

Devops用户使用手册

Version 1.1.0

REV 1

P/N 100-100-100

Date: 2019/04/02

目录

[第一章. 平台总览 4](#_Toc24632323)

[1.1. 项目管理人员总览-驾驶舱 4](#_Toc24632324)

[1.1.1. 应用系统概述 4](#_Toc24632325)

[1.1.2. 人员概括 5](#_Toc24632326)

[1.1.3. 本月epic上线情况 5](#_Toc24632327)

[1.1.4. 项目总体任务完成度 6](#_Toc24632328)

[1.1.5. 产能分析 6](#_Toc24632329)

[1.1.6. 容器资源 7](#_Toc24632330)

[1.1.7. ⅠⅡ级缺陷趋势 7](#_Toc24632331)

[1.1.8. 各阶段耗时分析 8](#_Toc24632332)

[1.2. 工作台 8](#_Toc24632333)

[1.2.1. 我参与的应用系统 8](#_Toc24632334)

[1.2.2. 我的任务 9](#_Toc24632335)

[1.2.3. 我的动态 9](#_Toc24632336)

[1.2.4. 我的消息 10](#_Toc24632337)

[1.3. 首页 10](#_Toc24632338)

[1.3.1. 应用系统信息 10](#_Toc24632339)

[1.3.2. 概览 11](#_Toc24632340)

[1.3.3. 研发动态 11](#_Toc24632341)

[1.3.4. Pipeline执行日志 12](#_Toc24632342)

[第二章. 服务管理 13](#_Toc24632343)

[2.1服务管理 13](#_Toc24632344)

[2.1.1服务的查询、新增、编辑和删除 13](#_Toc24632345)

[2.2pipeline 14](#_Toc24632346)

[2.2.1pipeline的查询、创建和定义 16](#_Toc24632347)

[2.2.2Pipeline的记录、克隆、执行与其它操作 23](#_Toc24632348)

[2.3服务构建 25](#_Toc24632349)

[2.4服务部署 28](#_Toc24632350)

[第三章. 代码仓库 30](#_Toc24632351)

[3.1代码仓库的查询、新增、编辑和删除 30](#_Toc24632352)

[第四章. 制品库 32](#_Toc24632353)

[4.1制品库的查询、新增、编辑和删除 32](#_Toc24632354)

[第五章. 环境管理 34](#_Toc24632355)

[5.1环境的新增、编辑和删除 34](#_Toc24632356)

[第六章. 版本管理 38](#_Toc24632357)

[6.1版本构建 38](#_Toc24632358)

[6.1.1版本构建的新增、编辑和删除 38](#_Toc24632359)

[6.2应用系统的处理逻辑 43](#_Toc24632360)

[6.2.1研发流程节点显示处理逻辑 43](#_Toc24632361)

[6.2.2研发、测试、运行一体化 44](#_Toc24632362)

[6.3版本部署 61](#_Toc24632363)

[6.2.1版本部署的新增、编辑和删除 61](#_Toc24632364)

[第七章. 运行管理 63](#_Toc24632365)

[7.1环境管理 63](#_Toc24632366)

[7.2服务管理 63](#_Toc24632367)

[7.3实例管理 63](#_Toc24632368)

[7.4负载管理 63](#_Toc24632369)

[第八章. 度量分析 64](#_Toc24632370)

[8.1进度分析 64](#_Toc24632371)

[8.2质量分析 65](#_Toc24632372)

[8.3效率分析 66](#_Toc24632373)

[第九章. 人员考核 68](#_Toc24632374)

[9.1工作分析 68](#_Toc24632375)

[9.2Story完成分析 68](#_Toc24632376)

[9.3SubTask完成分析 69](#_Toc24632377)

[9.4缺陷分析 69](#_Toc24632378)

[第十章. 项目设置 70](#_Toc24632379)

[10.1用户管理 70](#_Toc24632380)

[10.1.1用户信息的查询、增加、删除、和编辑 70](#_Toc24632381)

[10.2角色管理 74](#_Toc24632382)

[10.2.1角色的查询、增加、删除和编辑 74](#_Toc24632383)

[10.3工作流配置 76](#_Toc24632384)

[10.4任务模板 78](#_Toc24632385)

[10.4.1任务模板的查询、增加、编辑和删除 78](#_Toc24632386)

[第十一章. 操作引导 81](#_Toc24632387)

[11.1配置工作 81](#_Toc24632388)

[11.2创建服务 81](#_Toc24632389)

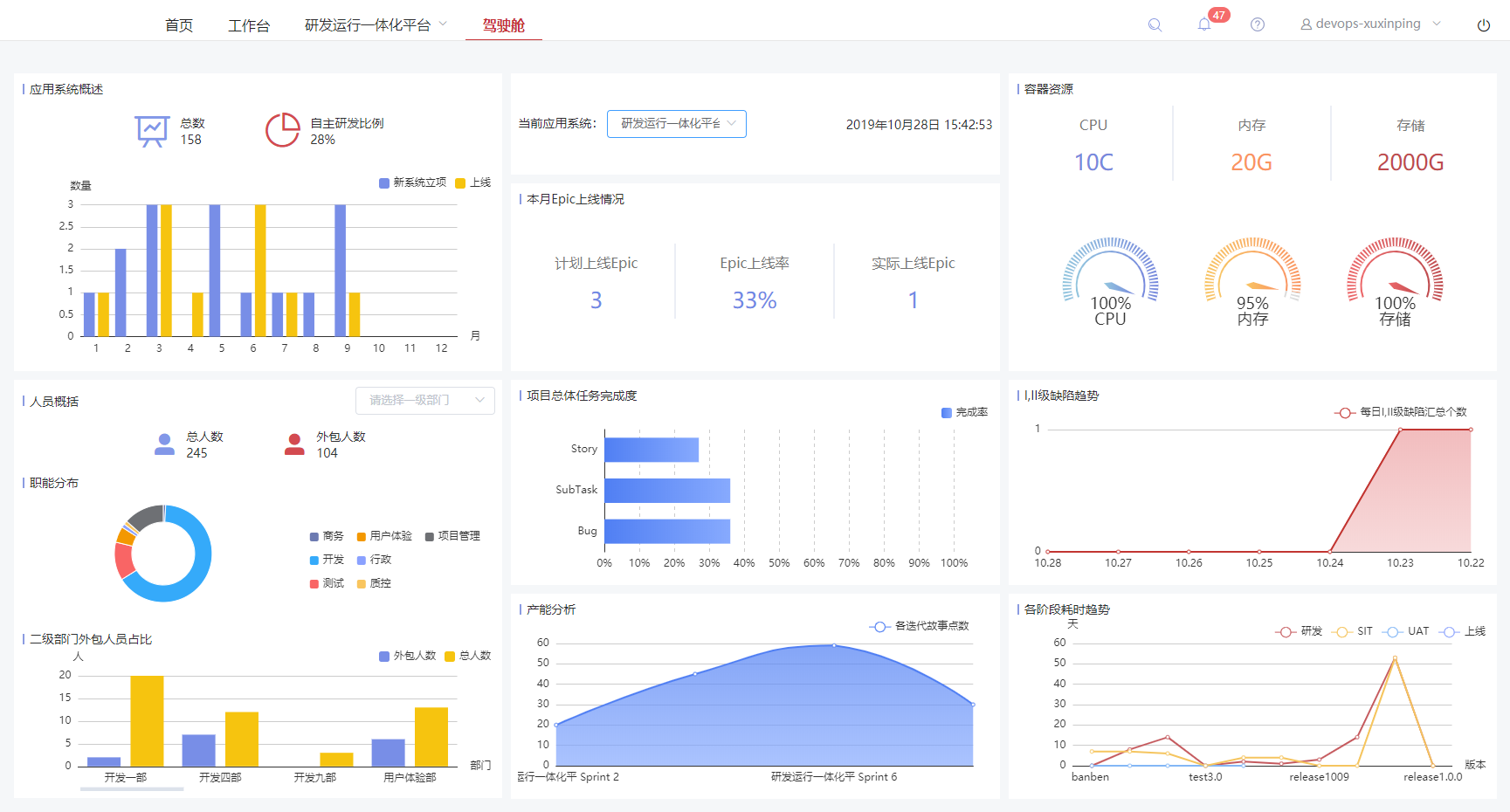
[11.3创建系统版本 82](#_Toc24632390)

[11.4查看工作流 82](#_Toc24632391)

# 平台总览

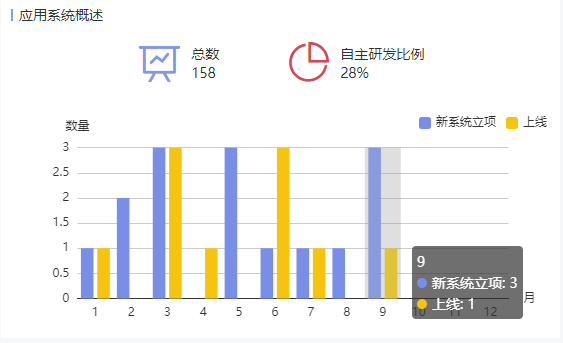
## 项目管理人员总览-驾驶舱

驾驶舱以卡片的形式显示当前平台的应用系统概述，人员概括，职能分析，二级部门外包人员占比，当前应用的本月epic上线情况，项目总体任务完成度，产能分析，容器资源，ⅠⅡ级缺陷趋势以及各阶段耗时趋势。



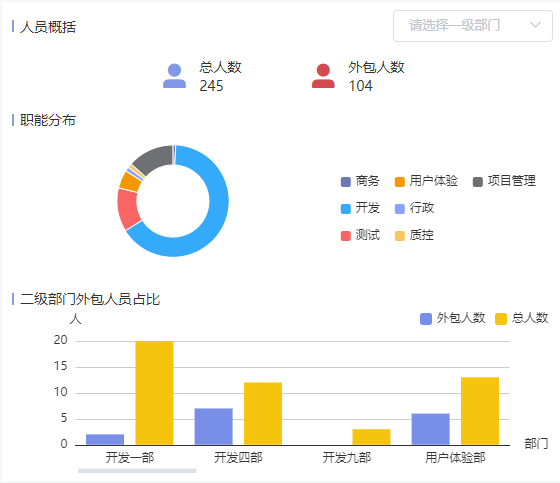
### 应用系统概述

应用系统概述卡片会列出当前记录在平台内每月上线数量和新系统立项。应用总数与自主研发比例是由admin进行驾驶舱配置。ⅠⅡ级缺陷趋势以及各阶段耗时趋势

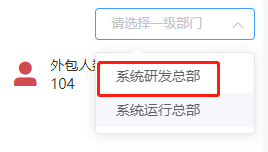


### 人员概括

人员概括卡片会列出当前平台内总人数与外包人数，并依据职能分布和二级部门外包人员占比进行图表展示

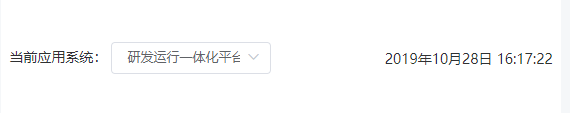


可根据一级部门筛选，以职能分析和二级部门外包人员占比进行查看



### 本月epic上线情况

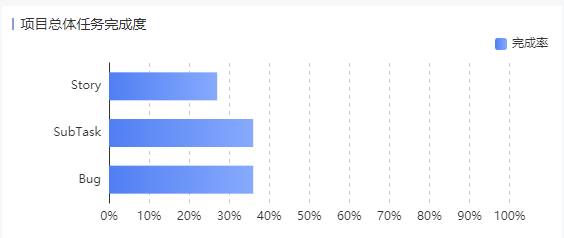
本月epic上线情况，项目任务完成度，产能分析，容器资源，ⅠⅡ级缺陷趋势以及各阶段耗时趋势这几个部分是应用研发情况各个不同指标的实时展示。可根据管理员管理的多个应用筛选展示



本月epic上线情况指当前应用在jira上建立的epic，并在当月完成上线数量统计和百分比展示

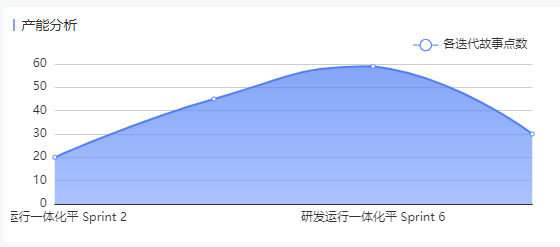
### 项目总体任务完成度

当前应用下story，story下关联的子任务和bug完成百分比展示



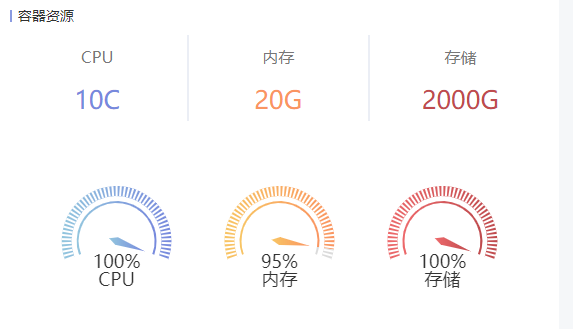
### 产能分析

当前应用各迭代已完成故事点数趋势展示



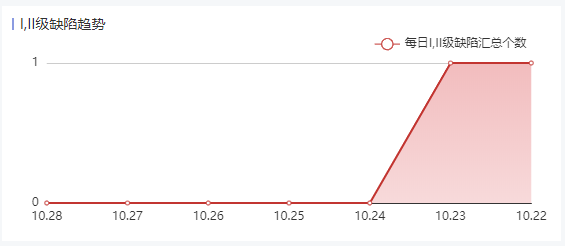
### 容器资源

当前应用下容器资源分配使用量展示



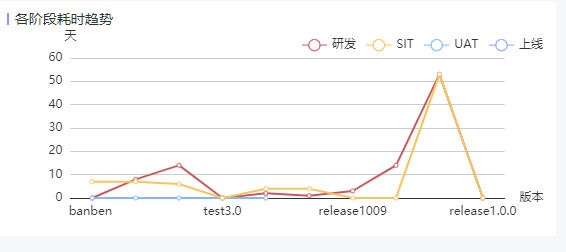
### ⅠⅡ级缺陷趋势

这里展示当前应用下今天及前六天每日ⅠⅡ级缺陷创建数量展示，注：这里统计的是当天创建总数



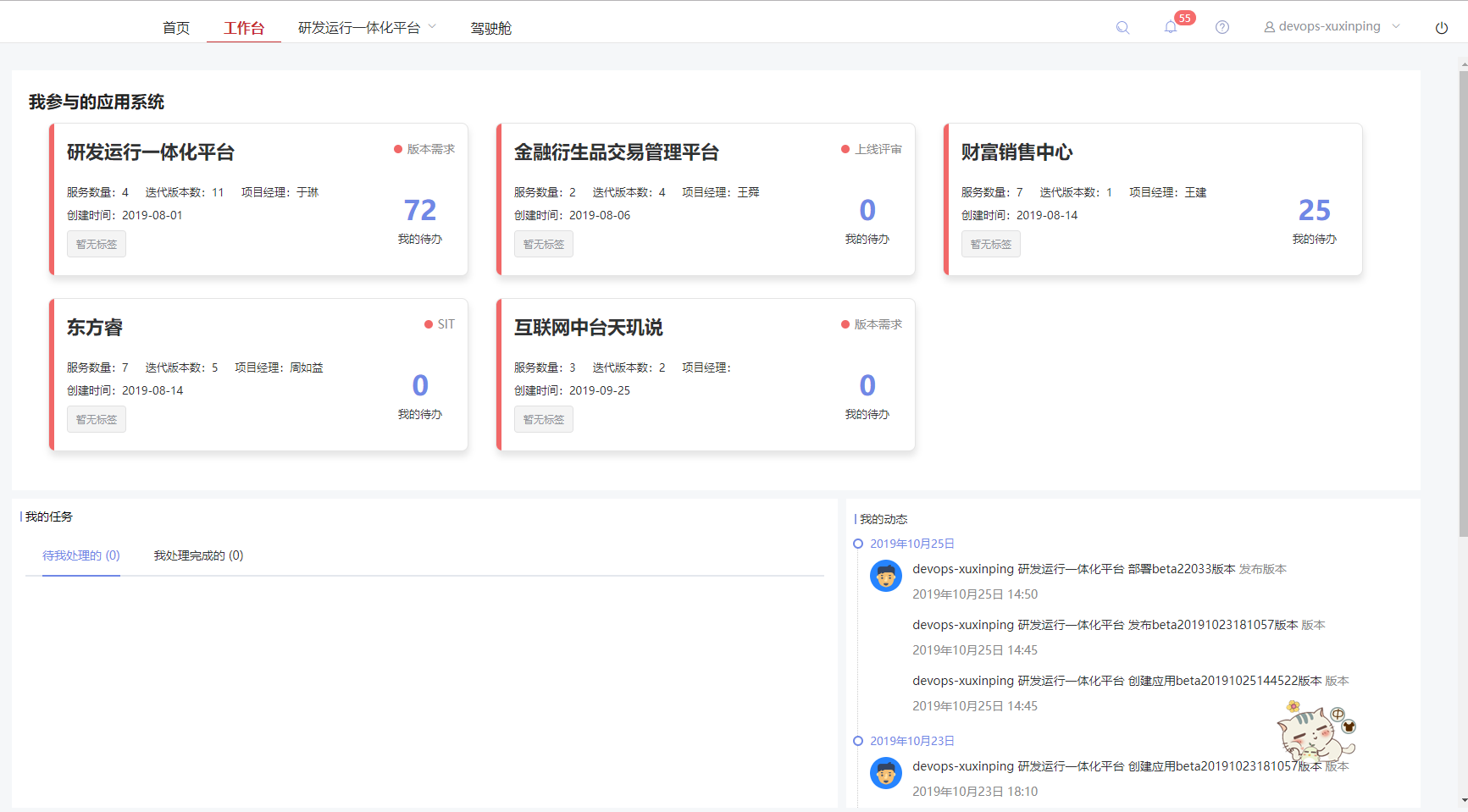
### 各阶段耗时分析

各阶段耗时分析卡片展示了当前应用不同版本不同阶段耗时情况，时间统计来源于研发运行工作流下方统计时间



## 工作台

工作台展示当前用户参与的所有应用，我的任务及动态以卡片的形式出现，点击卡片名称，可跳转至对应应用的概览首页



### 我参与的应用系统

以卡片的形式出现，点击卡片名称，可跳转至对应应用的概览首页

卡片上展示了应用名称、服务数量、迭代版本、项目经理，创建时间，当前用户的待办，以及最新版本的所在节点

待办表明当前用户在这个应用下有多少个待办的jira任务



### 我的任务

我的任务展示了当前用户在此应用下的需要处理的jira任务以及工作流任务如提测，部署



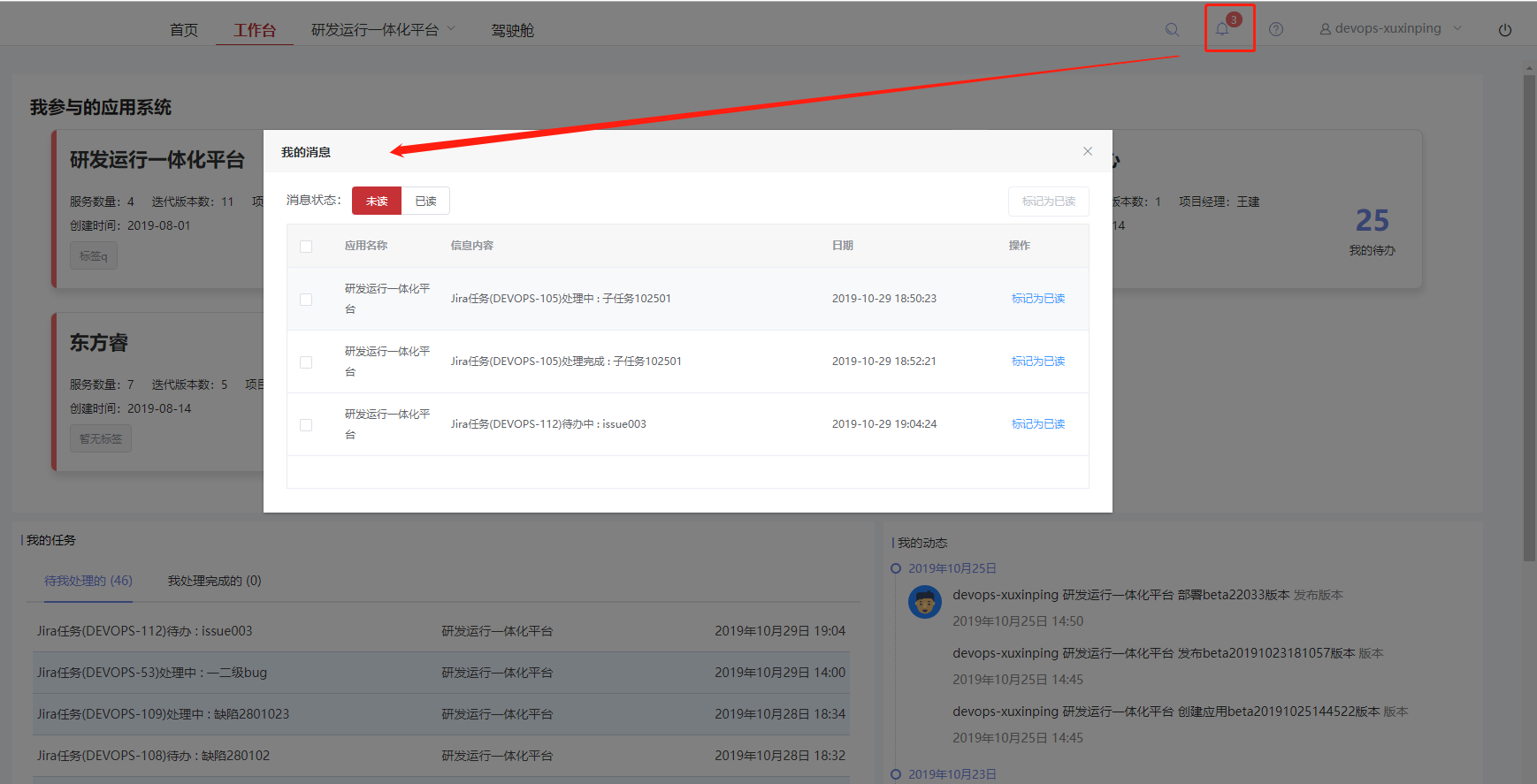
### 我的动态

我的动态展示了当前用户的操作记录



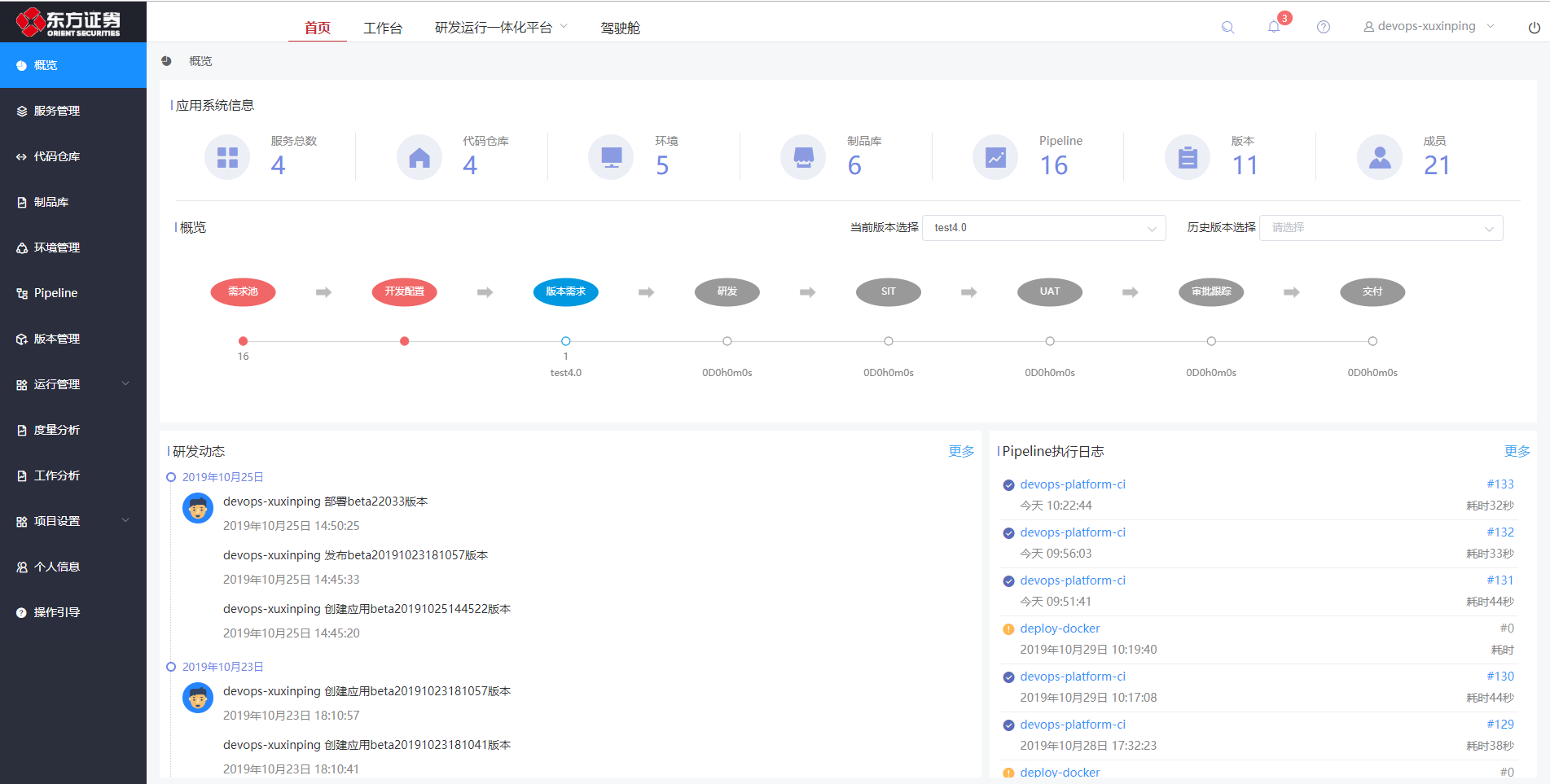
### 我的消息

我的消息这里展示了最新任务提示消息，包含当该用户被设置为jira任务的经办人，jira任务操作处理，流水线提测/部署等操作提示



## 首页

首页展示了当前应用的基本信息，包含应用系统信息，概览，研发动态和pipeline执行日志



### 应用系统信息

应用系统信息栏展示了当前应用的服务总数，代码仓库数，环境，制品库，pipeline，版本及成员，成员包含租户管理员和租户管理员导入的该应用的开发/测试人员



### 概览

概览展示了此应用最新版本采用的开发工作流及研发运行状态，各个节点功能如下：

a、需求池下方数字表明，当前应用没有关联版本的需求数量

b、开发配置，主要是配置该应用的代码仓库、申请环境、成员以及权限配置

c、版本需求，展示了应用当前版本规划需要完成的需求，包含story、缺陷和生产缺陷

d、研发，主要操作是提测即版本发布，当需求建立，由待办操作至开发中时，研发节点点亮，当需求完成到待测试时，可点击研发节点右侧sit提测按钮对待测试的需求进行过程版本提测，提测后，可点击研发节点的左侧构建按钮，跳转至pipeline页查看构建日志；

下方数字分子表示当前进入到开发及之后状态的需求，分母表示版本需求总数；

计时当前版本第一个需求在jira中进入开发中开始计时，当前版本最后一个需求在jira 中进入待测试结束计时

e、SIT，主要操作是对发布成功的提测版本进行版本部署、sit测试及UAT提测

下方数字分母显示当前版本需求数量，分子显示jira进入sit测试中及之后状态的当前版本需求数量

计时是当测试人员点击sit测试开始时开始计时，点击测试完成时结束计时，如果有多轮sit测试，最终将所有计时求和显示。

f、UAT，主要操作是uat版本部署、uat测试及上线申请

下方数字分母显示当前版本需求数量，分子显示jira进入uat测试中及之后状态当前版本的需求数量

计时当测试人员点击uat测试开始时开始计时，点击测试完成时结束计时，如果有多轮uat测试，最终将所有计时求和显示。

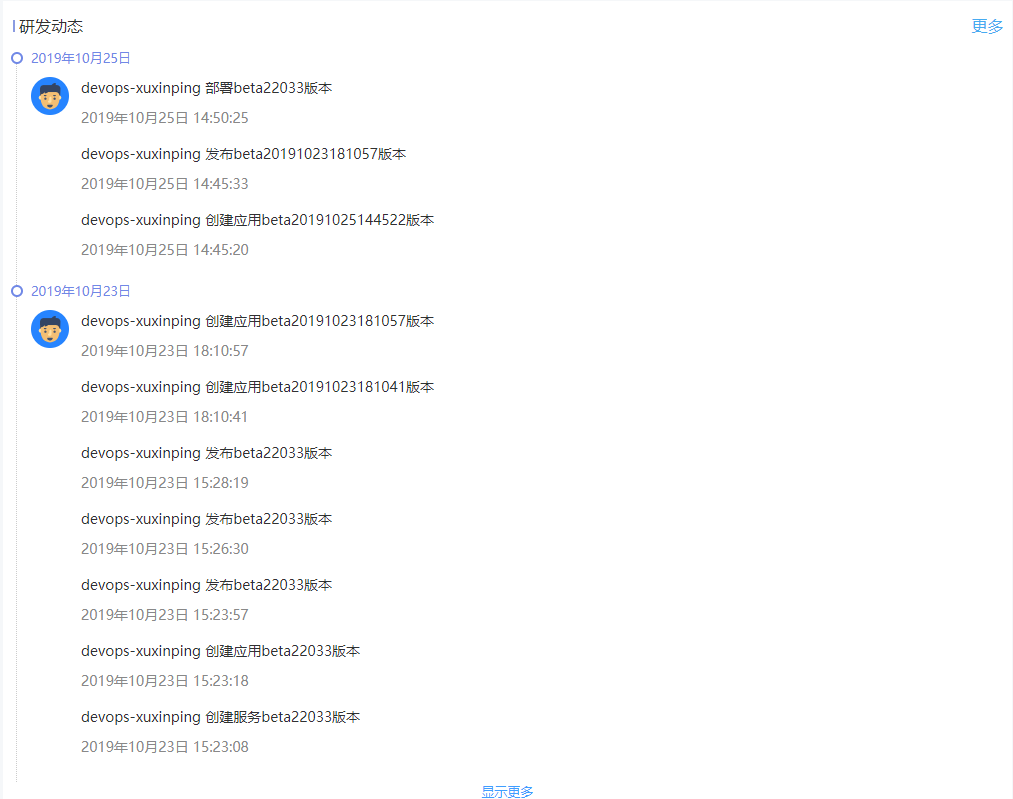
g、审批追踪，提交上线申请后版本

f、交付



### 研发动态

研发动态展示了当前用户操作记录，点击更多可以查看更多操作记录



### Pipeline执行日志

展示当前应用执行过的pipeline记录，点击pipeline名称，可进入pipeline详情，进行编辑，点击pipeline后面执行序号，可查看此次执行日志



# 服务管理

服务是DevOps平台应用系统的核心基础信息，代码仓库、Pipeline流水线、制品库、版本信息、版本发布都是围绕服务进行管理和操作的，这里要实现对服务的添加、编辑、列表显示、删除、Pipeline参数配置功能

需要注意的是:

1、平台已将jira上的应用纳管进来

2、应用关联的服务必须由平台创建

## 2.1服务管理

### 2.1.1服务的查询、新增、编辑和删除

1、服务列表

* 展示当前应用关联的多个服务
* 列表中可以查看服务名称，服务中文名，负责人。提供运行管理，编辑、删除入口；

2、查询

输入服务名称查询，支持模糊查询



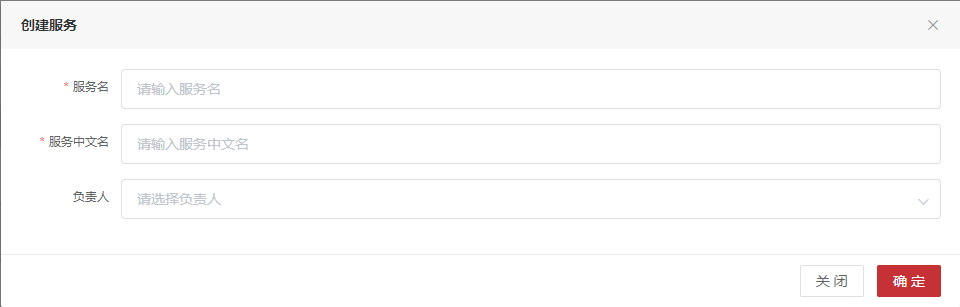
3、编辑，可对服务中文名，负责人，描述进行编辑，对于已经被引用即已经投入开发、版本构建的服务，不建议修改服务名称





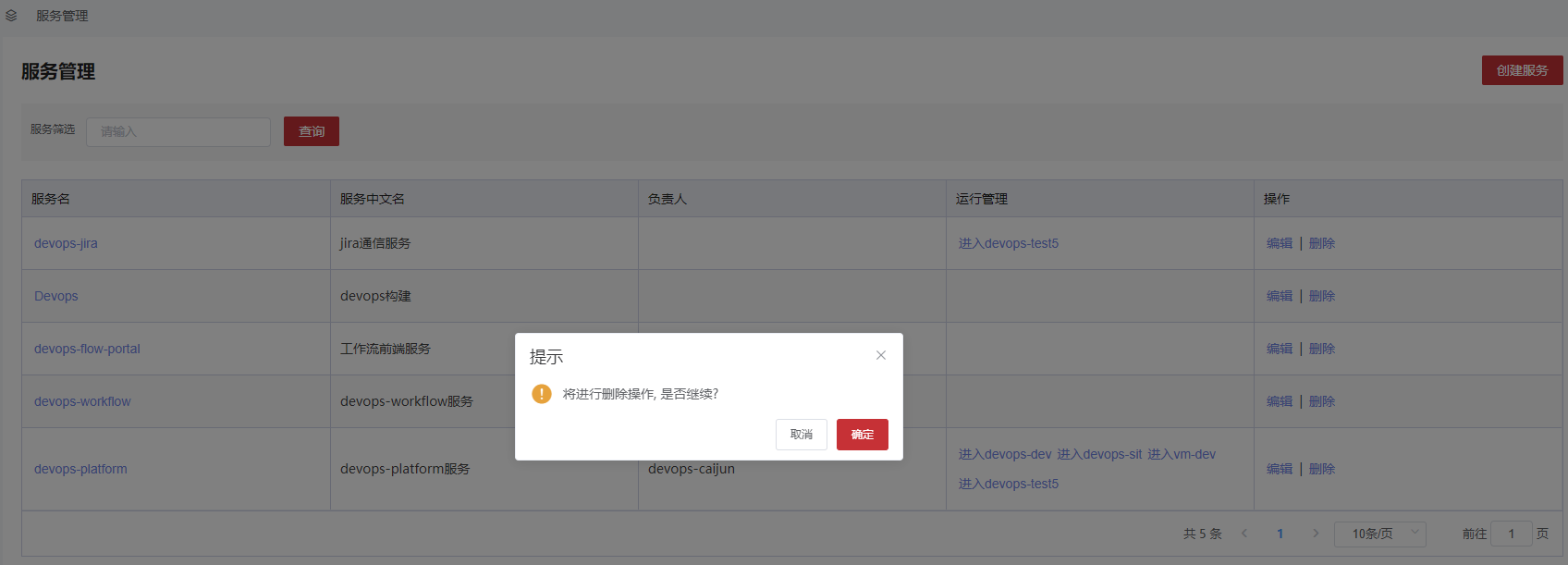
4、创建

点击创建按钮，进入页面，输入服务名称，服务中文名，负责人，点击确定。



5、删除

点击列表中的删除按钮，删除的前提，必须删掉服务下关联的pipeline、代码仓库、版本数据



6、运行管理

## 2.2pipeline

Pipeline是服务执行的流水线，包括源码检查、质量检查、编译构建、物理部署和容器部署等步骤，一个服务可能同时需要执行多条pipeline。

点击服务名称超链接，进入详情。

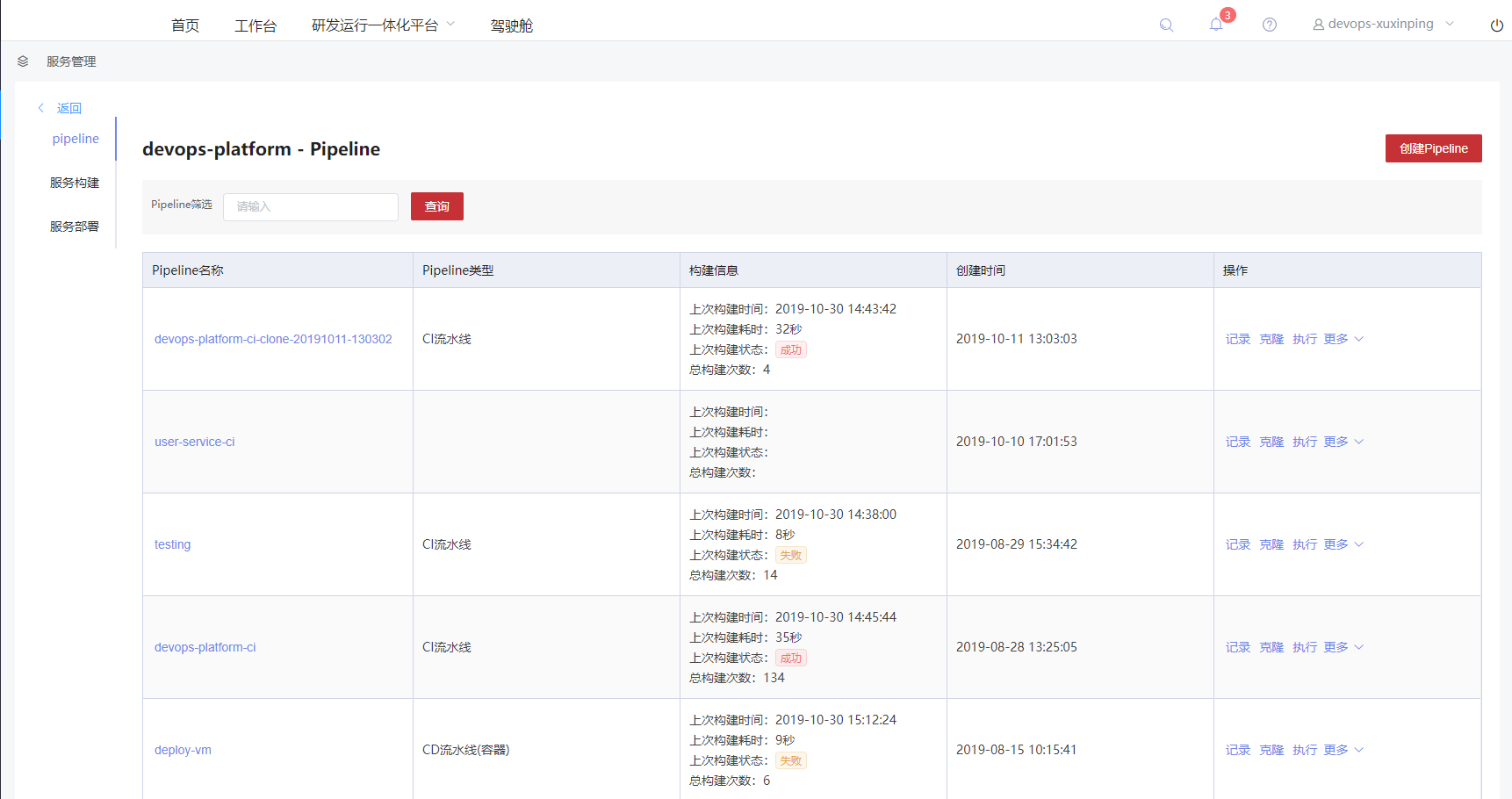


这里的pipeline和主界面中的pipeline功能基本一致



可以看到服务的pipeline列表，服务构建和服务部署。可以看到，展示服务的pipeline，有CI、CD（容器）、CD（物理）类型的pipeline

这里pipeline列表提供新增、编辑、删除、查看记录、克隆、执行等操作的入口；



### 2.2.1pipeline的查询、创建和定义

1、查询

输入pipeline名称查询，支持模糊查询

2、pipeline创建

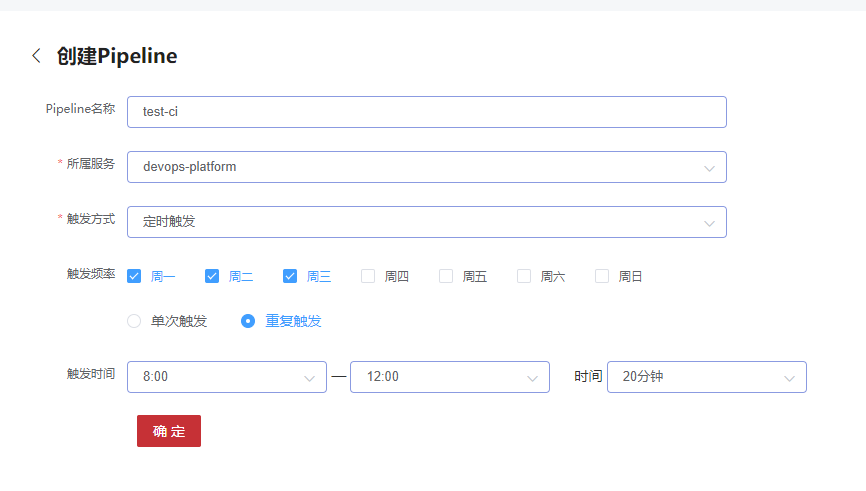
输入pipeline名称、所属服务及触发方式。



触发方式有人工触发，定时触发和自动触发。人工触发即需要手动点击执行才会触发；选择定时触发，需要输入定时触发的日期与时间，以及是单次触发还是重复触发。单次触发指指定日期在某个时间触发一次，触发频率可多选；



重复触发是指定日期内，某时到某时每间隔多长时间内触发，如下图所示，表示周一周二周三，8点到12点每隔20分钟触发此条pipeline，需要注意的是最后一次执行时间是12:40，而不是12:00

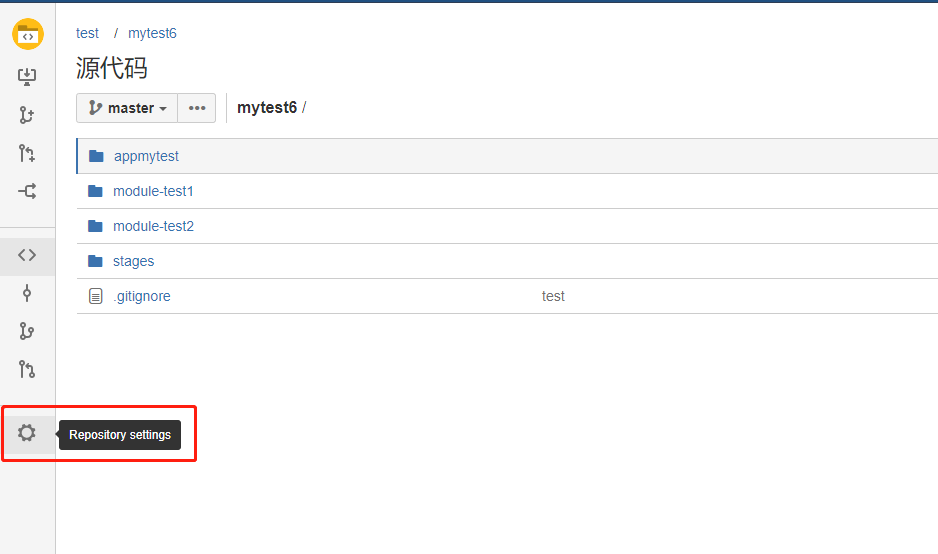


自动触发指开发者提交代码就会自动触发此条流水线，需要将此条pipeline的webhook添加到bitbucket的post webhook

初次创建选择自动触发下方是不展示生成webhook按钮的，需要将下方自定义配置完成后，重新编辑该pipeline，点击生成webhook



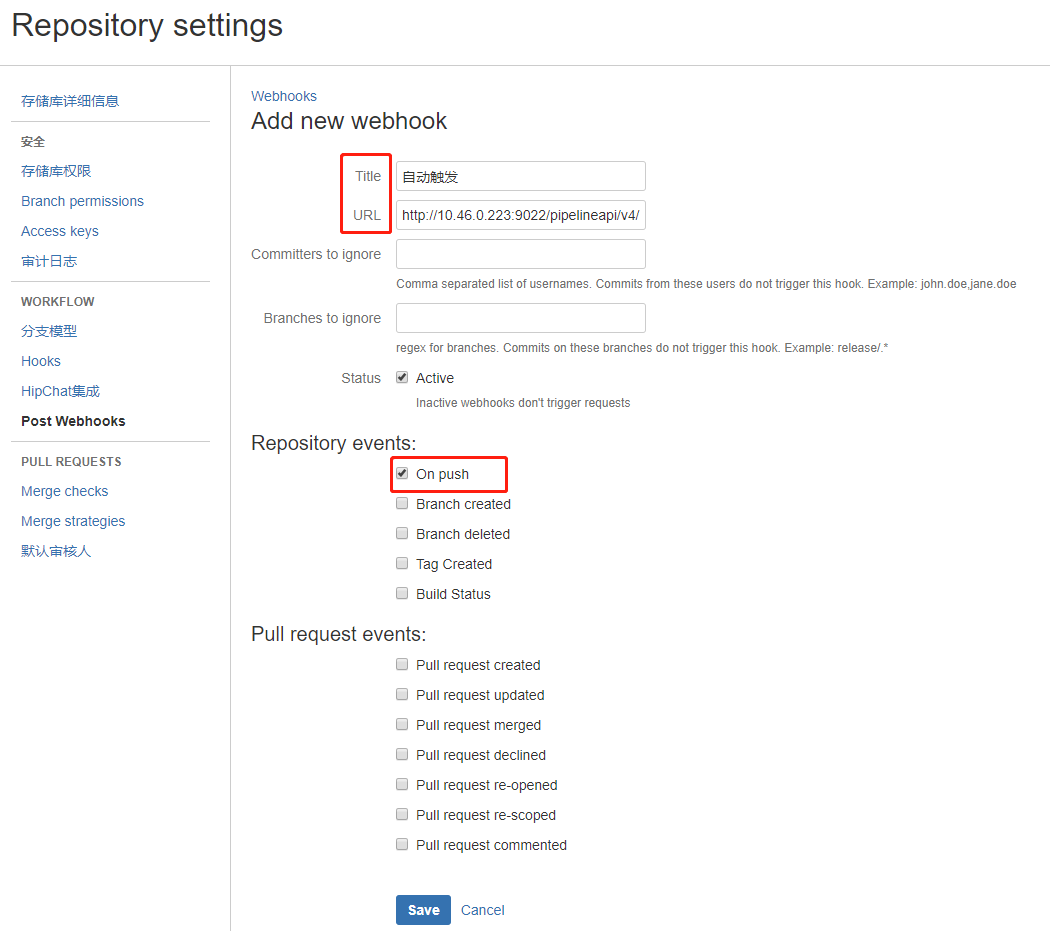
进入需要自动触发的服务代码仓库，选择settings



添加add webhook



将刚刚服务pipeline生成的webhook url贴进去

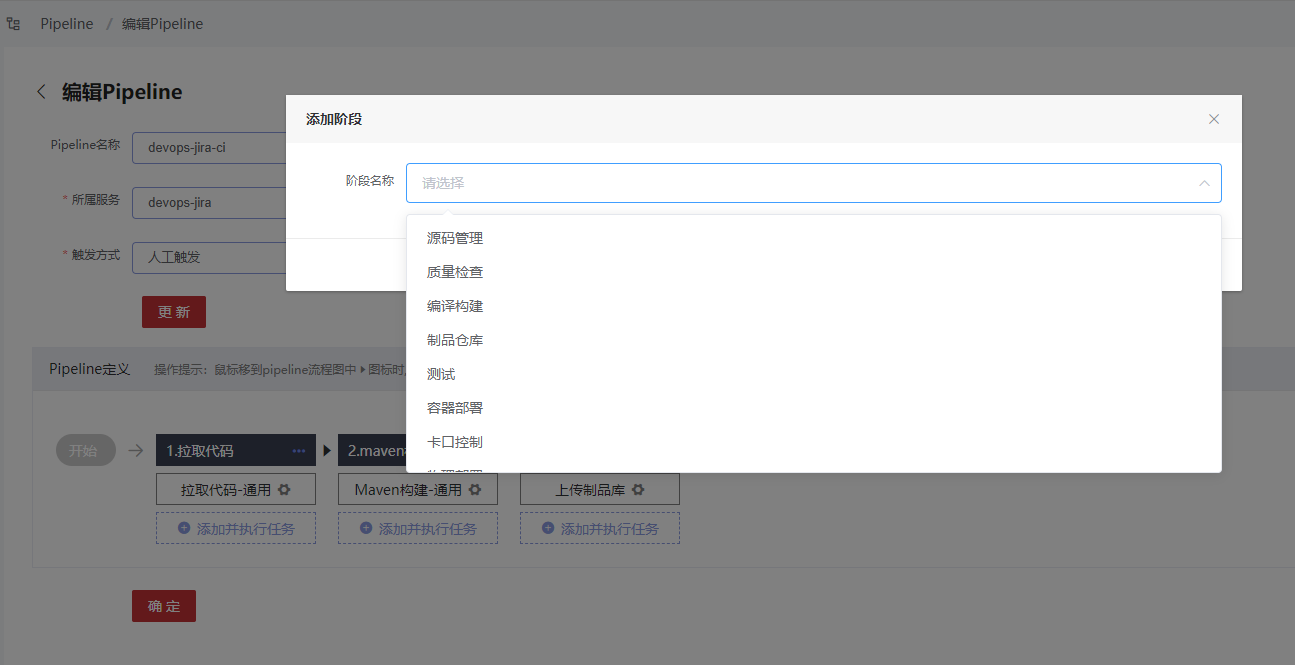


保存成功后，该服务pipeline自动触发设置成功，当提交代码时就会触发此条服务pipeline

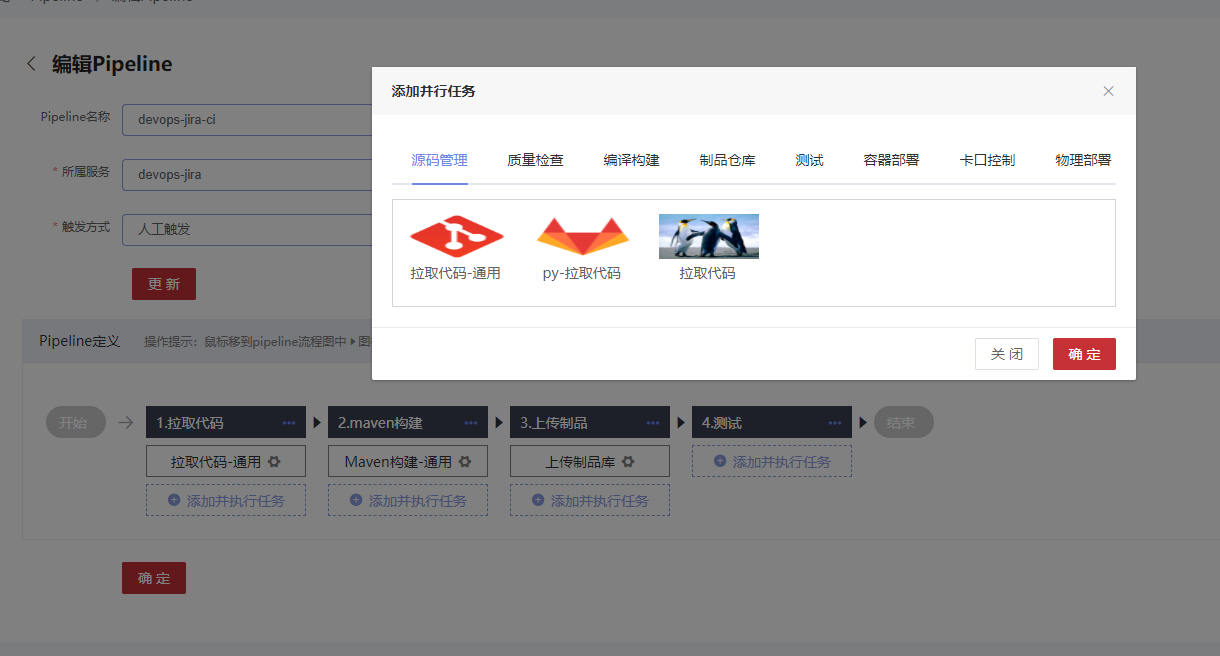
3、pipeline定义

指的是定义一条pipeline所拥有的处理阶段

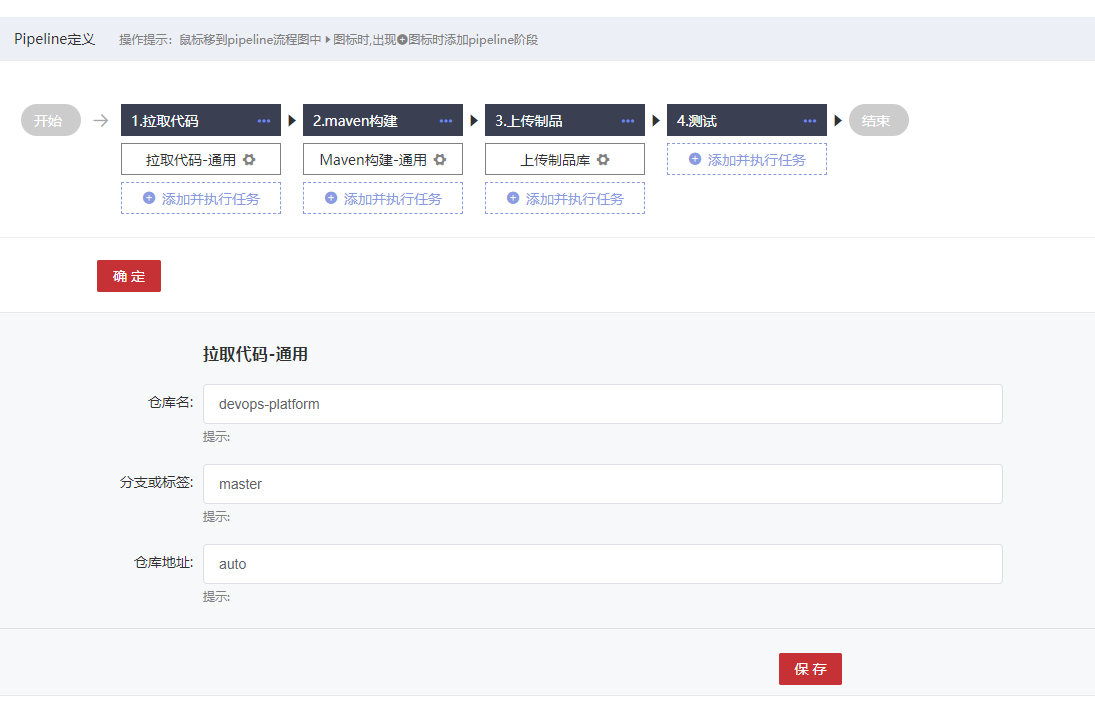


添加自定义阶段，如果是CI流水线，一般包含拉取代码、maven构建、sonar扫描和上传制品

下方添加任务模板，写入实际参数



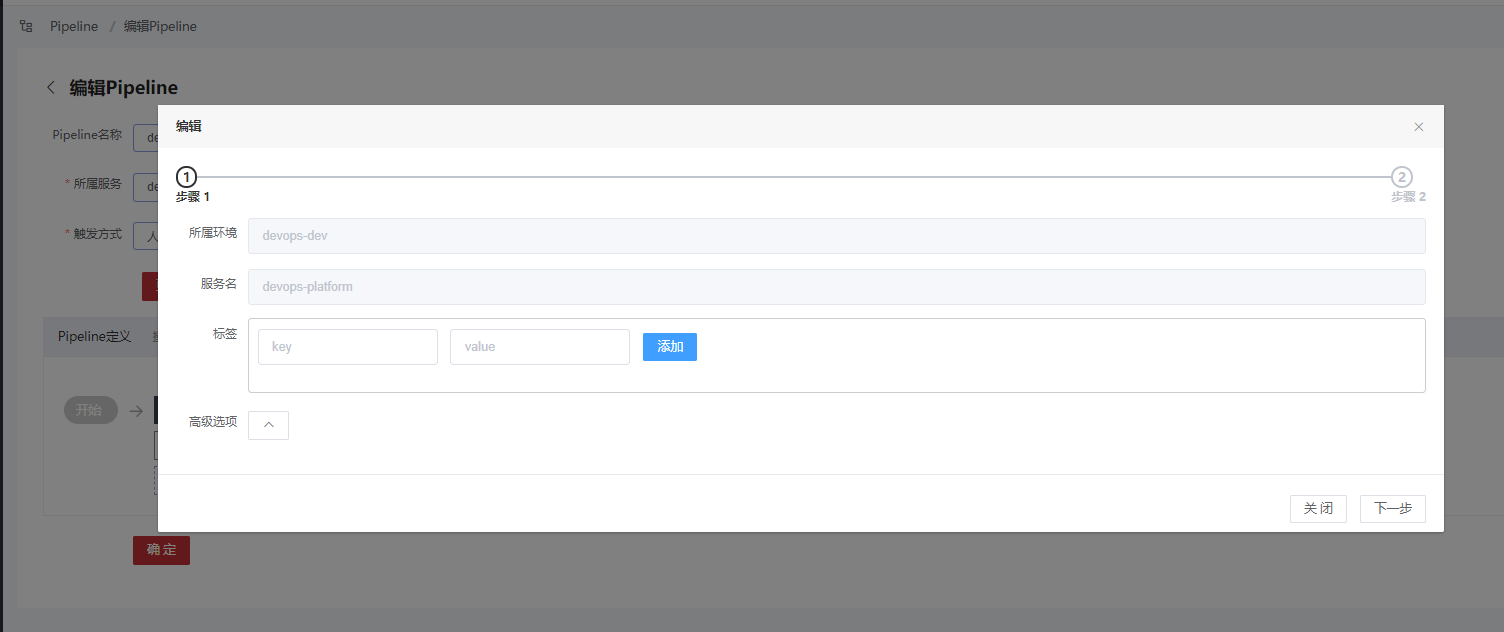
包含maven构建阶段，定义的Pipeline类型是CI流水线



包含容器部署阶段，定义的pipeline类型是CD流水线（容器）



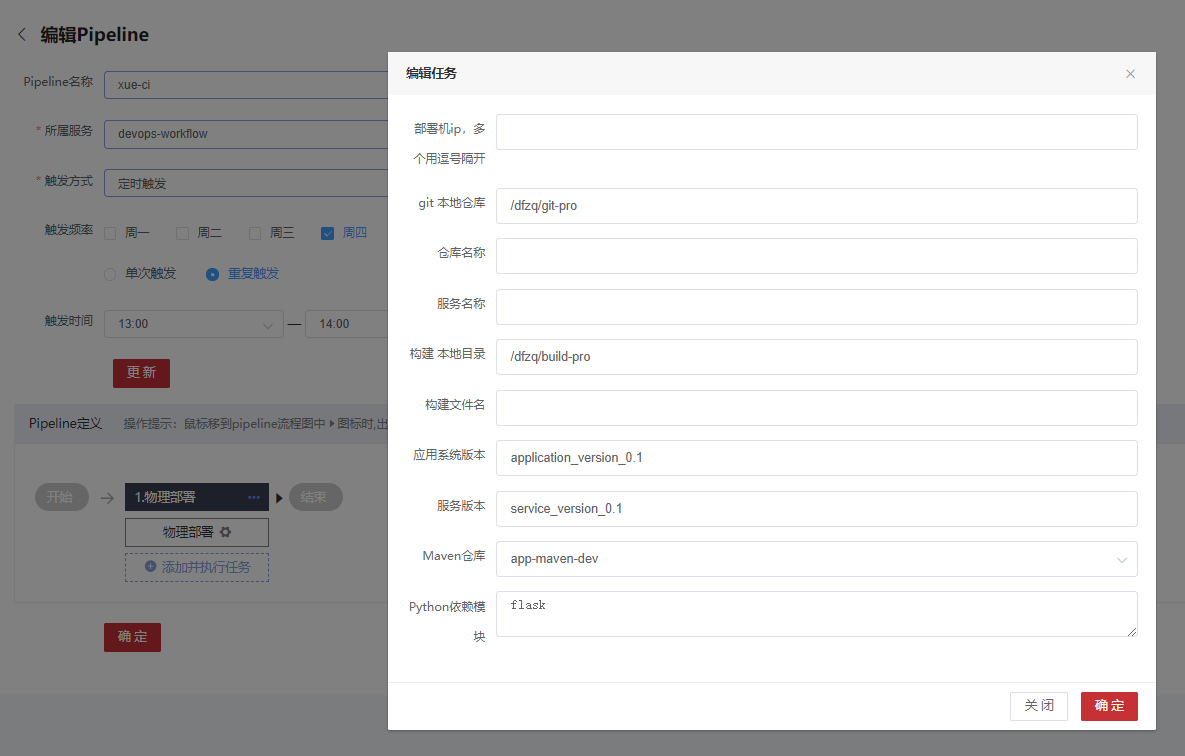
点击任务模板右侧设置按钮弹出容器部署参数



包含物理部署阶段，定义的pipeline类型是CD流水线（物理）



输入物理部署参数设置



### 2.2.2Pipeline的记录、克隆、执行与其它操作

1、执行

创建并定义完一条pipeline后，点击执行按钮开始执行此条pipeline。界面展示执行的进度以及执行的耗时、成功与否等状态信息；执行完毕，页面会自动跳转至此pipeline的执行记录界面。在此我们可以查看此条pipeline之前所有的执行记录和日志等信息

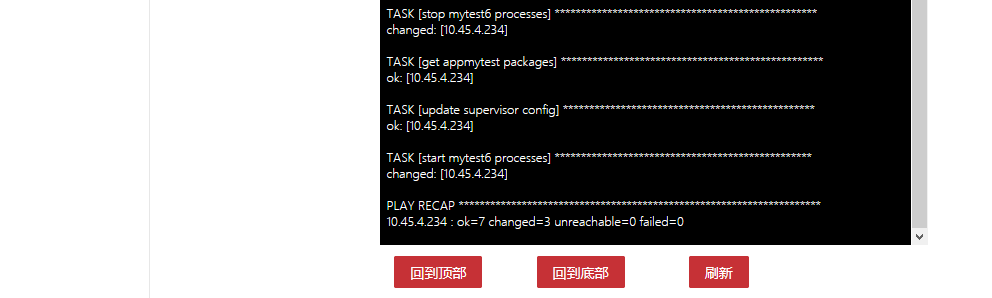


执行完毕后自动跳转至执行记录界面



查看此条pipeline阶段日志





2、记录

点击记录进入记录页面



可以查看一条pipeline的所有执行日志



3、克隆

可快速配置一条服务pipeline。点击克隆按钮，迅速克隆一条服务pipeline



需要注意的是克隆之后的触发方式和自定义的配置需要再根据实际服务配置更新



4、删除

点击更多下面的删除按钮可以删除此pipeline

### 2.3服务构建

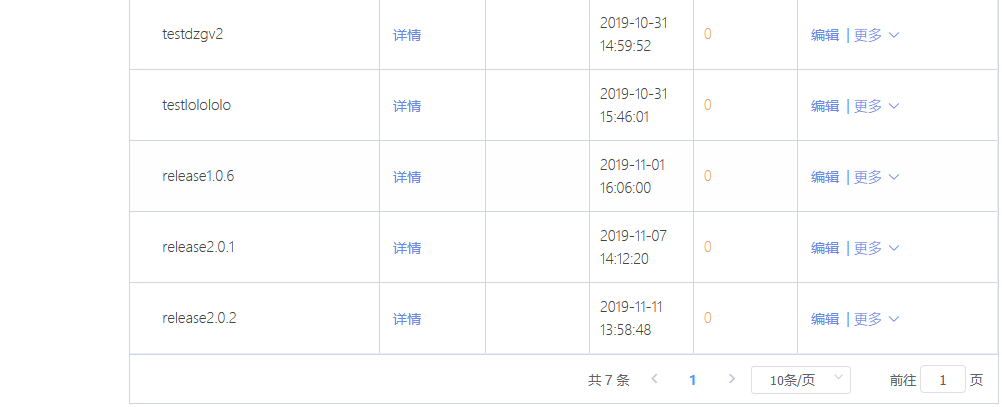
服务构建允许用户为同一个服务定义多个版本。

1、服务构建列表

* 服务构建列表展示服务版本、应用系统版本、执行pipeline、创建时间、规划需求数和可执行的各类操作
* 列表提供查询、编辑、删除、新增等操作的入口

2、输入服务名称查询，支持模糊查询





3、创建服务版本

创建版本之前首先需要此服务拥有构建或部署的pipeline，如图。名为cxf-serv的服务有两条pipeline，一条为ci流水线，一条为cd流水线(容器)



到服务构建里创建版本v1，此服务版本依赖以上两条pipeline，ci和cd，如图



成功创建了一个服务版本v1，它依赖两条pipeline，ci和cd



4、添加版本

V1版本创建完成后可以为其再次新增一个v2的子版本，如下





4、构建记录

查看服务当前版本构建记录，仅在此服务执行过后有记录，如下



5、编辑

修改服务所依赖的构建和部署pipeline

6、删除

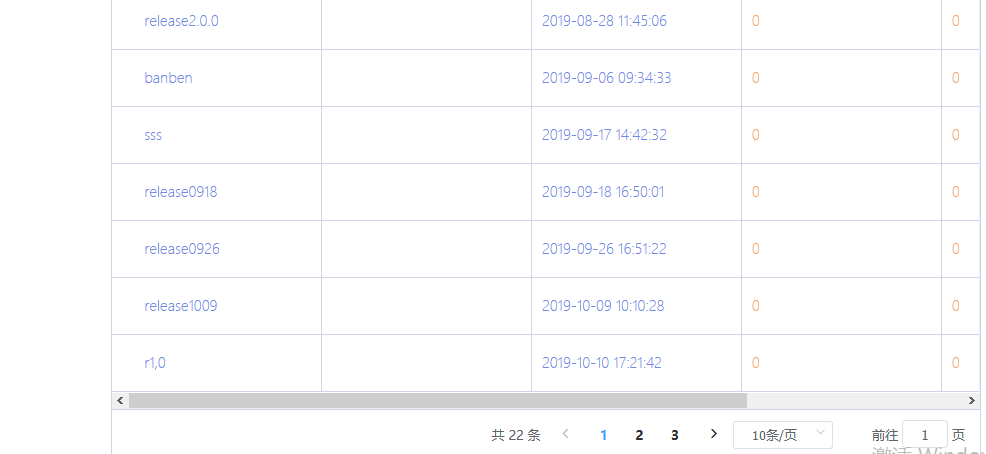
删除一条服务的版本

### 2.4服务部署

1、服务部署列表

* 服务部署列表展示服务版本、发版时间、创建时间、规划需求数和完成数量信息
* 列表提供查询入口、部署记录查询，支持模糊查询





# 代码仓库

代码仓库实现对各服务的代码仓库地址的管理的功能。

## 3.1代码仓库的查询、新增、编辑和删除

1、代码仓库列表

* 展示代码仓库中所有服务、服务所关联的仓库地址、用户凭证等信息
* 列表提供代码仓库的编辑、删除入口；

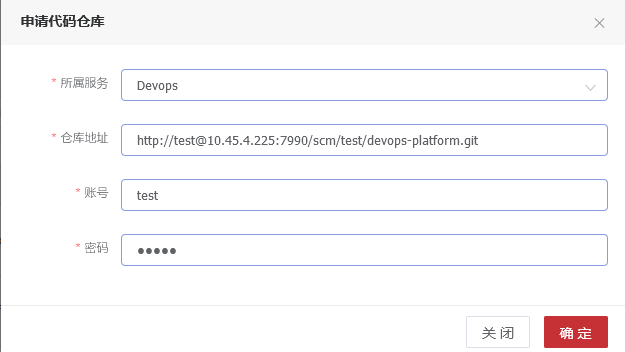


2、查询

输入服务名称、仓库地址或用户凭证等信息查询，支持模糊查询

3、创建代码仓库

点击右上角的创建代码仓库按钮，在弹出的选项卡中填写此代码仓库所属的服务、仓库地址和其账号、密码信息即可



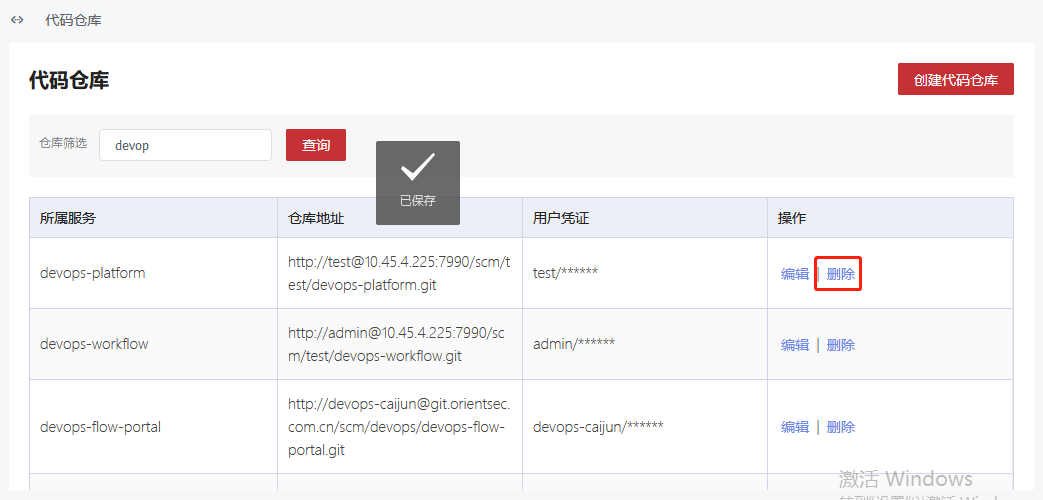
4、编辑

可以编辑一个已存在的代码仓库的信息。可以修改其所属服务、仓库地址和账号，一般一个服务仅可创建一个代码仓库，即服务与代码仓库所绑定，所以不建议修改所属服务和仓库的地址



5、删除

删除一个已存在的代码仓库，但是当代码仓库被引用时是不可以被删除的，只有引用(所关联的服务)删除后才能删除此代码仓库



仓库被引用无法删除



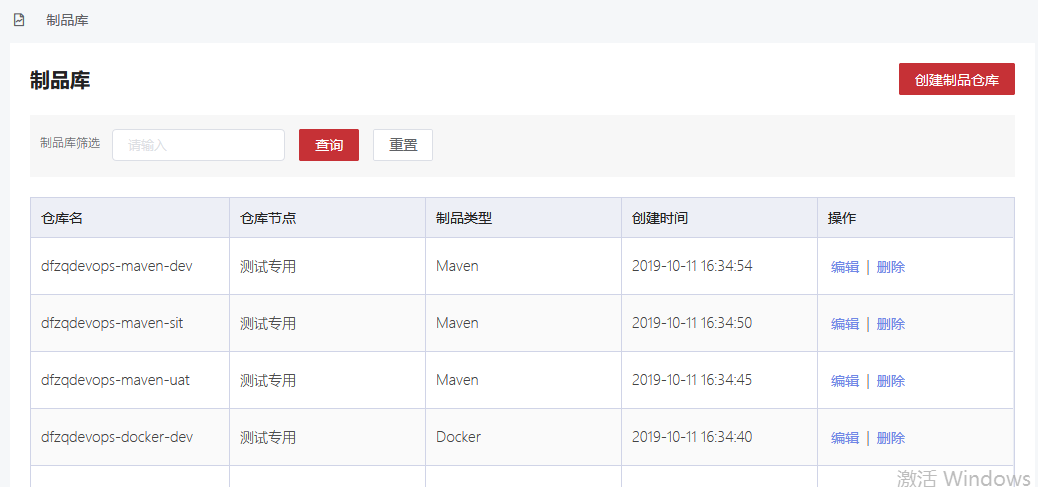
# 制品库

制品库是应用系统下服务成功构建后存放docker镜像或者二进制包的地方，在本项目中采用artifactory作为制品库工具支撑，在DevOps平台进行制品库管理，在创建制品库时，需要调用artifactory接口创建制品库，artifactory制品库中存放制品的默认目录

## 4.1制品库的查询、新增、编辑和删除

1、制品库列表

* 展示制品库中仓库名、仓库节点、制品类型、创建时间等信息
* 列表提供制品库的编辑、删除入口；

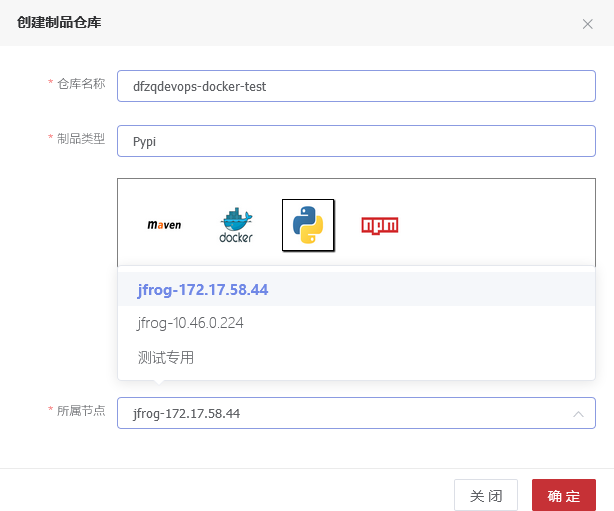


2、查询

输入服务名称、仓库地址或用户凭证等信息查询，支持模糊查询

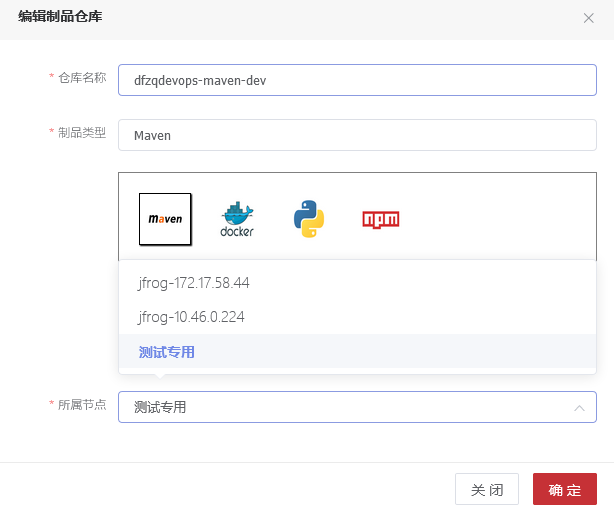
3、创建制品仓库

填写需要创建的制品仓库具体信息。注意，仓库名称的格式必须是：应用名称-类型-环境名，即关键字+docker/maven+dev/sit/uat ； 所属节点是具体的artifactory仓库



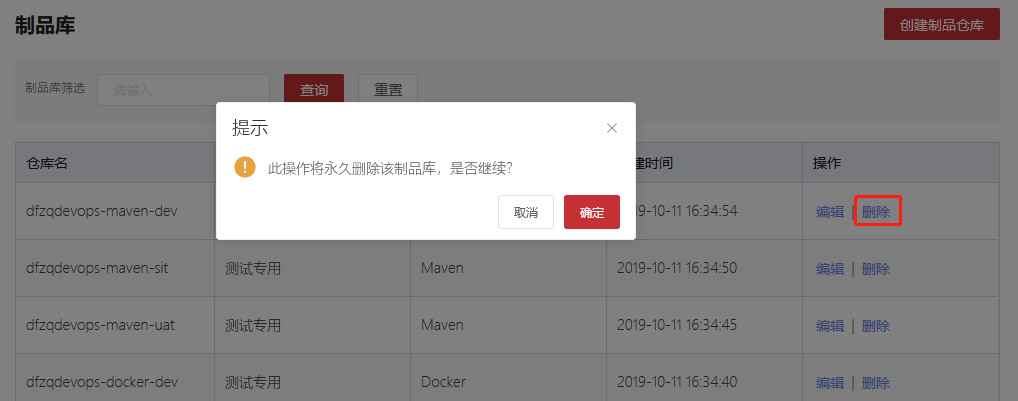
4、编辑

在编辑栏中可以编辑已存在仓库的仓库名称和所属节点，目前不支持修改制品的类型；并且所属节点也不建议修改



5、删除

点击删除按钮可以删除此制品库和其所关联的所有制品



# 环境管理

环境是指应用系统下部署服务的物理/虚拟机或docker资源空间的指向，一般是几台机器IP的集合或是K8S下的一个namespace，通过环境和资源空间关联关系的管理来为用户识别环境资源。

在DevOps平台创建docker资源类型的环境时，需要在BOC创建应用与之关联；创建物理/虚拟机资源的环境时，需要将机器ip、机器信息录入与之关联。

## 5.1环境的新增、编辑和删除

1、环境管理列表

* 展示环境的名称、资源类型、环境类型、配额等信息
* 列表提供制品库的编辑、删除和查询环境具体信息的入口；



2、查询

输入环境名称来查询环境信息，支持模糊查询

3、创建环境

创建环境中的资源类型指的是docker或者物理/虚拟机。若填写的是docker，则需要在BOC创建应用与之关联，并且设置应用的配额为DevOps录入的配额信息；若创建物理/虚拟机资源的环境时，需要将机器ip、机器信息录入与之关联，在这里物理/虚拟机资源可以添加多个配额。见下图

Docker类型环境



物理机/虚拟机类型环境



4、编辑

编辑选项可以提供编辑环境信息的入口。但是环境名称、资源类型、环境类型是不可以修改的。并且若是物理/虚拟机资源类型，在修改其ip、cpu、内存和存储之前需要先确保此环境中没有服务，否则会影响已存在的服务；若是docker资源类型，在修改其cpu、内存、实例和存储之前也需要确保此环境中没有服务，理由同上。



5、删除

点击操作下面的删除按钮可以删除此环境资源。在删除之前，需要先确保此环境中没有所需要的服务。



6、环境具体信息

点击环境名称超链接跳转至环境的详细信息页面中



下图是环境的详细信息页面，其中展示了环境包含的实例数量、服务总数、环境的物理ip列表、环境中包含的服务名称、进程号、状态等信息。用户可以启动/停止在当前环境中运行的服务



# 版本管理

应用系统的版本管理是在软件做一次发布时，对软件的需求、代码、制品和质量进行记录、跟踪、维护和控制的过程。应用系统在发布版本时，要记录应用系统本次发布所开发的需求、编译构建的代码，在版本发布完成后形成当前版本的制品信息和质量信息。应用系统的版本包括两种类型，分别是上线版本和测试版本，在一次上线版本形成之前，会发布多个测试版本进行测试，最终测试全部通过并确定上线后，最后的测试版本即封版为上线版本。

从应用系统的架构角度，应用系统由一个或多个服务组成，应用系统的版本是这些服务版本的合集，每个服务都是独立可发布的，所以应用系统版本和服务的版本是一对多的映射关系。

从应用系统版本发布的流程角度，版本有紧急版本和正常版本之分，他们的发布流程是有差别的，正常版本的流程一般为研发、sit测试、uat测试和上线，紧急版本的流程为研发、sit测试、和上线。

在DevOps平台周边相关集成平台和工具之间，DevOps平台的版本创建时要同步到Jira上，在Jira上创建需求做版本规划时，应将需求和版本进行关联，在DevOps平台则可以查看版本需求和需求提测等相关操作。

## 6.1版本构建

### 6.1.1版本构建的新增、编辑和删除

1、版本构建列表

* 展示应用系统的版本、它所依赖服务版本、规划需求数量、构建状态、创建时间等信息。
* 列表提供版本的新增、编辑、删除和查询等操作的入口；

2、查询

输入应用系统名称来查询版本构建信息，支持模糊查询



3、创建版本

填写需要创建的应用系统详细信息，就可以新增一个应用版本，如下图

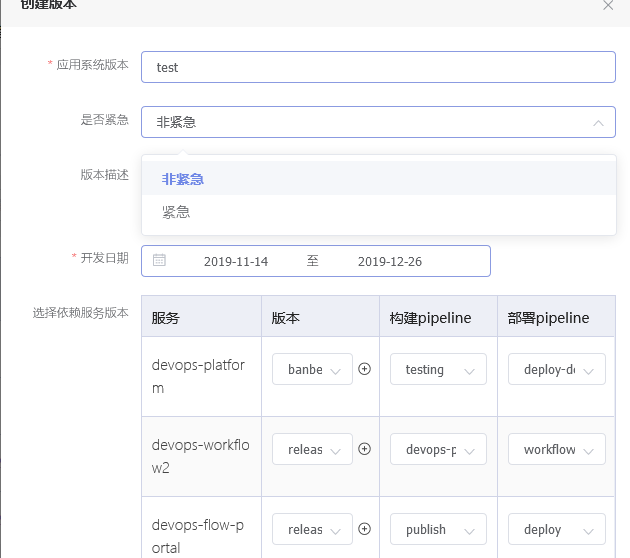
下图中的应用系统版本是指当前应用的版本号 ；

是否紧急可以表示当前应用版本的紧急与否 ；

开发日期确定了当前应用版本的开发日期 ；

依赖服务版本指的是当前应用版本所依赖的服务版本，同一个应用可以有多个版本，并且同一个应用可以依赖一个或者多个服务，即一对多的关系。服务依赖版本中的服务名、服务版本、构建pipeline和部署pipeline都和服务管理中的相应选项相对应

若不选择当前应用所依赖的服务版本，则系统自动令其依赖所有服务，并默认为所有的服务创建与当前应用同名的版本



1. 依赖服务版本

点击依赖服务版本下面的详细超链接可以看到当前应用版本所依赖的服务版本信息

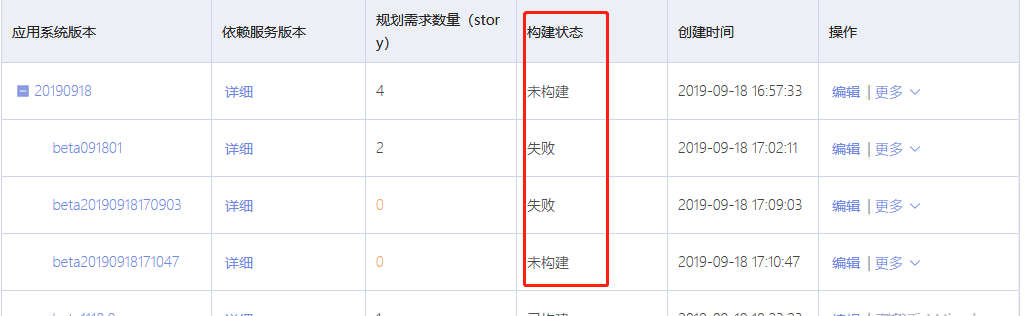


当前应用版本所依赖的服务版本中展示了所依赖的服务名称和其版本号



5、构建状态

构建状态描述了当前应用版本的构建成功与否



1. 编辑

编辑可以修改一个应用版本所依赖的服务版本等相关信息

1. 添加版本

点击操作/更多 下面的添加版本按钮可以为某个应用新增子版本，下图名为20190918版本的应用在添加版本以后包括了beta091801、beta20190919179093、beta10290918171047、beta1110.0和beta10290925173017这几个子版本



1. 版本构建

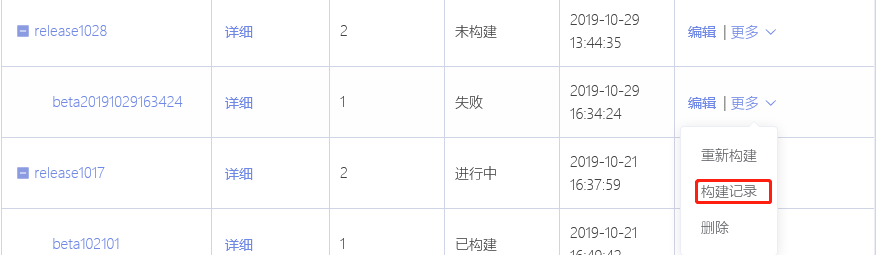
已经构建成功的应用可以查看其构建版本，如下图。

构建版本中包括了服务的版本号、服务名，story、ci、cd、sit测试和uat测试等信息





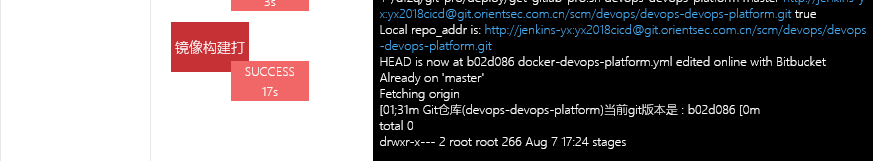
子版本的构建记录可以看到其构建记录如图





点击查看可以看到它的执行记录





重新构建可以重新构建此版本的应用，需要选择其版本号、pipeline和分支名称





## 6.2应用系统的处理逻辑

### 6.2.1研发流程节点显示处理逻辑

1. 流程节点状态颜色意义

* 灰色节点：灰色代表未开始
* 橙色节点：为节点进行中的状态
* 蓝色节点：为节点已成功完成的状态
* 红色节点：为节点失败的显示状态

1. 流程节点流转状态变化处理逻辑
2. 当版本规划结束，且版本有关联需求，研发流水线形成
3. 需求池节点默认显示进行中，代表需求是持续提出的。
4. 开发配置节点下各配置项，检查是否有配置信息，如果没有，则显示未开始，如果有至少一项未配置，则显示进行中，如果全部项都有配置信息，则显示完成状态。
5. 版本需求默认显示完成状态
6. 版本需求之后的状态全部为未开始。
7. 研发节点状态
8. 当前版本有一个需求进入研发中状态，则研发节点状态为进行中。
9. 当版本上线申请审批通过，则研发节点状态为完成。
10. SIT节点状态
11. 当研发经理提sit测试后，SIT节点状态为进行中。
12. 当某次sit测试失败，在下一次sit测试成功之前，SIT节点状态为失败。
13. 当某次sit 测试成功，如果此次测试包含版本所有需求，则SIT节点状态为完成，否则为进行中。
14. UAT节点状态
15. 当测试经理提uat测试后，UAT节点状态为进行中。
16. 当某次uat测试失败，在下一次uat测试成功之前，UAT节点状态为失败。
17. 当uat测试成功，UAT节点状态为完成。
18. 上线审批节点状态
19. 当项目经理发起上线申请，审批结束之前，上线审批节点状态为进行中。
20. 当上线申请审批通过后，上线审批节点状态为完成，反之则为失败。
21. 已上线节点状态
22. 当应用运维成功上线后，已上线节点状态为完成。
23. 流程节点计数处理逻辑
24. 需求池节点：显示jira未关联版本的需求数量
25. 版本需求节点：显示jira关联当前版本的需求数量
26. 研发节点：分母显示当前版本需求数量，分子显示jira进入开发中及之后状态的当前版本需求数量
27. SIT节点：分母显示当前版本需求数量，分子显示jira进入sit测试中及之后状态的当前版本需求数量
28. UAT节点：分母显示当前版本需求数量，分子显示jira进入uat测试中及之后状态当前版本的需求数量
29. 流程节点计时处理逻辑
30. 研发节点：当前版本第一个需求在jira中进入开发中开始计时，当前版本最后一个需求在jira 中进入待测试结束计时。如果此时再插入需求，则再次按这种方法计时，最终将所有计时求和显示。
31. SIT节点：当测试人员点击sit测试开始时开始计时，点击测试完成时结束计时，如果有多轮sit测试，最终将所有计时求和显示。
32. UAT节点：当测试人员点击uat测试开始时开始计时，点击测试完成时结束计时，如果有多轮uat测试，最终将所有计时求和显示。

### 6.2.2研发、测试、运行一体化

应用研发、测试、运行一体化平台操作流程的简述

1、需求、研发

首先用户创建名为mytest6的服务并为其创建构建和部署的pipeline，并为其添加版本。参考第二章服务管理，服务的创建、pipeline的创建、服务构建等章节。下图是创建好的名为mytest6的服务



Mytest6服务所依赖的pipeline(ci、cd)，如下



Mytest服务创建的服务版本，如下



创建应用版本，此处应用已存在，名称为DevOps，新建的版本为vs11。并且当前应用版本依赖上面创建的mytest6服务。可以参考第六章版本管理6.1版本构建

去版本管理中查看新建的应用版本vs11，可以看到已经成功创建



点击详细，查看其依赖的服务版本，查看无误，为mytest6，mytest6服务的版本号为cxf-serv-v1

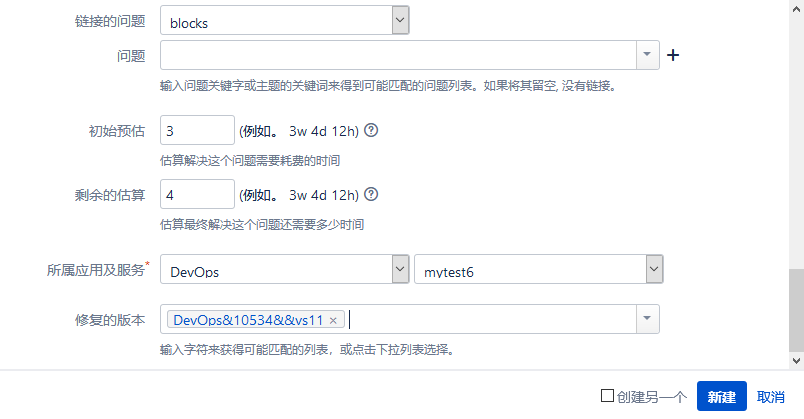


上面所有操作也可以去操作引导中执行，可参考第十一章操作引导

接下来研发经理去jira中DevOps项目创建问题，具体参数如下

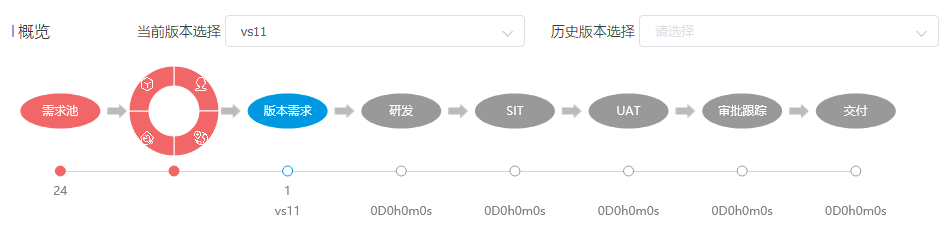


下图所属应用DevOps即为当前应用，应用版本vs11依赖的服务为mytest6.修复的版本指上面再平台中创建的应用版本vs11

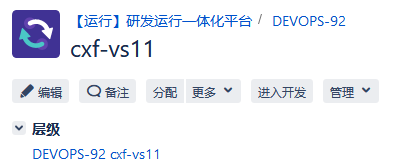


到概览页面中选择版本，找到上图创建的修复版本vs11，需要先进行开发配置

配置服务的代码仓库、成员、权限，并进行环境的申请，可参考第三章代码仓库的配置、第十章的10.1用户管理、10.2角色管理和第五章环境管理等章节



配置好代码库、成员、环境等资源后，进入jira，进入开发阶段



可以进入版本需求查看当前应用的版本、jira中的问题编号及其所处阶段等信息

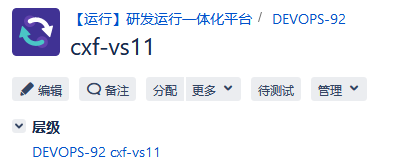




在这里，

1. 研发在规划版本需求开发过程中，完成一部分需求的开发，研发经理就可以进行sit提测，并且可以多次提测，提测选择的需求应该从当前规划版本完成需求中进行选择，并且有两类需求不可以选择，分别是：a）在此之前有提测单A（或者多个提测单）的状态处在测试未开始和进行中，那之前提测单A（或者多个提测单）中提测过的需求不可以选择 ；b）在此之前有提测单B（或者多个提测单）状态处在测试结束，那与提测单B（或者多个提测单）中未通过测试的需求相关联的服务的所有需求都不可以选择。
2. SIT提测界面的SIT提测版本下拉列表中显示的为当前上线版本下未发布测试版本号，如果没有任何测试版本号，也可以快速新建测试版本号。其它部分提测试操作处理逻辑同“版本管理”模块的测试版本发布处理逻辑一致。
3. 将Jira中当前提测的所有需求状态变为sit待测试。
4. SIT提测完成，SIT节点的部署按钮变为可点击状态。
5. 测试人员通过SIT节点进入SIT版本部署界面时，列表显示SIT提测单，选择即将要测试的测试单进行部署，部署处理逻辑和“发布版本”模块版本部署处理逻辑一致。
6. SIT部署完成，SIT节点的开始测试按钮变为可点击状态。

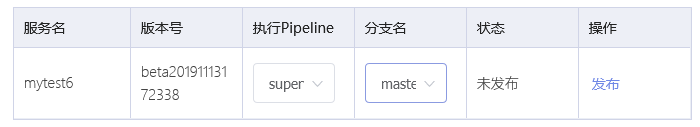
进入jira点击待测试，进入待测试阶段



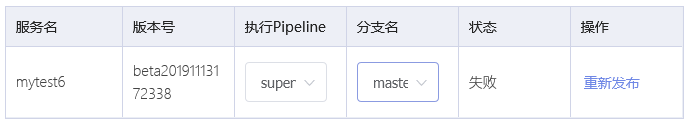


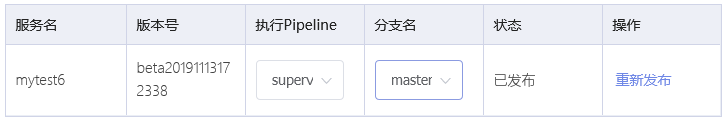
发布sit提测版本。Sit版本发布失败后可以多次提测。

可以查看服务的发布状态，并且可以多次发布









开始sit部署和测试





查看需求详细



Sit部署



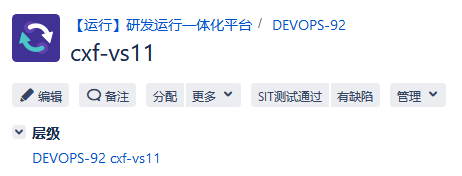
部署完成，Sit测试开始

1. SIT开始测试处理逻辑
2. 测试人员通过SIT节点进入SIT开始测试界面时，列表显示SIT提测单，选择即将要测试的测试单点击开始测试，更新提测单状态为测试中。
3. 将Jira中当前测试单的所有需求状态变为sit测试中。
4. SIT开始测试后，SIT节点的测试完成按钮变为可点击状态。
5. SIT测试完成处理逻辑
6. 测试人员通过SIT节点进入SIT测试完成界面时，列表显示SIT提测单，选择即将要测试完成的测试单点击测试完成，弹出输入测试结果页面，有两种情况：
7. 如果选择不通过，输入测试反馈信息，点击提交结束。
8. 如果选择通过，则系统自动检查a)、提测单的需求在jira中是否全部进入了sit测试完成状态；b)、当前提测单的需求在jira中有没有一二级的BUG;如果a条件不满足，则提示不可设置通过测试；如果a条件满足，b条件不满足，则提示存在一二级BUG，是否仍然要通过测试，如果选择继续通过，则将此次存在一二级BUG的测试通过操作进行记录审计。
9. 当所有需求SIT测试通过后,提测单状态变为测试通过，SIT节点的UAT提测变为可点击状态。

进入sit测试



若有缺陷，测试不通过



测试不通过



再次部署提交测试，当缺陷修复时，在jira中修改状态缺陷修复或sit测试通过







在平台中点击测试完成并提交



查看sit测试结果





sit测试完成进入Uat提测和部署阶段

1. 当测试人员经过几轮sit测试后，当前规划版本需求全部开发完成，测试人员认为当前某个sit测试版本（比如sit0.5）可以进入uat测试时，由测试人员进行uat提测（比如提测版本为uat0.6），那么与sit0.5相关联的需求编号、git branch和tag、制品包、质量信息都同步和uat0.6也进行关联，处理方式参考“版本管理”上线版本发布处理逻辑。

Uat提测



查看提测需求



点击uat提测，进入uat提测阶段

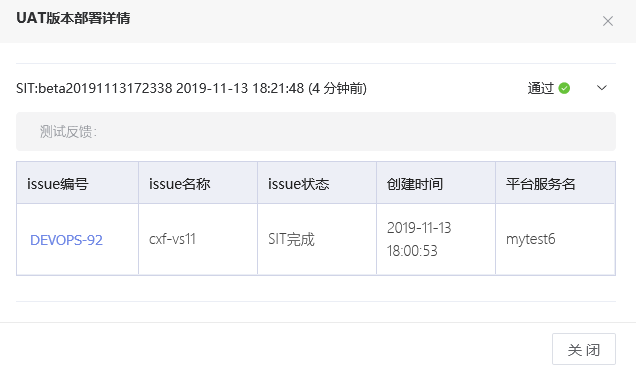




部署uat版本



查看uat版本部署详情



部署完毕，开始测试

1. UAT开始测试处理逻辑
2. 测试人员通过UAT节点进入UAT开始测试界面时，列表显示UAT提测单，选择即将要测试的测试单点击开始测试，更新提测单状态为测试中。
3. 将Jira中当前测试单的所有需求状态变为uat测试中。
4. UAT开始测试后，UAT节点的测试完成按钮变为可点击状态。
5. UAT测试完成处理逻辑
6. 测试人员通过UAT节点进入UAT测试完成界面时，列表显示UAT提测单，选择即将要测试完成的测试单点击测试完成，弹出输入测试结果页面，有两种情况：
7. 如果选择不通过，输入测试反馈信息，点击提交结束。
8. 如果选择通过，则系统自动检查a)、提测单的需求在jira中是否全部进入了uat测试完成状态；b)、当前提测单的需求在jira中有没有一二级的BUG;如果a条件不满足，则提示不可设置通过测试；如果a条件满足，b条件不满足，则提示存在一二级BUG，是否仍然要通过测试，如果选择继续通过，则将此次存在一二级BUG的测试通过操作进行记录审计。

Uat测试开始





测试完成



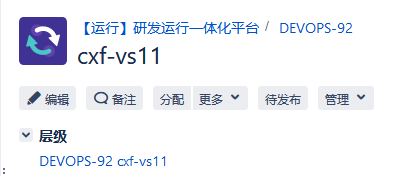


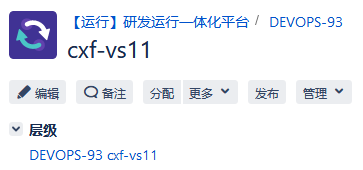
选择测试的通过与否



进入待发布

1. 进入上线申请界面，选择通过UAT测试的版本进行提交，调用OA接口，将上线申请单传入OA平台审批。
2. 上线审批节点处理逻辑
3. 当OA审批单通过后，平台对当前版本进行封版处理，处理逻辑参考“版本管理”模块上线版本发布处理逻辑。
4. 当OA审批单通过后，上线审批节点状态变为完成。
5. 当OA审批单通过后，将Jira中当前申请单的所有需求状态变为待发布。
6. 已上线节点处理逻辑
7. 当OA审批单备注上线完成后，平台已上线节点状态变为完成。



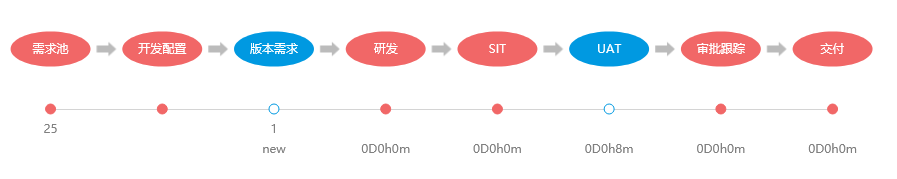


有两种发版方式





一条完整的工作流程



## 6.3版本部署

### 6.2.1版本部署的新增、编辑和删除

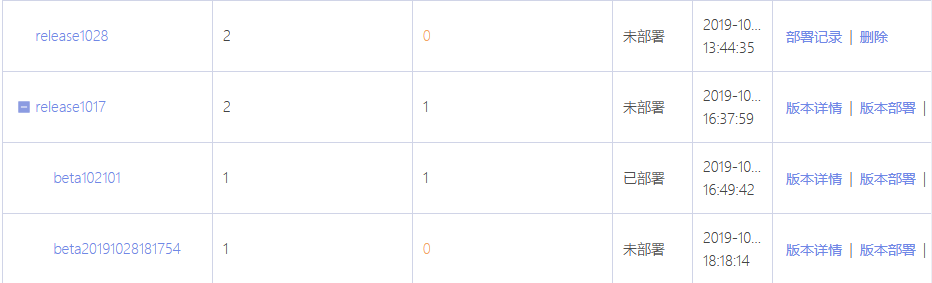
1、版本部署列表

* 展示应用系统的版本、规划需求数量、完成需求数量、部署状态、创建时间等信息。
* 列表提供版本的部署、部署记录查询、版本详情查看、删除等操作入口

2、查询

输入应用系统名称来查询应用版本部署信息，支持模糊查询





3、版本详情

版本详情会展示应用版本版本的具体信息

包括应用版本版本的需求数量、构建分支、构建日期、制品库等信息



4、部署记录

展示应用版本的部署记录，包括部署的作业编号、部署版本、部署状态、创建时间等信息。并可以查看其部署记录



1. 版本部署

版本部署需要先选择部署环境，并且可以手动选择需要部署的应用版本所依赖的服务版本，或者通过一键部署来快捷部署此应用版本



1. 删除

删除当前应用版本

# 运行管理

管理服务运行的环境等信息

### 7.1环境管理

列表包括了环境名称、部署类型、服务/任务书、cpu使用量、内存使用量等信息



下图是环境的详细信息页面，其中展示了环境包含的实例数量、服务总数、环境的物理ip列表、环境中包含的服务名称、进程号、状态等信息。用户可以启动/停止在当前环境中运行的服务



# 度量分析

度量分析是对应用的各个版本进行进度、质量和效率收集、统计和分析的功能

## 8.1进度分析

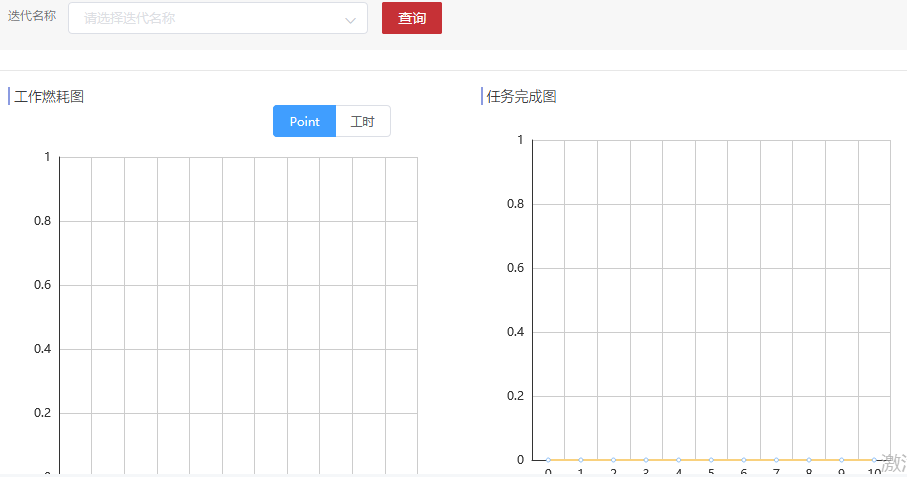
1、进度分析列表

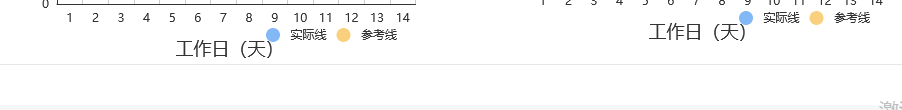
* 展示应用的各种进度信息，包括应用起止时间、story point相关、subtask相关等信息
* 输入应用版本名称来查询不同应用版本的进度信息，支持模糊查询
* 其中，起止时间是创建应用版本时所选的开发日期；story point、subtask数据来源是jira



2、工作燃耗和任务完成

可以通过输入应用版本迭代名称来查询其工作耗燃图和任务完成图





## 8.2质量分析

1、质量分析列表

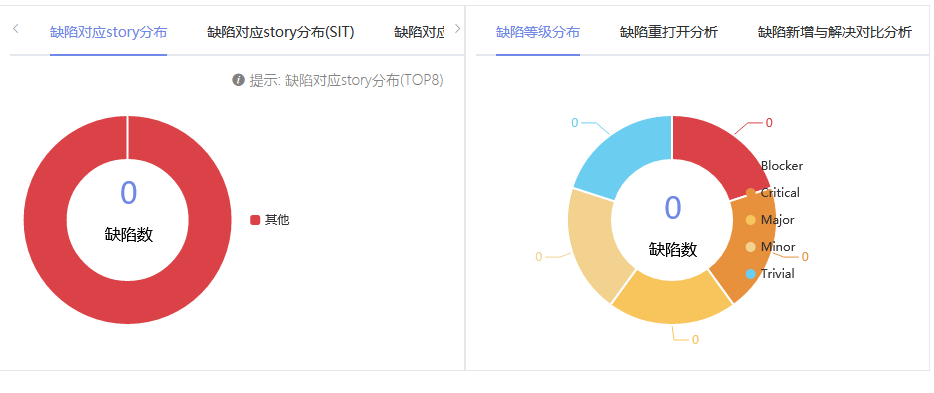
* 展示应用的各种质量信息，包括单元测试覆盖率、单元测试成功率、sonar相关、构建和部署相关等质量信息
* 输入应用版本名称来查询不同应用版本的质量信息，支持模糊查询



缺陷类型和缺陷程度



缺陷信息的图形统计

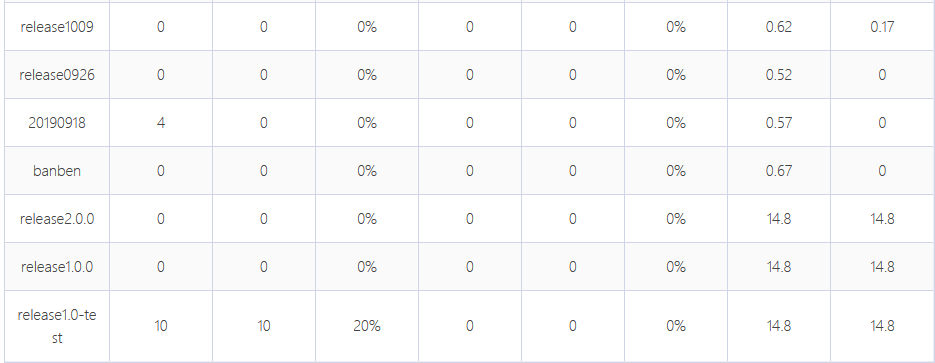


## 8.3效率分析

1、效率分析列表

* 展示应用的各种效率信息，包括应用的完成计划、实际完成数、story完成比例、subtask完成比例等信息
* 输入起止日期来进行时间的限制查询





完成信息的图标统计

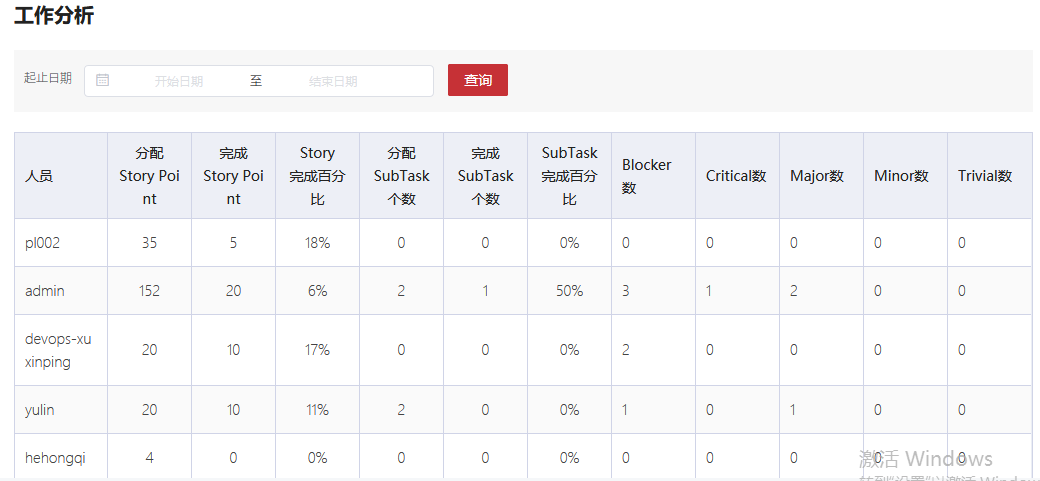


# 人员考核

## 9.1工作分析

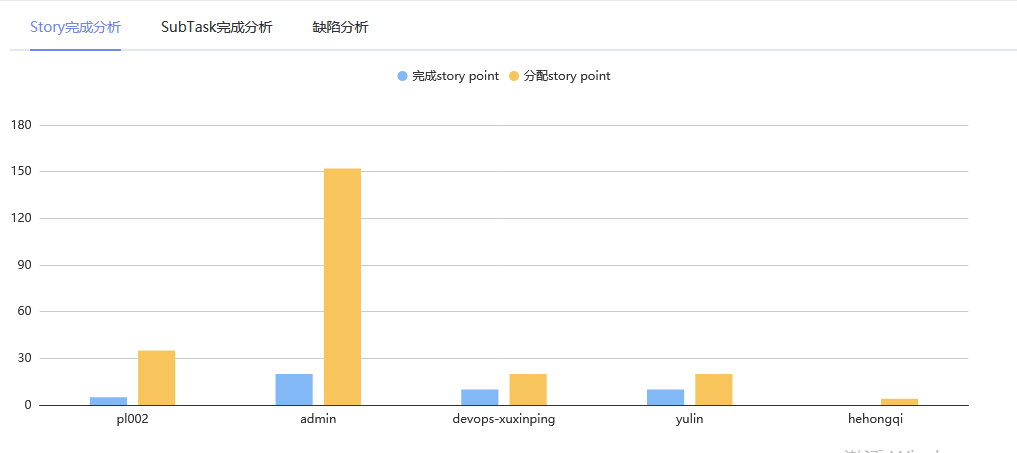
工作分析展示了各人员所分配的story point、完成的story point、所分配的subtask、完成的subtask等数据。这里的人员与jira中的经办人所对应

输入起止日期来进行时间的限制查询



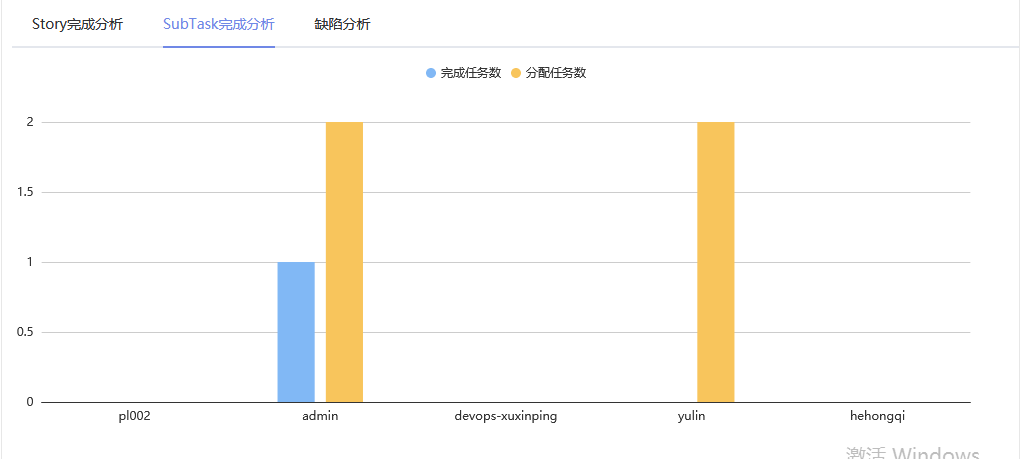
## 9.2Story完成分析

story分析展示了各经办人所完成的story和分配的story point数据统计



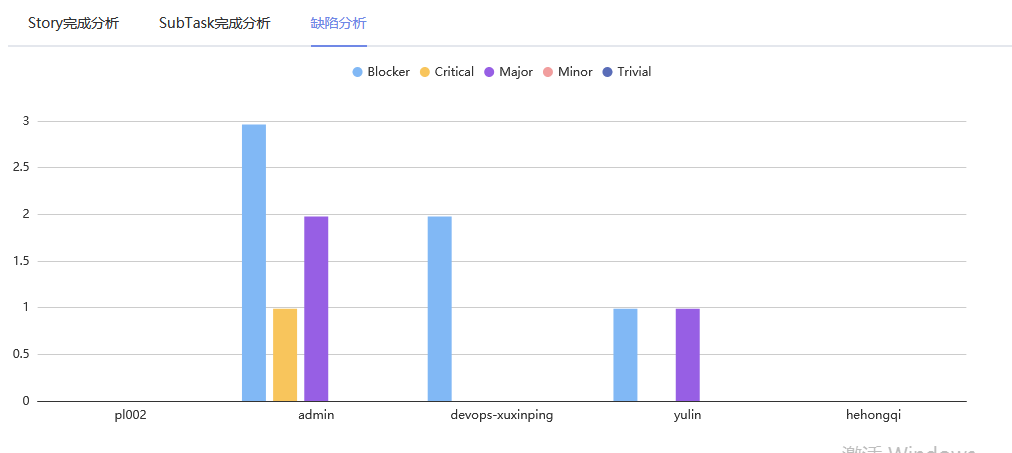
## 9.3SubTask完成分析

subtask完成分析展示了各经办人完成的任务数量、分配的任务数量的信息统计



## 9.4缺陷分析

缺陷分析展示了各经办人所拥有的问题连接类型和其数量统计信息



# 项目设置

## 10.1用户管理

### 10.1.1用户信息的查询、增加、删除、和编辑

1、用户管理列表

* 展示用户的用户名、姓名、所属组织、邮箱、手机、类型、状态等信息
* 列表提供用户信息的编辑、分配角色、重置密码、删除、冻结、导入等操作的入口



2、查询

输入用户名来查询用户的具体信息，支持模糊查询

3、导入

可以导入所选的用户，支持多用户导入



1. 新增

需要填入用户的用户名、姓名、邮箱、手机等必填信息来新增一个用户



1. 编辑

可以修改指定用户信息

1. 冻结

冻结该用户，使其无法进行操作

1. 分配角色

为指定用户分配角色，使其获取能够操作平台某些功能的权限。用户必须要有相应的权限才能对平台进行操作



1. 重置密码

重置用户的密码，重置后默认登录密码为123456。可以在登陆界面中再次修改



默认初始密码为123456，输入新密码进行修改



1. 删除和批量删除

删除指定用户/多个用户

10、用户详情

点击用户名超链接跳转至用户详情界面



在此可以看到用户的用户名、邮箱、权限等具体信息



## 10.2角色管理

### 10.2.1角色的查询、增加、删除和编辑

1、角色管理列表

* 展示角色的角色名称、角色描述、角色类型等信息
* 列表提供角色信息的编辑、详情查看入口



2、查询

输入角色名来查询角色的具体信息，支持模糊查询

3、新增

输入角色的名称，并为其选择一级和二级权限即可新增一个用户



4、批量删除

删除多个角色

5、编辑

编辑角色信息

6、详细信息

查看角色详细信息，包括其名称、权限等



角色的权限明细



## 10.3工作流配置

1、工作流

工作流为平台管理员配置的研发工作流公共模板，各应用系统在开发前配置工作流可选择应用这些工作流模板为应用系统的研发流程。目前平台默认内置两种类型的工作流，分别为：正常版本研发流程和紧急版本研发流程，如果需要定制流程，应由平台管理员定制工作流模板。

2、配置工作流列表

* 展示工作流的名称、工作流编码、状态、创建时间等信息
* 列表提供创建、编辑和删除工作流入口

如下图



3、创建工作流

一个新的应用只能有两种工作流程，即正常版本研发流程(上图public)和紧急版本研发流程(short)。如有更多需求，联系管理员



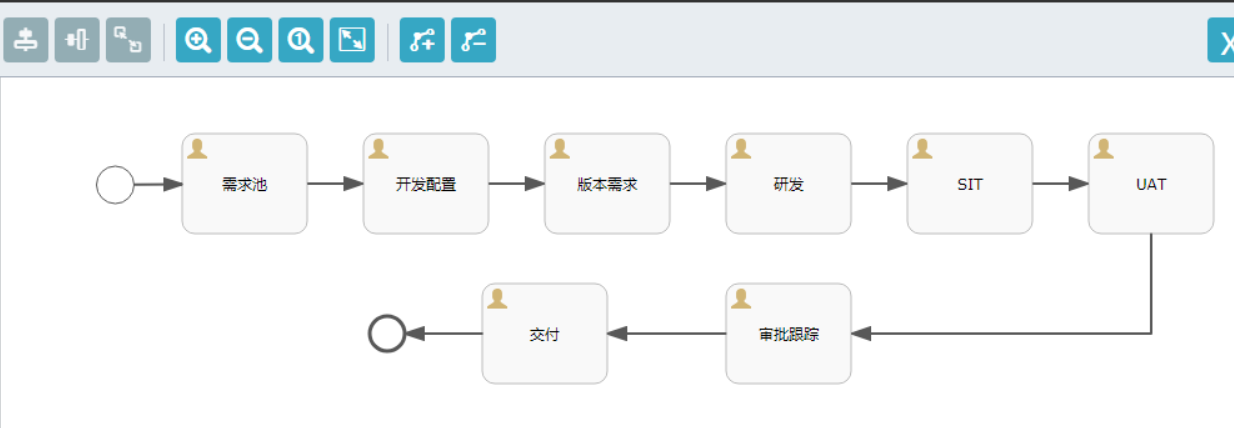
4、编辑、删除工作流

当一条工作流开始以后执行完毕之前不能编辑，删除也一样

并且工作流的删除必须与所有租户下的应用都解除关联后才可以



点击编辑跳转到工作流创建的图形绘制界面。绘制好工作流模型后，点击保存



## 10.4任务模板

任务模板是平台集成jenkins时，为了使用户能够可视化的自定义Pipeline而设计的任务模板格式，任务模板内容对应jenkins的jenkinsfile脚本，在脚本中可设置希望通过用户界面传递进来的jenkinsfile参数，参数支持用户自定义参数和系统参数两种，并支持多种参数类型，每种类型的参数输入会展示不同形式的输入方式。本功能实现对Pipeline模板的添加、编辑、删除和列表显示功能。

### 10.4.1任务模板的查询、增加、编辑和删除

1、任务模板列表

* 展示任务模板的模板名称、任务类型、模板内容、参数变量、创建时间等信息
* 列表提供任务模板信息的查看、编辑、删除入口

1. 输入任务模板名或任务类型称来查询任务模板，支持模糊查询



1. 查看模板内容、参数变量

任务模板的内容



参数变量



1. 创建任务模板

A、输入名称、类型、内容等参数信息来新建任务模板

B、任务模板参数

任务模板参数是jenkinsfile脚本的参数定义，支持自定义和系统参数类型。自定义参数需要提供参数名称、参数标识和参数类型信息。系统参数为系统定义的字段信息，默认包括：部署环境（$ENV）、服务（$SERVICE）、制品仓库（$LIBRARY）、配置中心（$CONFIGCENTER）、IP地址（$IP）、审批角色（$APPROVALROLE）、审批人（$APPROVALUSER），可根据需要进行扩展。参数类型默认包含：短文本、长文本、环境变量、脚本、系统参数。





1. 编辑

修改已创建的模板信息或者参数信息

# 操作引导

操作引导是用户对此平台操作的流程引导。此模块的功能是引导用户创建服务和应用，方便用户管理应用

## 11.1配置工作

见第十章10.3工作流配置



## 11.2创建服务

工作流配置完成点击下一步，进入服务的创建过程。

创建服务和服务管理功能相同，参考第二章服务管理



## 11.3创建系统版本

服务创建完成点击下一步，进入系统版本的创建过程。

创建系统版本和版本管理功能相同，参考第六章版本管理



## 11.4查看工作流

系统版本创建完成点击下一步，就可以查看所创建的工作流。

此处引导页和平台中概览功能、版本管理中应用系统的处理逻辑功能类似，具体见第一章平台概览、第六章6.2应用系统的处理逻辑、6.3版本部署等章节



