From Nova Network to Neutron and Beyond: A Look at OpenStack Networking

OpenStack Day Korea

February 5th, 2015

Agenda

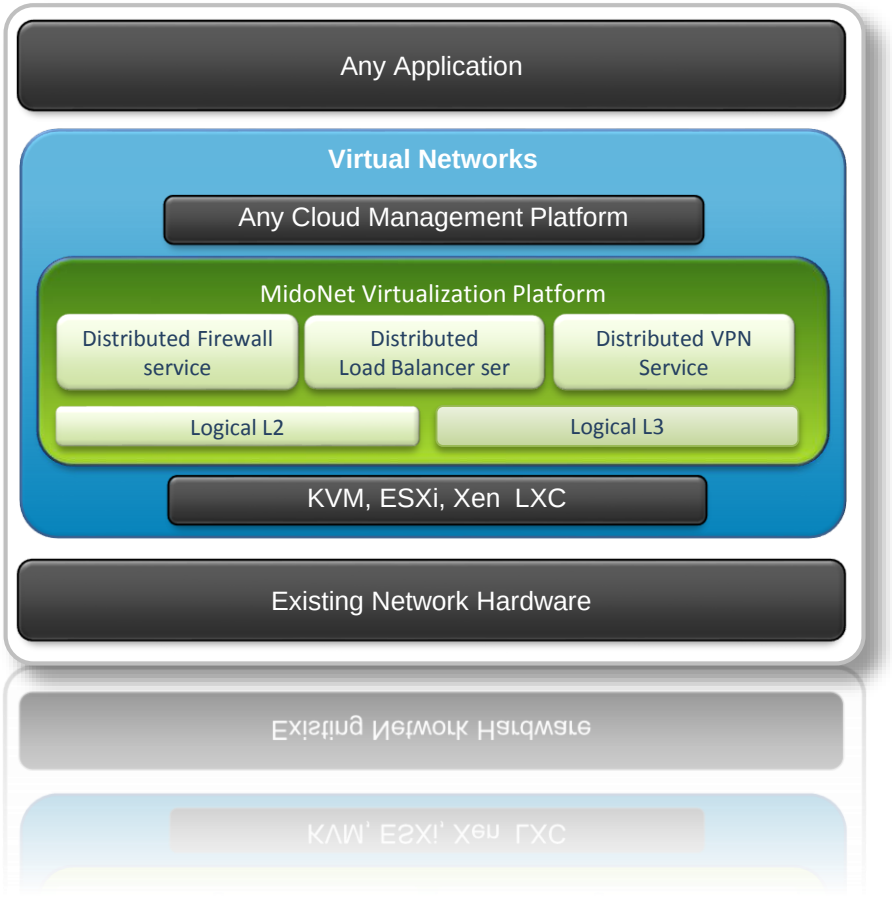
- Network Virtualization Requirements
- OpenStack and the Evolution of Neutron Networking
- Midokura Use Cases and Futures for NV

Network Virtualization Requirements

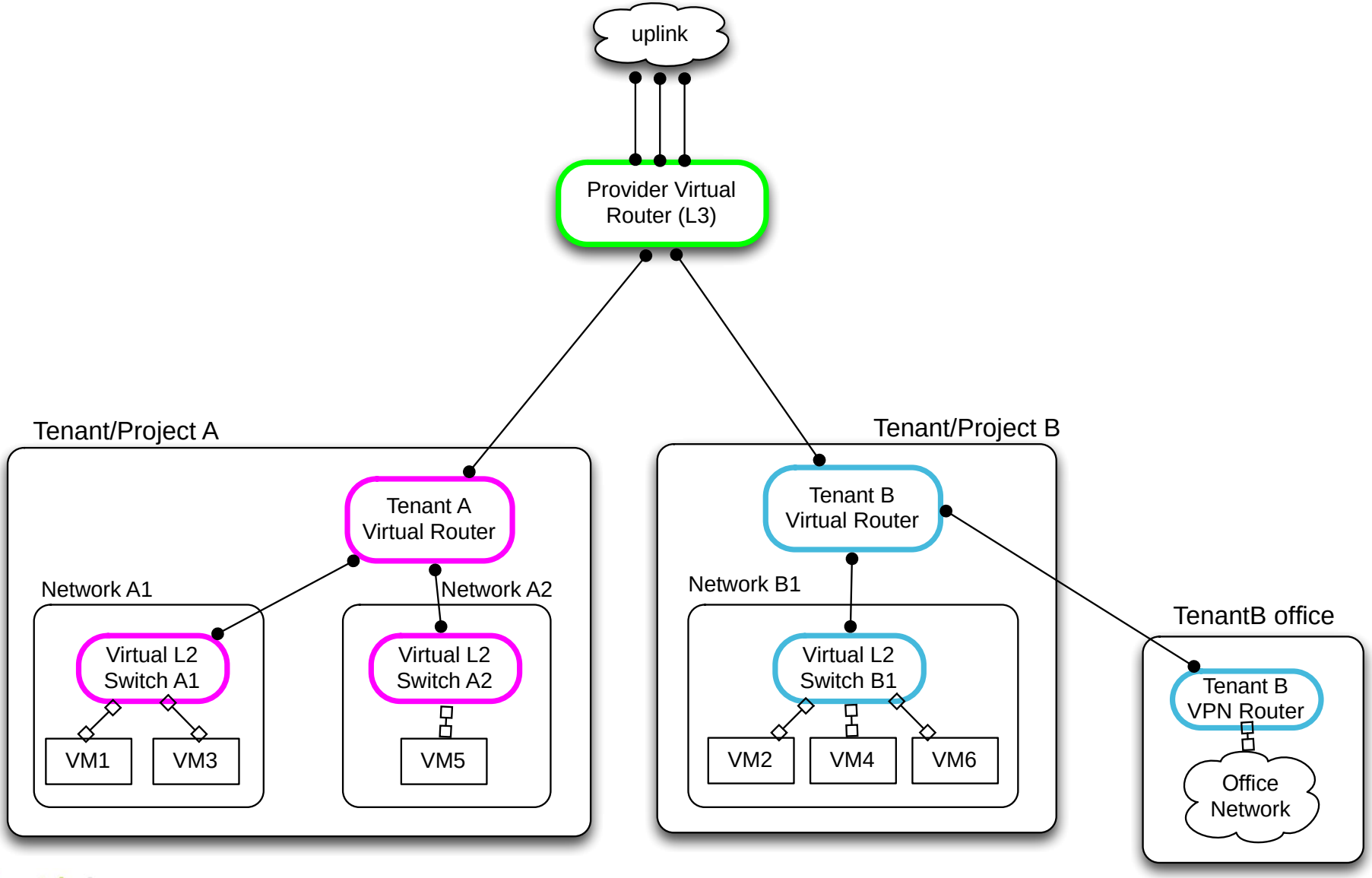
What is Network Virtualization (NV)?

Taking logical (virtual) networks and services, and decoupling them from the underlying network hardware.

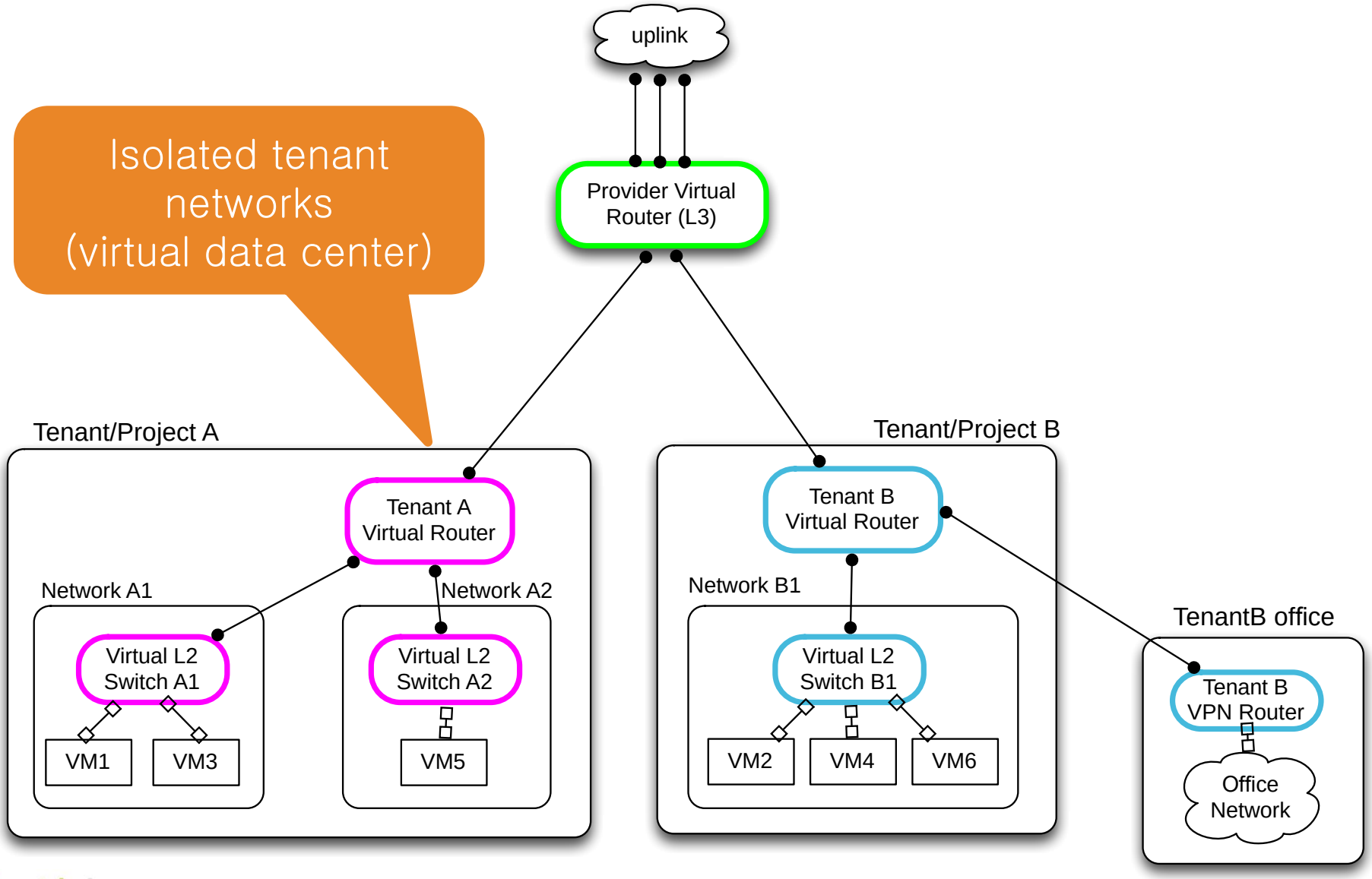
Well suited for highly virtualized environments.



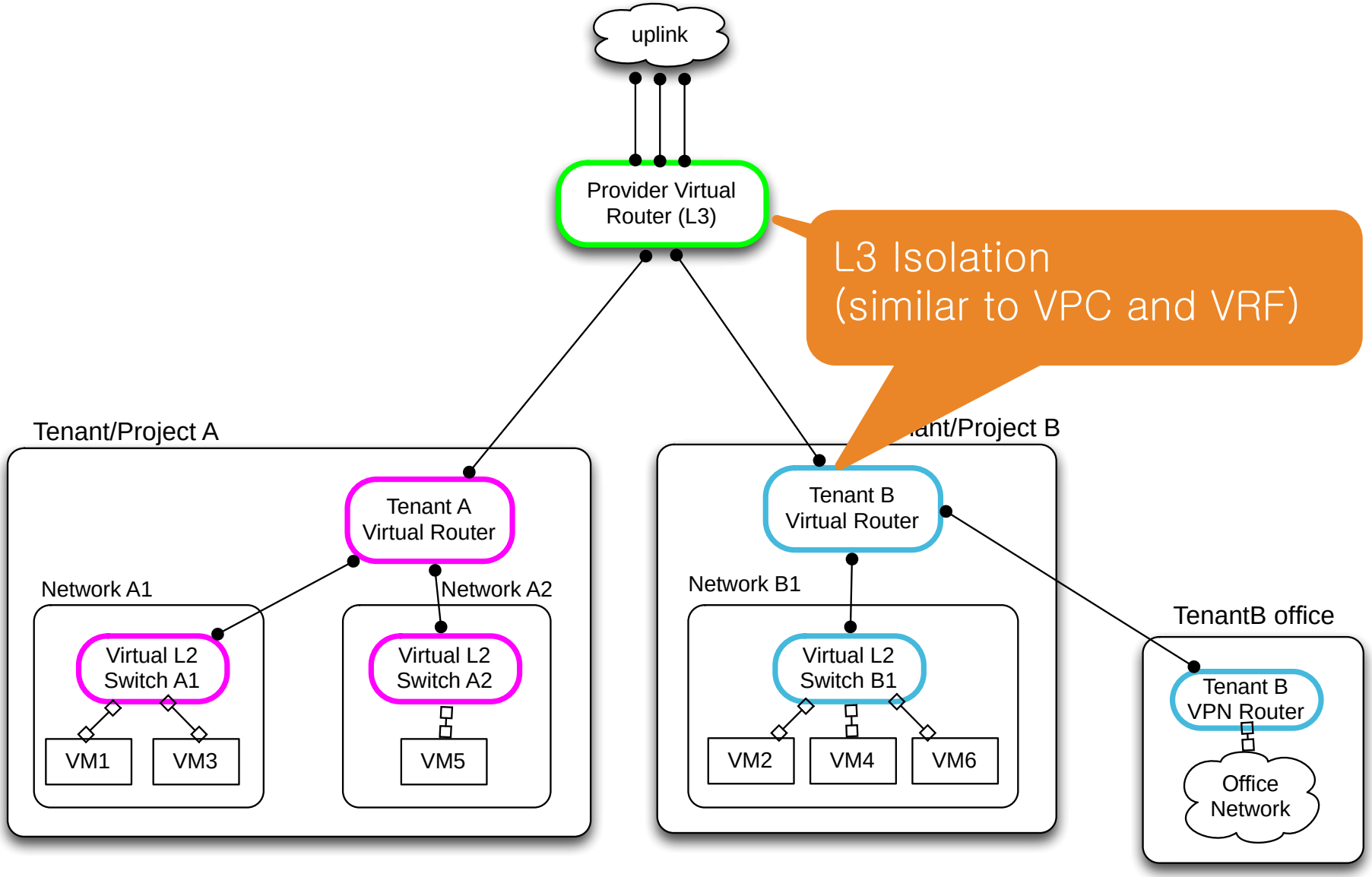
Requirements for NV



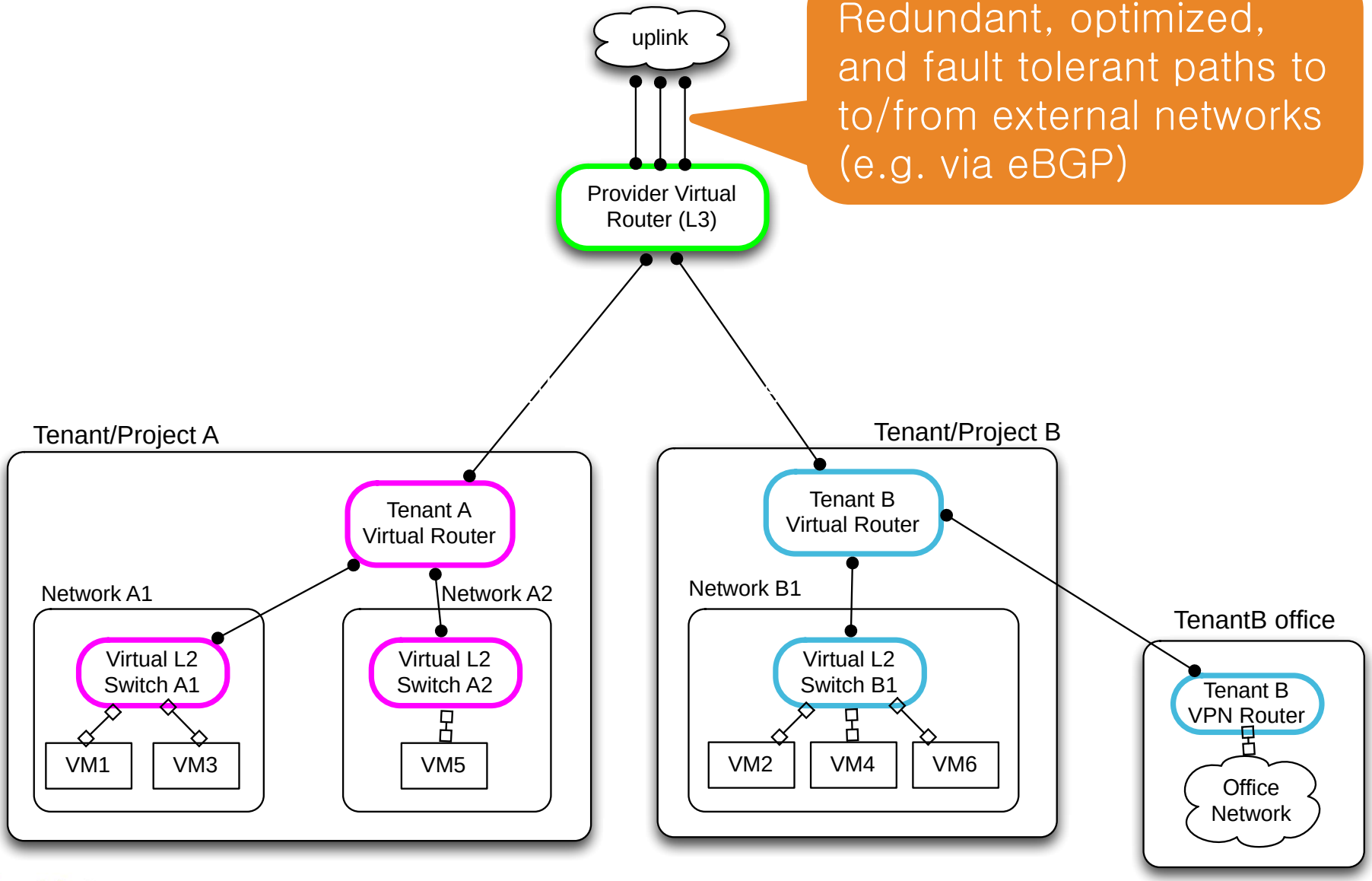
Requirements for NV



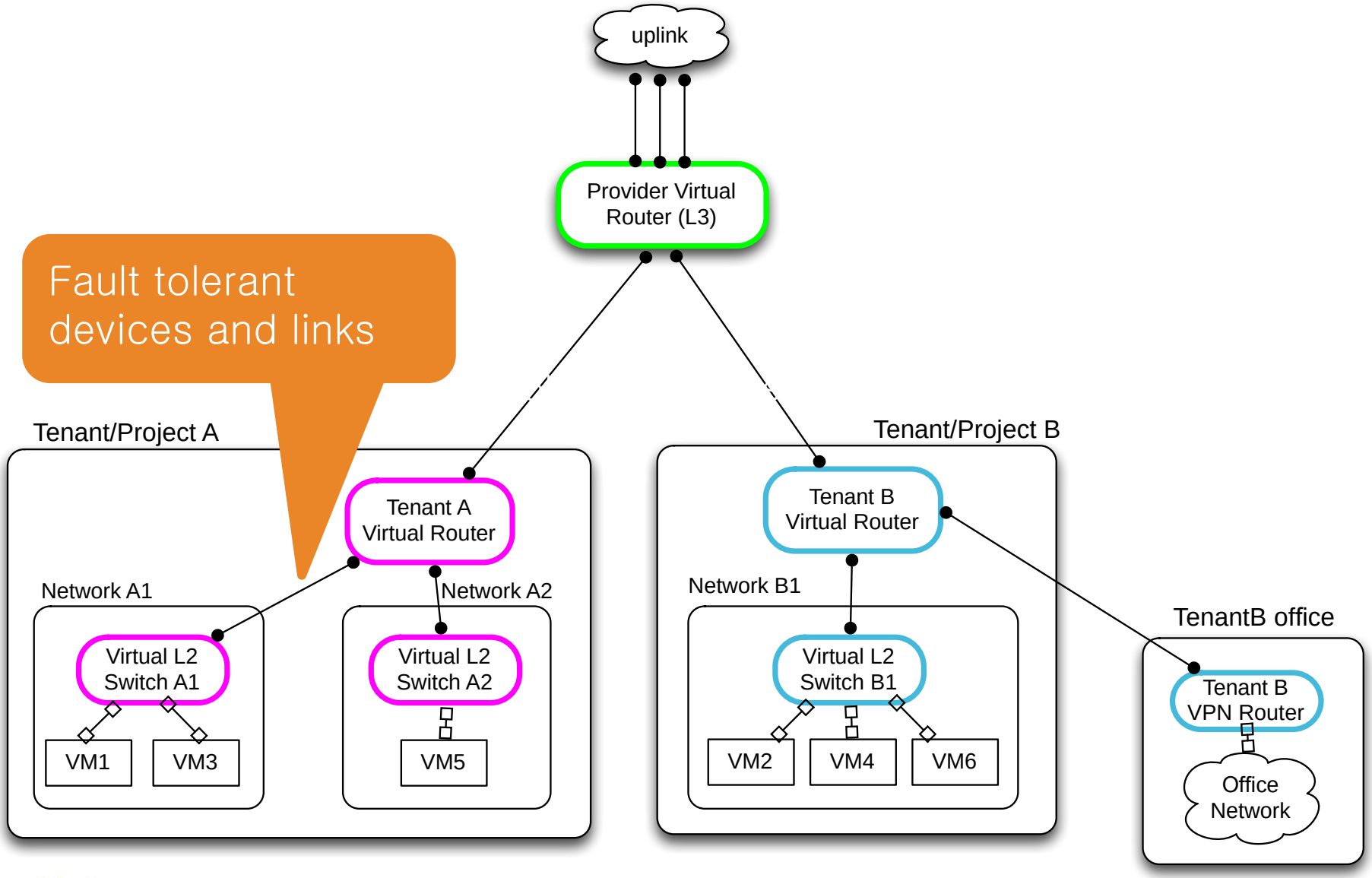
Requirements for NV



Requirements for NV



Requirements for NV



Requirements for NV

Device-agnostic networking services:

- Load Balancing
- Firewalls
- Stateful NAT
- VPN

Networks and services must be **fault tolerant** and **scalable**

Requirements for NV

Single pane of glass to manage it all.

Bonus Requirements for NV

Integration with cloud or virtualization management systems.

Optimize network by exploiting management configuration.

Single virtual hop for networking services

Fully distributed control plane (ARP, DHCP, ICMP)

Checklist for Network Virtualization

- Multi-tenancy
- Scalable, fault-tolerant devices (or device-agnostic network services).
- L2 isolation
- L3 routing isolation
  - VPC
  - Like VRF (virtual routing and fwd-ing)
- Scalable gateways
- Scalable control plane
  - ARP, DHCP, ICMP
- Floating/Elastic Ips
- Stateful NAT
  - Port masquerading
  - DNAT
- ACLs
- Stateful (L4) Firewalls
  - Security Groups
- Load Balancing with health checks
- Single Pane of Glass (API, CLI, GUI)
- Integration with management platforms
  - OpenStack, CloudStack
  - vSphere, RHEV, System Center
- Decoupled from Physical Network

Evolution of Network Virtualization

INNOVATION IN NETWORKING AGILITY

VLAN APPROACH

Manual End-to-End

VLAN configured on physical switches

- **Static**
- **Manual**
- **Complex**
- **Tenant state maintained in physical network**

Using VLANs for NV

- ❑ Multi-tenancy
- ❑ Scalable, fault-tolerant devices (or device-agnostic network services).
- ✓ L2 isolation
- ❑ L3 routing isolation
  - VPC
  - Like VRF (virtual routing and fwd-ing)
- ❑ Scalable gateways
- ❑ Scalable control plane
  - ARP, DHCP, ICMP
- ❑ Floating/Elastic IPs
- ❑ Stateful NAT
  - Port masquerading
  - DNAT
- ❑ ACLs
- ❑ Stateful (L4) Firewalls
  - Security Groups
- ❑ Load Balancing with health checks
- ❑ Single Pane of Glass (API, CLI, GUI)
- ❑ Integration with management platforms
  - OpenStack, CloudStack
  - vSphere, RHEV, System Center
- ❑ Decoupled from Physical Network

Evolution of Network Virtualization

INNOVATION IN NETWORKING AGILITY

VLAN APPROACH

OpenFlow

OPENFLOW REACTIVE APPOACH

Manual End-to-End

Reactive End-to-End

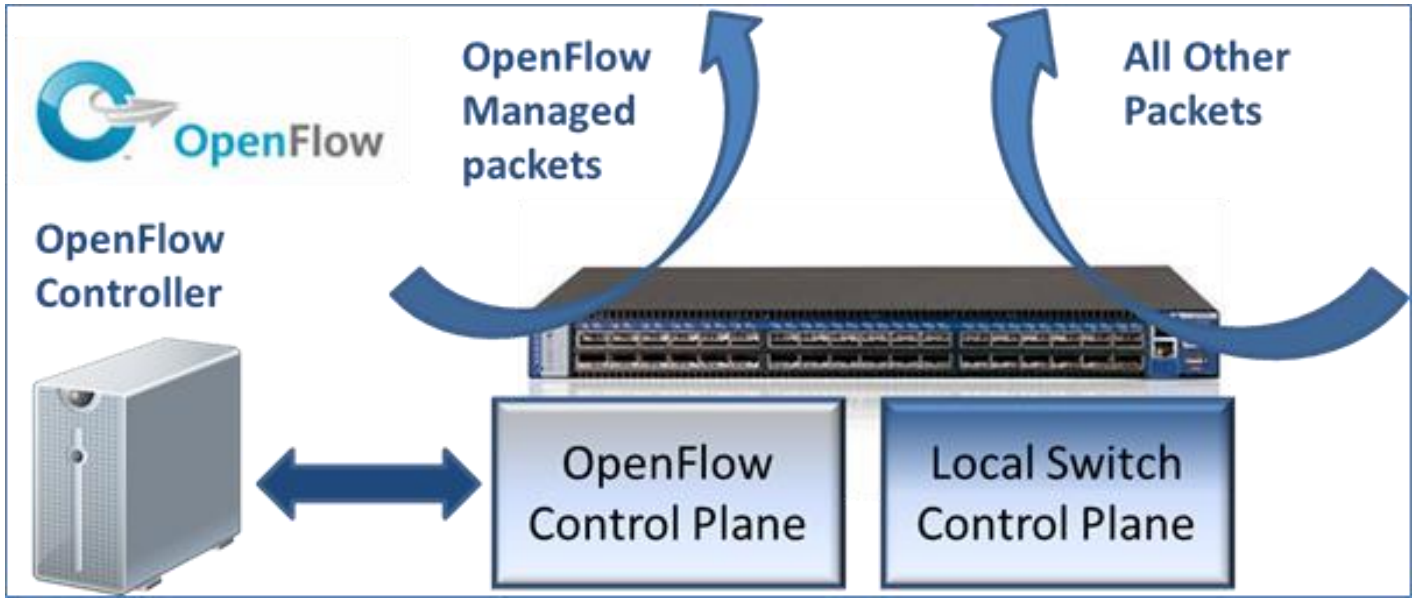
VLAN configured on physical switches

- **Static**
- **Manual**
- **Complex**
- **Tenant state maintained in physical network**

Requires programming of flows

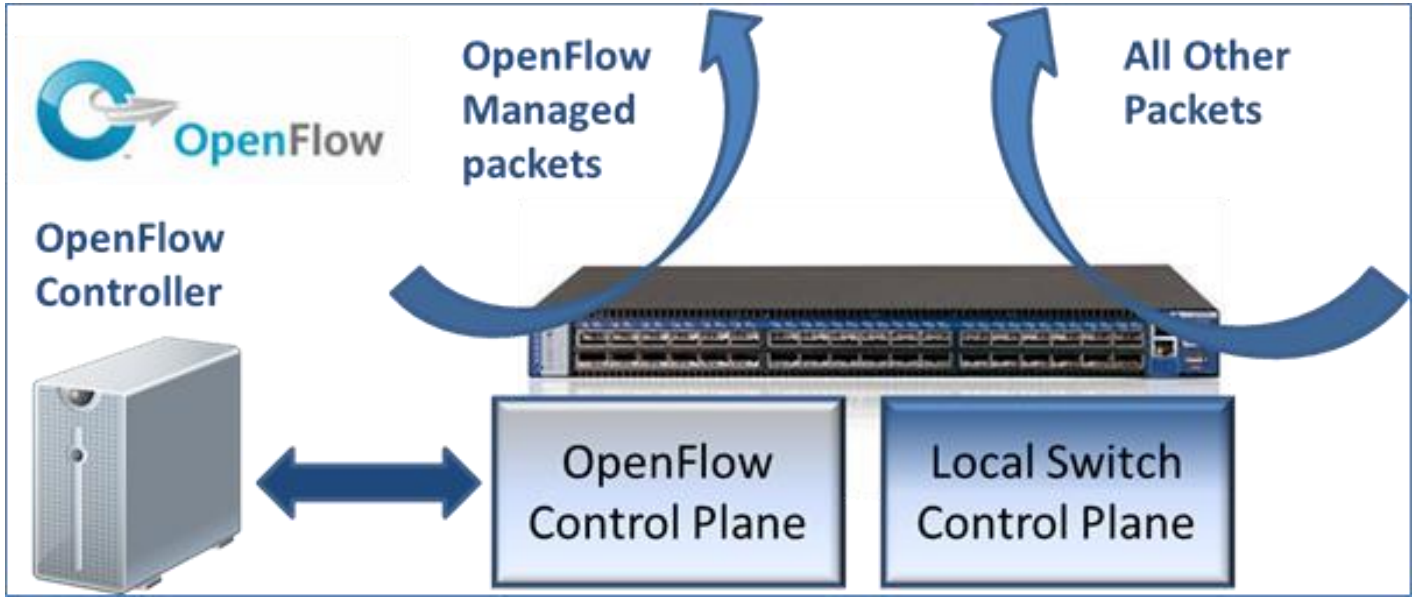
- **Limited scalability**
- **Hard to manage**
- **Impact to performance**
- **Still requires tenant state in physical network**

What is OpenFlow?



A communication protocol that gives access to the forwarding plane of a network switch over the network.

What is OpenFlow?



A centralized remote controller decides the path of packets through the switches

OPEN DAYLIGHT

big switch networks

NEC

Using OpenFlow for NV

- ✓ Multi-tenancy
- ❑ Scalable, fault-tolerant devices (or device-agnostic network services).
- ✓ L2 isolation
- △ L3 routing isolation
  - VPC
  - Like VRF (virtual routing and fwd-ing)
- ❑ Scalable gateways
- ❑ Scalable control plane
  - ARP, DHCP, ICMP
- ❑ Floating/Elastic IPs
- ❑ Stateful NAT
  - Port masquerading
  - DNAT
- ❑ ACLs
- ❑ Stateful (L4) Firewalls
  - Security Groups
- ❑ Load Balancing with health checks
- △ Single Pane of Glass (API, CLI, GUI)
- △ Integration with management platforms
  - OpenStack, CloudStack
  - vSphere, RHEV, System Center
- ❑ Decoupled from Physical Network

Evolution of Network Virtualization

INNOVATION IN NETWORKING AGILITY

VLAN APPROACH

OpenFlow

OPENFLOW REACTIVE APPOACH

PROACTIVE SOFTWARE OVERLAY

Manual End-to-End

Reactive End-to-End

Virtual Network Overlays

VLAN configured on physical switches

- **Static**
- **Manual**
- **Complex**
- **Tenant state maintained in physical network**

Requires programming of flows

- **Limited scalability**
- **Hard to manage**
- **Impact to performance**
- **Still requires tenant state in physical network**

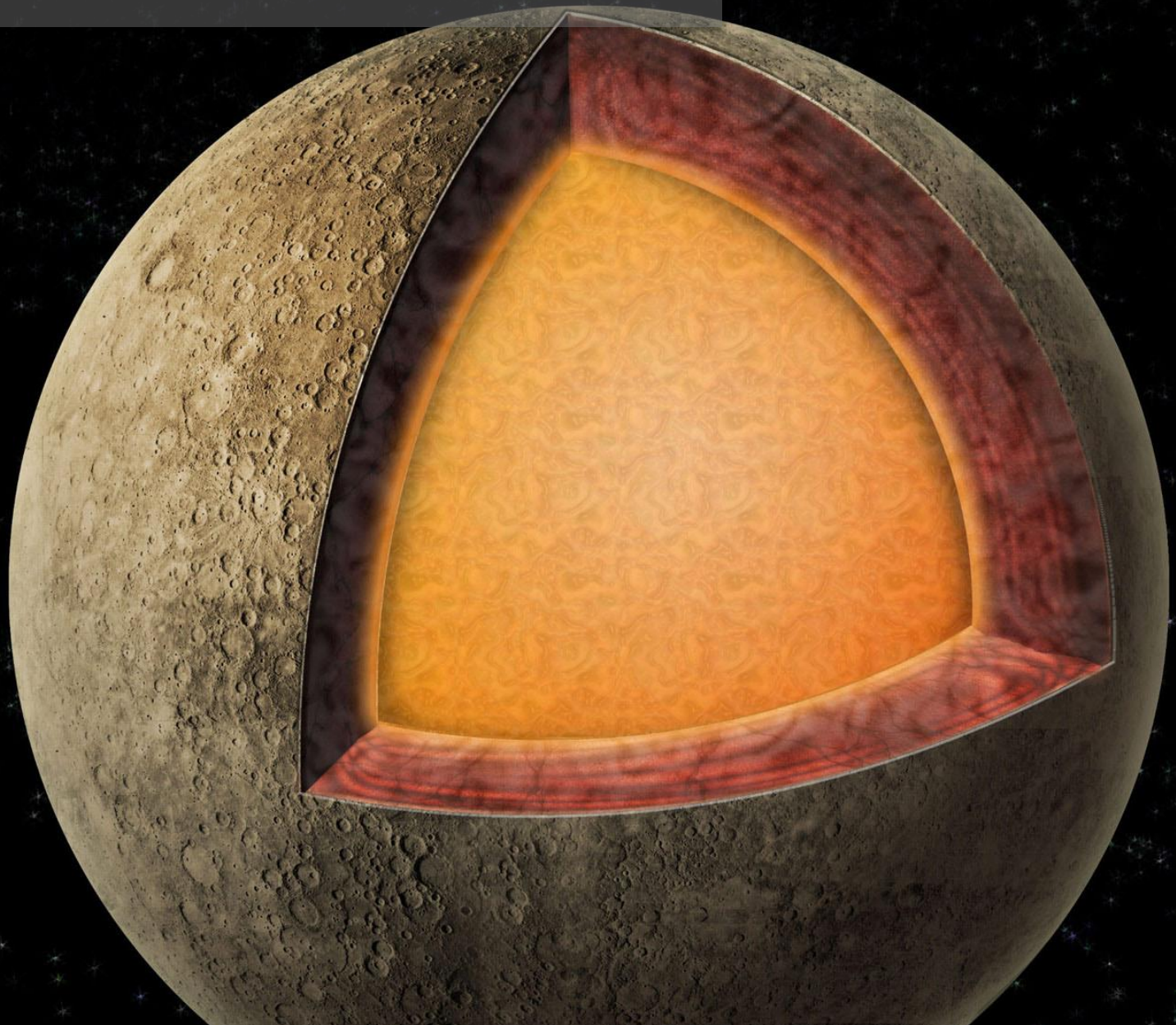
Decoupling hardware and software

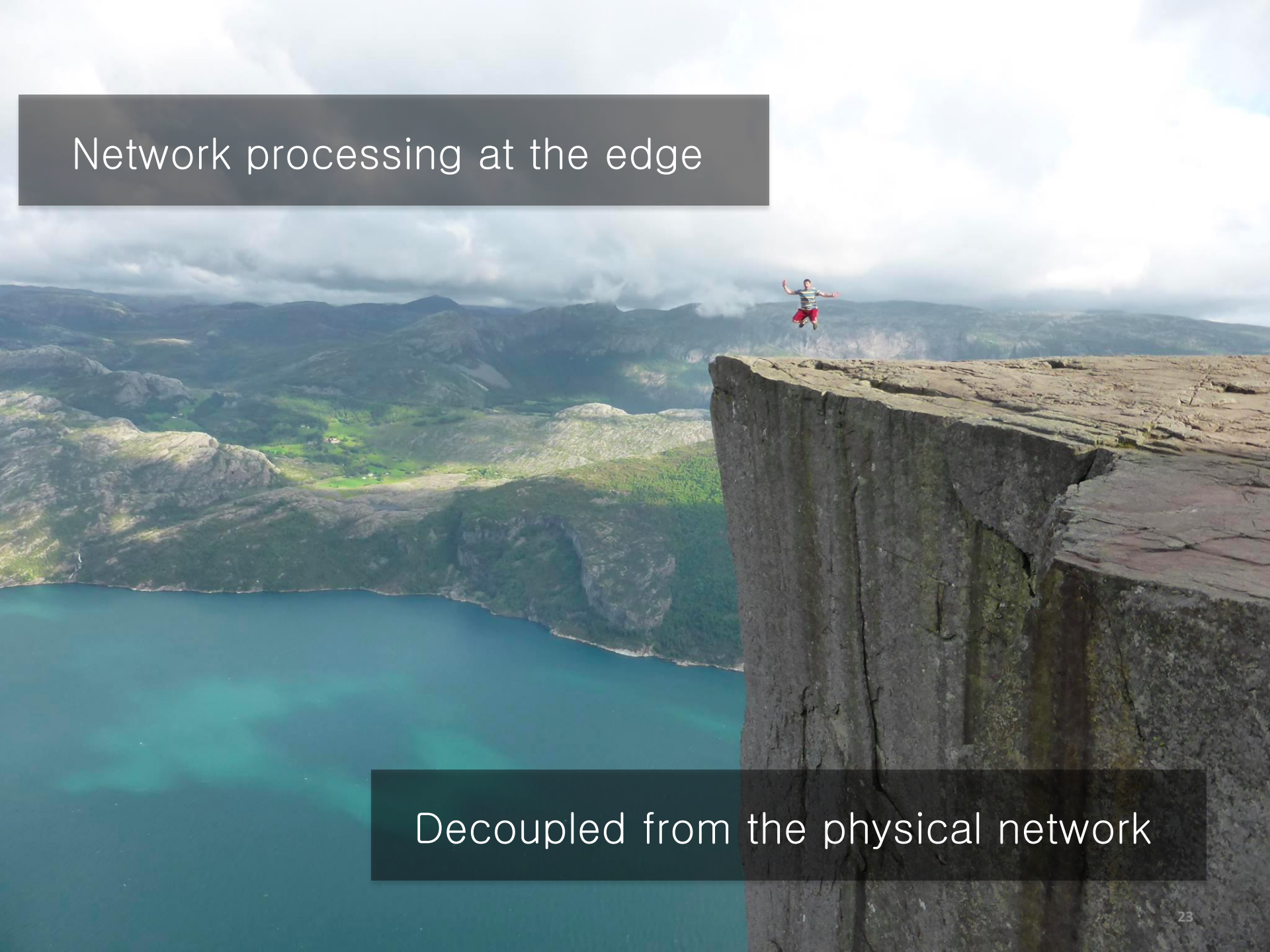
- Cloud-ready agility
- Unlimited scalability
- Open, standards-based
- No impact to physical network

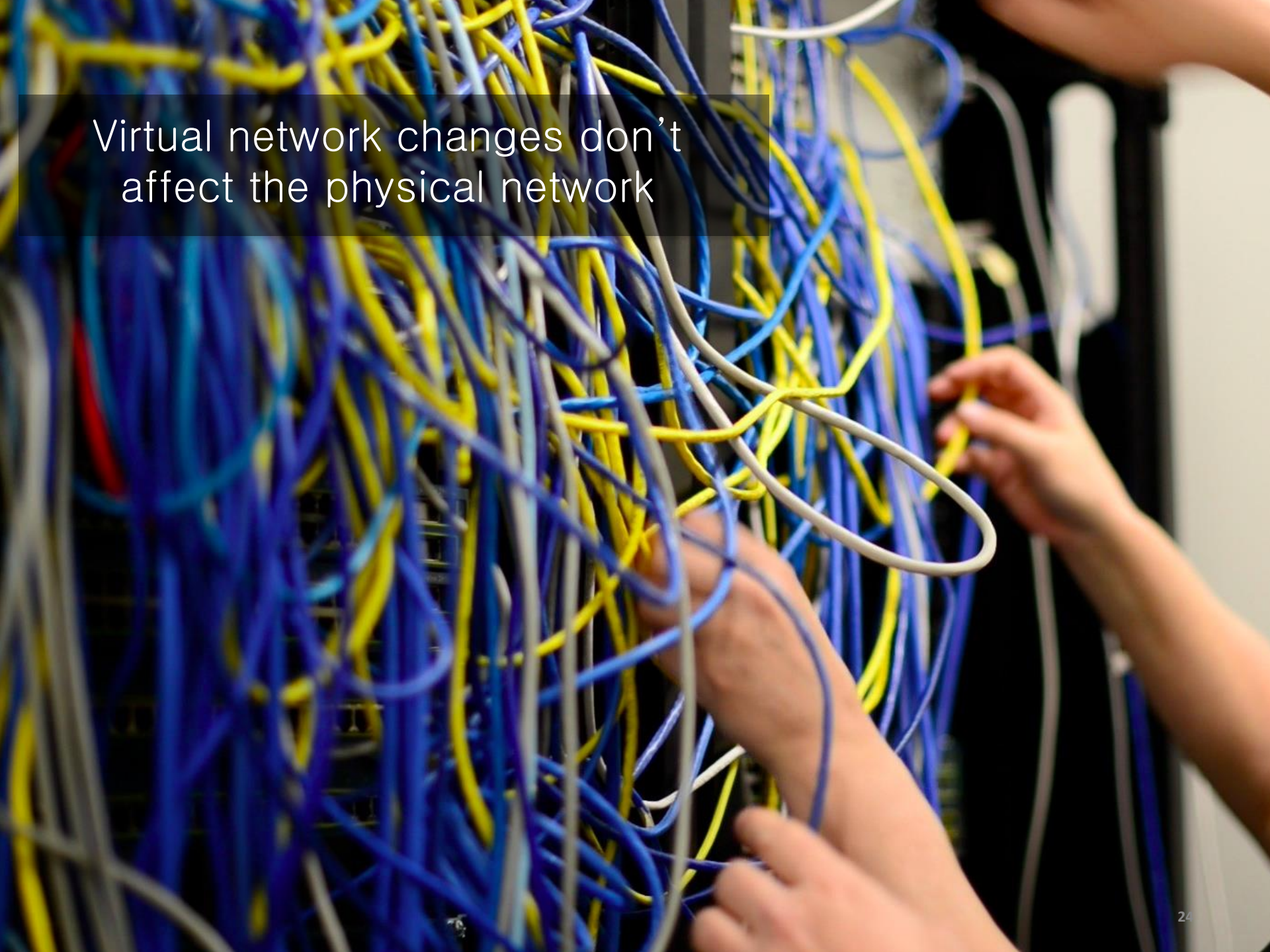
How do overlays achieve real network virtualization?


Encapsulation and Tunneling
Provides isolation

Stateless core. Stateful edge.

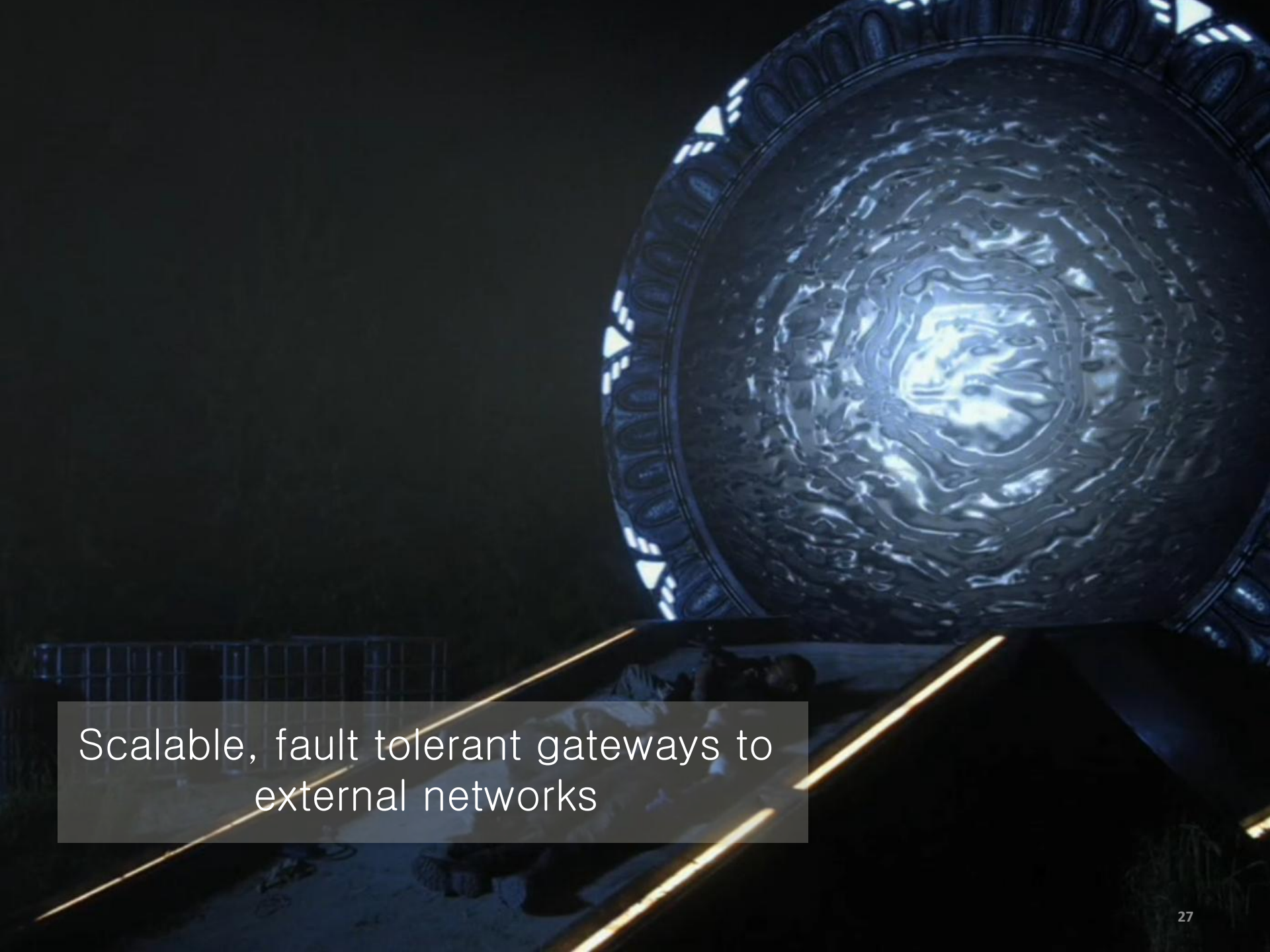



Network processing at the edge
Decoupled from the physical network


Virtual network changes don't
affect the physical network

Single virtual hop network services avoid “traffic trombones”


Centralized state and control for maximum agility


Scalable, fault tolerant gateways to external networks

Using Overlays for NV

- ✓ Multi-tenancy
- ✓ Scalable, fault-tolerant devices (or device-agnostic network services).
- ✓ L2 isolation
- ✓ L3 routing isolation
  - VPC
  - Like VRF (virtual routing and fwd-ing)
- ✓ Scalable Gateways
- ✓ Scalable control plane
  - ARP, DHCP, ICMP
- ✓ Floating/Elastic IPs
- ✓ Stateful NAT
  - Port masquerading
  - DNAT
- ✓ ACLs
- ✓ Stateful (L4) Firewalls
  - Security Groups
- ✓ Load Balancing with health checks
- ✓ Single Pane of Glass (API, CLI, GUI)
- ✓ Integration with management platforms
  - OpenStack, CloudStack
  - vSphere, RHEV, System Center
- ✓ Decoupled from Physical Network

Sounds great, but when will it be a reality?

Network Virtualization Overlays Today

vmware®

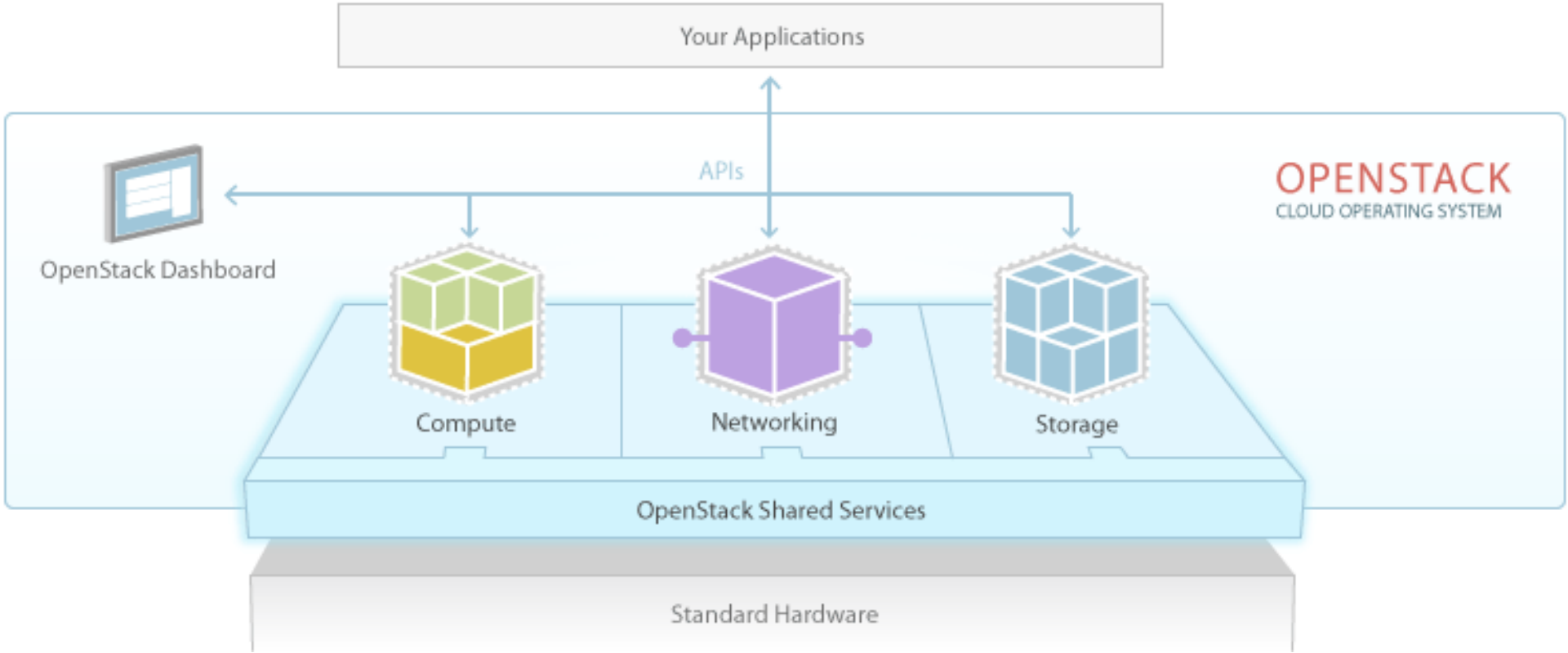
JUNIPER NETWORKS®

Alcatel·Lucent

PLUMgrid

OpenStack

What is OpenStack?



OpenStack Releases

Release schedule: time-based scheme with major release ~ every 6 months
Codenames are alphabetical:

- Austin: The first design summit took place in Austin, TX
- Bexar: The second design summit took place in San Antonio, TX (Bexar county).
- Cactus: Cactus is a city in Texas
- Diablo: Diablo is a city in the bay area near Santa Clara, CA
- Essex: Essex is a city near Boston, MA
- Folsom: Folsom is a city near San Francisco, CA
- Grizzly: Grizzly is an element of the state flag of California (design summit takes place in San Diego, CA)
- Havana: Havana is an unincorporated community in Oregon
- Icehouse: Ice House is a street in Hong Kong
- Juno: Juno is a locality in Georgia
- Kilo: Paris (Sèvres, actually, but that's close enough) is home to the Kilogram, the only remaining SI unit tied to an artifact

Before Neutron: Nova Networking

- Nova-Networking was the only option in OpenStack prior to Quantum/Neutron
- Original method from A release
- No IPv6 in first release but eventually introduced
- Still available today as an alternative to Neutron, but will be phased out

Options Available within nova-networking initially:

- Only Flat
- Flat DHCP

Limitations

- No flexibility with topologies (no 3-tier)
- Tenants can't create/manage L3 Routers
- Scaling limitations (L2 domain)
- No 3rd party vendors supported
- Complex HA model

Nova-network slightly evolves

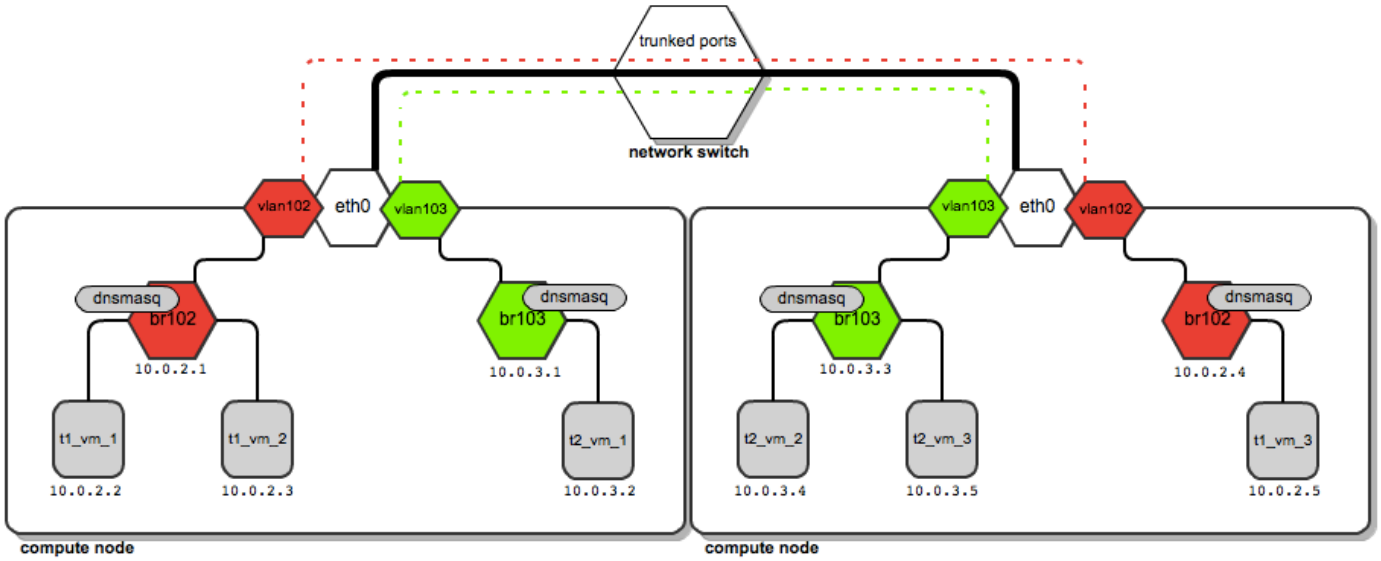
Introduced VLAN DHCP mode

Improvements:

- L2 Isolation – each project gets a VLAN assigned to it

Limitations

- Need to pre-configure VLANs on physical network
- Scaling Limitations - VLANs
- No L3
- No 3-tier topologies
- No 3rd party vendors



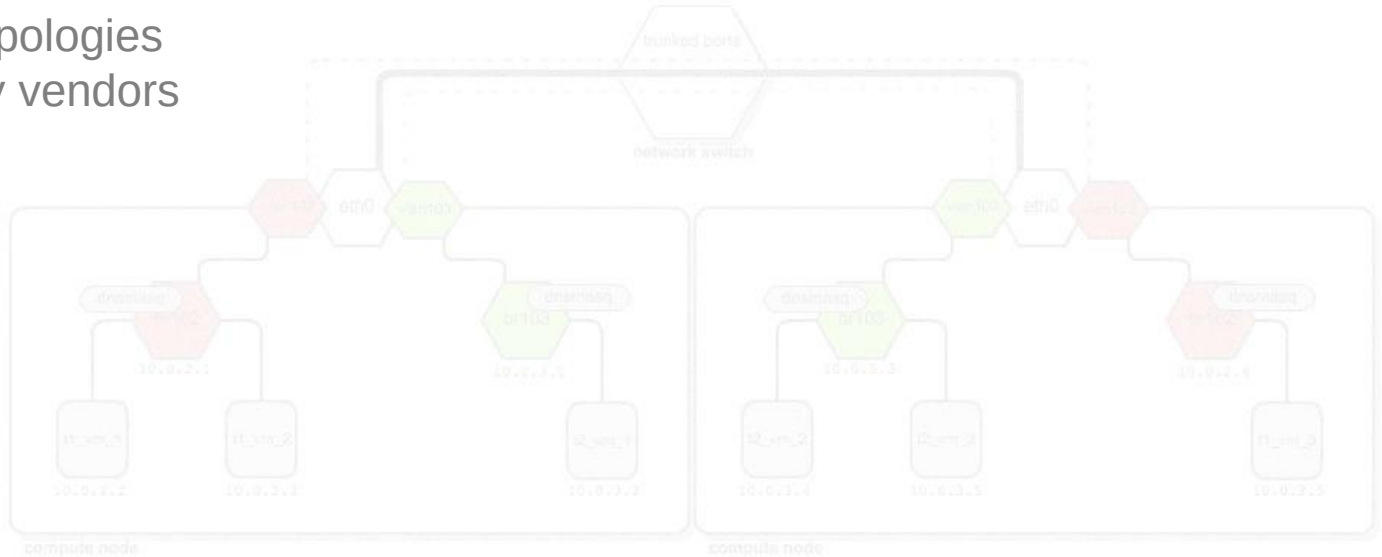
Nova-network slightly evolves

C & D Releases had two general categories:

- Flat Networking
- VLAN Networking

Limitations

- Need to pre-configure VLANs on physical network
- Scaling Limitations - VLANs
- No L3
- No 3-tier topologies
- No 3rd party vendors



Quantum

OpenStack Networking branches out of the Nova project

- Tech Preview of Quantum appeared in D release
- Brought ability to have a multi-tiered network, with isolated network segments for various applications or customers
- Quantum-server allowed for Python daemon to expose the OpenStack Networking API and passes requests to 3rd party plugins
- Officially released in Folsom Release

Introducing Neutron

- Name Change from Quantum to Neutron was announced in April 2013
- Legal Agreement to phase out code name “Quantum” due to trademark of Quantum Corporation

OpenStack Networking as a First Class Service

- Pluggable Architecture
- Standard API
- Many choices
- More Services (LBaaS, VPNaaS)
- Flexible network topologies

Plugins Available

- **MidoNet**
- OVS Plugin
- Linux Bridges
- Flat DHCP
- VLAN DHCP
- ML2
- NSX
- Plumgrid
- Nuage
- Contrail
- Ryu

Evolution of Neutron

Release Name	Release Date	Included Components
Austin	21 October 2010	Nova, Swift
Bexar	3 February 2011	Nova, Glance, Swift
Cactus	15 April 2011	Nova, Glance, Swift
Diablo	22 September 2011	Nova, Glance, Swift
Essex	5 April 2012	Nova, Glance, Swift, Horizon, Keystone
Folsom	27 September 2012	Nova, Glance, Swift, Horizon, Keystone, Quantum, Cinder
Grizzly	4 April 2013	Nova, Glance, Swift, Horizon, Keystone, Quantum, Cinder
Havana	17 October 2013	Nova, Glance, Swift, Horizon, Keystone, Neutron, Cinder
Icehouse	April 2014	Nova, Glance, Swift, Horizon, Keystone, Neutron, Cinder

Latest Neutron Features

Havana Release Brought:

- LBaaS: shipped an updated API and HAProxy driver support
- VPNaaS: VPN API supports IPSec and L3 agent ships with an OpenSwan driver
- FWaaS: enables tenant to configure security at the edge via the firewall API and on the VIF via the security group API
- New plug-in Modular Layer 2 (ML2): ML2 plugin supports local, flat, VLAN, GRE and VXLAN network types via a type drivers and different mechanism drivers

Icehouse Release:

- New vendor plugins, LBaaS drivers and VPNaaS drivers
- OVS plugin and Linux Bridge plugin are deprecated: The ML2 plugin combines OVS and Linux Bridge support into one plugin
- Neutron team has extended support for legacy Quantum configuration file options for one more release

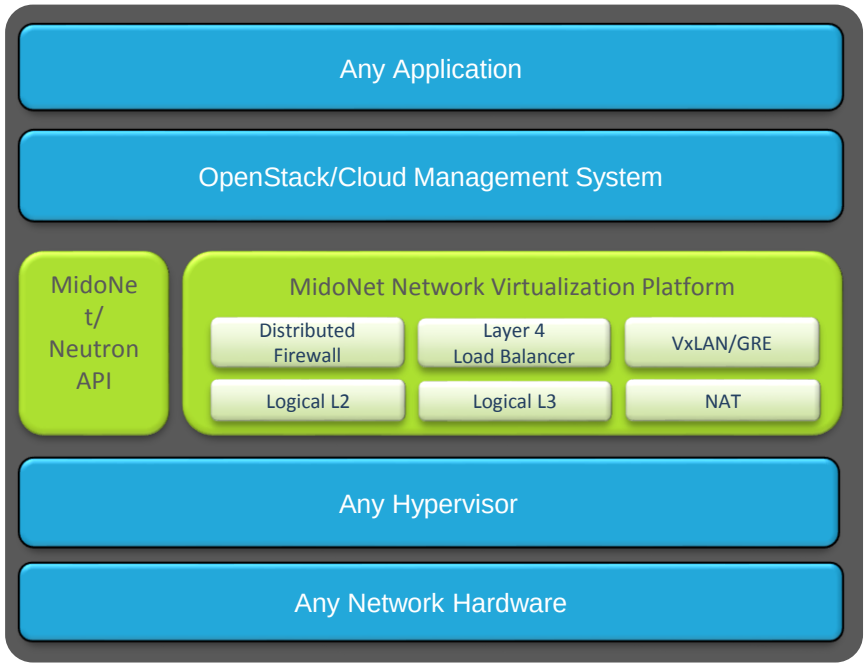
Upcoming Neutron Features

Expectations for Juno:

- Provide Distributed Virtual Routing (DVR) functionality: Define API to create and deploy DVRs to improve the performance
- Group-based Policy Abstractions for Neutron: API extensions for easier consumption of the networking resources by separate organizations and management systems
- IPv6 advancements:
  - Add RADVD to namespace to handle RAs,
  - Stateful and stateless DHCP for IPv6
- LBaaS new API driver and object model improvement for complex cases
- Quotas extension support in MidoNet plugin
- Incubator system:
  - Instead of only using the summit for developing new features, features can be developed and gestate over time

MidoNet Overview

MidoNet Network Virtualization Platform



Logical L2 Switching - L2 isolation and path optimization with distributed virtual switching
Interconnect with VLAN enabled network via L2 Gateway

Logical L3 Routing – L3 isolation and routing between virtual networks
No need to exit the software container - no hardware required

NAT – Provides Dynamic NAT, Port masquerading

Distributed Firewall – Provides ACLs, high performance kernel integrated firewall via a flexible rule chain system

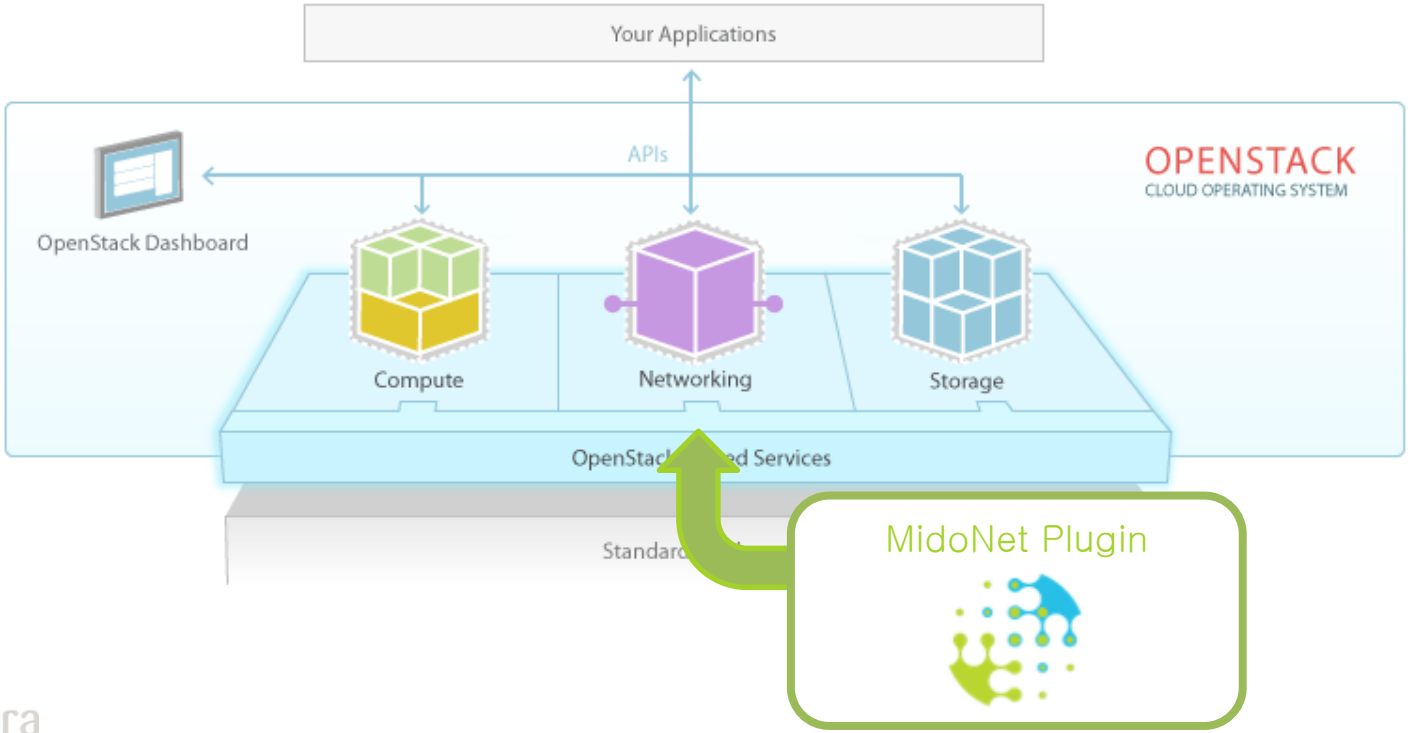
Logical Layer 4 Load Balancer – Provides application load balancing in software form - no need for hardware based firewalls

VxLAN/GRE – Provides VxLAN and GRE tunneling
Provides L2 connectivity across L3 transport. This is useful when L2 fabric doesn't reach all the way from the racks hosting the VMs to the physical L2 segment of interest.

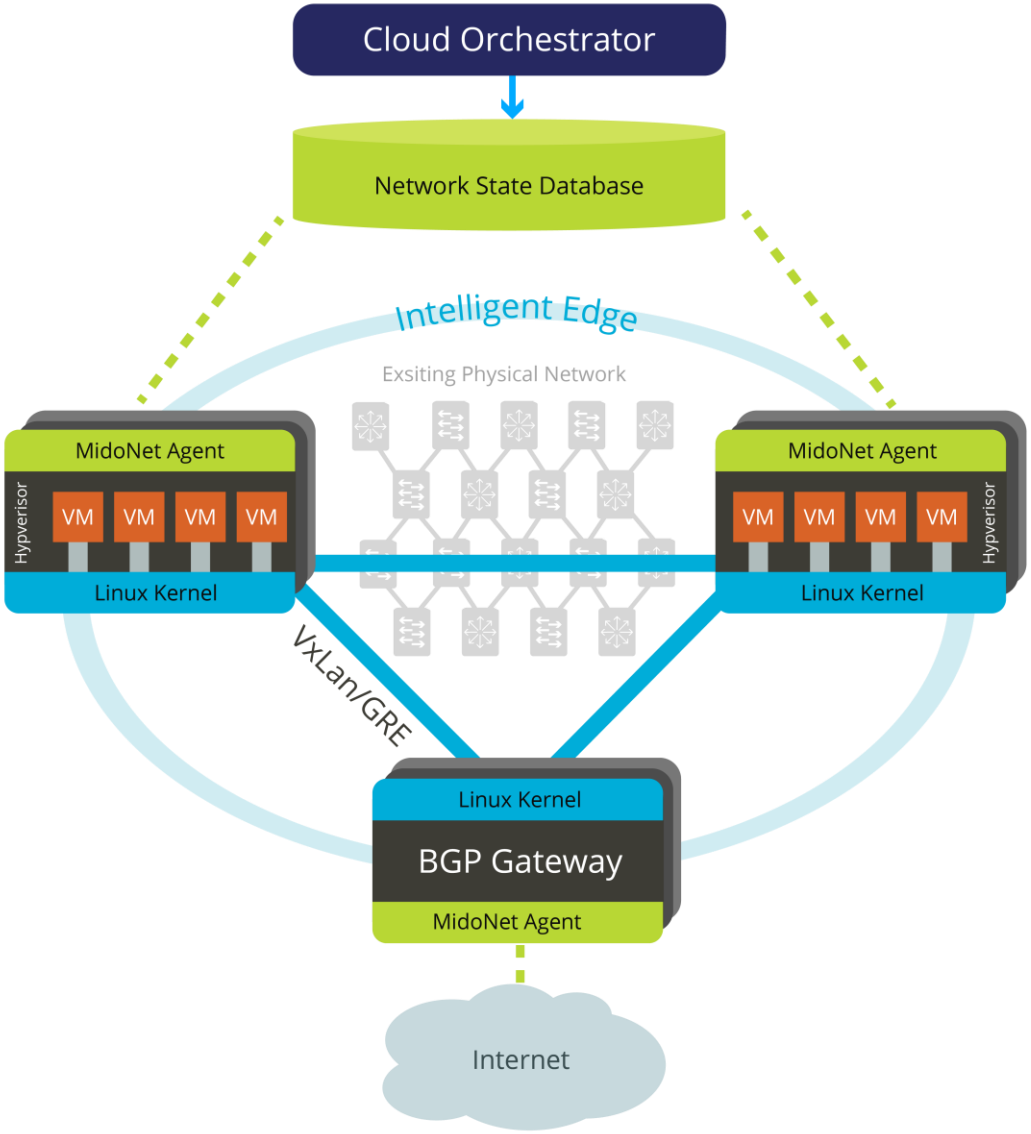
MidoNet/Neutron API– Alignment with OpenStack Neutron's API for integration into compatible cloud management software

OpenStack Integration

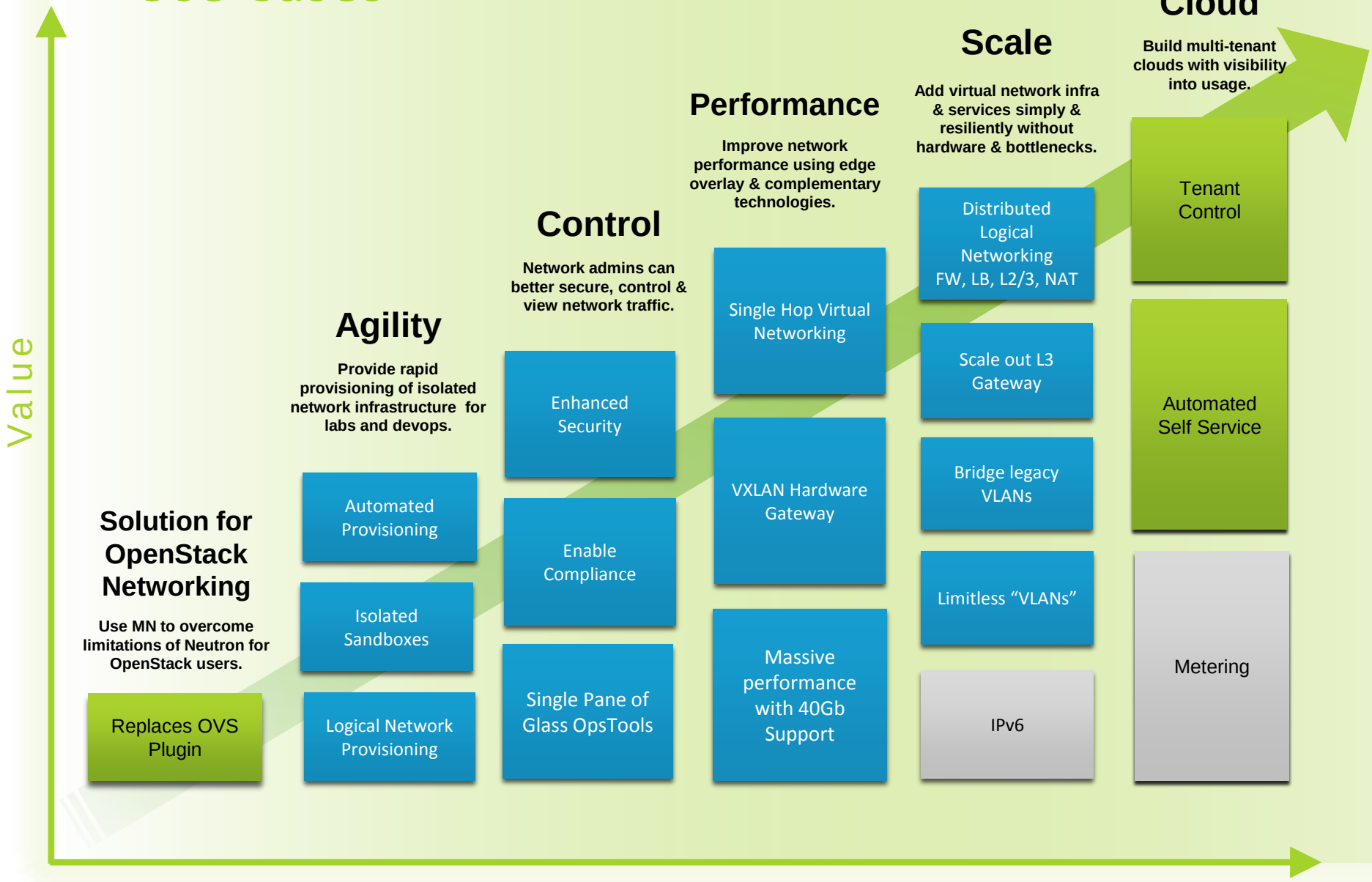
Easy integration with OpenStack:
MidoNet provides a plugin for Neutron.



Architecture Overview



Use Cases



midokura

So what's next for Network Virtualization?


Get more out of the physical network.

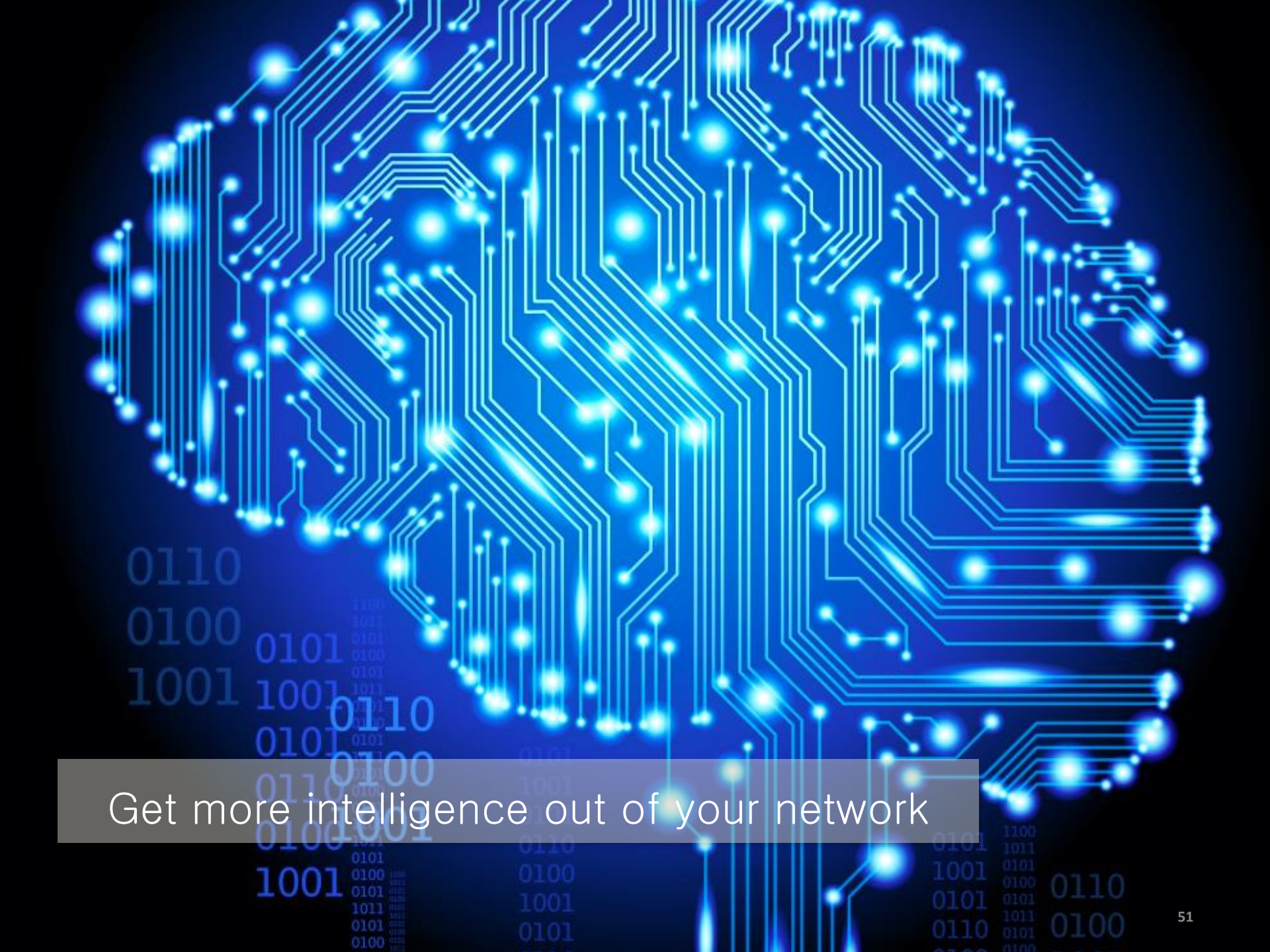
Network Virtualization decouples the logical network from the physical network.

NVOs can't ignore the physical network

Dynamic changes to logical network are **not dependent** on the **physical network** configuration.

Sharing state to and from the physical network can be **supplementary**.

- Monitoring
- Traffic Engineering

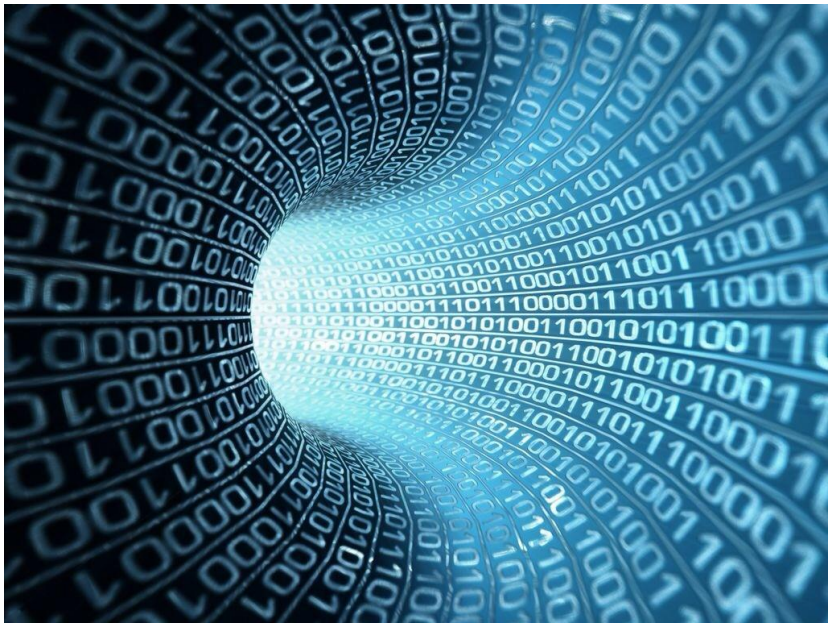

Get more intelligence out of your network

NVOs provide a wealth of information

NVOs **centralize information** on your network

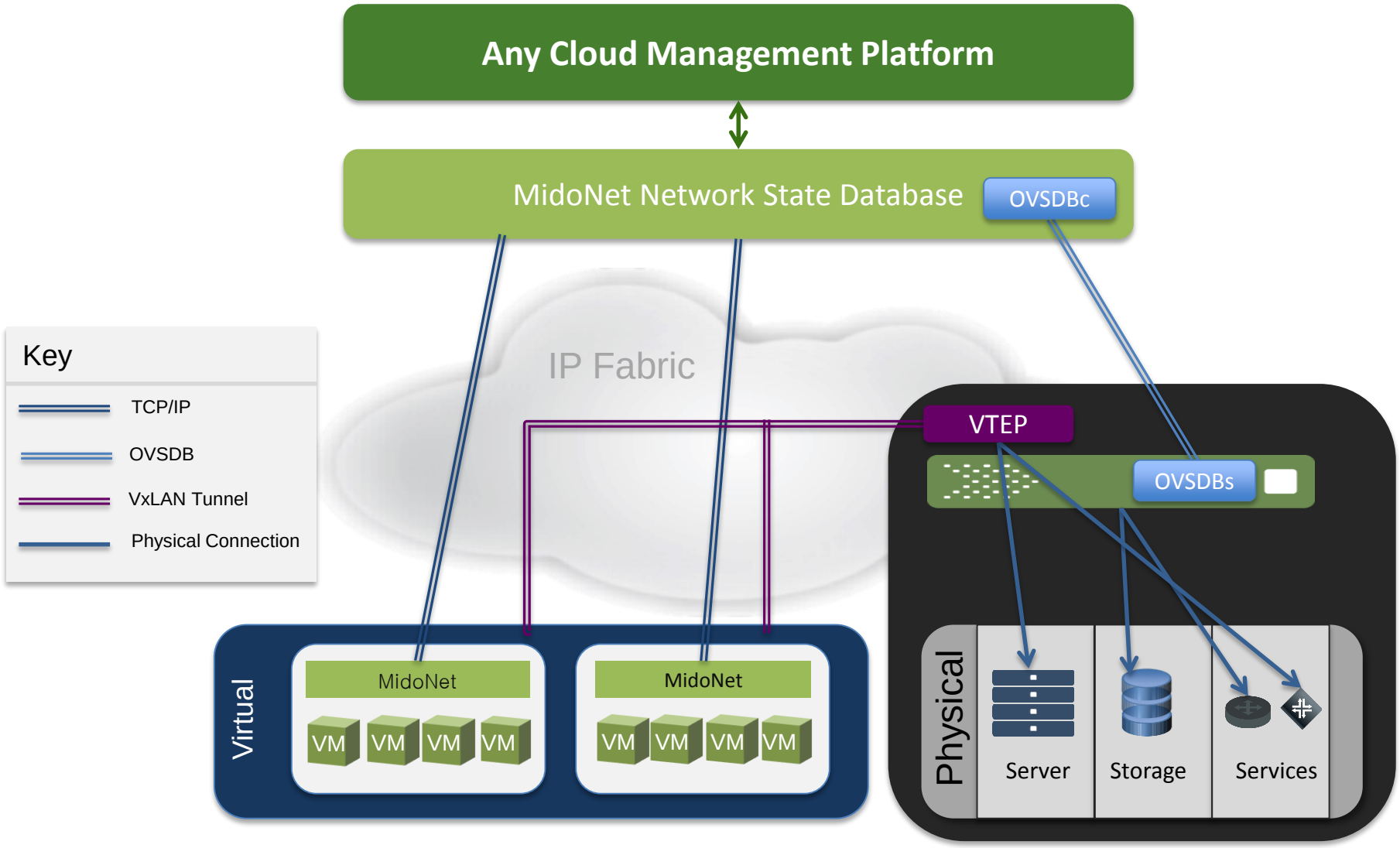
We can start taking advantage of this information

- Security
- Compliance
- Optimizing Networks




Bridge physical and virtual networks more efficiently

Midokura VTEP Solution

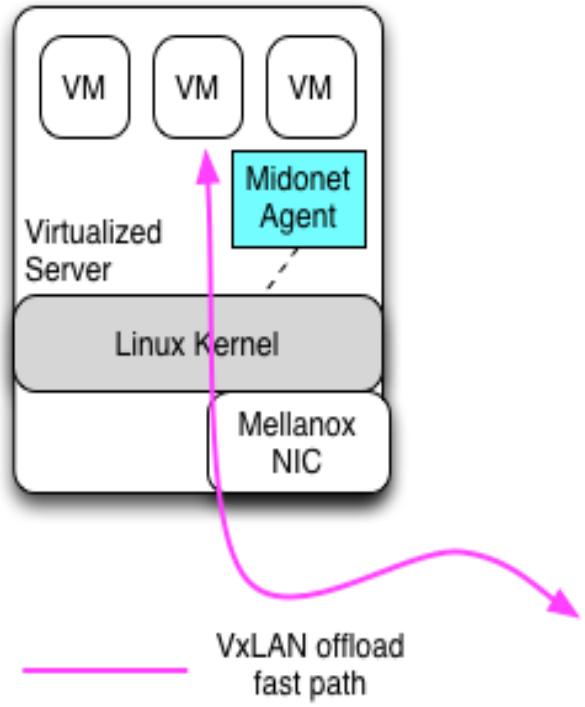



Break through performance barriers
of software networking

Performance

40Gb VxLAN Offloading: virtualized environments require high throughput infrastructure

- Integration with Mellanox provides 40 Gbps saturation
- VxLAN offloading improves CPU utilization levels
- Scale with performance through HW interconnect
- Increase throughput with offloading where no offloading would otherwise have flat results
- High bandwidth can now be achieved in software



Q&A

MidoNet Advantages

Check out our blog:

http://blog.midokura.com/

http://blog.midonet.org

Follow us on Twitter:

@midokura

@midonet

Thank You