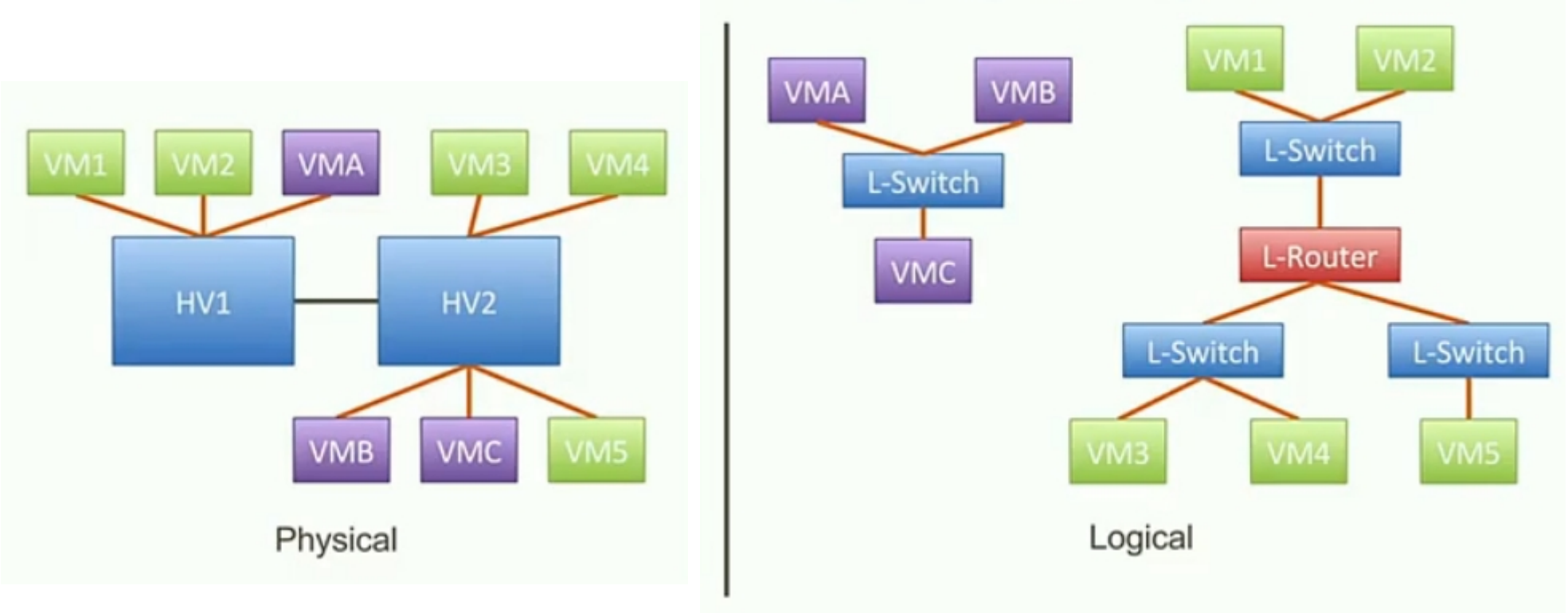
Understanding Virtual Networking

Virtual Networking Overview

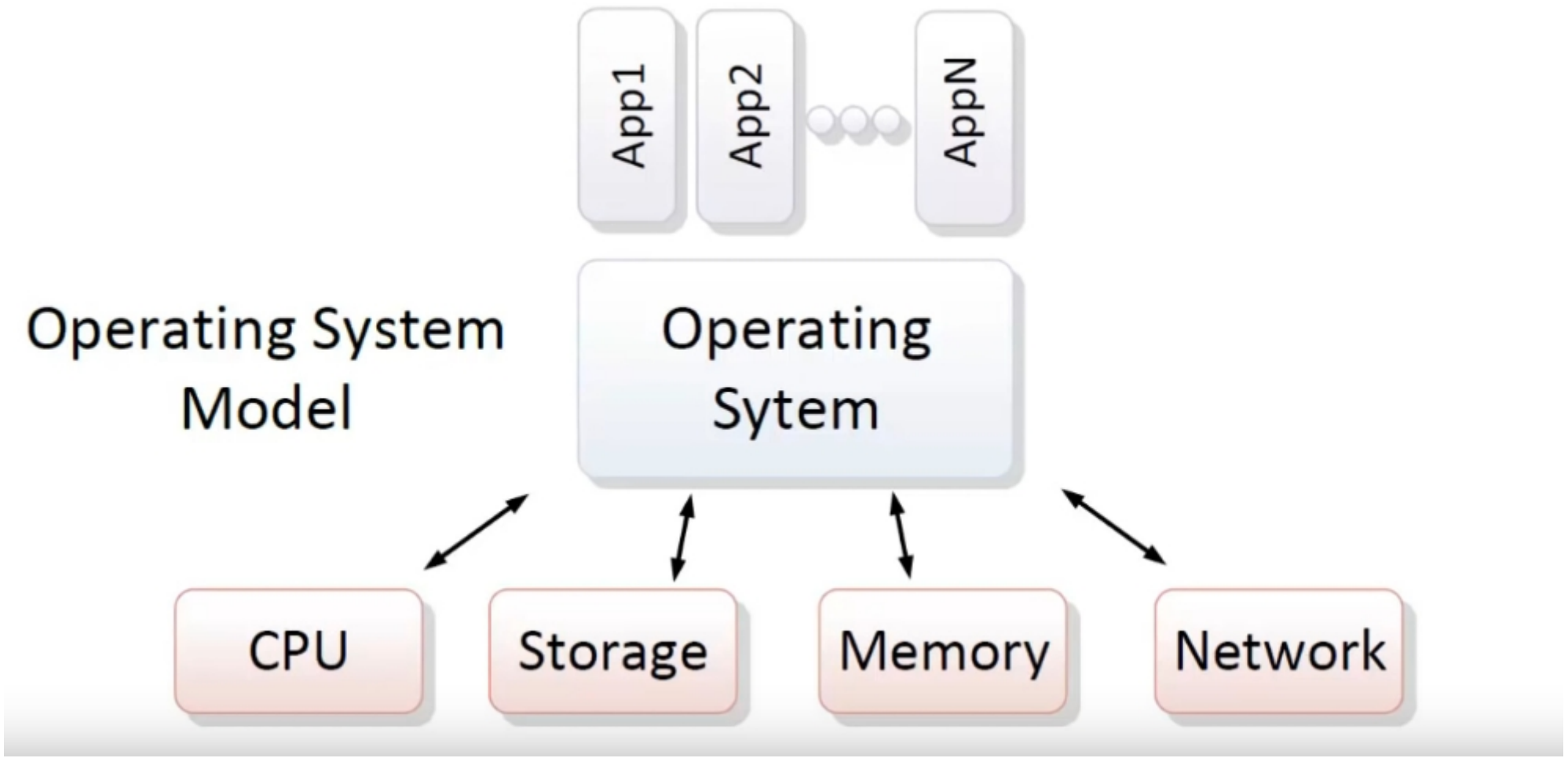
Provides a logical network abstraction on top of a physical network



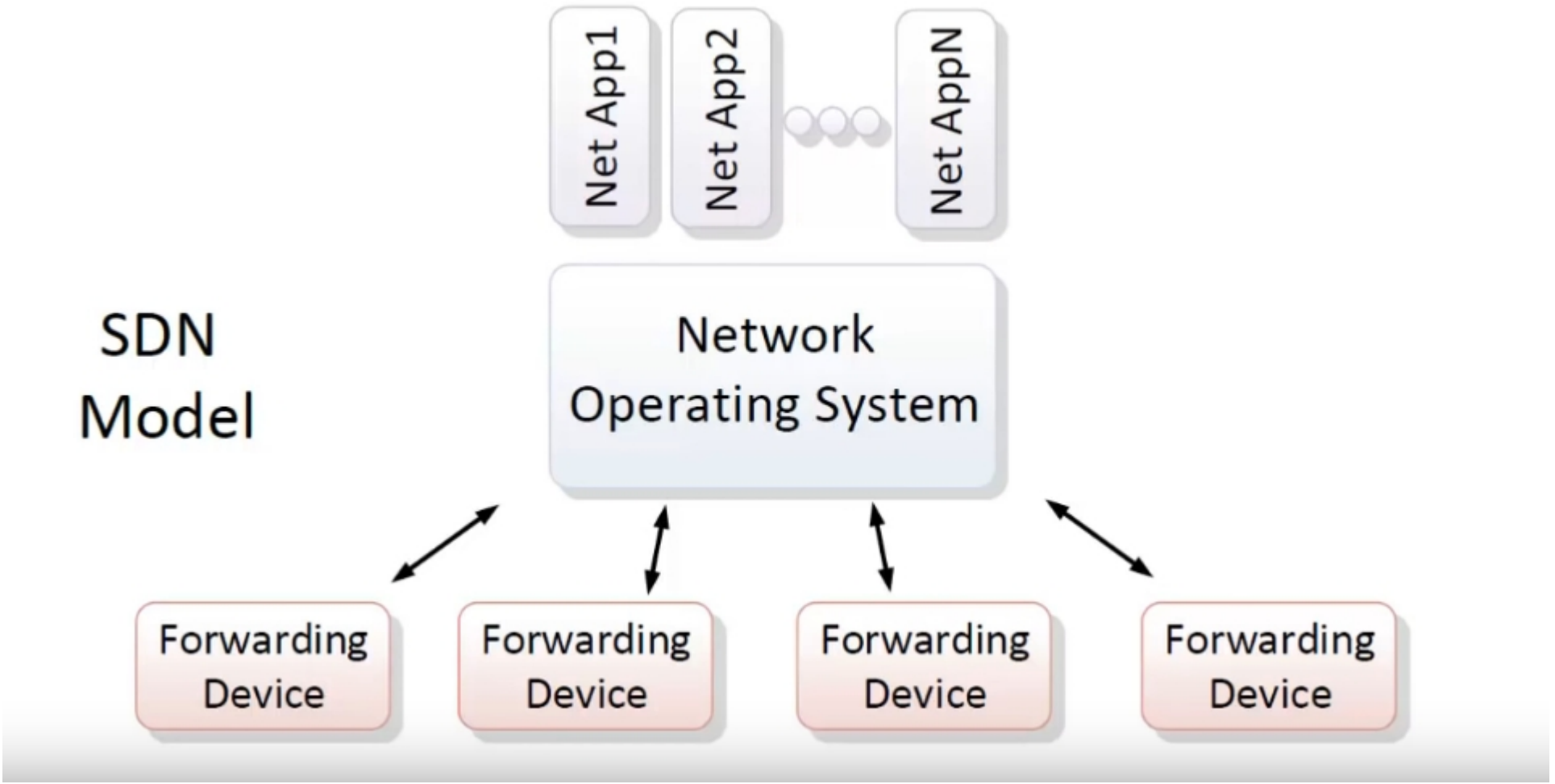
Understanding SDN

- **Decoupling Control Plane and Data Plane.**
- **Main goal is to have networking open and programmable.**

Operating Sytem Model



SDN Model


Net App1
Net App2
Net AppN
SDN
Model
Network
Operating System
Forwarding
Device
Forwarding
Device
Forwarding
Device
Forwarding
Device

Packet Flow

- > Packet header
- > Operation Involved
- > Understanding the Flow.

Forwarding Device

> Can be hardware switches which support a programmable interface like Openflow.

> Can be Software switch like openvswitch.

SouthBound Interface

> Also knows are Control to Data Plane Interface (CDPI).

> Interface between SDN controller and Forwarding device.

SDN Controller

Logically centralized entity in charge of translating the requirements from the SDN application layer down to the SDN data paths.

NBI and Application.

NBIs are interfaces between SDN applications and SDN controllers. They typically provide network views and enable expression of network behavior and requirements.

SDN applications are programs that communicate their network requirements and desired network behavior to the SDN controller over a northbound interface (NBI).

OVS ??

Open Vswitch is an opensource openflow capable virtual switch.

Allow us to connect virtual machines within a host or different hosts across networks.

> Support vlan tagging

> STP, LACP, Port mirroring

> Tunneling like GRE, VXLAN, IPSEC

> QOS support

What is OVN

> Virtual Network for OpenVswitch

> Developed with OVS Project

> License under Apache License

Features of OVN

> Manage network services like the typical physical network.

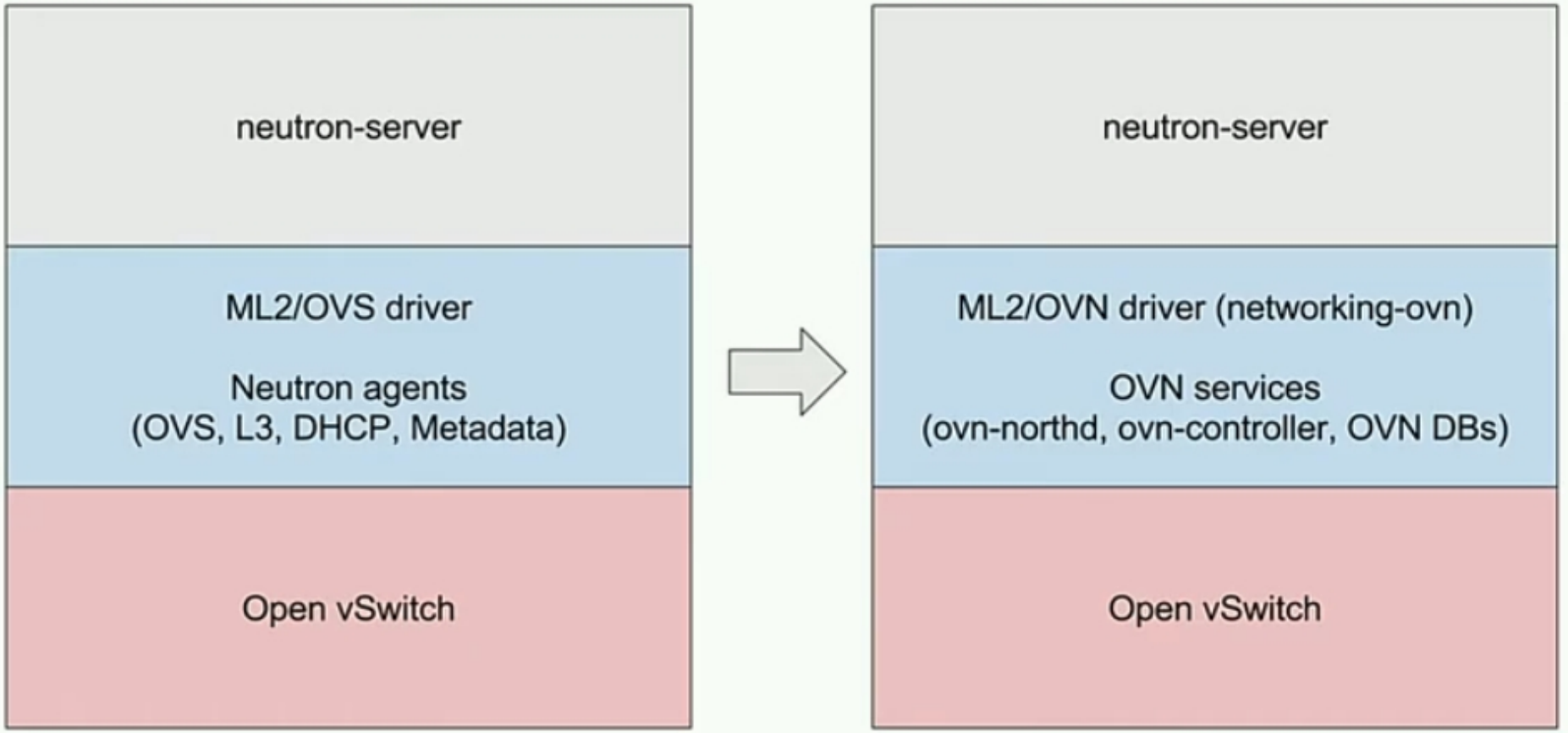
> Nat , Load Balancing, DHCP , Ipv4, Ipv6
L2, L3, ACL etc.

> Works with Linux, Hyper-V etc

> Designed to be integrated into other clouds as well .. Docker, neutron, Kubernetes etc.

OVN in openstack

How OVN fits into Neutron



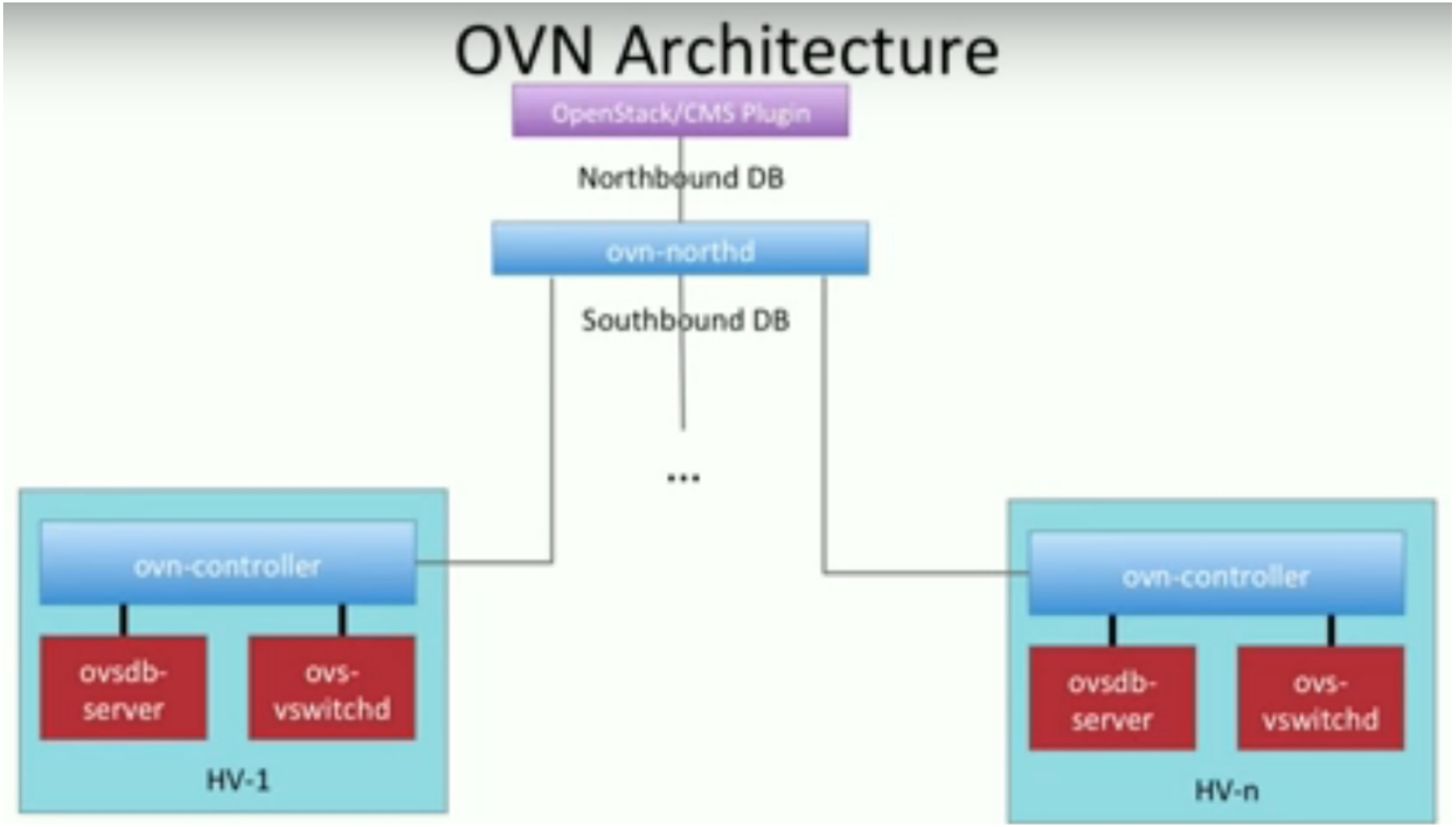
Why should openstack care ?

> Neutron is responsible for cloud networking.

> OVN is developed to implement many thing that neutron needed.

> It reduces developement burden on Neutron for OVS integration significantly leading to performance and scale improvements.

OVN Architecture



Architecture

> Configuration, coordination through Databases.

> Logical Flow, don’t worry abt physical topology

> Local controller on each hypervisor convert logical flow into physical flow state.

Differences ?

COMPONENT	ML2 WITH OVS	ML2 WITH OVN
agent/server communication	Uses the RabbitMQ messaging backend.	The ovsdb protocol is used.
L3HA dataplane	Implemented by creating a qrouter namespace.	The ovn-controller configures OpenFlow rules.
DVR API	The "distributed" flag is modifiable by an admin.	All traffic is distributed.
DVR data plane	Composed of namespaces, veth pairs, and iptables rules.	Composed of OpenFlow rules on the compute nodes.
East/West traffic	Traffic is routed through network nodes when DVR is switched off.	Traffic is distributed in all cases.
metadata service	Supported by DHCP namespaces on the controller nodes.	Runs on all compute nodes within the ovnmeta-xxx namespace.
DHCP Service	Provided by dhcp-xxx namespaces, with a dnsmasq process running inside each namespace.	Implemented using OpenFlow rules interpreted by the ovn-controller, and distributed across all compute nodes.

L3 in OVS (Existing Way)

> Agent Based.

> Overlapping IP address support using namespaces.

> Used the Linux IP stack and iptables

- forwarding

- NAT

OVN L3 Design

> Native support for Ipv4 and Ipv6

> Distributed

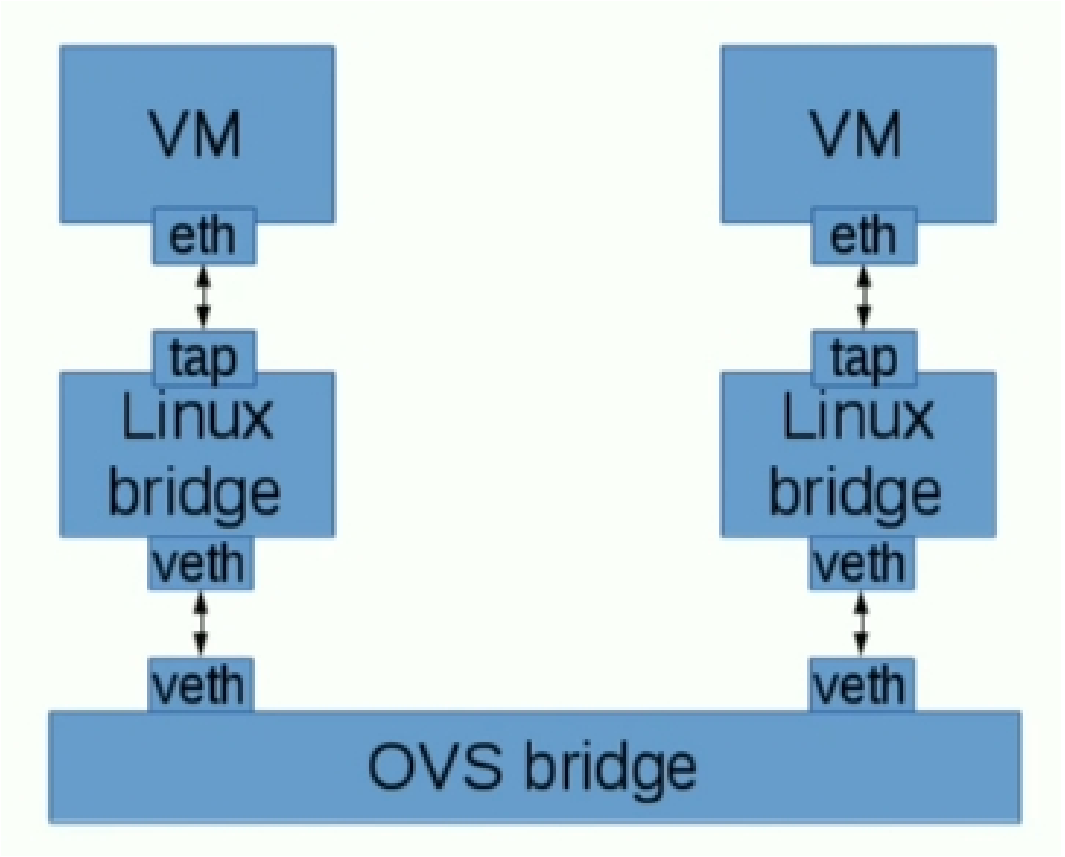
> No use of Neutron L3 agent

> Flow caching improves performance.

Security Group

> In Ovs – requires extra linux bridges and veth pair.

> Iptables are used.



Security Group in OVN

> Using Kernal Conntrack module directly from OVS.