

Atlassian DevOps

什么是 DevOps?

DevOps 是一套实践、工具和文化，可以自动化和整合软件开发和IT团队之间的流程。它强调团队能力、跨团队沟通和协作以及技术自动化。

DevOps 一词是开发和操作这两个词的组合，反映了将这些领域整合到一个连续的过程。

DevOps 如何运作?

DevOps 团队包括开发人员和 IT 运营部门在整个产品生命周期内协同工作，以提高软件部署的速度和质量。这是一种新的工作方式，一种文化改革，对团队和他们工作的组织有重大影响。

在DevOps 模型下，开发和运营团队不再“孤立”。有时，这两个团队合并成一个团队，工程师在整个应用程序生命周期（从开发和测试到部署和运营）中工作，并拥有一系列多领域技能。

DevOps 团队使用工具来自动化和加速流程，这有助于提高可靠性。DevOps 工具链帮助团队处理重要的 DevOps 工作，包括持续集成、持续交付、自动化和协作。

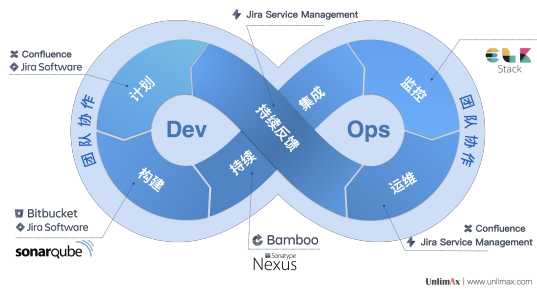
DevOps 有时应用于开发以外的团队。当安全团队采用 DevOps 方法时，安全性是开发过程中的积极和集成部分。这叫做 DevSecOps。

DevOps 生命周期

由于 DevOps 的连续性，可以以下面的图示表示，循环象征着在整个生命周期中需要不断协作和迭代改进。

本页面内容

- [DevOps 如何运作?](#)
- [DevOps 生命周期](#)
 - [计划](#)
 - [构建](#)
 - [持续集成](#)
 - [监控](#)
 - [运维](#)
 - [持续反馈](#)



DevOps 生命周期由六个阶段组成，代表开发（左侧）和运维（右侧）所需的流程、功能和工具。在每个阶段，团队都会进行协作和沟通，以保持一致性、速度和质量。

计划

DevOps 团队应该采取敏捷的方法来提高研发速度和质量。敏捷是项目管理和软件开发的一种迭代方法，可帮助团队将工作分别多次交付，以提高每次交付质量。

从计划开始就不断收集用户反馈，确定任务优先级。重要的是，每个人都可以分享和评论任何东西：想法、战略、目标、要求、路线图和文档等。

建议工具	功能
Jira Software 研发管理	● 路线图规划 ● backlog 梳理 ● 冲刺计划
Confluence 文档协作	● 需求管理及梳理 ● 设计文档 ● 可行性验证 ● 会议纪要

构建

Git 是一个开源的版本控制系统。它为分支、合并和重写存储库历史记录提供了出色的支持，这为开发构建过程带来了许多创新和强大的工作流程和工具。

建议工具		替代工具
Jira Software 研发管理	● 开发过程管理 ● 缺陷管理 ● 任务管理 ● 敏捷冲刺 ● 冲刺报告	
Xray Test Management for Jira 测试管理	● 测试用例管理 ● 测试计划管理 ● 测试报告 ● 需求、测试、缺陷覆盖报告 ● 自动化测试	
Bitbucket 代码管理	● 代码分支 ● 代码评审 ● 代码合并 ● 代码对比	Gitlab Git Integration for Jira
Confluence 文档协作	● 冲刺回顾 ● 技术讨论 ● 会议纪要	
SonarCube	● 代码规范检查	

持续集成

CI/CD允许团队频繁和可预测地发布优质产品，从源代码存储库到具有自动化工作流程的生产。团队可以频繁合并代码更改，部署功能标志，并整合端到端测试。

建议工具		替代工具
Jira Software 研发管理	● 版本发布	
Atlassian Bamboo	● 单元测试 ● 自动化测试 ● 代码打包 ● 产成品部署 ● 自定义串行/并行工作流程	Jekins Jenkins Integration for Jira Webhook to Jenkins for Bitbucket
Sonatype Nexus	● 私有 Maven 仓库管理	

监控

快速识别和解决影响产品正常运行时间、速度和功能的问题。自动通知您的团队更改、高风险操作或故障，以便您可以继续提供服务。

建议工具	
ELK stack	● 服务器运行状态报告 ● 应用日志收集分析

运维

管理向客户提供端到端的IT服务。这包括设计、实施、配置、部署和维护支持组织服务的所有IT基础设施所涉及的实践。

解锁DevOps团队之间协作的关键是确保他们查看相同的工作内容。报告事件时会发生什么？它们是否与软件问题相关联和可追溯？更改后，它们是否与产品发布相关联？

建议工具	
Jira Service Management 运维/客户服务	● 服务请求 ● 事件管理 ● 变更管理 ● 发布管理 ● 配置管理 ● 资产管理
Confluence 文档协作	● 知识库管理 ● 变更记录 ● 发布记录 ● 事件分析

持续反馈

DevOps团队应该评估每个版本并生成报告，以改善未来的版本。通过收集持续反馈，团队可以改进流程，并整合客户反馈来改进下一个版本。

建议工具	
Jira Service Management 运维/客户服务	● 服务请求 ● 收集反馈 ● 缺陷报告 ● 意见建议