

博客贴文

借助 IBM Cloud Pak for Application , 构建现代化、容器化的应用

基于容器的云原生开发不应该如此之难。IBM 的全新开源项目能帮助您更轻松
地设计、构建和部署应用。

作者 : Geoff Pirie

更新日期 : 2019 年 8 月 1 日 | 发布日期 : 2019 年 8 月 1 日

作为开发人员,您的目标是编写出色的代码,解决业务问题。效率对于开发人员来说至关重要,但是要想在团队里高效开展工作,光是脑袋里有想法还不够。您还需要适当的工具和技术,才能快速开展工作,构建最佳应用。

将新工具和框架集成到某种平台的挑战可能会分散您完成主要任务(即,解决业务难题)的精力。尽管 Kubernetes 成为了开发人员常用的平台,但是它本身并不是一个完整的开发平台。

您如何获得想要的基于容器的开发平台,将架构师、运营部门和开发人员的需求整合到一个位置?您可以自行构建一个这样的平台,但是您的团队可能目前尚不具备所需的时间、资源和知识。

而这正是 IBM 全新开源产品的用武之地。[IBM Cloud Pak for Application \(ICPA\)](#) 将基于 Kubernetes 的 Red Hat OpenShift 平台与开源项目 Kabanero 整合一体, Kabanero 支持企业更快速、更轻松的开发和部署面向 Kubernetes 和 Knative 的应用。

在这篇博文中，我们将详细探讨支持 ICPA 的基础开源软件，以及 ICPA 解决的一些常见的开发问题。

详论问题

通常，团队采用 Kubernetes 时，目标是简化开发和部署，获得容器化的所有优势。但是，他们没有考虑容器环境所带来的复杂性。Kubernetes 非常适合用于在服务器基础架构上管理容器布置，将工作路由至这些容器，并在出现故障后恢复容器，但是它无法为尝试构建云原生应用的开发人员和架构师提供全面支持。

为开发团队构建平台是一项重要的任务。从选择并集成正确的开源项目集合，到选择 IDE 和构建工具、管道、源控制、安全性扫描和测试，您需要考虑方方面面的事务。当您添加更新和安全补丁时，这又变成了一项全职开发工作。我们的目标是解决这些问题，降低复杂性，从而提供团队中所有开发人员的工作效率。可是怎么才能做到呢？

首先，我们应该认识到某种程度的一致性是好事。一致的工具能帮助团队高效地构建应用，分享知识。通过利用一致的工具，您可以构建每个人都能理解的一致的交付流程，从而减少人为失误。成功需要自动化。

加强企业内不同部门之间以及团队内部的协作是企业提高生产力的根本。架构师和开发人员需要采用一致的机制来捕获技术决策，然后与更多团队成员分享这些决策。

对于很多团队来说，使用容器就是满足这一需求的其中一步，容器将与 Kubernetes 配套使用。但是，容器提供的自由和灵活性意味着，您可能会因采用太多容器而告终。选择太多会衍生出复杂性、不一致和效率低下等问题。

毫不犹豫地

将原型应用于生产阶段

[Kabanero](#) 简化了构建容器应用的任务，以便将应用部署到混合云环境中。Kabanero 使用 [OKD](#)(Red Hat OpenShift 的上游项目) ,在 Kubernetes 上为您提供一个经过验证的强大基础。在该基础内，Kabanero 增加了关键的基础架构，以便提供持续集成/持续交付 (CI/CD) 管道与重要的服务(比如日志记录、监控、服务网格)，以及 Knative 无服务器功能。到目前为止还算不错。但是，是什么让 Kabanero 与众不同？

我们将在稍后更详细地探讨特定的开发工具，但是首先我们需要了解 Kabanero 如何提供通用方法，以便利用 IDE 扩展组件运用的一组一致的容器组件，定义、管理和构建应用。

Kabanero 引入了“集合”的理念。集合能够将多种语言、运行时和框架选择与基础架构绑定整合到一起，后者允许 Kabanero 插入到 Kabanero 基础之中。集合提供了一致的方法，用于捕获开发人员和架构师制定的协作式决策。一组预定义的集合就能消除集成基础架构时的猜测工作。

[Appsody](#) 堆栈是集合的一个关键组件。

Appsody 及其相关的命令行工具 (CLI) 为您构建容器应用提供基础构件。Appsody 将运行时和框架整合在容器内，同时还提供通用代码和服务。比如，Java MicroProfile 堆栈包含预定义的运行状况指标和端点，这样，利用堆栈构建的所有应用都有现成可用的通用功能。

开发人员可以迅速采用堆栈，着手构建新的云原生应用。Appsody 利用预定义的堆栈，不断为您的应用重构容器映像。最终，您的应用将始终构建于最新的堆栈之上，并且自动做好在容器内测试的准备，就像在生产环境中运行应用一样。

- 立即[试用 Kabanero](#)。
- 查看 [Kabanero GitHub 报告](#)。

利用管道自动化和 IDE 集成，更快速地构建应用

Kabanero 基础架构集成在 GitHub 中，借此，代码提交能够触发自动化 CI/CD 管道。通过利用 Kabanero 集合中包含的预定义集成点，管道基础架构能够对照最新版本的 AppSody 堆栈，编写应用代码。

由于应用使用的是通用堆栈，因此，应用在生产环境中运行时像其在开发环境中运行一样，对于这一点，开发人员信心十足。这样，您就能在代码投入生产后提高一致性，降低出错的风险。如果您处于待命状态，又不想半夜被吵醒的话，这一点对于您来说非常重要。

Kabanero 不只是基于 Kubernetes 的基础架构。它还是一个集成了热门 IDE（如 Eclipse 和 VSCode）的开发工具。[Eclipse Codewind 项目](#)旨在帮助开发人员从一开始就在容器内构建云原生微服务和基于功能的应用。该项目提供理解 AppSody 堆栈的扩展组件，这样，您就可以利用预定义的堆栈着手构建应用，并从 Codewind 在 IDE 内构建的自动化容器中受益。借助 Eclipse Codewind 内的性能仪表盘，您可以立即在开发周期了解新应用或服务的性能特点。

从开发入手，在需要时增加支持

Kabanero、AppSody、Codewind 和构建 Kabanero 基础的其他基础架构组件都是开源项目。通过结合利用这些技术，团队可以更轻松、更便捷地聚焦业务问题，而非基础架构决策。

在适当的时候，如果您的团队需要支持，IBM Cloud Pak for Application 将助您一臂之力。IBM Cloud Pak for Application 能够集成、保障和认证这些开源项目，为您提供单一解决方案，帮助您将应用重构为云原生模式。借助 ICPA，

您可以将现有应用迁移至更有效的基于 Kubernetes 的运营模式，并将变更降至最低。您的开发人员则可以专注于编写出色的代码，满足真正的业务需求。

了解更多

- [IBM Cloud Pak for Application](#)
- [Red Hat OpenShift on IBM Cloud](#)
- [Kabanero](#)
- [Appsody](#)
- [Eclipse Codewind](#)