

# UP600-OpenStack课程

Kevin Zou

# 云计算介绍



**WORKFORCE  
DEVELOPMENT PROGRAM**



**技术为王**

T: 400-700-0056 / 800-810-0056

北京 | 上海 | 深圳 | 沈阳 | 成都 | 西安 | 广州 | 大连 | 武汉

学员的增值就是我们的价值

# 云计算的定义

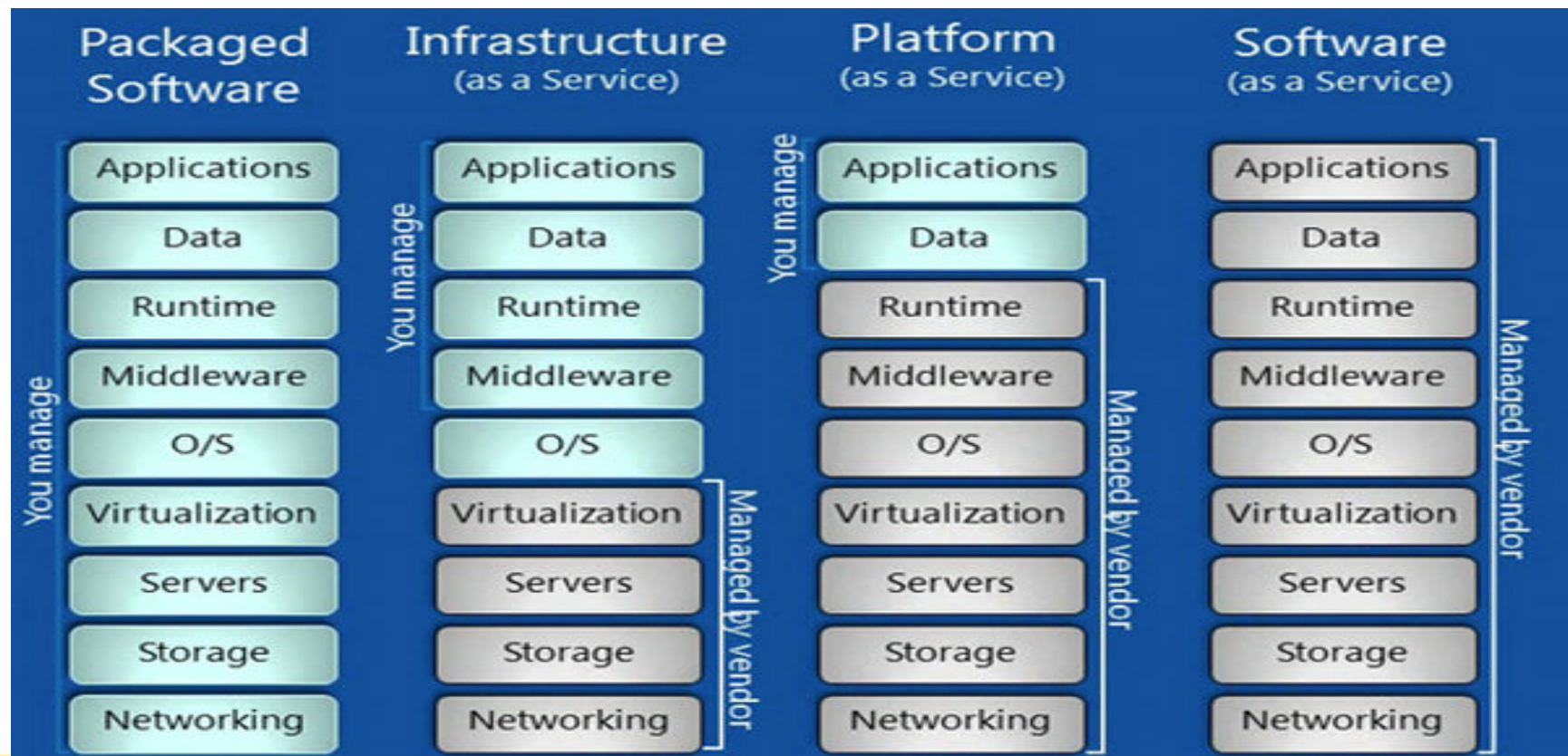
- 美国国家标准与技术研究院（NIST）定义
  - 一种模型
  - 可以实现随时随地、便捷并按需配置的资源池
  - 资源池中包括网络、计算、存储、应用及服务
  - 资源可以快速应用和释放
  - 可以使管理资源的工作量和工作交互降到最低程度

# 云计算的特征

- 自服务 ( Self-Service )
  - 允许云使用者配置带有计算资源的实例
- 全球可连接 ( Global Network Access )
  - 可直接通过因特网访问实例上的应用资源
- 多租户 ( Multitenancy )
  - 允许大量云使用者共享底层资源池

# 云计算的特征

- 弹性伸缩 ( Elasticity )
  - 可以横向或纵向扩展实例
- 遥测 ( Telemetry )
  - 资源使用可度量和监控，按需申请，动态收费



# IaaS: Infrastructure-as-a-Service ( 基础设施即服务 )

- 你不需要关心网络、存储、计算设备和场地
- 需要自己安装或选择操作系统
- 需要自己安装配置服务软件
- 需要自己管理系统和软件的安全性配置
- 灵活度最高、最操心

# PaaS: Platform-as-a-Service

## ( 平台即服务 )

- 你不需要关心任何硬件相关问题，也不用关心软件环境的搭建
- 需要自己选择合适的开发环境类型和数据库类型
- 需要自己定义开发架构和数据结构
- 需要自己上传程序代码和初始数据结构



# SaaS: Software-as-a-Service

## ( 软件即服务 )

- 你不需要考虑仅仅需要使用
- 选择你习惯的供应商
- 选择你习惯的操作方式
- 准备好浏览器，就像使用本地程序一样使用线上程序

# PaaS和IaaS比较

	PaaS	IaaS
开发环境	完善	简陋
应用支持度	优先支持	自定义安装
通用性支持	API开发	自定义安装
伸缩性支持	自动伸缩、支持负载均衡	手动伸缩、支持高可用性
整合度和整体成本	整合率高、成本更低	整合率低、需要人员维护
计费方式	按流量、连接、带宽等	按资源租用时间
学习难度	略高于本地开发	与本地开发类似

# OpenStack是什么

- OpenStack是由Rackspace和NASA共同开发的云计算平台
- OpenStack是一个由互联网高新技术公司组成的社区
- OpenStack是一个类亚马逊云平台的IaaS解决方案

# OpenStack是什么

- OpenStack是一个提供部署云操作平台的工具集
- OpenStack的主要代码是由Python写成
- 自2013年开始广泛的被互联网巨头自定义部署
- 开源共享且代码免费
- IBM、RedHat和VMWare均提供有偿技术支持

# OpenStack 新邪恶联盟



# OpenStack历史版本

- Openstack 采用6个月为一个开放周期
- 每个版本代号为首字母顺序按字典排序的某个城市名
- 新版本主要是阶段开发的汇总并将新技术固定下来

# OpenStack历史版本

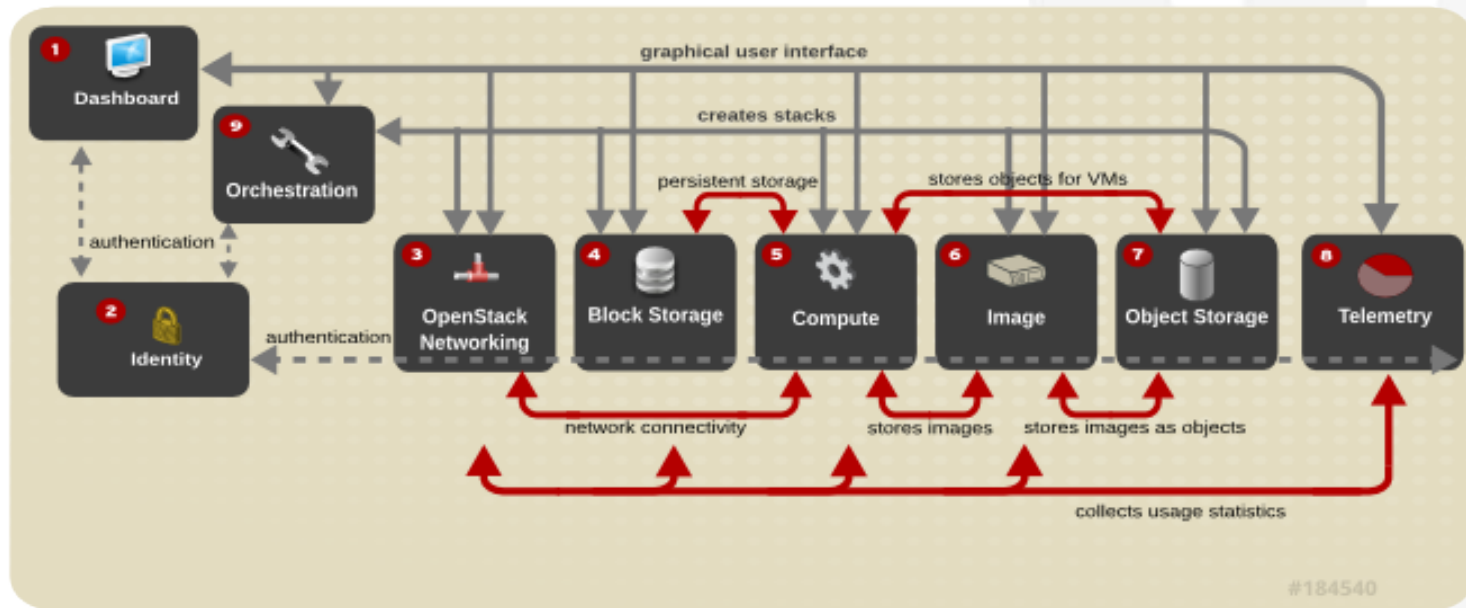
Openstack Austin	2010.10	Openstack Havana	2013.10
Openstack Bexar	2011.2	Openstack Icehouse	2014.4
Openstack Cactus	2011.4	Openstack Juno	2014.10
Openstack Diablo	2011.9	Openstack Kilo	2015.4
Openstack Essex	2012.3	Openstack Liberty	2015.10
Openstack Folsom	2012.10	Openstack Mitaka	2016.4
Openstack Grizzly	2013.4		

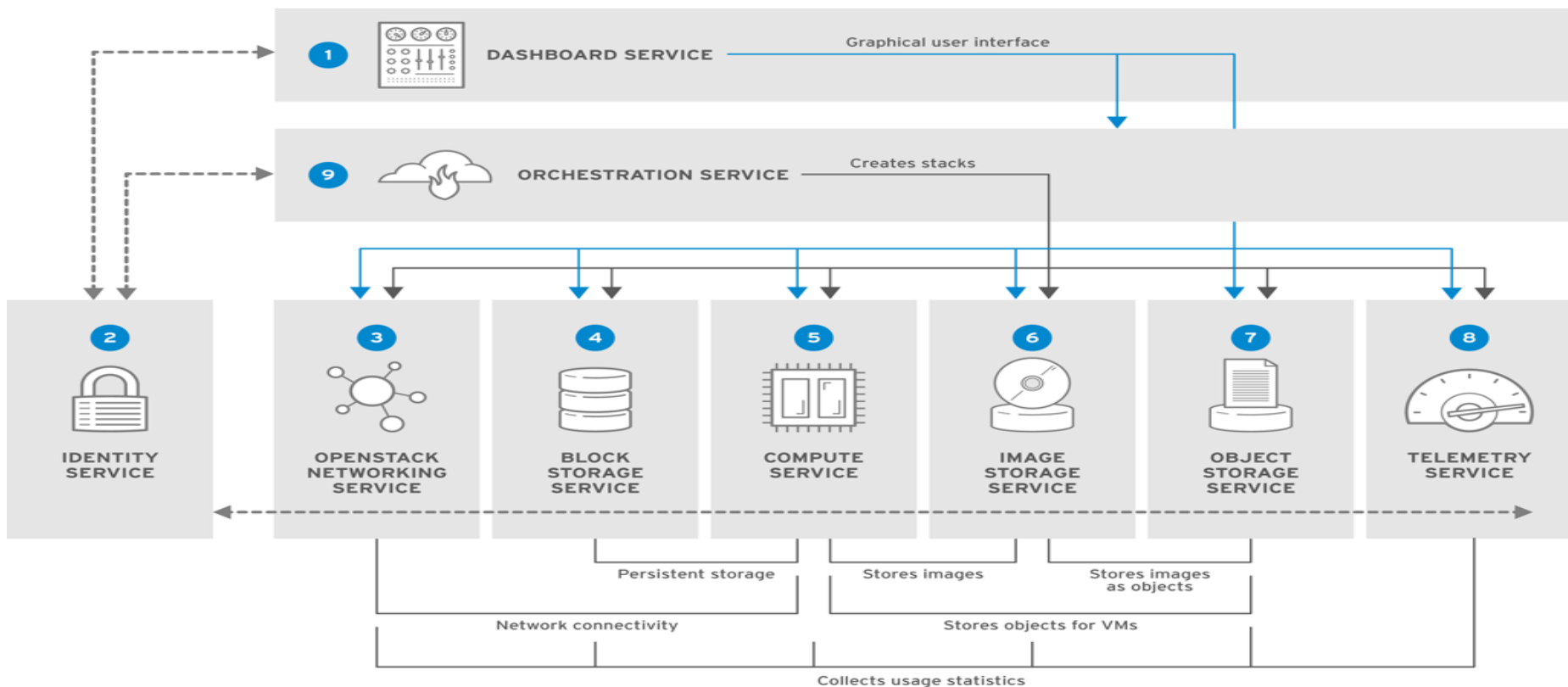
# OpenStack核心套件

- OpenStack是由多种功能性可替换技术型套件构成的
- 核心功能包括运行虚拟化、对象存储、块设备存储、虚拟网络、身份识别、镜像存储、WEB操作接口、数据采集和自动化部署等
- 每个组件都是由多个服务组成的，一个服务意味着运行着一个进程，根据OpenStack的规模，我们可以将所以服务安装在同一个机器上，也可以安装在多个机器上



# OpenStack核心套件结构





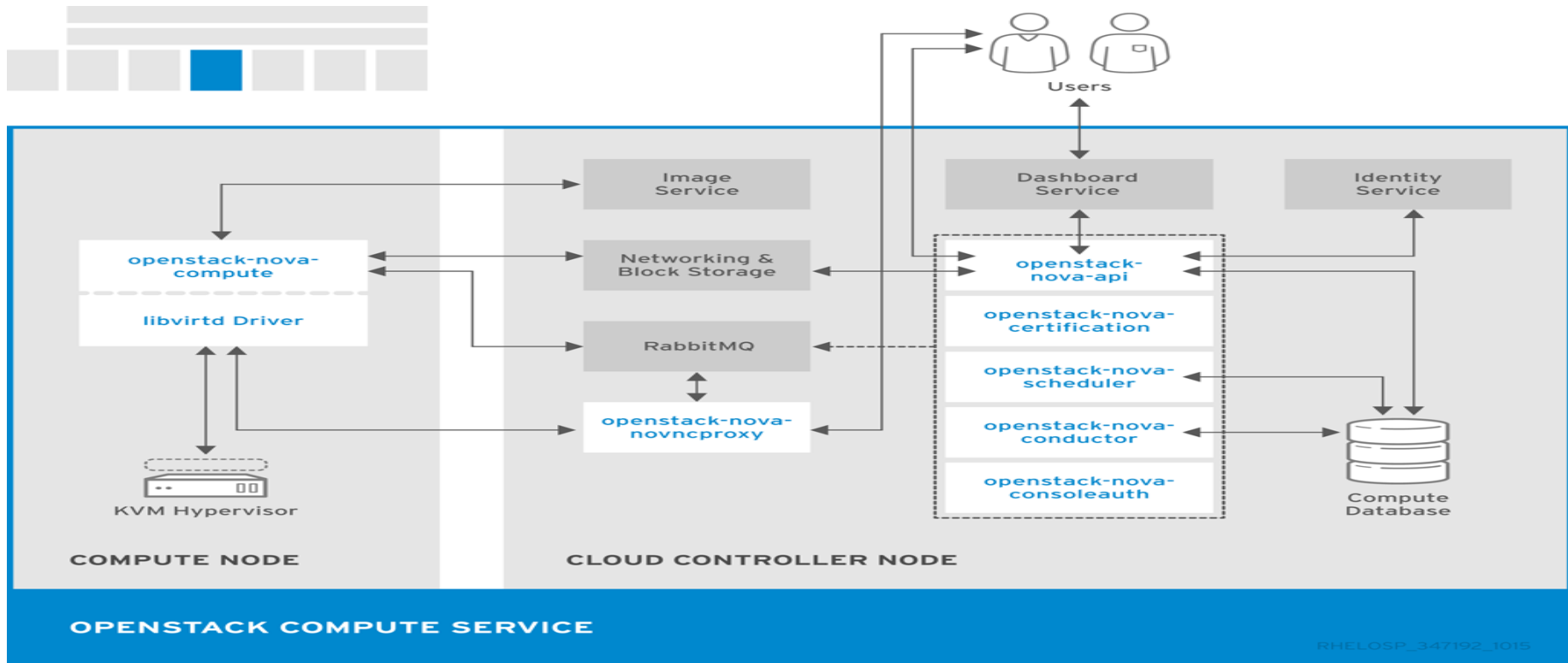
RHELOSP\_347192\_1015

# OpenStack核心套件-Nova

- 运行虚拟化套件—Nova
- Nova提供部署和管理虚拟机的功能
- 可以使用技术包括：Xen、KVM、VMWare、OpenVZ、Docker等
- 可以通过网页接口查看和管理云环境下多机及其上虚拟机的硬件资源运行状况

# Nova组件及进程

- nova-api
  - 对外提供服务的窗口，接收并响应来自用户的调用
- nova-compute
  - 物理机服务进程，执行一些与虚拟机相关操作
- nova-scheduler
  - 接收创建虚拟机的请求，并决定在哪台物理机上启动该虚拟机的调度器
- nova-conductor
  - 处于nova-computer与nova-db之间的一个组件，从安全考虑避免二者直接通信
- nova-console和nova-consoleauth
  - Nova提供的控制台服务，允许最终用户通过代理服务器访问虚拟机的控制台
- nova-cert和nova-objectstore
  - 提供x509验证管理服务和Glance中注册的S3接口服务



# OpenStack核心套件-Swift

- 对象存储套件—Swift
- Swift提供分布式存储大数据功能
- Swift可以存储任意类型数据，例如虚拟机镜像、文本文件、网站图片和虚拟磁盘
- Swift扩展性强、高性能、高可靠、跨平台
- 有被Ceph替换的趋势

# Swift组件及进程

- swift-proxy
  - 处于Swift系统内外部之间，负责接收API或HTTP请求，为了提高系统性能，一般会与memcached一起部署
- swift-account
  - 仅仅用于账户管理
- swift-container
  - 管理容器或文件夹得映射关系
- swift-object
  - 管理在存储节点上的实际对象



http(s)



Account Database



Container Database



Object Store

openstack-swift-account

openstack-swift-container

openstack-swift-object

service  
storage  
nodes



openstack-  
swift-proxy

Identity API

Identity Service

object  
storage  
API

Dashboard Service

object  
storage  
API

Compute Service

object storage API

Image Service

## OPENSTACK OBJECT STORAGE SERVICE

RHELOSP\_347192\_1015

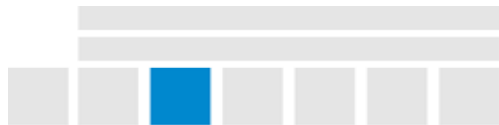


# OpenStack核心套件-Cinder

- 块设备存储套件—Cinder
- Cinder支持商业化的企业存储平台，例如NetApp、EMC、Nexenta等
- Cinder支持开源存储技术平台，例如iSCSI、FCoE、LVM等
- Cinder为OpenStack中的虚拟机提供附加虚拟硬盘

# Cinder组件及进程

- cinder-api
  - 接收外部API请求，并将请求交给cinder-volume处理
- cinder-volume
  - 负责与底层块存储服务交互，相应读写设备请求，底层块存储服务可替换
- cinder-scheduler
  - 当存储节点为多个时，负责找到最佳节点创建volume
- cinder-backup
  - 接受外部API请求，备份cinder卷数据，后端存储服务可替换



Users



Database

Message Broker

openstack-cinder-scheduler

openstack-cinder-api

openstack-cinder-backup

openstack-cinder-volume

Volume Backup  
Repository

Volume Back Ends

OPENSTACK BLOCK STORAGE SERVICE

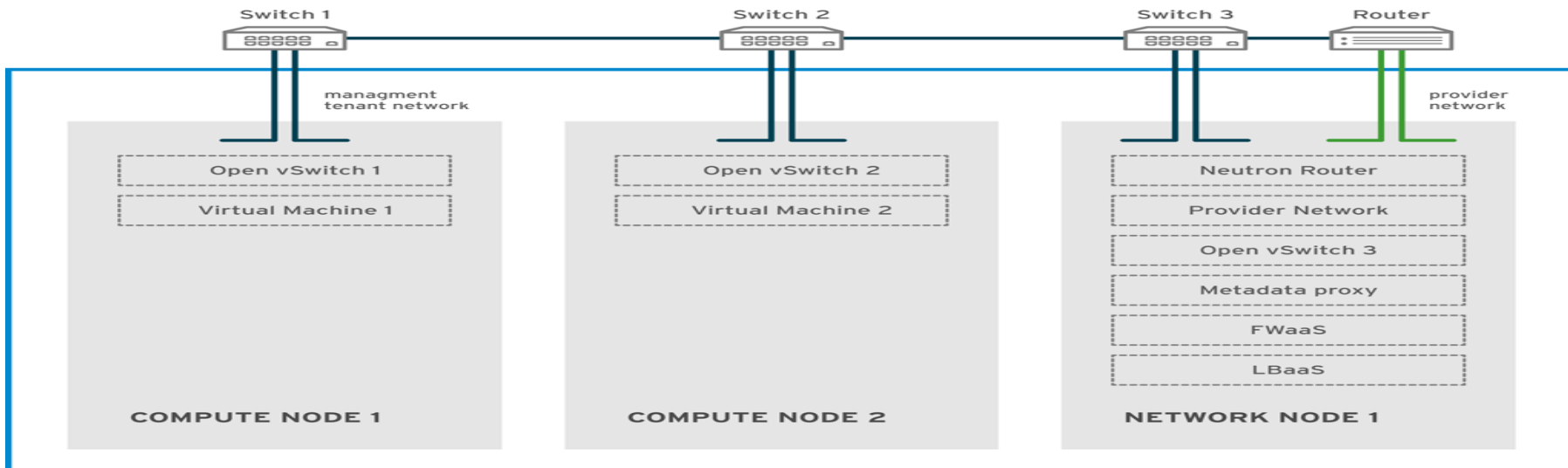
RHELOSP\_347192\_1015

# OpenStack核心套件-Neutron

- 虚拟网络组件—Neutron ( Quantum )
- Neutron是一个可扩展、即插即用、通过API管理云平台网络的套件
- Neutron是OpenStack的网络抽象层，以可代换的方式使用OpenvSwitch、Cisco UCS/Nexus、Linux Bridge、NEC OpenFlow等

# OpenStack核心套件-Neutron

- Neutron维护云平台虚拟机之间跨物理服务器的虚拟网络
- Neutron创建跨区域跨物理节点的虚拟2-3层网络环境
- Neutron为云平台中虚拟机提供的动态网络配置协议（DHCP）支持

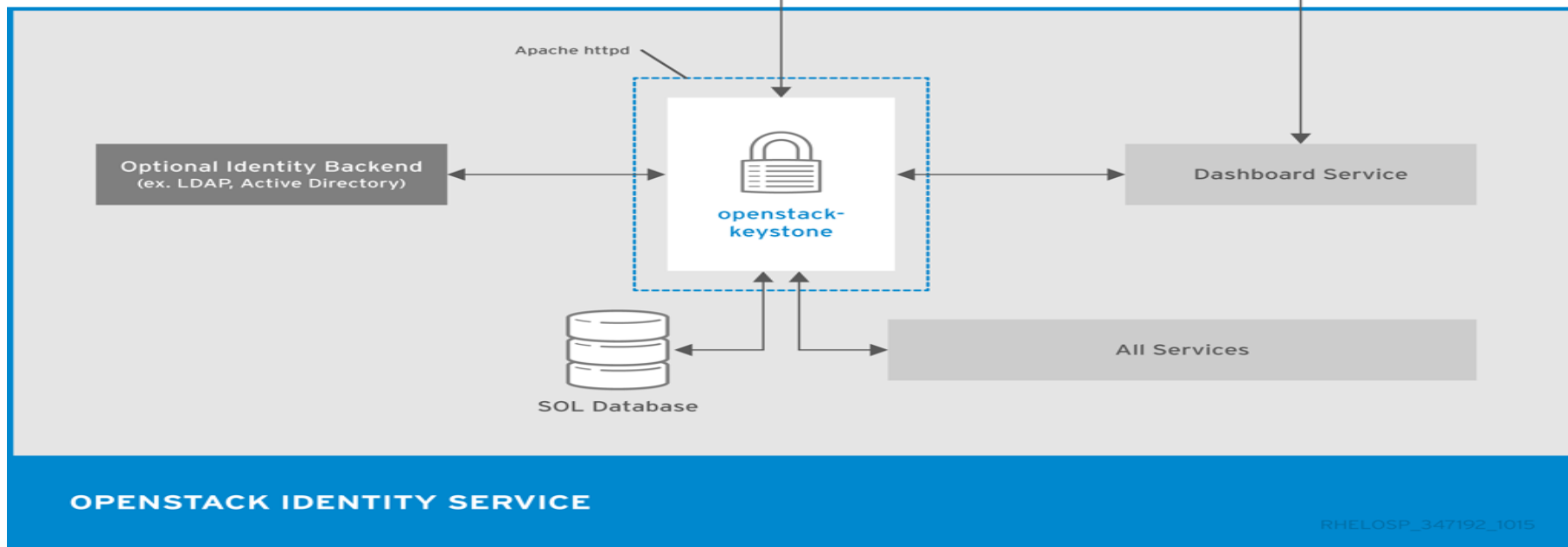


## OPENSTACK NETWORKING SERVICE

RHELOSP\_347192\_1015

# OpenStack核心套件-Keystone

- 身份识别套件—Keystone
- Keystone是OpenStack的核心认证组件
- Keystone提供中央目录集中认证服务
- Keystone提供多种认证方式，例如账号密码、令牌（Token）、Kerberos等



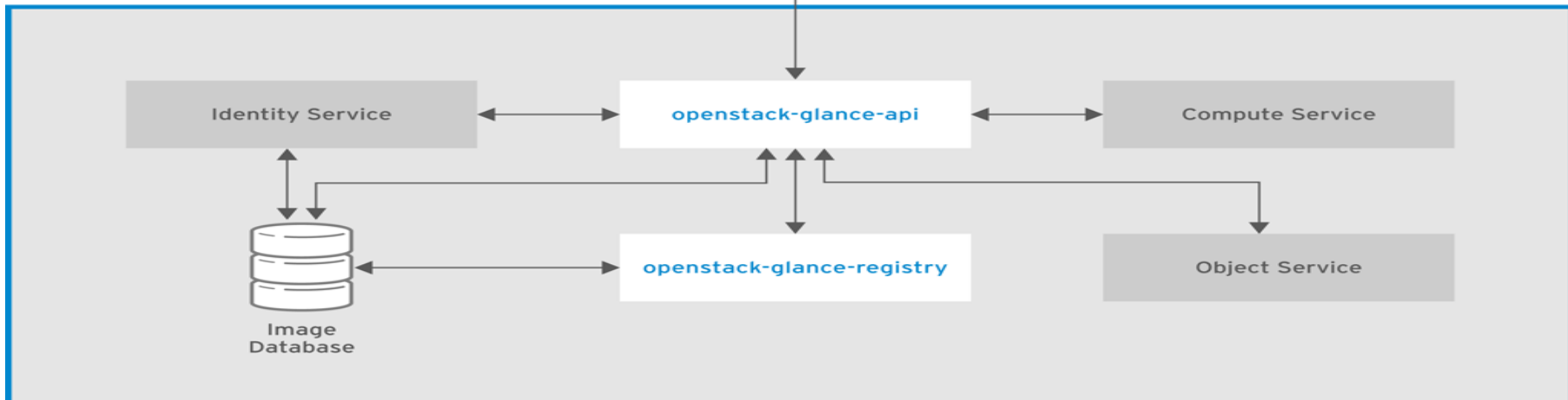


# OpenStack核心套件-Glance

- 镜像存储套件—Glance
- Glance提供虚拟机镜像文件的注册、查找和交付功能
- Glance统一了Openstack云环境系统的虚拟机模板及安装源
- Glance的存储源支持分布式对象存储



Users

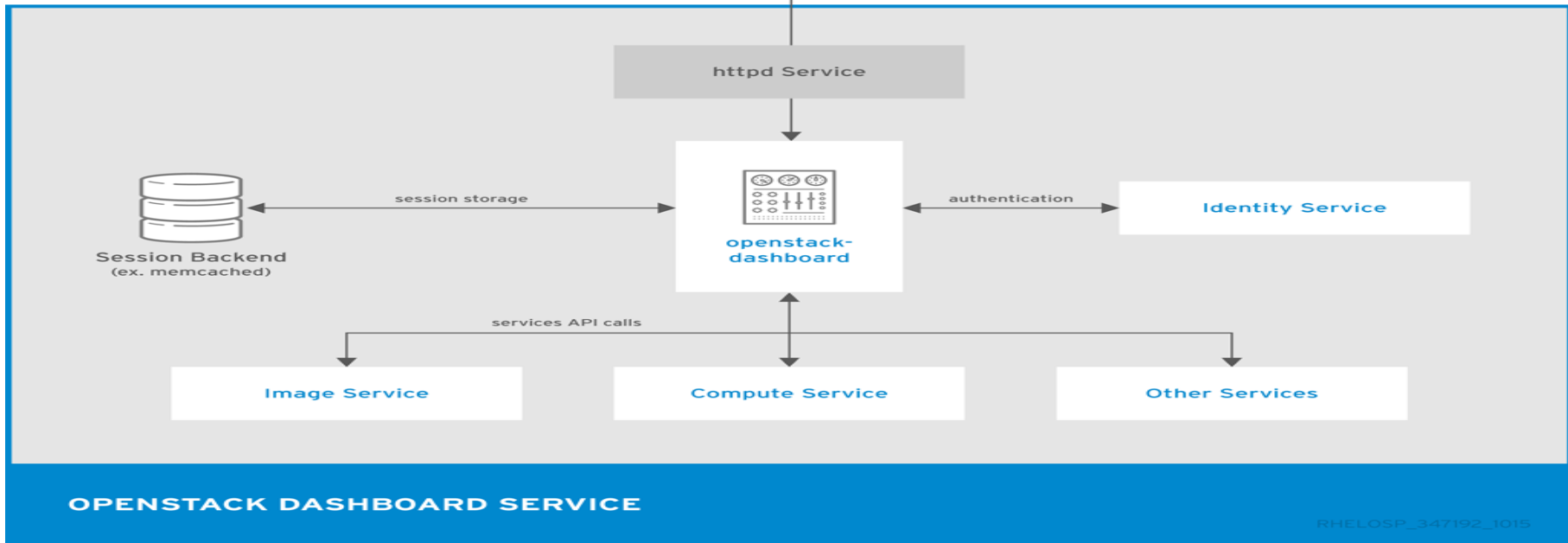


## OPENSTACK IMAGE STORAGE SERVICE

RHELOSP\_347192\_1015

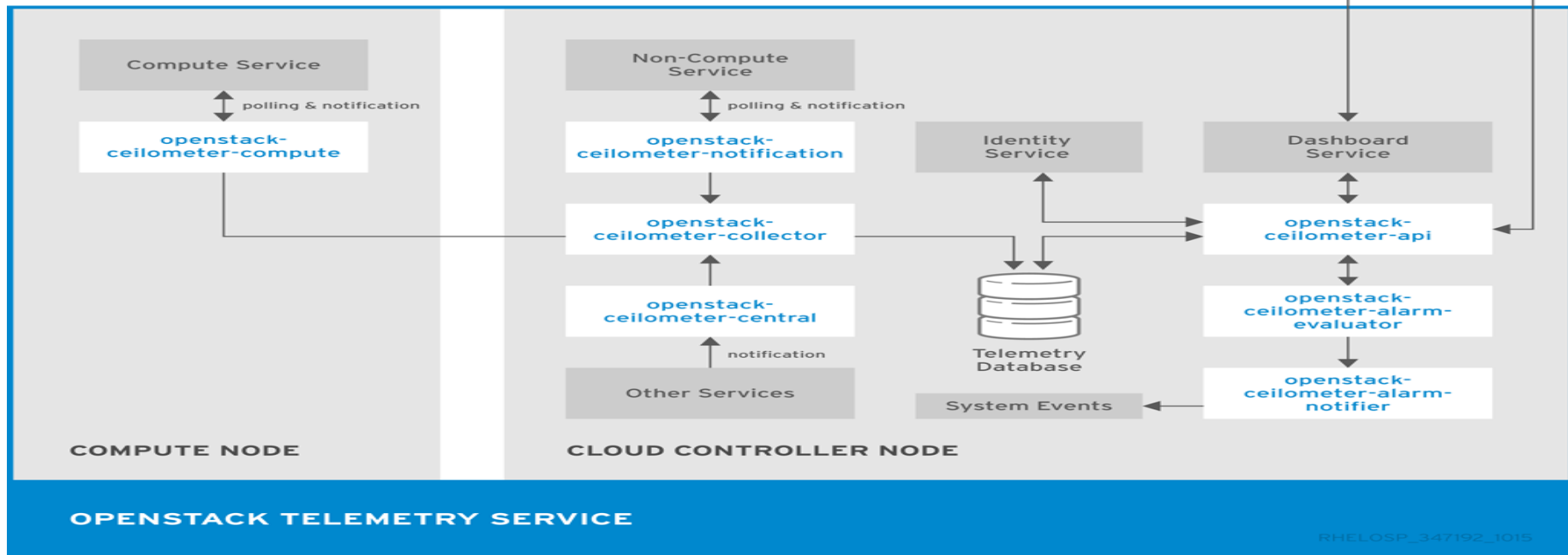
# OpenStack核心套件-Horizon

- WEB操作接口—Horizon
- Horizon为Openstack的管理提供一个图形化的页面接口
- Horizon是一个开放可扩展的网页APP
- Horizon统一了部署和管理云端服务和资源的方法



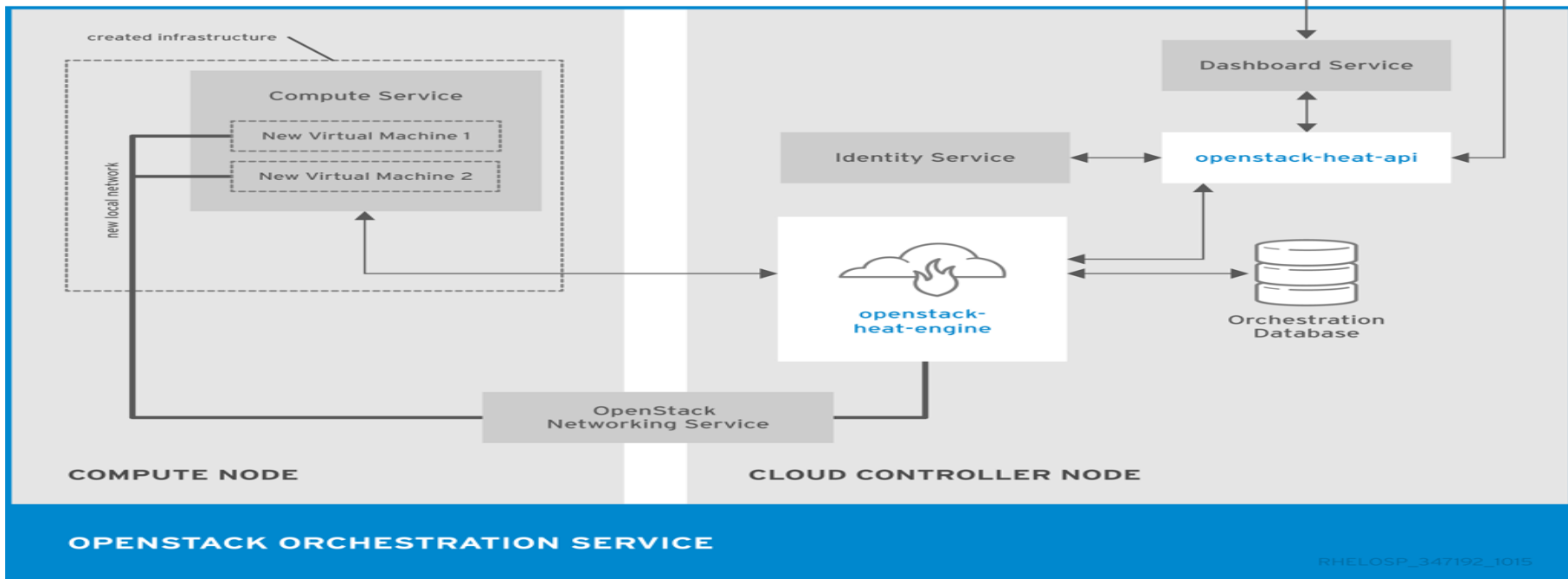
# OpenStack核心套件-Ceilonmeter

- 数据采集套件—Ceilonmeter
- Ceilonmeter提供Openstack中物理及虚拟硬件运行数据采集功能
- Ceilonmeter为Openstack监控数据及按需计费提供依据



# OpenStack核心套件-Heat

- 编配计划服务—Heat
- Heat旨在提供Openstack中业务流程计划的编配和实例及网络的自动化部署
- Heat会主动调用OpenStack中其它套件创建资源，可以根据Heat模板创建一个Stack，此Stack就是这个业务项目的别称



RHELOSP\_347192\_1015



# OpenStack 核心组建总结

- Horizon: 用于创建和管理实力的web浏览器用户接口
- Keystone : 身份验证和服务授权框架
- Neutron : 网络构建服务
- Cinder : 运行实例的永久块存储
- Nova : 节点上的虚拟机控制调度程序
- Glance : 虚拟机映像文件管理服务
- Swift : 分布式对象存储服务
- Ceilometer : 用以收集报表及资源使用计费服务
- Heat : 基于模板的虚拟机及虚拟网络部署服务

# OpenStack单节点与多节点

- OpenStack单节点模式—AllinOne
- OpenStack多节点分类
  - 控制节点 ( Controller Node )
  - 计算节点 ( Computer Node )
  - 网络节点 ( Network Node )
  - 存储节点 ( Storage Node )

# OpenStack AllinOne 单节点

- 主要用于测试和体验
- 安装部署方便且各组件直接本地通信
- 后期可添加更多功能节点迁移为多节点模式
- AllinOne=Controller + Computer + Network + Storage

# OpenStack 多节点 Controller

- 控制节点（Controller）主要由以下服务组成
  - 身份认证服务（Keystone）
  - WEB操作接口（Horizon）
  - 数据库服务（Mysql/Mariadb）
  - 数据采集服务（Ceilonmeter）
  - 编配计划服务（Heat）
  - 其他组件的控制服务（Neutron .....）

# OpenStack 多节点 Computer

- 计算节点 ( Computer ) 主要由以下服务组成
  - 计算服务 ( Nova )
  - 网络代理服务 ( Open vSwitch)
  - 其他各种组件代理服务 ( Ceilometer Agent .....)

# OpenStack 多节点 Network

- 网络节点 ( Network ) 主要由以下服务组成
  - 虚拟网络服务 ( Neutron )
  - 网络代理服务 ( Open vSwitch )
  - 二层和三层虚拟化插件服务 ( Network ML2 )
  - 其他各种组件代理服务 ( Ceilometer Agent .....)

# OpenStack 多节点 Storage

- 存储节点（Storage）主要分类如下
  - 镜像存储节点（Glance）
  - 块存储节点（Cinder）
  - 对象存储节点（Swift）
  - 分布式存储节点（Ceph）

# OpenStack安装方法

- 源代码编译安装—官方不推荐
  - 开发人员使用DevStack集成环境
- 二进制包管理器安装—官方推荐
  - RPM包管理方式—YUM源安装
  - DEB包管理方式—APT源安装
- 第三方安装管理器—简单快捷
  - openstack-packstack 基于命令行快速安装工具 ( 利用puppet)
  - Foreman 基于Web的快速安装工具集 ( 利用puppet)
  - Mirantis的fuel 安装方式 ( 类似Kickstart的方法 )



# OpenStack基础操作系统安装

- 选择物理机安装或虚拟机安装
- 虚拟机安装注意事项
- 安装精简操作系统Red Hat Enterprise Linux 7
- 为满足 OpenStack 安装配置磁盘分区和本地卷
- 关闭NetworkManager和FireWalld服务
- 打开network服务，并保证开机启动

# Red Hat OpenStack 项目说明

- Red Hat deploy OpenStack 开源公众项目
  - RDO
- Red Hat Enterprise Linux OpenStack Platform 服务收费项目
  - RHOSP

# Red Hat Deploy OpenStack

- RDO 项目主页-<http://openstack.redhat.com>
- 面向开发和测试人员
- Bug多且同版本Bug修复慢
- 界面为标准OpenStack Dashboard
- 没有在线技术支持

# Red Hat Deploy OpenStack

- 安装顺畅性差且包依赖关系复杂
- 新技术新特性跟随紧密
- 个人体验标准OpenStack首选
- 软件仓库EPEL和 fedorapeople
- <https://repos.fedorapeople.org/repos/openstack>

# Red Hat Enterprise Linux OpenStack Platform

- RHOSP主页
  - <https://access.redhat.com/products/red-hat-enterprise-linux-openstack-platform>
- 面向企业应用
- Red Hat 商业支持
- 稳定但不是最新版本 OpenStack , 但可持续支持周期长

# Red Hat Enterprise Linux OpenStack Platform

- 使用Red Hat定制Dashboard界面
- 安装顺畅且包依赖关系稳定
- 与Red Hat 其它产品结合度高
- 企业配置私有云 IaaS 首选之一
- 软件仓库RHN，更新及时

# 安装AllInOne的OpenStack环境

- 选择安装OpenStack类型
- 配置网络安装源或本地安装源
- 确认关闭本地 NetworkManager 及FireWalld服务
- 确认开启本地network服务
- 通过openstack-packstack安装OpenStack组件
- 生成自动应答文件，并使用应答文件安装

# Packstack安装过程

- 配置SSH-Key方便执行脚本和命令
- 使用Puppet工具安装配置环境
- Packstack生成本地安装日志
- 安装所需软件包并生成配置及安装脚本
- 意外终止安装过程排错



# 自动应答文件说明

- 自动应答文件使用方法
- 自动应答文件设置结构
- 组件安装筛选
- 用户密码配置
- 不同类型节点配置
- 网络配置

# 应答文件修改内容

- CONFIG\_DEFAULT\_PASSWORD=uplooking
- CONFIG\_SWIFT\_INSTALL=y
- CONFIG\_NTP\_SERVERS=172.25.254.254 此处ip为网络中的时间服务器ip
- CONFIG\_LBAAS\_INSTALL=y
- CONFIG\_NEUTRON\_FWAAS=y
- CONFIG\_NAGIOS\_INSTALL=n
- CONFIG\_KEYSTONE\_ADMIN\_PW=uplooking 此处定义了admin用户密码
- CONFIG\_CINDER\_VOLUMES\_CREATE=n 需要手动配置cinder卷组
- CONFIG\_HORIZON\_SSL=y
- CONFIG\_PROVISION\_DEMO=n 不安装默认测试预装环境

# 本地网络配置

- br-ex、br-int、eth0 说明
- 外联网桥br-ex设置
- 本地网桥端口eth0设置
- 本地网络测试

# 本地网络配置

- /etc/sysconfig/network-scripts/ifcfg-br-ex

- DEVICE=br-ex
- BOOTPROTO=static
- DEFROUTE=yes
- IPV4\_FAILURE\_FATAL=yes
- IPV6INIT=no
- ONBOOT=yes
- TYPE=OVSIntPort
- OVS\_BRIDGE=br-ex
- DEVICETYPE=ovs
- IPADDR=172.25.N.10
- NETMASK=255.255.255.0
- DNS1=172.25.254.254
- GATEWAY=172.25.N.254

DEVICE=br-ex

TYPE=OVSIntPort

OVS\_BRIDGE=br-ex

DEVICETYPE=ovs

不建议使用DHCP获取地址

# 本地网络配置

- /etc/sysconfig/network-scripts/ifcfg-eth0

- NAME=eth0
- ONBOOT=yes
- TYPE=OVSPort
- DEVICETYPE=ovs
- OVS\_BRIDGE=br-ex

TYPE=OVSPort

DEVICETYPE=ovs

OVS\_BRIDGE=br-ex

如果有可能建议配置更多网卡并bond起来

# 本地网络配置

- 查看当前设备配置状态
  - Ifconfig
- 查看本地OpenVSwitch配置状态
  - ovs-vsctl show
- 查看本地网络名字空间配置状态
  - ip netns show

# OpenStack命令行环境

- OpenStack用户命令行环境配置文件
  - keystonec\_USERNAME
- 配置内容
  - export OS\_USERNAME=admin
  - export OS\_TENANT\_NAME=admin
  - export OS\_PASSWORD=uplooking
  - export OS\_AUTH\_URL=http://172.25.N.10:5000/v2.0/
  - export OS\_REGION\_NAME=RegionOne

用户名  
项目名  
密码  
验证URI  
区域名

# Dashboard界面说明-Identity

- Projects
  - Project Members
  - Quota
- Users
  - Role

项目

项目成员

项目资源配额

用户

用户权限



# Dashboard界面说明-Admin

- Overview 概括
- ResourceUsage 资源使用情况
- Hypervisors 虚拟机管理器
- Host Aggregates 主机集合
- Instances 实例
- Volumes 云硬盘
- Flavors 云主机类型
- Images 镜像
- Networks 网络
- Routers 路由
- Defaults 默认值
- System Information 系统信息

# Dashboard界面说明-Project

- Compute
  - Overview
  - Instance
  - Volumes
  - Images
  - Access & Security

- Network
  - Network Topography
  - Network
  - Routes
  - Load Balancers
  - Firewalls

- Object Storage
  - Containers

## 计算

概括  
实例  
云硬盘  
镜像  
访问 & 安全

## 网络

网络拓扑  
网络  
路由  
负载均衡  
防火墙

## 对象存储

容器

# Dashboard操作说明-网络创建

- Admin用户登录Dashboard
- 修改admin密码
- 创建新项目(uplooking)和新用户 ( kevin )
- 为新用户 ( kevin ) 设置额外权限
- 创建全局虚拟公网网络 ( all-pub )
- 配置全局虚拟公网IP地址池 ( net1 )
- 授权虚拟公网网络为外部网络 ( kevin配置外部网络时 )

# Dashboard操作说明-网络创建

- 切换用户Kevin管理Uplooking项目
- 创建本地项目内网网络 ( uplooking-pri1 )
- 配置本地项目内网虚拟IP地址池 ( pri-net1)
- 创建本地项目内网到虚拟公网路由 ( route1 )
- 创建本地路由接口连接uplooking-pri1网络
- 创建本地路由网关接口并测试

# Dashboard操作说明-上传镜像

- Admin及普通用户均可上传镜像文件
- 可通过http://的URL指定镜像文件位置
- 可通过浏览器上传本地镜像文件
- 只有Admin用户能够设置镜像为 “Public”
- 设置 “Protected” 项后，需主动撤销才能删除镜像

# Dashboard操作说明-云主机类型

- 云主机类型就是创建虚拟机实例的资源分配模板
- Root Disk—实例/dev/vda设备image的克隆
- Ephemeral Disk—实例/dev/vdb设备数据磁盘
  - 默认设置为ext3文件系统，并挂载在实例/mnt目录
- Swap Disk—实例/dev/vdc设备虚拟内存磁盘
  - 默认设置为swap格式，并自动swapon

# Dashboard操作说明-登陆密钥对

- Project-Compute-Access&Security
  - Key Pairs
    - Create Key Pair
    - Import Key Pair
- 创建密钥对后只有一次下载私钥的机会
- 可以上传本地预创建的公钥
- 在创建实例时可选注入公钥

# Dashboard操作说明-浮动IP

- Project-Compute-Access&Security
  - Floating IPs
    - Allocate IP to Project
    - Release Floating IPs
- 浮动IP也可以在实例创建后动态分配给实例



# Dashboard操作说明-安全组

- Project-Compute-Access&Security
  - Security Groups
    - Create Security Group
    - Delete Security Groups
    - Manage Rules
- 安全组控制实例网络进出访问规则
- “default” 安全组不可删除
- Remote
  - CIDR 地址标识 ( 针对具体IP地址访问 )
  - Security Group 安全组标识 ( 针对安全组中的实例访问 )

# Dashboard操作说明-实例

- Project-Compute-Instance
  - Launch Instance
  - Soft Reboot Instances
  - Terminate Instances
  - Overview
  - Log
  - Console
  - Action Log

创建实例

软重启实例

销毁实例

# Dashboard操作说明-实例信息

- Instance Name
- Image Name
- IP Address
- Size
- Key Pair
- Status
- Availability Zone
- Task
- Power State
- Time since created
- Actions

实例名称（云主机名称）

镜像名称

IP地址（内网地址和浮动公网地址）

云主机类型（配置）

密钥对名称

状态

可用域（目前只有nova）

任务（迁移、扩容、重置）

电源状态

创建以来的时间

动作

# Dashboard操作说明-实例操作

- 绑定浮动IP、解除绑定浮动IP
- 使用安全组及其规则控制外部对实例内服务的访问
- 挂起实例、重启实例、关闭实例
- 调整云主机大小
- 创建实例的快照

# Dashboard操作说明-实例迁移

- 需要两个以上的运算节点 ( Computer Node )
  - 可以通过packstack添加新的运算节点
- 在线迁移需要运算节点共享存储
  - /var/lib/nova/instances目录
- 注意BUG-nova ssh bug
  - 权限和可登录Shell

# 手动添加、迁移节点

- ALLinOne环境安装节点作为Controller节点
- 手动添加新节点，深入理解OpenStack组件间关系
- 手动迁出ALLinOne环境中的组件，深入理解OpenStack运作机理

# 手动添加运算节点 ( Computer ) 1

- 安装运算节点nova相关软件包
  - openstack-nova-compute 、 openstack-selinux和sysfsutils
  - 依赖包自动安装
- 编辑配置文件nova.conf
  - 将AllinOne节点上的/etc/nova/nova.conf配置文件拷贝到运算节点相应目录下

# Nova.conf配置文件说明

- [DEFAULT]段
  - rabbit\_\* 配置消息队列服务所在位置及用户密码
  - vncserver\_\*和novncproxy\_\* 配置虚拟机实例连接VNC服务位置
  - 重要：需要修改vncserver\_listen和vncserver\_proxyclient\_address为本运算节点ip
  - sql\_connection 配置nova连接数据库位置及用户密码
- [glance]段
  - api\_servers 配置glance服务位置
- [keystone\_authtoken]段
  - 配置nova到keystone认证相关项目



# Nova.conf配置文件说明

- [libvirt]段
  - virt\_type配置技术节点使用的虚拟化技术方式
- [neutron]段
  - 配置技术节点连接网络节点的方式和服务位置
  - 安装Neutron和OpenVSwitch相关服务后才会生效

# 手动添加运算节点 ( Computer ) 2

- 安装运算记得neutron相关软件包
  - openstack-neutron 、 openstak-neutron-ml2和openstack-neutron-openvswitch
  - 依赖包自动安装
- 编辑配置文件neutron.conf
  - 将AllinOne节点上的/etc/neutron/neutron.conf配置文件拷贝到运算节点相应目录下

# Neutron.conf配置文件说明

- [database]段
  - 此字段内容在计算节点上是不需要的，计算节点上的neutron仅仅作agent
- [keystone\_authtoken]段
  - 配置neutron在keystone服务上认证

# 手动添加运算节点 ( Computer ) 3

- 编辑配置文件ml2\_conf.ini
  - 将AllinOne节点上的  
/etc/neutron/plugins/ml2/ml2\_conf.ini配置文件拷  
贝到运算节点相应目录下
- 创建plugin.ini链接文件
  - 创建/etc/neutron/plugin.ini链接文件指向  
/etc/neutron/plugins/ml2/ml2\_conf.ini

# ml2\_conf.ini文件内容

- [ml2]段
  - type\_drivers 虚拟网络类型
- [ml2\_type\_\*]段
  - 根据type\_drivers 制定类型配置

# 手动添加运算节点 ( Computer ) 4

- 编辑配置文件ovs\_neutron\_plugin.ini
  - 将AllinOne节点上的  
/etc/neutron/plugins/openvswitch/ovs\_neutron\_plugin.ini配置文件拷贝到运算节点相应目录下
- 需要修改ovs\_neutron\_plugin.ini文件
  - [ovs]段中local\_ip设为计算节点本地连接网络节点的ip地址

# ovs\_neutron\_plugin.ini文件内容

- [ovs]段
  - local\_ip 本地建立隧道链接的ip
- [agent]段
  - tunnel\_types 建立隧道类型
- [securitygroup]段
  - firewall\_driver OVS设置安全组和防火墙规则

# 手动添加运算节点（ Computer ） 5

- 设置并启动计算节点的nova-computer相关服务
  - #systemctl enable libvirt openstack-nova-computer openvswitch neutron-openvswitch-agent
  - #systemctl start libvirt openstack-nova-computer openvswitch neutron-openvswitch-agent



# 检查新的运算节点服务注册情况

- 在控制节点上执行命令查看服务注册情况
  - #nova service-list
- 在控制节点上执行命令查看计算节点虚拟化控制器情况
  - #nova hypervisor-list
- 注意运行环境及keystonerc\_admin的使用

# 检查新的运算节点服务网络情况

- 在控制节点上执行命令查看Neutron代理情况
  - #neutron agent-list
- 在控制节点上执行命令查看openvswitch及vxlan隧道情况
  - #ovs-vsctl show

# 配置nova迁移Key

- 使用packstack安装时会自动产生Key
  - /var/lib/nova/.ssh目录
  - id\_rsa 私钥
  - authorized\_keys 公钥
- 使用packstack安装时会自动产生known\_hosts
  - /etc/ssh/ssh\_known\_hosts

# 手动生成nova迁移配置文件

- 创建SSH密钥
  - #ssh-keygen -t rsa -b 2048 -f nova\_id\_rsa
  - 会产生两个文件nova\_id\_rsa和nova\_id\_rsa.pub
- 在每个计算节点上创建相应目录并复制相应SSH密钥
  - 具体操作见下一页

- NOVA\_SSH=/var/lib/nova/.ssh
- mkdir \${NOVA\_SSH}
- cp nova\_id\_rsa \${NOVA\_SSH}/id\_rsa
- cp nova\_id\_rsa.pub \${NOVA\_SSH}/authorized\_keys
- chmod -R 600 \${NOVA\_SSH}
- chown -R nova.nova \${NOVA\_SSH}
- ssh-keyscan -t rsa 每个计算节点主机名 >>/etc/ssh/ssh\_known\_hosts
- Chmod 644 /etc/ssh/ssh\_known\_hosts
- usermod -s /bin/bash nova
- **重要！！！保持每个计算节点的id\_rsa、authroized\_keys和ssh\_knowns\_hosts三个文件完全一致**

# 移除现有计算服务节点或模块 1

- 移除任何服务节点均需要在控制节点上完成
  - 首先配置工作环境-#source keystone\_admin
  - 列出现有计算服务及其节点-#nova service-list
  - 确定需要移除的服务及其节点，我们试验中移除AllinOne节点上的nova-compute服务

# 移除现有计算服务节点或模块 2

- 在要移除服务的节点上先禁用nova-compute服务
  - `#nova service-disable [hostname] nova-compute`
- 将需要移除的计算节点上所有实例迁移出去
  - `#nova host-servers-migrate [hostname]`
- Hostname 为需移除的节点主机名

# 移除现有计算服务节点或模块 3

- 查看实例迁移的状态
  - #nova migration-list
  - 状态为finished时迁移完毕
- 当每个实例迁移完毕后，它在nova中的状态是VERIFY\_RESIZE，需要我们手工确认其继续运行
  - #nova list
  - #nova resize-confirm [instance-name]



# 移除现有计算服务节点或模块 3

- 将移除节点上的nova-compute服务删除
  - #nova service-delete [id]
  - Id 为nova service-list列出的状态为disabled的nova-compute服务号
- 随后停止移除节点上的nova-compute运行服务
  - #systemctl disable openstack-nova-compute
  - #systemctl stop openstack-nova-compute