

WOTA

51CTO

World Of Tech 2017

全球架构与运维技术峰会

2017年4月14日-15日 北京富力万丽酒店

ARCHITECTURE



出品人及主持人：

**李庆丰** 新浪微博研发中心  
研发总监

---

容器技术实践

# 云计算使用误区及容器云实践

刘超，网易云首席解决方案架构师



**刘超**

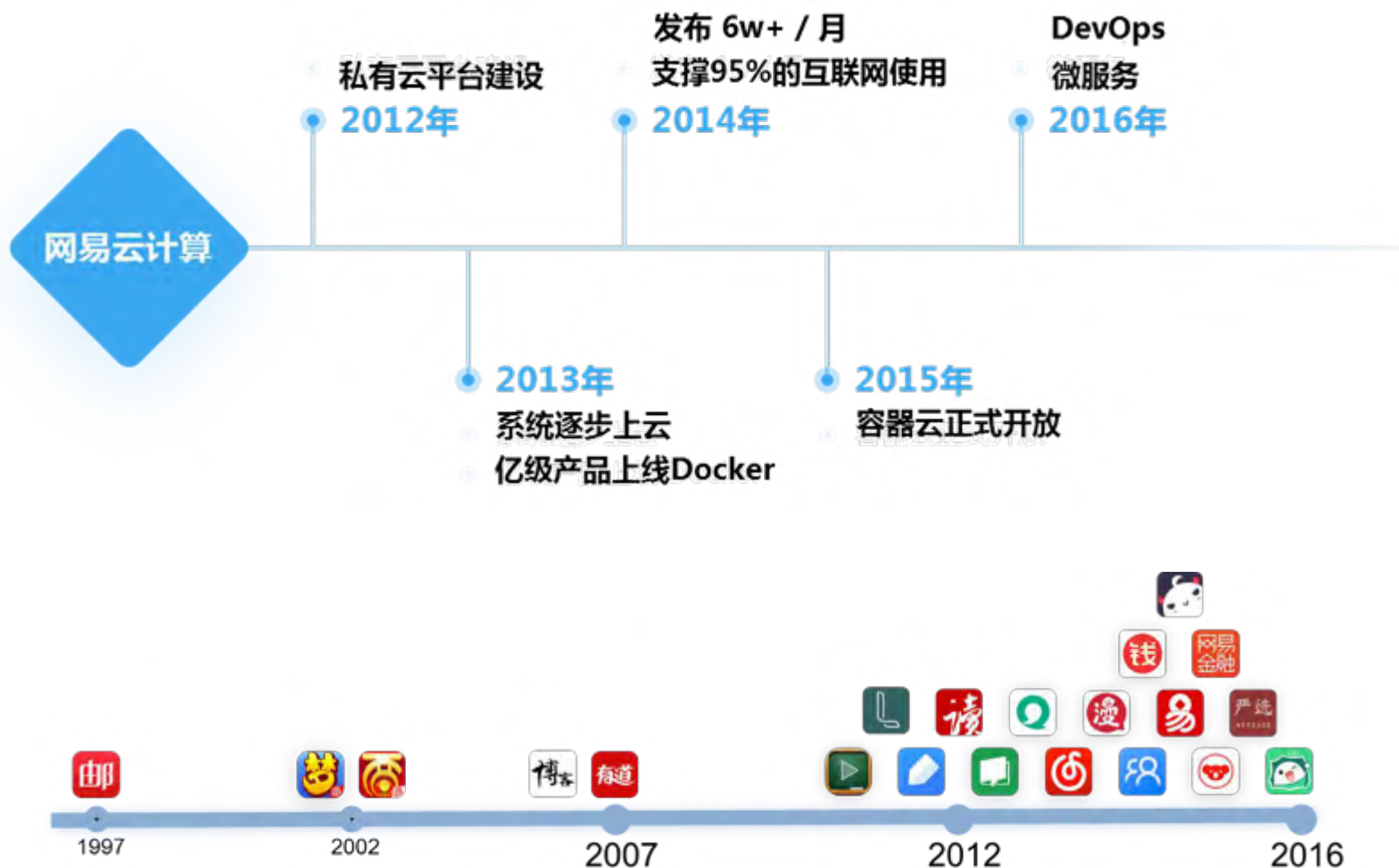
网易云

首席解决方案架构师

**分享主题：**

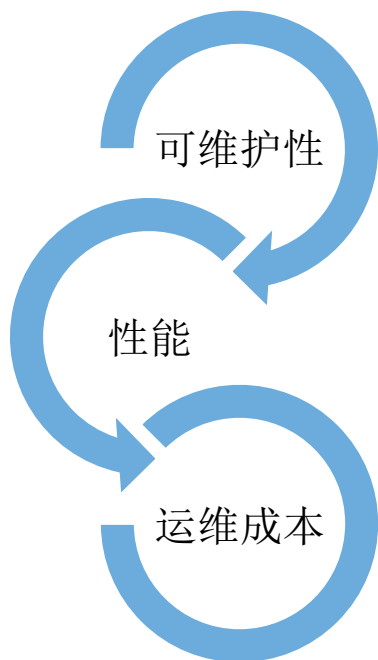
网易容器云实践与云计算的那些坑

# 网易云的历程





# 您享受到了云计算的优势了吗？

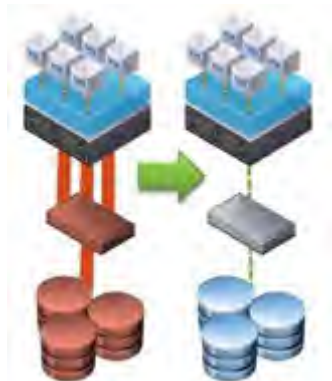


- 原来该写的脚本少写了么
- 原来该维护的表格少维护了么？
- 当访问量增加的时候，能够快速的扩展么？
- 需要的运维人员是少了还是多了，是不是更贵了，更难招了？

# 云计算的庞氏骗局？



- 稳定，性能好
- 可远程运维
- 仅需Linux系统知识即可运维



- 隔离性，使用率
- 传统虚拟化软件贵
- 会用虚拟化软件的，有证的人也贵



- 开源无绑定，兼容性
- 开源不免费，还要去适配，定制一过头，成本更加贵
- 不光会用，还要会开发，最好是core



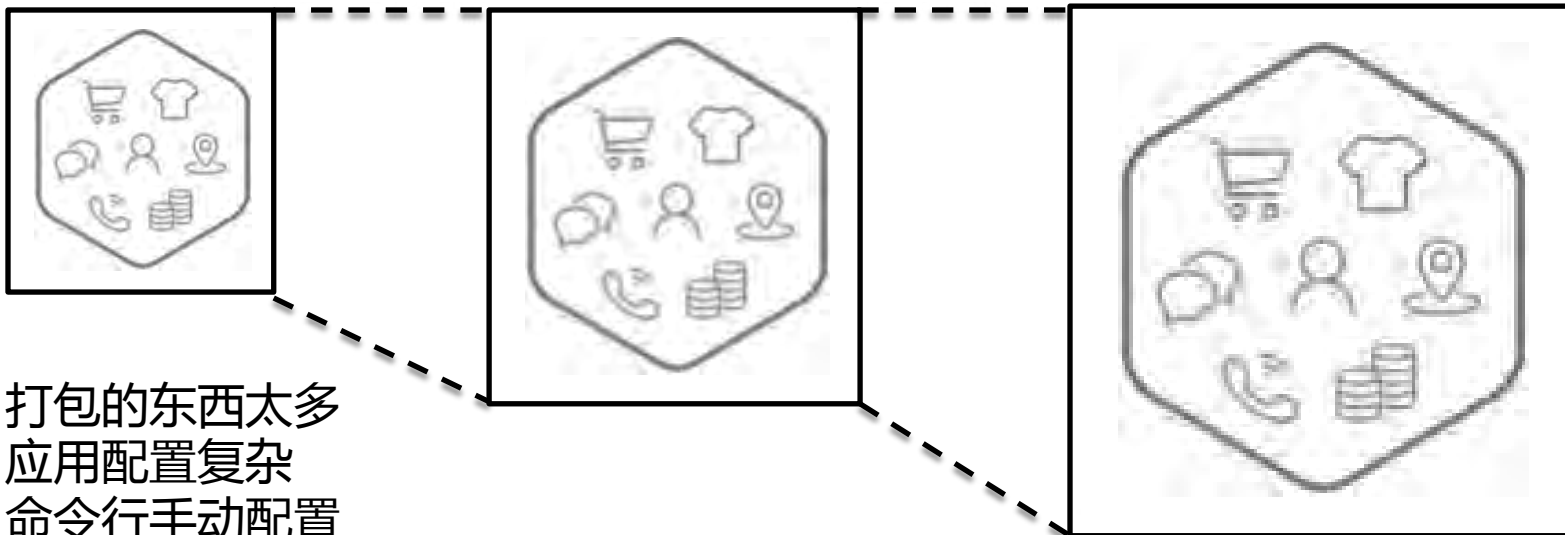
- 轻量级，易迁移，持续集成，DevOps
- 看起来简单，没有免费餐，不是偷了懒，就是提前赶，一旦想商用，全都不能免
- 全栈工程师不好找啊

# 我是不是用了假的云计算？



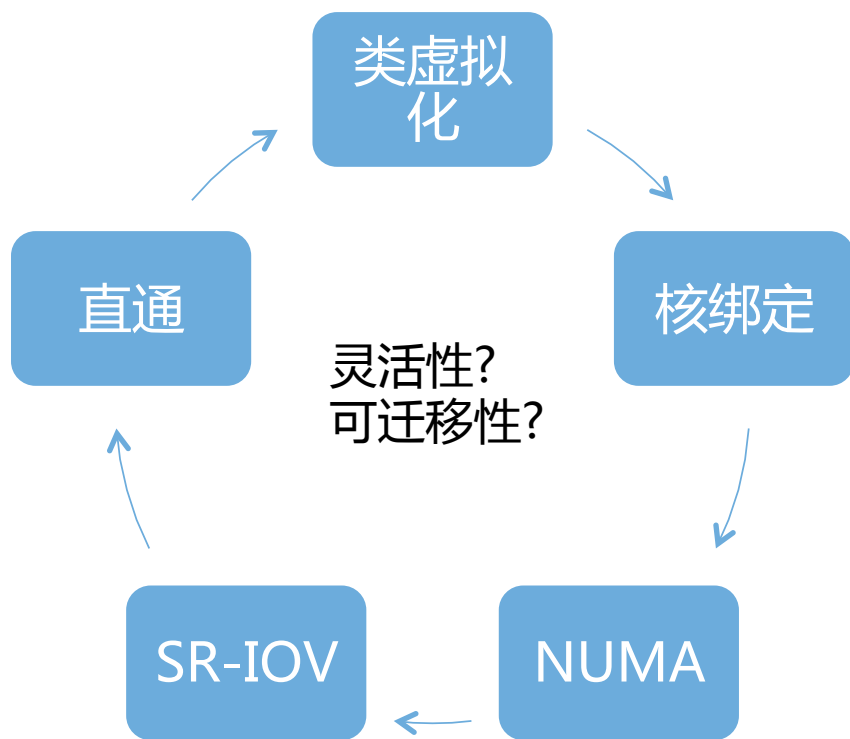


# 误区一：传统单体应用不加修改即进行虚拟机或者容器部署



- 打包的东西太多
- 应用配置复杂
- 命令行手动配置
- 应用启动速度很慢
- 无法横向扩展

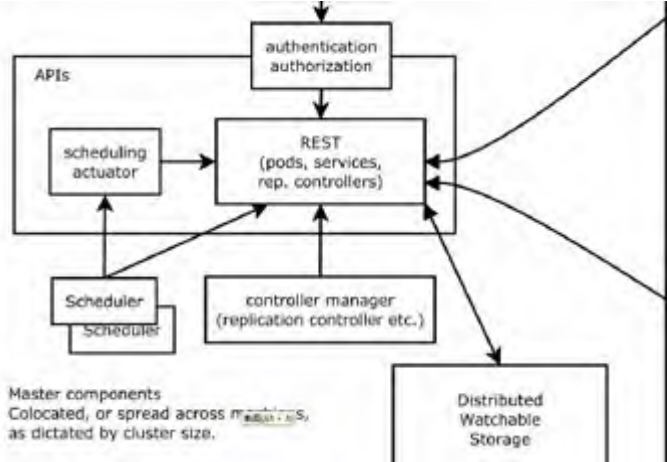
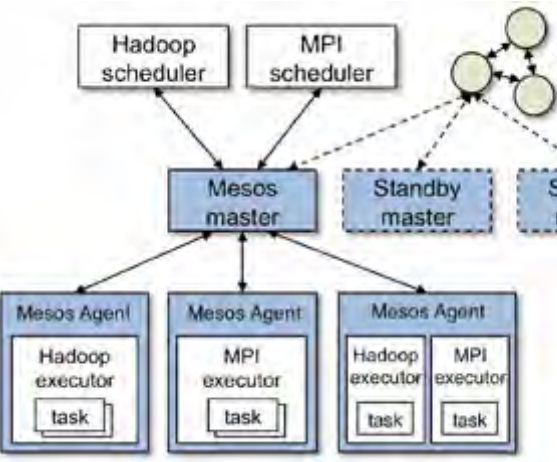
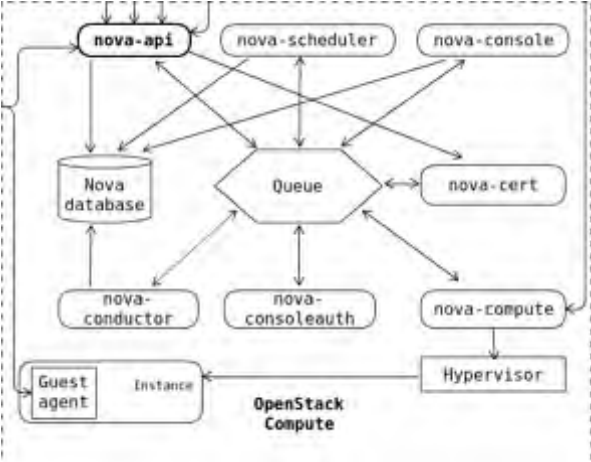
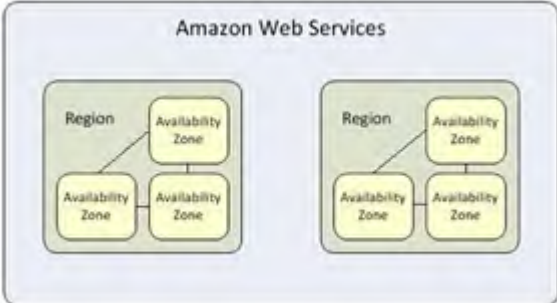
# 误区二：不想修改应用而期望虚拟化层的技术改进达到又要马儿好又要马儿不吃草的结果



横向扩展能力抵销虚拟化性能损耗

# 误区三：期望对虚拟机进行细粒度的调度，感知物理机和机架

ID	名称	规格	状态	创建时间	更新时间	所属区域	所属可用区	所属VPC
i-1234567890	实例名称	规格	运行中	2023-01-01 10:00:00	2023-01-01 10:00:00	cn-hangzhou	cn-hangzhou-f	vpc-12345678
i-0987654321	实例名称	规格	运行中	2023-01-01 10:00:00	2023-01-01 10:00:00	cn-hangzhou	cn-hangzhou-f	vpc-12345678
i-1111111111	实例名称	规格	运行中	2023-01-01 10:00:00	2023-01-01 10:00:00	cn-hangzhou	cn-hangzhou-f	vpc-12345678
i-2222222222	实例名称	规格	运行中	2023-01-01 10:00:00	2023-01-01 10:00:00	cn-hangzhou	cn-hangzhou-f	vpc-12345678
i-3333333333	实例名称	规格	运行中	2023-01-01 10:00:00	2023-01-01 10:00:00	cn-hangzhou	cn-hangzhou-f	vpc-12345678
i-4444444444	实例名称	规格	运行中	2023-01-01 10:00:00	2023-01-01 10:00:00	cn-hangzhou	cn-hangzhou-f	vpc-12345678
i-5555555555	实例名称	规格	运行中	2023-01-01 10:00:00	2023-01-01 10:00:00	cn-hangzhou	cn-hangzhou-f	vpc-12345678
i-6666666666	实例名称	规格	运行中	2023-01-01 10:00:00	2023-01-01 10:00:00	cn-hangzhou	cn-hangzhou-f	vpc-12345678
i-7777777777	实例名称	规格	运行中	2023-01-01 10:00:00	2023-01-01 10:00:00	cn-hangzhou	cn-hangzhou-f	vpc-12345678
i-8888888888	实例名称	规格	运行中	2023-01-01 10:00:00	2023-01-01 10:00:00	cn-hangzhou	cn-hangzhou-f	vpc-12345678
i-9999999999	实例名称	规格	运行中	2023-01-01 10:00:00	2023-01-01 10:00:00	cn-hangzhou	cn-hangzhou-f	vpc-12345678
i-0000000000	实例名称	规格	运行中	2023-01-01 10:00:00	2023-01-01 10:00:00	cn-hangzhou	cn-hangzhou-f	vpc-12345678



Master components Colocated, or spread across  $m$  nodes, as dictated by cluster size.

Distributed Watchable Storage

# 误区四：喜欢指定所有的机器的规格， 然后一次性进行包年包月购买

Reserved  
Instances



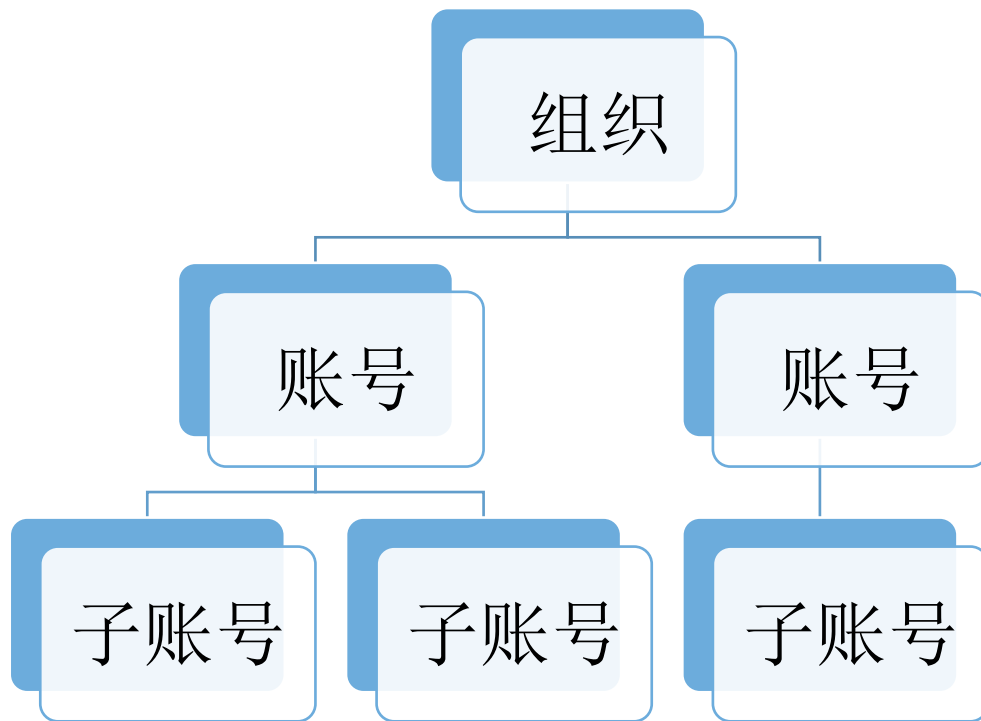
Instances



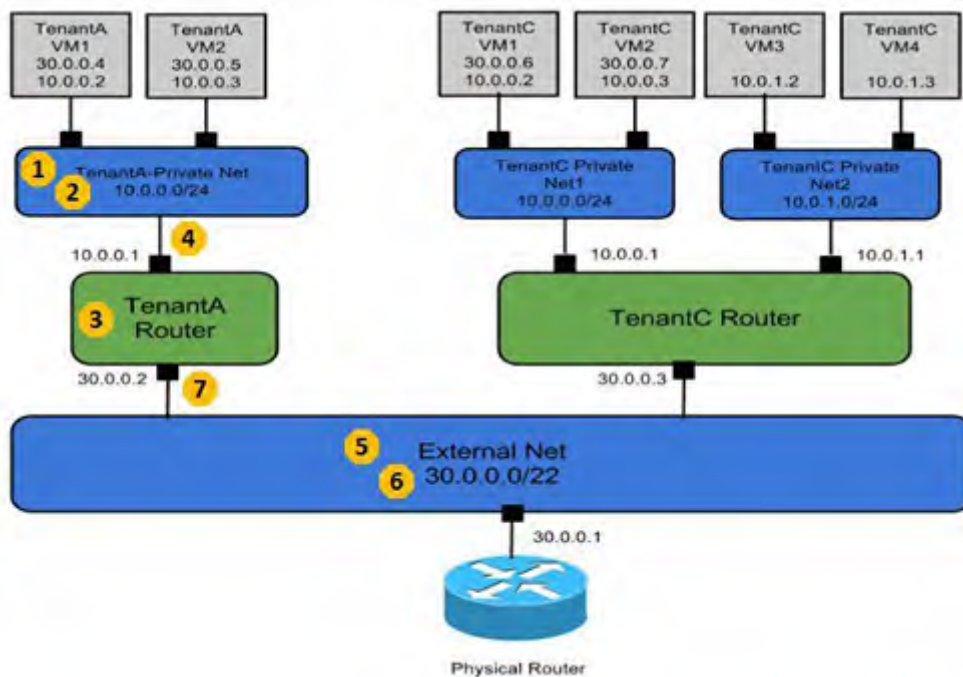
资源包 4核8G



# 误区五：全公司共用一个账号

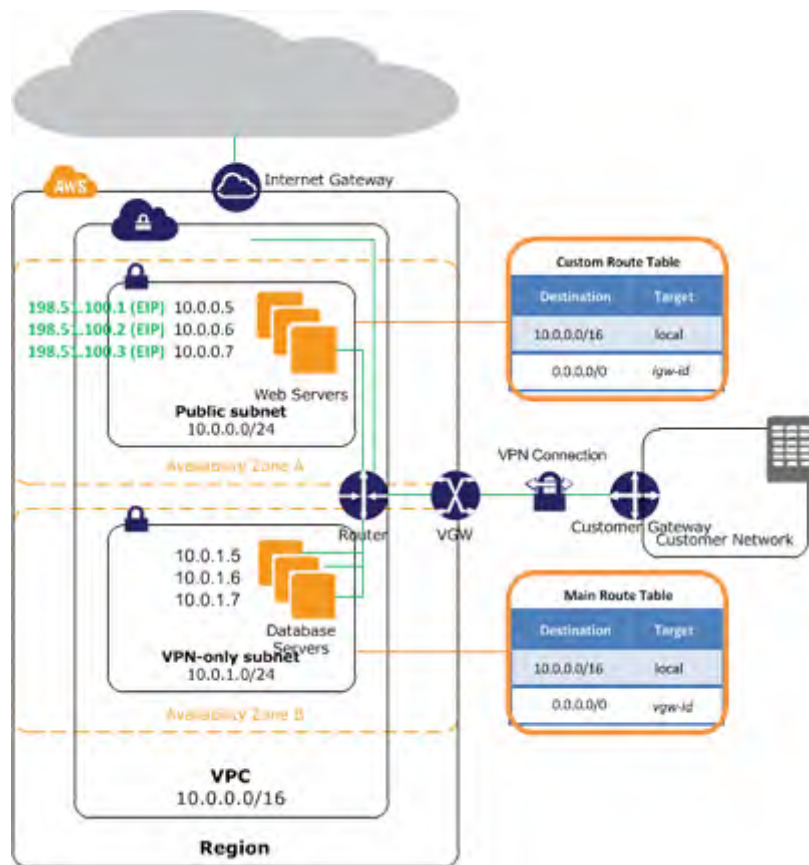


# 误区六：不规划和正确使用VPC进行网络隔离





# 误区七：所有的机器都带公网IP地址，并且使用用户名密码登陆



# 误区八：期望完全由基础设施层解决应用层的高可用问题

## Fault Tolerant

- 几乎实时
- 了解CPU, 内存, 硬盘

## 双活

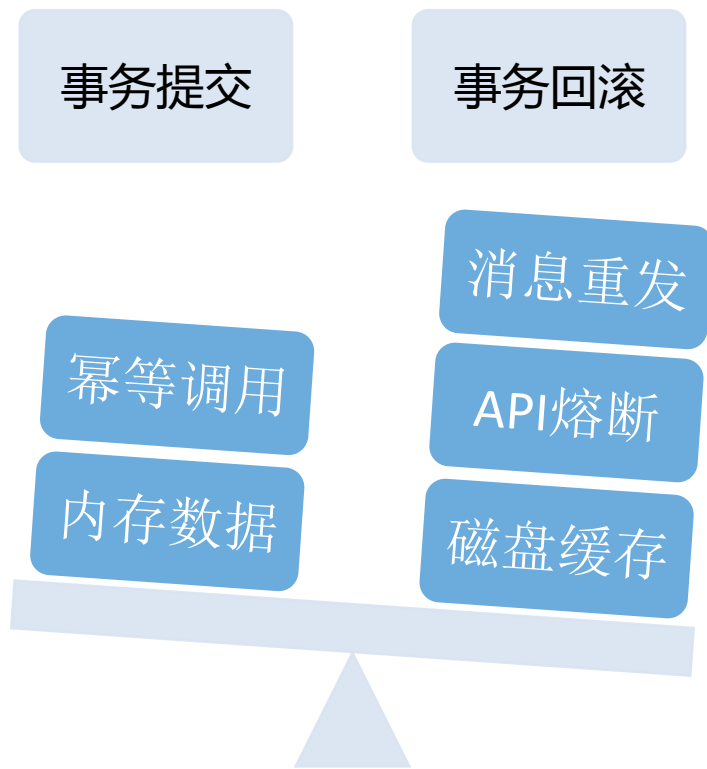
- 十几分钟
- 了解共享存储

## High Availability

- 秒到分钟级
- 了解共享存储

## 容灾

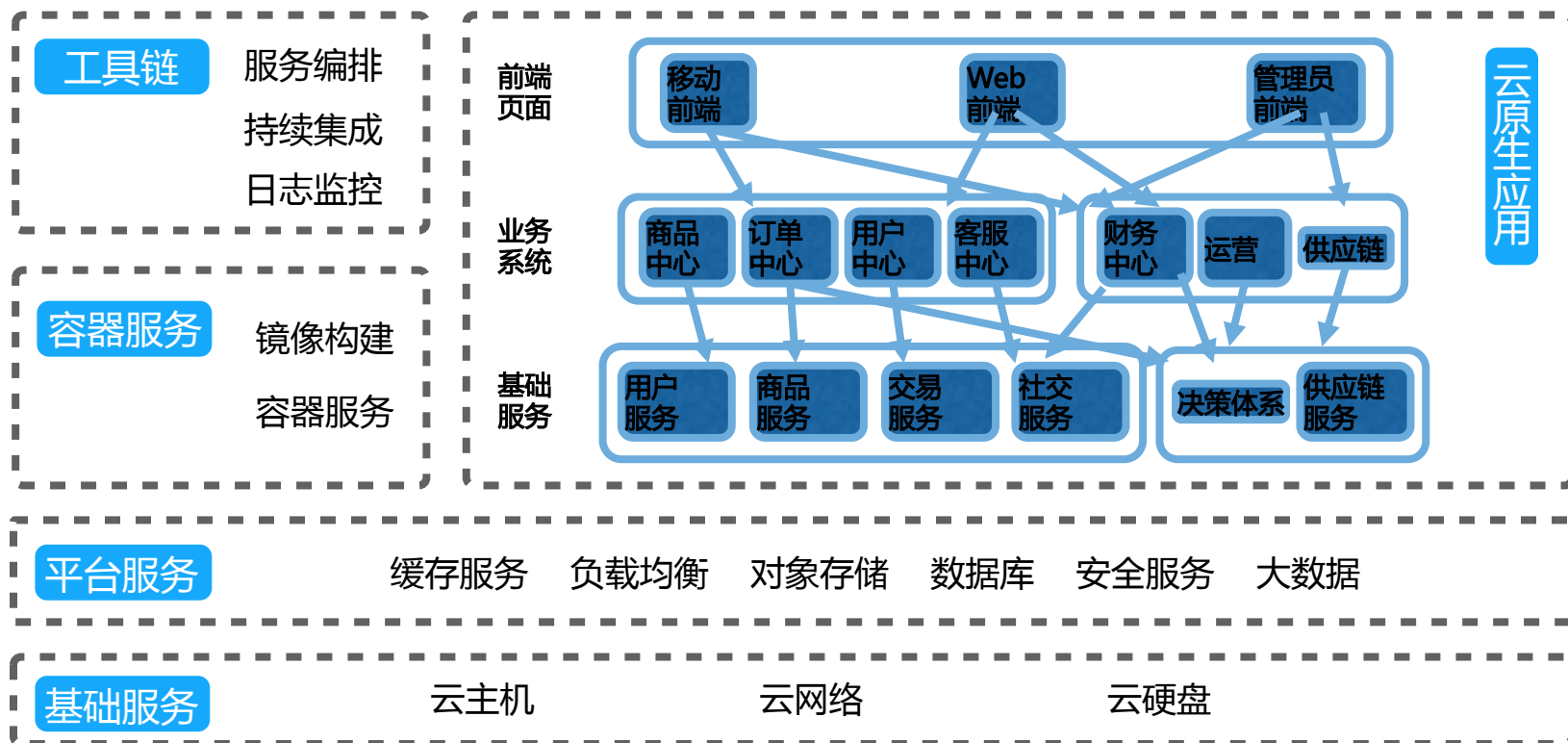
- 小时级别
- 了解数据



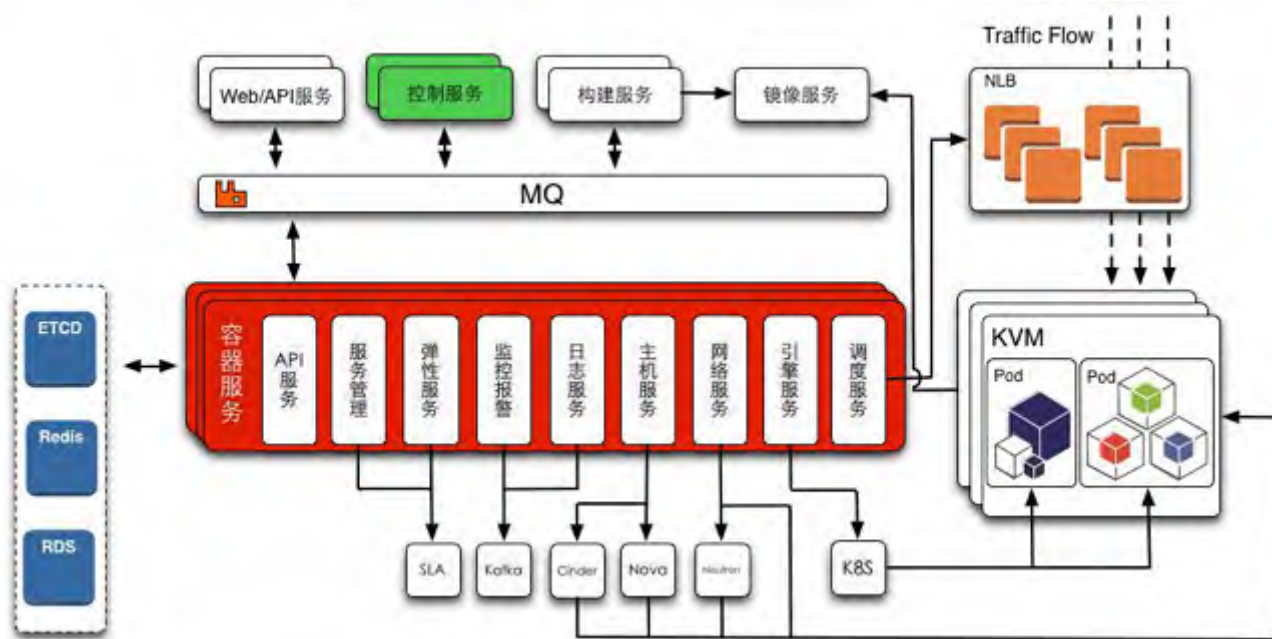
# 误区九：自己搭建数据库，大数据平台等公共基础设施



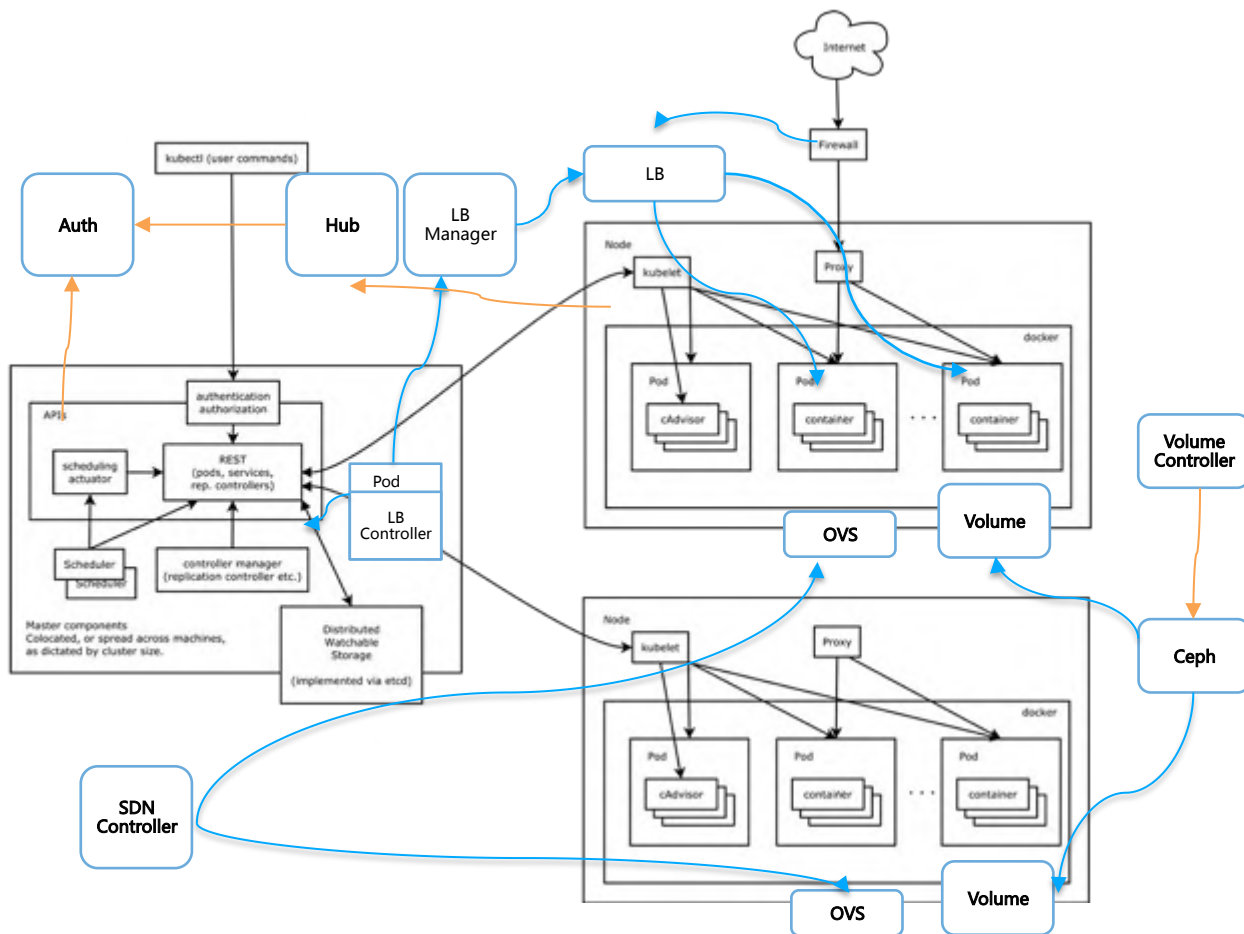
# 如何发挥云计算的优势



# 容器层实践-总体架构



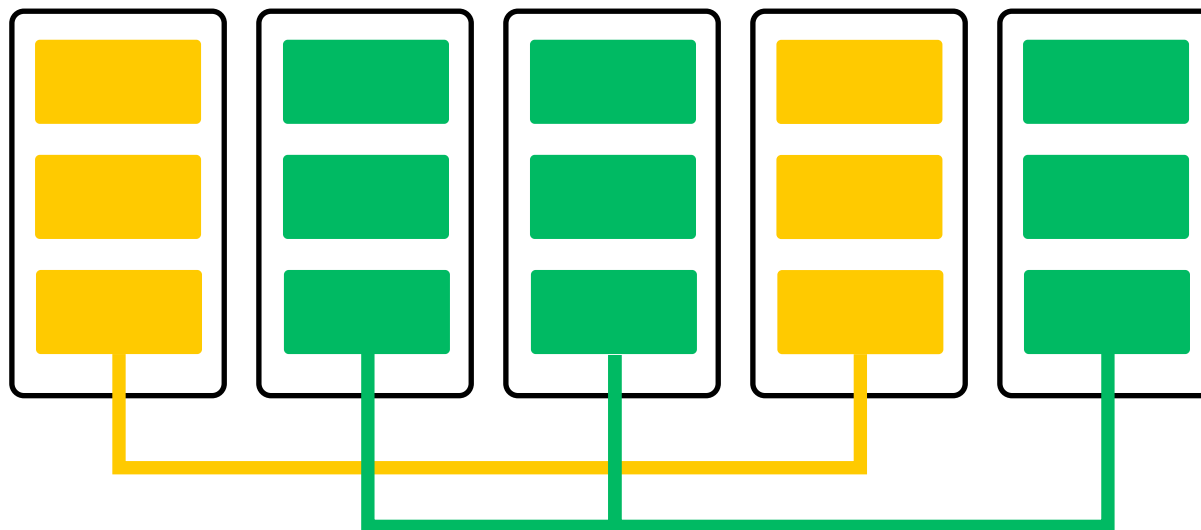
# 容器层实践-详细架构





# 容器层实践-计算安全

安全隔离：不同租户主机隔离，内核隔离，虚拟网络隔离

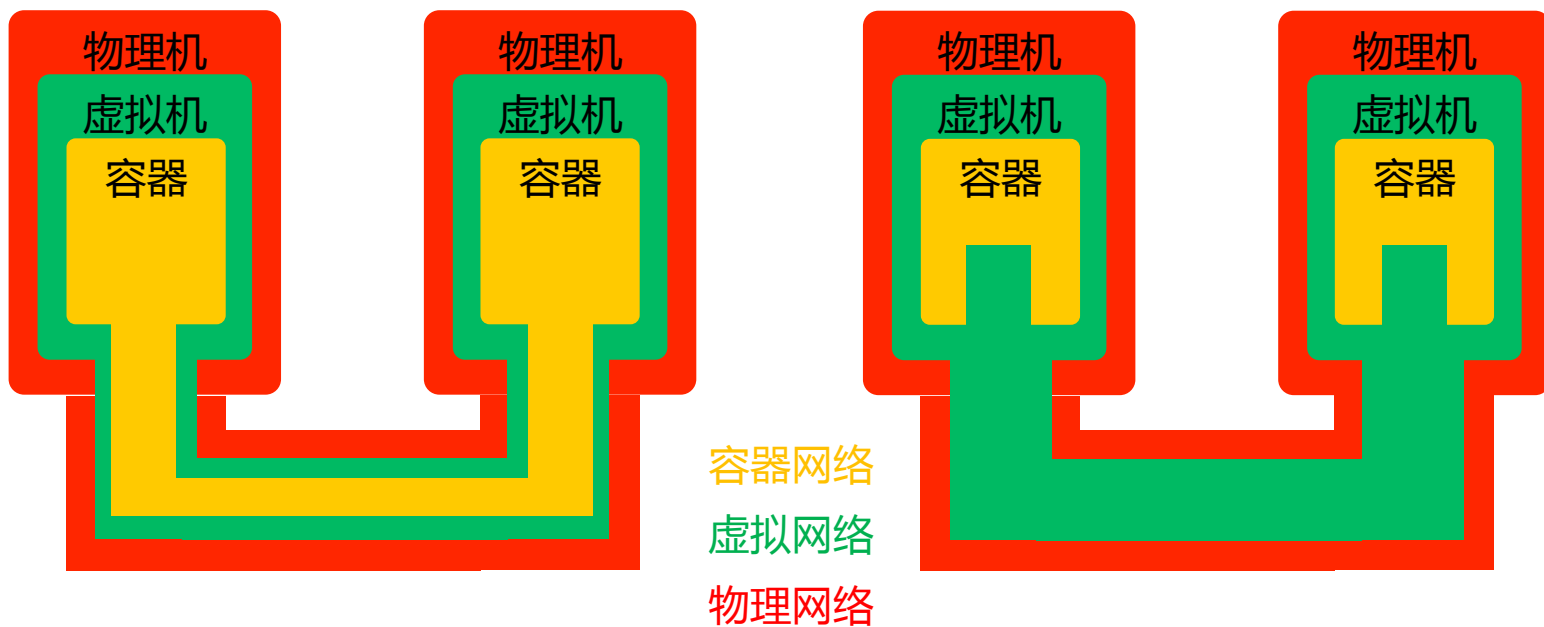


# 容器层实践-启动速度

- 问题
  - 容器运行于云主机，容器启动依赖于云主机先启动，而基于硬件虚拟化技术的云主机启动速度较慢
- 启动速度优化
  - 定制系统镜像，裁剪不必要服务启动加载项
  - 实现云主机IP静态化，加速网络初始化过程
  - 优化OpenStack创建云主机流程

# 容器层实践-网络存储

高性能：网络和存储无二次虚拟化，保证高吞吐量

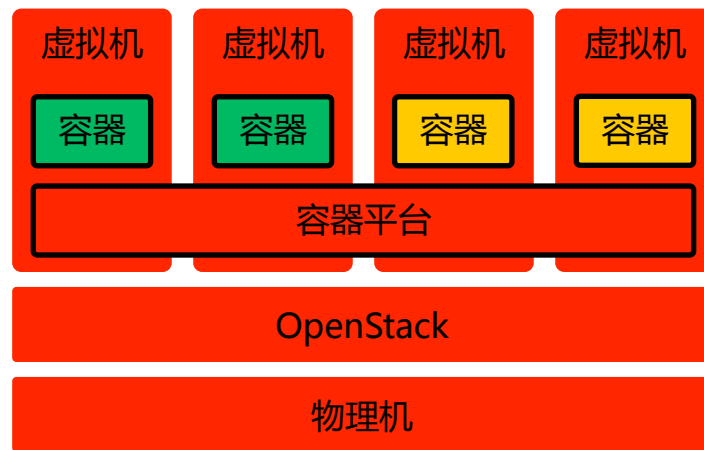
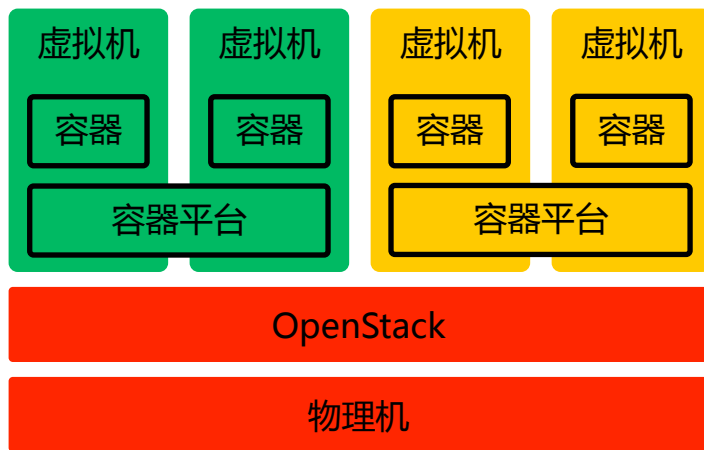


# 容器层实践-统一集群

大规模单容器集群，统一由云平台运维，用户仅仅需要关注应用

租户A应用运维人员负责

租户B应用运维人员负责

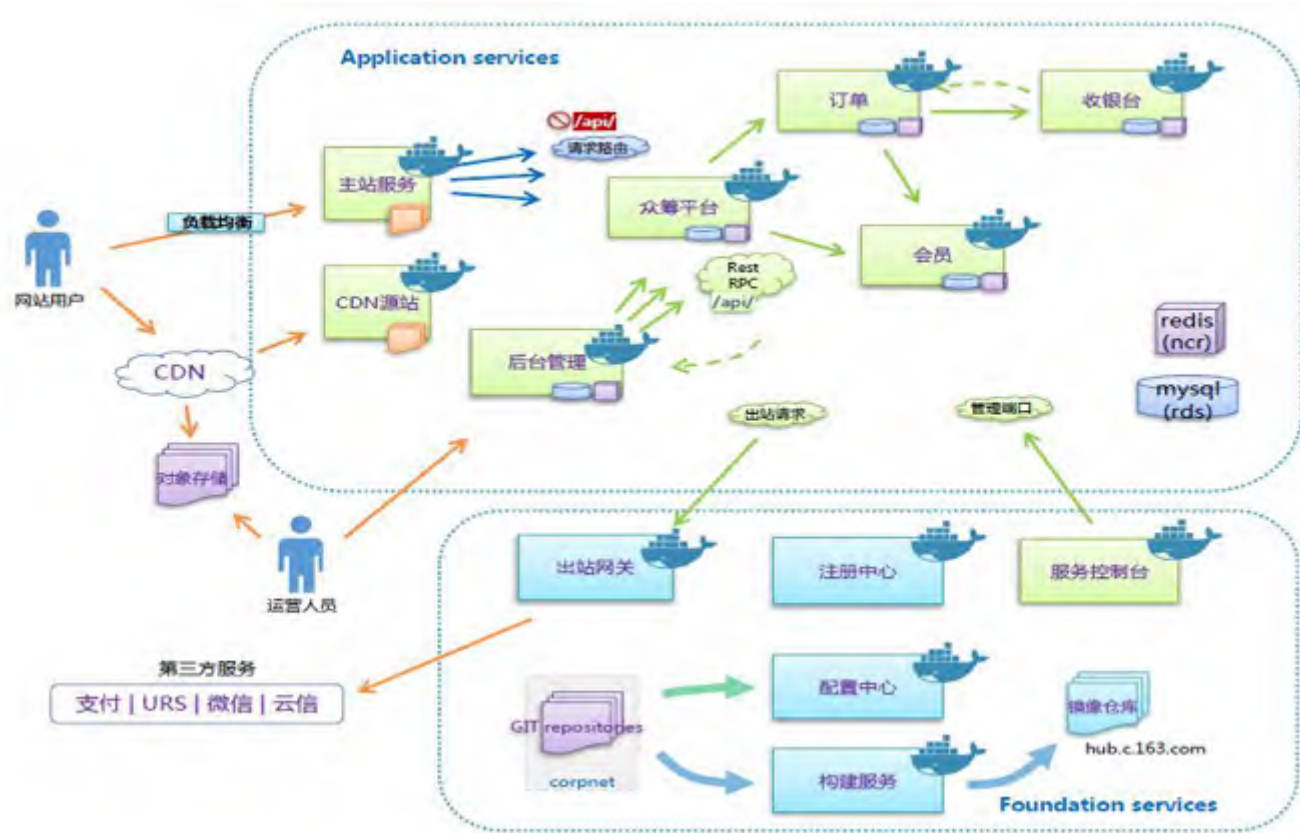


云计算运维人员负责

# 容器层实践-编排性能

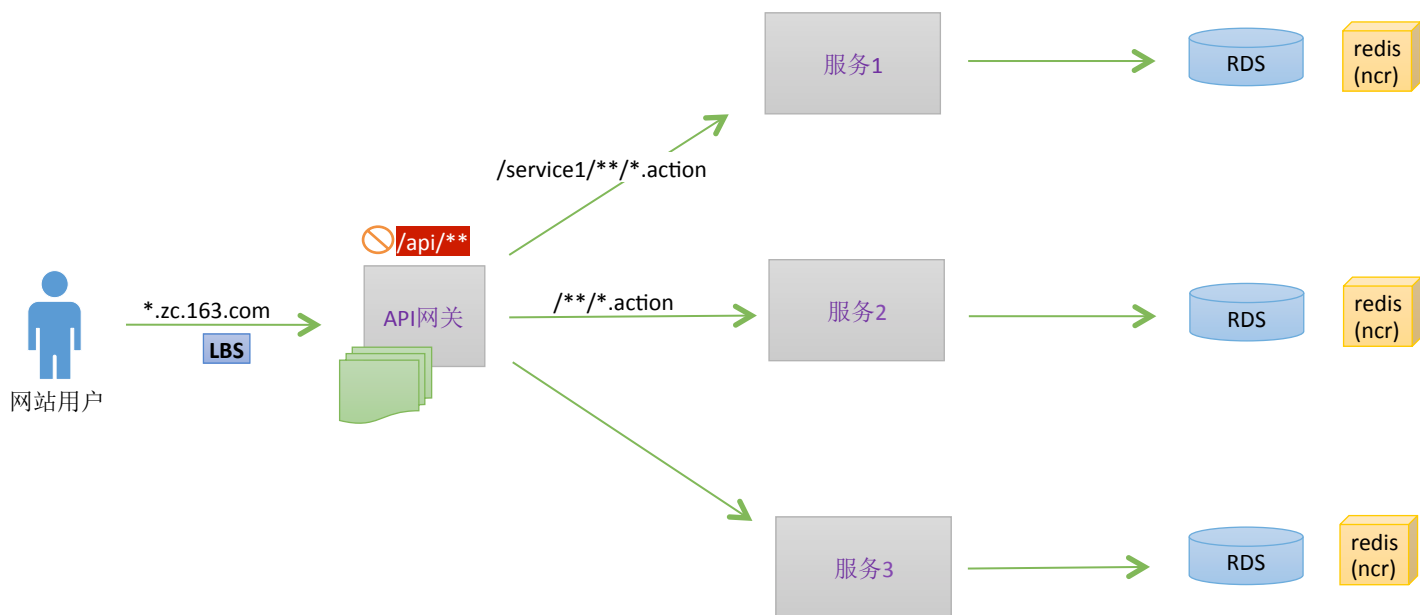
- 性能问题：
  - 原生K8S编排的可扩展性有瓶颈，无法支持更大规模节点的编排
  - K8S各组件在高水位下资源占用太大
- 优化手段
  - 实现调度器并行调度，提升集群调度能力
  - 实现副本控制器多优先级处理，提升容器创建速度
  - 精简内部负载均衡转发表
  - 优化api-server和kubelet、kube-proxy，减少内存资源占用

# 应用层实践-总体架构

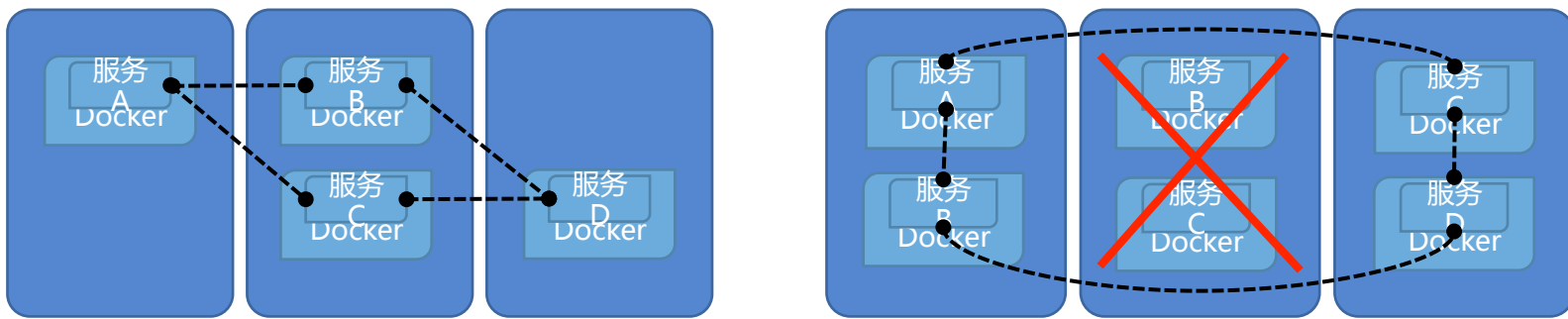




# 应用层实践-设计要点一：负载均衡 + API网关



# 应用层实践-设计要点二：服务拆分与服务发现



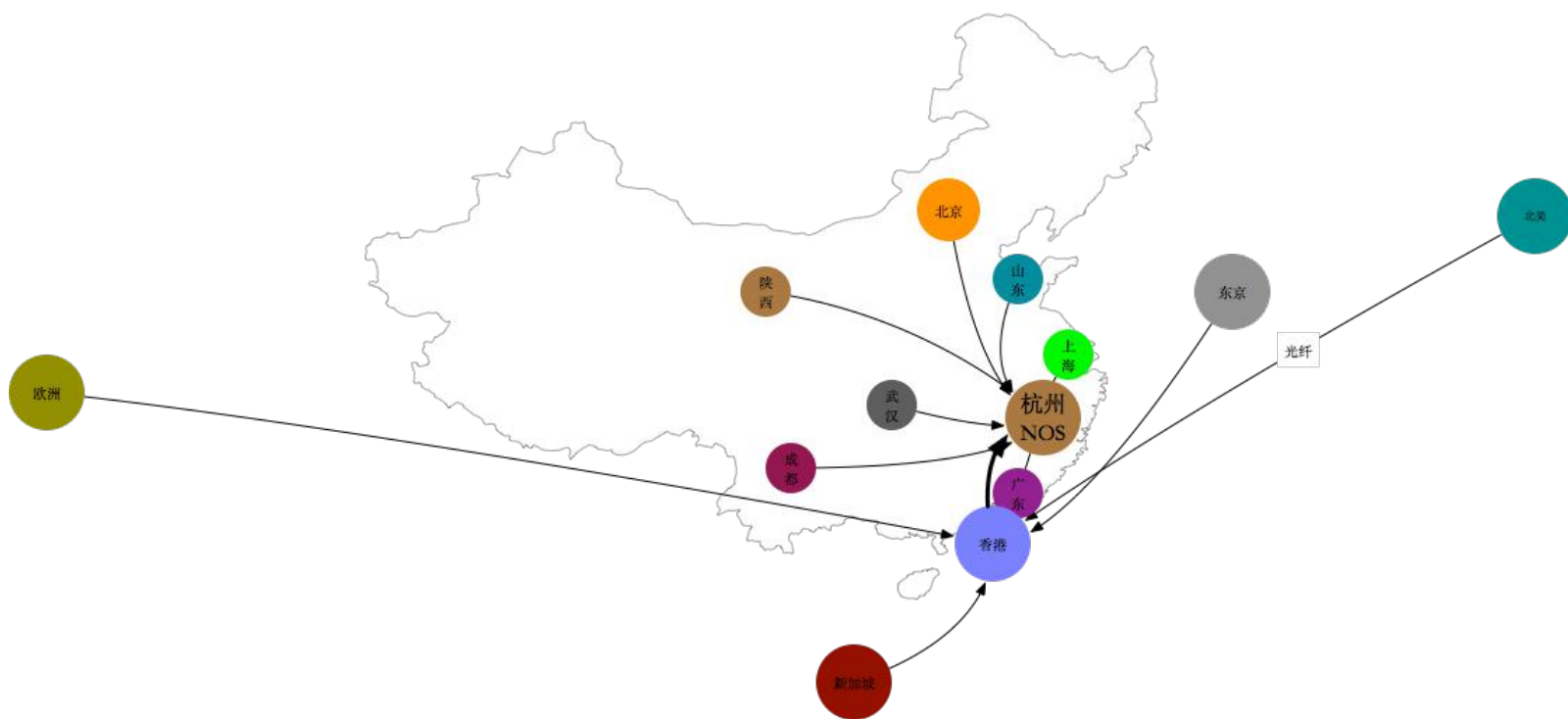
开发独立: 代码耦合度比较高，修改代码通常会对多个模块产生影响，操控难度大，风险高

上线独立: 单次上线需求列表多，上线时间长，影响面大

简化扩容: 由于业务多，每一次扩容需要增加的配置比较杂。一些不起眼的小业务虽然不是扩容的主要目的，也需要慎重考虑

容灾降级: 核心业务与非核心业务耦合，在关键时候互相影响

# 应用层实践-设计要点三：对象存储 +CDN下载



# 应用层实践-设计要点四：使用PaaS服务降低设计难度

基础设施



DDB



RDS



NOS



ZooKeeper



NQS



NKV



MCR



VCloud



NDIR

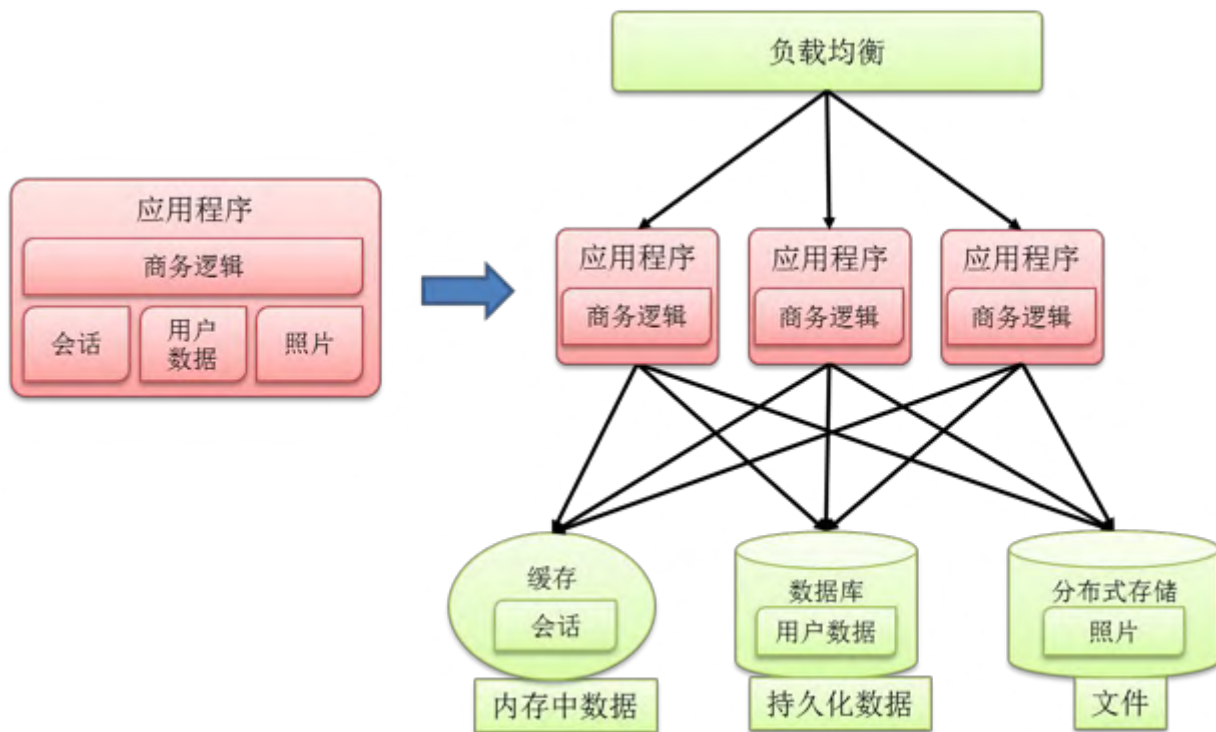


Hadoop

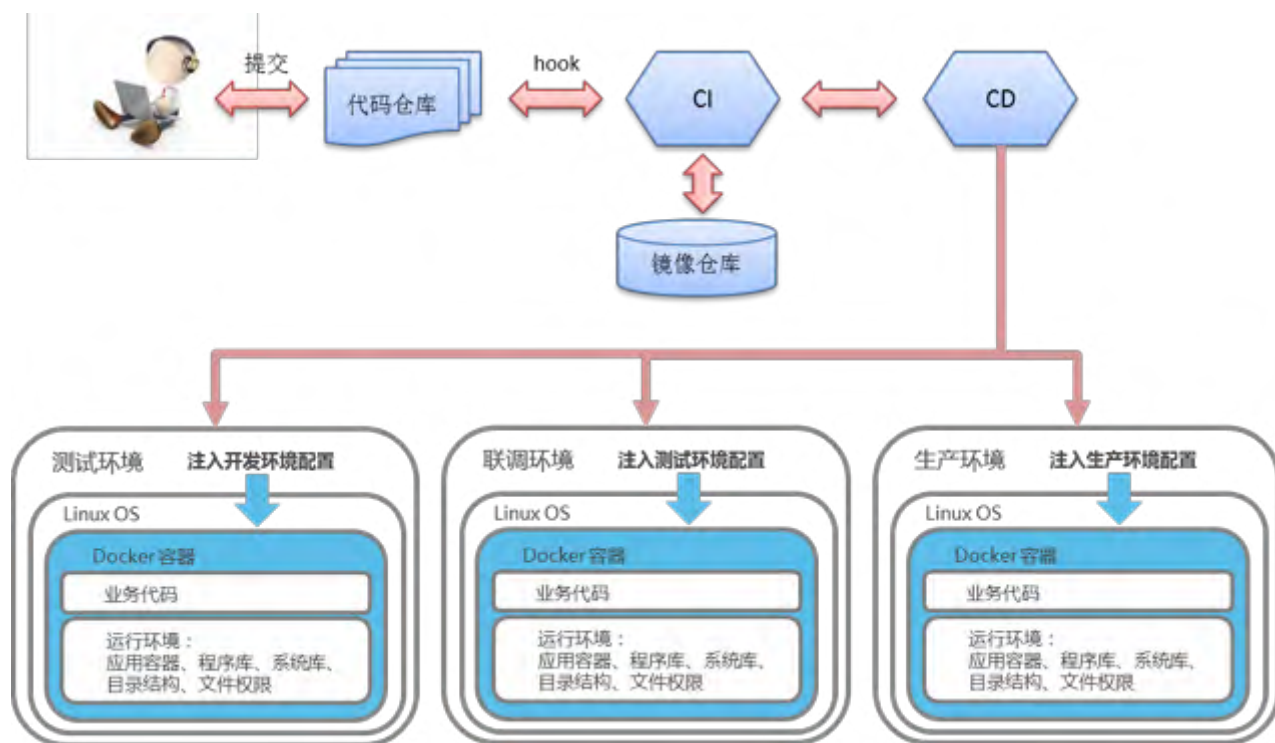


Storm

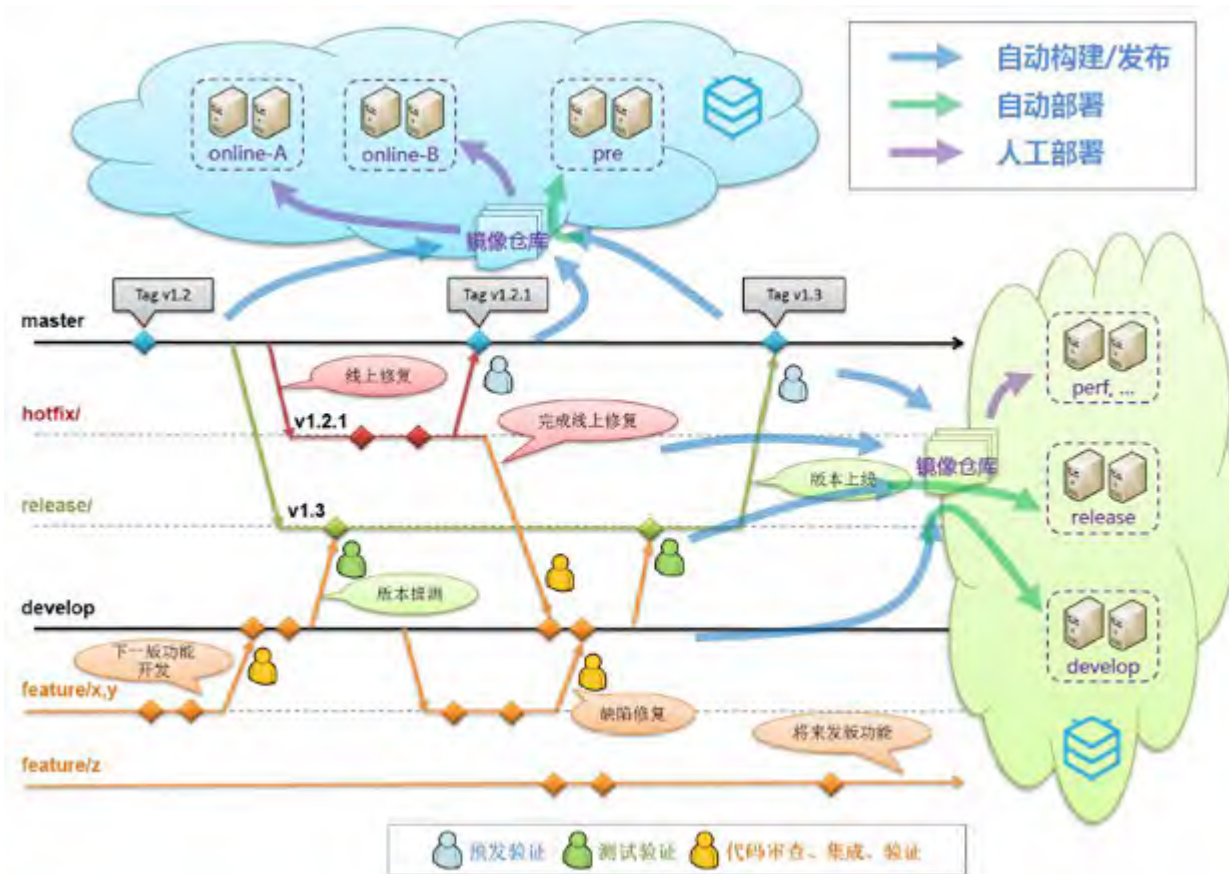
# 应用层实践-设计要点五：无状态化服务改造



# 应用层实践-设计要点六：容器作为持续集成持续交付的工具

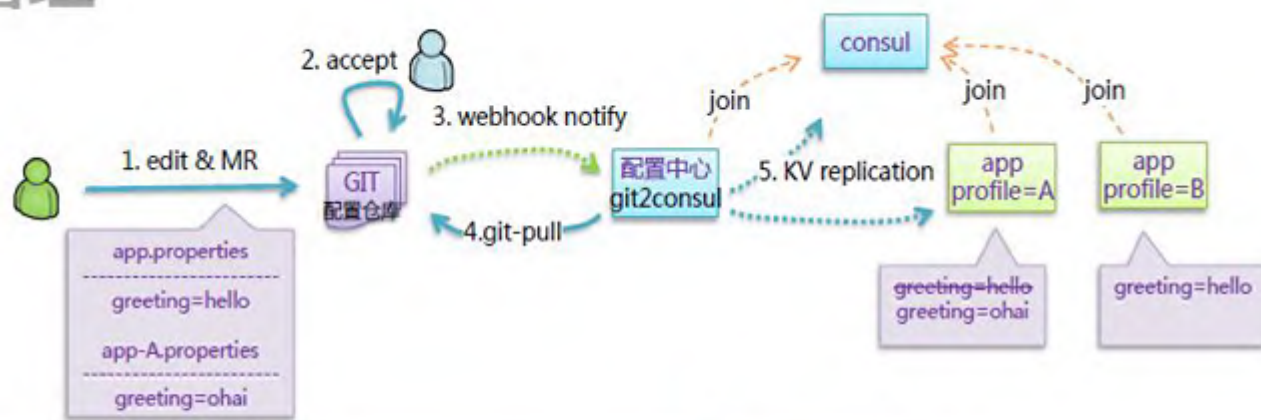


# 应用层实践-设计要点七：基于代码仓库的持续交付流程



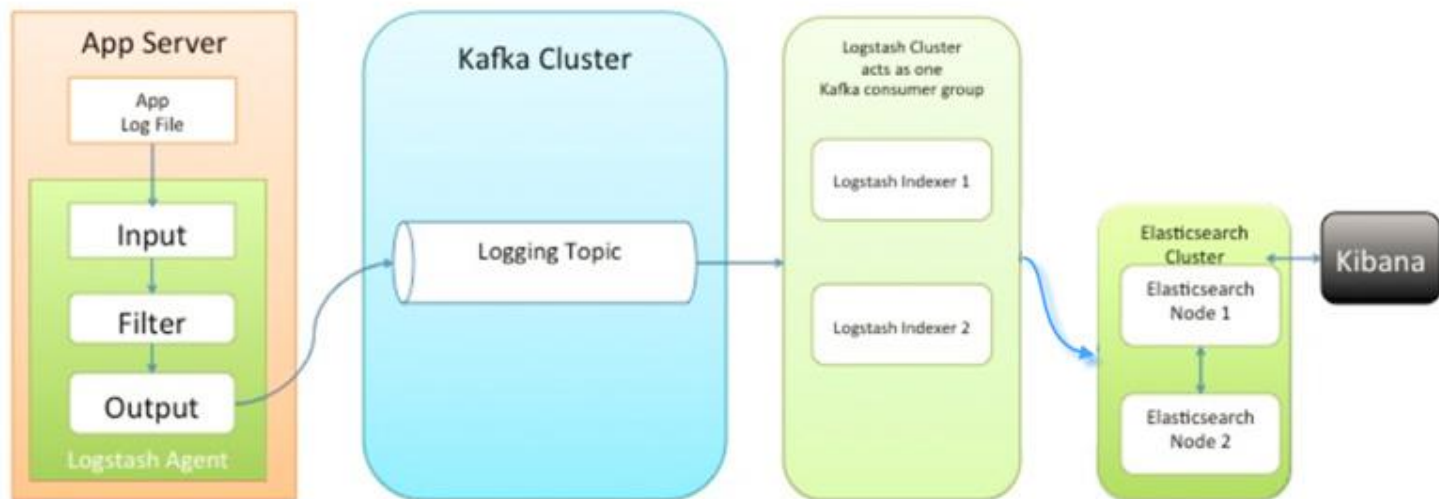
# 应用层实践-设计要点八：集中的配置管理

## 配置管理





# 应用层实践-设计要点九：基于流和搜索引擎的日志分析



Thank you !