**OpenStack最新版本Folsom架构解析**

**摘要：**OpenStack的第6版，版本代号为Folsom的最新版于今年九月底正式发布，Folsom将支持下一代软件定义网络(SDN)作为其核心组成部分。Folsom改进了现有代码的可用性和稳定性，包括185个新功能，最主要是虚拟网络方面的功能，而且这也是新成立的OpenStack基金会推出的第一个软件版本。

两年前OpenStack基于NASA的Nova项目和Rackspace的Swift项目合并得以建立，而今OpenStack已经成为云计算领域的一颗新星，继2012年四月发布Essex版本之后，在今年九月底OpenStack第6版Folsom正式发布，本文简要分析了OpenStack Folsom的架构。

**OpenStack最新的组件**

OpenStack目前有7个核心组件：Compute(计算)， Object Storage(对象存储)，Identity(身份认证)，Dashboard(仪表盘)， Block Storage(块存储)， Network(网络) 和 Image Service(镜像服务) 。下面将依次进行解释：

**Object Storage**(代号为“Swift”) 允许进行存储或者检索文件。目前已经有几好家公司开始提供基于Swift商业存储服务，这些公司包括KT，[Rackspace](http://www.rackspace.com/" \t "_blank)公司（Swift项目的发源地）和Internap，而且很多大公司内部也使用Swift来存储数据。

**Image Service**(代号为“Glance”) 是一个虚拟机镜像的存储、查询和检索系统，它提供了一个虚拟磁盘映像的目录和存储库，这些磁盘映像常常广泛应用于OpenStack Compute之中，而且这种服务在技术上是属于可选的，任何规模的云都适用于它。

**Compute**(代号为“Nova”) 根据需求提供虚拟服务。[Rackspace](http://www.rackspace.com/" \t "_blank)公司和[HP](https://www.hpcloud.com/" \t "_blank)提供商业计算服务正是建立在Nova之上，Mercado Libre和NASA（Nova项目的起源地）内部也是使用的Nova。

**Dashboard**(代号为“Horizon”) 为所有OpenStack的服务提供了一个模块化的web-based用户界面。使用这个Web GUI，可以在云上完成大多数的操作，如启动实例，分配IP地址，设置访问控制等。

**Identity**(代号为“Keystone”) 为所有的OpenStack服务提供身份验证和授权。它还提供了一个在特定OpenStack云服务上的服务目录。

**Network**(代号为“Quantum”) 在接口设备之间提供“网络连接作为一种服务”，而这些接口设备主要靠其他的OpenStack服务进行管理(最有可能是Nova)。该服务允许用户创建自己的网络，然后连接接口。Quantum提供一个可插拔的体系架构，它能支持很多流行的网络供应商和技术，Quantum是Folsom版本中的新项目。

**Block Storage**(代号为“Cinder”) 提供稳定的数据块存储服务。这个项目的很多代码最初是来自于Nova之中（就是the nova-volume service）。但是请注意，这是块存储（或者volumes），而不是类似于NFS或者CIFS文件系统，Cinder在Folsom中也是一个全新的项目。除了这些核心项目之外，也有一些“孵化”项目，未来可能会考虑列入到OpenStack的核心项目之中。

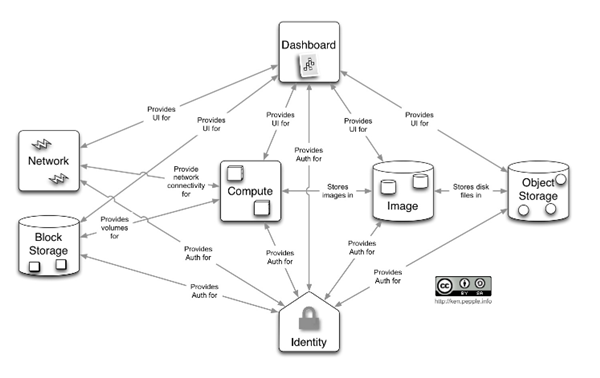
**对比AWS的服务**

虽然所有的OpenStack服务都具有自己的特色，但是很多人还是希望能看到它与AWS相似的部分，而且Amazon一直也是OpenStack的重要对手。

* Nova在概念上类似于AWS中的EC2服务，不过事实上，它拥有很多种方法可以实现对EC2 API的兼容性。
* Swift在概念上类似于S3服务，不过swift具有很强的扩展性、冗余和持久性。
* Glance提供了很多与Amazon AMI catalog相似的功能。
* Cinder提供类似于EBS块存储服务。

**概念架构**

OpenStack项目成立的目的是“提供一个大规模的可扩展的云操作系统”。要做到这一点，每一个组成服务的设计都要精心考虑，这样才能打造一个完整的IaaS平台。从概念上，我们可以描绘出各种服务之间的关系：

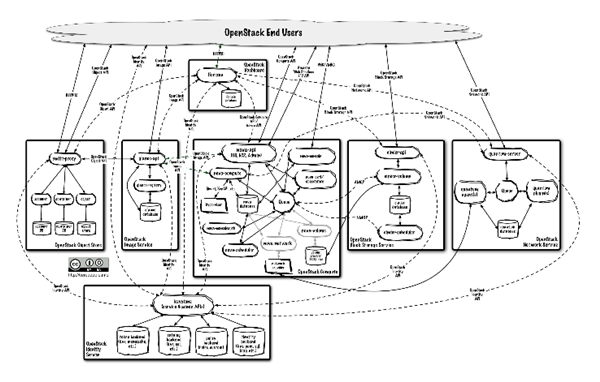


* Dashboard("Horizon") 提供了一个Web前端到OpenStack其他的服务的界面
* Compute("Nova") 存储和检索虚拟磁盘(images)和Image上相关的元数据(Glance)
* Network("Quantum") 提供虚拟网络
* Block Storage("Cinder") 提供存储。
* Image("Glance") 在对象存储(Swift)上能够完成虚拟磁盘文件的存储
* 所有的服务进行身份验证(Keystone)

这是一个程式化的简化版的体系结构视图，而且假定构建者使用所有的OpenStack服务进行最常见的配置操作，不过它也仅仅是显示“操作员”看到的云——并没有显示出云用户具体的使用过程，比如说用户如何进行直接的对象存储。

**逻辑架构**

正如你能想象到的那样，逻辑结构要比概念架构复杂得多的多(如图所示)。正如任何面向服务的架构图一样，如果想说明所有可能的服务通信组合，图就会迅速“乱成一团”。下面的图，仅仅显示了一个最常见的基于OpenStack的云架构。当然，随着OpenStack支持技术种类的多样化，它并不能代表唯一的架构图。



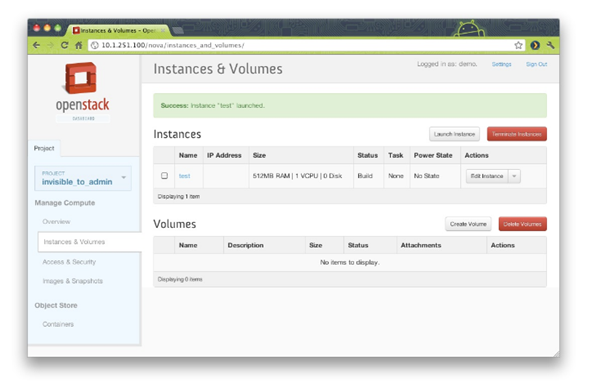
该图与上述的概念架构图是一致的：

* 最终用户可以通过一个公共的Web界面(Horizon)进行交互或者通过其API直接访问每一项服务
* 所有的服务进行身份验证都是通过一个共同的来源(通过Keystone)
* 个人服务通过他们公共的API进行交互(除了那些拥有特别权限的地方才需要管理员的命令)

下面的章节中，将会深入到每个服务的架构之中进行说明。

**Dashboard**

Horizon是一个模块化的[Django Web](https://www.djangoproject.com" \t "_blank)应用程序，它为终端用户和系统管理员提供界面来管理OpenStack服务。



和大多数Web应用程序一样，该体系架构是也是非常简单：

* Horizon通常使用Apache上的mod\_wsgi进行部署。代码本身被分离成可复用的python模块，通过逻辑（使用不同的OpenStack API进行交互）和presentation（对不同的站点很容易实现定制）实现。
* 一个数据库，不过因为它主要依赖于其他的数据服务，所以本身存储的数据非常少。

从网络架构的角度来看，这项服务需要客户的访问而且要能够跟每项服务公共的API进行交互。如果您希望使用的管理员功能（即其他的服务），也需要连接到他们的Admin API端点（这不是客户能随意访问的）。

**Compute**

Nova是OpenStack中最复杂的分布式组件，它通过大量的进程合作，将最终用户的API请求发送到正在运行的虚拟机之上。以下是这些进程的列表及其功能的描述：

* nova-api：接受和响应最终用户Compute API的请求。它支持OpenStack Compute API，Amazon EC2 API和一个特殊的Admin API。它还引发多数业务流程的活动(如运行一个实例)，并实施一些政策(主要是配额检查)。
* nova-compute：主要是一个人工守护进程，它可以通过虚拟机管理程序的API（XenAPI for XenServer/XCP, libvirt for KVM or QEMU, VMwareAPI for VMware等）来创建和终止虚拟机实例。虽然通过该进程做的事情是相当的复杂，但是它的基础原理却是非常的简单：接收队列中的动作，然后执行一系列的系统命令(如启动KVM实例)，同时更新数据库中的状态。
* nova-volume：给虚拟机分配额外持久化的存储，管理持久卷到计算实例的创建，连接和分离。一个新的OpenStack项目，Cinder，将最终替代nova-volume功能。在发布的Folsom版本中，nova-volume和Block Storage service(块存储服务)有类似的功能。
* nova-network：该人工守护进程与nova-compute和nova-volume非常相似。它接受队列中的网络任务，然后执行任务操纵网络（如设立桥接接口或更改iptables规则）。不过该项功能被移植到Quantum之中，已经成为一个独立的OpenStack服务。
* nova-schedule：从概念上说是OpenStack Nova中最简单的一段代码：从队列上得到一个虚拟机实例请求并且决定它应该在哪里运行(特别是它应该运行在哪台计算服务器主机之上)
* queue：提供了一个守护进程之间传递消息的中央枢纽。当前由[RabbitMQ](http://www.rabbitmq.com/" \t "_blank)实现，理论上可以是Python的ampqlib支持的任何AMPQ消息队列。新的Folsom版本支持[Zero MQ](http://www.zeromq.org/" \t "_blank)。
* SQL database：存储云基础设施的编译时和运行时的状态。这包括可用的实例类型，在使用中的实例，可用的网络和项目。从理论上讲，OpenStack Nova可以支持任何SQL-Alchemy支持的数据库，但是目前被广泛使用的数据库仅仅有sqlite3（只适用于测试和开发工作），MySQL和PostgreSQL。
* Nova还提供控制台的服务，让最终用户通过代理服务器访问他们的虚拟实例的控制台。这涉及到多个守护进程(nova-console，nova-vncproxy和nova-consoleauth)。

OpenStack 2013.2（代号Havana）最终版已经发布，除了增加OpenStack Metering（Ceilometer）和OpenStack Orchestration（Heat）两个新组件外，还完成了400多个特性计划，修补了3000多补丁。

具体技术细节可以参考各组件的版本发布说明：

* OpenStack Compute: https://launchpad.net/nova/havana/2013.2
* OpenStack Object Storage: https://launchpad.net/swift/havana/1.10.0
* OpenStack Image Service: https://launchpad.net/glance/havana/2013.2
* OpenStack Networking: https://launchpad.net/neutron/havana/2013.2
* OpenStack Block Storage: https://launchpad.net/cinder/havana/2013.2
* OpenStack Identity: https://launchpad.net/keystone/havana/2013.2
* OpenStack Dashboard: https://launchpad.net/horizon/havana/2013.2
* OpenStack Metering: https://launchpad.net/ceilometer/havana/2013.2
* OpenStack Orchestration: https://launchpad.net/heat/havana/2013.2

Docker的核心开发者Jerome Petazonni（他是今年中国云计算大会的演讲者）在[Hacker News上评论](https://news.ycombinator.com/item?id=6565779)指出，Havana版已经以Nova驱动的形式内置支持Docker容器， 这意味着：

测试OpenStack工作负载时，可以启动容器而不是虚拟机，启动时间将大大缩短

高性能计算场景下，由于容器开销小得多（配置正确的话接近于0），可以在OpenStack上运行，而无需担忧性能损失 。

OpenStack现在采用6个月一个开发周期的方式，每个代号均为首字母按字典顺序排到的某个城市名，依次是Austin、Bexar、Cactus、Diablo、Essex、Folsom、Grizzly和Havana。接下来的版本，叫Icehouse。