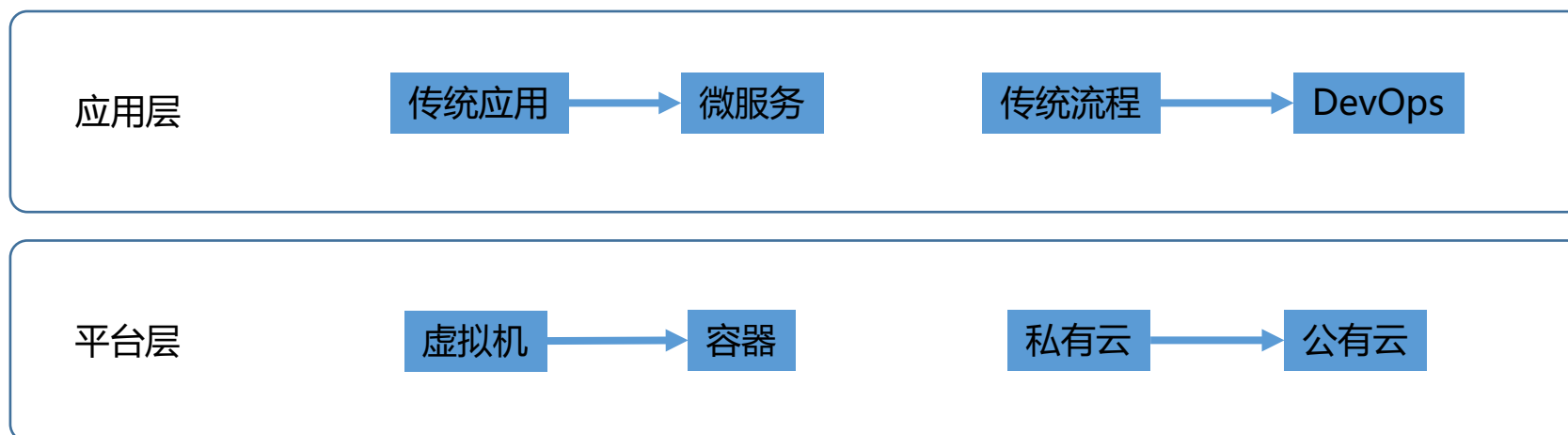
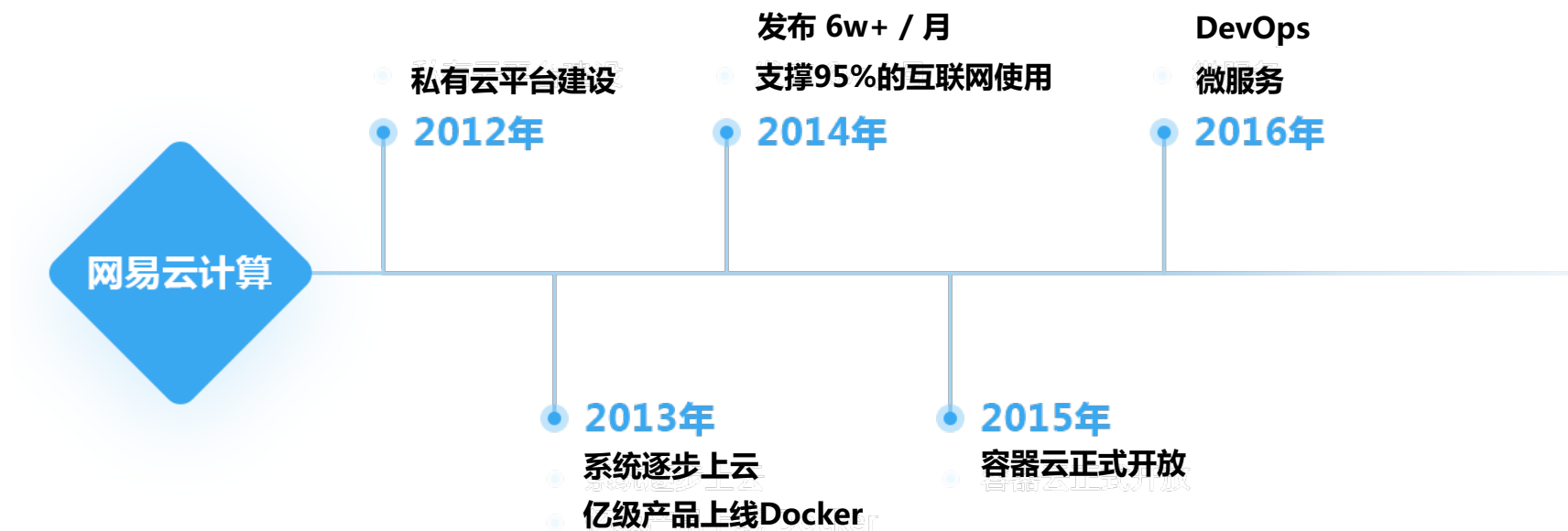


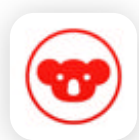
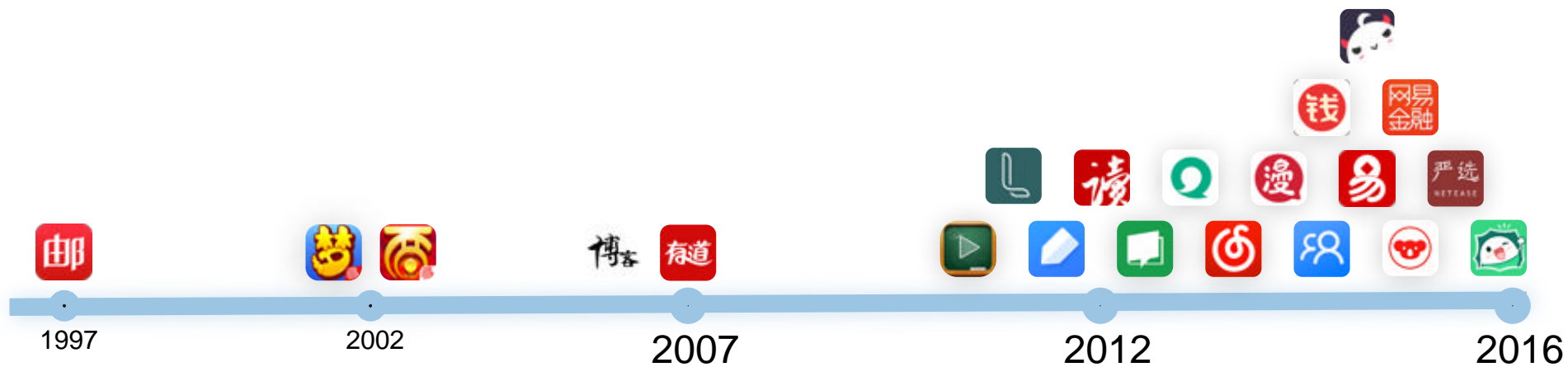
使用容器应对业务快速迭代和大规模部署 的运维挑战



网易云历程



网易云上的大规模应用



6.18

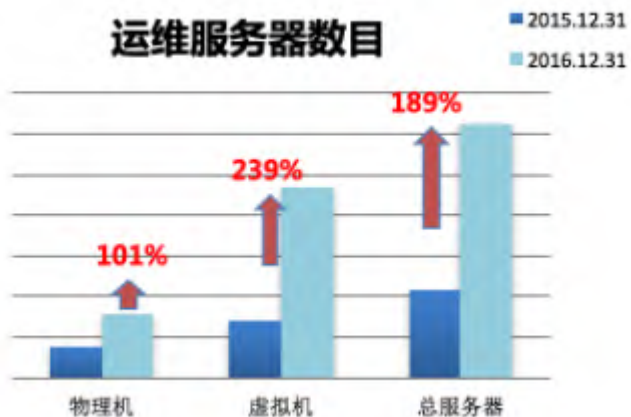
11.11



200,000,000用户

运维面临的挑战

网易云音乐、考拉海购等产品业务的迅猛发展
服务器规模和工单量快速增长



- 大量新的部署
- 节点规模更大



- 部署新版本
- 日常配置管理
- 排错

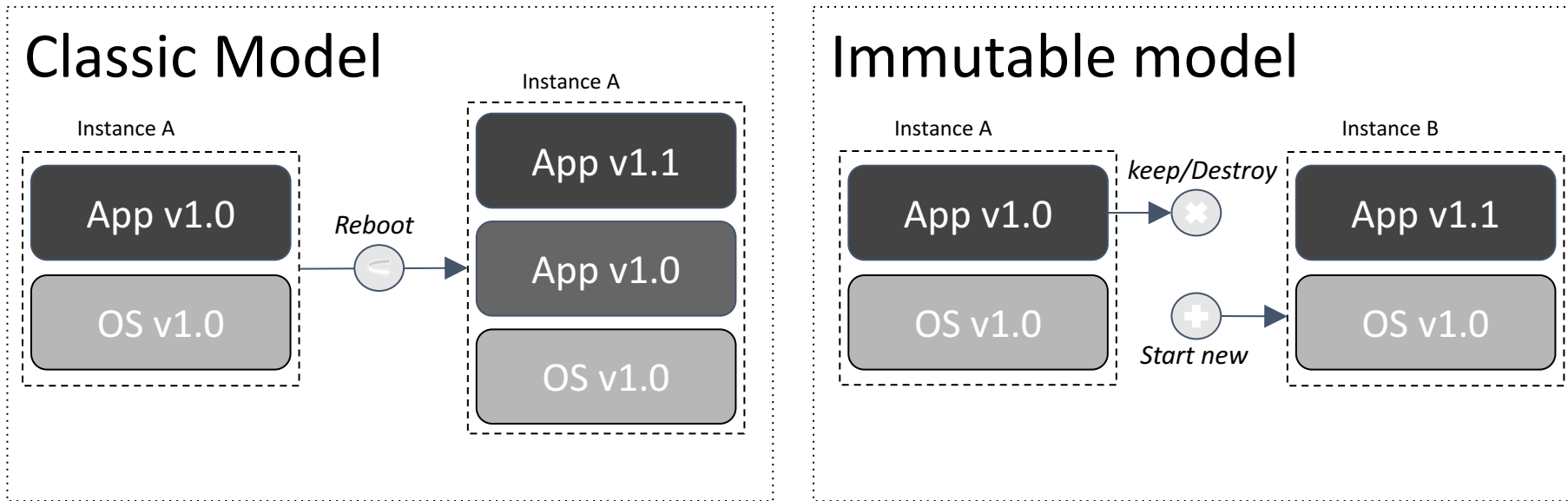
传统解决方法的问题

- 自动化工具
 - 可能节点无法连接
 - 可能操作过程中无法连接或操作失败
- 配置管理工具
 - 只能处理已定义的规则
 - 更新软件版本而不中断服务
 - 更新OS kernel
 - 将新环境配置到期望状态，但对更改的支持仍旧不够好



不可变基础架构

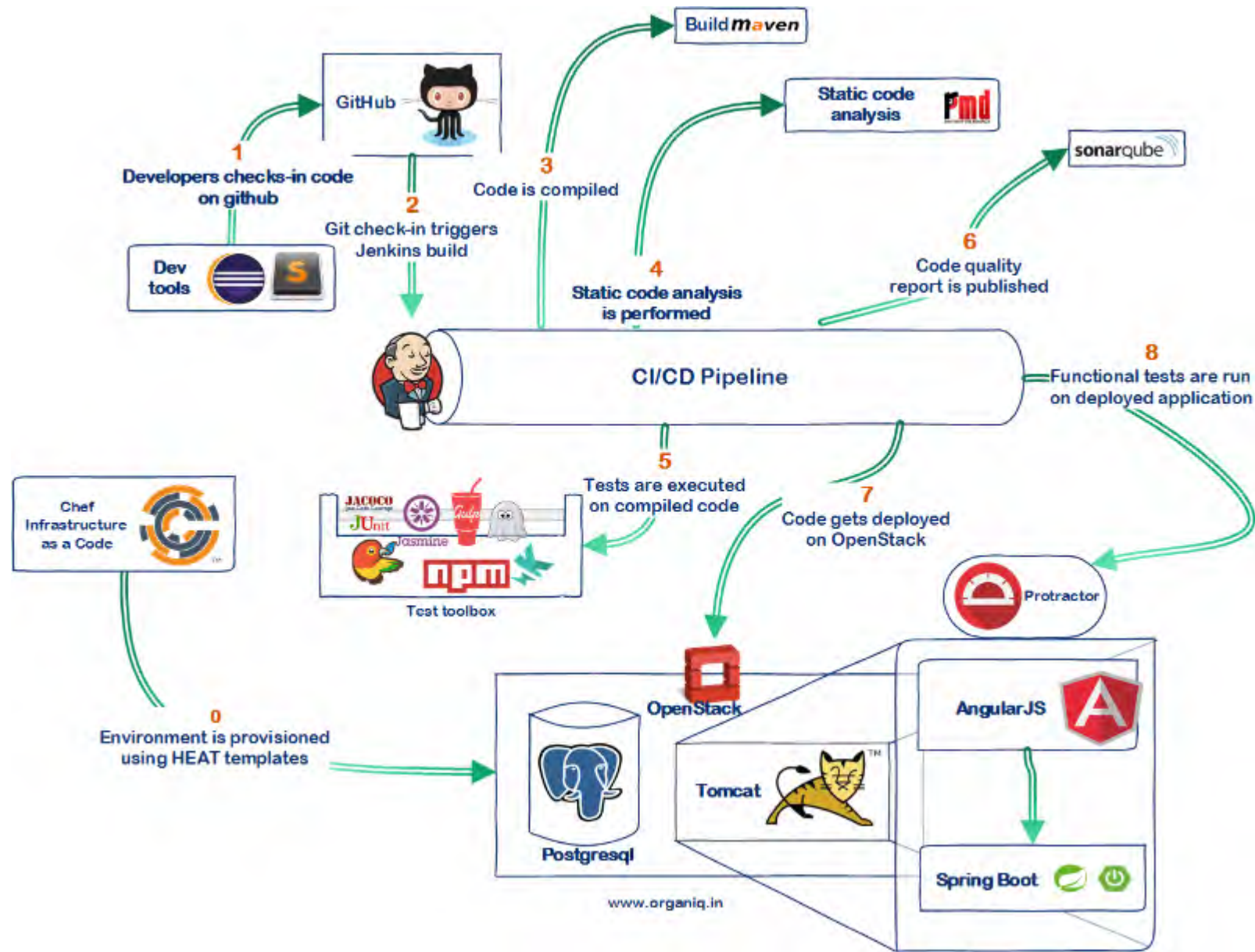
当需要对系统进行更改时，重新创建新节点，而不是更改现有节点



并不意味着整个系统永远不变
并不要求一个完全的无状态应用，而是适合有少部分有状态的分布式应用

相比传统模式的变化

- 把计算节点视为可替换的、可丢弃的
- 把原来在部署以后的配置、更新工作提前到构建阶段
- 变更可以做版本控制



当配置管理工具碰到 openssl漏洞 (1/2)

```
package { "openssl": ensure=> "installed" }
```

发现openssl漏洞

```
package { "openssl": ensure=> "1.0.1g" }
```

发现问题，停止使用新版本，回滚

```
package { "openssl": ensure=> "installed" }
```

已经是安装过的状态，不能回滚

```
package { "openssl": ensure=> "1.0.1f" }
```

这下应该可以了吧

当配置管理工具碰到 openssl漏洞 (2/2)

repo里没有这个包

好吧，我就想装个更旧的版本

repo里很可能更没有这个包

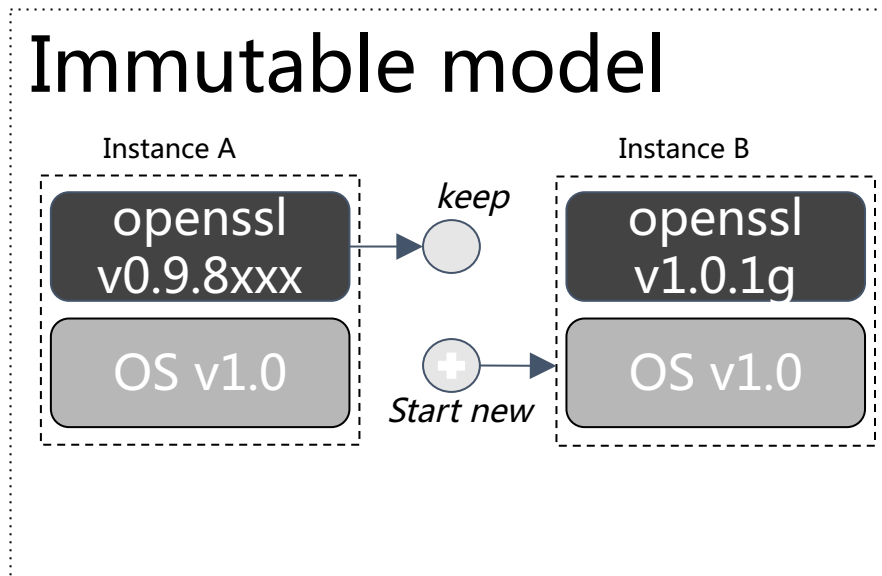
呃...实际上我们用的是 0.9.8xxx

安装1.0.1g的时候把所有相关依赖都升级了，回到旧版本变得非常麻烦

--Jérôme Petazzoni “Immutable infrastructure with Docker and containers”

当采用不可变基础设施

可以立即回滚-旧版本的openssl节点仍然保留着



为什么采用不可变基础架构

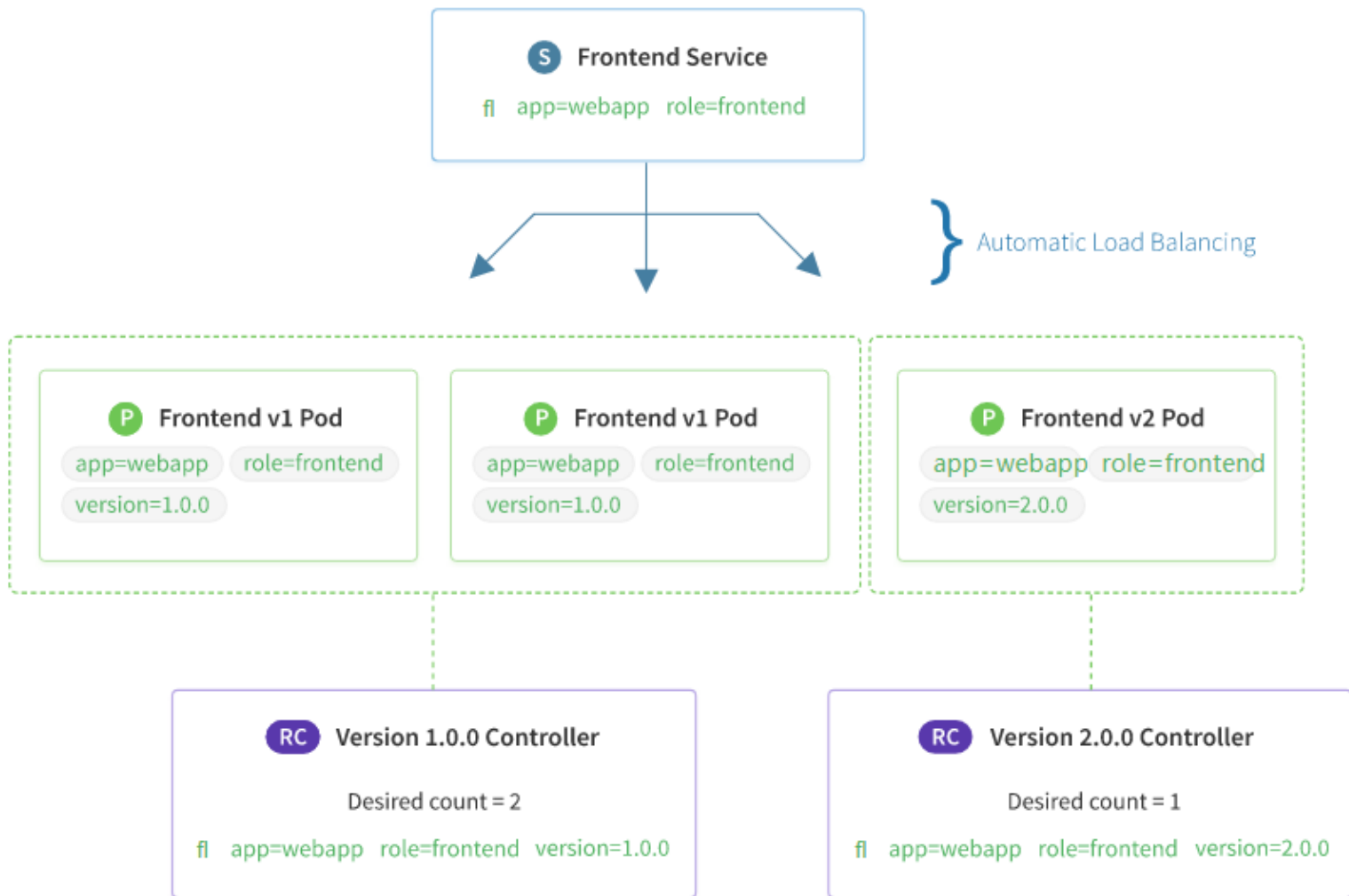
- 总是从干净状态开始，不会有遗留问题
- 开发、生产环境一致
- 容易回滚
- 尽早发现问题
- 简化变更管理，不必手工记录所有做过的操作
- 减少复杂状况的排错时间，部署新实例替换旧的比维护、修复有问题的实例更简单
- 部署更新成为例行工作并被自动化

为什么选择容器

- 容器本身具有不可变特性
- 支持版本管理- 便于实现everything as code
- 跨平台
- 允许环境变量注入，减少对配置管理工具的依赖
- 轻量、快速



结合编排工具进行滚动升级



```
kubectl rolling-update frontend-v1 -f frontend-v2.json
```

例外情况

小的修改变得麻烦，比如要修改一行css

- 传统方式：手动修改、验证、ok
几分钟

- 容器/不可变基础设施：手动修改、验证、然后...
 - 创建新镜像
 - 创建新节点
 - 切换到新节点数十分钟
一定时间后销毁旧节点

例外情况

小的修改变得麻烦，比如要修改一行css

- 传统方式：手动修改、验证、ok
几分钟

- 容器/不可变基础设施：手动修改、验证、然后...
 - 创建新镜像
 - 创建新节点
 - 切换到新节点数十分钟
一定时间后销毁旧节点

解决办法：自动化验证完成后的步骤

不可变基础设施的前提

- 自动化

- 创建、销毁计算节点
- 编排调度
- 服务发现
- 服务注册
- 配置管理
- 监控告警
- 健康检查
- 流量切换
- 日志收集

- 流程

- 组织



需要开发工作量来拼接各种工具

有状态应用和数据库如何处理

有状态应用容器化

有状态应用服务器 — 存储用户session。

- 解决办法：
1. 在负载均衡器上配置session affinity确保用户总是被分配到同一个容器上
 2. 把session存储在外部系统上，如Redis

常见问题：

1. 使用配置文件改为使用环境变量
2. 将log输出到STDOUT/STDERR

数据库

- 解决办法：
1. 使用云服务商的RDS、NoSQL等服务
 2. 挂载外部存储存放数据库文件和日志等需要持久化的内容

一些经验

- 不经常更改的部分放到base image中，减少拉取镜像的时间
- 收集日志并集中放置
- 充分利用云服务



网易云

共创云上精彩世界



www.163yun.com