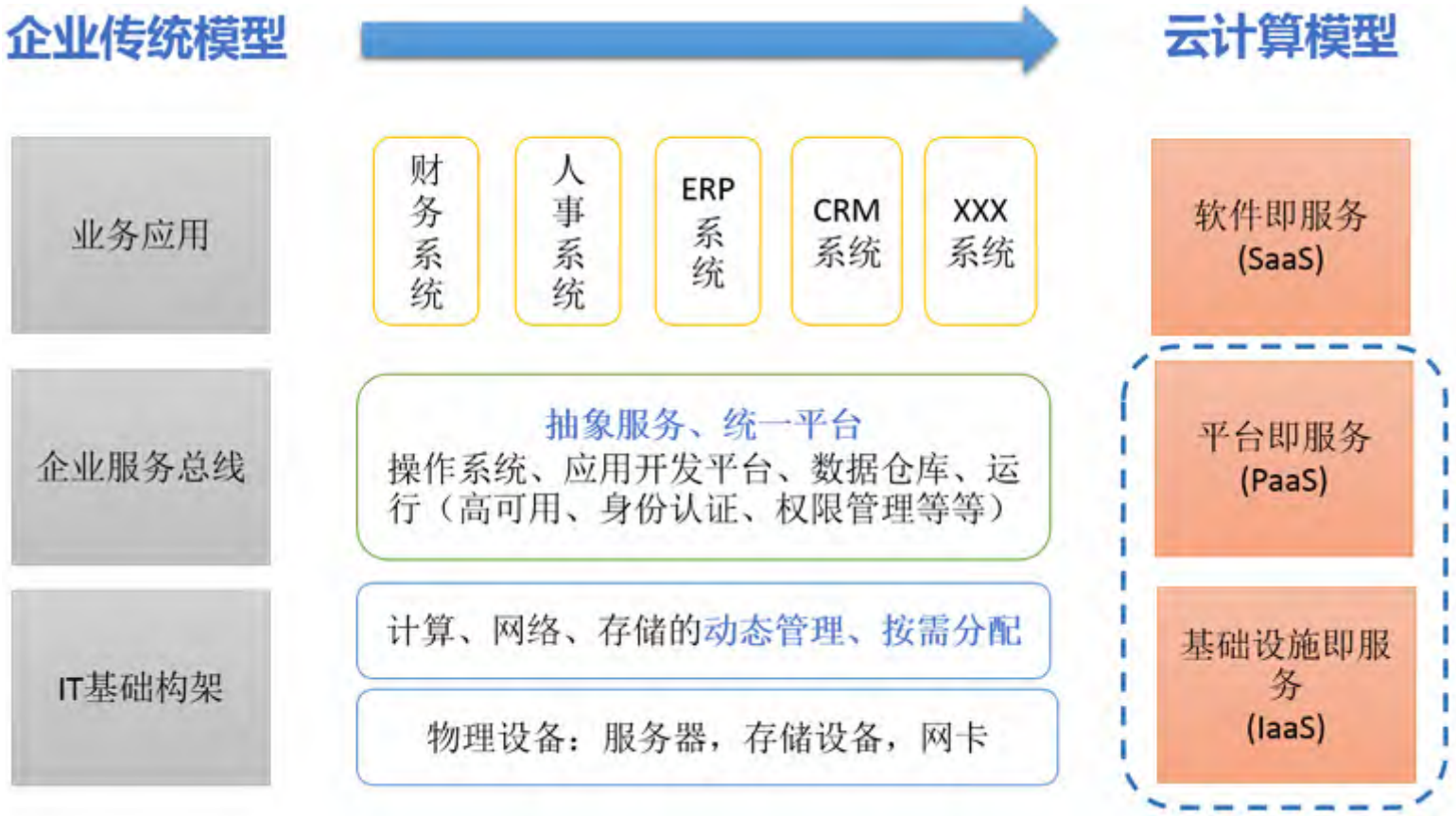
基于OpenStack和Kubernetes构建组合云平台

企业数据中心IT系统模型的转变



组合云平台的构建

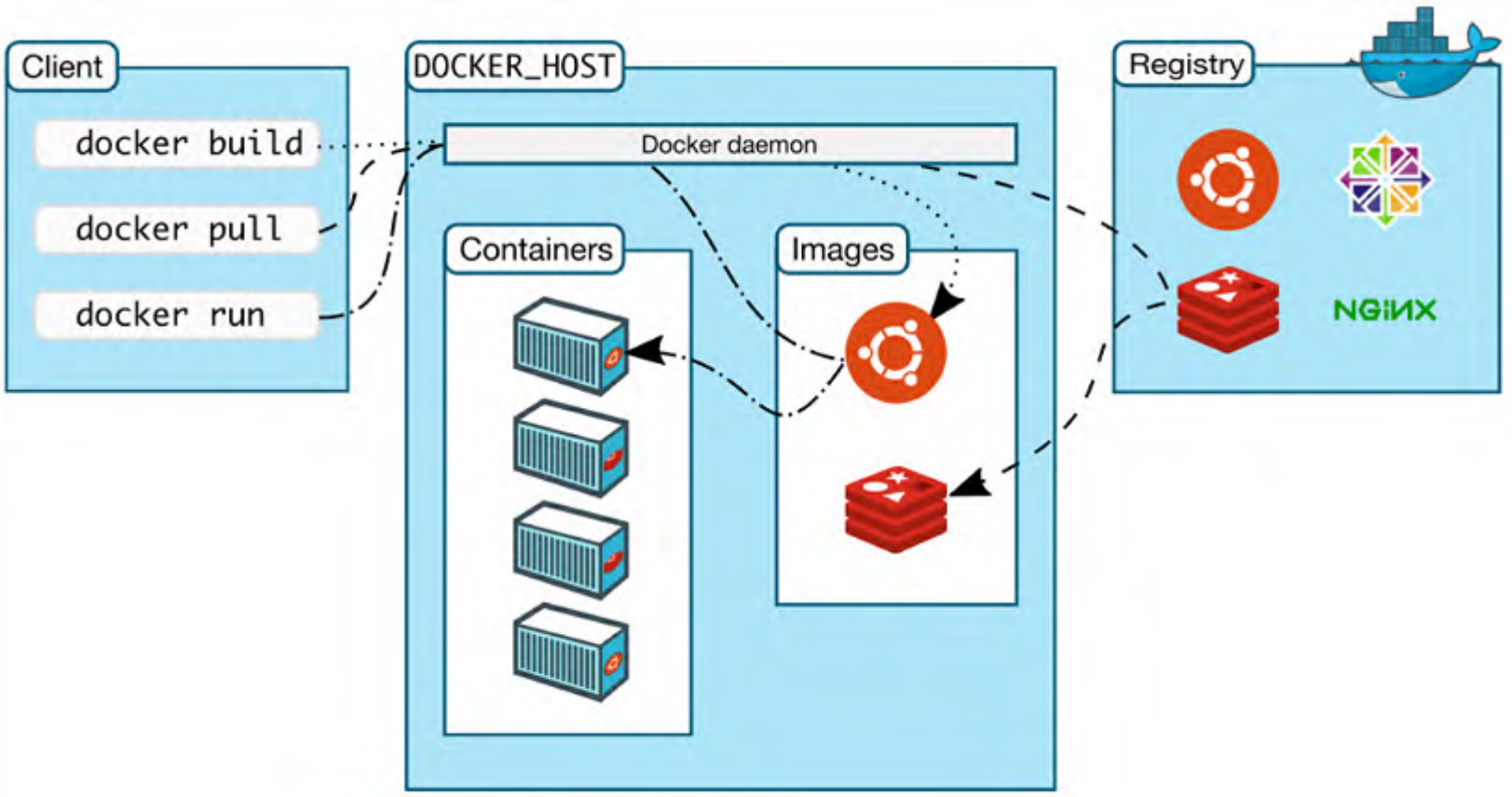
- 基础技术的选型
- 跨IaaS的容器集群部署
- 容器和虚拟机的组合编排
- 跨容器和虚拟机的网络服务
- 跨容器和虚拟机的存储服务
- 云平台基本服务架构和应用部署架构
- 我们趟过的那些坑

基础设施云平台和容器编排管理系统的选型

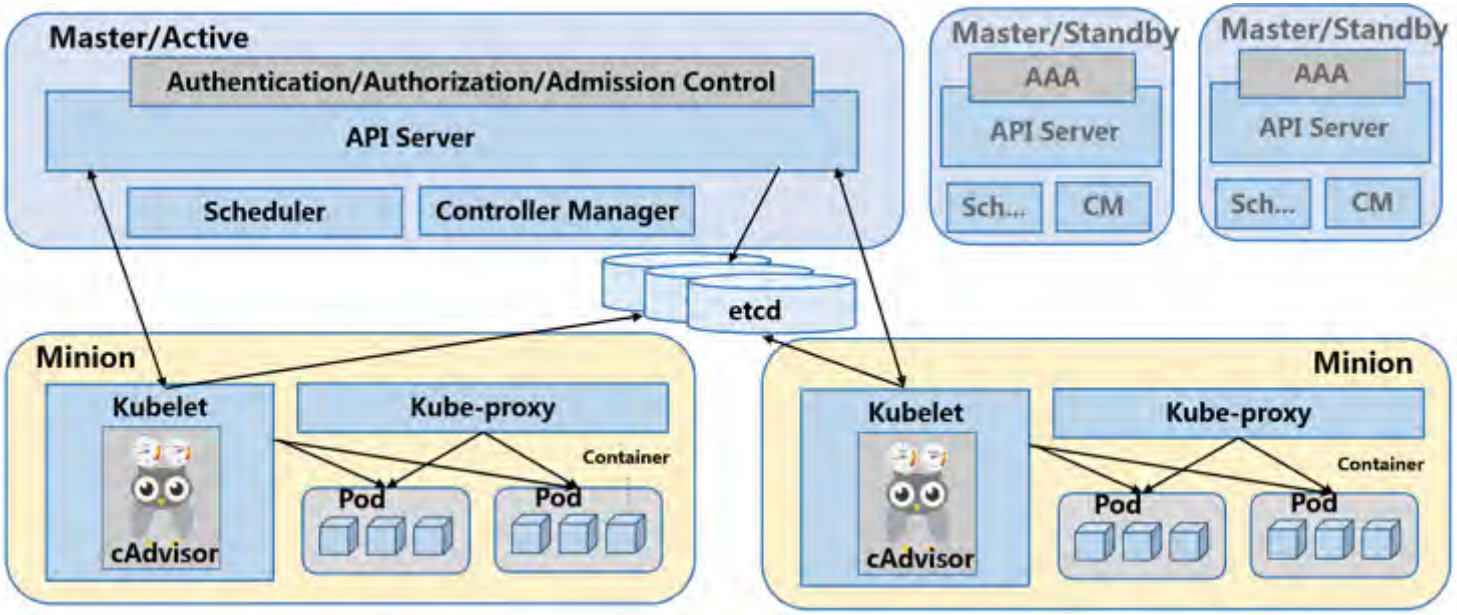
云平台技术选型

OpenStack提供了IaaS层服务也提供了诸多PaaS层服务

容器和Docker成为微服务架构的交付标准



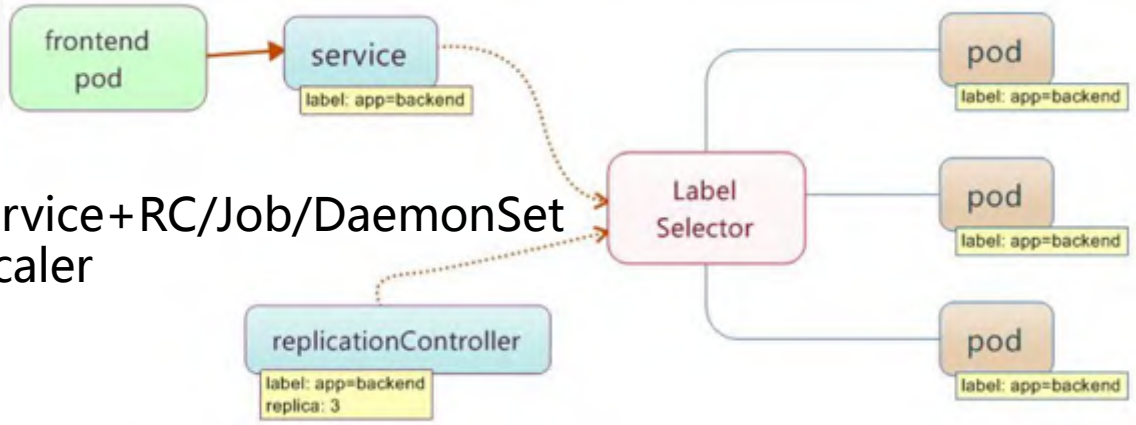
Kubernetes原生为生产环境而设计的容器集群编排系统



- 来自Google的简单一致的设计理念，原生为容器集群打造
- 原生服务发现
  - 统一的资源模型
  - 支持丰富的标签Label发现机制
- 原生负载均衡，高可用方案
- 原生的Rolling Update设计
- 为生产环境专门打造的容器集群

Kubernetes的统一资源模型和丰富的标签选择器

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: test-ebs
spec:
  containers:
  - image: gcr.io/google_containers/test-webserver
    name: test-container
    volumeMounts:
    - mountPath: /test-ebs
      name: test-volume
  volumes:
  - name: test-volume
    # This AWS EBS volume must already exist.
    awsElasticBlockStore:
      volumeID: <volume-id>
      fsType: ext4
```



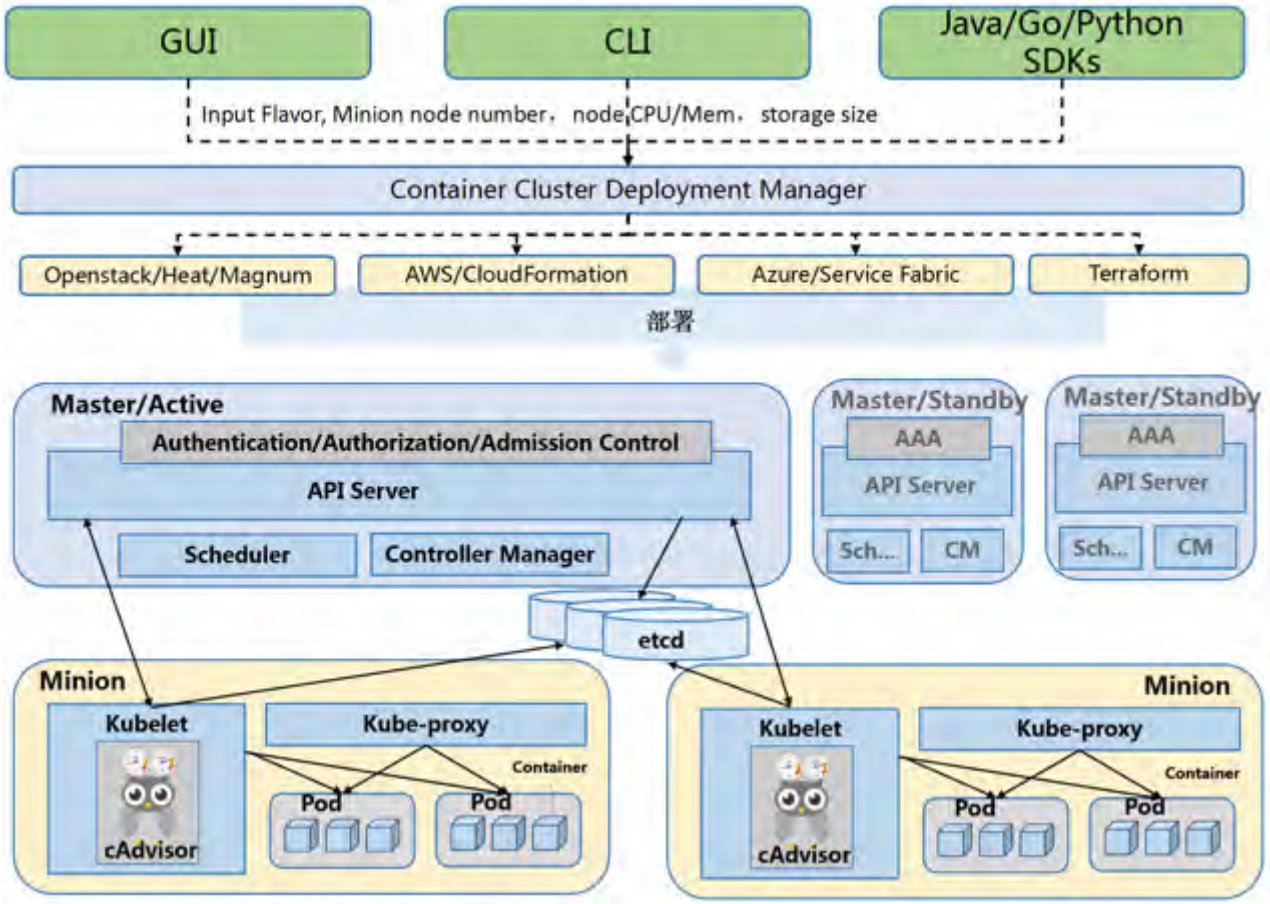
- 多镜像Pod
- 多种业务类型：Service+RC/Job/DaemonSet
- 自动伸缩：AutoScaler
- 多种Volumn驱动

组合云平台的构建

- 基础技术的选型
- 跨IaaS的容器集群部署
- 容器和虚拟机的组合编排
- 跨容器和虚拟机的网络服务
- 跨容器和虚拟机的存储服务
- 云平台基本服务架构和应用部署架构
- 我们趟过的那些坑

兼容不同IaaS的容器集群部署系统

利用不同的编排驱动将容器集群部署到不同云平台



组合云平台的构建

- 基础技术的选型
- 跨IaaS的容器集群部署
- 容器和虚拟机的组合编排
- 跨容器和虚拟机的网络服务
- 跨容器和虚拟机的存储服务
- 云平台基本服务架构和应用部署架构
- 我们趟过的那些坑

单纯的容器服务不足以满足企业云平台的需求

Martin Fowler

01 July 2015

Microservices provide benefits...

- Strong Module Boundaries: Microservices reinforce modular structure, which is particularly important for larger teams.

- Independent Deployment: Simple services are easier to deploy, and since they are autonomous, are less likely to cause system failures when they go wrong.

- Technology Diversity: With microservices you can mix multiple languages, development frameworks and data-storage technologies.

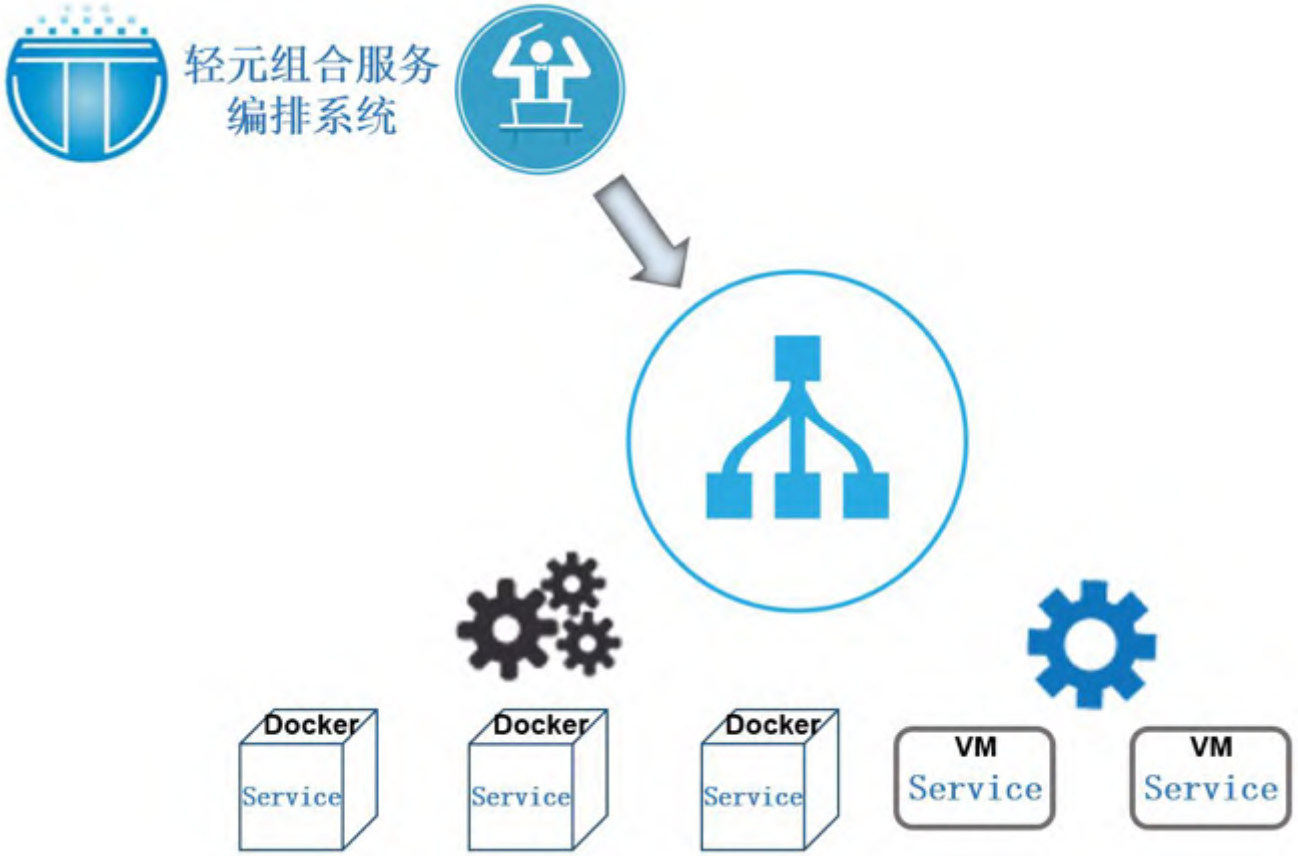
...but come with costs

- Distribution: Distributed systems are harder to program, since remote calls are slow and are always at risk of failure.

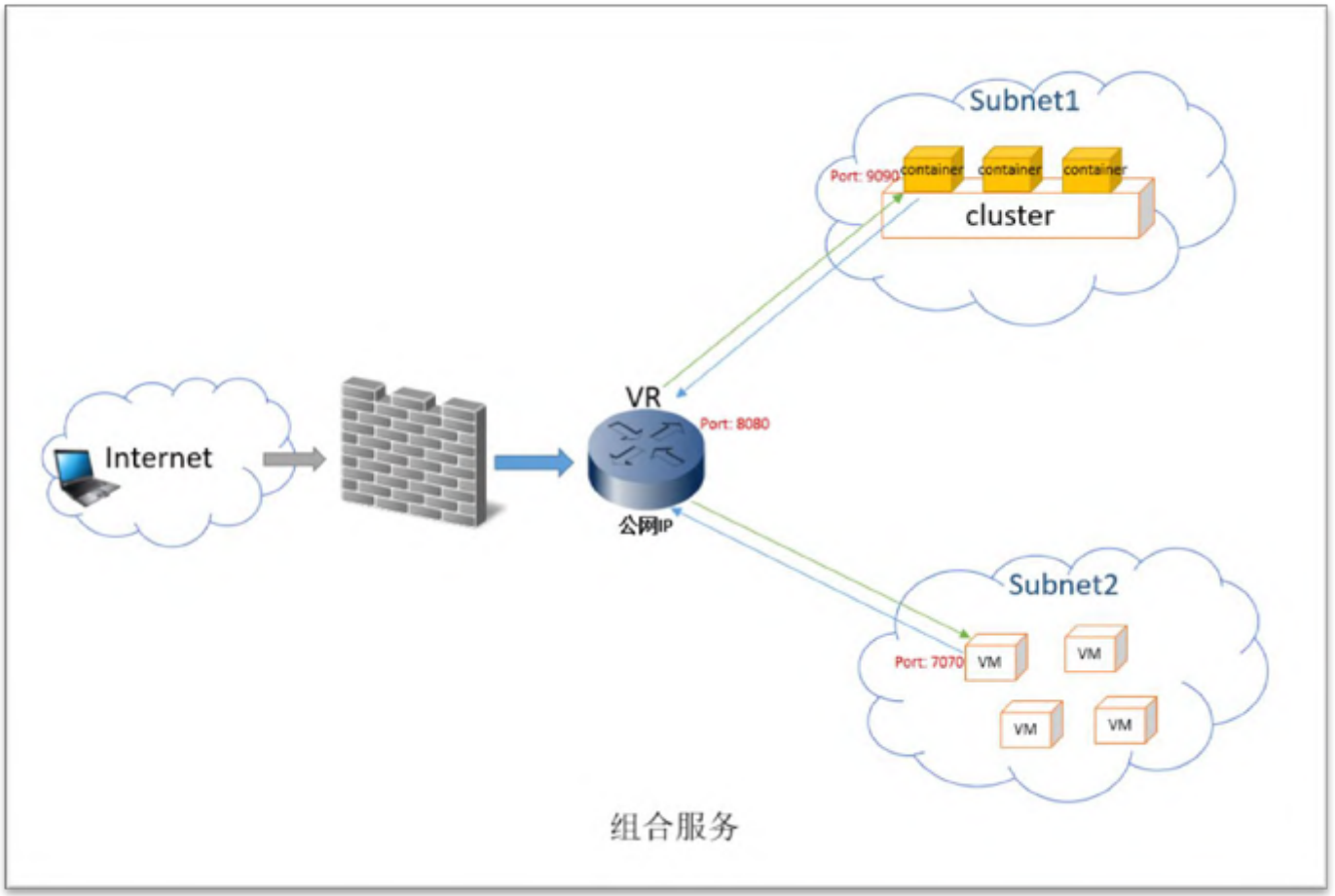
- Eventual Consistency: Maintaining strong consistency is extremely difficult for a distributed system, which means everyone has to manage eventual consistency.

- Operational Complexity: You need a mature operations team to manage lots of services, which are being redeployed regularly.

同一平台统一管理容器集群和虚拟机



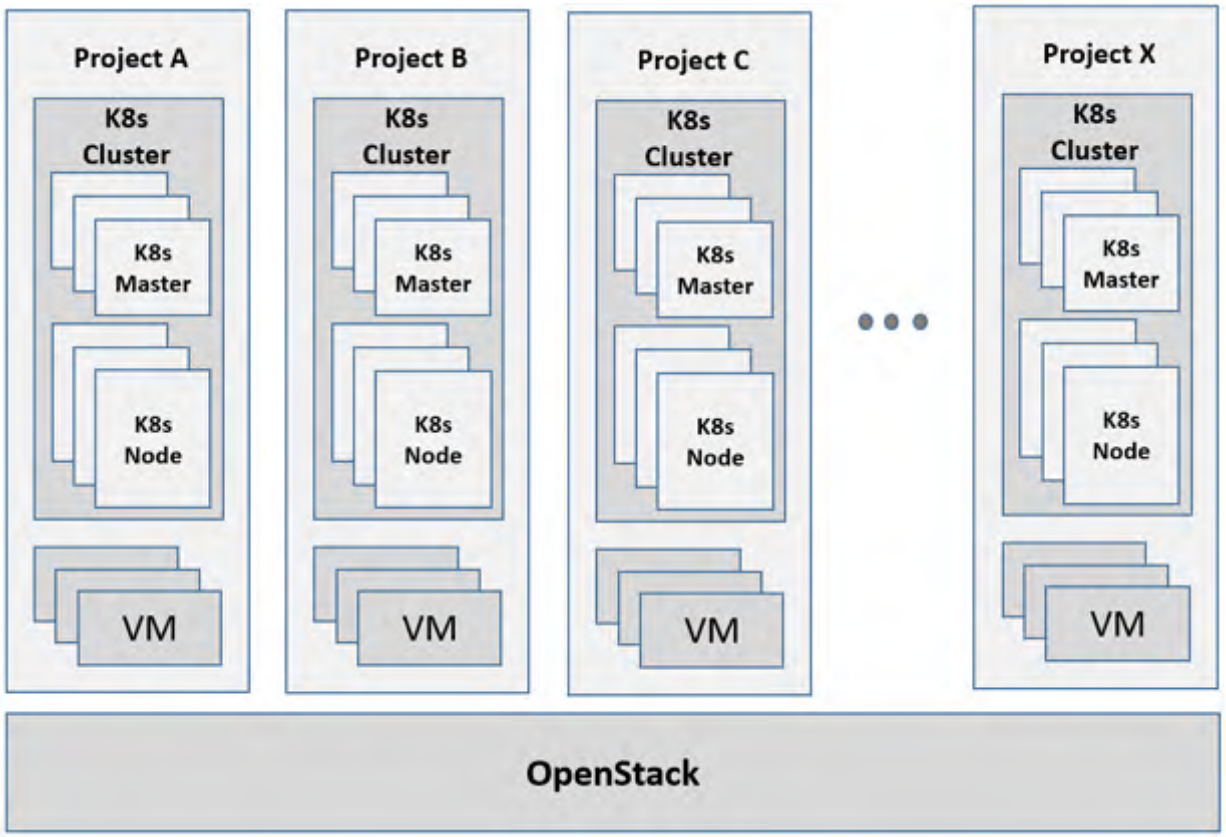
自动部署组合服务在用户私有网络通信



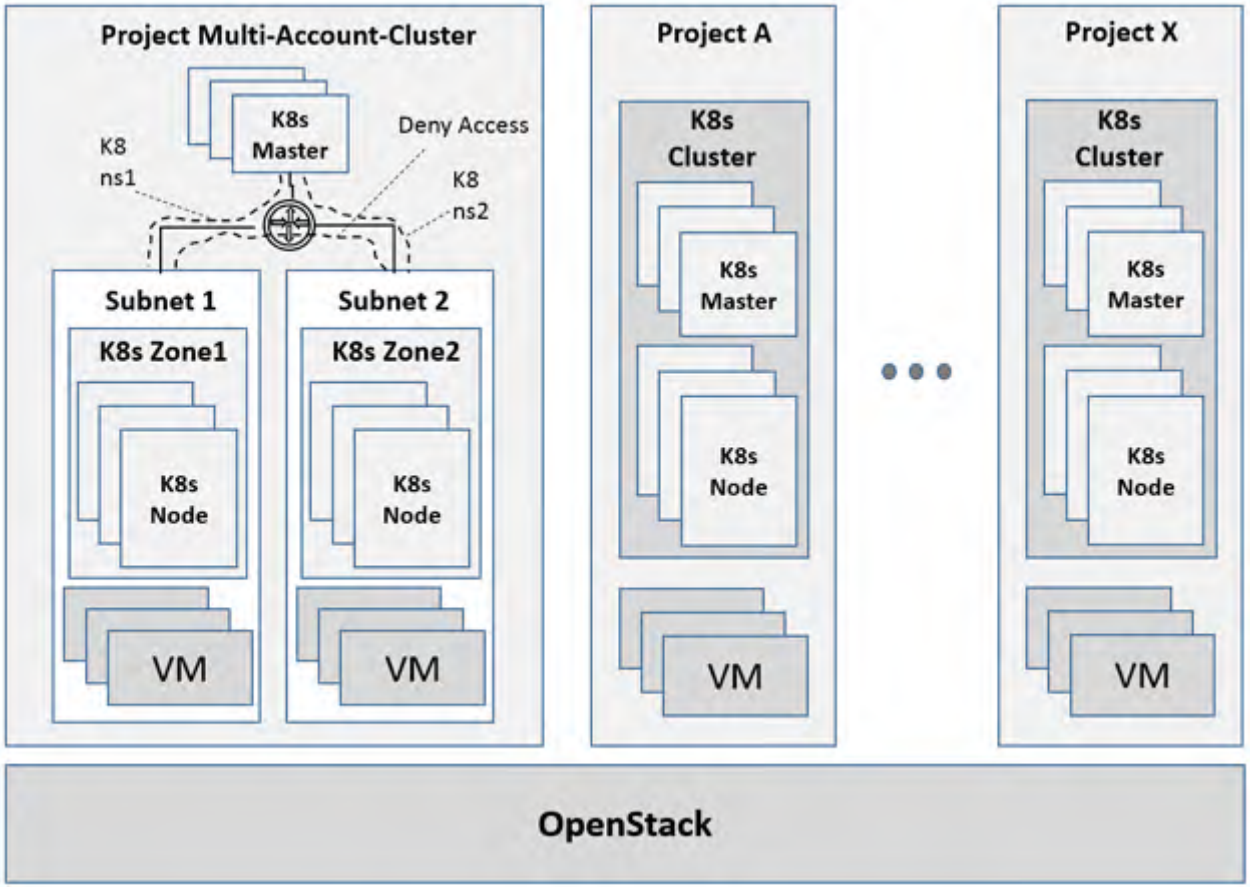
组合云平台的构建

- 基础技术的选型
- 跨IaaS的容器集群部署
- 容器和虚拟机的组合编排
- 跨容器和虚拟机的网络服务
- 跨容器和虚拟机的存储服务
- 云平台基本服务架构和应用部署架构
- 我们趟过的那些坑

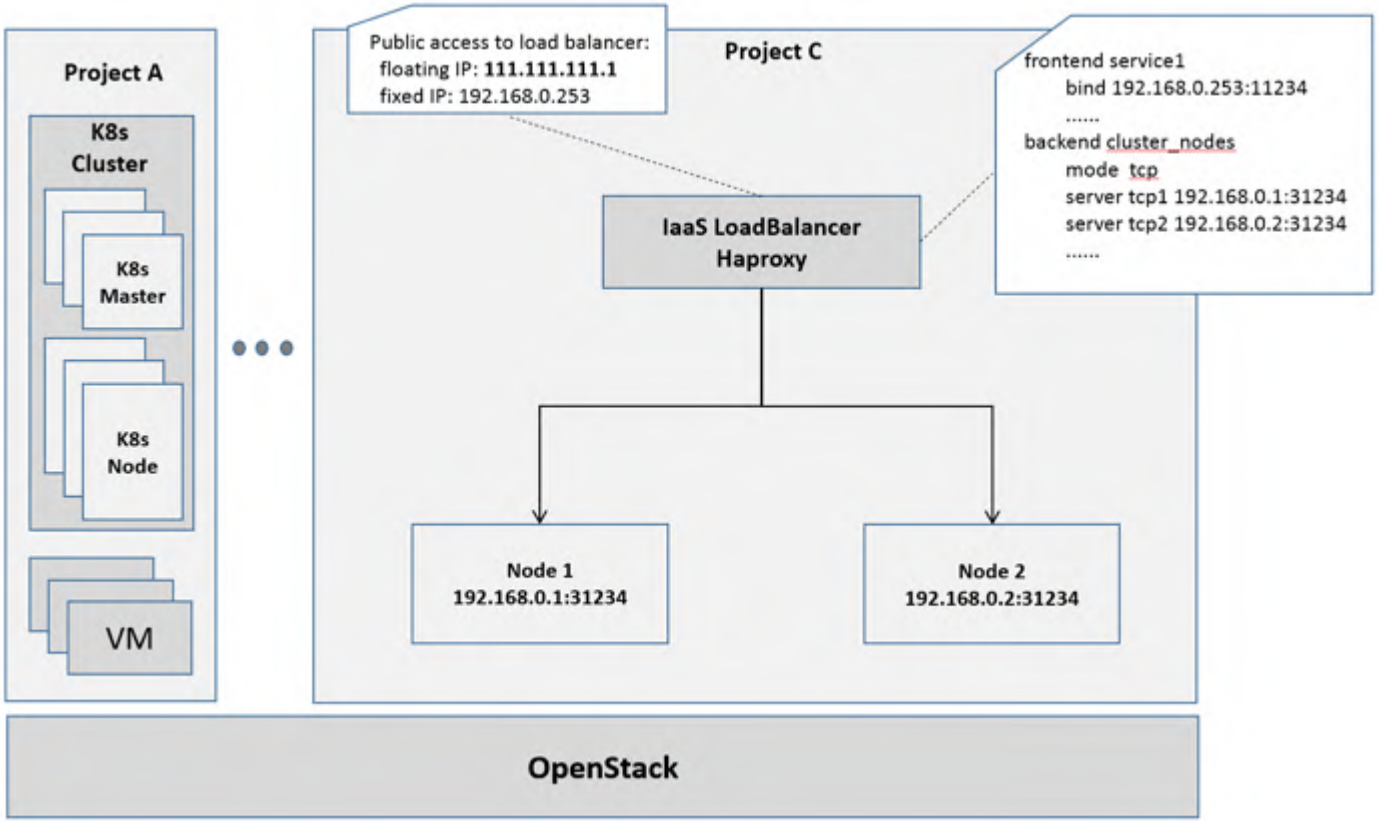
在Hypervisor和网络做多租户隔离



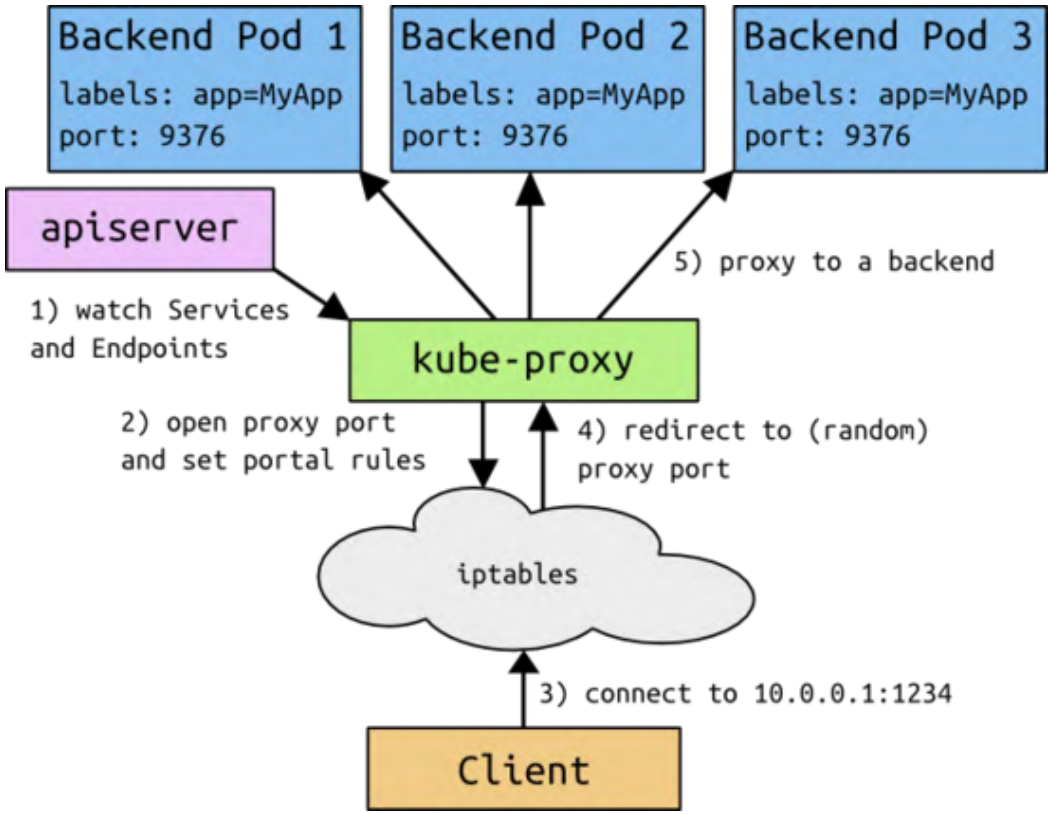
单租户模式下用子网和防火墙对用户集群做隔离



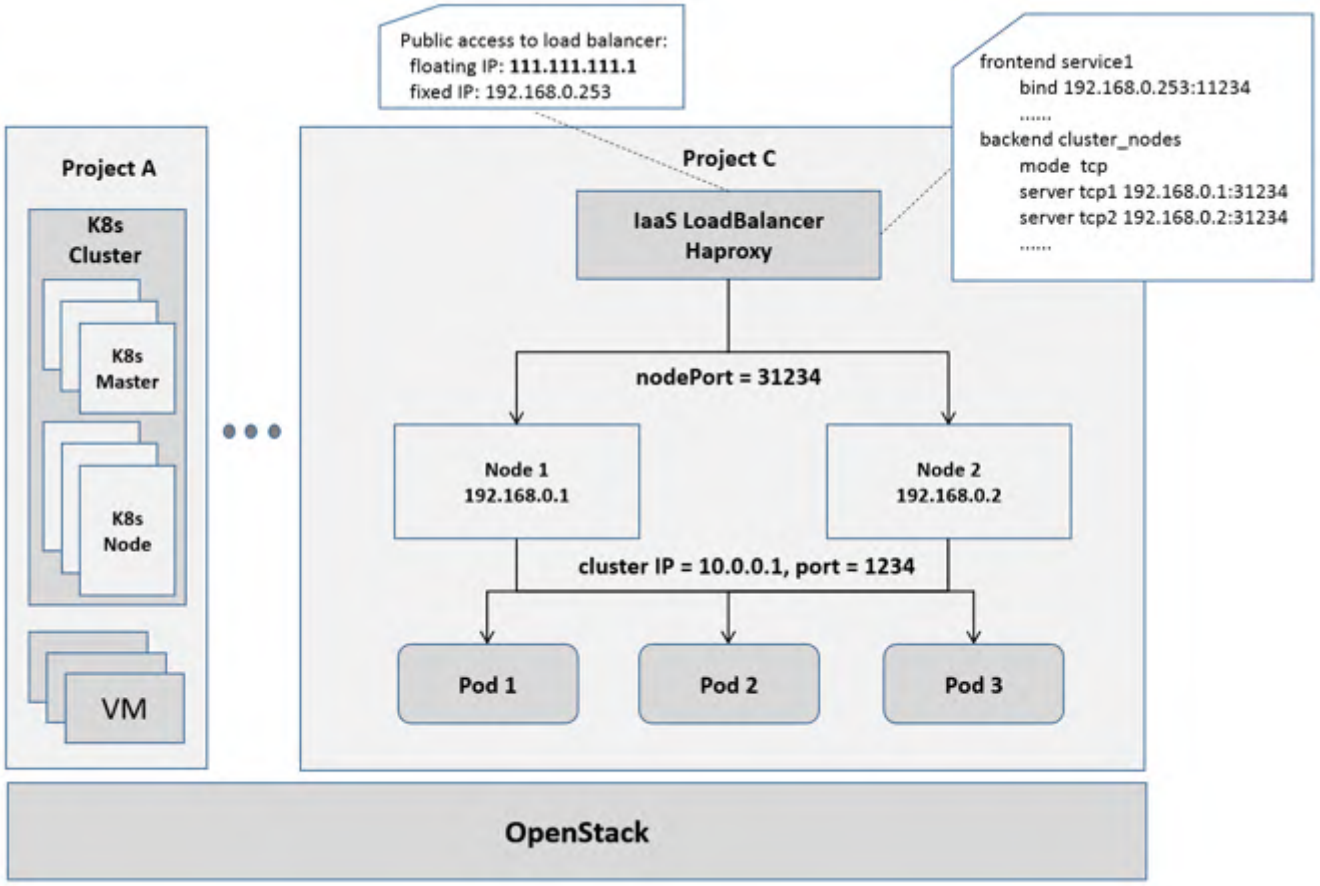
虚拟机服务的负载均衡


Project A
K8s Cluster
K8s Master
K8s Node
VM
Public access to load balancer:
floating IP: 111.111.111.1
fixed IP: 192.168.0.253
Project C
IaaS LoadBalancer Haproxy
frontend service1
bind 192.168.0.253:11234
.....
backend cluster_nodes
mode tcp
server tcp1 192.168.0.1:31234
server tcp2 192.168.0.2:31234
.....
Node 1
192.168.0.1:31234
Node 2
192.168.0.2:31234
OpenStack

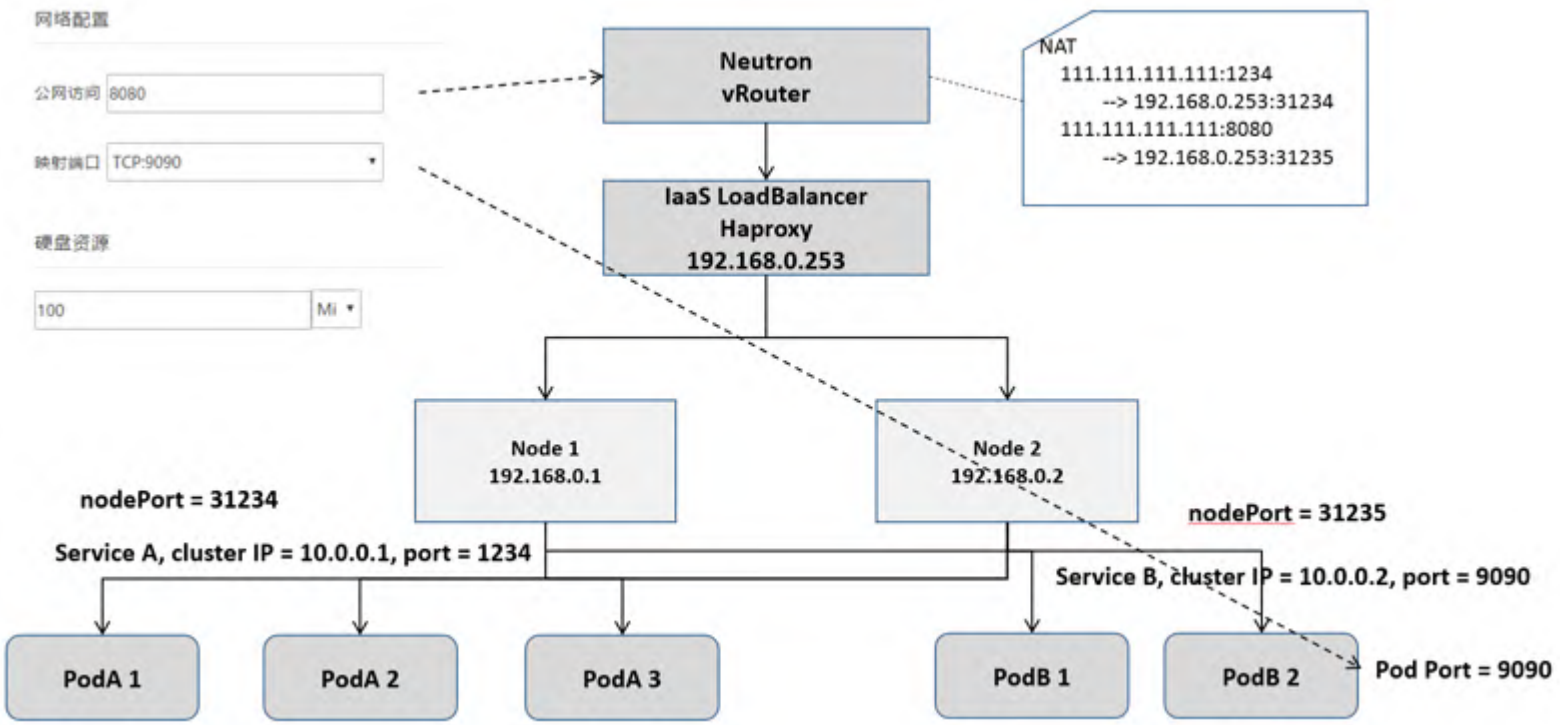
容器集群内部的负载均衡



发布容器集群服务到外网IP



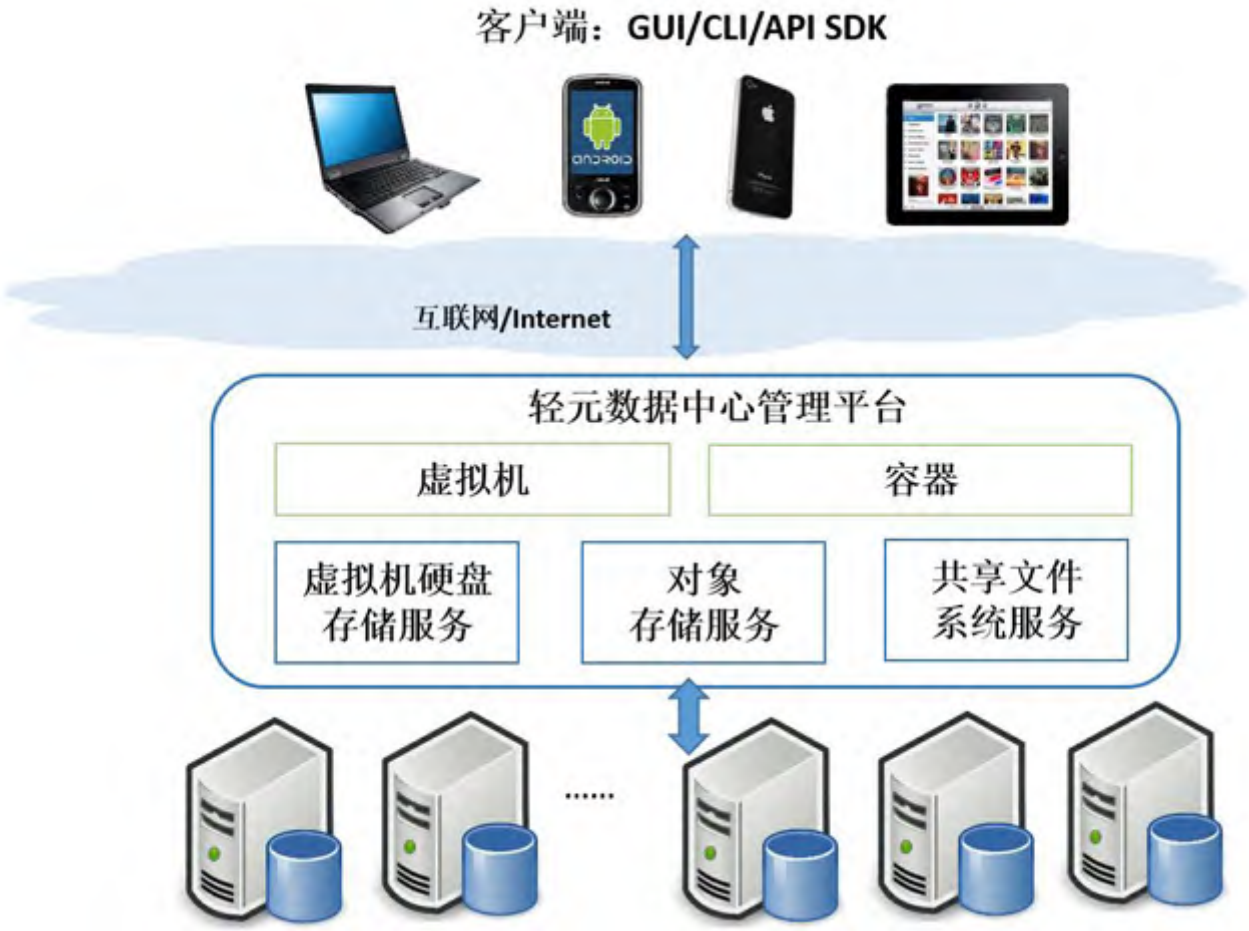
通过端口映射发布容器集群服务到外网节省公网IP



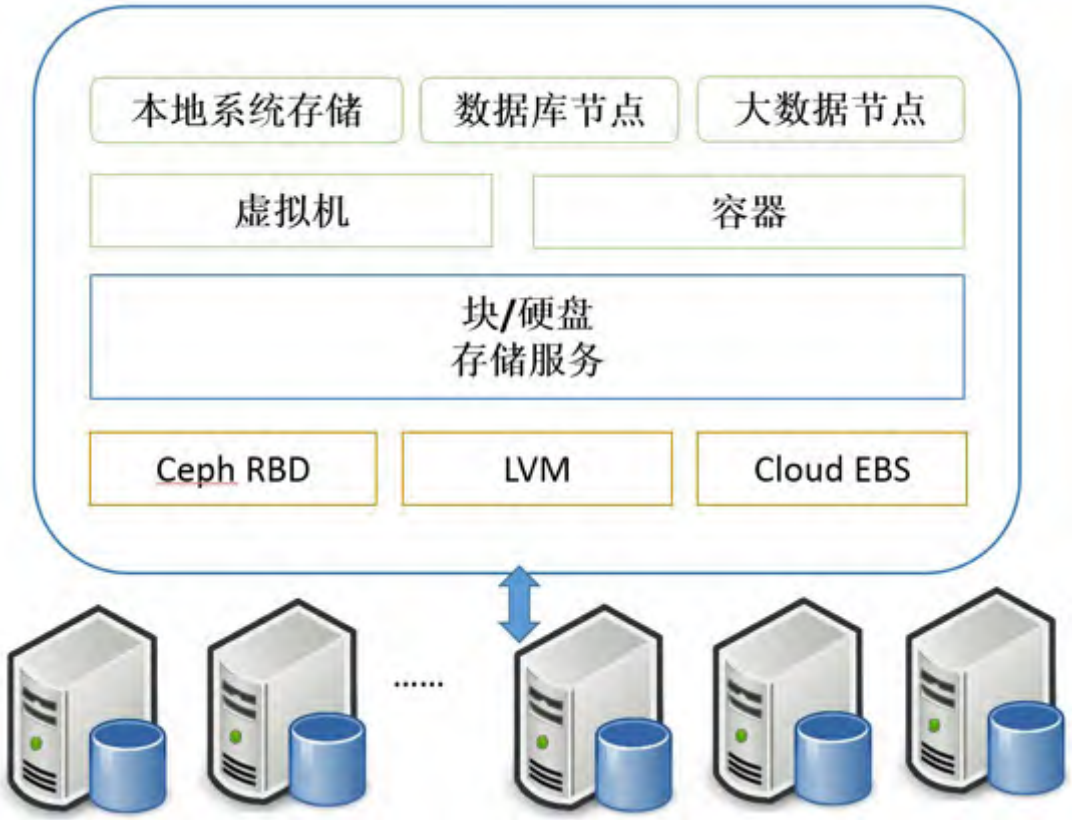
组合云平台的构建

- 基础技术的选型
- 跨IaaS的容器集群部署
- 容器和虚拟机的组合编排
- 跨容器和虚拟机的网络服务
- 跨容器和虚拟机的存储服务
- 云平台基本服务架构和应用部署架构
- 我们趟过的那些坑

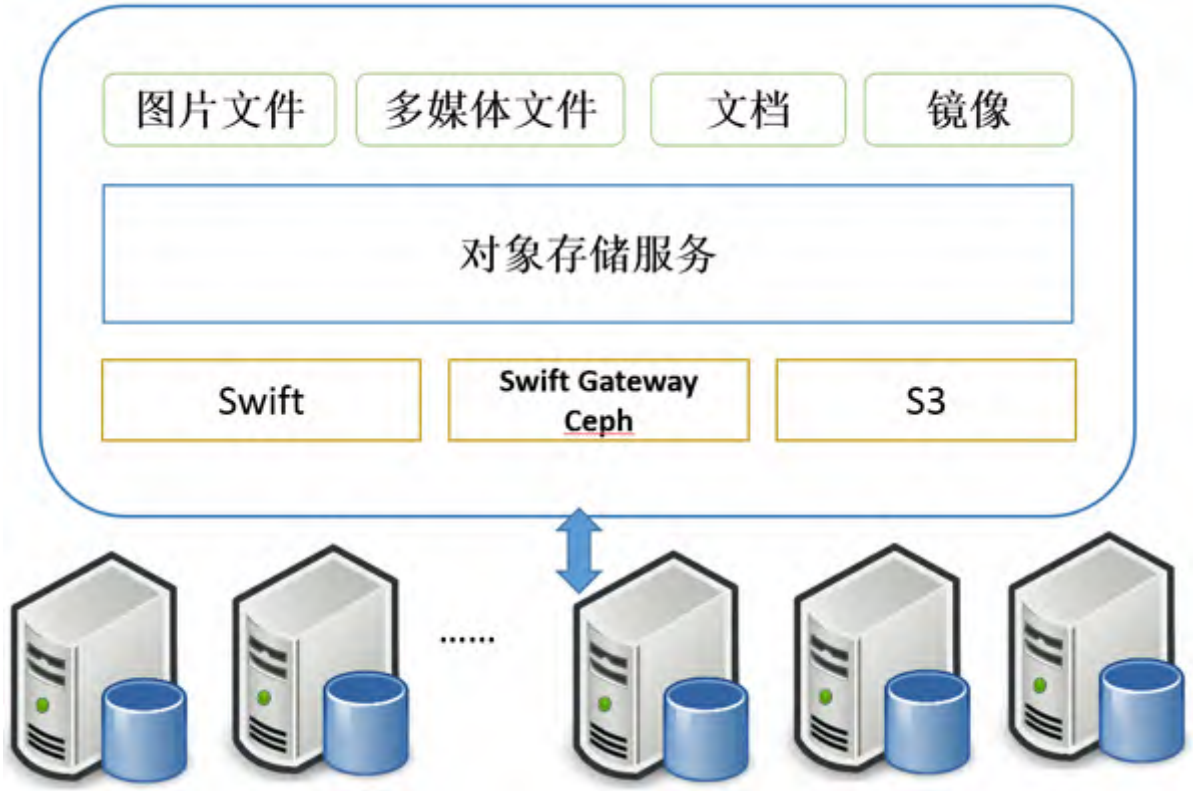
云平台的3种通用存储服务



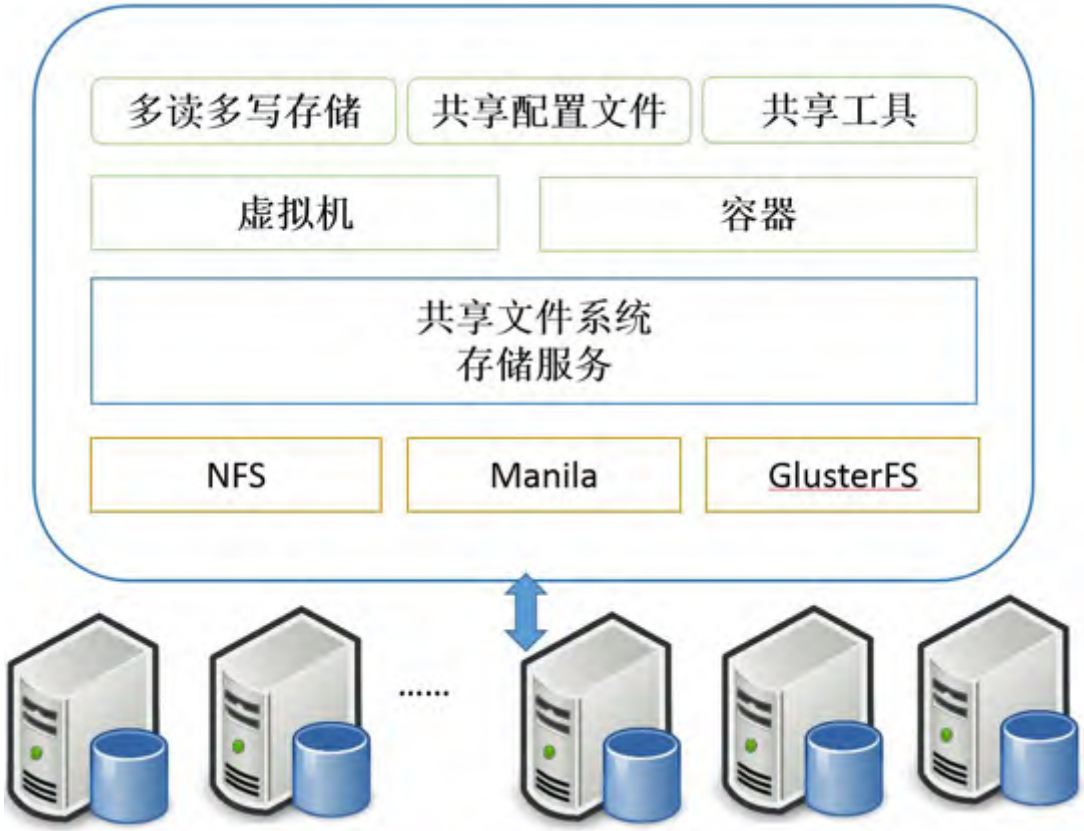
块存储服务、云硬盘



对象存储服务



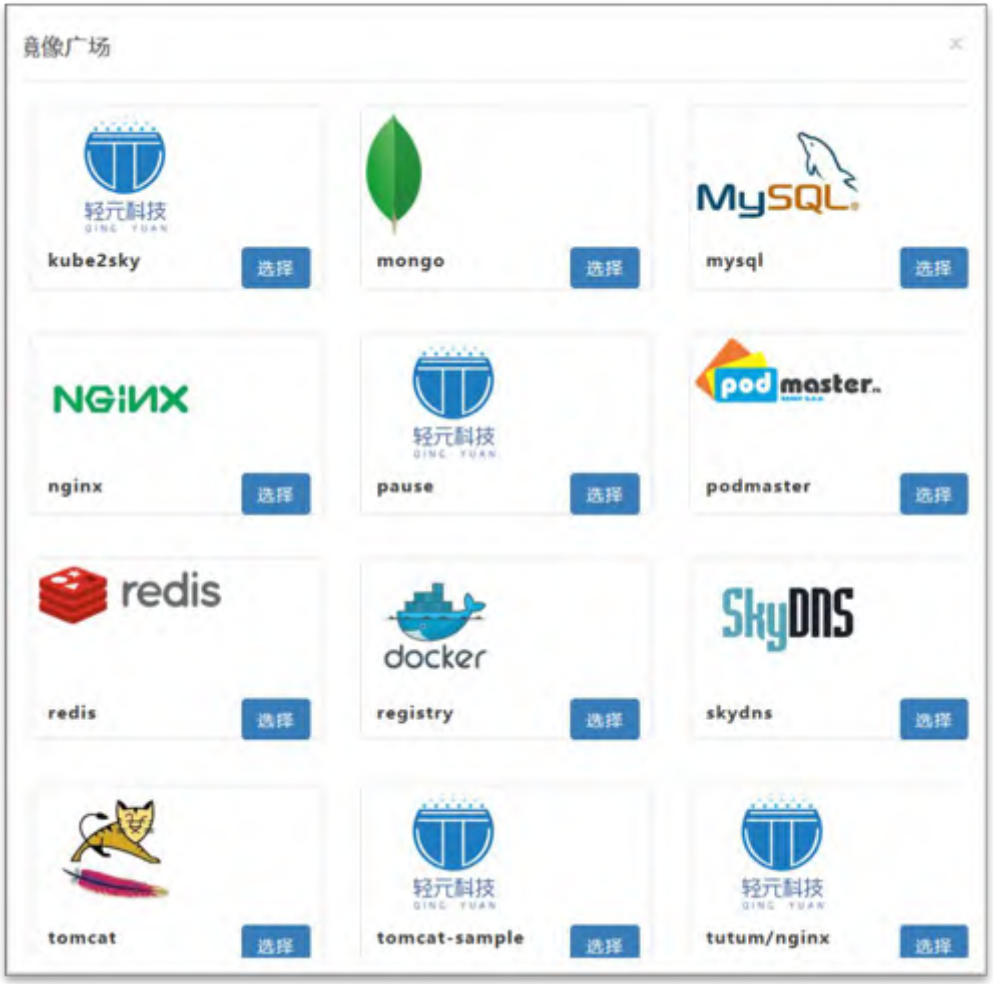
共享文件系统、共享目录



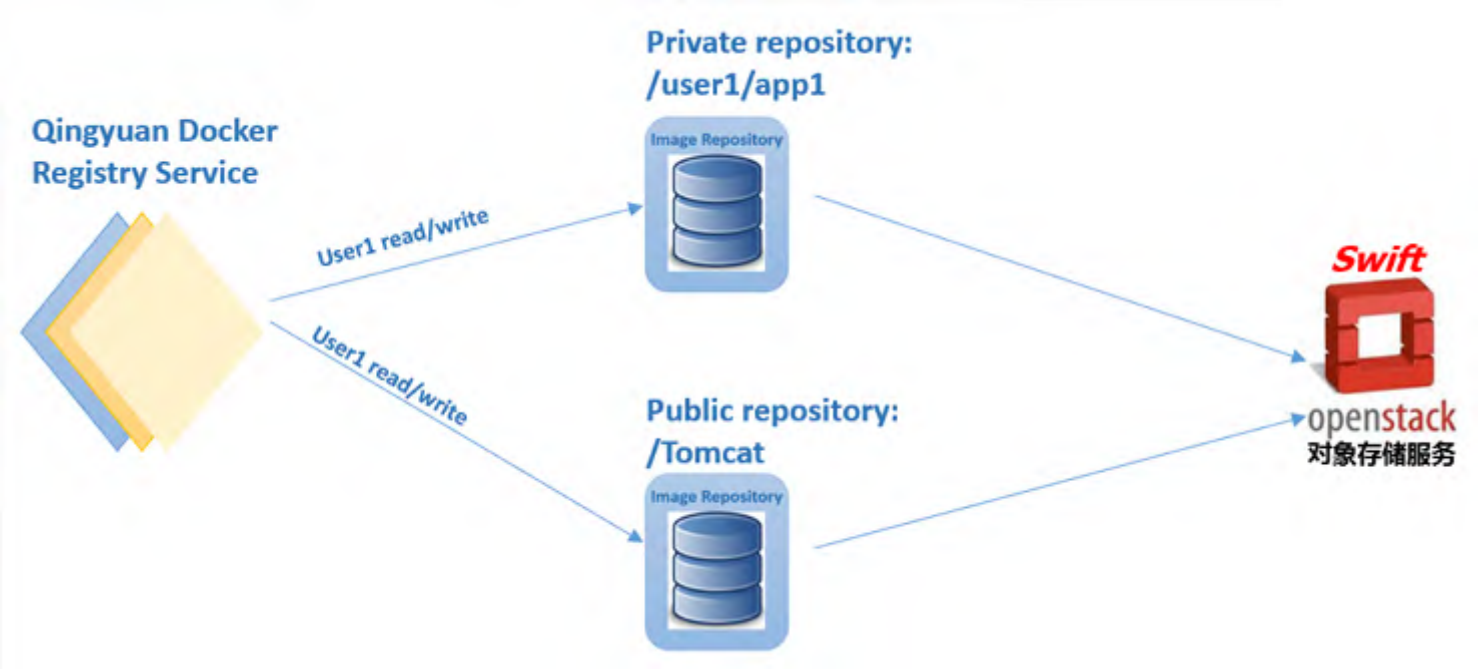
后端存储服务方案的选型

服务类型	可选技术	适用理由	注意事项
OpenStack 块存储	Ceph	高可靠，高可扩展，同Openstack集成较好，文档资料丰富	针对日志盘和网络的硬件配置和系统调优做仔细研究
Openstack块存储	LVM	定制使用本地存储的高性能虚拟机，例如用于Hadoop集群的虚拟机	对于扩展存储资源的调度缺乏自动化
Openstack/Kubernetes 对象存储	Swift	高可靠，高可扩展，支持纠删码，原生Openstack系统	Swift不是文件系统。需要不同于Ceph的后端配置管理和调优技能。
Openstack/Kubernetes 对象存储	Ceph	在块存储已经选用了Ceph的情况下，可以共用统一的后端配置管理平台和技能	在Openstack环境下性能不如Swift
Openstack/Kubernetes 文件共享系统	Openstack Manila/GlusterFS	根据需要可以定制NFS，CIFS服务，高可用	高可用备份方案仍在演化
Openstack/Kubernetes 文件共享系统	Cinder + NFS	非常容易快速搭建	缺乏成熟的高可用方案

镜像仓库的后端存储选型



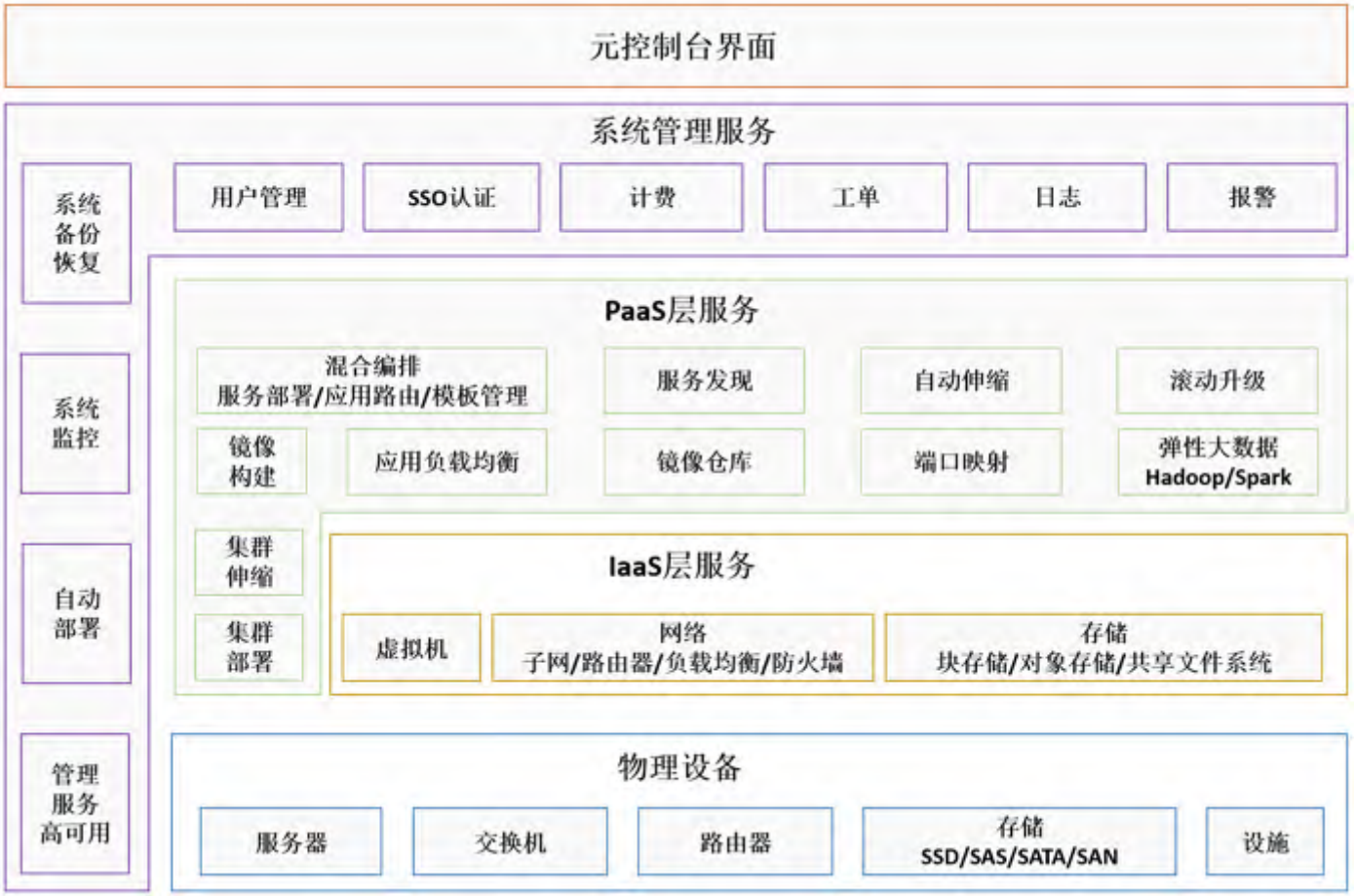
对象存储作为镜像仓库存储后端



组合云平台的构建

- 基础技术的选型
- 跨IaaS的容器集群部署
- 容器和虚拟机的组合编排
- 跨容器和虚拟机的网络服务
- 跨容器和虚拟机的存储服务
- 云平台基本服务架构和应用部署架构
- 我们趟过的那些坑

轻元数据中心操作系统服务架构



银行IT系统监控管理平台简介

项目背景

该银行的IT系统监控管理平台运行在VMware虚拟机环境上，在发展过程中遇到了这些挑战：

- *运行在VMware虚拟机上的应用服务随着业务量上升，软件授权费用越来越高*
- *历史软件包和历史日志存储在专业存储上，单位存储费用越来越高，且不易与大数据分析平台对接*
- *原有Oracle数据库系统面临去IoE和软件支持费用上升压力*

容器和虚拟机组合云服务对接弹性大数据

轻元科技帮助客户将原有的无状态应用迁移到容器集群管理平台上，将部分数据库迁移到MySQL和MongoDB，设备信息仍然留用Oracle数据库；将原有的软件包和历史日志存储迁移到了Swift对象存储上，并将弹性大数据分析平台与日志存储对接。

客户获得的收益

- 实现了无状态应用的容器化，大大降低资源使用率，降低软硬件费用。
- 验证实现了逐步去IoE的流程和架构设计。
- 文件存储迁移到了对象存储，大大降低平均单位存储费用。
- 实现了监控日志存储与弹性大数据平台的对接。

系统构架概况

某政府部门电子政务云项目简介

项目背景

该部门的业务系统（WebLogic)运行在物理机（曙光刀片机）上。与商业存储设备（杭州宏杉MacroSAN）和Oracle数据库的对接必须手动完成。该部门的IT系统面临了一下主要挑战：

- *不具备业务自动扩展功能，无法满足业务高峰需求。*
- *业务部署与业务升级复杂，无法适应快速的业务更新与调整。*
- *运维复杂，资源利用率低。*

对接传统数据库和国产SAN存储的云平台解决方案

轻元科技将OpenStack部署在了曙光刀片机上，并实现了与Oracle数据库和宏杉商业存储的对接。WebLogic服务整体迁移到了容器服务上，实现了PaaS（平台即服务）和IaaS（基础设施即服务）的统一管理。

客户获得的收益

- 实现了业务的自动收缩，轻松面对业务峰值的需求。
- 大大缩短了业务升级部署的时间。
- 降低了运维的复杂度，提高了资源利用效率。

系统构架概况

某创业园区专有云简介

项目背景

创业园区计划为园区内的3000多家企业提供一个免费的开发测试平台，加速区内企业的信息化，减轻园区内企业的IT投入负担，提升园区对优秀创业企业的吸引力。园区面临的挑战：

- *整体预算有限，如果采用虚拟机技术，无法公平地为所有企业提供免费资源；*
- *不具备任何运维云平台的技术能力。*

最大化资源使用率的容器云开发测试平台

轻元利用容器技术为创业园搭建了面对区内企业免费开放的开发测试平台。容器的粒度小、系统开销小，相比采用虚拟机技术整体资源使用高效率提高50%以上。创业园在很少的硬件投入下就满足了3000多家企业的需求。

客户获得的收益

- 采用容器技术，总体资源使用效率提升50%以上，用很少的预算就实现了满足了大量企业的需求。
- 将平台的运维托管给专业企业，降低了投入。
- 大大加快了区内企业的产品开发速度。

系统构架概况

地理信息系统私有云简介

项目背景

某地理信息系统公司后台的Hadoop大数据服务构建在物理机上，在发展过程中遇到了这些挑战：

- 基于物理机的部署方式，可扩展性差。硬件的资源使用效率低，无法灵活调度去满足一些其他业务的需求。
- 图片小文件的读写需求对Hadoop的Data Node压力过大，需要寻找更高效的存储方式。
- 长期公司考虑在异地再建一个数据中心，来提供数据安全性和服务稳定性，希望有一个统一管理的方案。

地理信息系统对接弹性大数据平台

轻元科技帮助客户将原有的大数据集群整体迁移到基于Hadoop的云平台上，将原有的HDFS存储迁移到了Swift对象存储上。并将公司部分需要动态伸缩的应用采用容器的形式发布。

客户获得的收益

- 实现了Hadoop集群的云化，使得Hadoop集群可以灵活扩展，同时释放了更多的资源来支撑企业的其他业务。
- 从HDFS迁移到对象存储，大大提高了图片等小文件的读写效率。
- 客户端应用迁移到了容器之上，实现了动态伸缩，降低了运维成本。
- 为未来多地数据中心的扩展奠定了坚实的基础。

系统构架概况

组合云平台的构建

- 基础技术的选型
- 跨IaaS的容器集群部署
- 容器和虚拟机的组合编排
- 跨容器和虚拟机的网络服务
- 跨容器和虚拟机的存储服务
- 云平台基本服务架构和应用部署架构
- 我们趟过的那些坑

Magnum创建baymodel失败

- 问题
  - 创建虚拟机镜像时没有指定os-distro属性
- 解决方案
  - 创建虚拟机镜像时一定要指定os-distro属性
    - glance image-create --name fedora-23-atomic --visibility public --disk-format qcow2 --os-distro fedora-atomic --container-format bare --progress --file ./Fedora-Cloud-Atomic-23-20151030.x86_64.qcow2
  - 对于已有的镜像，更新os-distro属性
    - glance image-update --os-distro fedora-atomic fedora-23-atomic

Bay创建后，kube-master中的kubernetes无法启动

- 问题
  - Magnum的部署脚本中需要从gcr.io/google_containers/...来下载镜像，由于众所周知原因，无法下载成功
- 解决方案
  - magnum-template-manage list-templates -d –p 确定magnum模板的安装位置，将./magnum/templates/下全部yaml和sh脚本中的gcr.io替换成墙内镜像仓库的地址。

Kubernetes的高可用部署问题

- 问题
  - 部署3个kube-master节点，3个节点的kube API Server不能同时工作，同时只能启动一个服务实例
- 解决方案
  - 升级到Kubernetes 1.2，3个kube API Server可以同时提供服务，scheduler和controller manager服务只有1个节点运行，其他两个节点处于备用状态
  - 前端用Haproxy做负载均衡，代理转发到3个API Server的后端IP

Kubernetes API Server的高可用部署的证书问题

- 问题
  - 当用HTTPS部署高可用API Server时客户端验证不通过
- 解决方案
  - 对3个API Server的后端服务用统一的证书
  - 证书签名中包含所有API Server的IP

Docker Registry对接Swift失败

- 问题
  - Docker Engine 1.10对接Swift有问题，向上push镜像失败
- 解决方案
  - 访问Docker Registry的Docker Engine版本升级到Dockers 1.11

今天分享的要点

- Kubernetes+OpenStack=容器和虚拟机组合服务
- 通过统一的容器集群部署服务跨平台部署Kubernetes——专为生产环境打造的容器集群系统
- 跨容器和虚拟机的网络服务：租户隔离、负载均衡、发布服务到公网访问、端口映射
- 跨容器和虚拟机的存储服务：块存储、对象存储、共享目录
- 完整数据中心操作系统软件栈：容器+虚拟机+弹性大数据平台

轻元科技
自主打造数据中心操作系统
谢谢

http://www.qingyuanos.com/opening.html
sales@qingyuanos.com
xwang@qingyuanos.com