

我们在 Neutron Server 小节学习到 Core Plugin，其功能是维护数据库中 network, subnet 和 port 的状态，并负责调用相应的 agent 在 network provider 上执行相关操作，比如创建 network。  
上一节也介绍了两个 Core Plugin：linux bridge plugin 和 open vswitch plugin。  
本节将详细讨论更重要的 ML2 Core Plugin。  
  
Moduler Layer 2（ML2）是 Neutron 在 Havana 版本实现的一个新的 core plugin，用于替代原有的 linux bridge plugin 和 open vswitch plugin。

**传统 core plugin 的问题**

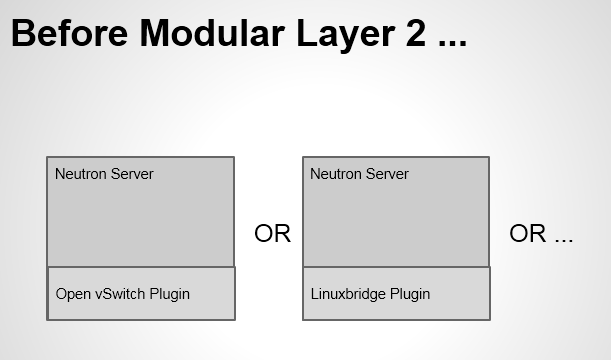
之所以要开发 ML2，主要是因为传统 core plugin 存在两个突出的问题。

**问题1：无法同时使用多种 network provider**

Core plugin 负责管理和维护 Neutron 的 network, subnet 和 port 的状态信息，这些信息是全局的，只需要也只能由一个 core plugin 管理。

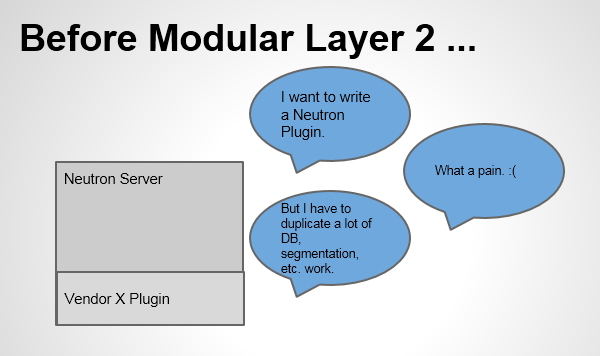
只使用一个 core plugin 本身没有问题。但问题在于传统的 core plugin 与 core plugin agent 是一一对应的。也就是说，如果选择了 linux bridge plugin，那么 linux bridge agent 将是唯一选择，就必须在 OpenStack 的所有节点上使用 linux bridge 作为虚拟交换机（即 network provider）。

同样的，如果选择 open vswitch plugin， 所有节点上只能使用 open vswitch，而不能使用其他的 network provider。



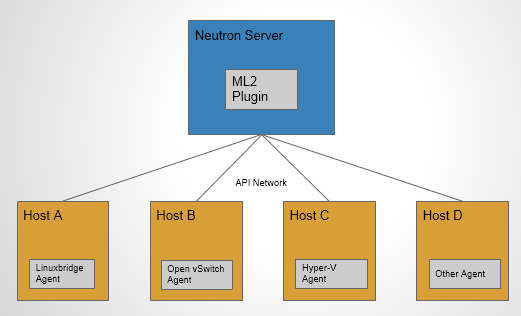
**问题2：开发新的 core plugin 工作量大**

所有传统的 core plugin 都需要编写大量重复和类似的数据库访问的代码，大大增加了 plugin 开发和维护的工作量。



**ML2 能解决传统 core plugin 的问题**

ML2 作为新一代的 core plugin，提供了一个框架，允许在 OpenStack 网络中同时使用多种 Layer 2 网络技术，不同的节点可以使用不同的网络实现机制。



如上图所示，采用 ML2 plugin 后，可以在不同节点上分别部署 linux bridge agent, open vswitch agent, hyper-v agent 以及其他第三方 agent。

ML2 不但支持异构部署方案，同时能够与现有的 agent 无缝集成：以前用的 agent 不需要变，只需要将 Neutron server 上的传统 core plugin 替换为 ML2。

有了 ML2，要支持新的 network provider 就变得简单多了：无需从头开发 core plugin，只需要开发相应的 mechanism driver，大大减少了要编写和维护的代码。

现在我们已经了解了 ML2 的必要性，下一节将学习 ML2 的架构。