



ZADIG

---

释放工程师创造力



**ZADIG**

**DevOps 价值链平台**



## 领先企业抢先实践 Zadig

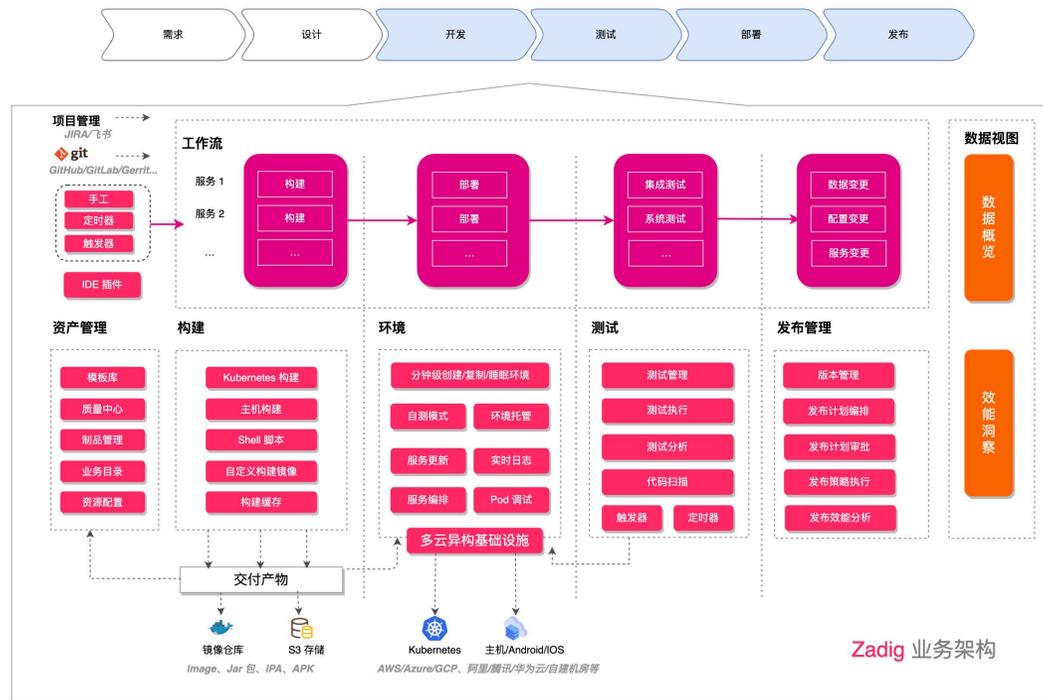
产业数字化核心资产是软件和数据：传统软件/配置/数据迭代方式已经无法适应，软件工程化时代已然到来。Zadig 软件工程平台是国内落地程度最深、使用范围最广（近千家企业）的云原生 DevOps 平台。



# Zadig 研发数字化转型方案正成为产业数字化战略的核心环节



# Zadig 设计思路：通过「平台工程」解决流程挑战，通过「技术升级」提升组织效能



01

**工程化协同**：“人、技术、流程、工具”四维协同基线，沉淀全流程数据，从感知到赋能，服务于工程师

02

**释放云基建能力**：链接任何云及自建资源（容器、主机、车机、端等），释放云原生价值和创新能力

03

**生态开放**：广泛开放系统模块和 OpenAPI，链接一切流程、服务、工具和上下游伙伴

04

**安全简单自主可控**：私有化部署，现有服务 0 迁移成本、体验丝滑接入容易、学习使用门槛极低

# 与现存 DevOps 方案对比：

现存做法大多以「单点工具 + 写脚本」或运管类平台为主，Zadig 则是面向开发者视角，中立，云原生一体化价值链平台。

现存方案	典型代表	方案特点分析	Zadig 优势
传统 Jenkins 方案	GitLab + Jenkins + 脚本化	运行效率低，管理维护成本高 方案局限性大，安全性风险高 无法支持敏捷交付模式	支持 <b>从需求到发布全流程敏捷交付</b> 。尤其面向多服务并行部署发布，云原生构建环境和运行环境，基础设施对接及企业级 SSO/权限管理等
运维管理类平台	蓝鲸 Rainbond KubeSphere KubeVela	面向资源管理的运维工具集 面向开发者，需结合 CI/CD 工具额外搭建全流程能力	<b>专门面向开发者的生产力平台</b> ，涵盖需求到开发，测试，运维的云原生一体化技术底座支撑
云厂商 DevOps 平台	华为云 DevCloud 阿里云效 腾讯 CODING	云厂商引流为主，锁定风险高 对多云跨地域支持不够 实施负担较重难以推广	面向 <b>多云友好，厂商中立</b> ，全球多地跨云跨域 安全可靠自动化部署
云原生 CI/CD 工具	Tekton Argo	使用门槛高、学习成本高 需要额外建设全流程能力	<b>接入和使用都极其简单</b> ，内置模板库和最佳实践，基于平台工程打造，可以轻松连接一切工具链
企业自建 DevOps 流程平台	围绕 Jenkins 或 CI/CD 工具搭建流程串接胶水平台	局限性大扩展性差 内部推广难度极高 做完后价值难被证明	<b>通用性、可扩展性、技术先进性</b> 强，可以灵活广泛接入各种技术和业务场景
基于代码管理的 DevOps 方案	Gitee 平台 GitLab 平台	局限性大、全流程安全性低 维护成本高	<b>支持多个服务并行构建部署、产品级发布</b> ，可灵活安全接入多个代码仓及周边工具链

# Zadig 核心特性：

真正意义的持续交付：以工程师体验为核心，价值交付为理念，完成需求到发布的全路径。

## 碎片化：手工协作 + 复杂工具链

一堆复杂脚本、维护成本极高  
员工手工操作费时费力易出错

手动更新服务、手动打包、交付效率低下、占据大量研发时间、研发利用率极低

环境不透明、测试效率低下、测试有效性低、大量手工、价值难以体现

上下游烟囱式、协作效率低、团队花大量时间在碎片化沟通和流程制定上、各方能力受限、无法快速响应市场需求

层级越高、对产研状态越模糊  
管理低效、延误战机

运维

开发

测试

发布

洞察

## 工程化：一个平台 一键发布

少量配置、快速拉起环境、稳定性有保障、减少 90% 手工操作、赋能开发、员工成就感高

workflow、环境配置自动更新、高效调试、消除手工操作、精准快速迭代、研发生产力/幸福感提升

自助运行、系统化管理、自动化程度高、测试有效性提升、质量有保障、横向赋能、技能提升

随时调用工程基线提供的能力、产品视角开发交付、团队高效协同、稳定迭代

产研数字化过程数据透明、关键指标易抽取、有能力合理调动资源、随时决策响应客户需求



产研全流程拉通需求到上线所需的代码、服务、配置和数据的一致性交付



1、

# 敏捷协作方案介绍



# 软件研发核心工程实践：基于质量工程的持续交付体系 (CI/CD、CO、CT、CS)

## 工程实践：

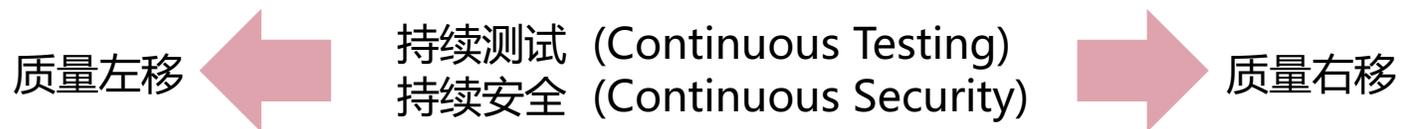
- 持续集成(CI 针对代码)
- 持续交付(CD 针对需求)
- 持续部署(CD 针对服务)
- 持续测试(CT 针对全流程)
- 持续安全(CS 针对全流程)
- 持续运营(CO 针对全流程)

## 涉及角色：

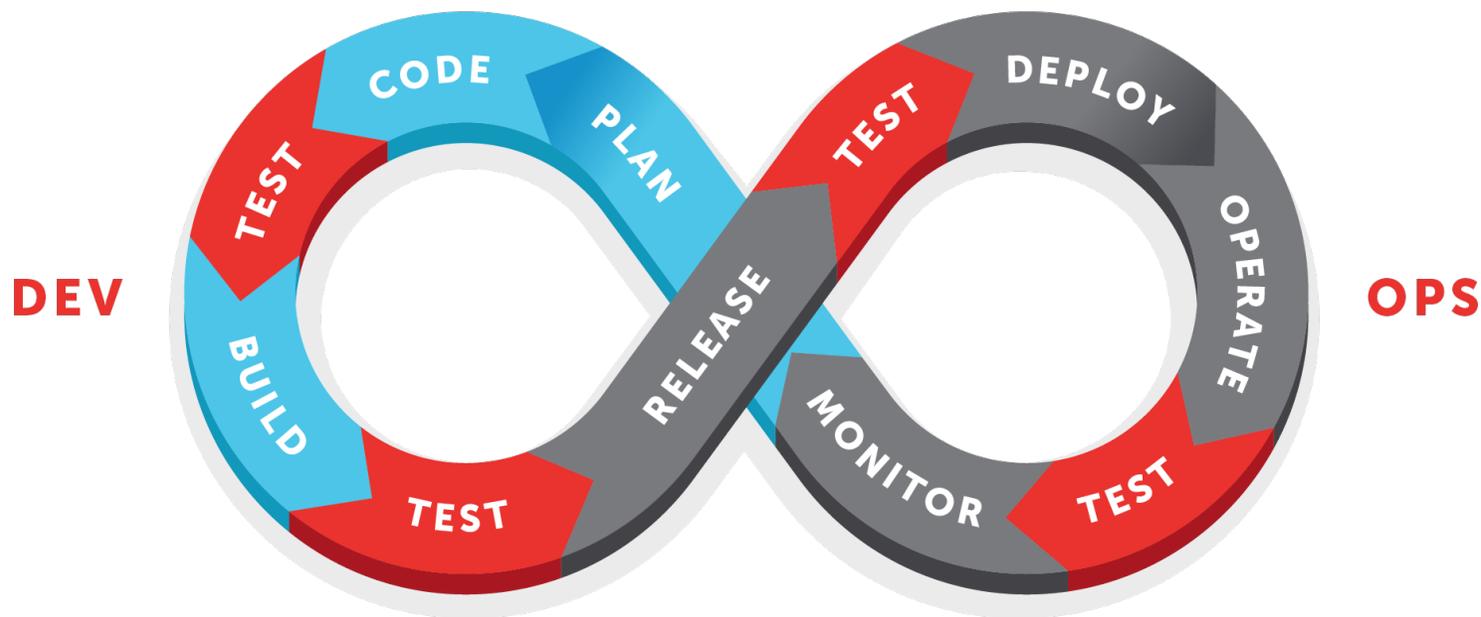
- 开发
- 测试
- 业务运维

## 平台建设方：

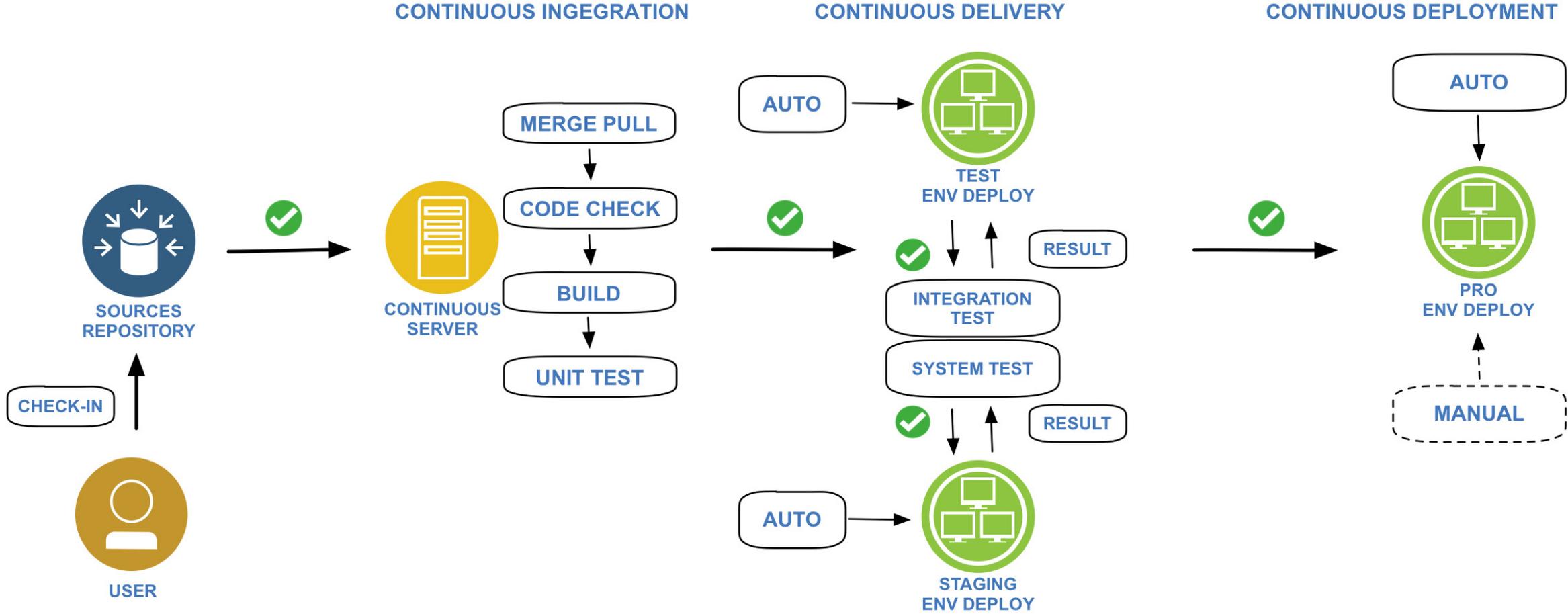
- DevOps/平台运维
- IT/基础设施
- 测试/安全团队



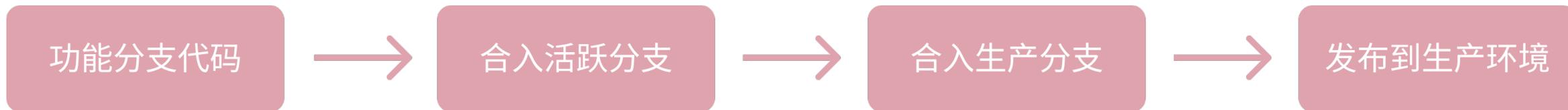
持续集成(CI) -> 持续交付(CD) -> 持续部署(CD) -> 持续运营(CO)



# 软件研发核心工程实践：交付工程 CI/CD 开发者实践



# 软件开发核心工程实践：全流程质量工程实践-持续测试CT/持续安全CS



CT/CS 质量工程

# Zadig 产研协同方案：自动化一切可以自动化的，让工程师专注创造

## 协同特点：

- 流程可定义
- 工具可扩展
- 能力可编排
- 价值可感知

## 面向角色：

- PO/TO
- 开发
- 测试
- 运维

## 服务编排：

- 测试服务
- IT 服务
- 安全服务





2、



# 用户核心场景 使用介绍

视频 Demo 演示 



**核心场景介绍**：不同角色工程师基于统一协作平面，操作使用自动化工作流和云原生环境

### 运维（管理员）

- 准备环境
- 准备 workflow

### 开发工程师

- IDE 热部署
- 更新镜像
- 更新配置
- 更新数据
- 日常调试

### 测试工程师

- 日常测试验证
- 自动化测试

### 发布工程师

- 升级预发环境
- 升级生产环境
- 变更数据库

# 管理员(运维)准备—— workflow

面向角色	功能描述	workflow名称样例	具体配置
开发工程师	CI 过程	project-unit-test project-scan	<ul style="list-style-type: none"><li>单元测试、代码扫描</li></ul>
	更新日常开发环境及 dev 业务配置	project-dev-workflow	<ul style="list-style-type: none"><li>构建、配置变更 ( Apollo/Nacos )、数据变更、部署、冒烟测试、项目管理任务变更</li></ul>
测试工程师	更新测试验证环境	project-sit-workflow	<ul style="list-style-type: none"><li>构建、配置变更 ( Apollo/Nacos )、数据变更、部署、接口测试</li></ul>
	更新集成测试环境	project-auto-sit-workflow	<ul style="list-style-type: none"><li>构建, 部署, 场景测试</li></ul>
发布工程师	更新预发环境	project-uat-workflow	<ul style="list-style-type: none"><li>质量门禁、构建、配置变更 ( Apollo/Nacos )、数据变更、部署 uat ( 变量变更、模板变更 )、全量回归测试</li></ul>
	更新生产环境	project-prod-workflow	<ul style="list-style-type: none"><li>审批、配置变更 ( Apollo/Nacos )、数据变更、部署生产环境 ( 变量变更、模板变更 )、场景测试</li></ul>

# 管理员(运维)准备—— workflow

项目 > multiservice-yaml > workflow

工作流 环境 服务 构建 测试 代码扫描 版本管理 + 新建工作流

所有 0.项目状态 1.开发专用 2.测试专用 3.发布专用 + ✎ -

<span>自定义</span> 构建 > 部署 > 冒烟测试		平均执行时间 125.7s	成功率 85.00%	最近执行 admin 04-19 16:55	<span>▶ 执行</span> <span>⚙️</span> <span>⋮</span>
<span>自定义</span> 构建 > 配置变更 > 部署		平均执行时间 12s	成功率 100%	最近执行 demo 03-30 20:08	<span>▶ 执行</span> <span>⚙️</span> <span>⋮</span>
<span>自定义</span> 构建 > DMS 数据变更 > 部署		平均执行时间 130.3s	成功率 42.86%	最近执行 admin 03-30 19:13	<span>▶ 执行</span> <span>⚙️</span> <span>⋮</span>
<span>自定义</span> 构建 > 配置变更 > 部署 > 冒烟测试		平均执行时间 241.3s	成功率 66.67%	最近执行 admin 03-30 14:06	<span>▶ 执行</span> <span>⚙️</span> <span>⋮</span>
<span>自定义</span> 构建 > mysql 数据库变更 > 部署		平均执行时间 99.5s	成功率 45.00%	最近执行 demo 02-28 21:50	<span>▶ 执行</span> <span>⚙️</span> <span>⋮</span>

## 管理员(运维)准备——环境

面向角色	环境描述	环境名称样例	场景介绍
开发工程师	用于日常开发、联调	dev	如果 Java 栈实现灰度能力，可以通过新建环境随时拉起开发自测子环境 如果采用 Istio + Skywalking 技术栈，通过 ZadiG 开启自测模式随时拉起子环境
测试工程师	用于测试手工验证	sit	自主升级环境，选择一个或多个 PR/MR 进行功能需求的并行验证。
	用于测试集成验证	auto-sit	执行自动化 CD 过程
	用于验收测试	uat	以此环境作为发布生产环境前的用户验收环节
发布/运维工程师	用于生产查看	prod	生产环境管理，变更过程需经过严格审批

# 管理员(运维)准备——环境

The screenshot shows a web interface for managing Kubernetes environments. The breadcrumb path is '项目 > multiservice-yaml > 环境'. The user is logged in as 'admin(admin)'. The main navigation bar includes ' workflows', '环境', '服务', '构建', '测试', '代码扫描', and '版本管理'. A '+ 新建环境' button is visible on the right.

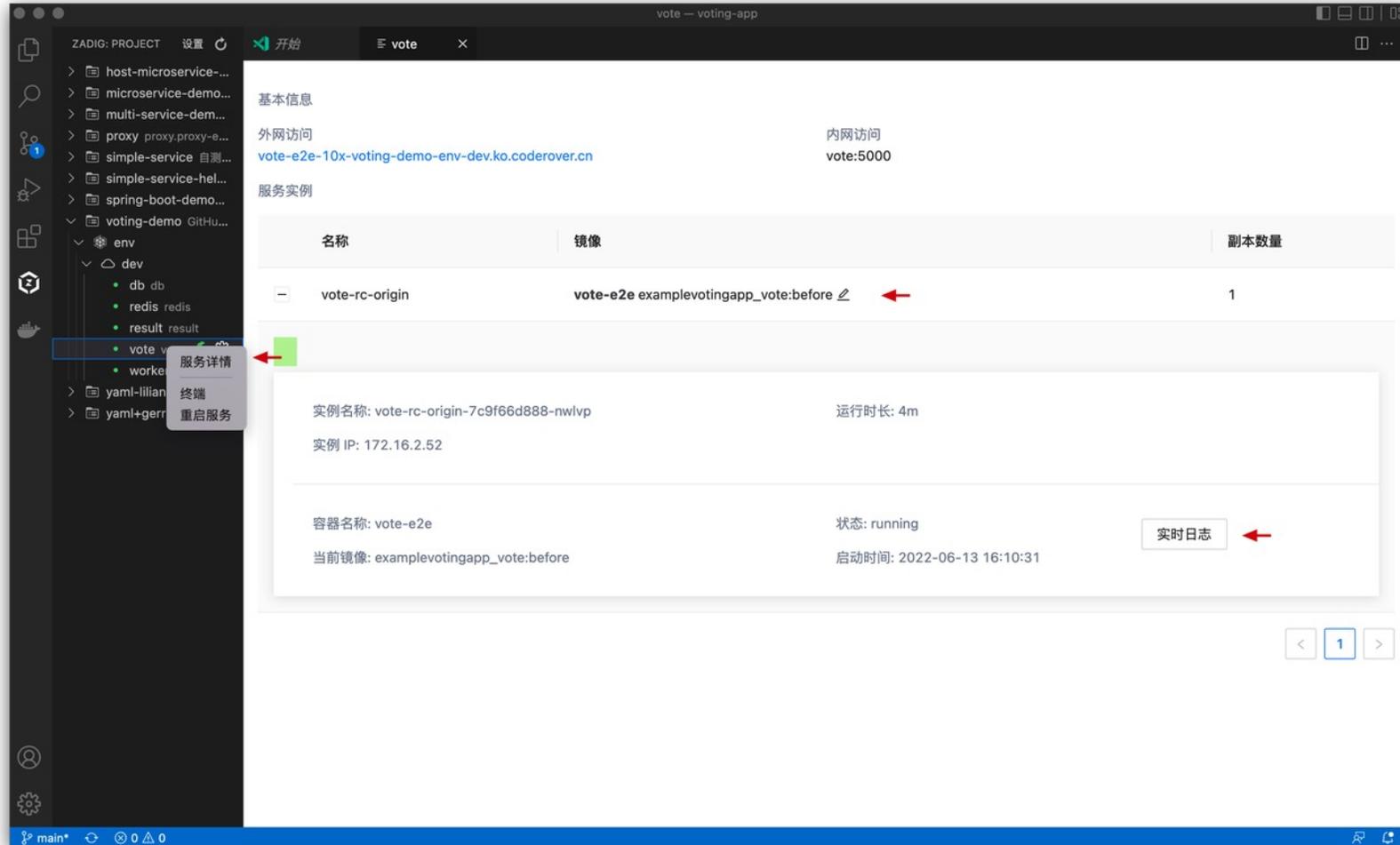
The environment selection bar shows 'dev', 'sit', 'uat', and 'prod 生产' (selected). Below this, the environment details are displayed: 'K8s 集群: 本地集群', 'K8s 命名空间: y-env-dev', and '镜像仓库: https://koderover.tencentcloudcr.com/test'.

The '工作负载' (Workloads) section is active, showing a table of Deployments. The table has columns for '名称' (Name), 'Ready', '镜像' (Image), '创建时间' (Creation Time), and '操作' (Actions). Three services are listed: 'service1', 'service3', and 'service2', all in a 'Ready' state.

名称	Ready	镜像	创建时间	操作
<a href="#">service1</a>	1/1	service1:20230106194231-1-main	2023-01-06 19:43:29	<a href="#">查看</a> <a href="#">YAML</a>
<a href="#">service3</a>	1/1	service3:20220929215820-8-feature-helm	2023-01-06 19:43:29	<a href="#">查看</a> <a href="#">YAML</a>
<a href="#">service2</a>	1/1	service2:20230104113852-2-main	2023-01-06 19:43:29	<a href="#">查看</a> <a href="#">YAML</a>

## 本地自测——IDE 热部署

1. 查看服务和环境
2. 本地代码热部署到远端



产品规划

需求开发

测试验证

变更发布

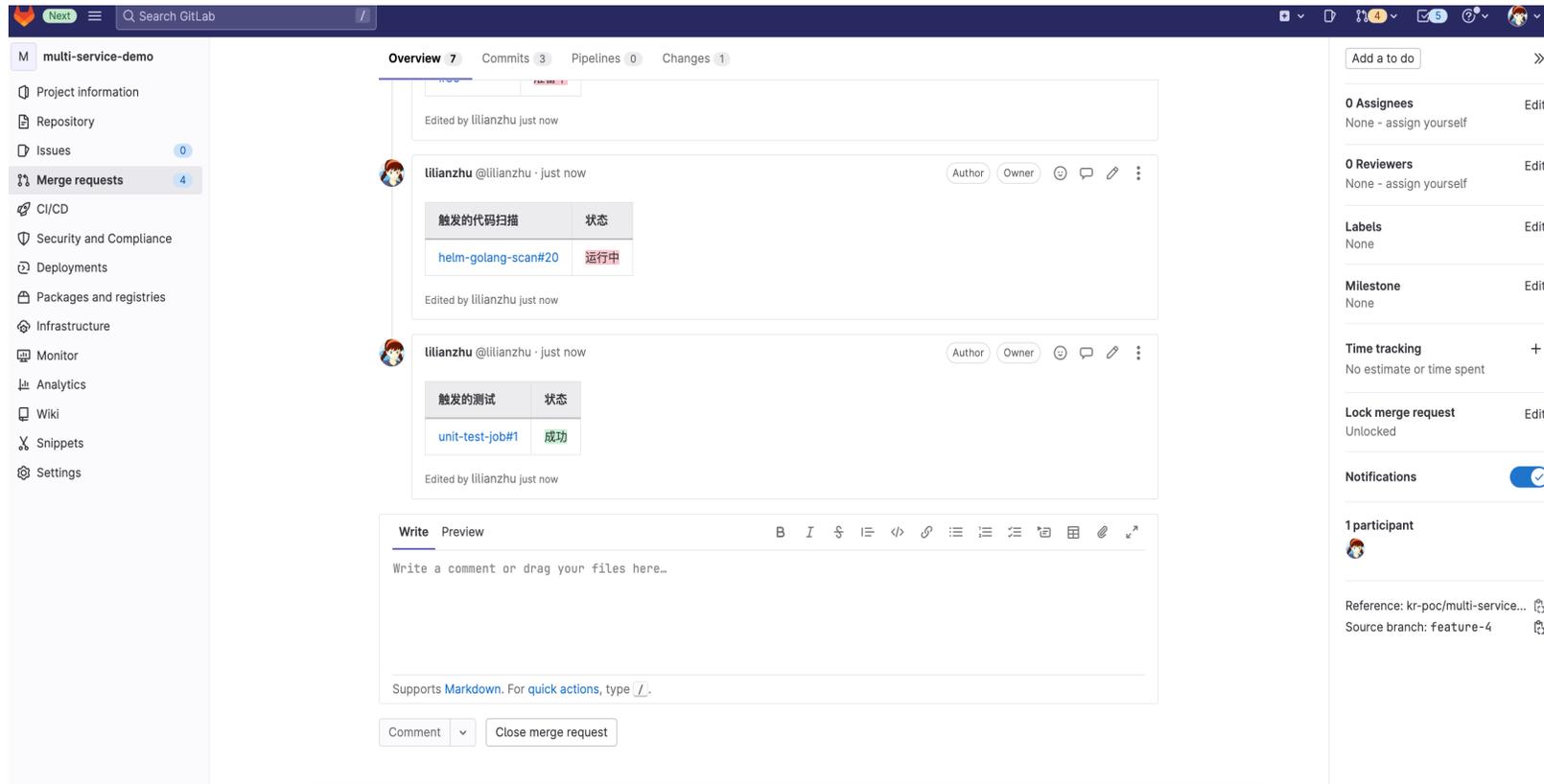
Sprint 发布

## 提交代码及 CI 过程

### 1. 本地编写代码推送到远端分支

1. 本地基于 develop 分支新建分支，在新建的分支上，编写代码
2. 推送代码到自己账号下的代码库
3. GitLab 上创建 MR

### 2. 自动触发 CI 过程，包括单元测试，代码风格检查、代码扫描。



需求开发

测试验证

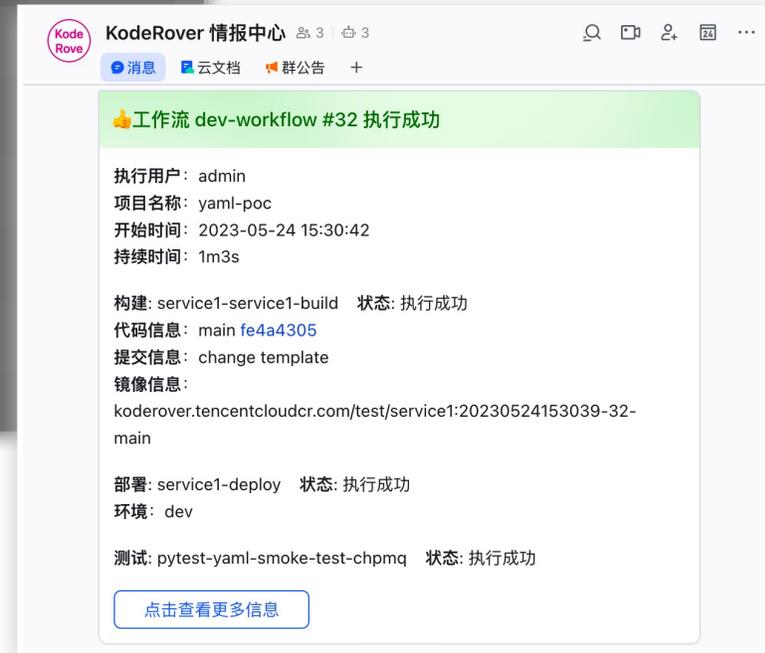
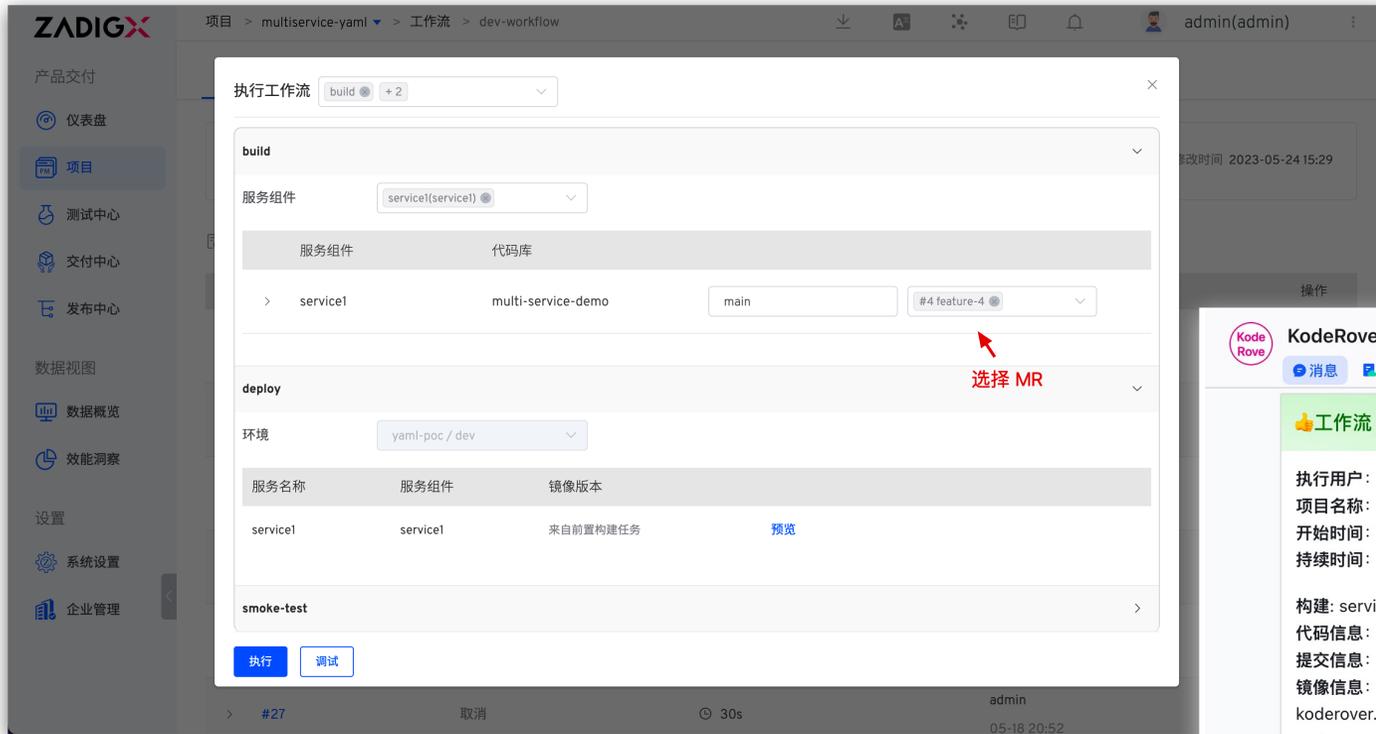
变更发布

Sprint 发布

## 单个工程师自测——手工/自动触发 dev workflows 执行

步骤包含：dev 环境构建->部署->冒烟测试 ->IM 通知

需求开发



测试验证

变更发布

Sprint 发布

## 多人做集成联调——更新不同服务

启动 dev 工作流，选择多个服务和对应的 MR 执行

产品规划

需求开发

测试验证

变更发布

Sprint 发布

The screenshot shows the ZADIG interface for configuring a workflow. The main window is titled '执行工作流' (Execute Workflow) and shows the 'build' stage configuration. A red arrow points to the '服务组件' (Service Component) dropdown menu, which is currently set to 'service1(service1)'. Below this, there is a table for configuring the build stage:

服务组件	代码库	分支	MR
> service1	multi-service-demo	main	#4 feature-4
> service2	multi-service-demo	main	#3 feature-3

Below the build stage, the 'deploy' stage is configured for the 'yaml-poc / dev' environment. It shows a table for service deployment:

服务名称	服务组件	镜像版本	操作
service1	service1	来自前置构建任务	预览
service2	service2	来自前置构建任务	预览

The interface also shows a sidebar with navigation options like '仪表盘', '项目', '测试中心', '交付中心', '发布中心', '数据视图', '效能洞察', '系统设置', and '企业管理'. The top right corner shows the user 'admin(admin)' and the current time '2023-05-24 15:29'.

## 多人做集成联调——更新同一个服务

启动 dev 工作流，选择多个服务以及其对应的多个 MR 执行

需求开发

产品规划

The screenshot shows the ZADIG interface with a modal window for configuring a workflow. The modal is titled '执行工作流' (Execute Workflow) and shows the 'build' step configuration. The 'service1' component is selected, and the 'multi-service-demo' code repository is chosen. A dropdown menu for MRs is open, showing options for '#4 feature-4', '#3 feature-3', and '#2 feature-2', with a red arrow pointing to the '#2 feature-2' option. The 'deploy' step is also visible, showing the 'yml-poc / dev' environment and a table of service configurations.

服务名称	服务组件	镜像版本	
service1	service1	来自前置构建任务	预览

Buttons at the bottom of the modal: 执行 (Execute), 调试 (Debug).

测试验证

变更发布

Sprint 发布

## 更新业务配置 (以 Nacos 为例)

执行对应环境的工作流，选择配置并按需修改，更新 Nacos 配置

需求开发

The screenshot displays the ZADIG interface for updating Nacos configurations. The main panel is titled "执行工作流" (Execute Workflow) and shows the "build" step. Under the "nacos" section, the "命名空间" (Namespace) is set to "dev" and the "配置选择" (Configuration Selection) is "DEFAULT\_GROUP/test1". The "配置内容" (Configuration Content) section shows a "YAML 变更对比" (YAML Change Comparison) view. The left pane shows the original configuration: 

```
1 a: ZadigX
2 c: ZadigX
```

. The right pane shows the updated configuration: 

```
a: b
c: o
a: ZadigX
c: ZadigX
```

. A red arrow points to the "变更对比" (Change Comparison) header. Below the configuration content, the "deploy" step is visible, with the "环境" (Environment) set to "yaml-poc / dev". On the right side of the interface, there is a sidebar with a user profile "admin(admin)", a "操作" (Operations) section, and a list of "克隆" (Clone) buttons.

测试验证

变更发布

Sprint 发布

## 更新项目管理任务状态

执行对应环境的工作流，选择对应任务

需求开发

The screenshot displays the ZADIG interface for managing tasks. On the left is a sidebar with navigation options: 产品交付, 仪表盘, 项目, 测试中心, 交付中心, 发布中心, 数据视图, 数据概览, 效能洞察, 设置, 系统设置, 企业管理. The main content area is divided into sections for different tasks:

- Service Component:** A dropdown menu shows 'service1(service1)'. Below it is a table with columns '服务组件' and '代码库'. A row for 'service1' shows 'multi-service-demo' and 'main', with a '请选择 PR' dropdown.
- deploy:** An environment dropdown shows 'yaml-poc / dev'. Below it is a table with columns '服务名称', '服务组件', and '镜像版本'. A row for 'service1' shows 'service1' and '来自前置构建任务', with a '预览' link.
- test:** A table with columns '测试名称' and '代码库'. A row for 'pytest-yaml' shows 'num\_test' and 'master', with a '请填写 PR 号' input field.
- jira-update:** A section for '变更的问题' with a list of items: 'MIN-45 服务变量易用性优化' and 'MIN-43 支持全局服务变量'. A red arrow points to the second item.

On the right side, there is a user profile for 'admin(admin)' with a '修改时间 2023-05-15 11:45' and a '操作' menu containing '克隆' buttons.

测试验证

变更发布

Sprint 发布

## 更新数据库

执行对应环境的工作流，输入 SQL 语句执行数据更新

需求开发

The screenshot displays the ZADIG interface for configuring a workflow. On the left, a sidebar contains navigation options: 仪表盘, 项目, 测试中心, 交付中心, 发布中心, 数据视图, 数据概览, 效能洞察, 设置, 系统设置, and 企业管理. The main area shows a workflow configuration for 'service1' in the 'multi-service-demo' code repository, specifically for the 'main' branch. The 'mysql' section is expanded, showing a 'query' field with the SQL statement 'update t1 set id=1 where typ=1;'. Below this, the 'deploy' section is expanded, showing a table of services:

服务名称	服务组件	镜像版本	
service1	service1	来自前置构建任务	预览

At the bottom of the workflow configuration, there are buttons for '执行' (Execute) and '调试' (Debug). The status bar at the bottom indicates a successful execution: '#15 成功 1m5s admin 02-03 14:00'. On the right side of the interface, there is a sidebar with a '操作' (Operations) section containing '克隆' (Clone) buttons for each service.

测试验证

变更发布

Sprint 发布

## 服务的调试——查看环境和服务状态

需求开发

项目 > multiservice-yaml > 环境 > dev > service2

工作流 环境 服务 构建 测试 代码扫描 版本管理

仪表盘 项目 测试中心 交付中心 发布中心 数据视图 数据概览 效能洞察 设置 系统设置 企业管理

基本信息

外网访问 无 内网访问 service2:20222

基本操作

配置管理 Yaml 导出 服务管理

服务实例 刷新

名称	镜像	副本数量	操作	详情
service2	service2 service2:20230519143709-30-main	1	重启实例 查看事件	

实例名称: service2-75c978cfd-jxbn5 运行时长: 5d 启动调试容器 alpha 重启 查看事件

实例 IP: 10.200.7.56 节点信息: 172.16.4.202(172.16.4.202)

健康探测: ready

容器名称: istio-proxy 状态: running 调试 实时日志

当前镜像: proxyv2:1.13.2 启动时间: 2023-05-19 14:37:27

容器名称: service2 状态: running 调试 实时日志

当前镜像: service2:20230519143709-30-main 启动时间: 2023-05-19 14:37:26

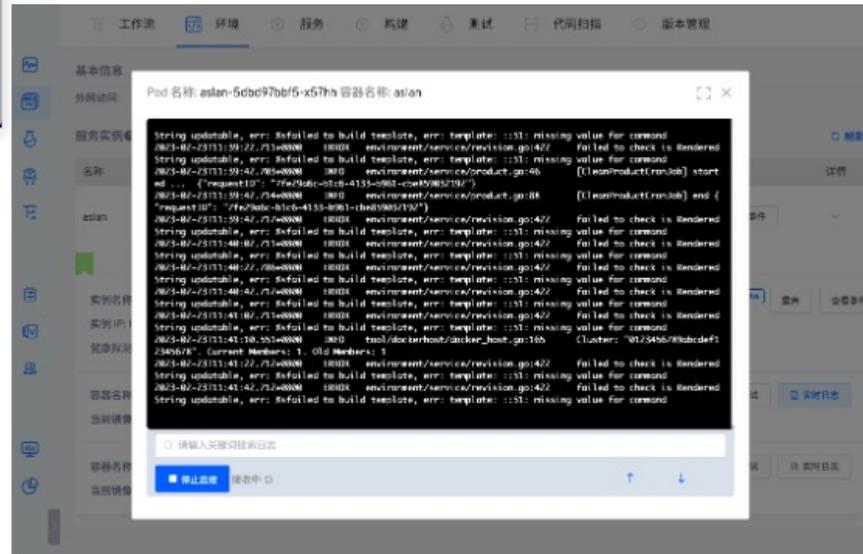
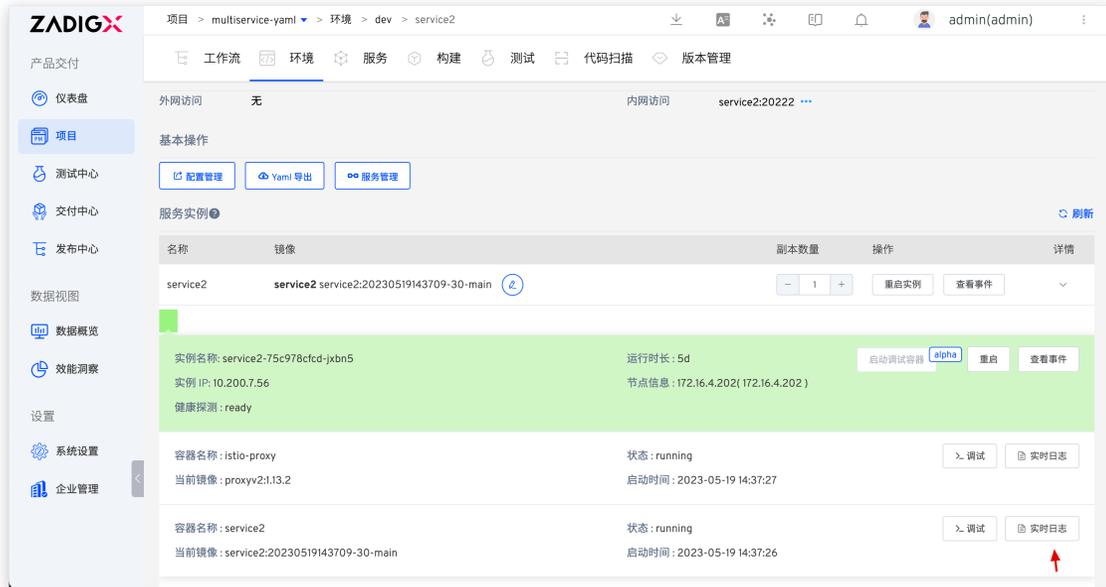
测试验证

变更发布

Sprint 发布

## 服务的调试——查看服务实时日志

需求开发



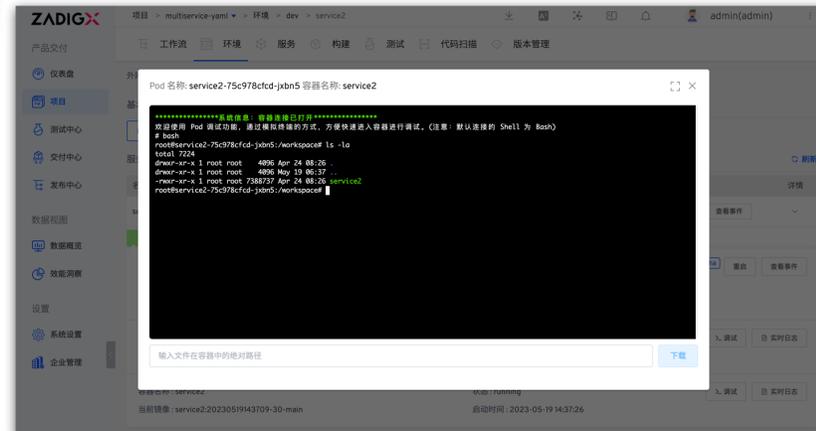
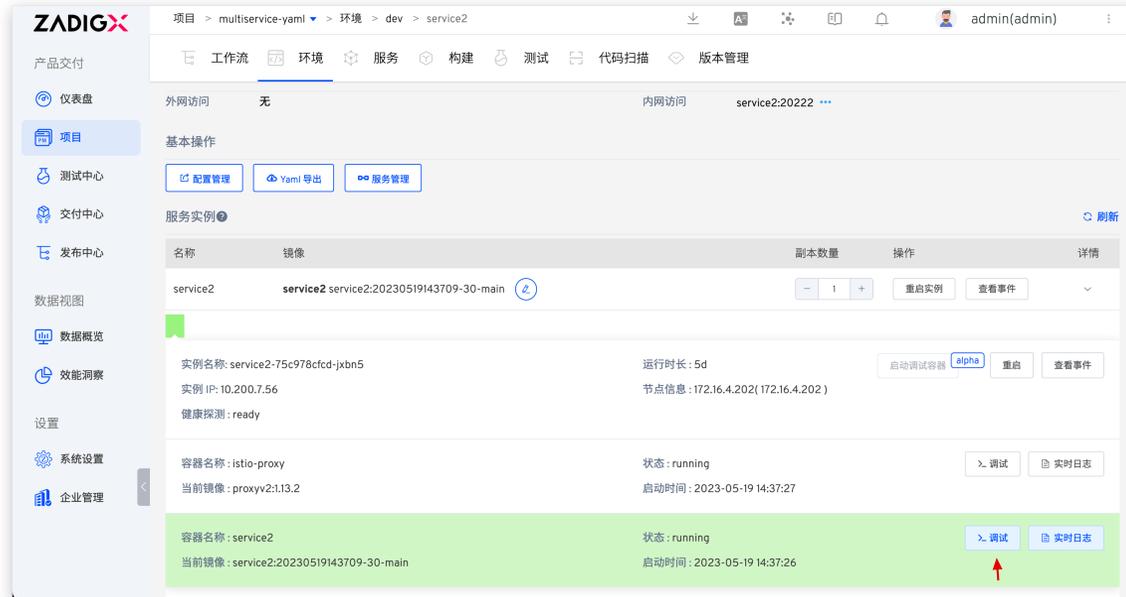
测试验证

变更发布

Sprint 发布

## 服务的调试——进入容器调试

需求开发



测试验证

变更发布

Sprint 发布

The screenshot displays the ZADIG web interface for managing a service named 'service2'. The breadcrumb path is '项目 > multiservice-yaml > 环境 > dev > service2'. The user is logged in as 'admin(admin)'. The interface includes a sidebar with navigation options like '仪表盘', '项目', '测试中心', '交付中心', '发布中心', '数据视图', '效能洞察', '设置', and '企业管理'. The main content area shows the '环境' (Environment) tab with '工作流', '环境', '服务', '构建', '测试', '代码扫描', and '版本管理' sub-tabs. Under '服务' (Service), there are buttons for '配置管理', 'Yaml 导出', and '服务管理'. The '服务实例' (Service Instances) section shows a table with columns for '名称', '镜像', '副本数量', '操作', and '详情'. A red arrow points to a '临时替换' (Temporary Replace) icon in the '操作' column for the 'service2' instance. Below the table, there are details for the instance, including '实例名称', '实例 IP', '健康探测', '容器名称', and '当前镜像'. There are also buttons for '启动调试容器', 'alpha', '重启', '查看事件', '调试', and '实时日志'.

项目 > multiservice-yaml > 环境 > dev > service2

admin(admin)

工作流 环境 服务 构建 测试 代码扫描 版本管理

基本信息

外网访问 无 内网访问 service2:20222 ...

基本操作

配置管理 Yaml 导出 服务管理

服务实例 刷新

名称	镜像	副本数量	操作	详情
service2	service2 service2:20230519143709-30-main	1	临时替换 重启实例 查看事件	

实例名称: service2-75c978cfd-jxbn5 运行时长: 5d 启动调试容器 alpha 重启 查看事件

实例 IP: 10.200.7.56 节点信息: 172.16.4.202(172.16.4.202)

健康探测: ready

容器名称: istio-proxy 状态: running 调试 实时日志

当前镜像: proxyv2:1.13.2 启动时间: 2023-05-19 14:37:27

容器名称: service2 状态: running 调试 实时日志

当前镜像: service2:20230519143709-30-main 启动时间: 2023-05-19 14:37:26

## 服务的调试——调整副本数量/重启实例

需求开发

The screenshot displays the ZADIG web interface for managing a service instance. The breadcrumb path is '项目 > multiservice-yaml > 环境 > dev > service2'. The user is logged in as 'admin(admin)'. The interface includes a sidebar with navigation options like '仪表盘', '项目', '测试中心', '交付中心', '发布中心', '数据视图', '效能洞察', '设置', and '企业管理'. The main content area shows the '服务实例' (Service Instance) management page for 'service2'. It features a table with columns for '名称', '镜像', '副本数量', '操作', and '详情'. The table lists the instance 'service2' with a current replica count of 1. Below the table, there are detailed instance information including '实例名称', '实例 IP', '健康探测', '容器名称', and '当前镜像'. The '操作' column contains buttons for '重启实例', '查看事件', '启动调试容器', '重启', '查看事件', '调试', and '实时日志'. Two red arrows point to the replica count controls and the '重启实例' button.

名称	镜像	副本数量	操作	详情
service2	service2 service2:20230519143709-30-main	- 1 +	重启实例 查看事件	▼

实例名称: service2-75c978cfd-jxbn5  
实例 IP: 10.200.7.56  
健康探测: ready  
运行时长: 5d  
节点信息: 172.16.4.202(172.16.4.202)  
容器名称: istio-proxy  
当前镜像: proxyv2:1.13.2  
状态: running  
启动时间: 2023-05-19 14:37:27  
容器名称: service2  
当前镜像: service2:20230519143709-30-main  
状态: running  
启动时间: 2023-05-19 14:37:26

测试验证

变更发布

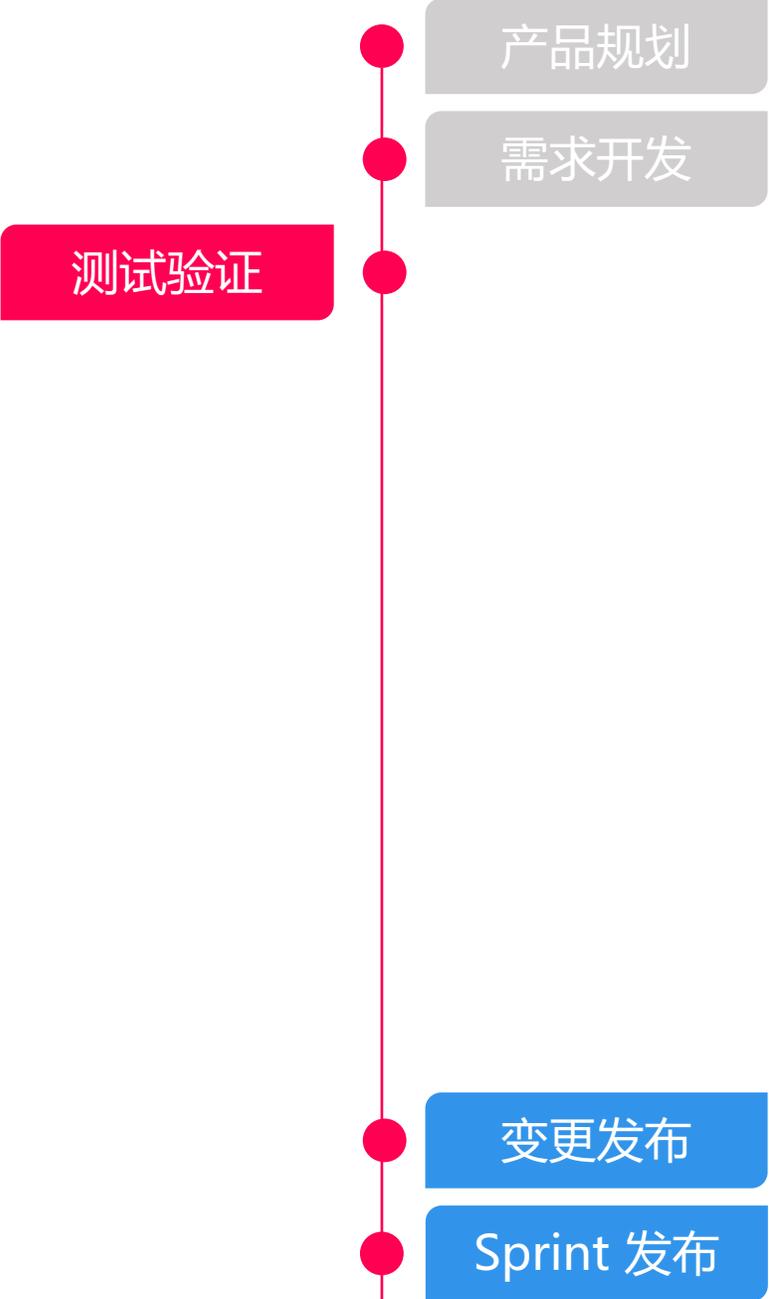
Sprint 发布



## 测试集管理——GitLab 仓库管理

支持冒烟测试/接口测试/UI 测试/场景测试/性能测试等

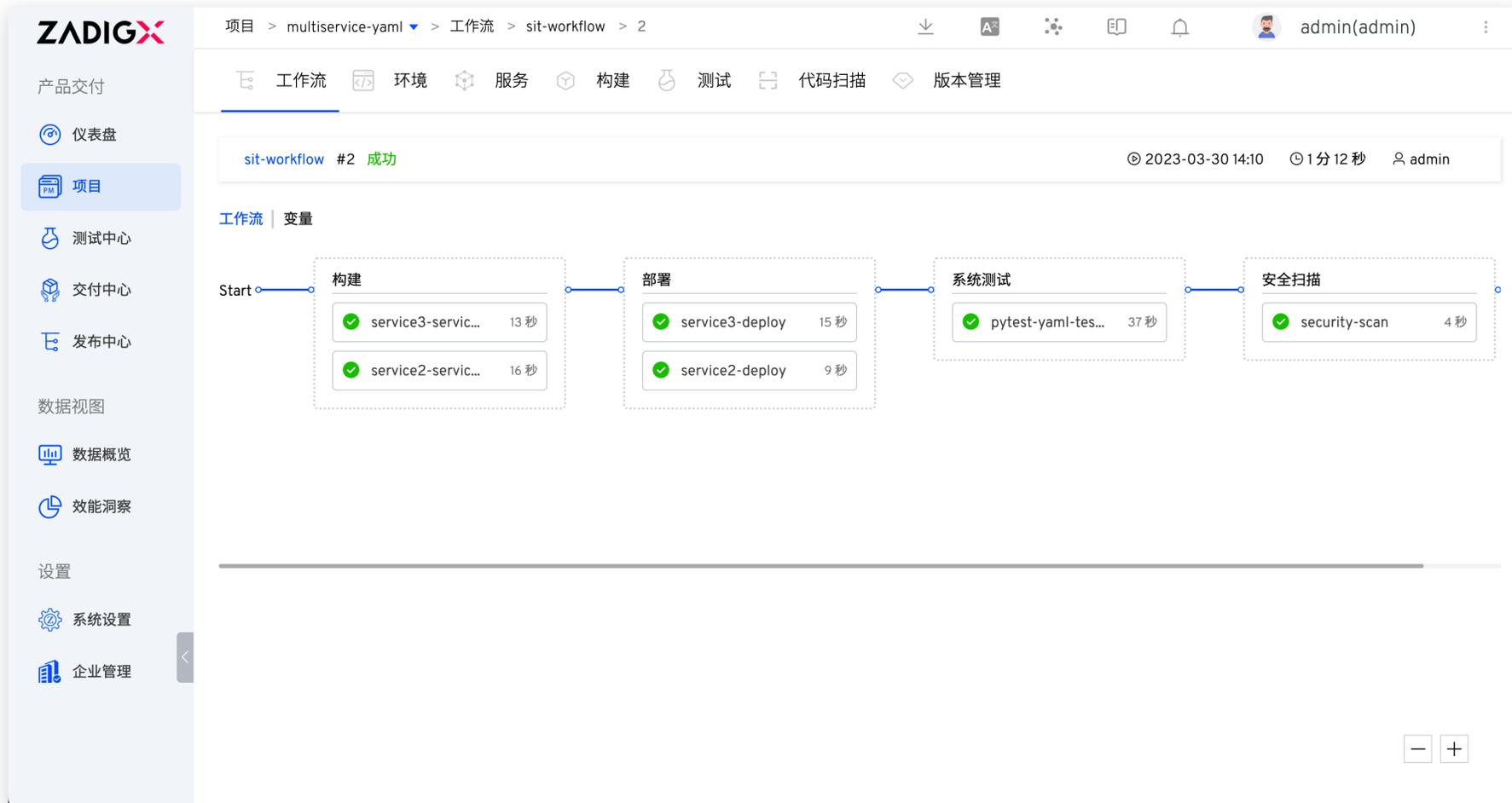
1. 本地编写测试脚本并针对 sit 环境本地自测
2. 没问题后提交到 GitLab 仓库



## 自动化 CD 过程：合并代码后自动触发更新集成环境

包括步骤：auto-sit 环境构建->部署->场景测试->安全扫描 ->IM 通知

测试验证



产品规划

需求开发

变更发布

Sprint 发布

## sit 发布——执行 sit workflows更新环境进行集成验证

包括步骤：构建 -> 部署 sit 环境 -> 接口测试 -> IM 通知

测试验证

产品规划

需求开发

The screenshot displays the ZADIGX web interface with a sidebar on the left and a main content area. The sidebar includes navigation items: 产品交付 (Product Delivery), 仪表盘 (Dashboard), 项目 (Projects), 测试中心 (Testing Center), 交付中心 (Delivery Center), 发布中心 (Release Center), 数据视图 (Data Views), 数据概览 (Data Overview), 效能洞察 (Performance Insights), 设置 (Settings), 系统设置 (System Settings), and 企业管理 (Enterprise Management). The main content area is divided into four sections: build, deploy, api-test, and security-scan. The build section shows a dropdown for 'service1(service1)' and a table with columns '服务组件' (Service Component) and '代码库' (Code Repository). The deploy section shows a dropdown for 'yaml-poc / sit' and a table with columns '服务名称' (Service Name), '服务组件' (Service Component), and '镜像版本' (Image Version). The api-test section shows a dropdown for 'pytest-yaml' and a table with columns '测试名称' (Test Name) and '代码库' (Code Repository). The security-scan section is currently empty. On the right side of the interface, there is a user profile for 'admin(admin)' and a list of actions including '操作' (Action), '克隆' (Clone), and '修改时间 2023-04-24 16:14' (Modified Time 2023-04-24 16:14).

变更发布

Sprint 发布

The screenshot displays the ZADIG interface for a workflow run. The top navigation bar shows the project path: 项目 > multiservice-yaml > 工作流 > sit-workflow > 13. The main content area shows a workflow diagram with four stages: 构建 (37s), 部署 (10s), 接口测试 (28s), and 安全扫描 (4s). The '接口测试' stage is highlighted, showing a successful result for 'api-test pytest-yaml'. Below the workflow, a table provides details for the test run:

项目名称	yaml-poc	测试名称	pytest-yaml
代码库(gitlab)	num_test	代码信息	master 2365575312
JUnit 测试报告	<a href="#">查看</a>	文件导出	<a href="#">下载</a>
Html 测试报告	<a href="#">查看</a>		

The '查看' link for the HTML test report is highlighted with a red arrow. Below the table, a terminal log shows the execution details for the 'job-executor Start' stage, including tool installation, git clone, and checkout operations.

## uat 发布——执行 uat workflow 做预发布验证

步骤包含：质量门禁->构建->nacos 变更->部署 uat 环境-> 回归测试->IM通知

测试验证

The screenshot displays the ZADIGX interface for configuring a workflow. The left sidebar contains navigation options: 产品交付, 仪表盘, 项目, 测试中心, 交付中心, 发布中心, 数据视图, 数据概览, 效能洞察, 设置, 系统设置, and 企业管理. The main panel shows a workflow configuration for 'code-scan' with a 'build' step and an 'update-nacos-config' step. The 'build' step includes a table of service components:

服务组件	代码库	分支	操作
service1	multi-service-demo	main	请选择 PR
service2	multi-service-demo	main	请选择 PR

The 'update-nacos-config' step is configured with '命名空间' set to 'dev' and '配置选择' set to 'DEFAULT\_GROUP/test1'. Below, the '配置内容' section shows a diff view for 'DEFAULT\_GROUP/test1' with the following changes:

```
1 a: ZadigX
2 c: ZadigX
```

The diff view shows a comparison between the current state and a previous state, with changes highlighted in red and green.

产品规划

需求开发

变更发布

Sprint 发布

## 生产环境发布——滚动发布，执行 prod workflow更新生产环境

步骤包含：发布审批-> nacos 变更-> 部署

产品规划

需求开发

测试验证

变更发布

项目 > multiservice-yaml > workflow > multiservice-prod-workflow > 10

工作流 | 变量

Start —> 人工审批 —> nacos 配置变更 —> 人工审批 —> 部署 —> End

人工审批 开始时间: 2023-05-24 16:13 4分58秒后审批超时

审批人	审核结果	审核时间	评论信息
admin	待审批	*	
demo	待审批	*	

审批

Sprint 发布

## 生产环境发布——蓝绿发布，执行 workflow 更新生产环境

步骤包含：部署蓝环境 -> 审批 -> 切换生产版本

产品规划

需求开发

测试验证

变更发布

项目 > 发布中心 > workflow > blue-green-deploy > 13

blue-green-deploy #13 成功 © 2023-02-01 11:23 39秒 admin

Workflow: Start -> 部署蓝环境 (5s) -> 检查蓝环境 (3s) -> 人工审批 -> 切换新版本 (4s) -> End

时间	类型	信息
2023-02-01 11:24:20	info	point service: reviews to deployment: reviews-v1675221823 success
2023-02-01 11:24:24	info	blue green deployment succeed, now service point to deployem: reviews-v1675221823

Sprint 发布



## 生产环境发布——金丝雀发布 执行 workflow 更新生产环境

步骤包含：部署金丝雀 -> 随机测试 -> 审批 -> 新版本全量发布

产品规划

需求开发

测试验证

变更发布

金丝雀发布 rolling-update-reviews 成功 4 秒

K8s service 名称	reviews	容器名称	reviews
集群	local-20221012195848	命名空间	bookinfo-canary-poc
镜像名称	reviews:v2		

时间	类型	信息
2023-01-10 15:06:49	info	canary deployment: reviews-zadiq-canary deleted
2023-01-10 15:06:49	info	updating deployment: reviews image
2023-01-10 15:06:51	info	deployment: reviews image updated successfully

Sprint 发布

## 生产环境发布——分批次灰度发布 执行 workflow 更新生产环境

步骤包含：灰度 20% -> 审批 -> 灰度 60% -> 审批 -> 新版本全量发布

产品规划

需求开发

测试验证

变更发布

项目 > 发布中心 > workflow > gray-deploy > 17

gray-deploy #17 成功 © 2023-01-18 15:25 1分6秒 demo

Workflow: Start -> gray-release-20 (first-gray-job-... 4秒) -> 人工审批 -> gray-release-60 (second-gray-job-... 4秒) -> 人工审批 -> gray-release-100 (last-gray-job-r... 7秒) -> End

灰度发布	last-gray-job-reviews	成功	7秒	
容器名称	Deployment/reviews		灰度百分比	100
集群	local-20221012195848		命名空间	bookinfo-gray-poc
镜像名称	reviews:v1			
时间	类型	信息		
2023-01-18 15:26:55	info	deployment: reviews replica set to 1		
2023-01-18 15:26:55	info	deployment: reviews image set to registry.cn-hangzhou.aliyuncs.com/minmin2/reviews:v1		
2023-01-18 15:26:58	info	gray release deployment: reviews-zadig-gray was deleted		

Sprint 发布

## 生产环境发布——Istio 发布， 执行 workflows 更新生产环境

步骤包含：部署新版本 20% 流量导入新版本 -> 审批 -> 100% 流量导入新版本

产品规划

需求开发

测试验证

变更发布

项目 > 发布中心 > 工作流 > istio release > 6

istio release #6 成功 © 2023-01-10 15:20 ⌚ 57 秒 👤 admin

工作流 | 变量

Start — istio-1 — 人工审批 — istio-2 — End

istio-1-reviews-v2 4 秒

istio-2-reviews-v2 6 秒

Istio 发布 istio-2-reviews-v2 成功 6 秒

容器名称	reviews-v2/reviews	集群	local-20221012195848
命名空间	min-istio	镜像名称	reviews:v2
新版本副本数百分比	100%	新版本流量百分比	100%

时间	类型	信息
2023-01-10 15:21:29	info	Modifying Virtual Service: bookinfo
2023-01-10 15:21:29	info	updating the original workload reviews-v2 with the new image: registry.cn-hanqzhou.aliyuncs.com/minmin2/reviews:v2
2023-01-10 15:21:29	info	Waiting for deployment: reviews-v2 to start
2023-01-10 15:21:35	info	switching the queries back to the original workload on virtual service: bookinfo
2023-01-10 15:21:35	info	deleting the destination rule created by zadig: reviews-v2-zadig
2023-01-10 15:21:35	info	Deleting the temporary deployment created by zadig: reviews-v2-zadig-copy

Sprint 发布

## uat 发布——版本管理

生成版本信息，包括：服务镜像，服务配置，服务启动顺序等等

产品规划

需求开发

测试验证

变更发布

Sprint 发布

The screenshot displays the ZADIG version management interface for a project named 'multiservice-yaml'. The breadcrumb path is '项目 > multiservice-yaml > 版本管理 > v3.0.0'. The user is logged in as 'admin(admin)'. The interface is divided into several sections:

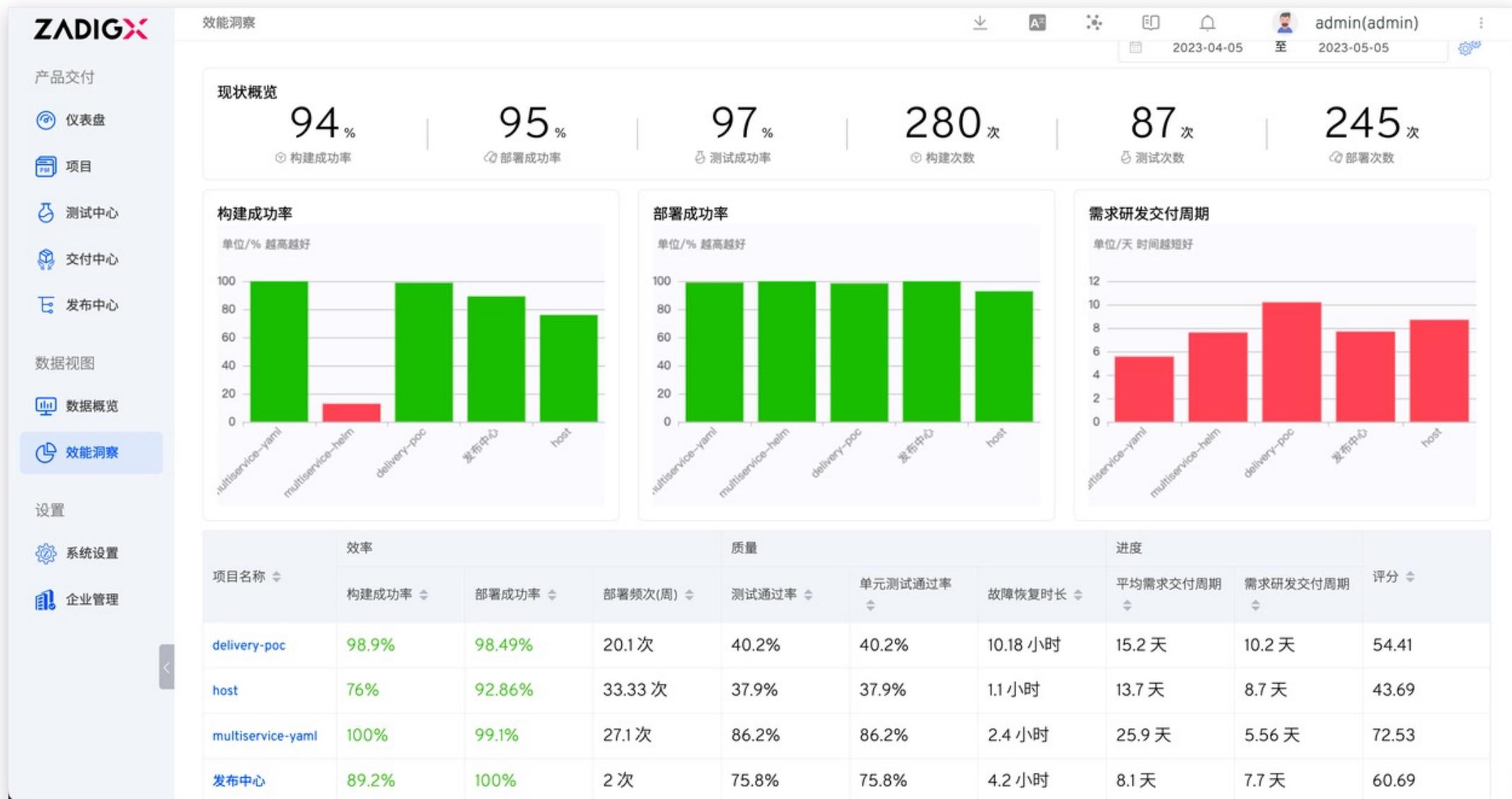
- 版本信息 (Version Information):** A table showing basic information for version v3.0.0.

属性	值	属性	值
版本	v3.0.0	所属项目	yaml-poc
workflow	xxx#3	描述	账号系统升级
创建人	admin	创建时间	2023-05-24 17:56
- 交付内容 (Deliverables):** A section titled '镜像和配置信息' (Image and Configuration Information) containing a table of service components.

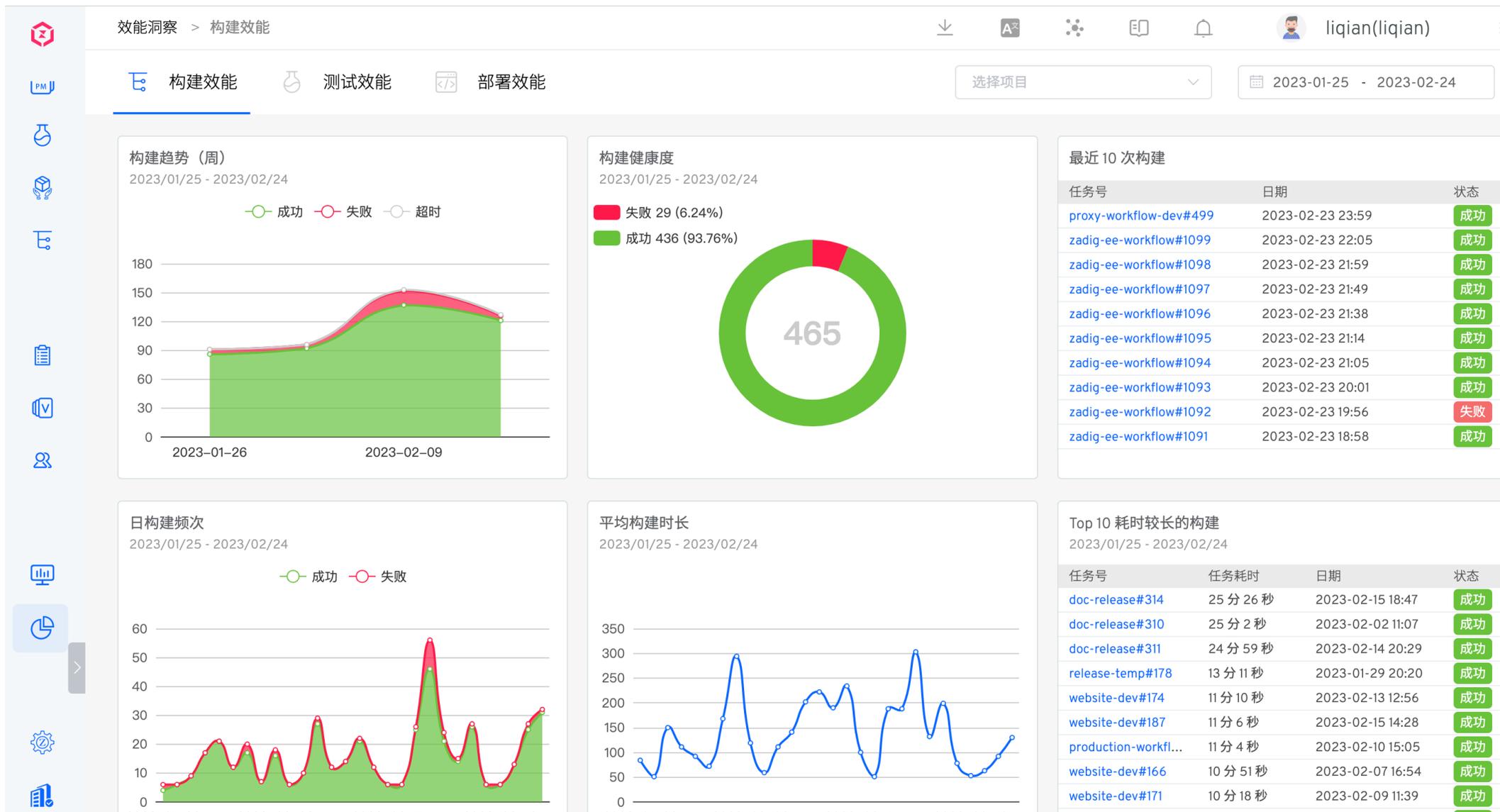
服务组件(服务名称)	镜像名称	服务配置
service1	registry.cn-hangzhou.aliyuncs.com/minmin2/service1:20230519161412-5-main	<a href="#">查看</a>
service2	registry.cn-hangzhou.aliyuncs.com/minmin2/service2:20230519162252-1-main	<a href="#">查看</a>
- 服务启动顺序 (Service Start Order):** A table showing the start order for services.

启动顺序	服务名称
0	service1, service2

# 业务负责人/产品经理 —— 项目整体运行状况



# 项目负责人——分析项目各个环境的变化过程及效能短板





3、

更多产品特性



# Zadig 产品特性：开源 Zadig 的一切

云原生 CI/CD、产研高效工程化协作、快速应对业务迭代

面向开发者的云原生环境

更新时间: 2022-07-04 10:44  
环境状态: 正在运行  
基本操作: [按钮]

服务名	状态(5/5)	镜像信息
db	Running	postgres:9.4
redis	Running	redis:alpine
result	Running	examplevotingapp...
vote	Running	examplevotingapp...

正在运行 [按钮] 产品工作流 #2 @ zadig-qa-workflow [按钮] 构建 部署 测试

正在运行 [按钮] 产品工作流 #2 @ zadig-ops-workflow [按钮] 交付物部署 测试

正在运行 [按钮] 工作流 #4 @ deploy-to-ust [按钮] 构建 部署 测试

灵活易用的高并发工作流

测试名称	测试类型	用例数量	执行次数	平均执行时间	成功率	操作
APITest	API 测试	389	316	1m27s	78.00%	[按钮]
E2ETest	E2E 冒烟测试	124	176	1m32s	92.00%	[按钮]
FullRegressionTest	全回归测试	223	141	5m29s	85.00%	[按钮]
performanceTest	性能测试	59	78	6m12s	78.00%	[按钮]
functionalTest	功能测试	89	84	1m3s	87.50%	[按钮]

高效协同的测试管理



客观精准的效能洞察

云原生 IDE 插件 (VS CODE)

Zadig Toolkit 是 Koderover 旗下开源产品 Zadig 的 Visual Studio Code 插件。该插件让开发者无需关注任何平台的差别，在 IDE 中即可获取 Zadig 产品核心能力，轻松地对 Zadig 项目中的环境进行查看和管理、做服务调试和升级、查看日志实时查看等。在日常开发过程中，代码提交后无需提交到 Git 仓库，也无须打包镜像，即可将服务部署到 Zadig 云端，方便开发者管理资源、联动和集成验证。

目前 Zadig Toolkit 支持的功能如下：

- Zadig 环境查看
- 服务列表、服务详情查看、服务详情查看、Pod 实时日志查看、登录到服务实例 Pod 中
- 远程调试本地应用程序、配置远程开发环境

前提条件

Zadig Toolkit 可以在 Windows、Linux 和 MacOS 中安装。在安装 Zadig Toolkit 之前，请确保已准备好如下准备工作：

- 已经安装 Visual Studio Code，版本要求 v1.82.0+

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: $T-Service$
  labels:
    app.kubernetes.io/name: $T-Project$
    app.kubernetes.io/instance: $T-Service$
spec:
  selector:
    matchLabels:
      app.kubernetes.io/name: $T-Project$
      app.kubernetes.io/instance: $T-Service$
  replicas: 1
```

变量列表

Key	描述
\$T-Project\$	项目名称
\$T-Service\$	服务名称

port: 20221, targetport: 20221

强大免运维的模板库

强大免运维的模板库

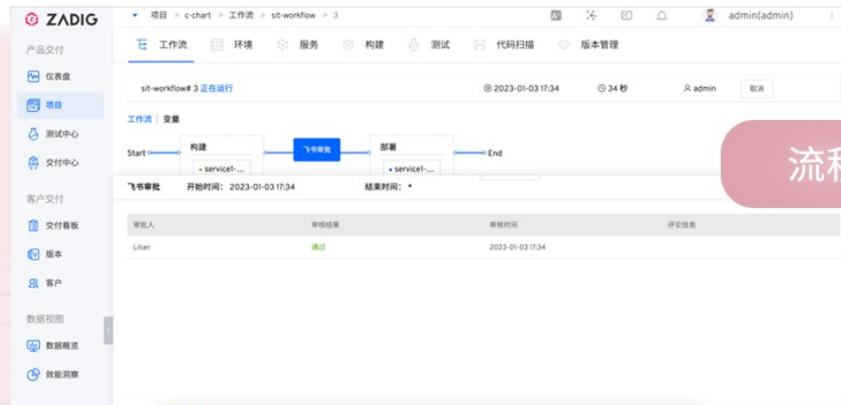
# Zadig 产品特性：发布中心

编排组织、流程、内外部系统，管理代码、配置、数据变更流程，支持灰度组合策略

多服务编排升级



配置变更

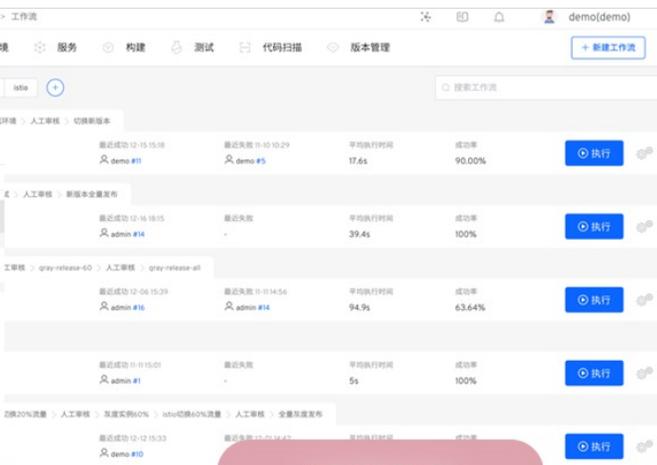


流程自动化



数据变更

生产观测



灰度发布

# Zadig 产品特性：客户交付

面向大客户全天候响应、全地域升级部署，提供自运维和专用服务通道，实现稳定高效交付和服务保障

**交付看板** **供应商产品生命周期管理**

trialGroup看板

- 版本采用率: 75%当前版本v1.0.1-rc
- 客户: 3 | 4 **创建客户** [详情](#)
- 最新版本: v1.0.1-rc [历史版本](#)
- 安装

vipGroup看板

- 版本采用率: 无数据
- 客户: **创建客户**
- 请先 **推送** 或 **创建版本** 到客户组
- 最新版本: **创建版本**
- 安装

**版本管理** **为客户提供生命周期管理**

当前版本: v1.0.0 序号1 (2022-08-19 15:35:44)

发现新版本 v1.0.2 序号3 **发现新版本**

所有版本:

- v1.0.2 序号3: 于2022-08-19 17:19:41发布 | 供应商发版 | 环境检测结果 | **部署** (部署于N/A)
- v1.0.1 序号2: 发布 | 供应商发版 | 环境检测结果 | **部署** (部署于N/A)

**管理不同渠道的版本**

zadig-ee

活跃版本

所有版本

- v1.0.1-rc 序号4: 活跃在 trialGroup | 创建时间: 2022-09-15 15:00:16 | 创建人: Lillian
- v1.0.2-beta 序号3: 非活跃 | 创建时间: 2022-09-02 15:07:17 | 创建人: Landy
- v1.0.1-beta 序号2: 非活跃 | 创建时间: 2022-09-02 15:07:17 | 创建人: Landy
- v1.0.0-beta 序号1: 非活跃 | 创建时间: 2022-09-02 15:07:17 | 创建人: Landy

**安装**

现有集群 | 内置集群

```
export VENDOR_SERVER=https://  
INSTALL_NAMESPACE=zadig-ee  
PLUTUS_MGO_ADDR=mongodb://127.0.0.1:27017
```

**复制**

**可部署到任何环境**

# 公司介绍

成立于 2018 年，KodeRover 是国内在云原生 DevOps 领域的领军企业。旗舰产品云原生 DevOps 软件工程平台 Zadig 正在成为数字化软件研发的新标配，帮助企业全面实现产研数字化转型。核心团队由 DevOps 领域云计算工程技术专家和高级工程师为主，核心成员是来自于谷歌，腾讯，七牛云，阿里的技术骨干。公司已经连续完成来自顶级资本的多轮融资。

自 Zadig 2021 年开源以来，公司连续两年被评为“开源中国”的“年度优秀开源技术团队”，2022 年度 36 氪「新经济之王」基础软件赛道的“年度企业”，2022 年度 GADI 汽车新智造创新行业峰会的“最佳数字化服务商企业”。经过几年的快速发展，Zadig 正在成为产业数字化时代软件交付领域的规则制定者，开启软件交付 3.0 时代，帮助企业打造全球领先的软件工程。



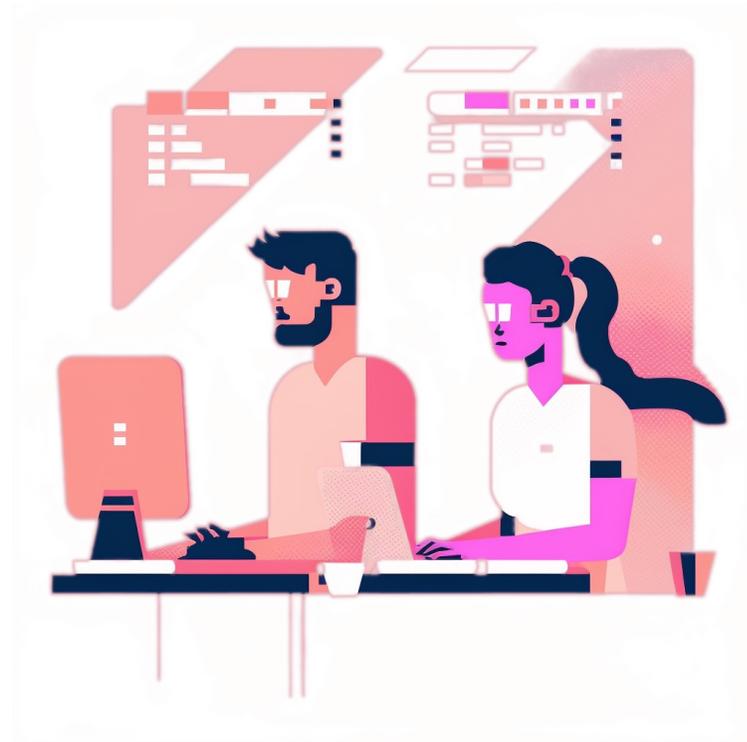
# 创始团队

## 李倩 (Landy) :

KodeRover 创始人兼 CEO , Zadig 开源作者/总架构师 , 前七牛云工程总监 , 阿里云 MVP , QCon/MSUP/TGO/极客时间效能特邀讲师。为数百研发团队 , 数千名工程师提供效能实战培训 , 包括腾讯 PCG/TEG、字节跳动、小红书等。专注在云原生 CI/CD、DevOps、平台工程架构等大规模软件研发的基础工程领域的研究和应用。主导研发的 Zadig 开源 DevOps 软件工程平台已被 2000 多家企业广泛使用 , 包括字节跳动飞书、极氪、小鹏、路特斯等数百家各行业领军企业。

## 郭健 (Grant) :

KodeRover 创始人兼 COO , 硅谷 21 年经验 , 谷歌早期工程师 , 硅谷连续创业者 , 斯坦福大学运筹工程学系全额奖学金毕业。





产研最佳实践

 KODEROVER

