

在学习 OpenStack 各服务之前，让我们先搭建起一个实验环境。

毋庸置疑，一个看得到摸得着而且允许我们随便折腾的 OpenStack 能够提高我们的学习效率。

因为是我们自己学习用的实验环境，CloudMan 推荐使用 DevStack<http://docs.openstack.org/developer/devstack/>DevStack 丰富的选项让我们能够灵活地选取和部署想要的 OpenStack 服务，非常适合学习和研究。

**部署拓扑**

首先我们来设计 OpenStack 的部署拓扑。

OpenStack 是一个分布式系统，由若干不同功能的节点（Node）组成：

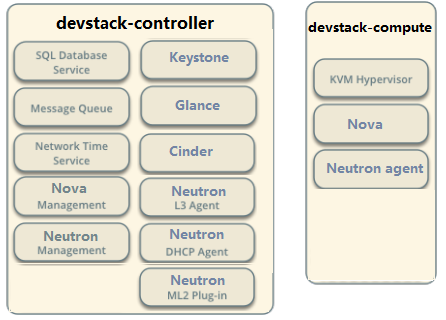
1. 控制节点（Controller Node）
2. 管理 OpenStack，其上运行的服务有 Keystone、Glance、Horizon 以及 Nova 和 Neutron 中管理相关的组件。
3. 控制节点也运行支持 OpenStack 的服务，例如 SQL 数据库（通常是 MySQL）、消息队列（通常是 RabbitMQ）和网络时间服务 NTP。
4. 网络节点（Network Node）
5. 其上运行的服务为 Neutron。
6. 为 OpenStack 提供 L2 和 L3 网络。
7. 包括虚拟机网络、DHCP、路由、NAT 等。
8. 存储节点（Storage Node）
9. 提供块存储（Cinder）或对象存储（Swift）服务。
10. 计算节点（Compute Node）
11. 其上运行 Hypervisor（默认使用 KVM）。
12. 同时运行 Neutron 服务的 agent，为虚拟机提供网络支持。

这几类节点是从功能上进行的逻辑划分，在实际部署时可以根据需求灵活配置，比如：

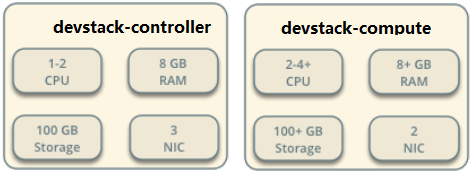
1. 在大规模OpenStack生产环境中，每类节点都分别部署在若干台物理服务器上，各司其职并互相协作。   
   这样的环境具备很好的性能、伸缩性和高可用性。
2. 在最小的实验环境中，可以将 4 类节点部署到一个物理的甚至是虚拟服务器上。   
   麻雀虽小五脏俱全，通常也称为 All-in-One 部署。

在我们的实验环境中，为了使得拓扑简洁同时功能完备，我们用两个虚拟机：

1. devstack-controller：控制节点 + 网络节点 + 块存储节点 + 计算节点
2. devstack-compute：计算节点



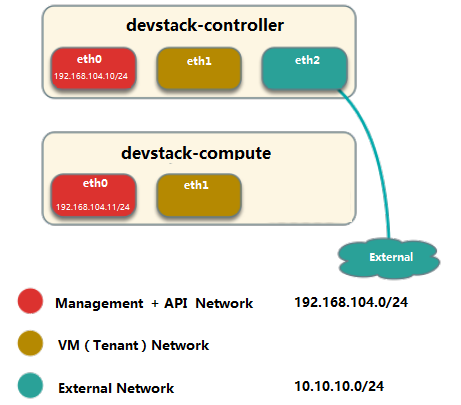
**物理资源需求**



CPU 和内存供参考。

如果是在自己的 PC 机上创建虚机部署，资源可能达不到，可以适当调整。

**网络规划**



网络上规划了三个网络：

1. Management Network：用于 OpenStack 内部管理用，比如各服务之间通信。
2. 这里使用 eth0
3. VM（Tenant）Network：OpenStack 部署的虚拟机所使用的网络。
4. OpenStack 支持多租户（Tenant），虚机是放在 Tenant 下的，所以叫 Tenant Network。
5. 这里使用 eth1
6. External Network：一般来说，Tenant Network 是内部私有网络，只用于 VM 之间通信，与其他非 VM 网络是隔离的。
7. 这里我们规划了一个外部网络（External Network），通过 devstak-controller 的 eth2 连接。
8. Neutron 通过 L3 服务让 VM 能够访问到 External Network。
9. 对于公有云，External Network 一般指的是 Internet。
10. 对于企业私有云，External Network 则可以是 Intranet 中的某个网络。

下一节开始部署控制节点和计算节点。