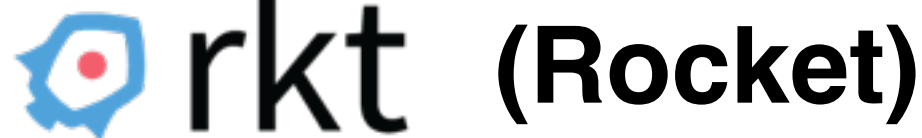




Docker 改变了什么？

周悦秋@好雨

Docker 并不孤单

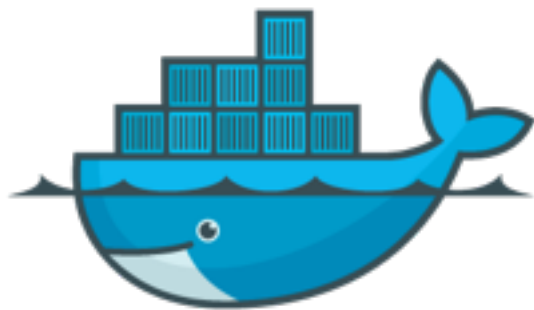


**Cloud Foundry
Garden**

什么是容器



非虚拟化

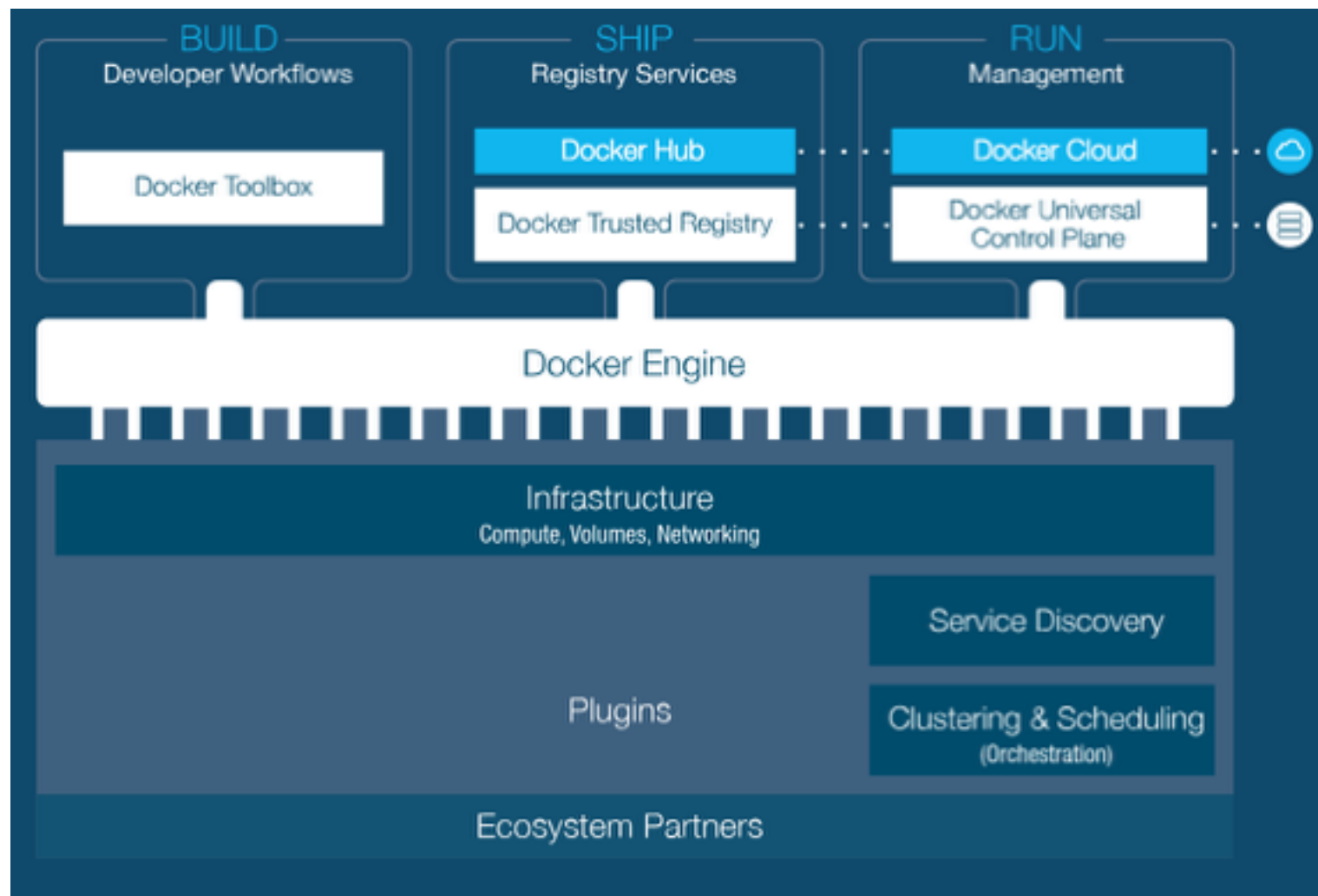


封装



高效

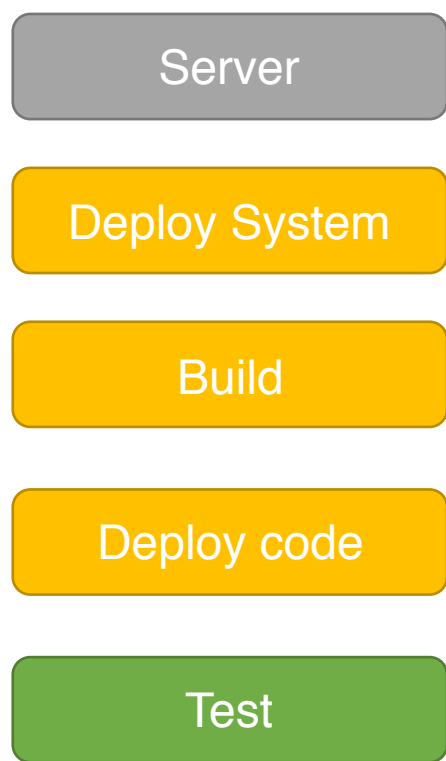
Docker 解决方案



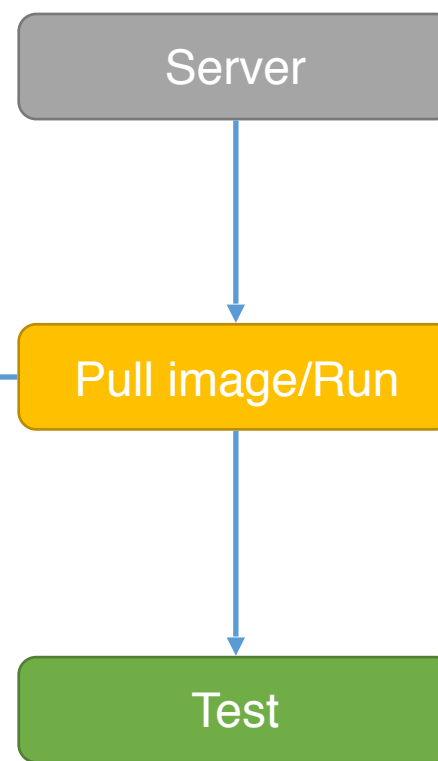
为产品的整个生命周期
提供一整套的解决方案。

Build

Build 传统方式



Build Docker方式



开发测试阶段

- 带来了封装的运行环境

彻底解决了环境的依赖问题

- 带来了全新&灵活的镜像制作方式 --- Dockerfile

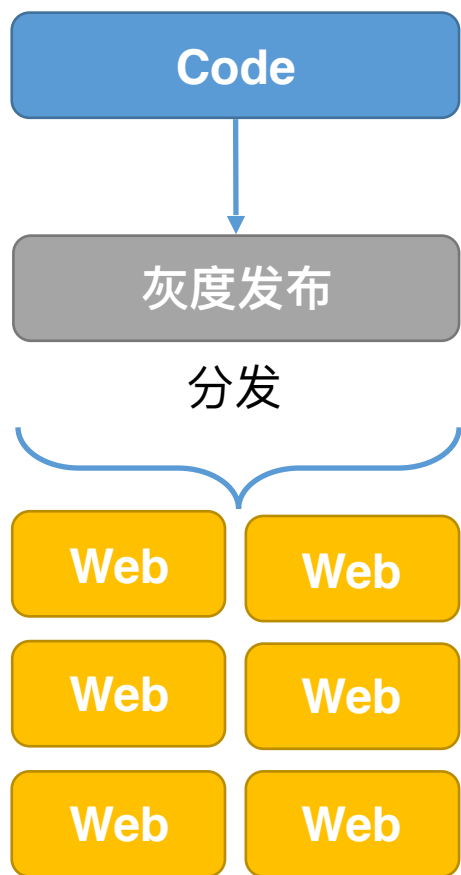
- 带来了全新的镜像管理方式 --- Docker Registry

- 改变了开发环境的部署方式

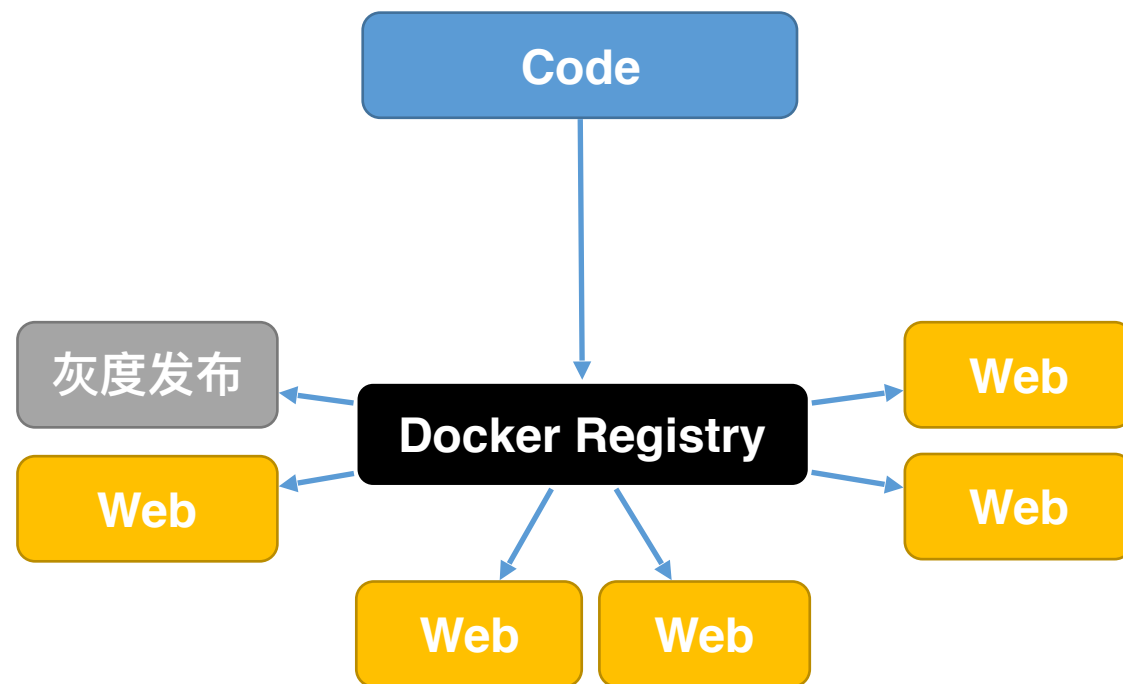
拉 (Pull) 来即用环境, 可以瞬间启动几十上百相同的开发环境

Ship

传统方式



Docker方式



发布阶段

改变了发布方式

一次推送 (Push) , 到处拉取 (Pull)

改变了代码回滚方式

拉取一次旧的镜像即可

彻底解决环境一致性问题

环境没变, 变的是代码和运行的位置

Run

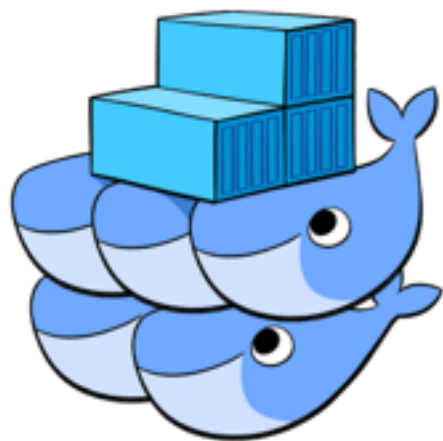
不同场景下的Run

测试环境



Docker Toolbox

私有环境



Docker Swarm

Docker公有云



Docker Cloud

业界分布式解决方案



kubernetes
by Google

(K8S)



MESOS



Marathon

运行阶段

原生支持分布式系统

封装整合、快速分发，快速启动

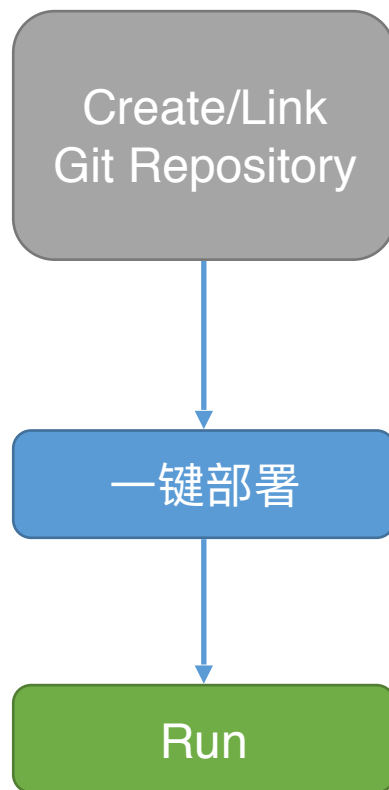
改变了分发方式

一次推送 (Push) ，到处拉取 (Pull)

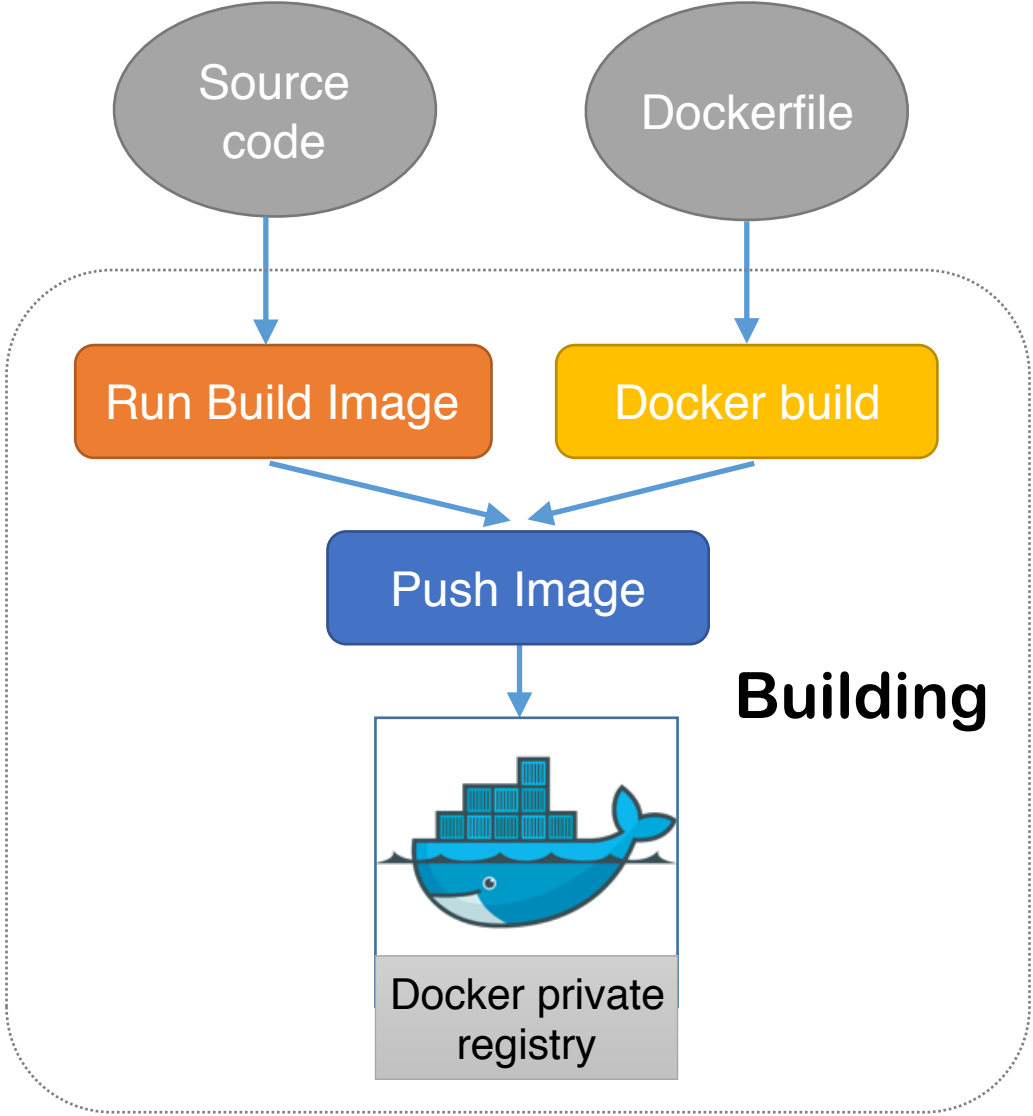
Docker on Goodrain

Docker 只是技术，业务产生价值。

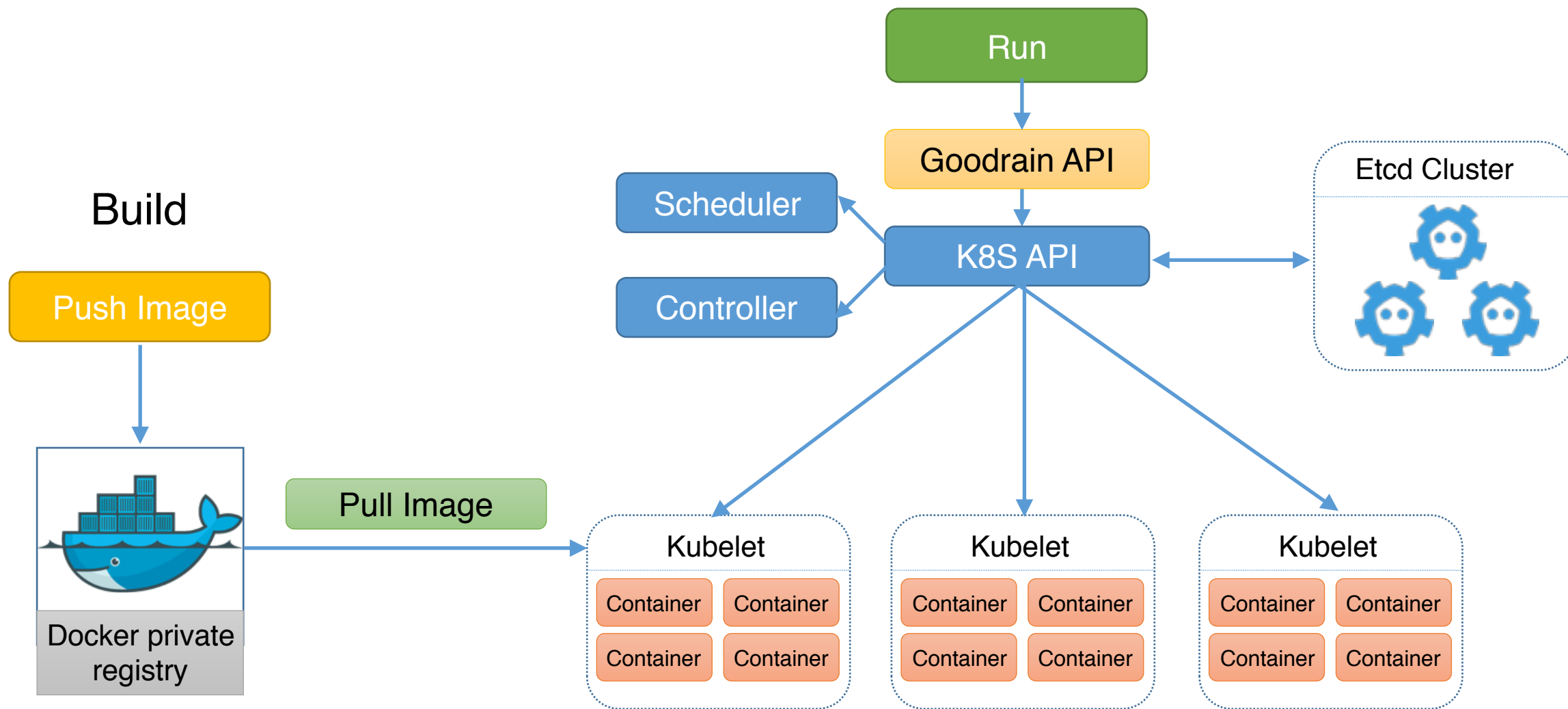
Build & Run 产品表现



Build 内部流程



Ship & Run 内部流程



总结

Docker 不是技术的创新，而是一种新的思维方式

- 封装：代码与环境合体 —— 应用
- 发布：像发布代码一样发布应用
- 分发：一次构建，到处运行

谢谢