选择步骤一中所创建应用的应用名称。

• 构出产物: 为步骤四中组件出参产生的结果。

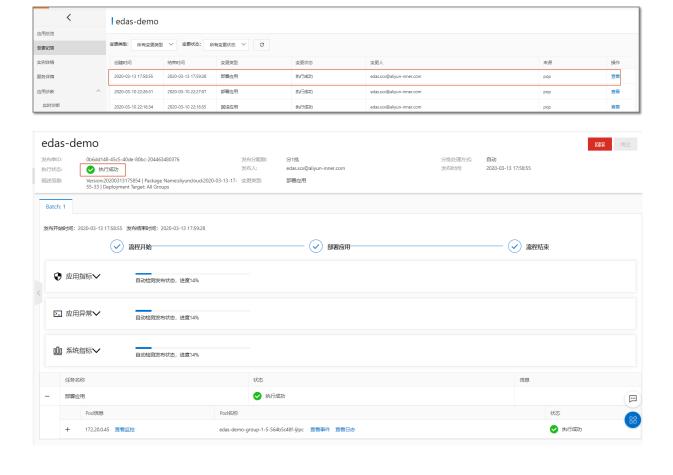
5. 单击右上角的运行。

云效开始依据配的流水线,集成应用将应用部署至EDAS。



步骤六: 结果验证

云效显示部署成功后,在EDAS控制台查看应用的变更记录,是否产生应用重新部署的变更记录,并显示成功。



常见问题

- 如何使用云效以JAR或者WAR包方式部署应用到EDAS?
 - 1. 在EDAS控制台以JAR或者WAR包方式创建应用。
 - **2.** 在步骤四:在云效中配置构建任务的步骤2中构建**阶段模板**选择**Java JAR/WAR包构建**,并在 弹出的任务步骤设置页面中,依据提示设置JAR或者WAR包构建物信息。
 - 3. 设置部署阶段任务并将应用至EDAS。

• 其他编程语言如何使用云效将应用部署到EDAS?

在流水线构建任务环节,您选择所需的编程语言模板,具体部署操作请参见#unique_43。



• 除Java应用外其他编程语言的应用想要部署在EDAS上,在EDAS创建应用时使用那种部署方式(JAR、WAR和镜像)?

使用镜像方式。使用云效部署时,切记您应用程序代码中须包含应用的Dockerfile文件。

6.5 金丝雀发布 (K8s 集群)

如需升级部署在容器服务 Kubernetes 集群中的 Spring Cloud 或 Dubbo 微服务应用,可以使用金 丝雀发布进行小规模验证,验证通过后再全量升级。

使用限制

- Dubbo 应用在使用金丝雀发布时没有使用限制。
- Spring Cloud 应用在使用金丝雀发布时有以下限制:
 - 暂不支持 Netflix Zuul、Spring Cloud Gateway。
 - 入口应用(承接外部流量的第一个应用节点)暂不支持灰度。

入口应用直接承接外部流量,由于暂不支持对这种流量进行截获和路由,因此对入口应用暂不能提供流量灰度控制。一种理想的模式是搭建微服务网关来作为承接外部流量的"入口应用",这样所有业务应用都在网关之后,就可以实现流量灰度控制了。

- 如果应用依赖 Deployment.Metadata.Name 或 Deployment.Metadata.Uid 的功能与配置,请勿使用灰度发布,否则发布后该原生功能将出现异常。

视频

操作步骤

- 1. 登录EDAS控制台。
- 2. 在左侧导航栏中选择应用管理 > 应用列表。
- **3.** 在**应用列表**页面选择**地域**和**命名空间**,然后单击具体应用名称。
- 4. 在应用详情页面右上角单击部署应用。
- 5. 在部署模式选择页面灰度发布(金丝雀)区域右上角单击开始部署。

- 6. 在灰度发布页面上传新版本应用的部署包、设置发布策略和灰度规则,然后单击确定。
 - a) 上传新版本应用的部署包。



b) 在**发布策略**区域配置发布策略参数。

右侧的发布策略配置信息会根据配置显示灰度发布流程。



发布策略参数说明:

- **首批灰度数量**: 灰度发布的应用实例数量。右侧会显示应用当前实例数,为了保证应用稳定性,灰度实例数不能超过应用实例总数的 50%。
- 剩余批次: 灰度发布后,剩余的应用实例按照设定的批次完成分批发布。
- 分批间处理方式:包括自动和手动两种方式,仅在灰度后剩余批次设置为2批及以上时需要设置。
 - 自动:根据**间隔**自动分批发布。**间隔**即剩余批次间的发布时间间隔,单位为分钟。
 - 手动:手动触发下一个批次的发布。
- 批次内部署间隔:每一批次内,如果应用实例数大于1,应用实例间的部署时间间隔,单位为秒。
- c) 设置灰度规则。

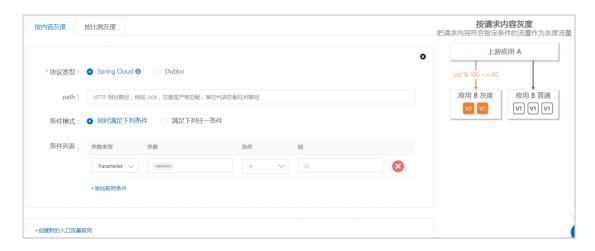
目前支持按内容灰度和按比例灰度两种方式设置灰度规则。

按内容灰度: 单击创建新的入口流量规则,设置入口流量规则。



说明:

您可以创建多个入口流量规则。



按内容灰度参数说明:

- **协议类型**:包含 Spring Cloud 和 Dubbo,按应用实际情况选择。

■ Spring Cloud:需要设置 path。

■ Dubbo:需要选择服务和方法。

条件模式:包含同时满足下列条件和满足下列任一条件。

- 条件列表: Spring Cloud 和 Dubbo 协议的条件有所不同。

■ Spring Cloud:包含 Cookie、Header和 Parameter 3 种方式,根据实际需求设置。

■ Dubbo: 根据应用实际的参数、参数值获取表达式设置。

• 按比例灰度:设置流量比例,流量会按配置的比例被转发到当前的灰度分组。

灰度发布启动后,EDAS 将先在指定的灰度分组中部署新版本应用,基本信息页面提示**应用有变更** 流程正在执行,处于执行中状态。

单击**查看详情**,进入**变更详情**页面查看部署进度和状态。

如果在灰度发布时,发现新版本有问题,还可以终止变更。终止变更后,提示 **应用处于金丝雀发布状态,并且该变更已经被终止。请先回滚应用,再进行其他操作。**

- 7. 监控灰度流量是否符合预期,详情请参见#unique_45。
- 8. 灰度流量验证完成后,在**变更详情**页面单击**开始下一批**,完成后续分批发布。

如果在验证过程中,发现新版本应用有问题,可以在**变更详情**页面右上角单击**立即回滚**。在弹出的**立即回滚**对话框确认回滚的影响,然后单击**回滚**。

结果验证

灰度发布后,在**基本信息**页面查看**部署包**是否为新部署的应用版本。在**实例部署信息**页面查看应用实例的**运行状态**是否为**运行正常**。

6.6 分批发布 (K8s 集群)

在应用发布与产品迭代中,经常采用分批发布的方式来控制上线风险。

分批发布简介

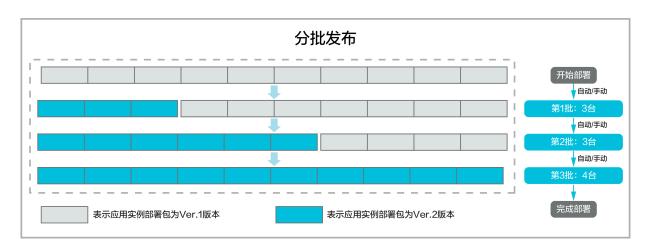
分批发布是按照一定的批次,每次只对应用的一部分实例进行升级的发布过程。分批发布过程中如果 出现故障,您可以终止变更过程并进行回滚,待问题修复后重新发布。

在容器服务 Kubernetes 集群中,当一个应用进行分批发布时,是将该应用内的应用实例数均分到每一个批次进行部署,如果不能均分则批次靠前数量较小,批次靠后数量较大。

场景示意

现有一个应用包含 10 个应用实例,每个应用实例的部署版本为 Ver.1 版本,现需将每个应用实例升级为 Ver.2 版本。

假设将所有应用实例分 3 批进行部署,根据分批发布的策略,该发布的流程示意如下图:



使用须知

• 在容器服务 Kubernetes 集群中,应用使用分批发布时会新建 Deployment 进行部署。

- 当您的应用包含了以下原生功能或配置时,请勿使用分批发布,否则发布后该原生功能将出现异常:
 - HPA
 - Rancher
 - Istio
 - 依赖 Deployment.Metadata.Name 或 Deployment.Metadata.Uid 的功能与配置

操作步骤

- 1. 登录EDAS控制台。
- 2. 在左侧导航栏中选择应用管理 > 应用列表。
- **3.** 在**应用列表**页面选择**地域**和**命名空间**,然后单击具体应用名称。
- 4. 在应用详情页面右上角单击部署应用。
- 5. 在部署模式选择页面分批发布区域单击开始部署
- 6. 在分批发布页面上传应用新版本的部署包。
 - WAR/JARWAR 包和 JAR 包部署仅仅是部署包形式不同,其它参数一致。
 - 镜像

镜像只需要选择新的镜像版本,无需设置其它参数。

7. 设置发布策略。

✓ 发布策略		
● 当您的应用使用了以下K8s原生功能或配置(包括HPA、Rancher、Istio以及依赖Deployment.Metadata.Name或 Deployment.Metadata.Uid的功能与配置)时,请勿使用灰度发布或分批发布。否则,应用部署之后,这些K8s原生功能或配置符出现异常。详情请参见使用须知。		
发布批次 2 🗘	发布策略配置信息	
分批间处理方式 自动 V 间隔 5 🔷 min	1 开始部署	
批次内部署间隔 10 0	2 [自动开始] 第1批:1台 批次阿屬Smin	
	3 [自动开始] 第2批:1台	
	4 部署结束	

发布策略参数说明:

- 发布批次:应用实例按照设定的批次完成分批发布。
- 分批间处理方式:包含自动和手动两种方式。
 - 自动:根据间隔自动分批发布。间隔即剩余批次间的发布时间间隔,单位为min。
 - 手动:手动触发下一个批次的发布。
- **批次内部署间隔**:每一批次内,如果应用实例数大于 1,应用实例间的部署时间间隔,单位为 秒。
- 8. (可选) 根据您的实际需求,选择是否需要配置**启动命令、环境变量、持久化存储、本地存** 储、应用生命周期管理和日志收集设置。
- 9. 设置完成后,单击确定。

结果验证

到应用中查看应用变更详情,查看分批发布的状态,待所有批次都执行成功则说明分批发布成功。

在应用详情页查看实例部署信息,当实例的版本已变为 Ver.2 并且所有实例运行状态均为 **Running** 时,则说明发布成功。

回滚应用

在分批发布期间,只要至少有一个应用实例未升级为新版本,就认为发布处于进行中状态。在监视升级时,如果您注意到第一批升级的应用实例停止响应,您可以进入变更详情页面单击**立即回滚**,将已发布的实例回滚到以前的服务包和配置。

在分批发布时,如果遇到异常,可参考#unique_46进行排查、解决。

部署包不可用、健康检查失败等异常报错会导致应用升级失败,当前应用变更会自动终止并回滚。

• 升级时单批次最大超时时间为 30 分钟,如果是超时导致变更流程暂停,您需进入**变更详情**页面手动终止发布流程并回滚。

7 用户指南(旧版)

7.1 容器服务K8s集群的应用生命周期管理

本文档将介绍如何在容器服务K8s集群中创建、部署、更新、扩缩容、启停、重置、重启和删除应用等管理操作。

创建应用

您可基于Dubbo、Spring Cloud或HSF的框架来开发应用,并将应用程序构建为WAR包、JAR包或镜像,然后将应用部署至EDAS的容器服务K8s集群中。

- 在容器服务K8s集群中使用镜像方式部署应用
- 在容器服务K8s集群中使用JAR包或WAR包部署应用
- #unique 48
- 使用 Cloud Toolkit 插件快速部署应用到容器服务 K8s 集群
- 使用toolkit-maven-plugin插件部署应用到容器服务K8s集群
- 使用云效部署Java应用至EDAS(K8s集群)

部署应用

在EDAS创建一个未部署的应用后,您可以在应用**基本信息**页面右上角单击**部署应用**进行部署发布。 已部署过的应用,可在应用**基本信息**页面右上角单击**部署应用**来升级应用版本。



注意:

升级应用时选择的部署包的类型需跟第一次部署时选择的部署包类型保持一致。

扩缩应用

应用扩容即通过增加应用实例的数量来增加应用的计算容量,应用缩容即通过减少应用实例的数量来减少应用的计算容量。您可以在应用实例负载过高时,在应用基本信息页面右上角单击**应用扩缩**来增加应用实例;在应用闲置时,您可在应用基本信息页面右上角单击**应用扩缩**来减少应用实例。

删除应用

您可以在应用**基本信息**页面的右上角单击**删除应用**,对应用进行删除。删除应用即删除这个应用相关的所有信息,释放该应用下的所有实例(Pod)。

7.2 应用设置

7.2.1 为 Kubernetes 集群中的应用添加负载均衡 SLB

在 EDAS 中的容器服务 Kubernetes 集群或自建 Kubernetes 集群中创建应用后,为应用添加公网或私网负载均衡 SLB 可实现应用的公网或私网访问。本文以容器服务 Kubernetes 集群中的应用为例介绍如何为应用添加公网 SLB。

背景信息

为容器服务 K8s 集群或自建 K8s 集群中的应用绑定 SLB 有以下区别。

- 自建 K8s 集群: 应用绑定 SLB 都需新购 SLB 实例,每个应用独享 SLB 实例。
- 容器服务 K8s 集群:应用绑定 SLB 可以新购 SLB 实例来独享 SLB 实例,也可以复用 SLB 实例。



说明:

复用 SLB 会有一些约束条件和特别的操作方式,详情请参见容器服务 K8s 集群中的应用如何复用 SLB。

前提条件

在 SLB 控制台#unique_51。



注意:

创建的 SLB 和要绑定 SLB 的应用要在同一个 VPC 内。

给应用绑定 SLB

- 1. 登录EDAS控制台。
- 2. 在左侧导航栏中选择应用管理 > 应用列表, 在应用列表页面单击具体的应用名称。
- 3. 在基本信息页签的应用设置区域,单击负载均衡(公网)右侧的添加。



说明:

如果您已经配置过负载均衡,则在此处会显示负载均衡的 IP 和端口信息,您可单击**修改**进入配置页面修改负载均衡信息,或可单击**解绑**解除 SLB 的绑定。

- 4. 在添加公网 SLB 配置确认对话框,设置负载均衡参数,然后单击确认。
 - 选择 SLB: 您可以新建 SLB 或使用已有的 SLB。
 - 新建 SLB: 系统会为您的应用自动购买一个新的 SLB 实例, SLB 的信息会显示在下方。您也可以单击**查看价格**了解 SLB 价格。



注意

新建的 SLB 不能被其它应用复用。如果您要复用 SLB, 请选择已有的 SLB。

- 使用已有的 SLB: 在右侧的下拉菜单中选择一个已经创建的 SLB 实例。
- SLB 端口:公网负载均衡前端端口,通过该端口访问应用,可设置范围为 1~65535。
- 容器端口:进程监听的端口。一般由程序定义,例如: Web 应用默认使用 8080 端口。
- 网络协议: 默认为 TCP, 不可更改。

结果验证

在浏览器地址栏中输入 <负载均衡 IP>:<端口号> , 例如 115.28.XX.XX:80 , 再按回车键即可进入各自的应用首页。

如果负载均衡右侧未出现 IP 和端口信息,则表示绑定负载均衡失败,请进入变更记录查看变更详情,根据变更记录排查并修复失败原因。

问题反馈

如果在使用容器服务 Kubernetes 集群过程中有任何疑问,欢迎您扫描下面的二维码加入钉钉群进行 反馈。



7.2.2 容器服务 K8s 集群中的应用如何复用 SLB

容器服务 K8s 集群已支持复用 SLB, 您在 EDAS 的容器服务 Kubernetes 集群中部署的多个应用可以添加同一个 SLB, 以便节省资源,同时还能提供应用的固定 IP 地址,方便运维。

前提条件

在为多个应用添加同一个 SLB 前,请先完成以下工作:

升级容器服务 Kubernetes 集群的 Cloud Controller Manager (CCM) 组件。升级步骤请参见#unique_52。

容器服务 Kubernetes 集群包含多个模板,需要将标准托管集群(Managed Kubernetes)和标准专有集群(Dedicated Kubernetes)模板的 CCM 组件升级到以下指定版本。

- 标准托管集群(Managed Kubernetes)的 CCM 需为 v1.9.3.112-g93c7140-aliyun 版本以上。
- 标准专有集群(Dedicated Kubernetes)的 CCM 需为 v1.9.3.106-g3f39653-aliyun 版本以上。



注意:

如果未升级容器服务 Kubernetes 集群的 CCM 组件,将导致**复用 SLB 失败,已有监听被删除、应用流量跌零。**

#unique_51。



注意:

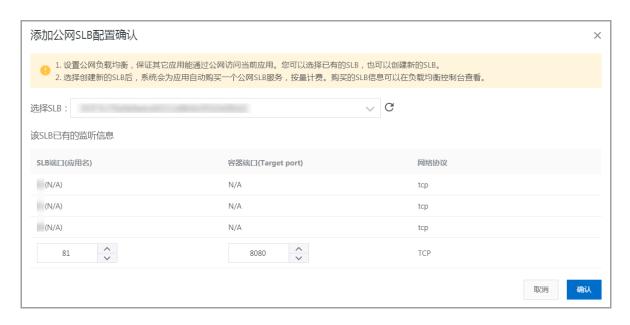
- 您在 EDAS 应用详情页添加负载均衡时 EDAS 为您代购的 SLB (新建 SLB) 不能被复用。
- 创建的 SLB 和要添加 SLB 的应用必须在同一个 VPC 内。
- 了解 SLB 的#unique_53。

复用 SLB

复用 SLB 即为您的多个应用添加 SLB,操作不再赘述,请参见为 Kubernetes 集群中的应用添加负载均衡 SLB。本文仅介绍在**添加公网 SLB 配置确认**对话框中的操作注意事项。

• 选择 SLB要在右侧的下拉列表中选择上一步创建的 SLB,而不能使用 EDAS 为您代购的 SLB。

• SLB 端口要设置为任意未被占用的端口。



已占用的 SLB 端口会显示在对话框中,如上图,80 端口已被其它应用占用,则需要为当前应用设置 80 之外的任意端口。

结果验证

- 1. 登录 负载均衡管理控制台。
- 2. 在左侧导航栏单击实例管理。
- 3. 在实例管理页面单击复用的 SLB 实例名称。
- **4.** 在**实例详情页面**单击**监听**页签,然后检查**端口**、(应用)**名称、服务器组**等信息是否和您配置的信息一致。



为多个应用添加 SLB 后,这些应用即可通过同一个 SLB 的 IP 地址和不同端口进行访问。

7.3 应用监控

7.3.1 应用总览

在**应用总览**页面上,您可以查看应用的健康状况关键指标,包括总请求量、平均响应时间等总体指标,应用所提供的服务、所依赖的服务的相关指标,以及 CPU 使用量、内存使用量等系统信息。

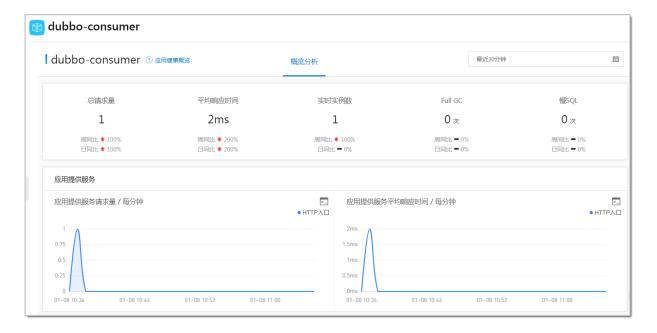
功能入口

- 1. 登录控制台,在左侧导航栏中单击应用列表或应用管理 > 应用列表。
- 2. 在应用列表中选择目标应用,在左侧导航栏中选择应用监控 > 应用总览。

概览分析

概览分析页签上展示以下关键指标:

- 选定时间内的总请求量、平均响应时间、实时实例数、FullGC 次数、慢 SQL 次数,以及这些指标和上周的同比、上一天的环比升降幅度。
- 应用提供服务:应用提供服务的请求量和平均响应时间的时序曲线。
- 应用依赖服务:应用依赖服务的请求量、平均响应和应用实例数的时序曲线,以及 HTTP-状态码统计。
- · 系统信息: CPU、MEM 和负载的时序曲线。



7.3.2 应用详情

7.3.2.1 JVM 监控

JVM 监控功能用于监控重要的 JVM 指标,包括堆内存指标、非堆内存指标、直接缓冲区指标、内存映射缓冲区指标、GC(垃圾收集)累计详情和 JVM 线程数等。本文介绍 JVM 监控功能和查看 JVM 监控指标的操作步骤。

功能介绍

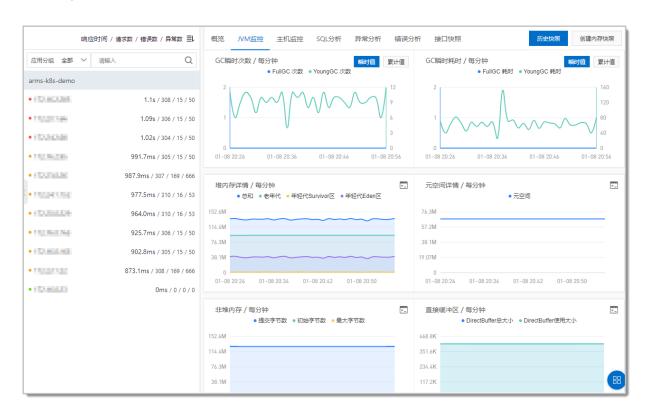
JVM 监控功能可监控以下指标:

- GC(垃圾收集)瞬时和累计详情
 - FullGC 次数
 - YoungGC 次数
 - FullGC 耗时
 - YoungGC 耗时
- 堆内存详情
 - 堆内存总和
 - 堆内存老年代字节数
 - 堆内存年轻代 Survivor 区字节数
 - 堆内存年轻代 Eden 区字节数
- 非堆内存
 - 非堆内存提交字节数
 - 非堆内存初始字节数
 - 非堆内存最大字节数
- 元空间
 - 元空间字节数
- 直接缓冲区
 - DirectBuffer 总大小(字节)
 - DirectBuffer使用大小(字节)

• JVM 线程数

- 线程总数量
- 死锁线程数量
- 新建线程数量
- 阻塞线程数量
- 可运行线程数量
- 终结线程数量
- 限时等待线程数量
- 等待中线程数量

图 7-1: JVM 监控



查看 JVM 监控指标

- 1. 登录ARMS控制台。
- 2. 在左侧导航栏中选择应用监控 > 应用列表。
- 3. 在应用列表页面选择您想查看的应用。

4. 在应用详情页面选择您想查看的节点,并在页面右侧单击 JVM 监控页签。

JVM 监控页签内展示了 GC 瞬时次数、GC 瞬时耗时、堆内存详情、非堆内存详情和 JVM 线程数的时序曲线。

- 单击 GC 瞬时次数/每分钟和 GC 瞬时耗时的瞬时值/每分钟面板右上角的瞬时值和累计值按钮,可以切换查看 GC 瞬时次数和 GC 瞬时耗时的瞬时值或累计值的时序曲线,默认为瞬时值。
- 单击各监控面板上的指标名称(例如 GC 瞬时次数),可打开或关闭该指标在图表中的可见性。



说明:

每个图表必须至少有一个指标设为可见,这意味着当图表中只有一个指标时,您无法关闭该指标的可见性。

7.3.2.2 主机监控

主机监控功能用于监控 CPU、内存、Disk(磁盘)、Load(负载)、网络流量和网络数据包的各项指标。本文介绍主机监控功能和查看主机监控指标的操作步骤。

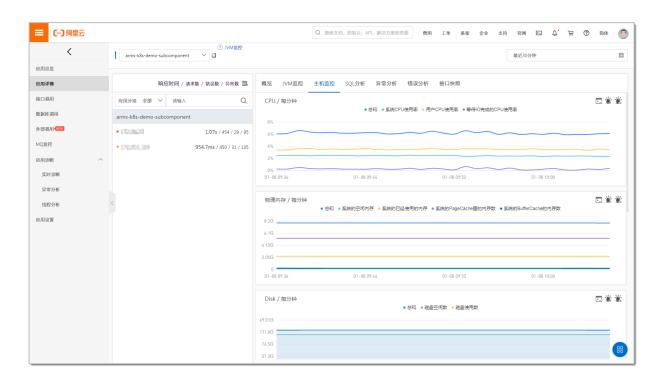
功能介绍

主机监控功能可监控以下指标:

- CPU
 - CPU 使用率总和
 - 系统 CPU 使用率
 - 空闲 CPU 使用率
 - 用户 CPU 使用率
 - 等待 IO 完成的 CPU 使用率
- 物理内存
 - 系统总内存
 - 系统空闲内存
 - 系统已使用内存
 - 系统 PageCache 中的内存
 - 系统 BufferCache 中的内存

- Disk (磁盘)
 - 系统磁盘总字节数
 - 系统磁盘空闲字节数
 - 系统磁盘使用率
- Load (负载)
 - 系统负载数
- 网络流量
 - 最近 30 秒平均每秒网络接收的字节数
 - 最近 30 秒平均每秒网络发送的字节数
- 网络数据包
 - 最近 30 秒平均每秒网络接收的报文数
 - 最近 30 秒平均每秒网络发送的报文数
 - 最近 30 秒平均每秒网络接收的错误数
 - 最近 30 秒平均每秒网络丢弃的报文数

图 7-2: 主机监控



查看主机监控指标

- 1. 登录ARMS控制台。
- 2. 在左侧导航栏中选择应用监控 > 应用列表。

- 3. 在应用列表页面选择您想查看的应用。
- 4. 在应用详情页面选择您想查看的节点,并在页面右侧单击主机监控页签。

主机监控页签内展示了 CPU、内存、Disk(磁盘)、Load(负载)、网络流量和网络数据包的时序曲线。

• 单击各监控面板上的指标名称(例如系统 CPU 使用率),可打开或关闭该指标在图表中的可见性。



说明:

每个图表必须至少有一个指标设为可见,这意味着当图表中只有一个指标时,您无法关闭该指标的可见性。

• 单击监控面板右上角的两个报警图标,可以查看已有报警的报警点和创建新的报警。创建报警的方法参见#unique_59和创建报警。

7.3.3 应用接口调用监控

接口调用功能用于监控应用下的接口调用详情,包括 SQL 分析、异常分析、错误分析、链路上下游和接口快照。

功能介绍

在应用监控的接口调用页面上,您可以查看该应用下的调用接口的调用详情。ARMS 可自动发现与监控以下 Web 框架和 RPC 框架中提供的接口:

- Tomcat 7+
- Jetty 8+
- Resin 3.0+
- Undertow 1.3+
- WebLogic 11.0+
- SpringBoot 1.3.0+
- HSF 2.0+
- Dubbo 2.5+

接口概览

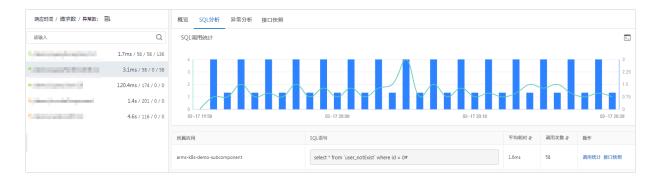
接口调用页面的概览页签列出了应用监控探针自动发现的所有接口。您可以按照响应时间、请求数、错误数或异常数对该列表排序。选中一个服务,即可在概览页签上查看该服务的详细调用拓扑,以及请求数、响应时间、错误数的时序曲线。

20200707



SQL 分析

SQL 分析页签展示的是左侧选中服务的代码段内所发起的 SQL 请求列表。借助此页签,您可以找出是哪一个 SQL 造成某个服务过慢。您还可以单击某个 SQL 中的**接口快照**来查看一个 SQL 执行逻辑所处的完整代码链路。



异常分析

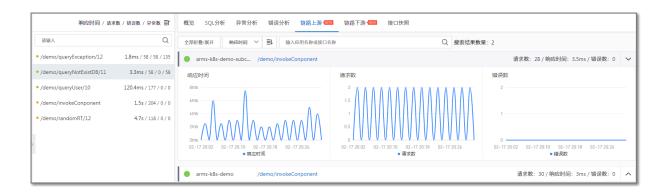
异常分析页签展示的是左侧选中服务的代码段内所抛出的 Java 异常。您还可以单击某个异常中的**接口快照**来查看一个异常堆栈所处的完整代码链路。



链路上游和链路下游的接口调用情况

链路上游和**链路下游**页签分别列出了应用上游(调用应用的一方)和应用下游(被应用调用的一方)的接口及其调用性能指标,包括请求数、响应时间和错误数。

图 7-3: 链路上游页签

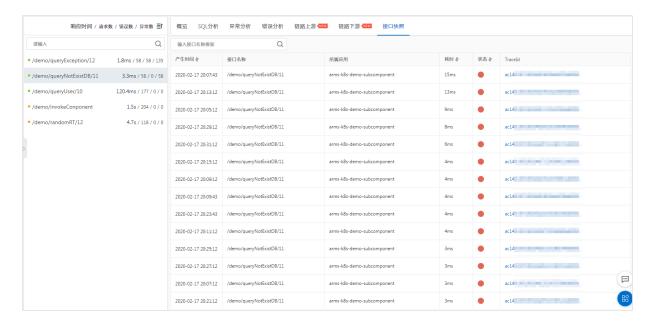


在链路上游和链路下游页签上,可按需执行以下操作:

- 在页签顶部单击全部折叠/展开,即可折叠或展开下方的所有接口。
- 在页签顶部的搜索框内输入应用名称或接口(Span)名称的关键字,并单击放大镜图标,即可筛 选出符合条件的接口。
- 单击接口信息所在的折叠面板,或者单击行末的上箭头或下箭头,即可展开或折叠该接口的性能指标信息。

接口快照

在服务链路快照中,您可以看到该服务接口中单次调用的调用堆栈、执行的明细 SQL、抛出的具体异常信息,以及接口中的参数详情。



7.3.4 高级监控

应用实时监控服务 ARMS(Application Real-Time Monitoring Service)是一款阿里云应用性能管理(APM)类监控产品。EDAS 可以无缝对接 ARMS 应用监控。您部署在 EDAS 上的应用可以通过开启高级监控获得 ARMS 提供的 APM (Application Performance Management)功能,从而对您的应用进行进阶的性能管理。

如果您想启用高级监控,请参考#unique 63。

7.3.5 报警管理

7.3.5.1 创建报警

通过创建报警,您可以制定针对特定监控对象的报警规则。当规则被触发时,系统会以您指定的报警方式向报警联系人分组发送报警信息,以提醒您采取必要的问题解决措施。

前提条件

• 创建联系人:仅可将联系人分组设为报警的通知对象。

背景信息

默认报警条件:

- 为避免您在短时间内收到大量报警信息,系统 24 小时内对于持续的重复报警信息仅发送一条消息。
- 如果 5 分钟内没有重复报警,则会发送恢复邮件,通知数据恢复正常。
- 发送恢复邮件后,报警的状态会重置。如果该报警再次出现,会被视为新报警。

报警控件本质是数据集的数据展示方式,所以在创建报警控件的同时,会创建一个数据集来存储报警控件的底层数据。



说明:

新建报警大约在 10 分钟内生效,报警判断会存在 1~3 分钟的延时。

创建报警

若需为应用监控任务创建一个 JVM-GC 次数同比报警, 具体操作步骤如下:

- **1.** 登录控制台,在**应用列表**页面单击目标应用,然后在左侧导航栏中选择**通知报警 > 报警策略管理**
- 2. 在报警策略管理页面的右上角单击创建报警。

- 3. 在**创建报警**对话框中输入所有必填信息,完成后单击**保存**。
 - a) 填写报警名称,例如: JVM-GC 次数同比报警。
 - b) 在**应用站点**列表中选择应用,在**应用组**列表中选择应用组。
 - c) 在类型列表中选择监控指标的类型,例如: JVM 监控。
 - d) 设置维度为**遍历**。
 - e) 设置报警规则。
 - A. 单击同时满足下述规则。
 - **B.** 编辑报警规则,例如:N=5 时 JVM_FullGC 次数的平均值与上小时同比上升 100% 时则报警。

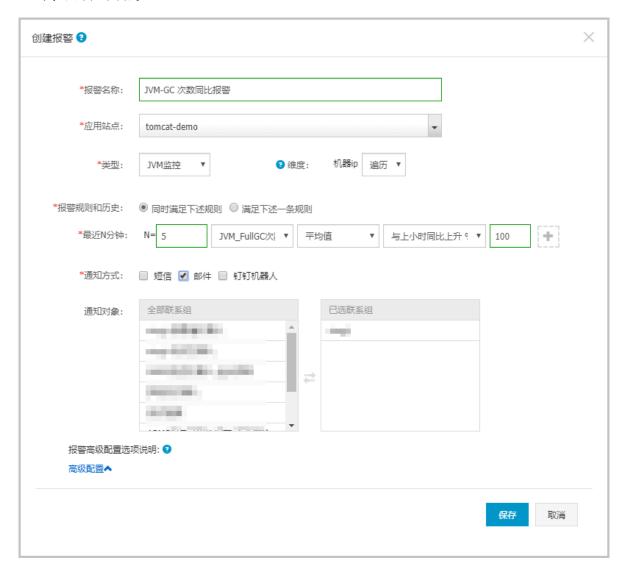


说明

20200707

若需设置多条报警规则,单击**报警规则**右侧的 + 号,即可编辑第二条报警规则。

- f) 勾选通知方式。例如: 勾选邮件。
- g) 设置通知对象。在**全部联系组**框中单击联系人分组的名称,该联系人分组出现**在已选联系组**框中,则设置成功。



通用基础字段含义

创建报警对话框的基础字段含义见下表。

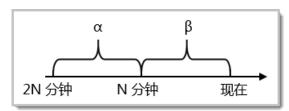
字段	含义	说明
应用站点	已创建的监控任务。	在下拉菜单中选择。

字段	含义	说明
报警类型	报警指标的类型。	三种报警的报警指标类型各不相同:
		 应用监控报警:应用入口调用、应用调用类型统计、数据库指标、JVM 监控、主机监控和异常接口调用。 前端监控报警:页面指标、API 指标、自定义指标和页面 API 指标。 自定义监控报警:基于已有下钻数据集创建报警和基于已有通用数据集创建报警。
报警维度	配置报警指标(数据集)的维度,可选择为:无、=、遍历。	 配置为无:报警内容中透出这个维度所有数值的和。 配置为 =:具体内容需手动填写。 配置为遍历:会在报警内容中透出实际触发报警的维度内容。
最近 N 分钟	报警判断最近 N 分钟内数据 结果是否达到触发条件。	N 的范围为:3~3600 分钟。
通知方式	支持邮件、短信和钉钉机器 人三种方式。	可勾选多种方式。若需设置钉钉机器人报警请参 见设置钉钉机器人报警。
报警静默期开关	可选择为开启或关闭 <i>,</i> 默认 为开启状态。	 开启报警静默期开关:若数据一直处于触发状态,首次触发报警后,24 小时后才会发送第二次报警信息。当数据恢复正常,会收到数据恢复通知并解除报警。若数据再次触发报警,则会再次发送报警信息。 关闭报警静默期开关:若报警连续触发,将会每分钟发送一次报警信息。
报警级别	包括警告、错误和致命。	-
通知时间	报警发送时的通知时间。此时间范围外将不发送报警通知,但仍会有报警事件记录。	查看报警事件记录请参见管理报警 。
通知内容	自定义的报警通知内容。	您可以编辑默认模板。在模板中,除 \$报警名称、\$筛选、 \$报警时间和 \$报警内容等 4 个变量(暂不支持其它变量)为固定搭配,其余内容均可自定义。

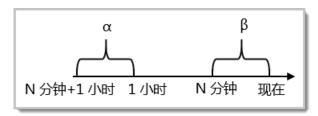
20200707

通用复杂字段含义:环比与同比

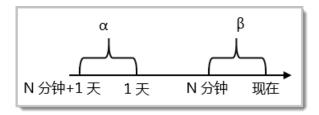
• 环比上升 / 下降 %: 若 β 为最近 N 分钟的数据(可选择为平均值、总和、最大值和最小值), α 为前 2N 分钟到前 N 分钟的数据, 环比为 β 与 α 做比较。



• 与上小时同比上升 / 下降 %: 若 β 为最近 N 分钟的数据(可选择为平均值、总和、最大值和最小值), α 为上小时最近 N 分钟的数据,与上小时同比为 β 与 α 做比较。



• 与昨日同比上升 / 下降 %: 若 β 为最近 N 分钟的数据(可选择为平均值、总和、最大值和最小值), α 为昨日同一时刻最近 N 分钟的数据,与昨日同比为 β 与 α 做比较。



通用复杂字段含义:报警数据修订策略

报警数据修订策略可选择为补零、补一或补零 Null(默认)。此功能一般用于无数据、复合指标和环 比同比等异常的数据修复。

• 补零: 将被判断的数值修复为 0。

• 补一: 将被判断的数值修复为 1。

• 补零 Null: 不会触发报警。

应用场景:

• 异常情况一:无数据

用户 A 想利用报警功能监控页面访问量。创建报警时,选择前端监控报警,设置报警规则为 N = 5 时页面访问量的总和小于等于 10 则报警。若该页面一直没有被访问,则没有数据上报,不会发送报警。为解决此类问题,可将报警数据修订策略勾选为补零,将没有收到数据视为收到零条数据,符合报警规则,即可发送报警。

• 异常情况二:复合指标异常

用户 B 想利用报警功能监控商品的实时单价。创建报警时,选择自定义监控报警,设置变量 a 的数据集为当前总价,变量 b 的数据集为当前商品总数,报警规则为 N = 3 时 (当前总价)/(当前商品总数)的最小值小于等于 10则报警。若当前商品总数为 0 时,复合指标(当前总价)/(当前商品总数)的值不存在,则不会发送报警。为解决此类问题,可将报警数据修订策略勾选为补零,将复合指标(当前总价)/(当前商品总数)的值视为 0,符合报警规则,即可发送报警。

• 异常情况三:指标环比、同比异常

用户 C 想利用报警功能监控节点机用户使用 CPU 百分比。创建报警时,选择应用监控,设置报警规则为 N = 3 时节点机用户使用 CPU 百分比的平均值环比下降 100%则报警。若最近 N 分钟用户的 CPU 故障无法工作,即 α 无法获取,导致环比结果不存在,则不会发送报警。为解决此类问题,可将报警数据修订策略勾选为补一,将环比结果视为下降 100%,符合报警规则,即可发送报警。

后续步骤

您可以在管理报警系统中查询和删除报警记录。

7.3.5.2 管理报警

在报警策略管理页面上,您可以管理账号下的所有报警规则,并查询报警事件和报警通知的历史记录。

管理报警规则

- 1. 在**应用列表**页面单击目标应用,然后在左侧导航栏中选择**通知报警 > 报警策略管理**。
- 2. (可选) 在报警规则页签的搜索框中输入报警名称,并单击搜索。



说明:

您可以输入报警名称的一部分内容进行模糊搜索。

- 3. 在搜索结果列表的操作列中,按需对目标报警规则采取以下操作:
 - 如需编辑报警规则,请单击**编辑**,在**编辑报警**对话框中编辑报警规则,并单击**保存**。
 - 如需删除报警规则,请单击**删除**,并在**删除**对话框中单击**删除**。
 - 如需启动已停止的报警规则,请单击**启动**,并在**启动**对话框中单击**启动**。
 - 如需停止已启动的报警规则,请单击**停止**,并在**停止**对话框中单击**确定**。
 - 如需查看报警事件历史和报警通知历史,请单击操作区域内的报警历史,并在报警历史页签上 查看相关记录。

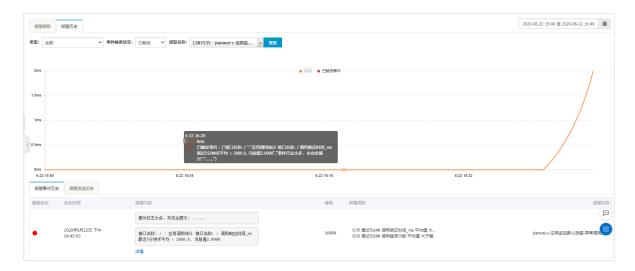


查询报警历史

关于报警规则何时因为什么事件被触发的历史记录,以及触发报警规则后发送给指定报警联系人的报 警通知历史记录,都可以在**报警历史**页签搜索。

- 1. 在报警规则和历史页面单击报警历史页签。
- 2. 在报警历史页签上选择事件触发状态和输入报警名称,并单击搜索。

报警历史页签上的图表展示了当前报警数据与报警事件的折线图和柱状图、报警数据与报警触发事件的联系以及报警触发详情等信息。折线图表示报警数据,柱状图表示报警事件。



3. 将页面向下滚动至底部的报警事件历史页签,可查看报警事件的历史记录。



说明:

仅触发状态为**已触发(触发**列中显示红色圆点)时才会发送报警通知。



4. 单击报警发送历史页签,可查看已触发报警发送的报警通知(短信、邮件等)记录。

7.3.5.3 创建联系人

报警规则被触发时会向您指定的联系人分组发送通知,而在创建联系人分组之前必须先创建联系人。创建联系人时,您可以指定联系人用于接收通知的手机号码和邮箱地址,也可以提供用于自动发送报警通知的钉钉机器人地址。

前提条件

如需将钉钉机器人添加为联系人,则需要先获取钉钉机器人的地址。详情请参见设置钉钉机器人报警。

操作步骤

- **1.** 登录控制台,在**应用列表**页面单击目标应用,然后在左侧导航栏中选择**通知报警 > 报警策略管理**
- 2. 在报警策略管理页面的右上角单击创建报警。
- 3. 在联系人页签上,单击右上角的新建联系人。
- 4. 在新建联系人对话框中编辑联系人信息。
 - 如需添加联系人,请编辑联系人姓名、手机号码和邮箱。



说明:

手机号码和邮箱必须至少填写一项。每个手机号码或邮箱只能用于一个联系人。至多可添加 100 个联系人。

• 如需添加钉钉机器人,请填写机器人名称和钉钉机器人地址。



说明:

获取钉钉机器人地址的方法参见设置钉钉机器人报警。

后续步骤

- 如需搜索联系人,请在联系人页签上,从搜索下拉框中选择姓名、手机号码或 Email,然后在搜索框中输入联系人姓名、手机号码或邮箱的全部或部分字符,并单击搜索。
- 如需编辑联系人,请单击联系人右侧操作列中的编辑,在更新联系人对话框中编辑信息,并单击确定。
- 如需删除单个联系人,请单击联系人右侧操作列中的删除,并在删除对话框中单击删除。
- 如需删除多个联系人,请勾选目标联系人,单击**批量删除联系人**,并在**提示**对话框中单击**确定**。

7.3.5.4 创建联系人分组

创建报警规则时,您可以将联系人分组指定为报警通知对象,当报警规则被触发时,ARMS 会向该联系人分组中的联系人发送报警通知。本文介绍如何创建联系人分组。

前提条件

创建联系人

操作步骤

- 1. 登录控制台,在应用列表页面单击目标应用,然后在左侧导航栏中选择通知报警 > 联系人管理。
- 2. 在联系人组页签上,单击右上角的新建联系组。
- 3. 在新建联系组对话框中填写组名,选择报警联系人,并单击确定。



说明:

如果**报警联系人**列表中没有选项,则您需要先创建联系人。

后续操作

如需搜索联系组,请在联系人组页签的搜索框中输入联系人分组名称的全部或部分字符,并单击搜索。



注意:

英文搜索关键字区分大小写。

• 如需编辑联系组,请单击联系人分组右侧的铅笔图标,并在**编辑联系组**对话框中编辑相关信息。

• 如需查看联系组中的联系人信息,请单击联系人分组右侧的下箭头图标来展开联系组。

图 7-4: 查看联系组中的联系人信息





说明:

您可以在展开模式下移除联系组中的联系人。如需移除,请单击目标联系人操作列中的移除。

如需删除联系组,请单击联系人分组右侧的删除(X)图标。



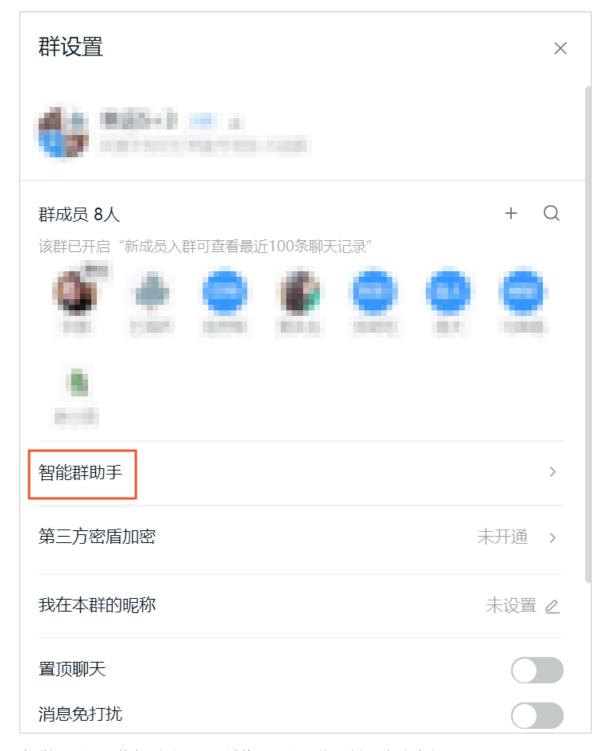
注意:

删除联系组之前,请确保没有正在运行的监控任务,否则可能导致报警等功能失效。

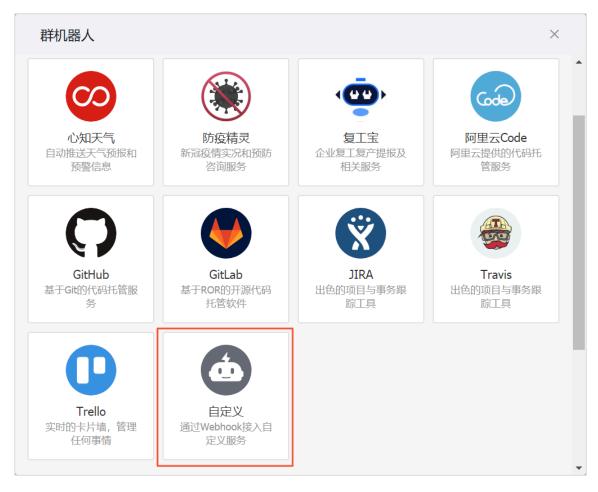
7.3.5.5 设置钉钉机器人报警

设置钉钉机器人报警后,您可以通过指定钉钉群接收报警通知。本文介绍设置钉钉机器人报警的操作步骤。

- 1. 获取钉钉机器人地址。
 - a. 在 PC 版钉钉上打开您想要添加报警机器人的钉钉群,并单击右上角的群设置图标。
 - b. 在群设置弹框中单击**群机器人**。



c. 在群机器人页面单击添加机器人区域的 + 图标, 然后选择添加自定义。



d. 在添加机器人对话框中编辑机器人头像和名称,然后单击完成。



e. 在添加机器人对话框中复制生成的机器人地址。



- 2. 在控制台的联系人管理页面将钉钉机器人创建为联系人。请参见创建联系人。
- 3. 创建一个联系组,并选择上一步创建的联系人为报警联系人。请参见创建联系人分组。
- 4. 设置报警规则。
 - 若您未创建报警,请先创建报警,并选择通知方式为**钉钉机器人**,设置通知对象为第 3 步创建的联系组。详情请参见创建报警。
 - 若您已创建报警任务,则需编辑报警,并选择通知方式为**钉钉机器人**,设置通知对象为第 3 步创建的联系组。详情请参见管理报警。

操作至此,您已成功设置一个钉钉机器人报警。当报警触发时,您将在设置接收报警的钉钉群中收到报警通知。例如:



7.4 查看应用变更

当您在 EDAS 上进行应用部署、启动、扩容/缩容等生命周期操作后,可以跳转到应用详情页查看当前变更状态,也可以通过变更记录页面查看该应用的历史变更记录。

查看变更详情

下面以一次部署应用为例说明如何查看应用变更。

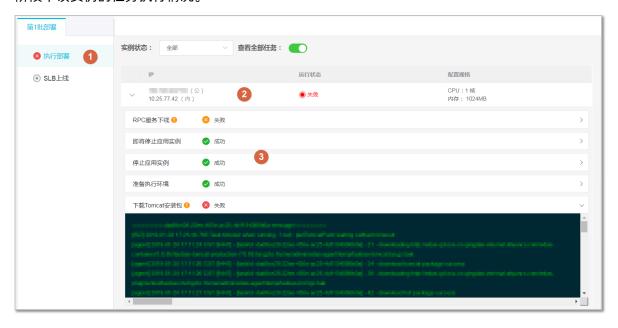
1. 在执行完应用变更操作后,返回**应用详情页**。

在应用详情页上方,会提示应用有变更流程正在执行,处于执行中状态。

2. 单击详情点击,进入变更详情页面,查看该应用的变更信息及实时状态。



- 变更概要信息:包括变更流程 ID、执行状态、变更类型等信息。
- 变更流程执行信息:包含整个流程(Process)的每个阶段(Stage),每个阶段(Stage)包含若干具体任务(Task)。
 - 如果是单实例任务,则在各阶段通过图标标识执行任务及结果。
 - 如果是多实例任务,则在每个阶段中基于实例分别展示执行任务及结果。
- **3.** 在变更流程执行区域左侧的**第 x 批部署**下方单击某个阶段,然后在右侧单击实例 IP 地址,查看该阶段中该实例的任务执行情况。



4. 单击具体任务,可以展开该任务执行的日志。 对于执行失败的任务,系统会自动展开失败日志。同时异常情况可以参考变更流程问题排查指 南进行处理。

查看应用变更记录

- **1.** 在应用详情页面左侧的导航栏中单击**变更记录**,查看该应用的所有变更操作的记录。
- 2. 在操作列单击**查看**,即可查看具体变更详情,及每一步操作的详细信息。

7.5 应用事件

7.5.1 查看应用事件

EDAS支持查看K8s原生的应用事件,帮助您了解应用运行时的状态,方便快速聚焦问题。

操作步骤

- 1. 登录EDAS控制台。
- 2. 在左侧导航栏中选择**应用管理 > 应用列表**,在**应用列表**页面单击具体的应用名称。
- 3. 在应用详情页面的左侧导航树, 单击应用事件。
- 4. 设置事件搜索过虑条件,并单击搜索。

当前搜索条件如下:

- ・ 来源类型:包括应用实例Pod、SLB (Service)、应用 (Deployment)、自动弹性 (HorizontalPodAutoscaler)。
- 来源名:输入事件的来源名,如应用名称、应用实例名称。
- 事件原因: 输入事件原因, 例如Pod的FailedScheduling。
- 事件等级:包含Waring和Normal。



注意:

如果事件中存在Waring级别的事件,请重点关注,检查您的应用。

更多信息

查看应用事件后,您可以依据具体的事件信息以及您的业务,进行相应的应用操作。

- SLB绑定
- #unique_73
- #unique_74
- 应用监控

常见问题

部署在EDAS上的K8s应用运行时,出现的问题一般为Pod问题。关于Pod问题请参见常见Pod问题。

7.5.2 常见Pod问题

部署在EDAS上的应用运行时,出现的问题一般为Pod问题。本文介绍常见的Pod问题以及处理方法。

Pod常见问题列表

- ImagePullBackOff
- CrashLoopBackOff
- RunContainerError
- 处于Pending状态的Pod
- 处于未就绪状态的Pod

ImagePullBackOff

当Kubernetes无法获取到Pod中某个容器的镜像时,将出现此错误。

可能原因:

- 镜像名称无效,例如拼错镜像名称,或者镜像不存在。
- · 您为image指定了不存在的标签。
- 您检索的镜像是私有registry,而Kubernetes没有凭据可以访问它。

解决方法:

- 前两种情况可以通过修改镜像名称和标记来解决该问题。
- 第三种情况,您需要将私有registry的访问凭证,通过Secret添加到Kubernetes中并在Pod中引用它。

CrashLoopBackOff

如果容器无法启动,则Kubernetes将显示错误状态为: CrashLoopBackOff。

可能原因:

- 应用程序中存在错误,导致无法启动。
- 未正确配置容器。
- Liveness探针失败太多次。

解决方法:

您可以尝试从该容器中检索日志以调查其失败的原因。如果容器重新启动太快而看不到日志,则可以使用命令: \$ kubectl logs <pod-name> --previous。

RunContainerError

当容器无法启动时, 出现此错误。

可能原因:

- 挂载不存在的卷,例如ConfigMap或Secrets。
- 将只读卷安装为可读写。

解决方法:

请使用kubectl describe pod命令收集和分析错误。

处于Pending状态的Pod

当创建应用过程中,在变更记录中该Pod一直Pending状态。

可能原因:

- 集群没有足够的资源(例如CPU和内存)来运行Pod。
- 当前的命名空间具有ResourceQuota对象,创建Pod将使命名空间超过配额。
- 该Pod绑定到一个处于pending状态的 PersistentVolumeClaim。

解决方法:

- 检查\$ kubectl describe pod <pod name>命令输出的"事件"部分内容或者在控制台查看应用事件。
- 对于因ResourceQuotas而导致的错误,可以使用\$ kubectl get events --sort-by=.metadata. creationTimestamp命令检查集群的日志。

处于未就绪状态的Pod

如果Pod正在运行但未就绪(not ready),则表示就绪探针失败。

可能原因:

当就绪探针失败时,Pod未连接到服务,并且没有流量转发到该实例。

解决方法:

就绪探针失败失败是应用程序的特定错误,请检查\$ kubectl describe pod <pod name>命令输出的"事件"部分内容,或者在控制台查看对应的应用事件。

7.6 限流降级

EDAS已支持使用AHAS实现应用Spring Cloud应用、Dubbo应用和HSF应用的限流降级,支持实时查看限流降级详情和动态变更规则,全面保障您的应用的可用性。

前提条件

• 请确保您使用的是**专业版**或**铂金版**的EDAS。

• 请确保您已开通AHAS,详情请参见开通AHAS。



说明:

在使用AHAS时需要单独付费。

背景信息

ECS集群和容器服务Kubernetes集群均支持使用AHAS实现限流降级。

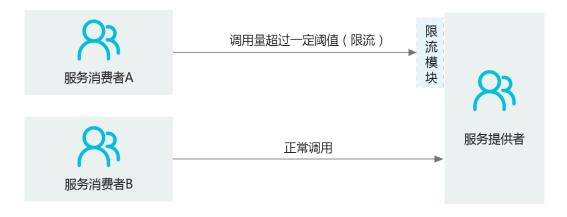
- 新部署的应用将默认使用AHAS组件进行限流降级。
- 部署的已有应用如果未使用过限流降级,当您使用限流降级时,将默认使用AHAS进行限流降级。
- 部署的已有应用如果已使用过限流降级,将继续使用原有的实现方式。详情请参见#unique_77。

限流降级简介

限流

限流可以理解为是一个控制流量阈值或调节比例的功能。在前端网站面对大流量访问的时候,可以通过调节流量阈值来控制通过系统的最大流量值,防止大流量对后端核心系统造成破坏,导致服务不可用,保证系统安全可靠运行。

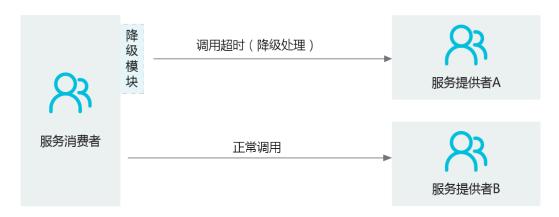
通过在服务提供者端配置限流模块代码,并在EDAS中配置限流策略后,使服务提供者具备限流功能。此时服务消费者去调用服务提供者时,所有的访问请求都会通过限流模块进行计算,若服务消费者调用量在一定时间内超过了预设阈值,则会触发限流策略,进行限流处理。



降级

在EDAS中,降级通常用于对下游出现超时的非核心服务提供者进行低优先级调用,确保上游核心应用(服务消费者)不被影响。

通过在服务消费者端配置降级模块代码,并在EDAS中配置降级策略,使服务消费者具备降级功能。此时服务消费者去调用服务提供者时,若服务提供者服务响应时间超过了预设阈值,则会触发降级策略进行降级处理。



操作步骤

- 1. 登录EDAS控制台。
- 2. 在左侧导航栏中选择应用管理 > 应用列表, 在应用列表页面单击具体的应用名称。

3. 在应用详情页左侧导航栏中单击限流降级,然后在展开的菜单中单击具体的功能菜单。

AHAS提供的限流降级功能具体包括:

- **监控详情**: 查看应用中的所有资源的数据分布、历史水位对比情况,详情请参见#unique 78。
- 簇点链路:请求链路页面会展示当前应用在单个实例上的所有资源以及实时的调用数据。包含两种视图:
 - 平铺视图:不区分调用链路关系,平铺展示资源的运行情况。
 - 树状视图:根据资源的调用链路关系,展示树状结构。

详情请参见#unique_79。

- **流控规则**: 监控应用流量的QPS或线程数等指标,当达到您指定的阈值时立即拦截流量,以避免被瞬时的流量高峰冲垮,从而保障应用的可用性。详情请参见#unique 80。
- **降级规则**:监控应用下游依赖应用的响应时间或异常比例,当达到您指定的阈值时立即降低下游依赖应用的优先级,避免应用受到影响,从而保障应用的可用性。详情请参见#unique_81。
- **系统规则**:从应用的总体Load、RT、QPS和线程数四个维度监控应用数据,对应用的入口流量进行控制,让应用尽可能运行在最大吞吐量的同时保证系统整体的稳定性。详情请参见#unique_82。
- 机器列表:展示应用接入AHAS流控降级服务的所有实例。当一段时间没有向AHAS控制台发送心跳时,此实例会自动被标记为失联状态。
- 操作日志:记录当前云账号及当前云账号创建的RAM用户对应用进行的操作,包括具体资源名、操作的内容、操作时间和操作人ID等,方便您追踪应用和资源的变更。
- **权限管理**:如子账号(即RAM用户)需配置和推送AHAS中应用的流控降级规则,您需要对子账号单独授予读写权限,从而严格管控权限,降低由于错误推送配置引发线上问题的可能性。详情请参见#unique 83。

7.7 弹性伸缩

7.7.1 弹性伸缩(K8s)

在分布式应用管理中,弹性伸缩是很重要的一个运维能力。弹性伸缩能够感知应用内各个实例的状态,并根据状态动态实现应用扩容、缩容。在保证服务质量的同时,提升应用的可用率。

为什么使用弹性伸缩

互联网、游戏类等应用在促销活动期间容易出现突发性流量洪流,SLA和资源成本不易平衡,极易造成系统响应延迟、系统瘫痪等问题。EDAS继承阿里巴巴应对双11的流量洪流技术,提供秒级自动弹性功能,保证SLA的同时也节省机器保有成本。多适用于互联网、游戏以及社交平台等行业。

监控指标弹性

EDAS通过监控您应用的CPU使用率和内存使用率,依据弹性策略自动为您扩容和缩容应用实例。



注意:

- 单个应用内最多可配置一条监控指标弹性策略。
- 弹性策略启用时,请勿进行应用生命周期管理操作,请停用弹性策略后,再执行。
- 执行应用变更(如部署应用、应用扩缩、变更规格等)时,无法添加弹性策略。
- 1. 登录EDAS控制台。
- **2.** 在左侧导航栏中选择**应用管理 > 应用列表**,在**应用列表**页面单击具体的应用名称。
- 3. 在应用详情页面单击**实例部署信息**页签,并单击弹性伸缩折叠面板,展开折叠面板后单击**添加弹**性策略。
- 4. 在右侧滑出的添加弹性策略面板中配置弹性规则,配置完成后单击确定。
 - **策略名称**:弹性策略的名称,须以字母开头,且由字母、数字、下划线、中划线组成。长度范围为1~64字符。
 - 策略类型:目前仅支持监控指标弹性策略,定时弹性策略即将上线。
 - 触发条件: 支持CPU使用率和Mem使用率。
 - 最大应用实例数: 触发弹性伸缩条件后, 应用扩容, 其实例数可达到的目标值。
 - 最小应用实例数: 触发弹性伸缩条件后,应用缩容,其实例数可达到的目标值。



说明:

单选CPU使用率和Mem使用率时,当前应用的CPU使用率或者Mem使用率大于或者等于所设的目标值,则对应用进行扩容,其应用实例数不超所设的最大应用实例数;反之,进行缩容,其应用实例数不低于所设的最小应用实例数。

• 全选**CPU使用率**和**Mem使用率**时,如果二者使用率同时大于或者等于所设目标,则应用进行扩容,其应用实例数不超所设的**最大应用实例数**;反之,进行缩容,其应用实例数不低于所设的最**小应用实例数**。

最大应用实例数和最**小应用实例数**的计算公式:目标实例数 = 当前实例数 × (当前指标/期望指标)

5. 在监控指标策略列表,单击操作列的启用。

触发弹性策略后,如果EDAS依据所设策略对应用实例进行扩容或者缩容,表示自动弹性成功。

结果验证(自动弹性伸缩)

使用弹性伸缩前,应用实例数如下图所示。



弹性策略如下图所示。



启用该弹性策略后,EDAS 自动监测 CPU 使用率,当满足触发条件时,EDAS 依据弹性策略进行应用实例扩容、缩容。

相比弹性伸缩前,应用实例增加1个Pod,表示通过弹性伸缩功能为应用扩容实例成功。



更多信息

弹性策略启用后,您可以对弹性策略进行**删除、停止、启用**和编辑,还可以进行一系列应用管理操作。

• 查看应用变更

- #unique_86
- #unique_74
- 应用监控
- #unique_87
- 限流降级
- #unique 73

7.8 日志诊断

7.8.1 实时日志

实时日志适用于容器服务 Kubernetes 集群、EDAS 自带 Kubernetes 集群和自建 Kubernetes 集群中部署的应用。

查看实时日志

当应用出现异常情况的时候,可以通过查看实时日志来排查容器(Pod)相关问题。

- 1. 登录EDAS控制台。
- 2. 在左侧导航栏中选择应用管理 > 应用列表, 在应用列表页面单击具体的应用名称。



说明:

请选择集群类型为**容器服务 K8s 集群、EDAS 自带 K8s 集群**或**自建 K8s 集群**,实时日志为 K8s 集群专属功能。

- 3. 在应用详情页面左侧的导航栏中选择实时日志。
- 4. 在Pod名称下拉框内选择要查看日志的 Pod, 查看实时日志。

7.8.2 日志目录

当您的应用出现异常情况的时候,可以通过查看实例和应用级别的日志来排查问题。EDAS提供了日志目录收藏、日志查看、日志搜索的功能。

收藏日志目录

日志目录页面包含 EDAS 相关的默认日志目录,您可以收藏(添加)应用日志目录。收藏日志之后即可查看该目录下的实例日志,详情请参见查看实例日志。

在收藏日志目录时还可以将该目录**添加到日志服务**,以便在**日志搜索**页面查看和搜索该目录下的应用日志。



说明:

收藏目录和取消收藏仅对日志目录可用。

- 1. 登录EDAS控制台。
- 2. 在左侧导航栏中选择应用管理 > 应用列表, 在应用列表页面单击具体的应用名称。
- 3. 在应用详情页面左侧的导航栏中选择日志管理 > 日志目录, 然后在日志目录页面单击收藏目录。
- **4.** 在**收藏目录**对话框中输入**应用日志目录**,选择是否将该目录**添加到日志服务**,然后单击**添加**。 输入应用日志目录时,请遵循以下要求:
 - 此目录必须在 /home/admin 目录下。
 - · 完整目录中必须包含 log 或者 logs。
 - 目录最后必须以斜杠 /结尾, 表示添加的是一个文件夹。

查看实例日志

在收藏了应用日志后,可以查看实例和应用级别的日志。

- 1. 在应用详情页面左侧导航栏中选择日志管理 > 日志目录。
- **2.** 在**日志目录**页面单击日志目录左侧的展开按钮 > , 然后在展开的日志文件列表的操作列单击**查 看**。

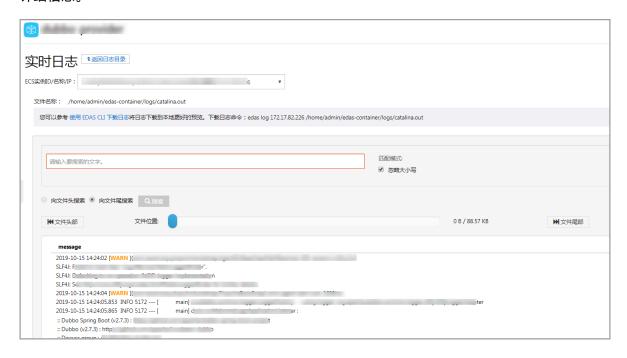




说明:

日志框架配置的文件下的日志文件除了可以查看日志详细信息,还可以修改该日志文件的级别。日志级别由低到高依次为:TRACE、DEBUG、INFO、WARN 和 ERROR。修改后,系统会将所设置的级别及更高级别的日志显示在页面下方。

3. 在**实时日志**页面顶部单击 ECS **实例 ID/名称/IP** 右侧的下拉箭头,选择实例,查看该实例中的日志详细信息。



在页面右下角单击**开启实时追加**,可以一直加载文件的最新追加内容(类似于tailf命令的效果)。

为 RAM 用户授予日志服务权限

如果您是 RAM 用户,需要通过子账号使用日志服务(包括将日志目录或文件添加到日志服务和查看应用日志和分布式搜索),还需要由云账号(主账号)在访问控制 RAM 中为 RAM 用户授权。操作步骤如下:

- 1. 使用云账号登录RAM访问控制控制台。
- **2.** 在左侧导航栏中单击**用户**,然后在**用户**页面的用户列表中找到您要授权的 RAM 用户,如 doctest,在操作列单击**添加权限**。
- 3. 在添加权限页面选择权限下面的系统权限策略右侧的文本框输入log,单击AliyunLogReadOnlyAccess将该权限添加到右侧列表中,单击确定。



说明:

被授权主体会默认加载,无需设置。如果您需要同时给多个 RAM 用户授权,在**被授权主体**下面的文本框通过关键字搜索并添加。

4. 在**授权结果**页面查看 RAM 用户和被授权的权限,确认无误后,单击**完成**。

取消收藏日志目录

取消收藏日志目录即移除指定的应用日志目录。移除后,该日志目录将不再显示在**日志目录**页面上,也不能再查看该目录下的实例日志,但并不会删除实际的日志目录及文件。



说明:

默认目录也可以取消收藏。

在取消收藏日志目录时,还可以选择并从日志服务中移除。

- 如果未从日志服务中移除,则可以查看该目录下的原有应用日志。
- 如果从日志服务中删除,则不能再查看该目录下的应用日志。
- 在日志目录页面选中某个日志目录,然后单击取消收藏。
- 在**取消收藏日志目录**对话框中,确认要取消收藏的日志目录,选择是否**并从日志服务中移除**,单 击**确定**。

7.8.3 查看应用的文件日志和容器标准输出日志

EDAS对接了日志服务SLS。如果在EDAS的容器服务Kubernetes集群创建或部署应用时开启了日志服务功能,则可以查看该应用的文件日志和容器标准输出日志。本文介绍如何为应用开启日志服务SLS及查看应用的文件日志和容器标准输出日志。

前提条件

- 开通日志服务SLS。
- 确保应用中每个实例预留了0.25核CPU和25 MB内存的可用资源。

步骤一: 为应用开启日志服务SLS

在容器服务Kubernetes集群中,日志服务需要在应用创建或部署时开启。详情请参见在容器服务K8s集群中使用镜像方式部署应用或在容器服务K8s集群中使用JAR包或WAR包部署应用中设置日志服务步骤。

步骤二: 查看文件日志和容器标准输出日志

- 1. 登录EDAS控制台。
- 2. 在左侧导航栏选择应用管理 > 应用列表。
- 3. 在页面顶部选择**地域**和**命名空间**,然后在**应用列表**页面单击开启了日志服务的应用名称。
- 4. 在应用详情页面左侧的导航栏选择日志管理 > 文件日志。

5. 在文件日志页面可以分别查看文件日志和容器标准输出日志。本文以查看文件日志为例进行介绍。单击**文件日志**页签,然后在文件日志列表中的**操作**列单击**查看文件日志**。



页面将跳转到日志服务控制台。

6. 在日志服务控制台查询及分析日志。详情请参见查询与分析。

8上传本地镜像

如果您本地有一个 Nginx 镜像,需要在 Kubernetes 集群中部署应用时使用,则需要先上传本地镜像。

前提条件

准备一台带有弹性公网 IP的 ECS 实例(或者该实例做过 NAT), 才能把本地构建好的镜像上传到阿里云镜像仓库。

操作步骤

- 1. 登录容器镜像服务控制台。
- 2. (可选) 如果您是第一次访问,系统会提示您设置登录密码,设置完成后,单击**确定**。



- 3. 创建命名空间。
 - a) 在左侧导航栏选择**默认实例 > 命名空间**。
 - b) 在**命名空间**页面顶部选择**地域**,然后在页面右上角单击**创建命名空间**。
 - c) 在**创建命名空间**对话框中,在**命名空间**文本框中输入命名空间名称,单击**确定**。



说明:

命名空间名称设置后不可修改。长度为2-30位,可填写小写英文字母、数字、下划线(_)和中划线(-)。下划线(_)和中划线(-)首位或末位。



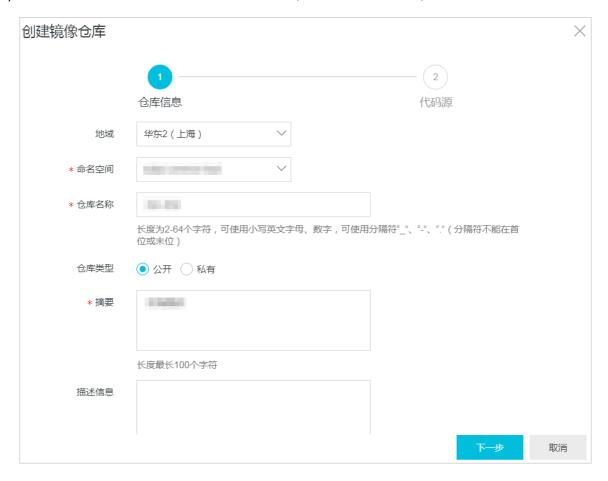
- 4. 创建镜像仓库。
 - a) 在左侧导航栏选择默认实例 > 镜像仓库。
 - b) 在镜像仓库页面单击创建镜像仓库。



说明:

在镜像仓库页面无需选择地域和命名空间,在创建仓库时可以设置。

c) 在**创建镜像仓库**配置向导对话框的**仓库信息**中,设置镜像仓库信息,然后单击**下一步**。



镜像仓库参数说明:

• 地域:在列表中选择创建镜像仓库的地域。

• 命名空间: 在列表中选择创建镜像仓库的命名空间。

• 仓库名称:输入镜像仓库的名称。

仓库类型:包含公开和私有两种类型,根据实际需求选择。

• 摘要:输入镜像仓库相关的信息。

• 描述信息: 输入镜像仓库的描述信息。

• 设置代码源: 当前版本只支持通过命令行推送镜像到镜像仓库, 请选择本地仓库。

d) 在**创建镜像仓库**配置向导对话框的**代码源**中,设置代码源信息,然后单击**创建镜像仓库**。

创建镜像仓库						×
	◆ ← ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○				- 2 代码源	
代码源	⊼Code	GitHub	Bitbucket	私有GitLab	本地仓库	
	您可以通过命令行推送镜像到镜像仓库。					
				上一步	创建镜像仓库	取消

代码源参数说明:

- 代码源:根据实际情况,可以选择云Code、GitHub、Bitbucket、私有GitLab和本地仓库。
- 代码变更时自动构建镜像: 当分支有代码提交后会自动触发构建规则。
- **海外机器构建**:构建时会在海外机房构建,构建成功后推送到指定地域。
- 不使用缓存:每次构建时会强制重新拉取基础依赖镜像,可能会增加构建时间。
- 5. 登录到带有弹性公网 IP 的 ECS, 然后执行 Docker 登录命令, 并输入密码。

\$ sudo docker login --username=test@aliyun.com registry.cn-beijing.aliyuncs.com



说明:

阿里云不同的可用区对应不同的 registry,以北京为例,登录时必须指明 registry 域名为 cnbeijing。其它可用区详见#unique_94。并输入您的用户名和登录密码。

登录成功之后会显示 Login Succeeded。

- 6. 重命名镜像。
 - 如果您本地的镜像是通过构建生成的,构建过程中需要指定 tag 信息。
 - \$ sudo docker tag [ImageId] registry.cn-beijing.aliyuncs.com/mw/nginx:[镜像版本号]
 - 如果您本地的镜像是从其它仓库下载的,已有了 registry 名、命令空间、仓库名和 tag 名
 称,上传到阿里云 registry 之前需要执行 Docker tag 命令进行修改。

\$ sudo docker tag docker.io/nginx:1.13.5 registry.cn-beijing.aliyuncs.com/mw/nginx : [镜像版本号]

7. 上传镜像到阿里云 Docker registry。

\$ sudo docker push registry.cn-beijing.aliyuncs.com/mw/nginx: [镜像版本号] 如果没有报错信息,您本地的 Nginx 镜像就成功上传到阿里云 Docker registry 了。