



云效 使用指南

文档版本: 20220712



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。

- 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。
- 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。
- 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. 阿里云网站上所有内容,包括但不限于著作、产品、图片、档案、资讯、资料、网站架构、网站画面的安排、网页设计,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
⚠ 危险	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。	⚠ 危险 重置操作将丢失用户配置数据。
⚠ 警告	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。
〔∫〉 注意	用于警示信息、补充说明等,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。
⑦ 说明	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。	⑦ 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,单击 确定 。
Courier字体	命令或代码。	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
[] 或者 [alb]	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 {act ive st and}

目录

1.持续交付流水线(老版)	05
1.1. 私密配置项	05
2.Helm私有仓库	09
2.1. 使用Helm私有仓库	09
3.Maven私有仓库	13
3.1. Maven仓库	13
3.2. 公共代理库	17
3.3. 将已有私库迁移至云效私库	19
3.4. 在云效构建中使用Maven私有仓库服务	21

1.持续交付流水线(老版)

1.1. 私密配置项

在应用构建中,通常会需要一些配置项,如:

- 1. 功能开关
- 2. 依赖系统的URL
- 3. 数据库链接用户名密码

对于前两项,云效没有提供额外的支持,推荐您直接在代码中保存不同的配置文件,然后在构建时根据 PACKAGE_LABEL 的环境变量,选取正确的配置文件。请阅读使用传入参数改变构建行为,了解更多关于 PACKAGE_LABEL 的用法。

第三项中的配置项会涉及一些私密信息,不适合放在代码库中。云效提供了私密配置项的保存功能。您可以 在应用的 私密配置项 页面(https://rdc.aliyun.com/ec/app/xxx/securityConfig)添加和配置应用级别 的私密配置项。比如:

新增key	
key	value
db_password 🖻	****** 🧪

如果您需要在多个环境都使用私密配置项,则可以使用如下的方式:

新增key

key	value
prod_db_password	****** 🥜
prepub_db_password	****** 🧪
testing_db_password m	****** 🧷

配置好这些私密配置项之后,在进行构建时,云效会把这些配置项转换成为一个明文的文件,并将其放置在 根目录下的 rdc_security_config.properties 中,比如:

rdc_security_config.properties:

db password=somepasswd

您可以在自己的构建脚本中读取该文件,并按照您自己的方式进行使用。

实例

接下来给出一个实际使用的例子:

代码库结构:

application.prepub.properties的内容:

```
key1=somevalue
key2=somevalue
key3=somevalue
db.password=${db_password}
```

私密配置项内容:

私密配置项可以保存一些您不希望被放到代码库中的配置项,并在构建时提供给您。详情见文档。 新增key

key	value
prepub.db_password 🖻	****** 🥜
produciton.db_password 🖮	****** 🥜

构建配置(这里给出两个包标签,在构建时会打出两个包):

弱级设置	
高级配置	
第1个包 删除	
包标签 prepu	o (?)
十添加参数 ⑦	
第2个包 删除	
包标签 produ	ction (?
十添加参数 ⑦	
十添加包⑦	

build.sh内容:

// 将rdc security config.properties中与当前PACKAGE LABEL有关的配置提取出来,去除PACKAGE LABEL前 缀,并转化成为可以source的格式。在这个例子中也就是: export db password=xxxxxx,最后存入临时文件 cat rdc security config.properties | grep \${PACKAGE LABEL} | sed s/^\${PACKAGE LABEL}./expor $t \geq /g > tmpconfig$ cat tmpconfig // source临时文件,将配置导入到环境变量中 source tmpconfig // 对指定配置文件(config/application.\${PACKAGE LABEL}.properties)中的环境变量占位符使用环境变量 中的值替换,从而变成私密配置项中配置的值 perl -pe 's/\\$\{(.*)}/ \$ENV{\$1} /e' < config/application.\${PACKAGE LABEL}.properties > co nfig/application.properties // 打印出生成的配置文件的内容,用于调试。实际使用过程中去除该行: cat config/application.properties // 删除临时文件及其它环境的配置项,因为产物包不需要这些 rm -rf config/application.*.properties tmpconfig // 做其它的打包工作,比如mvn package等

根据构建配置, 云效会打出两个包, 下面是 prepub 这个包的构建日志:



对照上面的 build.sh ,可以看到 config/application.properties 的值已经变成了预期的内容。

2.Helm私有仓库

2.1. 使用Helm私有仓库

点击<mark>开通私有仓库</mark>用户在云效开通私有仓库服务后,会默认打开Helm私有仓库支持

创建命名空间

当前云效每个企业最大命名空间数为5个

进入私有仓库,并进入Helm私有仓库命名空间选项,并点击"创建命名空间"按钮,按照对话框提示输入需 要创建的仓库名称。

(-)	首页	工作台	项目 👻	研发 🗸	通用 👻			新建企业x 👻	Q 搜索	+	۵	٠	Ø	zhe****@aliyun.com	简体中文
	<mark>云数私有仓库</mark> Maven仓库 > Helm仓库 ~														
			命名空间					也是时去							
			命名空间		创建于		地址						操作		
	命名空间														
	Chart制库														
				创建	命名空间			M + ENE			~				
				CSAE							^				
					*命名空间:	33874-	命名空间名称								
						** H121*3 H10*1 B07									
									确定	取	肖				

创建命名空间

创建完成后, 云效会自动为当前命名空间生成仓库访问地址, 以及独立的用户名/密码

(-)	首页	工作台	项目 👻	研发 👻	通用 👻		新建企业x ▼	Q 搜索	+	۵	٠	Ø	zhe****@aliyun.com	简体中文	
	云效私有仓库		命名空间										创建命名	空间	
	Maven仓库 >														
ι.	Helm仓库 ~		命名空间		创建于	地址							操作		
1.	命名空间		33874-test		2019-05-31 12:02:13	https://repomanage	.rdc.aliyun.com/helm_n	epositories/3387	'4-test				删除		
	Chart制库														
			操作指南												
			使用仓库												
			添加Helm仓	;库(基于安全	考虑页面上不显示密码	,点击右侧的复制按钮可以	获取包含密码的配	置)回							
			\$ export :	NAMESPACE=	33874-test										
			\$ helm re	po add \$NAI						sernam			password=*****		
			发布Char	t											
			安装Helm P	Push插件											9
			\$ helm pl	ugin insta	ll https://github.co	om/chartmuseum/helm-push	ı								
			发布Chart目	录											

仓库列表

添加Helm仓库

查看操作指南,并复制添加命令以及用户名密码(仓库用户名密码会自动复制到粘贴板)

云效

操作指南

使用仓库

添加Helm仓库(基于安全考虑页面上不显示密码,点击右侧的复制按钮可以获取包含密码的配置) []

\$ export NAMESPACE=33874-test

复制命令

```
export NAMESPACE=<Your NAMESPACE>
helm repo add $NAMESPACE https://repomanage.rdc.aliyun.com/helm_repositories/$NAMESPACE --u
sername=<Your Username> --password=<Your Password>
helm repo list # 查看已添加的仓库
```

上传Chart包到Helm仓库

云效Helm私有仓库兼容官方Helm协议,可以使用Helm-Push插件上传Chart

安装Helm-Push插件

helm plugin install https://github.com/chartmuseum/helm-push

准备Chart

```
$ helm create rdc-chart-demo
Creating rdc-chart-demo
```

使用Helm Push上传Chart

```
$ helm push rdc-chart-demo $NAMESPACE
Pushing rdc-chart-demo-0.1.0.tgz to 33874-test...
Done.
```

查看上传结果,进入云效Helm私有仓库,并进入Chart制库,如下所示:

Θ	首页	工作台	项目 ◄	研发 👻	通用 👻	新建企业x	┏ Q搜	索 +	٩	٠	0	zhe****@aliy	un.com	简体中文
	云效私有仓库 Maven仓库 >		Chart制库				全部命名空(1 ~ 请输入		
	Helm仓库 ~		应用		描述	命名王	命名空间 创建于			更新于			作	
Ŀ.	命名空间 Chart制库		rdc-ch	hart-demo	A Helm chart for Kubernetes	33874	test	2019-05-31	2:14:10	20	19-05-3	1 12:14:10 详	情	
1														

char制库

查看已上传的Chart实例,在详情中,我们可以查看当前Chart的详细信息以及相应的版本记录:

C)	首页 工	作台	项目 ▼	研发 👻	通用 ▼		新建企业x ◄	Q 搜索	+	٩	۰	Ø	zhe****@aliyun.com	简体中	文
	云效私有仓库 Maven仓库 >		< rdc-charl	t-demo											
	Helm仓库 ~ 命名空间 Chart制库		基本信息 当前版本: 0 命名空间: 3 描述: A Helm	1.0 3874-test In chart for Kube	rnetes										
			通知:云效Helm仓库兼容Helm原生仓库服务helm/chartmuseum,用户可以直接使用helm-push plugin x									×			
			操作指南	版本											
			版本				更新于								
			0.1.0				2019-05-31 12	2:14:10							
															○ 联

Chart详情

部署Chart到Kubernetes集群

Chart上传到云效Helm私有仓库之后,用户可以使用Helm命令行工具将Chart部署到Kubernetes集群:

更新版本索引:

\$ helm repo update
Hang tight while we grab the latest from your chart repositories...
...Successfully got an update from the "33874-test" chart repository
...Successfully got an update from the "stable" chart repository
Update Complete.
 Happy Helming!

查询Chart

```
$ helm search $NAMESPACENAMECHART VERSION APP VERSION DESCRIPTION33874-test/rdc-chart-demo 0.1.01.0A Helm chart for Kubernetes
```

部署Release实例:

\$ helm install 33874-test/rdc-chart-demo NAME: vocal-billygoat LAST DEPLOYED: Fri May 31 12:19:13 2019 NAMESPACE: default STATUS: DEPLOYED RESOURCES: ==> v1/Service TYPE CLUSTER-IP EXTERNAL-IP PORT(S) AGE NAME vocal-billygoat-rdc-chart-demo ClusterIP 172.19.233.76 <none> 80/TCP 1s ==> v1beta2/Deployment NAME DESIRED CURRENT UP-TO-DATE AVAILABLE AGE vocal-billygoat-rdc-chart-demo 1 1 1 0 1s ==> v1/Pod(related) NAME READY STATUS RESTARTS AGE vocal-billygoat-rdc-chart-demo-6dc7944d84-nfvrg 0/2 Init:0/1 0 0s NOTES: 1. Get the application URL by running these commands: export POD_NAME=\$(kubectl get pods --namespace default -1 "app=rdc-chart-demo,release=voc al-billygoat" -o jsonpath="{.items[0].metadata.name}") echo "Visit http://127.0.0.1:8080 to use your application" kubectl port-forward \$POD NAME 8080:80

3.Maven私有仓库

3.1. Maven仓库

制品仓库 Packages (新版)

云效制品仓库 Packages(https://packages.aliyun.com/)是阿里云出品的一款企业级私有仓库服务,提供基于 Maven、Gradle、NPM 等软件包管理工具的企业级私有仓库服务,用于管理企业级依赖托管。仓库 支持管理 Maven 制品和 NPM 制品,并支持配置远程仓库,一键迁移现有私库。提供租户隔离、权限控制、 高可用存储等服务,全面保障企业制品安全。

立即体验

Maven 私有仓库(老版)

在云效中如果需要上传、下载私有的二方库,可以使用云效的企业级Maven私有仓库服务。

C) 管理控制台	产品与服务 ▼ 首页 我的	项目 服务	Q 搜索 ・	• • •	* 0	humps and \$120 and	简体中文
Ш	服务						
🖵 工作台							
🗞 应用	已启用(8)	全部					
<i>▶</i> 测试	可拖拽改变顺序	移除服务不会丢失数据,重新启用数据仍然存在。					
р () (4) П	ŝ	发布					
/ / 17.16</th <th>应用</th> <th>🍣 🖷 😵</th> <th></th> <th></th> <th></th> <th></th> <th></th>	应用	🍣 🖷 😵					
▲ 实验室	B	应用 流水线 私有仓库					
₽ 特性分支	测试	测试					
经33 度量		<i>∂</i>					
🗇 工作项	代码	测试 实验室					
☆ 设置	<u>.</u>	开发					
* ****	实验室						
	p						
	特性分支						
	eta	其他					
	度量	æ ⁹					
		度量					
	工作项						
	私有仓库						
							1907

这样在左侧菜单栏会看到'私有仓库'链接。

Θ	管理控制台	产品与服务 ◄	首页	我的项目	服务			Q 搜索	+		۵
	Ш										
P	工作台	待办									
æ	应用				♂ 恭喜你,E	完成所有得	赤				
Ø	测试										
	代码	工作项							☑ 进入我的□	L 作项	
≖	实验室	2	任务: 1	2	任务: 1	0	任务: 0	0	任务:0		
ų	特性分支	待处理	歌函: 0 需求: 1	我创建	載昭:0 建 需求:1	我跟踪	款间:0 需求:0	我收藏	歌唱:0 需求:0		
æ	度量	■ 处理中 -	test						test	0中	
ً	工作项		hangtown	添加了注释:	现在进展如何?			2	017-11-30 21	1:38:57	
	私有仓库	【 待处理 ▼	test						test	〇中	
\$	设置										
	«										

开通仓库

虽然启用了私有仓库服务,但云效并没有真正的为您创建企业级Maven私有仓库。点击左侧菜单栏'私有仓 库'链接后,如果您是企业管理员,会出现以下界面:

c:)	管理控制台	产品与服务 🗸	首页	我的	项目	服务		Q 搜索	+	Ω	٠	0	huanga******@#35.com	简体中文
	Ш													
P														
80	应用						当前企业尚未开通仓	厍服务						
#	测试						点击开通							
	代码													
≞	实验室													
ų	特性分支													
æ	度量													
ø	工作项													
۵	设置													

'点击开通'即可开通仓库服务。

企业的普通用户并没有开通仓库的权限,则需要联系您的企业管理员进行开通操作。

开通成功以后显示界面如下:

e	管理控制台	产品与服务 ▼ 首页	我的项目 服务		•testzjidemo	- Q.搜索 + ⊆	🔉 🌣 💡 testzjidemo	简体中文
Ģ	工作台	云效私有仓库	仓库浏览				今上传文件 留远程	全库
&	应用	Maven仓库 ~						
	代码	仓库浏览	Repository	Туре	Policy	Path		
6 0	度量	制品搜索	33374-release-rCOoR4	hosted	RELEASE	https://repo.rdc.aliyun.com/repo	ository/33374-release-rCOoR4	
,	測试	Helm仓库 >	33374-snapshot-Sf43cm	hosted	SNAPSHOT	https://repo.rdc.aliyun.com/repo	ository/33374-snapshot-Sf43cm	
ø	工作项				请点击仓库列表,查看详	情		
ų	特性分支							
දිං	流水线		仓库配置 本地仓库迁移					
	私有仓库		下 北 Settings yml配罟					
٠	设置		古坡下型eettings.xml立件。					
			Elig Instraettings.vinit Citr					
			手动配置Settings.xml					
			1.在servers节点添加如下配置[]					
			<servers></servers>					

仓库地址

云效会自动为该企业生成两个Maven私库,一个是Release仓库,用于存储正式版本的二方库;另一个是 Snapshot仓库,用于存放Snapshot版本的二方库。

÷	\rightarrow C $rac{1}{2}$ https:	://repomanage.rdc.aliyun.com/	/my/repo?spm=a2c40.rdc_myind	ex.0.0.5ac62a00Jfq1	77q						\$ 0
c	管理控制台	产品与服务 ▼ 首页	我的项目服务		Otestz	jldemo ▼ Q 搜索	+	<u>ه</u>	?	testzjldemo	简体中文
Ģ	工作台	云效私有仓库	仓库浏览						令上	传文件 日日 〕	記程仓库
&	应用	Maven仓库 ~							_		
	代码	仓库浏览	Repository	Туре	Policy	Path					
6 26	度量	制品搜索	33374-release-rCOoR4	hosted	RELEASE	https://repo.r	dc.aliyun.co	m/repository/333	74-release-r0	OoR4	
ø	测试	Helm仓库 >	33374-snapshot-Sf43cm	hosted	SNAPSHOT	https://repo.r	dc.aliyun.co	m/repository/333	4-snapshot-	Sf43cm	
۵	工作项				请点击仓库列表	, 查看详情					_
¥	特性分支										
ୢୖୡ	流水线		仓库配置 本地仓库迁移								
			下 載 Settings xml西	罟							
٠	设置		直接下载settings.xml文件	ue ↓							
			手动配置Settings	cml							
			1在servers节点添加加下和雪	ā							
											<u>^</u>
			<server> <server> cidande releases clida</server></server>								

settings.xml配置

私有仓库不允许匿名上传和下载二方库, 云效为每个私有仓库生成了相应的用户名和密码。请注意不要泄露 该用户名和密码。

\leftarrow	\rightarrow C $https$://repomanage.rdc.aliyun.com/	my/repo?spm=a2c40.rdc_m	nyindex.0.0.5ac62a00Jfq77q						
e	管理控制台	产品与服务 ▼ 首页	我的项目服务。		etestzjidem	o ▼ Q 搜索	+ 4	• •	Ø	testzjldemo
		云效私有仓库	合庆浏览							
P	工作台	Maven仓库 ~	(B)+1/195							
&	应用	仓库浏览	Repository	Туре	Policy	Path				
	代码	制品搜索	33374-release-rCOoR4	hosted	RELEASE	https://repo.rdd	c.aliyun.com/rep	ository/33374	4-release-rC	OoR4
æ	度量	Helm仓库 >	33374-snapshot-Sf43cm	hosted	SNAPSHOT	https://repo.rdd	c.aliyun.com/rep	ository/33374	I-snapshot-	Sf43cm
ø	測试									
ø	工作项				请点击仓库列表,查看	详情				
ų	特性分支		合库配置 本地会	库汗移						
శ్రీ	流水线									
	私有仓库		下载Settings.x	ml配置						
\$	设置		直接下载settings.xm	1文件→						
·			手动配置Settir	ngs.xml						
			1.在servers节点添加如T							
			<pre><servers></servers></pre>	d> //username> deoSc //d> //d> //username> 下面實匠]						

用户可以通过该页面下载完整的settings.xml文件,也可以根据自己的需求在settings.xml文件中添加公共仓库的镜像地址。

上传二方库

用户可以通过UI上传二方库。目前支持通过GAV的模式进行上传,单个二方库的大小限制为300M。

\leftarrow	\rightarrow C $https:$	//repomanage.rdc.aliyun.co	m/my/repo?spm=a2	2c40.rdc_myindex.	0.0.24cc2a00bx2B9d							☆ ()
eə	管理控制台	产品与服务 ▼ 首页	我的项目	服务		testzjidemo 🔻	Q 搜索	+	•	¢ 0	testzjldemo	简体中
Ţ	工作台	云效私有仓库	仓库社	体文件						今 上传	这件 昭远	程仓库
	任务	Maven仓库 ~				>						
æ	应用	仓库浏览	Reposito	*仓库	33374-release-rCOoR4 🗸							
	代码	制品搜索	33374-re	*Group	请输入		epo.rdc.aliy	un.com/repo	sitory/333	74-release-rCC	oR4	
63	度量	Helm仓库 >	33374-sn	*Artifact	请输入		epo.rdc.aliy	un.com/repo	sitory/333	874-snapshot-St	43cm	
ø	測试											
ø	工作项			*Version	请输入							
ų	特性分支		仓库	Classifier	请输入							
&	流水线		T	*Packaging	请选择 🗸							
T	私有仓库		直	. Eile	洗搔文件 未洗择任何文件							
4	一键提升		手	*rile								
٥	设置		1.在	*生成POM文件								
				का झ	m 当							-
				N BW	E-1 X+							

检索

对二方库的检索支持关键字搜索和GAV搜索两种模式。

٣	品与服务▼自贝	则日	服券		etestzjidemo 🔻	Q 搜索 🕇	⇔ ♦	? test	zjidemo 简体屮乂
E	云效私有仓库	制品搜索			FF	有仓库 🗸 关	踺字搜索 🗸 ja	va	Q
	Maven仓库 ~								
	仓库浏览	仓库	文件名	groupId	artifactId	version	classifier	packaging	操作
	制品搜索	release	java-demo-1.0.war	com.example	java-demo	1.0		war	⊎ ⊗
1.	Helm仓库 >	release	java-demo-2.0.war	com.example	java-demo	2.0		war	⊎⊗

用户可以查看检索出来的二方库的基本信息,也可以下载二方库。

e	管理控制台	产品与服务 👻 首页	我的项目	服务		testzjidemo 🔻	Q 搜索	+		٠	Ø	testzjldemo	简体中文
							Ŷ	件详情					×
P	工作台	云效私有仓库	制品搜索				所有仓		夕称	iava-der	no-1.0 wa	r	
	任务	Maven仓库 ~							H10.				
_	动用	仓库浏览	仓库	文件名	groupId	artifactId		文件	\$大小	5.04MB			
669		制品搜索			1			最后修3	(natia	2018-12	-05 13:32	27	
	代码		release	java-demo-1.0.war	com.example	java-demo		AXUL ISK	XHJIPJ				
8 2	度量	Heim包库 >	release	java-demo-2.0.war	com.example	java-demo		下書	划地址	http://a	archiva-sto	rage-prod.oss-cn-	beijing.al
ø	测试									R4/col	n/example	/java-demo/1.0/ja	/a-demo
ø	工作项									Keyld	LTAIfU51	SusnnfCC&Signat	ure=Rtsx
ų	特性分支									521000 「日本樹	接在1小时	内有效	,
۔ چ	遊水網												
~~	0605-556							mave	n仿翰	<depen< th=""><th>dency></th><th></th><th></th></depen<>	dency>		
	私有仓库							mave	TIPONE	<gro <artif< th=""><th>upld>com. actld>iava</th><th>example-demo</th><th>></th></artif<></gro 	upld>com. actld>iava	example-demo	>
4	一键提升									<vers< th=""><th>sion>1.0<!--</th--><th>version></th><th></th></th></vers<>	sion>1.0 </th <th>version></th> <th></th>	version>	
≖	实验室									<th>ndency></th> <th>no</th> <th></th>	ndency>	no	
\$	设置												

在云效构建中使用私有仓库服务

默认情况下如果用户开通了Maven私有仓库服务,那么通过云效构建时,云效会自动尝试从该企业的私有仓 库中拉取需要的二方库,用户无需进行额外的配置。而如果用户自己在代码库根目录中定制了settings.xml 文件,那么就需要用户自行把私有仓库的配置信息添加到该文件中。具体可以通过前面的介绍来了解私有仓 库的settings.xml配置。

用户可以查看在云效构建中使用Maven私有仓库服务了解更多内容。

3.2. 公共代理库

maven.aliyun.com代理了很多公共的maven仓库。使用maven.aliyun.com中的仓库地址作为下载源,速度更快更稳定。

仓库名称	代理源地址	使用地址
central	https://repo1.maven.org/maven 2/	https://maven.aliyun.com/reposi tory/central 或 https://maven.aliyun.com/nexus /content/repositories/central

- ----

仓库名称	代理源地址	使用地址
jcenter	http://jcenter.bintray.com/	https://maven.aliyun.com/reposi tory/jcenter 或 https://maven.aliyun.com/nexus /content/repositories/jcenter
public	central仓和jcenter仓的聚合仓	https://maven.aliyun.com/reposi tory/public 或 https://maven.aliyun.com/nexus /content/groups/public
google	https://maven.google.com/	https://maven.aliyun.com/reposi tory/google 或 https://maven.aliyun.com/nexus /content/repositories/google
gradle-plugin	https://plugins.gradle.org/m2/	https://maven.aliyun.com/reposi tory/gradle-plugin 或 https://maven.aliyun.com/nexus /content/repositories/gradle- plugin
spring	http://repo.spring.io/libs- milestone/	https://maven.aliyun.com/reposi tory/spring 或 https://maven.aliyun.com/nexus /content/repositories/spring
spring-plugin	http://repo.spring.io/plugins- release/	https://maven.aliyun.com/reposi tory/spring-plugin 或 https://maven.aliyun.com/nexus /content/repositories/spring- plugin
grails-core	https://repo.grails.org/grails/cor e	https://maven.aliyun.com/reposi tory/grails-core 或 https://maven.aliyun.com/nexus /content/repositories/grails-core
apache snapshots	https://repository.apache.org/sn apshots/	https://maven.aliyun.com/reposi tory/apache-snapshots 或 https://maven.aliyun.com/nexus /content/repositories/apache- snapshots

配置指南

maven配置指南

打开maven的配置文件(windows机器一般在maven安装目录的conf/settings.xml),在 <mirrors> 标签中添加mirror子节点:

```
<mirror>
<id>aliyunmaven</id>
<mirrorOf>*</mirrorOf>
<name>阿里云公共仓库</name>
<url>https://maven.aliyun.com/repository/public</url>
</mirror>
```

如果想使用其它代理仓库,可在 <repositories></repositories> 节点中加入对应的仓库使用地址。以使用spring代理仓为例:

```
<repository>
<id>spring</id>
<url>https://maven.aliyun.com/repository/spring</url>
<releases>
<enabled>true</enabled>
</releases>
<snapshots>
<enabled>true</enabled>
</snapshots>
</repository>
```

gradle配置指南

在build.gradle文件中加入以下代码:

```
allprojects {
    repositories {
        maven { url 'https://maven.aliyun.com/repository/public/' }
        mavenLocal()
        mavenCentral()
    }
}
```

如果想使用maven.aliyun.com提供的其它代理仓,以使用spring仓为例,代码如下:

```
allProjects {
    repositories {
        maven { url 'https://maven.aliyun.com/repository/public/' }
        maven { url 'https://maven.aliyun.com/repository/spring/'}
        mavenLocal()
        mavenCentral()
     }
}
```

3.3. 将已有私库迁移至云效私库

本文档帮助您将已有Maven私库中的制品包批量迁移到云效的Maven私库中。

您可以在云效私有仓库管理界面,获取到详细的本地仓库迁移操作说明,如下图所示:

~	$\rightarrow G$ in https:	://repomanage.rdc.aliyun.com/m	ny/repo						
e	管理控制台	产品与服务 ▼ 首页	戏的项目 服务		testzjidemo	- Q 搜索	+ 🗅	¢ 0	testzjldemo
	 工作台 任务 成用	云奴私有仓库 Maven仓库 ~ 仓库浏览	仓库浏览 Repository	Туре	Policy	Path		¢۲	传文件 🔵 🖻 远禄
	代码	制品搜索	33374-release-rCOoR4	hosted	RELEASE	https://repo.rdc.ali	iyun.com/reposito	ry/33374-release-r	COoR4
æ	度量	Helm仓库 >	33374-snapshot-Sf43cm	hosted	SNAPSHOT	https://repo.rdc.ali	iyun.com/reposito	ry/33374-snapshot	t-Sf43cm
ø	测试				请点击仓库列表,查看1	羊情			
ت ات	工作项 特性分支		仓库配置 本地仓库迁移						
6g	流水线 私有仓库		下载迁移工具 migrate-local-repo-tool.jar ,↓,						
4	一鍵提升		迁移操作说明]					
≞	实验室		通过以上链接下载本地仓库迁	多工具migrate-local-repo-tool.ja	ar				
٠	设置 《		运行该迁移工具(请确保您的	JDK版本为1.8及以上),如下所	듔示:				
			\$ java -jar nigrate-local-repo-to	ol.jar-cd ~/\$HOME/.m2/repository/~ ~	t "http://10.21.11.11/repository/rele	ases/" -u admin -p adminl	23		
			参数注解:						

本地制品批量迁移

操作步骤:

1.下载迁移工具migrate-local-repo-tool.jar

2.在您本地运行该迁移工具,(请首先确保您的JDK版本为1.8及以上)。运行命令如下:

```
java
-jar migrate-local-repo-tool.jar
-cd "/$HOME/.m2/repository/"
-t "http://10.21.11.11/repository/releases/"
-u admin
-p admin123
```

参数注解:

-cd 您要迁移的本地目录,例如: /\$HOME/.m2/repository/

-t 目标仓库地址(您可以在【私有仓库】界面点击仓库地址,获取您的目标仓库地址)

-u 用户名

-p 密码

注:用户名和密码为您要上传的目标仓库用户名及密码,您可在setting.xml中获取对应仓库的username和 password

根据您的实际需求指定合适的参数,然后执行该命令,稍等片刻,您的本地仓库中的a制品将会被批量迁移 到云效中您所指定的Maven私库中。

如果迁移的本地目录中文件目录过多或者目录层级过深,可能会导致迁移命令卡死或者返回异常。推荐做法 是只迁移你自己的私有制品到私有仓库中,构建时拉取公共制品包可以使用我们提供的公共代理库。比如假 设你的私有制品都放置在 /\$HOME/.m2/repository/com/alibaba/** 目录中,你可以

将 com/alibaba/**/ 目录复制一份到一个空的目录中,比如复制到 /tmp/repo/ 中中,然后运行迁移命 令时将 -cd 命令参数指定为 /tmp/repo/ ,这样迁移工具只会迁移你的私有制品。

添加现有的私库作为云效私库的远程仓库

云效提供了关联其他仓库为远程仓库的功能。这样云效私库既具备上传下载包的能力,又具备代理其他仓库 的能力。当使用云效私库下载包时,它也会尝试从远程仓库拉取包,并且缓存在云效私库。

(-)	首页	首页	工作台	项目 👻	研发 👻	通用 ▼	服务	付起 💷 📲 👻	Q 搜索	+	۵	٠	Ø	huangb******@126.com	简体中文
E	<mark>云效私有仓库</mark> Maven仓库 ~		仓库浏览								(⑦ 使月	用指南	今 上传文件	程仓库
	仓库浏览		仓库名称		ŝ	类型	策	ŝ	路径						
	制品搜索 Helm仓库 >		4373-release-2	ZiGm8 ¹⁴ 设置	远程仓库⑦	hantad	D		http://4	4 400 44	0 1 AE /wa >	 itory 	//4373-r	elease-ZiGm8K	
			4373-snapsho	it-gmZ		*仓库: 4373-	release-ZiGm ⁹ K					itory	//4373-s	napshot-gmZH4E	
					*远程仓库均	地址 ⑦: 「请输ン	ι								
			私有仓库(吏用指	远程仓库则	账号 ⑦: 【请输】	λ								
			Maven配置	指南	远程仓库图	密码 ②: [请输)	ι								
			下载Se	etting		关联仓	库 取 消								
			直接下载s	etting											
			或者手	动配置修改	Settings	.xml									
			1.在server <servers></servers>	rs节点添加如T	配置喧										
			<server> <id>rot <userver> </userver></id></server> <server> <id>rot <userver> <userver></userver></userver></id></server>	tc-releases name>v44 wo. J>Jx5p1W > tc-snapshotsname>0	username> GVg>	>									

设置远程仓库

远程仓库地址为您想要代理的私库地址,这个地址必须是公网可以访问的。如果该私库可以匿名访问,那么 无需配置访问账号和密码。点击关联仓库可以保存配置。您随后可以修改配置或者解除关联。注意只有已经 缓存在云效私库的制品包才能被搜索到。注意:如果远程仓库的网络环境不佳会导致拖慢云效私库的下载速 度。

3.4. 在云效构建中使用Maven私有仓库服 务

当用户开通了Maven私有仓库服务后,云效会为用户生成两个私有仓库,一个用于存放release版本的二方库,一个用于存储SNAPSHOT版本的二方库。

release仓库地址示例:

https://repo.rdc.aliyun.com/repository/24409-release-87w1FL/

SNAPSHOT仓库地址示例:

https://repo.rdc.aliyun.com/repository/24409-snapshot-AA0Hx0/

如果用户项目代码库的根目录没有Maven的settings.xml文件,那么云效构建时会为用户自动生成一个 settings.xml文件。该文件不仅包括了maven.aliyun.com等公共仓库地址,也自动引入了该企业的两个私有 仓库地址。所以用户通过云效构建时,无需任何额外配置就可以实现下载私有仓库中的二方库。

如果用户项目代码库的根目录定制了Maven的settings.xml文件,那么用户需要自行将私有仓库的配置信息 添加到该文件中。具体可以参考Maven私有仓库服务。

通过云效上传二方库到私有仓库

如果想通过流水线发布二方库到私有仓库,可以先在项目代码库根目录的pom.xml中指定分发的仓库地址, 示例如下:

```
<distributionManagement>
<repository>
<id>releases</id>
<url>http://repo.rdc.aliyun.com/repository/24409-release-87w1FL/</url>
</repository>
<snapshotRepository>
<id>rdc-snapshots</id>
<url>https://repo.rdc.aliyun.com/repository/24409-snapshot-AAOHx0/</url>
</snapshotRepository>
</distributionManagement>
```

项目代码库根目录的 <应用名>.release 中指定构建命令为上传二方库,例如:

```
build.command=mvn clean deploy -Dmaven.test.skip
```

如果你的pom.xml配置的软件包版本是以 -SNAPSHOT 结尾,比如版本为 1.0-SNAPSHOT ,会自动发布到 snapshot仓;如果想发到release仓,可以将命令改为 build.command=mvn clean deploy - Dmaven.test.skip -P release 。

然后在云效中创建一条流水线,创建一个构建任务。示例配置如下:

的机力和	构建		nonze obict	
別段白你	79)建	01-71 [E\$F10	別 校正 三 日 ムリ かし キマ シ	
E务设置				
	1. 构建1 ×	任务类型	构建	
	十 添加任务	1200702	172	
		*任务名称	构建1	
		*代码库地址	24409-archiva/archivatest.git	~
		*分支名称	master	\sim
		*构建配置	archivatest.release	~
		上传构建包 ②	\bigcirc	

构建

注意:需要将上传构建包参数关闭。

也可以在构建命令中指定分发的仓库地址。方式是在Maven命令中指定 -DaltDeploymentRepository 参数。

build.command=mvn -DaltDeploymentRepository=rdc-releases::default::https://repo.rdc.aliyun. com/repository/24409-release-87w1FL/ deploy -Dmaven.test.skip alt Deployment Reposit ory指定了 id::layout::url 。在云效的Maven私有仓库服务中release仓库的id 为 rdc-releases 。SNAPSHOT仓库的id为 rdc-snapshots 。layout一般使用默认值default,而url则为 release仓库或SNAPSHOT仓库的url。上传到SNAPSHOT仓库的示例命令如下:

build.command=mvn -DaltDeploymentRepository=rdc-snapshots::default::https://repo.rdc.aliyun .com/repository/24409-snapshot-AA0Hx0/ deploy -Dmaven.test.skip

单应用同时支持应用构建和二方库发布

如果一个项目既要实现打包和部署,又要为其他项目提供SDK二方库,那么单个 build.command 配置就无 法满足这种场景。您可以使用传入参数改变构建行为中的方式,使用 PACKAGE_LABEL 区分不同的构建命 令。一个完整的例子如下。

在代码库根目录的 <应用名>.release 文件中指定如下配置项:

deploy.build.command=mvn -DaltDeploymentRepository=rdc-releases::default::https://repo.rdc. aliyun.com/repository/24409-release-87w1FL/ deploy -Dmaven.test.skip

这个配置项使用前缀 deploy 作为包标签。然后创建一条流水线,在构建任务中打开高级配置项,进行如下配置:

1+212170	仰建	加特到下一时间		
予 设置				
	1. 构建1 ×	任务类型 枚	建	
	十 添加任务	*任务名称	勾建1	
		*代码库地址	24409-archiva/archivatest.git	~
		*分支名称	naster	~
		*构建配置	rchivatest.release	~
		上传构建包 ⑦		
		√ 高级设置		
		─ 高级面 第1个		
		色	标签 deploy ①	

构建任务带标签

与上一个构建任务唯一不同的是这里将高级配置中的包标签的值改为 deploy ,这样触发构建时执行的就 是 deploy.build.command 中指定的命令。