目录

目录 1
使用技巧 5
角色与权限 5
企业设置 5
全局设置 5
成员设置 5
用户组 5
通知与待办 5
个人中心 6
个人设置 6
SSH公钥 6
访问令牌 6
放在http请求参数中使用(使用简单) 7
或者放在http请求头中使用(更安全) 7
服务集成 7
服务使用权限 7
Doker镜像源 7
SonarQube 7
产品介绍 7
使用准备 7
主界面 7
创建代码库 7
代码下载与提交 8
在线浏览操作与编辑 8
查看代码质量问题 9
代码历史 9
对比 9
分支 10
评审 10
评审的类型 10
评审列表 10
评审创建 10
评审详情 10
DCR的更新 11
DCR的特点 11
DCR和MR结合 11
统计 11
设置 11
基本设置 11
提交规则 12
成员与角色 12
冲突检测 12
运维操作 12
安全设置 13
操作审计 13
二级目录设置 13
产品介绍 13
相关术语 13
角色与权限 14
成员 14

- 访客 14
- 资源使用权限 14

执行记录 14
流水线列表 14
手动执行 14
新建与编辑 14
- 基本信息配置： 15
- 通知提醒： 15
- 环境变量： 15
- 阶段&任务： 15

流水线复制 15
流水线插件 16
- 编译类插件 16
- 发版插件 16
- 主机部署插件 16
- Sonarqube插件 17
- Maven单测插件 17
- JUnit插件 18
- Jacoco覆盖率插件 18
- Cobertura覆盖率插件 18

构建集群 19
- 集群导入 19
- 集权权限 19
- 集群使用 20
- 集群编辑 20

产品介绍 20
相关术语 20
- 半成品库 20
- 成品库 20
- 镜像库 20
- 包和版本 20

半成品库 20
- 内置库与新建 21
- 权限与角色 21

成品库 21
镜像库 21
- 镜像源设置 22
- 镜像库工作原理 22
- 镜像清理 22
- 镜像库权限 22

自动代理及读写分离 22
产品介绍 22
相关术语 22
角色与权限 22
主机组管理 22
- 新建 22
- 编辑 23
- 删除 23

部署任务 23
- 部署任务 23
- 部署任务状态 23
- 重新部署 23

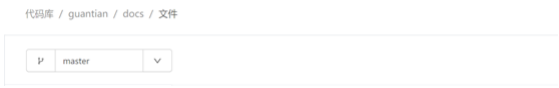
回滚 23
新建 23
流水线自动部署 24
部署模板 24
基本设置 24
主机设置 24
任务设置 25
主机列表 25
新建主机 25
代理机设置 25
产品介绍 25
相关术语 25
项目列表 25
创建项目步骤 26
计划与看板 26
列表视图 26
看板视图 26
查看归档计划 27
卡片查询 27
卡片查询步骤 27
卡片回收站 27
项目设置 27
基本设置 28
成员设置 28
运维操作 28
卡片类型 28
流程状态 28
字段集 28
卡片设置 29
字段设置 29
内容模板 29
工作流 29
产品介绍 30
产品术语 30
测试空间管理 30
用例管理 30
套件管理 30
测试执行 31
产品介绍 31
相关术语 31
角色与权限 31
K8S集群的角色与权限 31
环境的角色与权限 31
集群管理 32
导入 32
编辑 33
环境管理 33
新建 33
编辑 33
删除 33
审计 33
PV管理 33

访问模式 34
新建NFS类型PV 34
删除 35
查看 35
部署管理 35
部署模板 35
新建部署模板 35
权限设置 36
容器设置 36
健康检查 37
部署任务 37
任务状态 38
部署实例 38
Pods 38
Events 39
更新实例 39
删除 39
ConfigMap 39
密文 39
服务 39
域名 40
证书 41
PVC 41
自动化部署 41

使用技巧

平台整体使用方面主要有如下技巧，希望用户朋友在阅读之前有一个快速了解：

1. 可以用用户组功能简化权限配置，详情参考本手册“用户组管理”章节
2. 系统所有功能API，均可使用个人token用脚本自动化换成，详情请参考本手册“个人中心>访问令牌”章节
3. 使用顶部导航（通常称为面包屑导航），实现功能快速定位和切换。

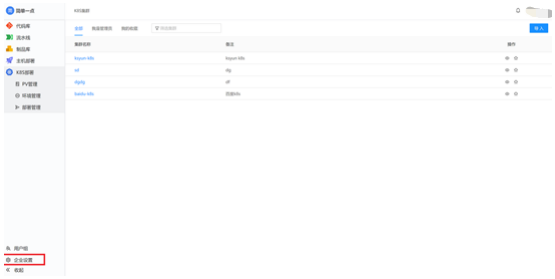


角色与权限

角色分为企业管理员和普通用户两种：
企业管理员：具有企业的管理权限以及各子产品的管理员权限。
普通用户：普通用户不具有企业管理权限，其具体权限是在各子产品中定义。例如：某个普通用户可以作为某代码库管理员，对该代码库具有管理员权限。

企业设置

企业设置是企业管理员对企业进行统一设定的功能。主要包括全局设置、成员设置等功能。此功能菜单位于系统功能界面左下角，只有管理员可见。



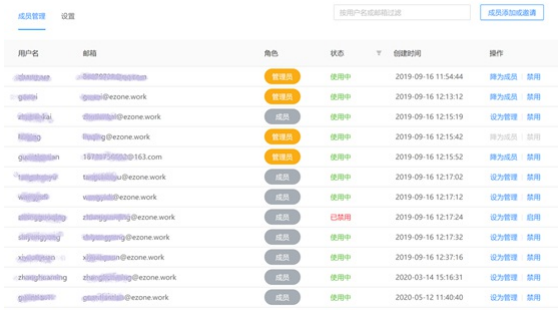
全局设置

全局设置功能较为简单，主要设置企业的一些基本信息。

成员设置

成员设置功能较多，主要包括“成员管理”和“设置”两个标签页。

成员管理：



在成员管理中，管理员可以设置其他成员的角色、或者对成员账号进行禁用操作。被禁用的账号将不再能够访问系统。

设置：
企业管理员可以在“设置”标签页，对企业的账号注册和邀请策略进行统一设置，以保障企业的账号注册安全。

用户组

为了简化权限管理，内置了用户组管理功能，这样组内有同事离职入职，至需要变更用户组成员即可，不需要为该同事单独添加权限，非常简单。
用户组功能菜单位于系统操作界面左下角：分为我所在的、我管理的、系统内置三类用户组。

系统内置了all和user-all两个用户组，分别表示企业所有成员及所有非管理员用户。
在“我管理的”标签页，可对名下有管理权限的用户组进行成员管理。

通知与待办

作为协同平台的基础功能,内置了通知与待办功能，位于系统界面右上角区域。用一个铃铛表示。铃铛有红点时，表示有未读的通知或者未完成的待办。

个人中心

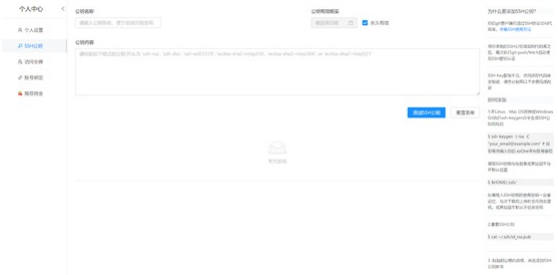
在系统界面右上角，鼠标移动到个人头像或用户名区域附近，即可显示个人中心的下拉菜单：

个人设置

个人设置中提供了基本的账号信息设置功能，如邮箱、手机、用户名（只读）、密码等。

SSH公钥

使用场景：SSH公钥功能仅用于代码下载，不可用于其他用途。方便下载代码的时候每次都输入密码，以及用账号密码的方式密码泄露的风险。该密钥是和机器绑定的（每台机器的SSH公钥不同），即使公钥泄露，通过别的机器依然无法使用该公钥下载代码。具体使用办法参见界面右侧说明区域。



访问令牌

使用场景：常用于自动化脚本、不需要暴露用户密码，且不希望该脚本具有用户的所有权限。
使用范围：ezOne所有的API均可以使用。
使用步骤：

1. 在个人设置中创建权限适宜的令牌

2. 通过浏览器插件，获取需要的API相关信息、URL、方法(GET、POST、PUT、DELETE)，其中GET方法仅需读权限，其他方法需要写权限

2. 使用如下方式调用API(此处以GET方法为例)，将令牌信息放在请求参数或者http请求头中(此种方式比放在URL的参数中更安全)，如果用户在请求参数和http请求头中放了俩不同的token，我们会以请求参数中的为准

放在http请求参数中使用(使用简单)

curl -X GET "https://xxx.ezone.work/xxxxxxxx?access_token=令牌字符串"

或者放在http请求头中使用(更安全)

curl -X GET "https://xxx.ezone.work/xxxxxxxx" -H "access_token:令牌字符串"

服务集成

企业效能平台本身是一个涵盖面非常广的大型分布式平台，但是整个企业研发场景可能涉及第三方的服务和平台，服务集成功能就是为解决此类问题提供的功能。点击左侧“服务集成”菜单即可进入到服务集成功能界面。目前提供了Docker镜像源和SonarQube服务配置。

服务使用权限

服务的使用权限分为企业内公用和保密。服务的连接信息配置好以后，会保存在系统中，当其他用户需要使用该服务的时候通过系统鉴权。
企业内公用的服务对于企业内的所有成员开放，保密服务只有服务配置中添加的成员可用。

Doker镜像源

企业效能平台本身已经提供了内置的Docker制品库和Docker镜像库功能，所以除非有特殊情况，一般情况下无需配置第三方的Docker镜像源。
第三方Docker镜像源在配置后可以在需要Docker镜像的场景下使用：例如流水线ezPipeline的docker镜像执行器插件配置、ezK8S容器部署系统的部署模板配置。详情请参考ezPipeline Docker插件及ezK8S容器部署系统相关章节。

SonarQube

Sonarqube服务器配置后，目前可以在流水线ezPipeline的SonarQube扫描服务插件中使用。下图的Sonarqube服务添加界面中的Token是Sonarqube服务的用户token，不是企业效能平台系统的用户Token，请勿混淆。

产品介绍

代码托管是企业效能平台产品系列中的代码托管产品，为用户提供Git代码托管，分支和版本管理；在线浏览、在线编辑、代码评审、代码扫描的能力。
代码托管的代码库以“目录/代码库名”的形式组织，方便在企业中不同的产品线、不同的部门将代码放在某一个目录下进行集中管理和批量设置。

使用准备

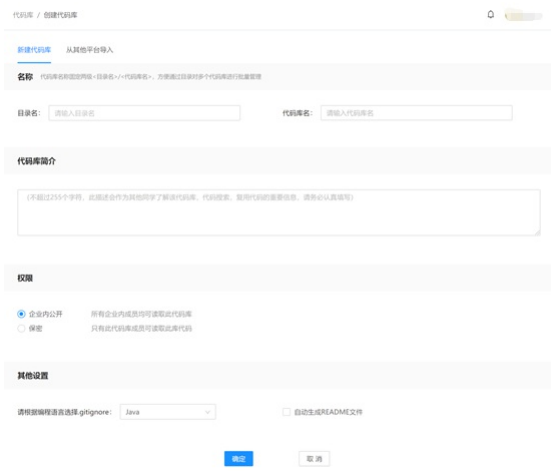
使用前请在“个人中心”设置好个人密码即可支持通过账号密码方式下载或者提交代码； 在“个人中心”设置好ssh公钥即可在机器上(每台机器的ssh公钥不同)直接下载和提交代码。

主界面

点击左侧“代码库”主菜单即可显示代码托管主界面，主界面以“全部”、“我是管理员”、“我的收藏”三个标签页分别显示了代码库列表的分组。而且系统会自动记住你上次的选择，方便下次使用。第一次进入主界面时不会出现代码库下的功能菜单，当选择一个代码库进入后，会显示代码库操作的二级菜单。

创建代码库

代码托管主界面右上角“创建代码库”按钮可进行代码库的创建，创建可以新建或者导入。



导入功能支持从SVN或者Git仓库进行，对于大库，建议用客户端直接用git push（Git库）或者git svn clone(SVN库)命令直接推送。

代码下载与提交

通过主界面代码库列表的代码库URL可以获取代码库通过ssh协议或者https协议进行下载的地址，并通过git push命令进行代码提交。
注意：如果代码库分支被设置成了保护分支，则通过git push命令向该分支提交代码时，会被服务器端拒绝并自动生成一个DCR(direct code review)类型的代码评审。

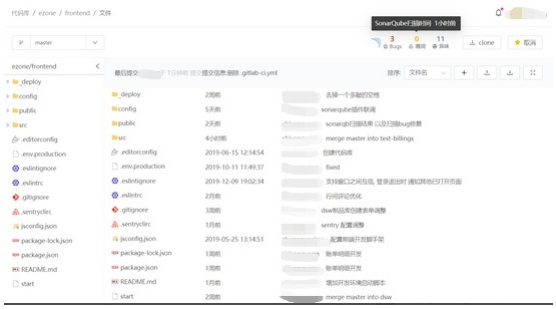
在线浏览操作与编辑

通过“文件”菜单，可以对代码库文件进行在线浏览、文件上传、下载、删除、文件在线编辑等功能，并可对代码库进行收藏、取消收藏。

对代码文件可以进行在线编辑，markdown文件还支持预览模式，右上角缩略图支持快捷拖动。

查看代码质量问题

如果在流水线中配置了Sonarqube代码扫描插件，则该插件运行结束后，会将代码扫描结果推送到代码库对应分支的文件浏览页面，点击即可查看扫描详细报告。



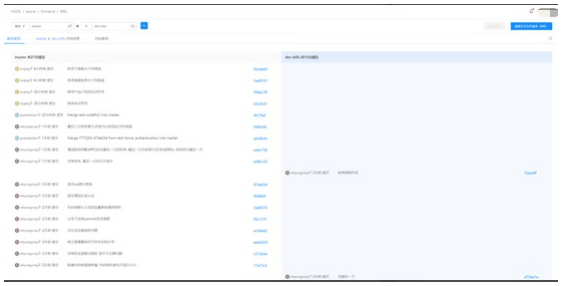
代码历史

通过“历史”菜单，可以查看不同分支和Tag上的commit历史；
点击commit或者文件夹按钮，可以浏览该commit对应的文件；
点击某个文件，可以查看该文件的历史。

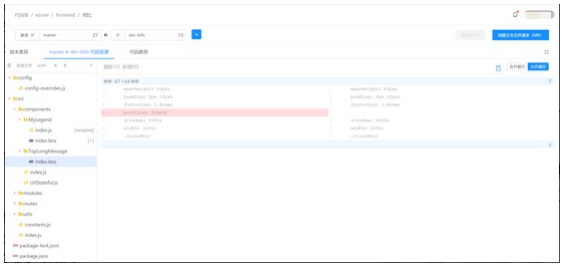
点击commit可以查看该commit代码变更详情，并进行一些常见操作。

对比

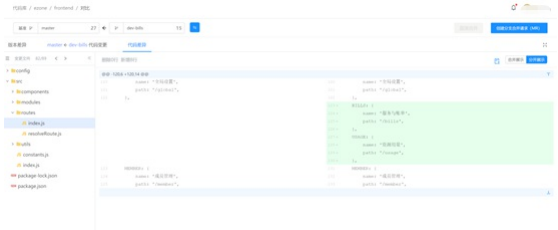
比视图提供任意分支和Tag之间的比较，并且提供三种视图： “版本差异视图”、“单向代码变更视图”和“代码差异视图”。
版本差异视图： 提供不同分支或者Tag的commit差异,例如本例左侧显示master分支上的27个特有commit、右侧显示dev-bills分支上特有的15个commit。



单向代码变更视图： 提供右侧分支或者Tag的最新commit相对和左侧Base commit(也即24个commit之前的那个commit，左右的共同祖先commit)的代码变更。

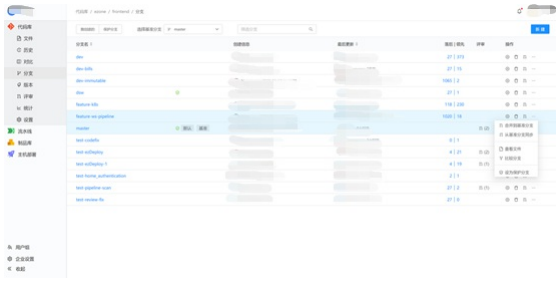


代码差异视图： 提供左右两边分支或者tag的最新commit之间的代码差异。



分支

分支视图提供了对分支的管理能力：新建分支、删除分支、合并、比较、设为保护分支等功能，显示不同分支之间commit的相对领先落后数量，以及当前该分支上尚在进行中的评审。可以有效的防止误删尚有进行中的评审的分支。点击评审，即可在右幅窗口显示具体的评审。



评审

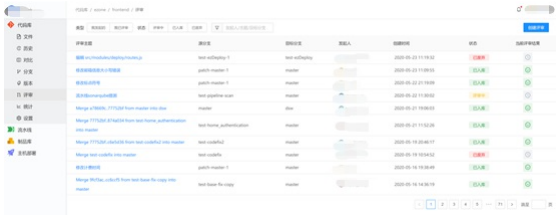
评审的类型

支持如下两种代码评审模型的代码评审系统

1. MR（Merge request）业界典型的分支合并模型
2. DCR(Direct code review) 直接代码评审，提交代码自动生成评审，无需特殊客户端、无需特殊命令。例如下图：

评审列表

可以按照不同的状态和类型对代码评审进行筛选，并直观的显示评审的各项信息。

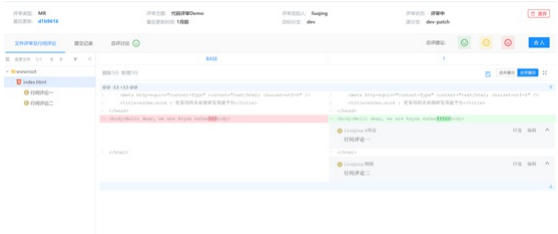


评审创建

可通过下列方式创建代码评审
1、通过评审列表右上角的“创建评审”进行创建MR
2、在线编辑保护分支上的文件，保存时会自动生成临时分支并且创建MR
3、通过分支页面，进行分支合并操作时，自动创建MR
4、通过IDE插件或者Git原生客户端用git push命令向保护分支提交代码，自动生成DCR
5、通过IDE插件或者Git原生客户端通过git push -o dcr命令向非保护分支提交代码，自动生成DCR

评审详情

评审详情界面显示了该评审相关的所有信息，评审人非常方便的进行行间评论并且在右上角进行总评建议。方便的针对不同的commit进行左右对比评审。
并且提供了关于评审的许多便捷操作，例如全屏、过滤、显示全部等。
此外：评审详情提供了代码视图（默认视图，方便快速进行代码评审），还提供了提交记录、总评讨论视图。



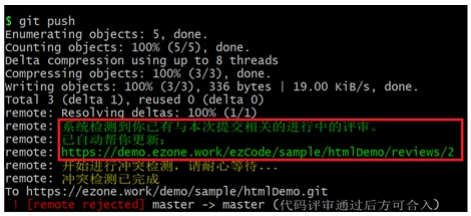
提交记录视图可以方便的查看该评审中的代码commit记录。

总评讨论视图展示了总评建议的给出情况，并且可以方便的向评审人发送评审通知（点击右侧铃铛图标）。

DCR的更新

对于DCR，由于没有一个可见的源分支实体，如果评审被打回，想继续修改，继续更新评审可以使用如下方法：

1. 在本地开发环境上直接修改，直接push，非常简单。系统会自动识别并更新当前评审。



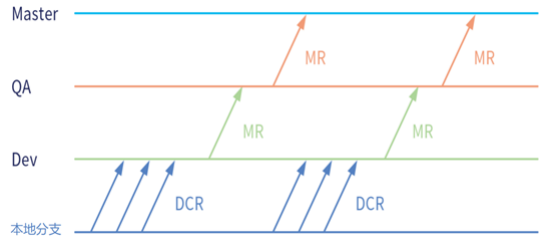
2. 如果更换了电脑，失去了开发环境，则可以通过评审详情页面先将代码fetch到本地，然后修改编辑后提交即可。

DCR的特点

1.评审粒度更细：相对于MR分支合并粒度的评审，动辄数个文件，上千行代码，DCR更适合企业研发场景提倡的少量多次提交，持续集成与交付。
2.工程师使用更便捷：无需登录网页端，直接用IDE原生插件或者Git原生客户端直接即可发起；不用脱离开发环境，导致工作思路中断。
3.企业内开源协同只适合用DCR： MR需要赋予贡献方分支创建与合并、向源分支直接提交代码的权限，故会导致企业的代码安全无法维护；开源协同的PR模式不仅需要Fork代码，导致企业代码到处被复制混乱不堪，而且开源贡献者还需要在新Fork出来的代码库上搭建流水线，从而进一步阻碍其贡献意愿。DCR只需要将企业内需要开源协同的库设置为“公开”类型、企业其他成员即可直接向代码库发起DCR贡献代码（企业其他成员即使向非保护分支提交代码，也会被自动拦截生成DCR）。

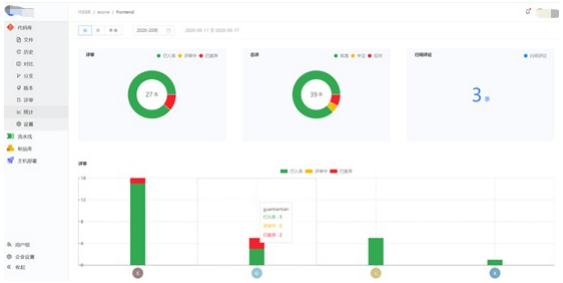
DCR和MR结合

在日常开发中，完全可以结合DCR和MR优点共同使用，例如下图示例：
工程师在本地分支开发，频繁使用DCR向远端dev分支提交代码评审；
某feature开发完毕，发起Dev分支向QA分支的MR，通知测试人员提测；
测试人员经过QA分支上的自动化测试及其他测试后，发起从QA分支到Master分支的MR，在Master分支发布版本、上线。



统计

统计视图提供了按不同时间维度查看代码评审提交数据的功能：

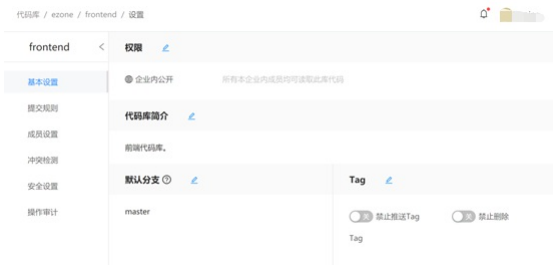


设置

代码托管提供了强大的代码库设置能力。包括基本设置、提交规则、成员设置、冲突检测、安全检测、操作审计等。

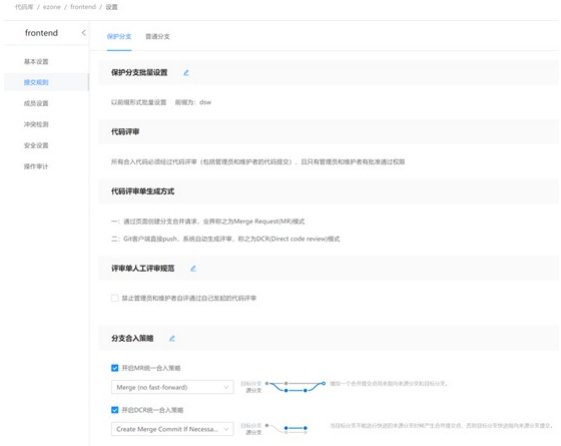
基本设置

基本设置可以对代码库的权限类型、说明、以及基本的默认分支、Tag的操作进行设置。



提交规则

提交规则可分别对保护分支和普通分支进行详细的提交及合入规则设置。
保护分支： 合入代码必须经过代码评审、且只有代码库管理员和维护者批准通过、无其他人反对才可以合入的分支。
普通分支： 除了保护分支之外的分支均属于非保护分支。



成员与角色

ezCode保密代码库的成员分为三种角色：管理员、维护者、开发者。
ezCode公开代码库的成员分为四种角色：管理员、维护者、开发者、访客。
访客仅有公开库读权限；向公开库任何分支Push代码系统会自动创建代码评审，且只能由代码库成员评审及合入。
各角色权限的详情请参考平台“角色权限详情”标签页。

冲突检测

在团队协同多分支开发场景下，代码合并时的冲突是一件令人十分沮丧的事情，ezCode专利技术提供代码提交时自动进行分支冲突检测的功能。不管是向非保护分支直接提交代码，抑或是通过DCR向保护分支提交代码，都会自动自动触发冲突检测预警。

运维操作

运维操作目前提供代码库GC（垃圾清理）、代码库删除两种操作。

安全设置

目前安全设置仅提供对代码库进行删除的慎重操作功能，后续会提供更多安全设置。

操作审计

在企业级场景下，对代码库操作的行为可审计是非常必要的功能。此功能提供了按时间段、操作类型和操作人对代码库操作进行审计查询的能力。

二级目录设置

通过系统顶部的面包屑导航，可以切换到二级目录设置界面。二级目录设置为部门或者团队统一管理其代码库权限提供了方便的入口。

产品介绍

流水线是企业效能平台产品系列中的持续交付流水线及构建资源管理产品，为用户提供流水线编排、自动化任务插件、任务的调度与执行，任务资源自动创建、调度与释放的能力，实现高效率持续交付。

相关术语

流水线： 来源于工业自动化生产领域，表示由ezPipeline编排的一系列自动化任务，完成从代码提交到部署上线的研发流程。
阶段： 流水线术语，对应英文Stage，常表示某一个研发过程，常由若干个自动化任务组成。例如一条流水线可以由编译、代码扫描、测试、上线等阶段组成。
任务： 流水线术语，对应英文Job，流水线上处理任务的最小单元。例如测试阶段可由部署测试环境、API测试、压力测试若干任务组成。
构建： 使用较为宽泛，流水线的一次执行、job的一次执行可以成为一次构建。构建类任务通常指生成了制品的任务，但是并不严格。
制品： 将源码打包处理后用以被依赖使用或者运行的文件被称为制品。例如程序包、Jar、war、docker镜像等
构建号： 流水线的每次执行，会生成一个构建号，用以识别.
构建版本： 每次流水线执行自动生成的版本，根据国际通行的语义化版本号约定，构建版本以四位数字版本号命名。
发布版本： 通过流水线发版插件生成的正式版本，通常是测试通过后才会发正式版本。根据国际通行的语义化版本号约定，发布的正式版本以三位数字版本号命名。
下图是流水线的一次执行记录，相关术语标注如下：

插件： ezPipeline中的任务，都是由对应的流水线插件实现的，方便用户可以通过插件对流水线功能进行扩展。
环境变量：流水线每次构建时，为方便自动化任务通过脚本进行操作，内置了一批环境变量（系统环境变量），并且用户还可以自行配置环境变量（用户自定义环境变量）。环境变量的使用范围是在整个构建过程的所有任务。
其中系统环境变量包括：
SYS_BUILD_REPO 代码库名称
SYS_BUILD_BRANCH 分支名
SYS_BUILD_COMMIT commitID
SYS_BUILD_SNAPSHOT_VERSION 构建版本号
SYS_BUILD_RELEASE_VERSION 发布版本号
SYS_PIPELINE_NAME 流水线名称
SYS_PIPELINE_BUILD_NUMBER 流水线构建号，流水线范围内自增
SYS_PIPELINE_BUILD_ID 流水线构建id，ezPipleine全局唯一
SYS_TRIGGER_USER 流水线触发人

角色与权限

流水线的角色与其关联的代码库角色密切相关。主要分为如下两类：

成员

成员默认是代码库成员，具有流水线的所有权限。
后续会支持在代码库成员基础上添加其他人员，以解决一些企业的测试及OP没有代码库权限，从而无法对流水线进行操作的需求。

访客

默认是代码库访客，仅具有流水线的只读权限；但不能查看编译日志
仅企业内公开的代码库对应的流水线存在访客。

资源使用权限

流水线在执行时，需要使用各类资源，例如构建集群、主机组、制品库等。这些都属于第三方资源，这些资源都是在真正使用时，才调用第三方系统API进行鉴权。
对于代码提交自动触发的流水线：使用代码提交者身份
对于定时触发的流水线：使用流水线最后编辑人的身份
对于手动触发的流水线：使用手动触发人的身份
对于API触发的流水线：使用个人令牌主人的身份

执行记录

执行记录显示了最近的流水线构建记录，并且支持按代码库、分支和流水线名称筛选。如果流水线的第一个阶段在设置的时候设置为手动触发，则流水线会自动生成一个空的执行记录，如下图#440号构建。方便在需要的时候点击执行。 现代软件研发的最佳实践提倡每一次变更都有自动化构建。

流水线列表

流水线管理主界面提供了流水线列表及对流水线的常用操作：如新建流水线、手动执行、编辑、查看执行记录、复制和删除。

手动执行

值得注意的是：所有流水线默认都是支持手动执行的。点击执行按钮，即可弹出手动执行的确认框。选择对应的分支后，流水线即在该分支的最新commit上执行构建。

新建与编辑

流水线新建和编辑功能界面基本相同，目前包括四个部分：基本信息、通知提醒、环境变量、阶段&任务。

基本信息配置：

主要配置流水线名称、对应的代码库（新建时可选，编辑时不允许修改）、关联的分支信息以及触发条件。
关联分支信息：支持全部分支关联及部分分支关联。部分分支关联又支持精确匹配、前缀匹配和后缀匹配三种形式。
触发条件支持：支持代码库变更触发及定时触发。手动触发无需配置，全部支持。对于定时触发，还支持在有代码变更时才执行。

通知提醒：

通知提醒功能可以自定义设置在任务执行的过程中，哪些事件会被通知到相关人员，以及任务超时的设置。

环境变量：

除了系统内置的环境变量，ezPipeline支持用户添加自定义的环境变量。且环境变量在整个流水线的所有任务中均可用。
环境变量的使用：可以在流水线的插件脚本中通过${环境变量名}的方式使用。
对于保密信息，可以通过密文环境变量实现。

阶段&任务：

此区域支持配置流水线的阶段、并且在每个阶段中支持串行&并行任务的配置。支持拖拽。
每个阶段的触发方式支持自动触发（按基本信息设置区域中的自动触发规则）或者手动触发。

流水线复制

当前仅支持在同一代码库下复制流水线。
注意：流水线复制时，会将密文环境变量自动剔除，不参与复制。

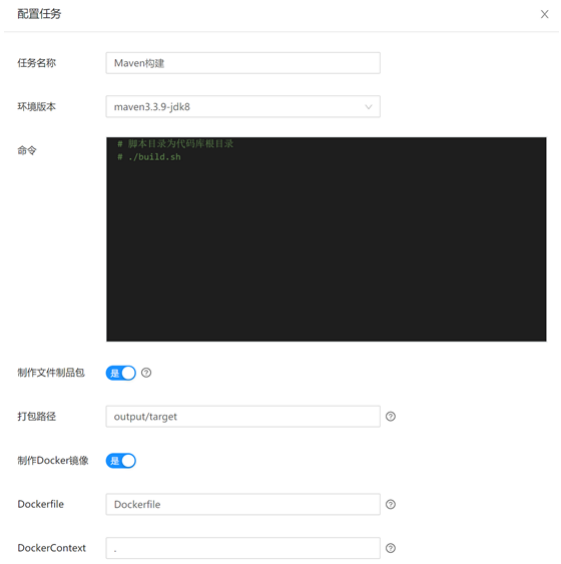
流水线插件

编译类插件

添加任务选择插件时，类型选择 ”编译构建”即可查看到所有的编译类插件。这些插件的共同特点是会自动完成如下事项：
自动下载代码 > 执行编译脚本 > 自动上传制品（仅自动上传文件类型和docker类型制品）。
其中自动下载代码会使用ezCode的专利技术，仅下载该构建任务对应的commit，而不是将代码库全部下载下来。

下面以maven构建插件为例说明：
设置任务名称；
需要的maven版本号；
运行的编译命令（可以将编译脚本放在代码库中例如名字叫做build.sh，此处直接通过./build.sh执行该脚本）
如果需要上传jar或者war文件、则在编译命令中使用mvn deploy命令上传，系统会根据pom.xml中的配置上传到指定制品库；

文件类制品：打开制作文件制品开关，插件会自动打包并上传构建产出到ezPackage文件类半成品库（半成品库中版本号即是构建版本号），只需要在构建脚本中将需要打包的文件放在指定路径下即可。
Docker镜像：打开制作docker镜像开关，插件会自动打包并上传docker镜像到ezPackage的docker半成品库（半成品库中版本号即是构建版本号）

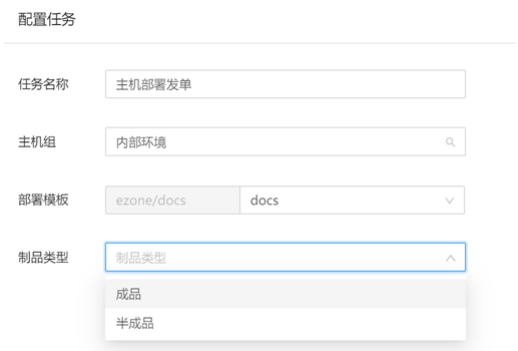


发版插件

通常软件开发会在代码经过测试后的稳定版本进行版本发布。存档为一个稳定的版本。此时可以使用发版插件来完成。
发版插件会自动完成如下事项：
基于当前流水线构建对应的代码库commit，给代码库打Tag
自动将文件类半成品发布为成品（根据当前构建号对应的构建版本号去半成品库查找，如果没有则放弃）
自动将docker类的半成品发布为成品（根据当前构建号对应的构建版本号去半成品库查找，如果没有则放弃）

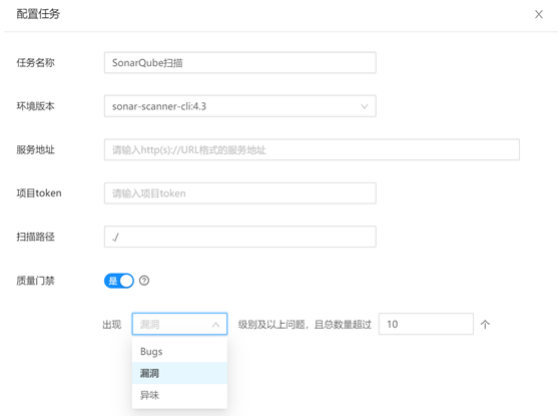
主机部署插件

主机部署插件的功能是自动创建ezDeploy部署任务。
使用主机部署插件之前，需要先在通过流水线配置好主机组、并且在主机组中配置好部署模板。
在主机部署插件中，只需要简单的选择需要将应用部署到哪个主机组、使用哪个部署模板、选择半成品制品还是成品即可。
如果选择半成品，则插件会自动根据构建版本号创建流水线部署任务单。
如果选择成品，则插件会自动根据发布版本号创建流水线部署任务单。



Sonarqube插件

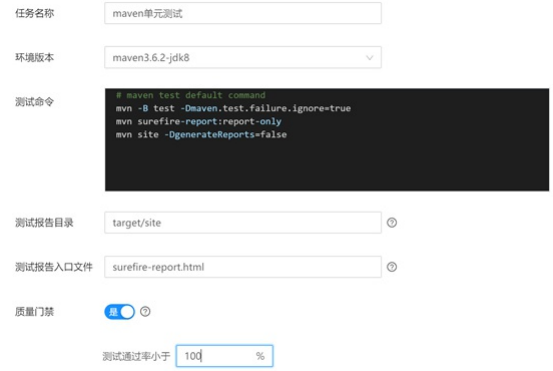
代码扫描是当前流水线环节的重要一步，ezPipeline内置了Sonarqube扫描插件，只需要简单配置下Sonarqube信息，即可进行扫描。并支持质量门禁。
项目token是用户在sonarqube服务器上的令牌
项目名称默认和代码库名称相同；项目版本默认是流水线的构建版本号；



执行结果如图：

Maven单测插件

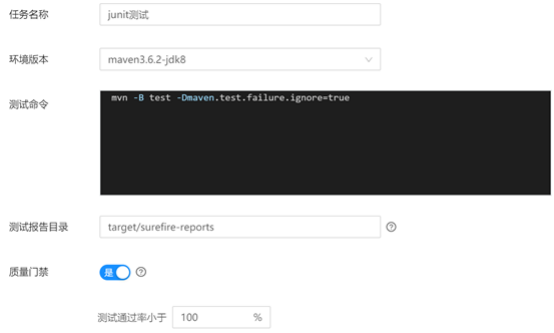
Maven单测插件提供了执行maven单测任务、生成测试报告、查看结果数据等功能。
插件基本配置： 在下图的插件配置中，只需要配置测试命令、测试报告生成的相对路径、测试报告入口文件即可运行。
质量门禁： 此处可设置当测试通过率低于某数值时，测试任务会被认为失败。



执行结果样例如下：鼠标移动到图示位置，可以查看运行结果，点击“查看报告”，可以查看报告

JUnit插件

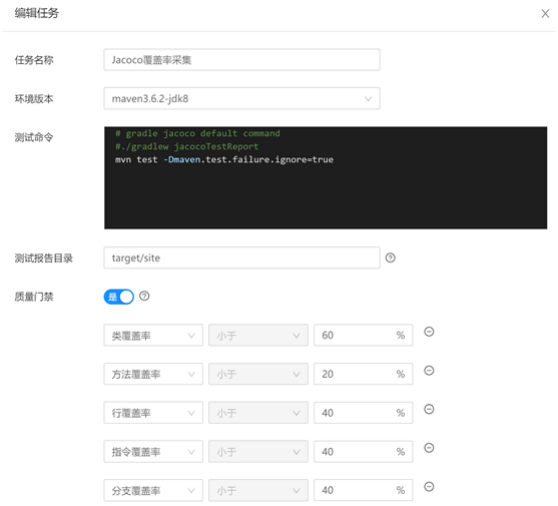
JUnit插件相对简单，只提供了Junit运行、测试结果查看功能。
插件基本配置：如下图所示，和Maven单测插件相比，只需要设置测试报告目录用以服务端对执行结果进行解析即可。因为JUnit的测试结果只有xml文件，故插件无需像Maven单测插件那样设置报告查看入口文件。
质量门禁：此处可设置当测试通过率低于某数值时，测试任务会被认为失败。



插件执行结果样例如下：JUnit插件暂未提供html页面报告查看功能，后续会优化提供。

Jacoco覆盖率插件

Jacoco覆盖率插件提供了一个开箱即用，可以方便的执行测试并统计测试覆盖率的方法。
插件基本配置：如下图所示，通过配置基本maven环境与版本、命令与测试报告目录即可。
质量门禁：此处可设置当测试通过率低于某数值时，测试任务会被认为失败。Jacoco内置了多种覆盖率统计功能，故可以详细设置多个质量门禁。

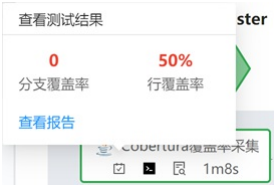


插件执行结果样例如下：可以查看覆盖率数据及点击查看报告

Cobertura覆盖率插件

Cobertura覆盖率插件提供了一个开箱即用，可以方便的执行测试并统计测试覆盖率的方法。
插件基本配置：如下图所示，通过配置基本maven环境与版本、命令与测试报告目录即可。
质量门禁：Cobertura插件提供了分支覆盖率和行覆盖率两种质量门禁。当覆盖率低于门禁设置时，会导致任务失败。

插件执行结果样例如下：可以查看覆盖率数据及点击查看报告



构建集群

集群导入

点击上图中右上角的导入按钮，可以导入K8S集群。导入的方法在界面上有详细说明，其中API Server地址需要使用基础地址。
服务令牌获取方法及步骤：
1、创建一个名为 ezone-admin-service-account.yaml的文件，内容如下：

```
apiVersion: v1
kind: ServiceAccount
metadata:
  name: ezone-admin
  namespace: kube-system
---
apiVersion: rbac.authorization.k8s.io/v1beta1
kind: ClusterRoleBinding
metadata:
  namespace: kube-system
  name: ezone-admin
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: ClusterRole
  name: cluster-admin
subjects:
  - kind: ServiceAccount
    name: ezone-admin
namespace: kube-system
```

2、在需要导入的集群上运行如下命令，创建流水线进行构建任务调度需要的账号ezone-admin

```
kubectl apply -f ezone-admin-service-account.yaml
```

如果你当前操作K8S的账号没有创建集群级别账号的权限，你可以启用K8S Basic Authentication认证方式后，使用如下命令以admin身份执行

```
kubectl apply -f ezone-admin-service-account.yaml --username=admin --password=<password>
```

如果成功执行，会得到如下输出

```
serviceaccount "ezone-admin" created
clusterrolebinding "ezone-admin" created
```

3、运行如下命令获取令牌

```
kubectl -n kube-system describe secret $(kubectl -n kube-system get secret | grep ezone-admin | awk '{print $1}')
```

4、复制令牌

```
Name:         ezone-admin-token-b5zv4
Namespace:    kube-system
Labels:       <none>
Annotations:  kubernetes.io/service-account.name=ezone-admin
              kubernetes.io/service-account.uid=bcfe66ac-39be-11e8-97e8-026dce96b6e8

Type:  kubernetes.io/service-account-token

Data
====
ca.crt:     1025 bytes
namespace:  11 bytes
token:      <authentication_token>
```

集权权限

在导入集群时，可对集群设置权限。
企业内公用：企业所有成员均可使用此构建资源
非企业公用：只有添加到该构建集群成员的人员有使用权限 集群管理员可对集群进行设置

集群使用

构建集群导入之后，即可在例如docker镜像执行器的设置中选择需要使用的集群。如果在构建任务执行过程中，用户没有该集群的使用权限，则会执行报错。

集群编辑

只有(集群)管理员才可以对该构建集群进行编辑。编辑时仅可以修改集群名称、权限、备注等信息。如需对集群进行编辑，切换至“我是管理员”标签页，对有管理权限的构建集群进行编辑。

产品介绍

制品库是企业效能平台产品系列中制品和镜像库产品，用以向企业提供一站式的制品管理功能，支持通用文件、Docker、Maven、Helm、NPM 、Pypi、Composer等常见制品库类型，同时可以与本地各构建工具和云上的持续集成，持续部署无缝结合。为研发团队提供优质高效的构建物管理服务，把控构建物质量。

相关术语

半成品库

用以存储半成品的制品库。所谓半成品即是未经过测试确认，未发布正式版本的制品。
举例：流水线的编译类插件会按照构建版本号（四位数字版本）自动将生成的制品上传到对应类型的半成品库。半成品库中的制品可以设置定时删除策略。
设置半成品库和成品库的目的：是为了更清晰的区分制品，防止混杂在一起导致的操作错误。其次是企业对于半成品和成品的存储策略也不同，研发过程中产生的大量半成品通常会被自动清理，长期保存的价值很小。

成品库

用以存储成品的制品库。所谓成品即经过测试确认，发布正式版本的制品。
举例：ezPipeline的发布版本插件会自动将该次构建对应的四位版本半成品按照生成的发布版本号(三位数字版本)发布到成品库中。
注意：ezPackage系统中，半成品库和成品库是成对出现的。

镜像库

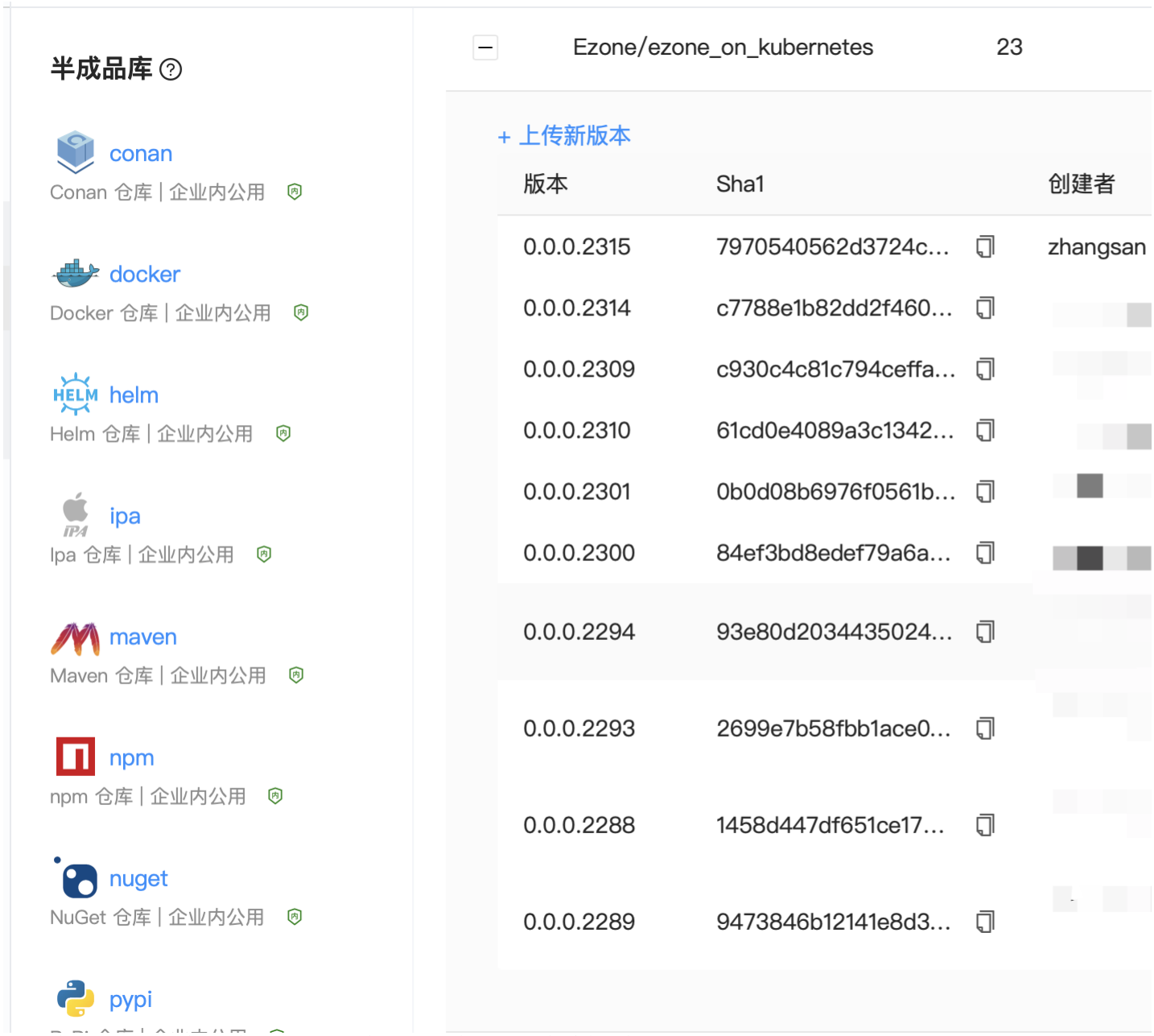
在实际企业环境中，通常会面临如下问题：
1.构建系统大多没有外网权限
2.即使构建系统有外网权限，从外网下载制品很慢
镜像库就是为了解决上述两个问题而提供的功能。为企业建立一个外网制品源的代理，缓存在镜像库中，这样只需要镜像库有外网权限就可以为构建系统提供高速稳定的服务。

包和版本

制品库的制品组织按照 库 > 包 > 版本的层级结构组织。一个制品库下可以有很多制品包、每个制品包下可以有很多版本。

半成品库

制品库内置了多种常见制品库类型，且每种制品类型中，半成品库和成品库是成对出现的。点击包名，即可浏览该包下的所有版本，以及版本的详情。



对于制品库的设置，也是半成品库和成品库统一设置的。半成品库可以设置自动清理策略。成品库不会被自动清理。

每种不同类型的制品库，点击“使用方法”标签页，即可查看关于该制品类型的详细说明文档。
如果有自动化任务需要访问制品库，可以使用个人token代替个人密码。具体参考“平台基础功能 > 个人中心 > 访问令牌”。

内置库与新建

ezOne为每个企业内置了常用的各种类型制品库，这些制品库的权限类型为企业内公用。新建功能并未默认提供，对于有特殊需求的用户会按需开放创建多个制品库的能力。自行创建的制品库类型可以分为“保密”及“企业内公用”两种。

权限与角色

对于“保密”类型制品库，用户分为：管理员和成员。管理员具有该制品库一切权限，成员具有制品库的读写权限。企业其他成员无法访问该制品库。
对于“企业内公开”类型制品库： 只需要设置管理员，管理员具有该制品库一切权限。企业其他用户都可以读写此制品库内制品。
对于成品库：则只有管理员可删除制品。成员不具有删除权限。

成品库

成品库和半成品库的使用方法相同。

镜像库

镜像源设置

每种不同的制品镜像源可以设置多个镜像源。只有企业管理员可以对企业镜像库进行设置。此操作对普通用户不可见。

镜像库工作原理

当镜像库收到制品包下载请求时，如果发现制品包已经存在，则直接响应客户请求，将制品包返回。如制品包不存在，则会从配置的镜像源中，由上而下逐个去寻找并下载制品缓存起来，直到找到为止。如果最终找不到，则会返回用户，该制品不存在。

镜像清理

ezPackage内置了对镜像库的智能清理策略，可以设置长期无人使用的镜像超时自动清理。

镜像库权限

因为镜像库的作用只是代理外网公共镜像，故镜像库未开放写入权限，只开放了读权限。企业内所有用户均可从镜像库下载制品。

自动代理及读写分离

对于半成品库和成品库，制品库内置了一项专利技术：自动代理及读写分离。具体举例如下（以docker半成品库举例）：

1. 如果通过docker push命令向该半成品库推送镜像，ezPackage识别该请求为“写”流量，将镜像推送到docker半成品库。
2. 如果通过docker pull命令向该半成品库拉取镜像，ezPackage识别该请求为“读”，会自动按照 docker半成品库 > 企业docker镜像库 > 镜像库配置的外网镜像源的顺序自动下载并按需缓存到镜像库中。
   这样用户在使用和配置的时候，就无需配置和关注多个制品库。

产品介绍

主机部署运维是企业效能平台产品系列中的主机部署运维产品，为用户提供构建制品快速便捷部署到线上集群的能力。支持虚机、物理机等环境；支持应用部署及自动化运维脚本部署。可以将用户的应用或者需要自动执行的脚本部署到主机。结合流水线的自动化部署插件，更可实现自动化运维。

相关术语

制品： 主机部署运维只能部署文件包类型的制品到主机。
主机： 一台物理机或者虚拟机。
主机组： 通常由用途相关的一组机器组成一个主机组。例如由线上的一组机器可以组成主机组“线上环境”。
部署模板： 为提升部署效率，将部署任务的各项配置提取成一个模板。
部署任务： 部署模板的一次部署执行，将某个版本的文件包制品按照部署模板的设置，部署到特定的主机环境中。
代理机： 由于企业环境中，不同用途的机器会划分为不同网段，例如线上环境和线下环境网络是隔离的，此处就需要通过一台具有双网卡的代理机实现网络的联通。

角色与权限

管理员： 具有主机组一切权限。
成员： 默认只具有创建部署模板的权限，如果管理员通过主机组设置，允许成员创建或者修改部署模板，则普通成员亦可创建或者修改部署模板。

主机组管理

点击“主机部署”主菜单，可以进入主机组管理界面，主机组管理界面提供了“全部”、“我是管理员”、“我的收藏”三个标签页。主机部署运维会自动记录选择的标签页，下次进入主机部署运维时会自动切显示该标签页。
主机组管理功能提供了新建、编辑、删除、收藏等功能。

新建

点击右上角新建按钮，可以新建主机组。新建界面提供主机组名称、描述、管理员和成员设置等功能。

编辑

只有在“我是管理员”标签页才提供主机组的编辑功能。

删除

此操作属于危险操作，只有该主机组下的主机和部署模板清空后才可删除主机组。

部署任务

部署任务

部署任务菜单提供了部署任务列表及新建部署任务的功能。

部署任务状态

部署任务分为：新建、等待审批、进行中、失败、部署成功、部署失败。
由于部署任务新建后即开始执行或者进入等待审批状态，故新建状态一般不可见。
如果部署任务需要审批，其新建后会自动等待审批状态。

重新部署

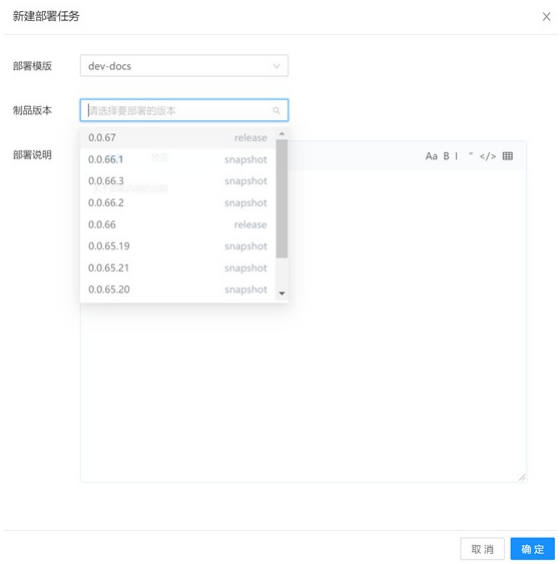
对于已经失败的部署任务，才可以进行重新部署。但是重新部署不会覆盖本部署记录，只是会重新发起一条部署任务而已。

回滚

对于已经部署成功的部署任务，才可以进行回滚操作。 主机部署运维的回滚和传统意义的回滚不太一样，是指回滚“至”某个版本。其优点是传统意义的回滚等同于取消最新一次的部署，而主机部署运维的“回滚至”则支持回滚至任何一个已经部署成功的版本。

新建

新建部署任务时，只需要选择部署模板、制品版本、填写部署说明即可提交一次部署任务。



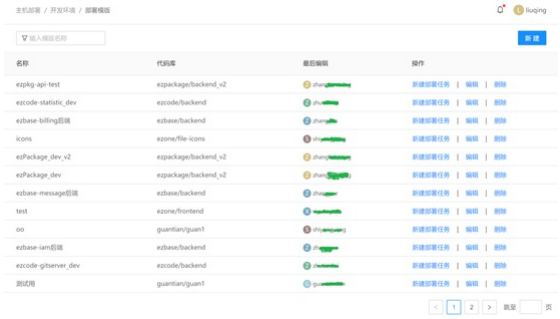
其中制品版本是从制品库的文件类制品中获取。四位版本的是半成品库中的文件制品，三位版本的是成品库中的文件制品。

流水线自动部署

主机部署运维可以通过流水线插件实现自动部署，详情请参考流水线“主机部署插件”章节

部署模板

部署模板菜单提供了对部署模板进行管理的功能，可以显示部署模板列表、对部署模板进行编辑、删除和新建操作，并且可以直接基于某部署模板发起部署任务。默认情况下只有主机组管理员才可以对部署模板进行操作。

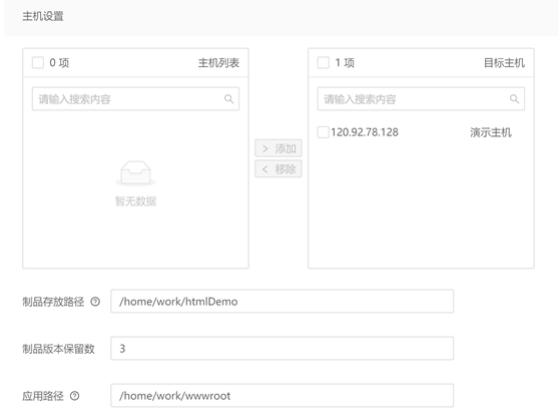


基本设置

部署模板的基本设置设置了模板名称、代码库（会根据代码库名到文件类制品库中自动下载制品包）、以及是否需要审批等设置。

主机设置

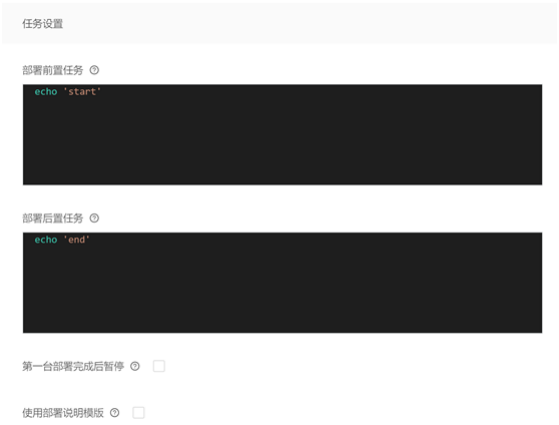
主机设置用以设置将制品部署到主机组的哪些目标主机上；
制品存放路径：表示部署器将制品存放在主机组的位置，用以快速回滚等场景。
应用路径：web服务器设定的应用路径
特别注意： 应用路径不能和制品存放路径相同、且该目录必须由部署器自动创建或者是空目录。否则会导致部署失败！ 原理是部署器会在下载制品后，自动解压，生成一个文件夹，并且创建一个该文件夹到应用路径的软链。如果应用路径已经存在，且为非空，则部署器会报错。如果该应用路径已经存在，但是是空路径，部署器会自动删除该路径并且创建软链。



任务设置

部署前置任务主要是停服脚本
部署后置任务通常是服务启动脚本
部署任务的真正执行过程：

1. 部署服务从制品库下载制品 2.服务服务将制品上传至目标主机 3.执行部署前置任务 4切换版本软链 5.执行后置任务
   这样真正前置和后置任务之间，服务中断的时间很短。只是一个软链切换过程。
   如果是集群部署，前端有负载均衡，就可以做到无感上线。



主机列表

主机列表功能提供了对主机组内的主机管理的功能，主机组管理员可以通过此功能对主机进行管理。主机组普通用户可以查看主机信息。

新建主机

添加主机时，填入主机ip、或者hostname、域名；端口、用户名和密码即可。部署器会通过ssh对主机进行连接和进行部署。无需在主机上安装agent，使用更便捷。

代理机设置

由于企业环境中，不同用途的机器会划分为不同网段，例如线上环境和线下环境网络是隔离的，此处就需要通过一台具有双网卡的代理机实现网络的联通。
打开代理机设置开关（只有主机组管理员有权限操作、主机组普通用户只能查看），即可对代理机进行设置。
注意： 由于代理机是具有双网卡双IP的，此处的IP一定是和部署系统在同一网段的IP地址。

产品介绍

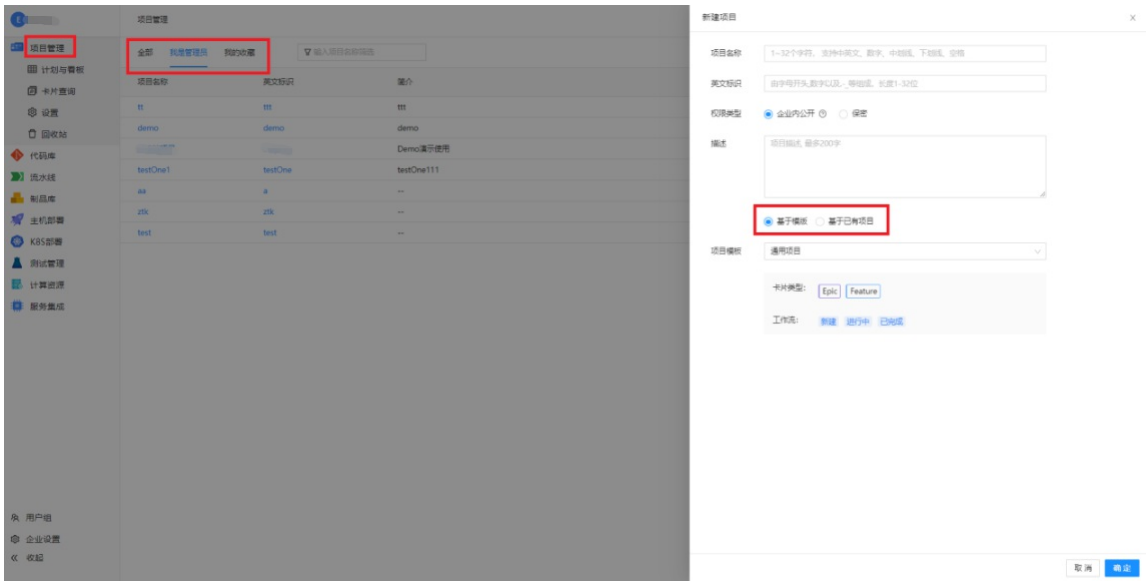
ezProject是金山云企业效能平台（KDF）产品系列中的项目管理产品，支持需求和缺陷管理、支持计划和看板、卡片查询、回收站、项目设置、项目报表等功能，和上下游工具结合，可实现整个研发过程可视化，软件全生命周期数据链贯通可追溯。

相关术语

项目： 也称做项目空间，用以承载功能的一个独立单元，一个企业内可以创建多个项目。
卡片： 指项目中任何一个工作项，业界也习惯称之为一个卡片，因为之前经常把工作项打印成卡片贴墙上。
需求池： 也称做Backlog，用于存放未列入计划的各种待办事项，不止能放需求，也可以放缺陷。
计划： 一个有开始和结束时间范围的时间段，在这个计划里可以放入待办工作项，就像一个工作项的盒子，所以也可以叫做计划盒。

项目列表

如下图所示，点击左侧菜单“项目管理”，可以以不同标签页查看企业项目列表。



创建项目步骤

1. 点击右上角“新建”按钮，可以进入项目创建界面。
2. 在创建项目时，可以选择项目是保密还是公开，但是此时不能进行详细的权限设置。
3. 项目创建完成后，可以进入项目设置菜单进行详细设置，同时还可以选择基于模板或者其他已经存在的项目快速创建新项目。

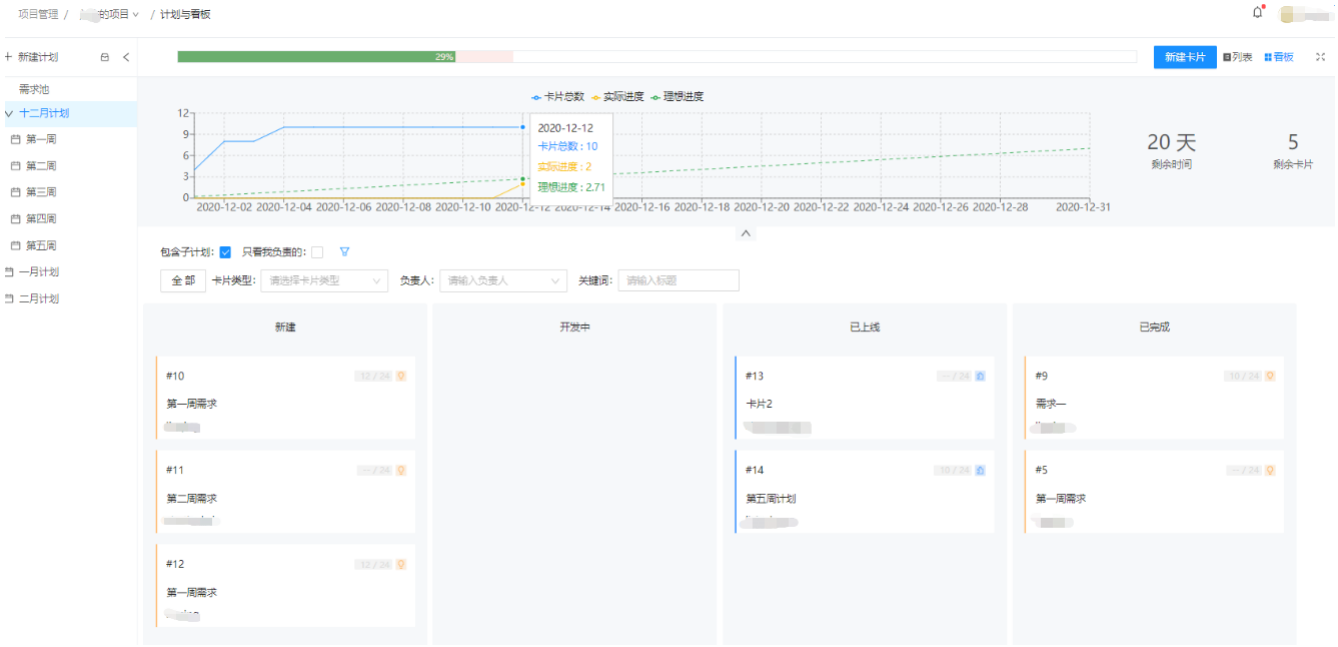
计划与看板

列表视图

列表视图是针对项目计划会场景进行了诸多特殊优化的功能。
例如支持：
1. 上下拖动排优先级
2. 左右拖动到计划盒排计划
3. 承诺交付线便于鼓励大家制定更高的目标时，又守住交付底线
4. 顶部人员工作量分工清晰展示
5. 支持对计划进行归档

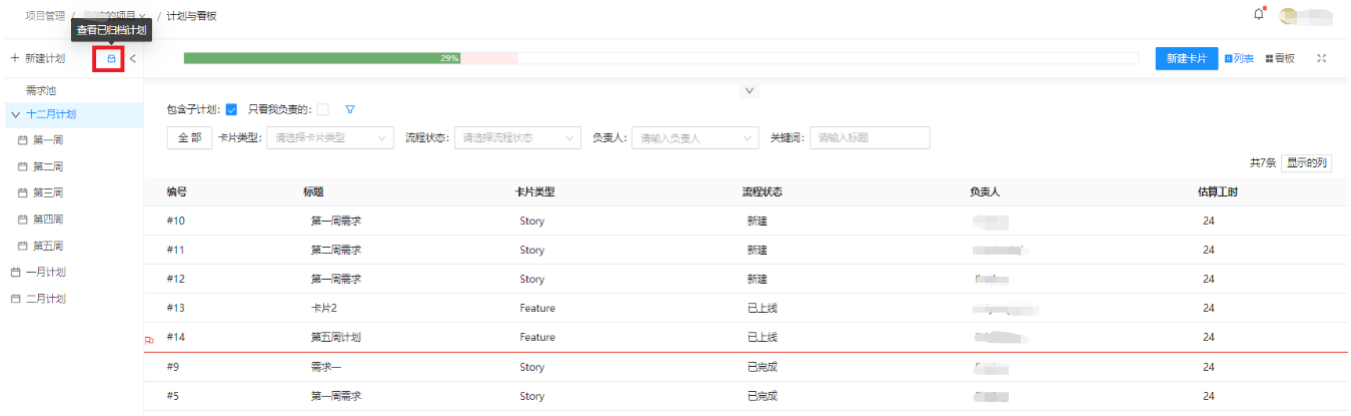
看板视图

看板视图适合在项目进行过程中可视化查看项目状态，并且支持全屏模式、支持站会场景、支持拖拽查看卡片状态、同样也支持对计划进行操作。



查看归档计划

对于已经归档的计划，可以点击图示菜单查看并且支持解除归档。



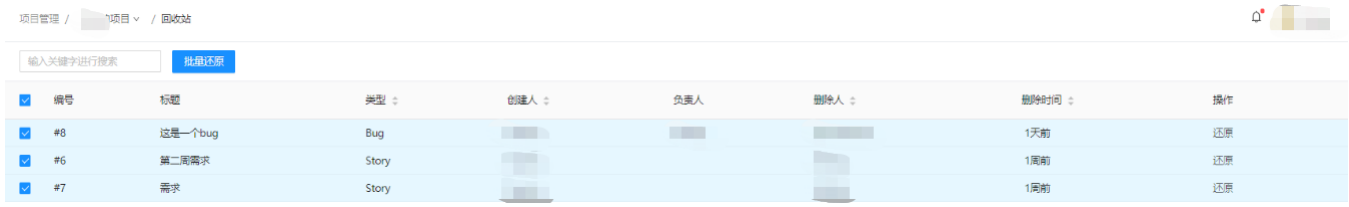
卡片查询

卡片查询步骤

卡片查询内置三个基本查询、点击“全部” 设置筛选条件。
当修改了查询条件后，可以点击“另存为”按钮存储为一个自定义查询。

卡片回收站

对于已经删除的卡片，可以在卡片回收站中进行还原。



项目设置

基本设置

在基本设置中可以对项目的英文标识，权限类型等基本信息进行设置。
权限类型的说明参见Tooltip中的文字说明。
对于企业内公开的项目：企业所有成员均有项目的读权限，并且可以在项目的需求池新建卡片，用于公共项目收集反馈。

成员设置

管理员：具备项目所有权限。
成员： 具有除了改变项目设置外的所有权限。
访客：访客具有项目的读、项目卡片的评论及在项目需求池中创建卡片用以反馈建议的权限。

运维操作

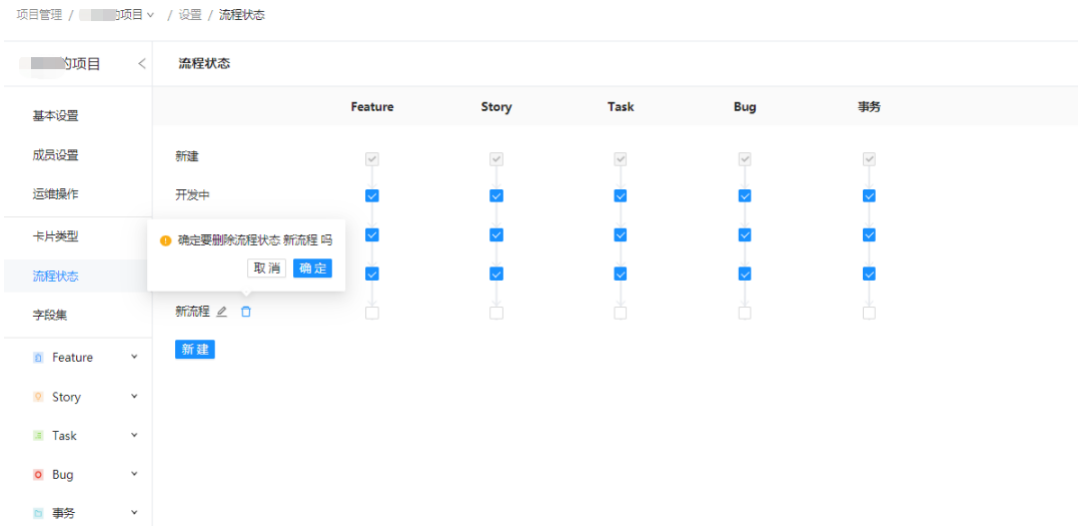
可以在运维操作设置界面中删除项目。

卡片类型

此处定义的是整个项目的卡片类型开关。ezProject默认支持6种卡片类型，包括通用的事务类型卡片用于非研发任务、非研发项目。

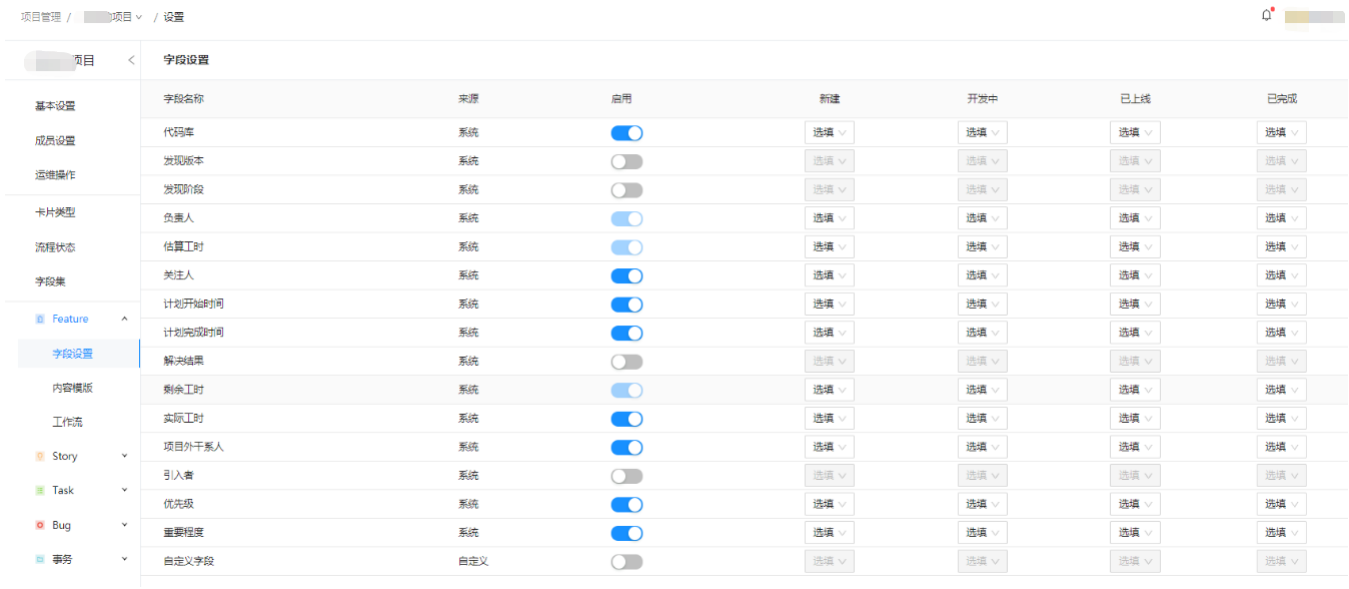
流程状态

此处设置整个项目的流程状态全集，以及不同类型的卡片启用的状态集合。
可以对状态进行新建和删除操作。
注意事项： 删除状态时，会提示之前该状态的卡片这状态批量转换为其他状态，否则会导致数据不一致。



字段集

此处设置的字段集是项目中所有卡片类型使用的字段的集合，每种卡片类型只需要设置自己需要的字段开关即可。在此处统一配置避免不同卡片重复添加相同或者近似的字段。
系统内置的字段不可编辑和删除、项目特殊需求可以通过新建自定义字段。

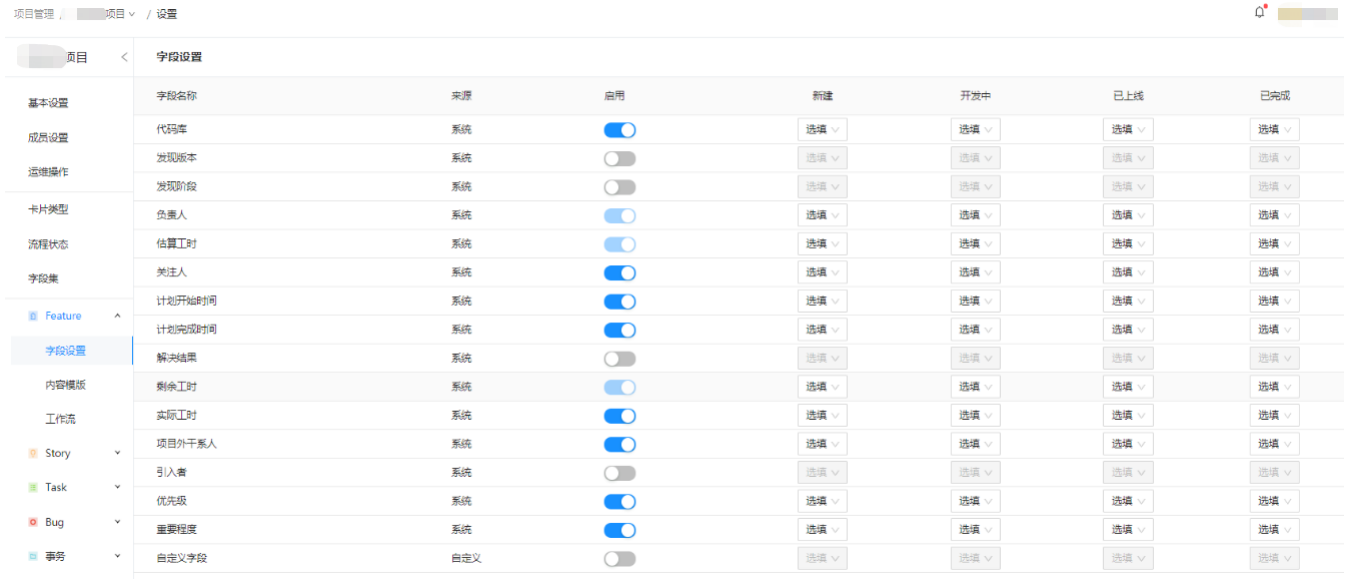


卡片设置

对于项目中启用的卡片类型，可以分别设置他们的字段、模板和工作流。

字段设置

字段设置中，可以设置对某个特定的卡片类型，启用项目字段集中的哪些字段，以及其他更多设置。

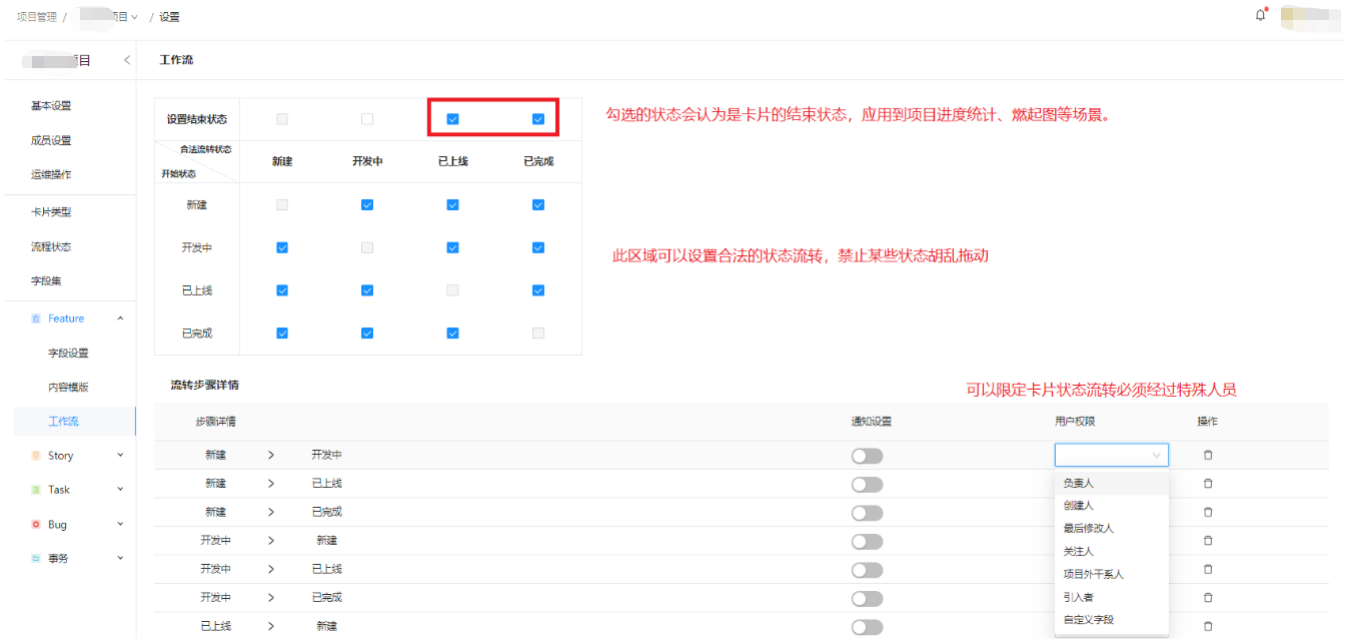


内容模板

作为团队研发规范，可以为每种卡片类型设置规定的模板。

工作流

卡片工作流可以设置哪些状态为结束状态，结束状态的设置会影响项目进度的统计，并且可以通过设置状态的流转规则来禁止某些状态间乱拖动，其次是可以设置某些状态间的流转是否进行权限约束。



产品介绍

ezTest是金山云企业效能平台（KDF）产品系列中的测试管理平台，为用户提供测试空间管理、测试用例管理、测试套件管理及测试执行等功能支持。未来还会逐步集成接口测试及压力测试功能，打造综合测试管理平台。

产品术语

测试空间： 在企业内，不同的部门、不同的团队、不同的项目不可能把测试相关的内容全部都堆放在一起，通过不同的测试空间进行隔离。
测试用例： 对应术语TestCase，一个测试的基本单元，可以包含测试的准备，若干测试步骤、测试后的清理工作。
测试套件： 对应术语TestSuite，由若干个测试用例可以组成一个测试套件。
测试执行： 可以选择测试套件进行测试执行，并记录测试结果，结果支持以报表的形式查看。

测试空间管理

点击图中“测试管理”主菜单，即可进入测试空间管理界面。
在测试空间中，分别以三个标签页分类查看测试空间。
点击“新建”按钮可以新建测试空间。

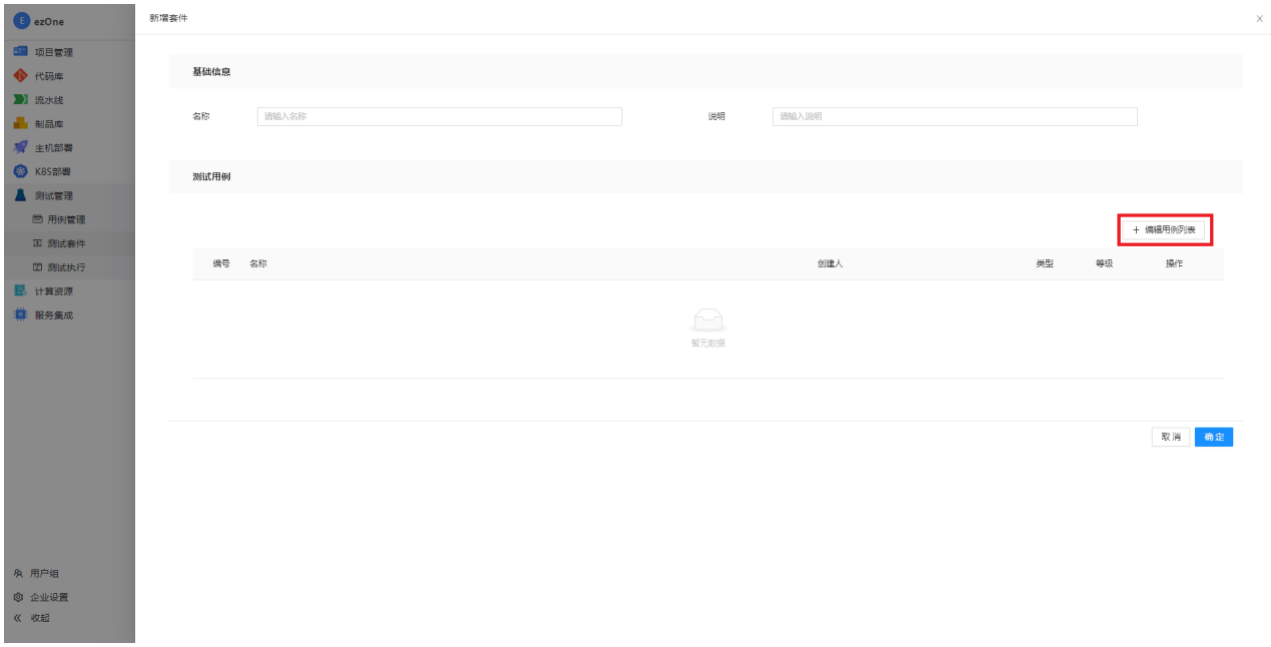
测试空间的分类：
企业内公用： 企业成员均可使用此空间，具有除空间设置权限外的所有使用权限。
保密： 只有成员可使用。

用例管理

进入“用例管理”菜单，可以为测试用例建立不同文件夹，并在文件夹下建立用例，方便归类管理。

套件管理

可以将一些测试用例组成测试套件，针对不同的功能模块进行测试。
点击“套件管理”菜单，即可对测试套件进行管理，在测试套件新建页面，可以点击“编辑用例列表”，方便进行用例筛选。



测试执行

进入“测试执行”菜单，可以对测试结果进行记录，查看测试报表。

创建测试执行时，可以选择是否基于测试用例。在创建测试执行之后，可以自由选择测试用例。

产品介绍

ezK8S是企业效能平台产品系列中的容器部署产品，提供把镜像部署至Kubernetes集群的能力。主要功能包括Kubernetes集群的导入和管理、环境(命名空间)管理、部署管理及PV管理等功能。

相关术语

集群：指导入的Kubernetes集群
环境：指Kubernetes集群的命名空间
PV：指Kubernetes集群的持久化卷，K8S集群层级概念 PVC：指Kubernetes集群的持久化卷声明，命名空间层级概念
部署模板：替代yaml部署配置文件的图形可视化化模板，方便后续部署时提升部署效率
部署任务：部署模板的一次实际执行称为一个部署任务
部署实例：通过部署任务创建的实际运行的实例，实例名称根据部署模板和日期自动生成，可以在部署任务创建时修改

角色与权限

K8S集群的角色与权限

K8S集群层级只有管理员一个角色，管理员可以对集群进行所有操作。

环境的角色与权限

环境对应于K8S集群下的一个命名空间，在环境层级有两类用户：管理员和普通用户。

- 环境管理员：可以对环境进行管理操作
- 普通用户： 具有除了对环境进行编辑、资源配额设置和资源限制设置之外的其他操作权限

集群管理

点击左侧“K8S部署”菜单，即可进入集群管理界面。

导入

在使用ezK8S之前，首先要将集群导入。对于支持CNCF标准的Kubernetes集群均可导入。导入方式和ezPipeline导入Kubernetes构建集群时方法相同。点击上图中右上角的导入按钮，可以导入K8S集群。导入的方法在界面上有详细说明，其中API Server地址需要使用基础地址。

服务令牌获取方法及步骤如下。

1. 创建一个名为 ezone-admin-service-account.yaml的文件，内容如下:

```
apiVersion: v1
kind: ServiceAccount
metadata:
name: ezone-admin
namespace: kube-system
---
apiVersion: rbac.authorization.k8s.io/v1beta1
kind: ClusterRoleBinding
metadata:
namespace: kube-system
name: ezone-admin
roleRef:
apiGroup: rbac.authorization.k8s.io
kind: ClusterRole
name: cluster-admin
subjects:
- kind: ServiceAccount
name: ezone-admin
namespace: kube-system
```

2. 在需要导入的集群上运行如下命令，创建流水线进行构建任务调度需要的账号ezone-admin kubectl apply -f ezone-admin-service-account.yaml 如果当前操作K8S的账号没有创建集群级别账号的权限，可以启用K8S Basic Authentication认证方式后，使用如下命令以admin身份执行

```
kubectl apply -f ezone-admin-service-account.yaml --username=admin --password=<password>
```

成功执行后，会得到如下输出

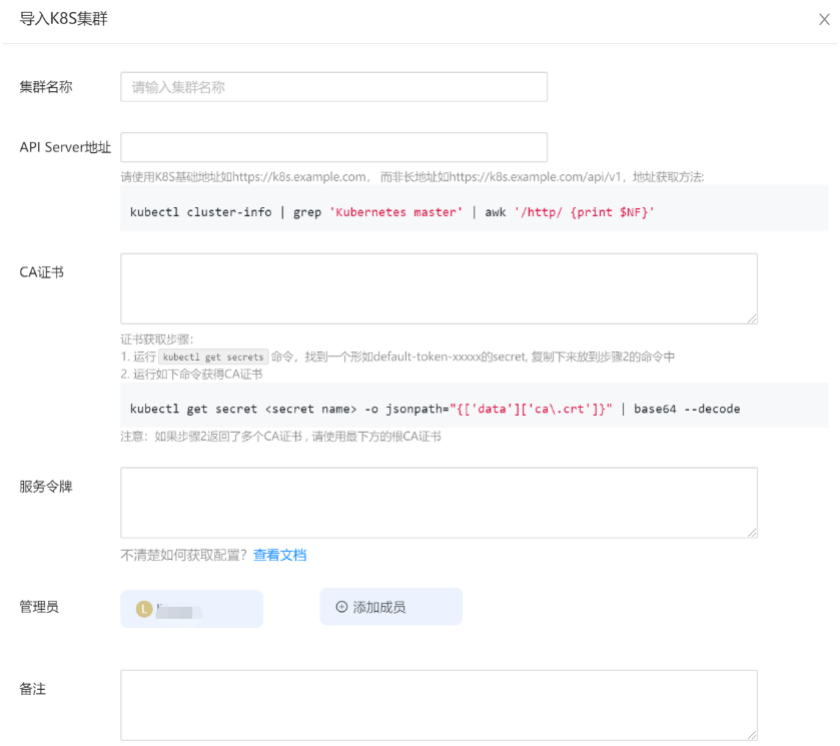
```
serviceaccount "ezone-admin" created
clusterrolebinding "ezone-admin" created
```

3. 运行如下命令获取令牌

```
kubectl -n kube-system describe secret $(kubectl -n kube-system get secret | grep ezone-admin | awk '{print $1}')
```

4. 复制令牌

```
Name:         ezone-admin-token-b5zv4
Namespace:    kube-system
Labels:       <none>
Annotations:  kubernetes.io/service-account.name=ezone-admin
              kubernetes.io/service-account.uid=bcfe66ac-39be-11e8-97e8-026dce96b6e8
Type:  kubernetes.io/service-account-token
Data
====
ca.crt:     1025 bytes
namespace:  11 bytes
token:      <authentication_token>
```



编辑

在集群管理界面，切换到“我是管理员”标签，即可对有权限的集群进行编辑。编辑模式仅支持对集群名称和备注编辑。

环境管理

选择某个Kubernetes集群后，会在左侧出现“环境管理”、“部署管理”、“PV管理”二级菜单。点击“环境管理”进入环境管理界面。如果想切换到别的Kubernetes集群下进行环境管理，可以在顶部面包屑切换。

新建

只有集群管理员才有在集群下创建环境的权限。新建环境时，如果该环境设置为“企业内公用”，则企业成员均有该环境的使用权限。否则只有被添加进来的用户有环境使用权限。

编辑

只有环境的管理员才有集群编辑权限，编辑环境时仅能编辑备注和人员。

删除

只有环境管理员才能删除环境，环境被删除时，需要先行将Configmap，密文，服务配置，部署实例，域名，证书，pvc清空后或删除。

审计

作为企业级应用，ezK8S提供了环境操作审计功能，以查看操作历史。

PV管理

点击左侧PV管理二级目录，即可进入到PV管理界面。提供PV的新建、刷新、删除及查看功能。因为PV是在远程Kubernetes集群创建，故提供刷新功能以同步状态。

访问模式

Kubernetes中PV的访问模式分为如下三类：

- ReadWriteOnce – PV 能以 read-write 模式 mount 到单个节点。
- ReadOnlyMany – PV 能以 read-only 模式 mount 到多个节点。
- ReadWriteMany – PV 能以 read-write 模式 mount 到多个节点

新建NFS类型PV

如下图，创建NFS类型PV时需要注意服务器IP需要是Kubernetes集群可访问的IP.

新建本地类型PV 如下图所示，路径是在Kubernetes集群节点上的路径。对于本地类型PV，如果设置访问模式为ReadOnlyMany或者ReadWriteMany是没有意义的。

删除

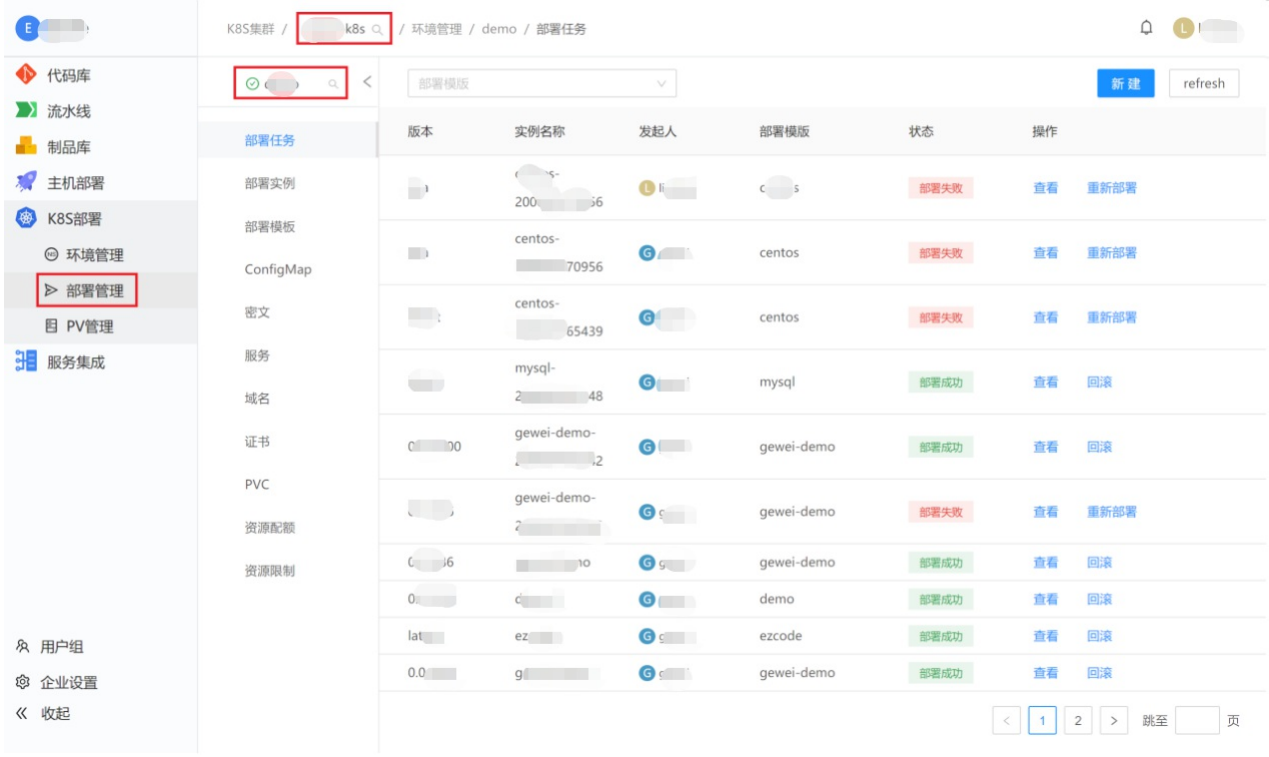
只有集群管理员才可以对集群下的PV进行管理，PV属于集群，不属于环境，所以只有集群管理员才可以删除PV。

查看

点击PV管理界面的查看按钮即可查看PV详情。

部署管理

点击左侧“部署管理”菜单，即可进入部署管理界面。部署管理界面功能较多，所以引入了三级菜单。可以在顶部面包屑切换不同的Kubernetes集群，在三级菜单顶部切换环境。



部署模板

进行容器化部署，可以通过部署模板简化操作。将需要的配置项通过模板进行固化后再后续的部署任务中，可以按照模板配置进行自动部署。
点击左侧“部署模板”菜单，可以查看现有部署模板列表，并对现有部署模板进行编辑，新建等操作。
部署模板主要包括基本设置、权限设置、容器设置、健康检查和部署模板几个部分。

新建部署模板

副本数量：是该模板创建部署任务时默认的副本数量，在发起部署任务的时候，可以修改。
服务端口：是容器启动时的默认服务端口。
镜像源：可以选择ezOne内置的docker制品库、docker镜像库或者通过“服务集成”配置的第三方docker镜像库。

权限设置

开启部署任务审批后，所有部署任务需要经过环境管理员或者集群管理员审批后方可执行。

容器设置

容器设置部分是可选设置，点击右上角的“收起/展开”按钮可以展开或者收起编辑区域。

健康检查

健康检查是高级设置中的可选项，健康检查内包括容器存活检查（对应Kubernetes的Liveness配置）和容器就绪检查（对应Kubernetes的Readiness配置）两个检查开关。每种检查又包括TCP端口检查、Http请求检查和执行命令检查三种检查方法。

部署任务

创建好部署模板之后，可以基于部署模板创建部署任务。新建部署任务时需要选择部署模板，选择部署模板之后，会根据部署模板和日期自动生成实例名称（支持手动修改）及部署模板中配置的默认副本数量及部署说明文案。如果实例名称已经存在，则此部署任务会更新该部署实例。如果该实例名称不存在，则部署任务会创建该实例。

任务状态

部署任务包括如下状态：

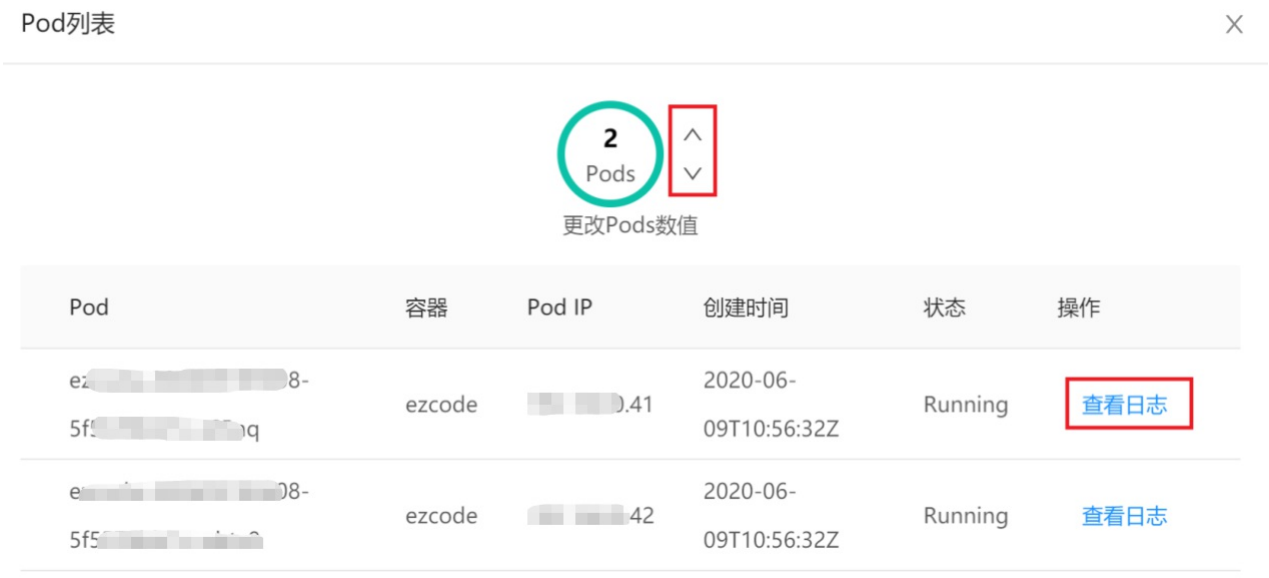
- 部署成功： 此时可进行的操作包括查看及回滚。点击回滚按钮，即可将版本回滚至某特定版本。
- 部署失败：此时可进行的操作包括查看、重新部署。重新部署会按照之前的部署任务配置创建一条新的部署记录。
- 待审批：对于需要审批的部署任务，在部署任务创建之后，会进入待审批状态。此时可进行的操作包括查看及审批（只有环境或集群管理员才有审批权限）。
- 部署中：此时可进行查看操作。

部署实例

部署实例是部署任务运行成功后生成的运行中的实例。在创建部署任务时，会根据部署模板和时间自动生成实例名称（支持手动修改）。
同一个部署模板可以被部署出多个不同名的部署实例。对于同名部署实例，部署任务会用更新实例的方式进行部署。例如下图中的ezcode和ezcode-200609185608两个部署实例都是由同一个部署模板ezCode及同一个镜像版本0.0.0.2创建。

Pods

点击Pods按钮，右浮界面可以查看Pods的数量及详情。Pods的数量由部署任务中设置的“副本数量”决定，并可在此处支持手动调整Pod数量实现服务弹性伸缩。每个Pod的名称由实例名称加上一串随机



数字组成。
点击“查看日志”按钮可以查看该pod的日志。

Events

Events常用于排查Kubernetes集群错误。此数据非持久化保存，请注意。

更新实例

点击更新实例按钮进行实例更新时，是通过创建一个新的部署任务实现的。在新的部署任务中，部署模板和实例名称不可修改（否则就是创建新实例了），可以设置不同的副本数量和版本。

删除

由于删除操作时在Kubernetes集群上进行操作，点击删除后，可点击右上角的Refresh按钮来更新实例状态。

ConfigMap

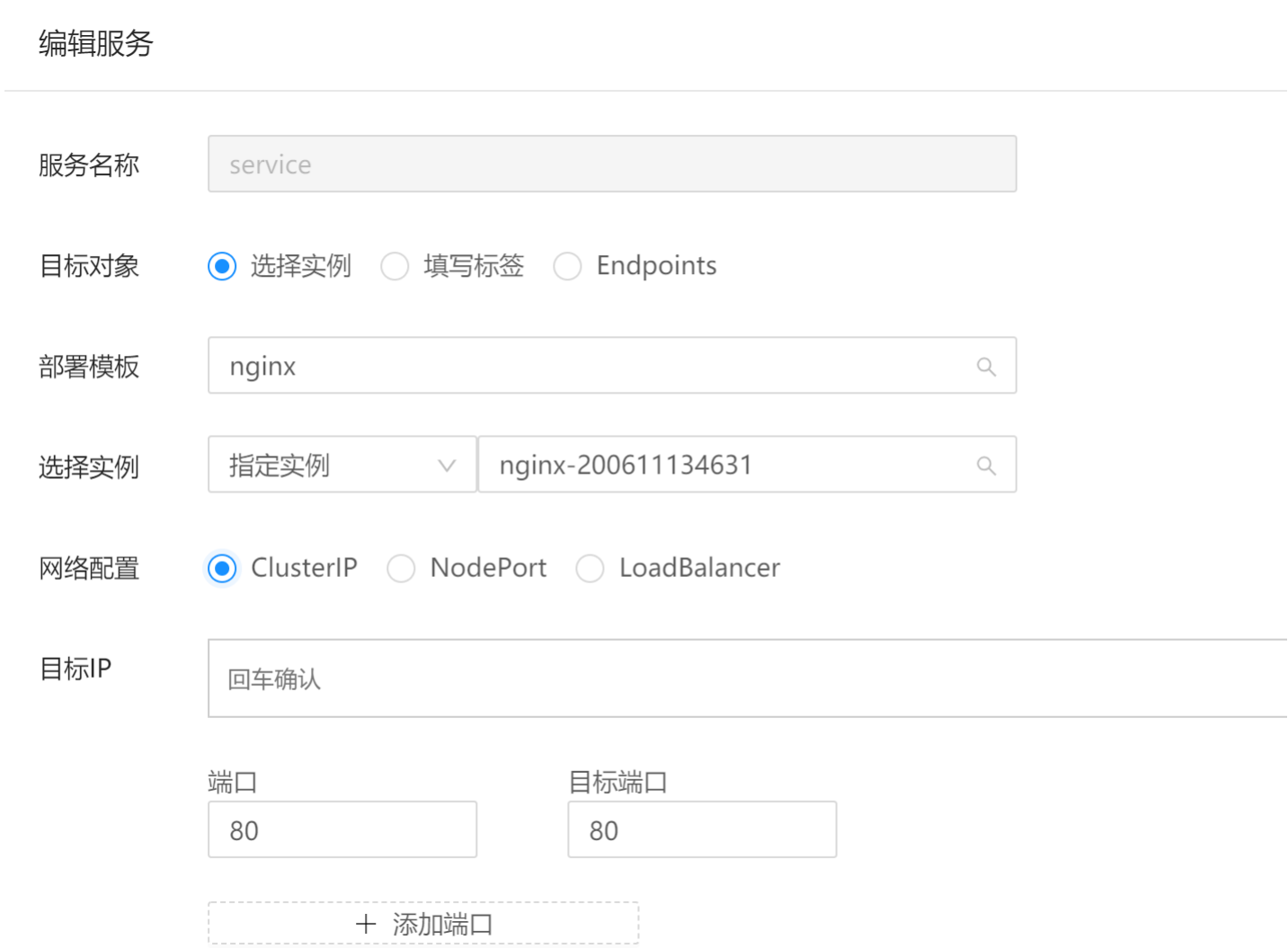
ConfigMap和密文相关操作主要是添加key和value对。

密文

密文对应Kubernetes集群的Secret概念，和Configmap的操作类似。

服务

服务对应Kubernetes集群的Service概念，pod通过Service向外部提供可访问的服务。
通过ezK8S的服务绑定功能，可以将Service和部署实例通过实例名称、标签或者Endpoint等方式绑定。



域名

域名的设置中需要注意：如果选择的是https的域名，则需要选择相应的https证书。也即只能选择域名匹配的证书。

证书

对于https的服务，需要对应的证书文件。ezK8S提供了服务的证书管理功能，并提供了两种证书上传方式。文件上传及文本复制方式。
上传证书时，会检测证书内容的合法性。对于日期是否过期等会做友情提示。

PVC

PVC对应Kubernetes集群的PersistentVolumeClaim概念，创建PVC之前，应该先创建PV，不同类型的PV，需要创建与之对应类型的PVC才能被使用。

自动化部署

除通过本章节介绍的部署任务进行手动部署外，还可以通过流水线插件插件进行自动化部署。详情参考流水线插件。