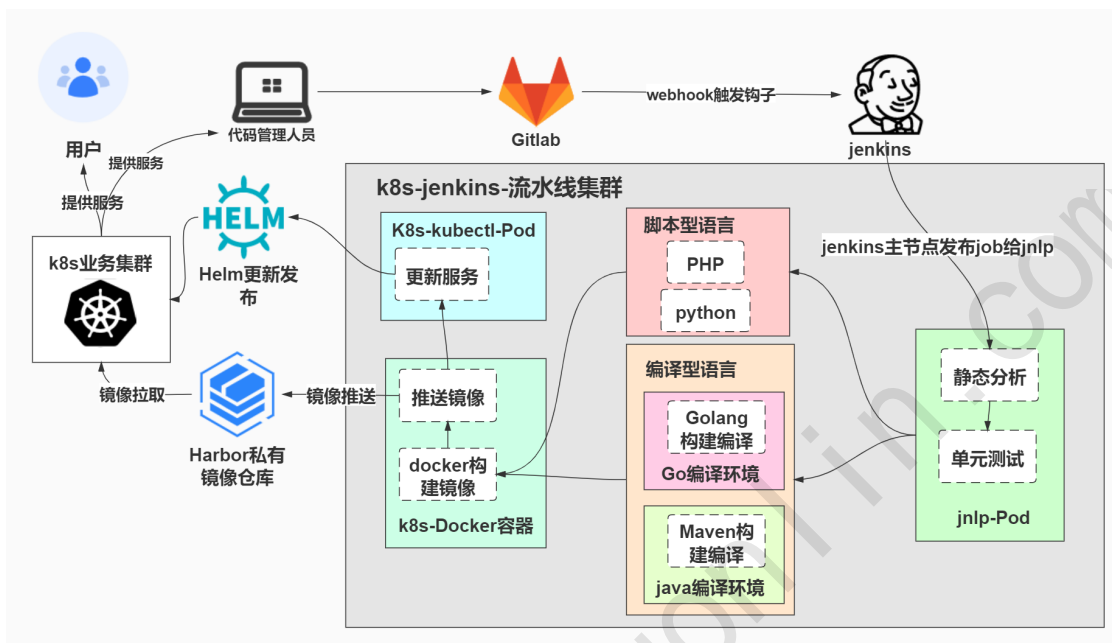


jenkins 面试题



一、jenkins 是什么？

Jenkins 是一个开源的、可扩展的持续集成、交付、部署（软件/代码的编译、打包、部署）的基于 web 界面的平台。允许持续集成和持续交付项目，无论用的是什么平台，可以处理任何类型的构建或持续集成。

二、为什么使用 jenkins

Jenkins 是一种使用 Java 编程语言编写的开源持续集成软件工具，用于实时测试和报告较大代码库中的孤立更改。Jenkins 软件使开发人员能够快速找到并解决代码库中的缺陷，并自动进行构建测试。

三、CI/CD 是什么？

CI(Continuous integration, 中文意思是持续集成)是一种软件开发时间。持续集成强调开发人员提交了新代码之后，立刻进行构建、（单元）测试。根据测试结果，我们可以确定新代码和原有代码能否正确地集成在一起。

CD(Continuous Delivery, 中文意思持续交付)是在持续集成的基础上，将集成后的代码部署到更贴近真实运行环境(类生产环境)中。比如，我们完成单元测试后，可以把代码部署到连接数据库的 Staging 环境中更多的测试。如果代码没有问题，可以继续手动部署到生产环境。

四、什么是 jenkins pipeline?

Pipeline, 简而言之, 就是一台运行于 Jenkins 上的工作流框架, 将原本独立运行于单个或者多个节点的任务连接起来, 实现单个任务难以完成的复杂流程编排与可视化。

Jenkins Pipeline 是一组插件, 让 Jenkins 可以实现持续交付管道的落地和实施。

五、什么是 jenkinsfile?为什么使用 jenkinsfile

Jenkinsfile 是一个文本文件, 其中包含 Jenkins Pipeline 的定义, 并已签入源代码管理

虽然用于定义管道的脚本语法和 jenkinsfile 类似, 但通常认为在项目中定义管道 Jenkinsfile 并检查源代码管理是最佳实践。

为所有分支和请求自动创建一个管道构建过程。

管道上的代码审查/迭代。

审核追踪管道

六、什么是 Blue Ocean

Blue Ocean 是 pipeline 的可视化 UI。同时他兼容经典的自由模式的 job。Jenkins Pipeline 从头开始设计, 但仍与自由式作业兼容, Blue Ocean 减少了经典模式下的混乱并为团队中的每个成员增加了清晰度。Blue Ocean 的主要特点包括:

连续交付 (CD) 管道的复杂可视化, 可以让您快速直观地理解管道状态。

管道编辑器 - 通过引导用户通过直观和可视化的过程来创建管道, 从而使管道的创建变得平易近人。

个性化以适应团队中每个成员的基于角色的需求。

在需要干预和/或出现问题时确定精确度。Blue Ocean 显示的标注了关键步骤, 促进异常处理和提高生产力。

七、jenkins 的优势是什么?

Jenkins 的优势包括:

在开发环境的早期阶段, 错误跟踪很容易。

提供大量的插件支持。

对代码的迭代改进。

构建失败会在集成阶段进行缓存。

对于每个代码提交更改, 都会生成一个自动生成报告通知。

为了将构建报告的成功或失败通知开发人员, 它与 LDAP 邮件服务器集成在一起。

实现持续集成的敏捷开发和测试驱动的开发。

通过简单的步骤, 即可自动完成 maven 发布项目。

八、如何确保你的项目构建不会在 Jenkins 中破裂?

1、首先, 使用所有单元测试在本地计算机上执行成功的全新安装。

2、检查所有代码更改。

3、与存储库同步, 以确保所有必需的配置和 POM 更改以及所有差异都被检入到存储库中。



九、如何将 Jenkins 从一台服务器移动到另一台服务器？

- 1、首先，复制相关的作业目录，然后将作业从 Jenkins 的一个安装滑动到另一个安装。
- 2、通过使用不同的名称克隆作业目录来复制现有作业。
- 3、通过重命名目录来重命名现有作业。

十、jenkins 手动重启的方式

- 1、(Jenkins_url)/ restart: 强制重启，而无需等待构建完成。
- 2、(Jenkins_url)/ safeRestart: 允许所有正在运行的构建完成。

十一、Maven 和 jenkins 分别是什么？

Maven 是一个构建工具，简而言之是 ant 的继承者。它有助于构建和版本控制。但是，Jenkins 是持续集成系统，其中 maven 用于构建。 Jenkins 可用于自动化部署过程。

十二、jenkins 常用插件？

ssh remote hosts 这个可以在远程服务器上面执行脚本。

Role Strategy Plugin 用来精细化管理权限。

SCM: 除 CVS 和 Subversion 外需要实现与源代码控制系统支持的插件。 3 L1 0# q2 R & _+ U3 B

Triggers: 事件监听并触发构建的插件。例如，URL 改变触发器将监控一个 URL；当地址内容发生改变，这个触发器就将执行一次作业。

Build tools: 实现额外构建工具的插件，如 MSBuild 和 Rake。如果您想在 Hudson 中构建非 Java 的软件时这些就特别有用。

Build wrappers: 通常涉及时执行在受控制的构建过程本身之前和之后事件的插件。例如，VMware 插件将在构建之前启动一个客户虚拟机，建立和然后在构建完成后关闭它。这在您可能需要访问 VM 以执行单元测试的情况下是非常有用的。

十三、你们公司常用的插件？

Maven 2 project: 管理工具。管理项目报告，生成站点，管理 JAR 文件等
HTML publisher
Copy artifact
Join
Green Balls