产品简介

CICDKit 是蓝鲸 DevOps 平台中为软件开发项目提供持续集成、持续部署支持的 SaaS 套件。通过页面上简单的信息填写即可以配置出一个适合开发团队特点的持续集成、持续部署流水线，自动化完成软件开发项目过程中包括代码仓库集成、编译构建、单元测试、代码检查、制品归档、镜像构建、自动部署等各个环节，减少开发和运维之间协作的时间损耗，使团队整体更加高效地协同工作、迭代功能、发布产品。平台不仅内置了轻量级的容器构建环境以满足拿来即用的需求，还提供了构建机的扩展集成，用来实现不同团队复杂的构建支持。此外，平台也提供了度量统计报表，通过对流水线的历史执行状况进行统计和分析，供项目经理做管理优化提供数据参考。

术语解释

DevOps： DevOps（ Development 和 Operations 的组合词）是一组过程、方法与系统的统称，用于促进开发（应用程序/软件工程）、技术运营和质量保障（QA）部门之间的沟通、协作与整合。

CICD： CICD 是 持续集成（Continuous Integration）和持续部署（Continuous Deployment）简称。指在开发过程中自动执行一系列脚本来减低开发引入 bug 的概率，在新代码从开发到部署的过程中，尽量减少人工的介入。

流水线： 流水线是指在程序执行时多条指令重叠进行操作的一种准并行处理实现技术。

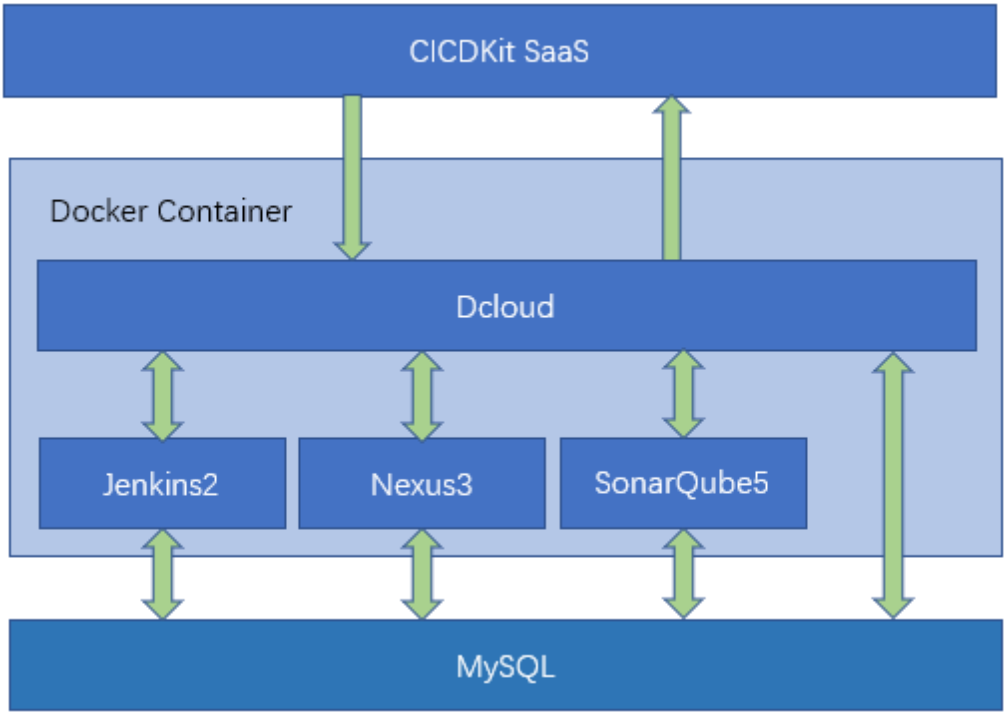
Docker： 是一个开源的应用容器引擎，让开发者可以打包他们的应用以及依赖包到一个可移植的镜像中，然后发布到任何流行的 Linux 或 Windows 机器上，也可以实现虚拟化。容器是完全使用沙箱机制，相互之间不会有任何接口。

Dockerfile： 是用来构建Docker镜像的构建文件，是由一系列命令和参数构成的脚本。

归档： 是指存储有组织的数据。归档的目的是长时间存放有组织的数据集，确保其将来能够被精细地检索。

构件： 构件是系统中实际存在的可更换部分，它实现特定的功能，符合一套接口标准并实现一组接口。构件代表系统中的一部分物理实施，包括软件代码（源代码、二进制代码或可执行代码）或其等价物（如脚本或命令文件）。

产品架构图

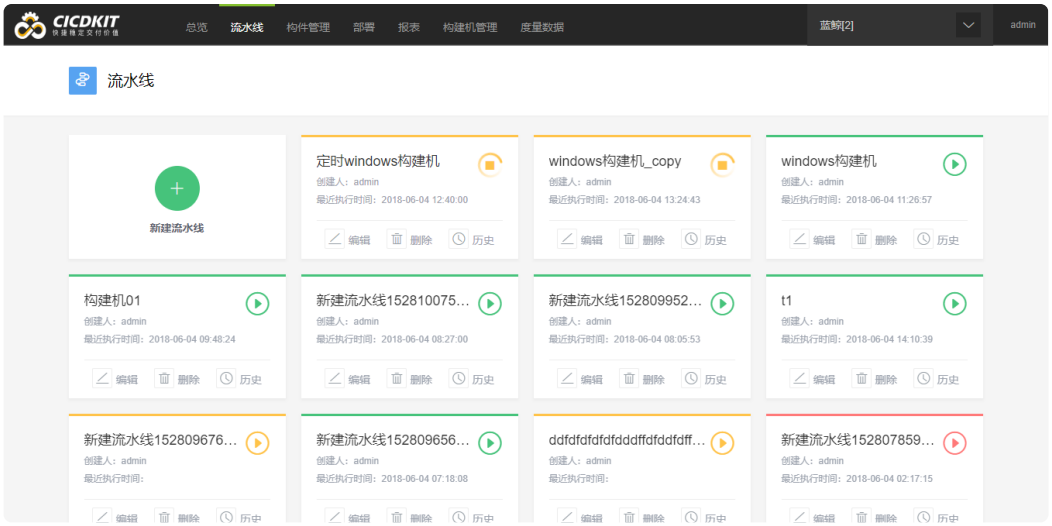


首页总览

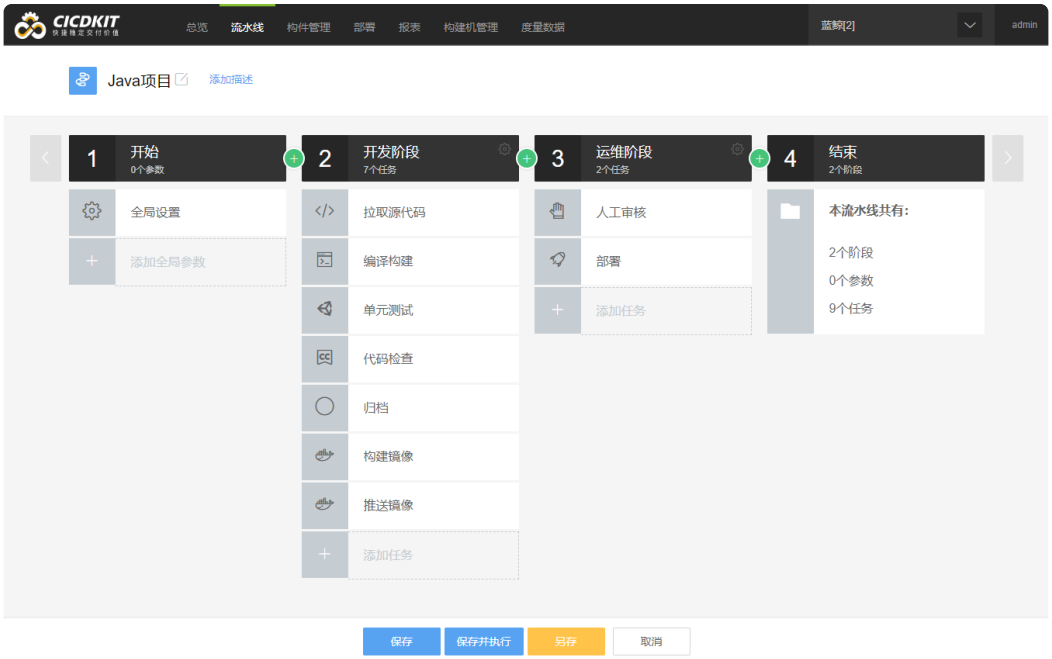
首页总览包括流水线、代码、部署等统计汇总信息，支持自定义常用任务，可查看流水线的最近执行记录。

流水线

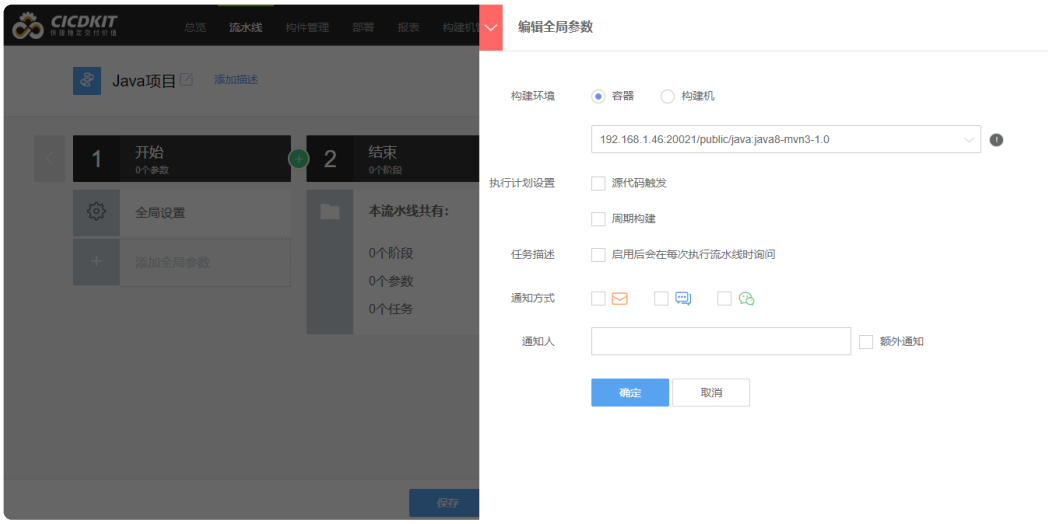
流水线页面默认显示流水线列表，可快捷启动、停止已经设计好的流水线，也可点击对应链接进入编辑、执行历史等功能页面。



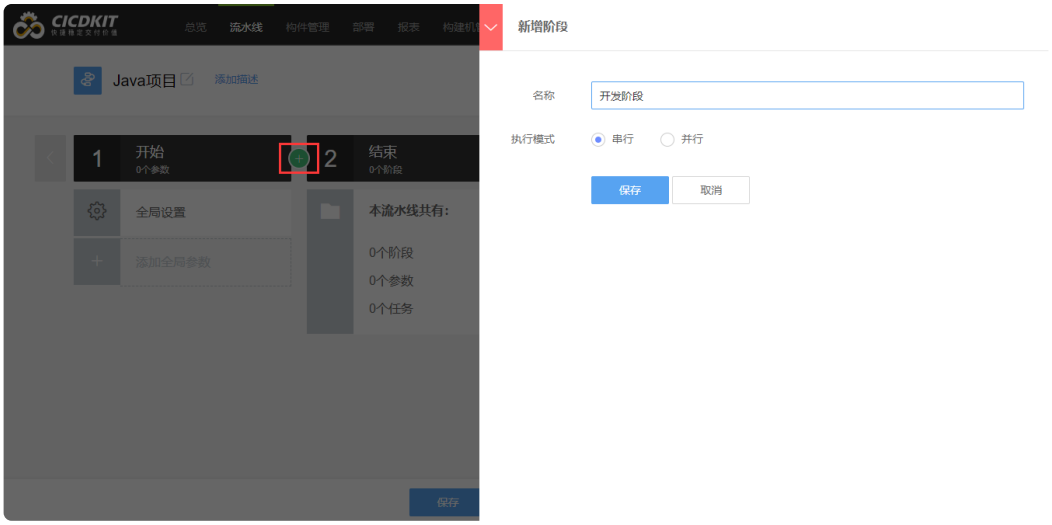
流水线编辑页面包括全局参数设置、阶段设计、原子节点编排等功能，可根据团队需求自定义符合需求的流水线。



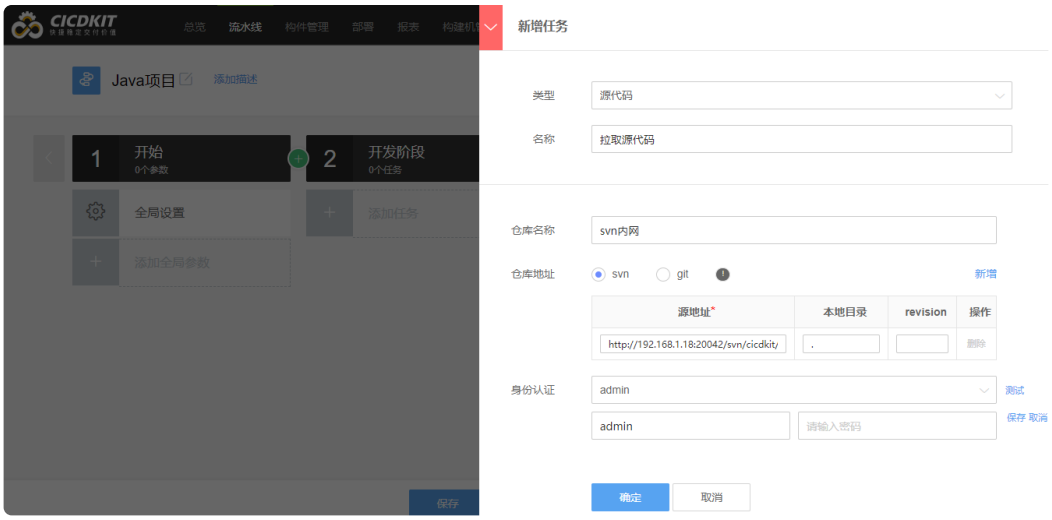
全局设置用于配置流水线的构建方式、触发机制及通知类型，构建环境支持内置容器构建或自定义构建机构建。



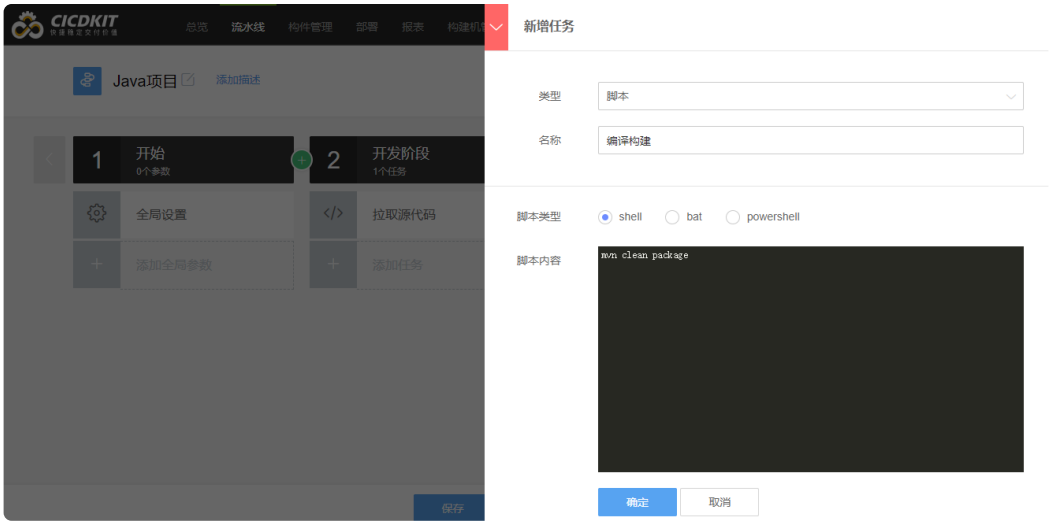
可自定义添加多个不同阶段，并设置阶段内原子节点之间是串行或并行。



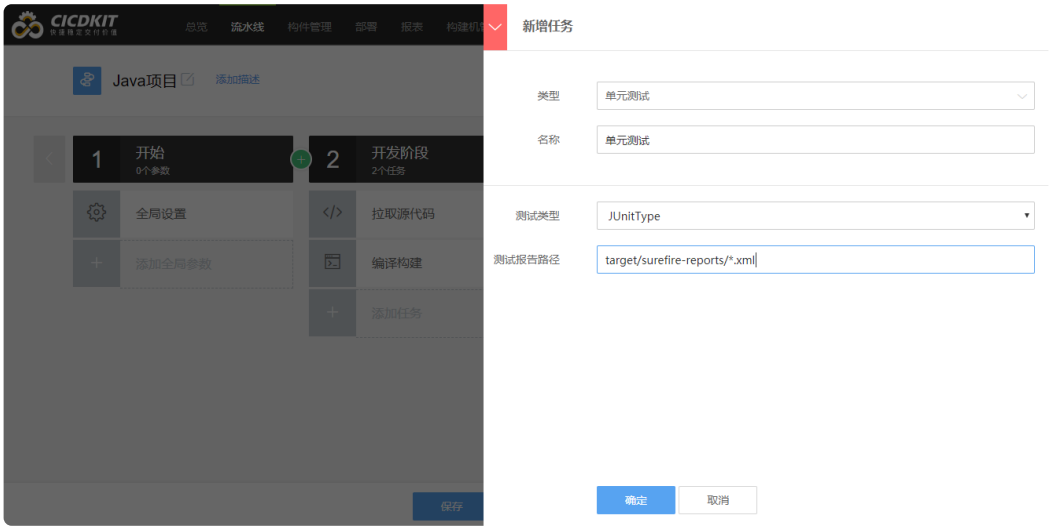
代码原子节点，配置从代码仓库拉取源代码的相关信息，支持 git 和 svn 两种方式。



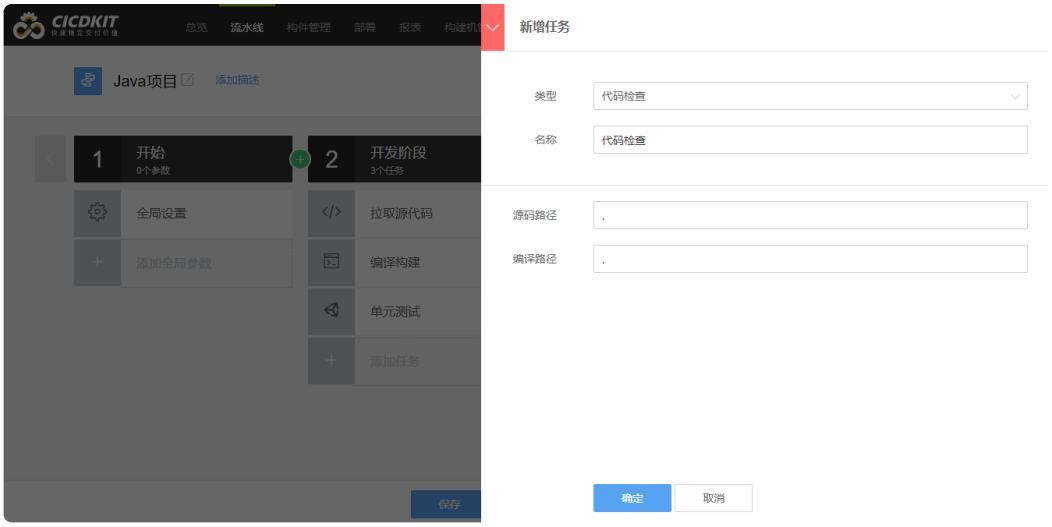
脚本原子节点，编写在目标构建环境中执行的脚本， Linux 构建环境支持 shell， Windows 构建环境支持 bat 和 powershell 。



单元测试原子节点，用于将单元测试代码的运行结果加工成结构化数据，并在报表中的单元测试报告栏目显示。



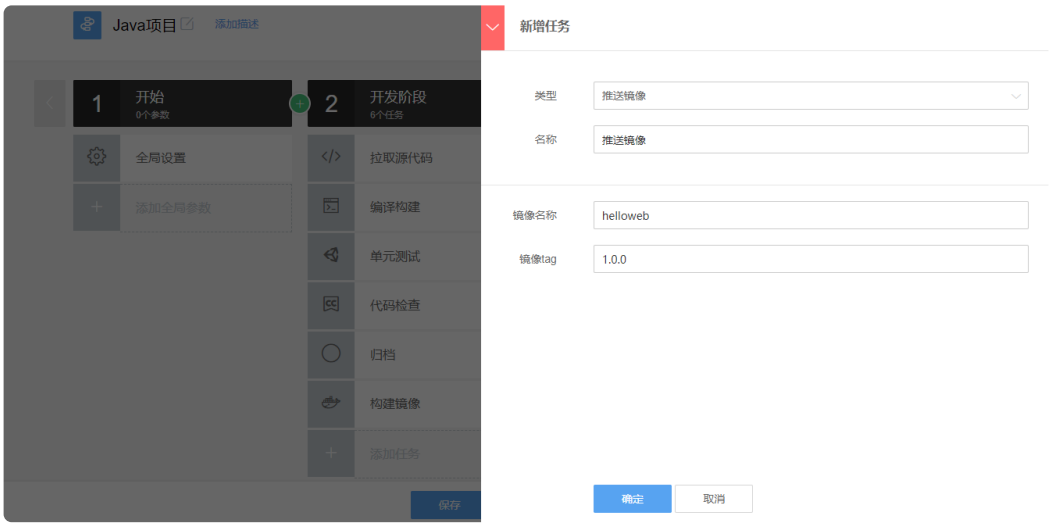
代码扫描原子节点，用于对源代码进行静态扫描，发现其中的漏洞、潜在 Bug、重复代码等信息，并在报表中的代码检查栏目展示。



归档原子节点，用于将编译输出的程序包归档到构件仓库，供程序部署时使用。

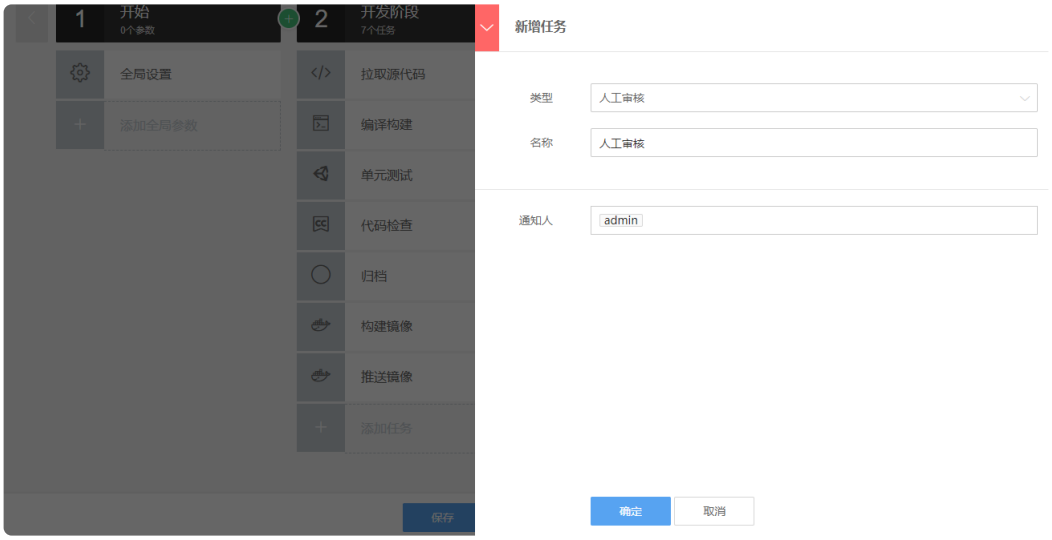
构建镜像原子节点，用于将程序包打包生成一个 Docker 镜像，Dockerfile 需由开发或运维团队设计编写。

推送镜像原子节点，将上一步构建的镜像归档到镜像仓库中。



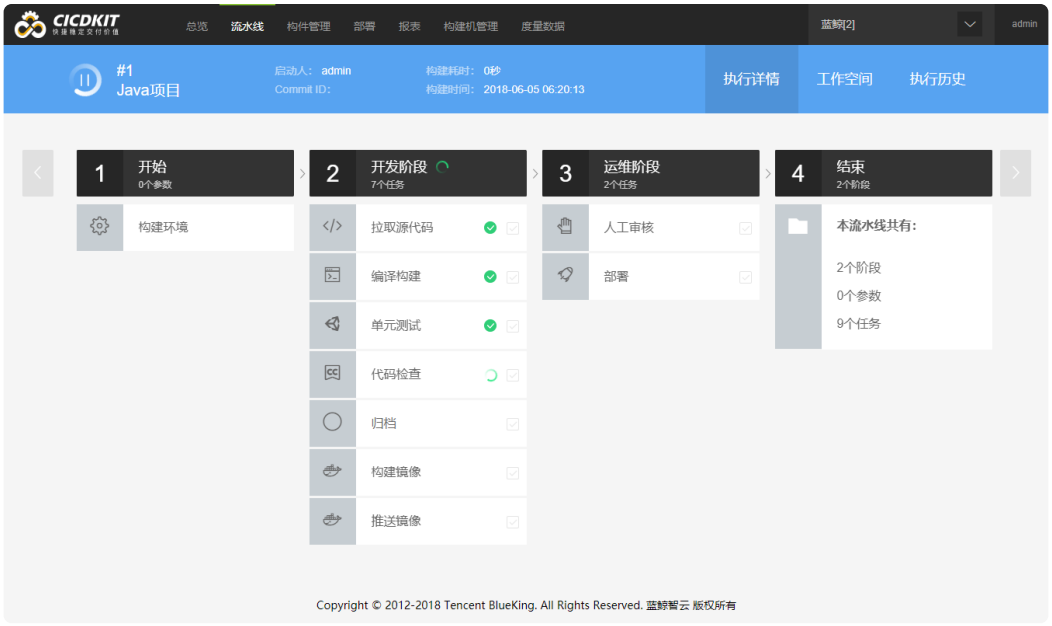
构建并推送镜像，即构建镜像和推送镜像两个步骤的合并。

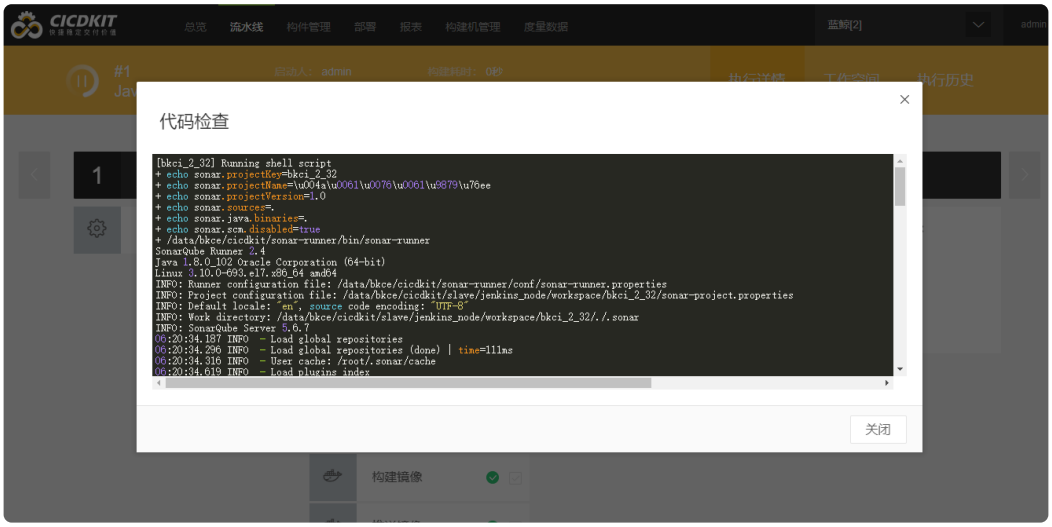
人工审核原子节点，用于简单的审批确认，流水线运行到该节点会自动挂起，人工确认后继续执行，也可选择终止运行。



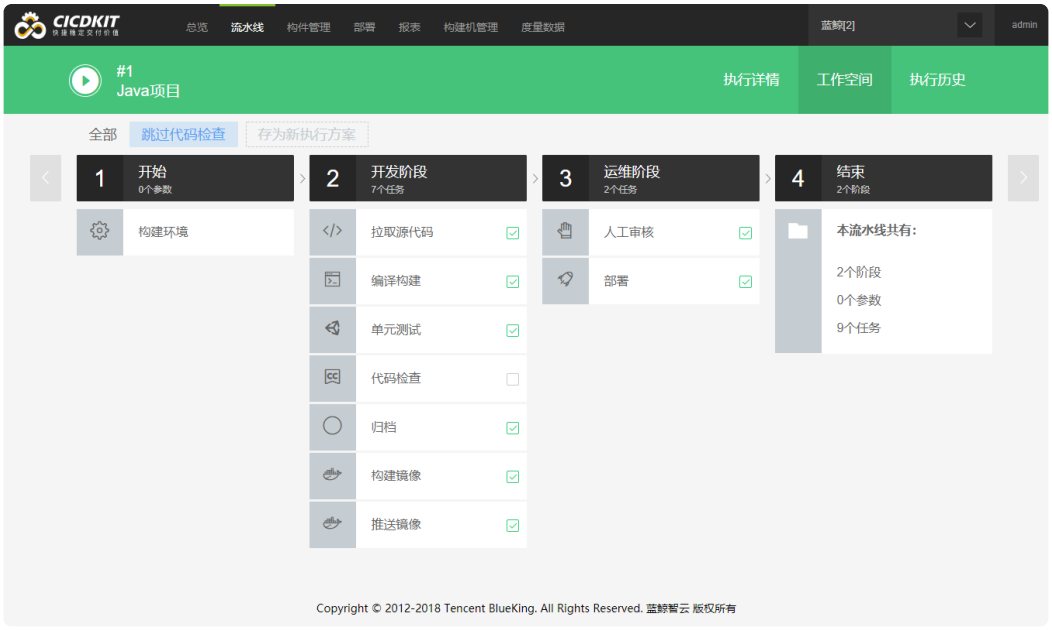
部署原子节点，调用标准运维定义的部署流程，实现程序包的批量发布部署。

流水线执行过程可视化，点击原子节点可查看各阶段日志。

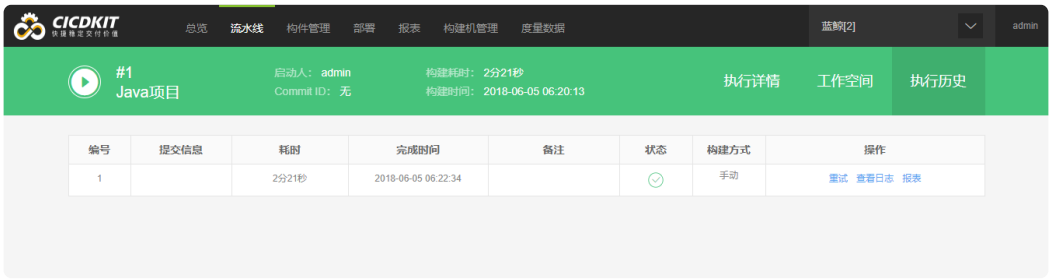




工作空间里可自定义流水线执行的子集，比如跳过一些非必须的节点以加快构建速度。

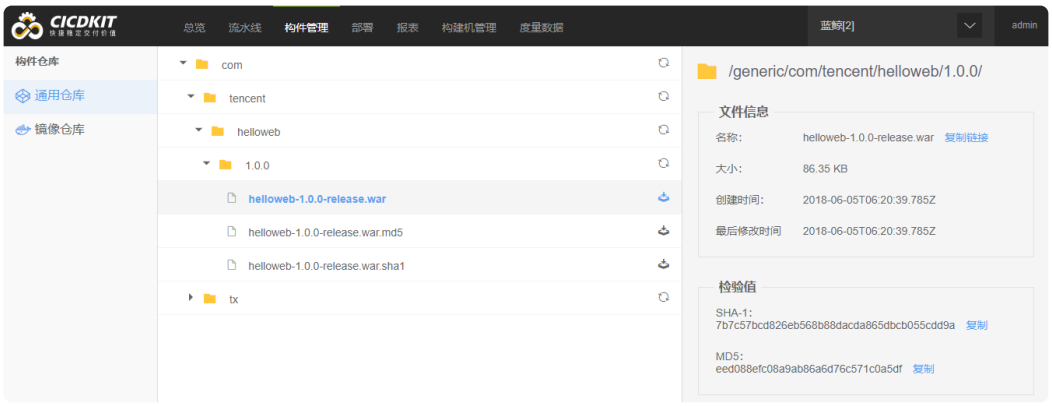


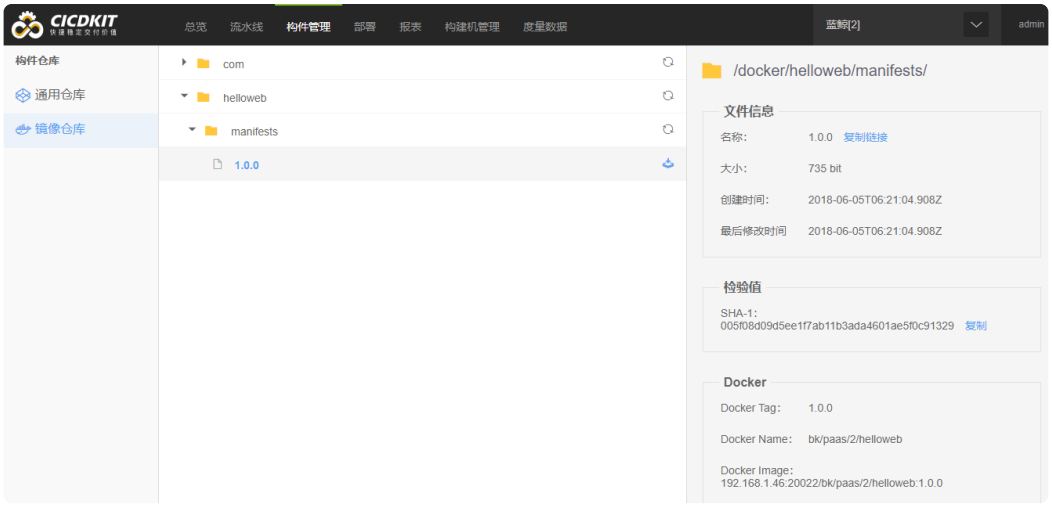
执行历史可查看具体某个流水线的历史执行记录，包括执行结果、日志、相关说明等。



构件管理

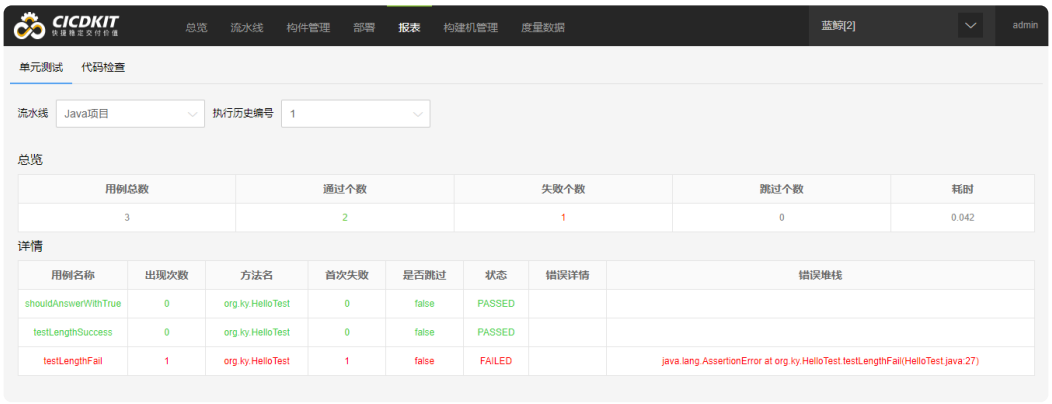
流水线中执行归档或镜像推送操作的程序包，会统一存放一份在构件仓库中。构件仓库分为通用仓库和镜像仓库两种，所有类型的二进制程序包存放在通用仓库，Docker 镜像存放在镜像仓库。通用仓库可复制链接进行程序包下载，镜像仓库则可通过 Docker Image 的进行 pull 获取镜像。



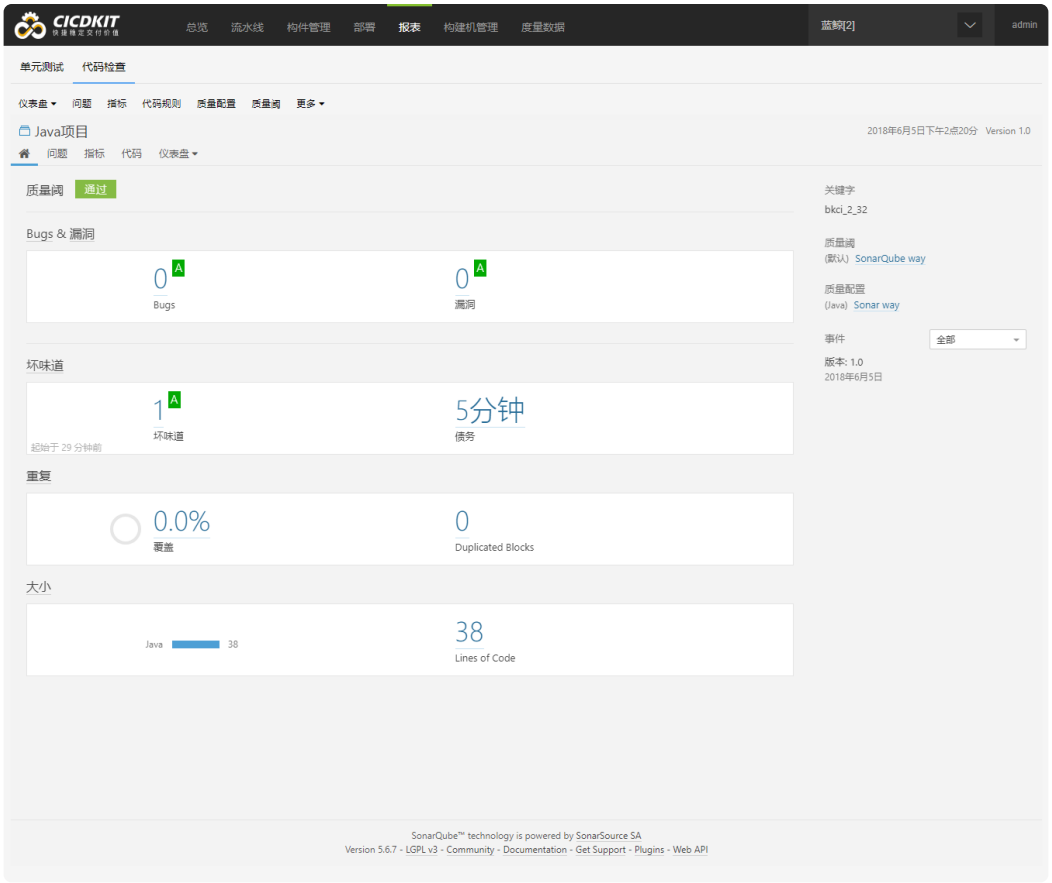


报表

如果流水线的执行过程包括单元测试节点，那么执行完成后会在单元测试的页面生成一份测试报表。



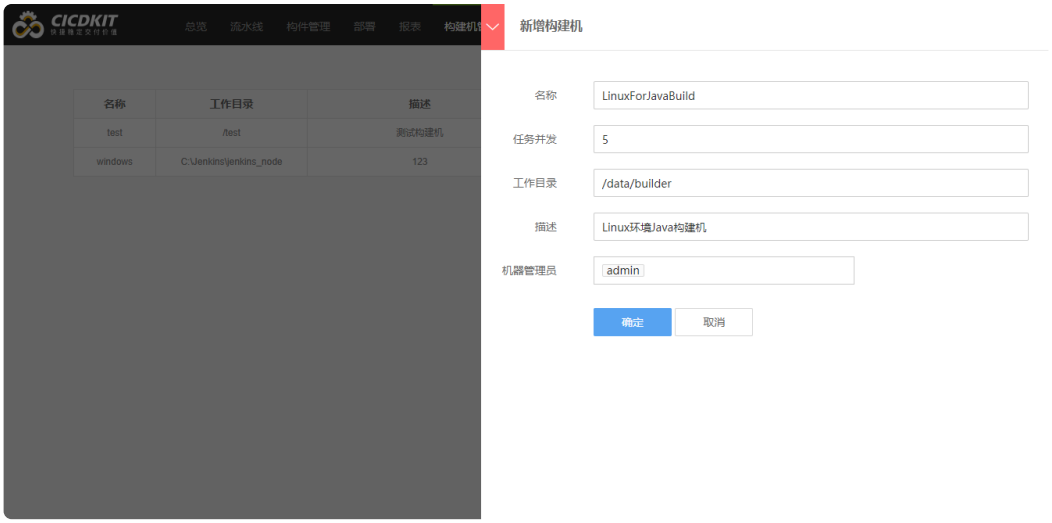
如果流水线的执行过程包括代码检查节点，那么执行完成后会在代码检查页面生成对应项目的扫描结果报告。



构建机管理

CICDKit 除了可以在容器中构建程序，还支持自定义构建机来执行构建任务。构建机指用户配置好的一台可以编译构建程序的物理机或虚拟机，通过安装一个 Agent 程序来加入 CICDKit 的调度，当流水线指定构建机构建时，具体的构建任务会转移到该构建机上来执行。

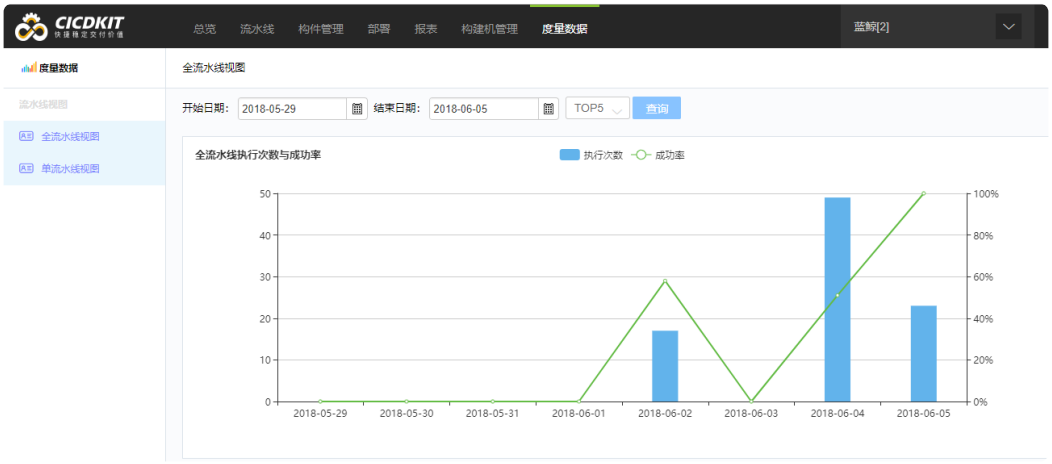
新建构建机，定义相关参数。



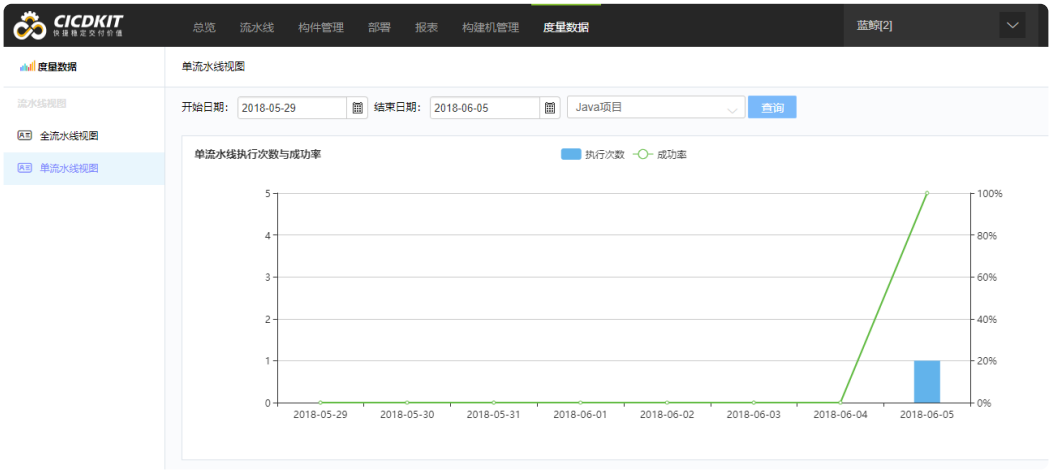
点击激活按钮，根据说明在构建机上进行相应的操作，构建机处于在线状态则准备就绪，可以在流水线的全局设置中采用构建机构建，并选择在线状态的构建机。

度量数据

全流水线度量数据：



单流水线度量数据：



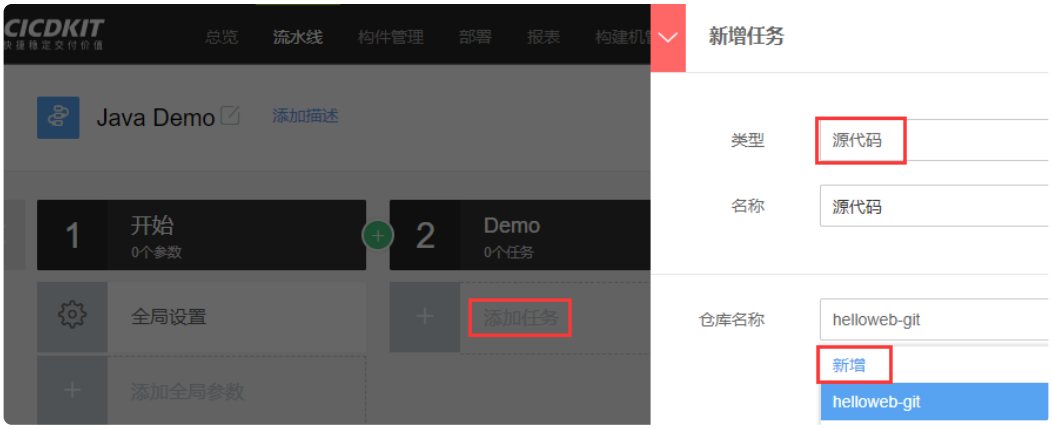
快速入门

新建流水线并修改流水线名称：

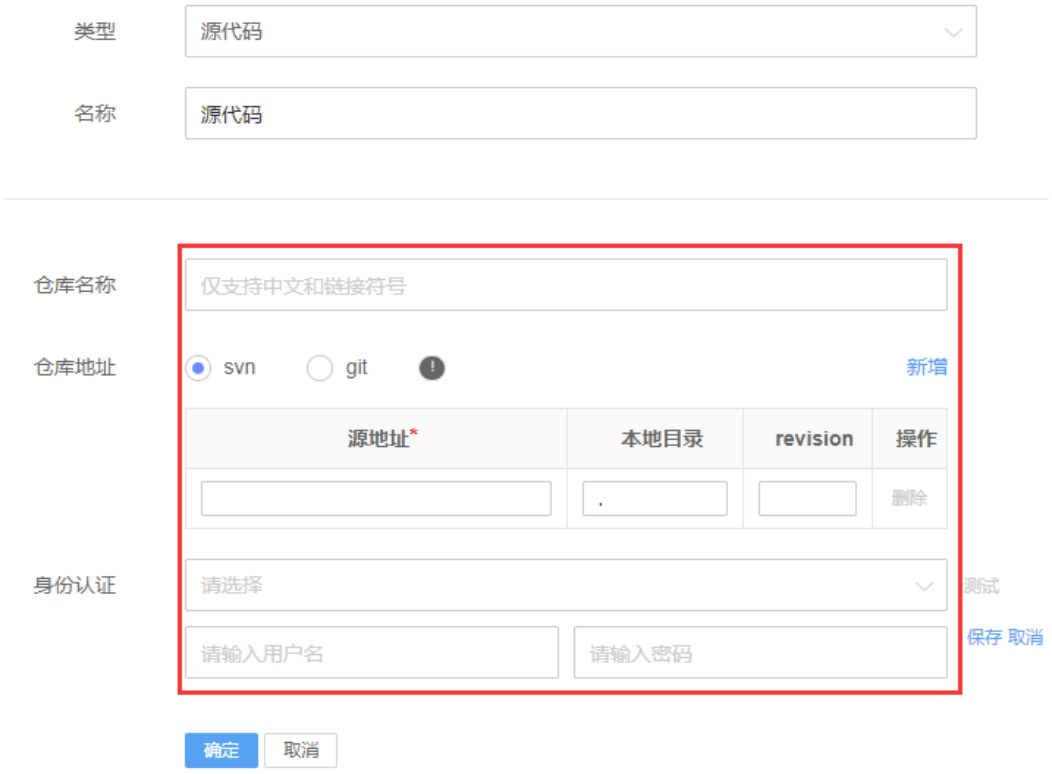
点击全局设置，选择容器构建，并选择 Java 构建镜像：

新增一个阶段，选择串行：

新增一个任务，选择源代码，并添加一个代码仓库：



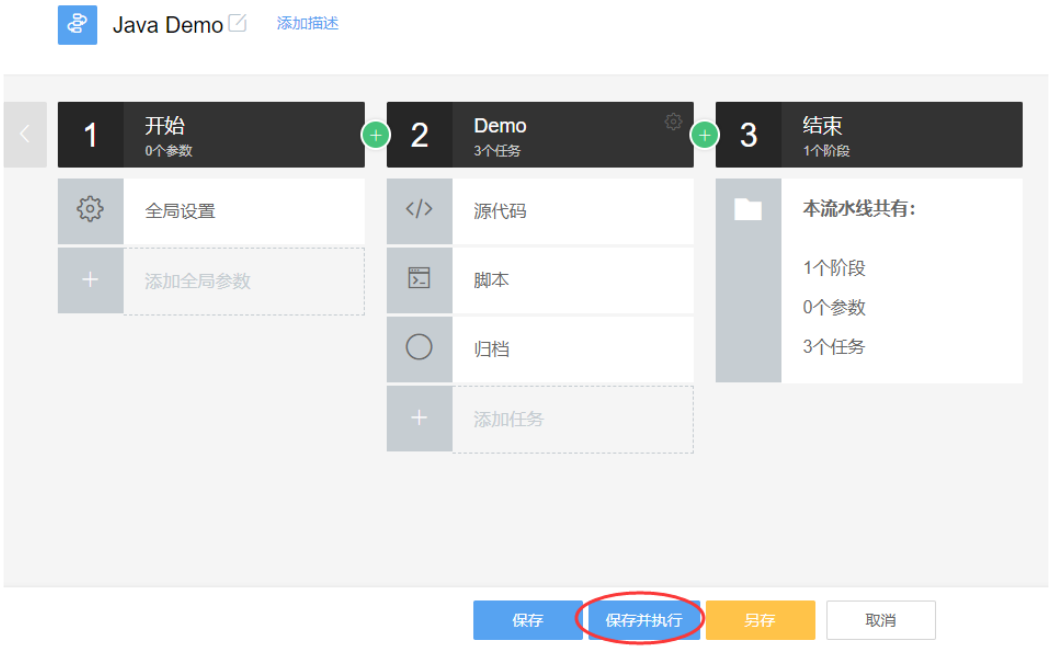
填写代码仓库的配置信息（这里请填写一个 Java Maven 项目的真实信息）：

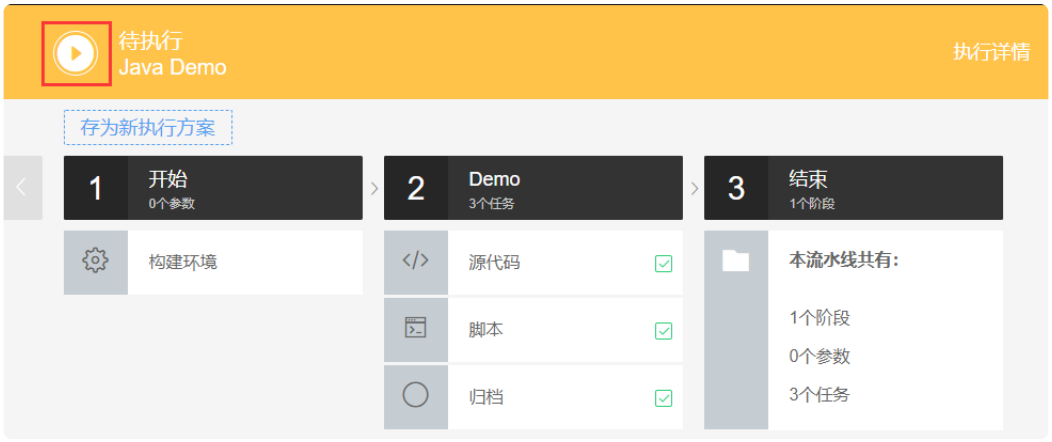


添加一个脚本节点，选择 shell 类型，并输入编译 Java Maven 项目的指令： mvn clean package

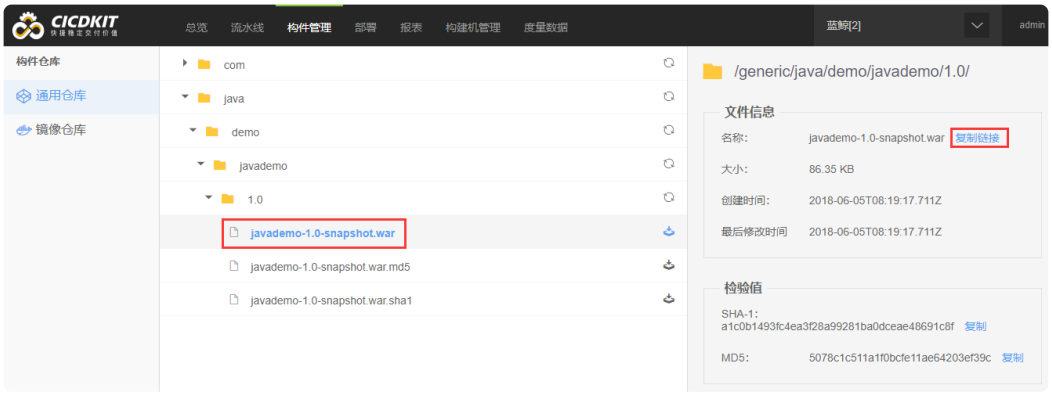
添加一个归档节点，填写归档路径（即构件仓库中的相对路径）、版本号、归档项目（项目名称）、归档文件类型（保持跟编译后的程序包类型一致）、归档分类（可自定义，比如 snapshot、release 等）、源文件路径（即编译产出的程序包路径，可用相对路径）：

保存并执行流水线，先检查测试一下是否正常。

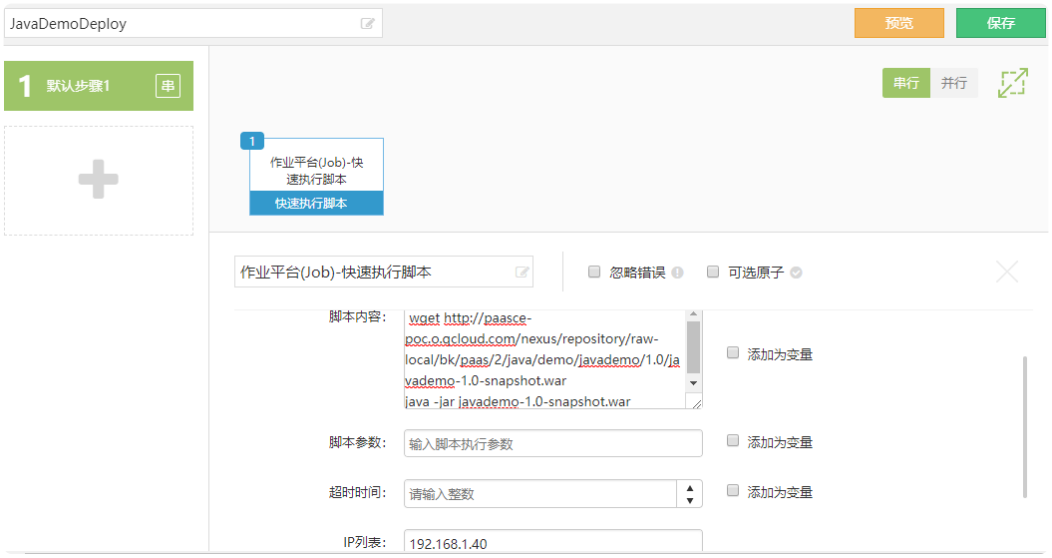




若执行没有报错，到构件管理栏目中找一下刚才归档的文件，复制链接可以获取到该程序包的下载地址：



到部署栏目新增一个单节点快速执行脚本的简单的流程，脚本内容包括下载程序包（即 wget 代码仓库中的程序包下载地址）、启动程序（如果是 SpringBoot 项目可直接 java -jar 启动，如果是传统项目则需准备 Tomcat 之类的服务器）等步骤，并填写部署的目标服务器 IP、用户等信息：（该步骤更多细节请参考标准运维相关产品介绍）



回到流水线编辑界面，增加一个部署节点，并选择刚刚创建的部署流程：

保存后再次执行流水线，即可完成一个项目的简单 CI、CD 过程，更多功能节点可按需进行配置和使用：

场景案例

如何使用构建机

构建机功能为不同开发语言、不同开发风格的团队提供了一个个性化定义的扩展功能，理论上只要代码可以在某一台物理机或虚拟机上进行程序的编译构建，就可以接入到 CICDKit 的流水线中。构建机的接入步骤大致如下：

- 准备一台可以独立完成构建任务的构建服务器，该服务器应该包括编译构建程序的必要环境配置和组件，比如 Java 项目至少需要包括 JDK、Maven/Gradle 等编译工具，并且该服务器应该与蓝鲸平台相关服务器保持网路畅通；
- 在 CICDKit 构建机管理中，新增一个构建机；

- 在新增完的构建机列表右侧的点击激活按钮，下载 agent.jar 并放到构建服务器上；

- 在构建服务器上安装 JDK（如果已安装过 JDK 则跳过此步骤）；
- 拷贝激活页面的命令，到构建服务器上 agent.jar 所在目录运行；

- 成功运行后构建机会处于在线状态，然后就可以在流水线定义中使用。

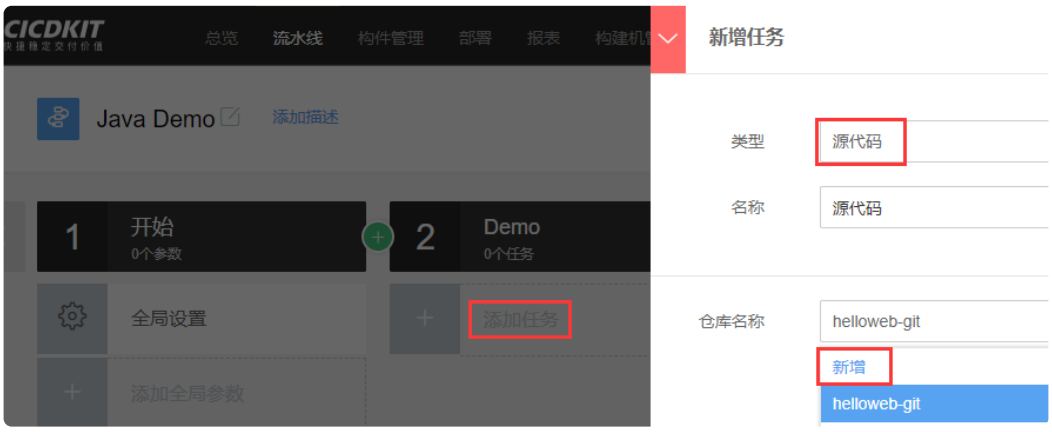
场景示例

新建流水线并修改流水线名称：

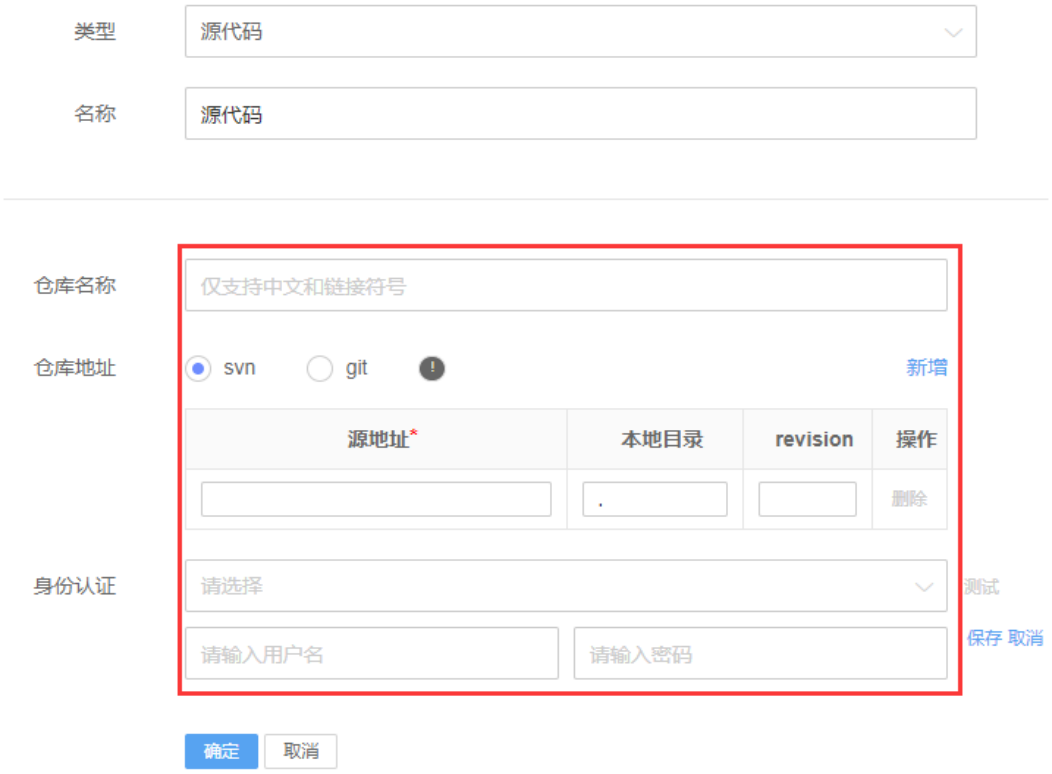
点击全局设置，选择容器构建，并选择 Java 构建镜像：

新增一个阶段，选择串行：

新增一个任务，选择源代码，并添加一个代码仓库：

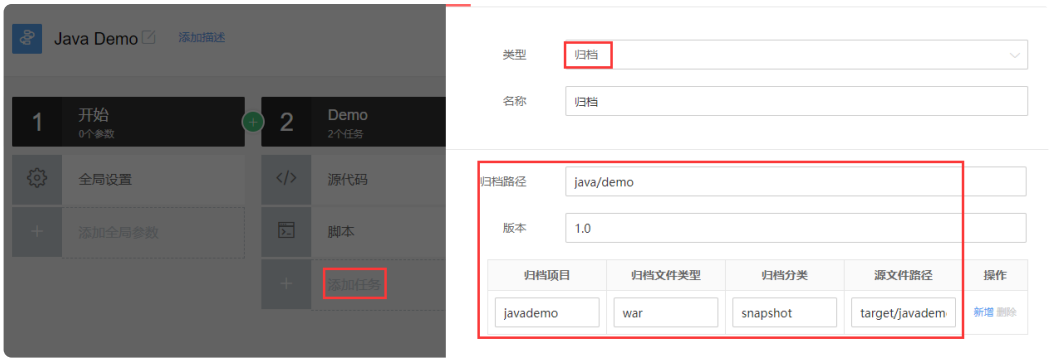


填写代码仓库的配置信息（这里请填写一个 Java Maven 项目的真实信息）：

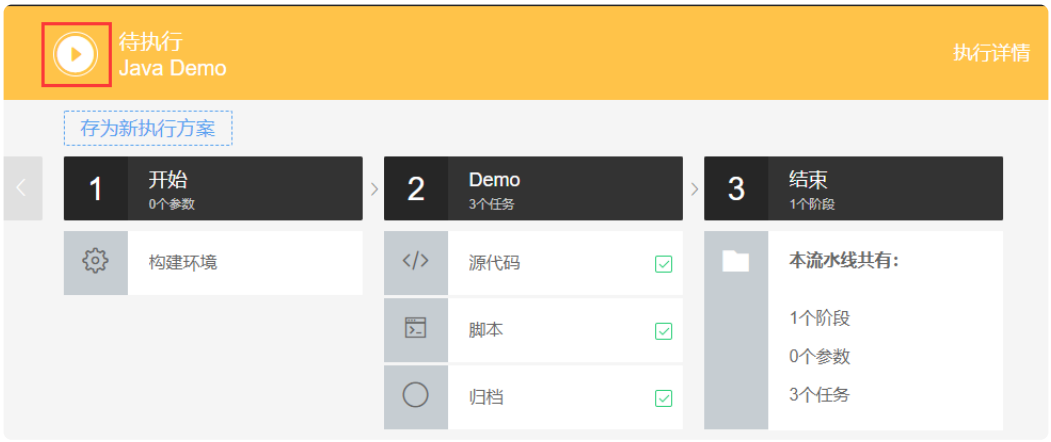


添加一个脚本节点，选择 shell 类型，并输入编译 Java Maven 项目的指令：mvn clean package

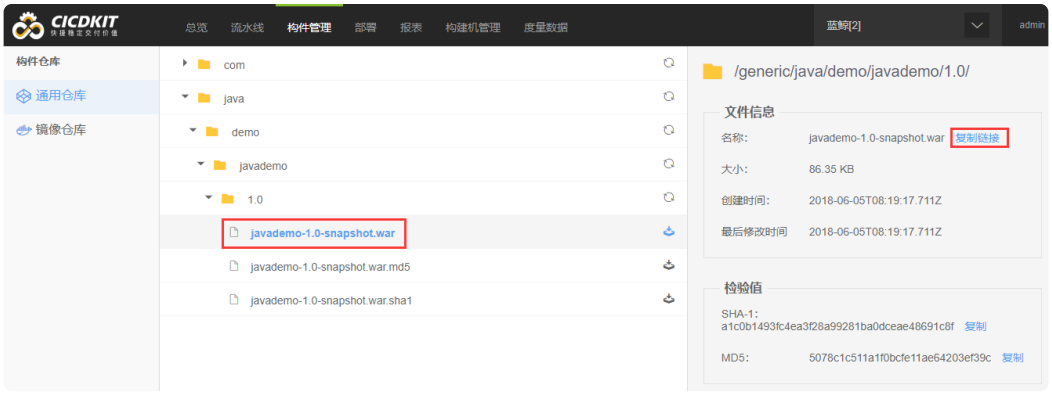
添加一个归档节点，填写归档路径（即构件仓库中的相对路径）、版本号、归档项目（项目名称）、归档文件类型（保持跟编译后的程序包类型一致）、归档分类（可自定义，比如 snapshot、release 等）、源文件路径（即编译产出的程序包路径，可用相对路径）：



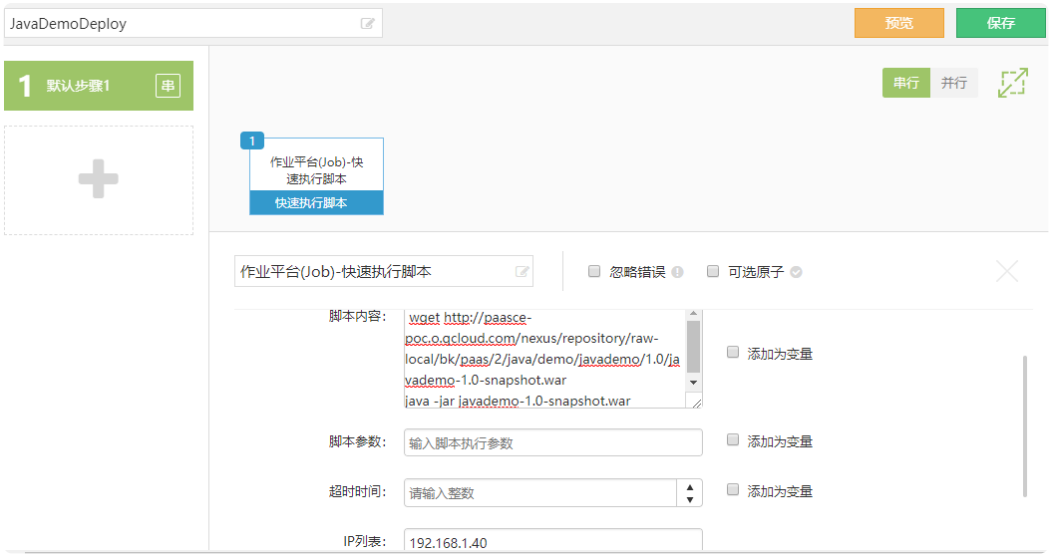
保存并执行流水线，先检查测试一下是否正常。



若执行没有报错，到构件管理栏目中找一下刚才归档的文件，复制链接可以获取到该程序包的下载地址：



到部署栏目新增一个单节点快速执行脚本的简单的流程，脚本内容包括下载程序包（即 wget 代码仓库中的程序包下载地址）、启动程序（如果是 SpringBoot 项目可直接 java -jar 启动，如果是传统项目则需准备 Tomcat 之类的服务器）等步骤，并填写部署的目标服务器 IP、用户等信息：（该步骤更多细节请参考标准运维相关产品介绍）



回到流水线编辑界面，增加一个部署节点，并选择刚刚创建的部署流程：

保存后再次执行流水线，即可完成一个项目的简单 CI、CD 过程，更多功能节点可按需进行配置和使用：

常见问题

CICDKit 目前支持哪些开发语言的持续集成？

CICDKit 内置了 Java 和 C++ 的构建容器，可以满足标准 Java 项目和 C++ 项目开箱即用的持续集成。除此之外，通过自定义构建机，理论上任何开发语言的程序，只要可以在构建机上进行编译构建，都可以接入流水线中，完成持续集成和持续发布。

CICDKit 可以单独部署吗？

CICDKit 作为蓝鲸平台的 SaaS 应用，会依赖蓝鲸的部分平台功能和组件，因此在部署 CICDKit 之前，需要先准备好蓝鲸平台，蓝鲸社区版即可集成 CICDKit。