

Overlay-based Virtual Networking for IaaS Clouds

Dan Mihai Dumitriu, dan@midokura.com

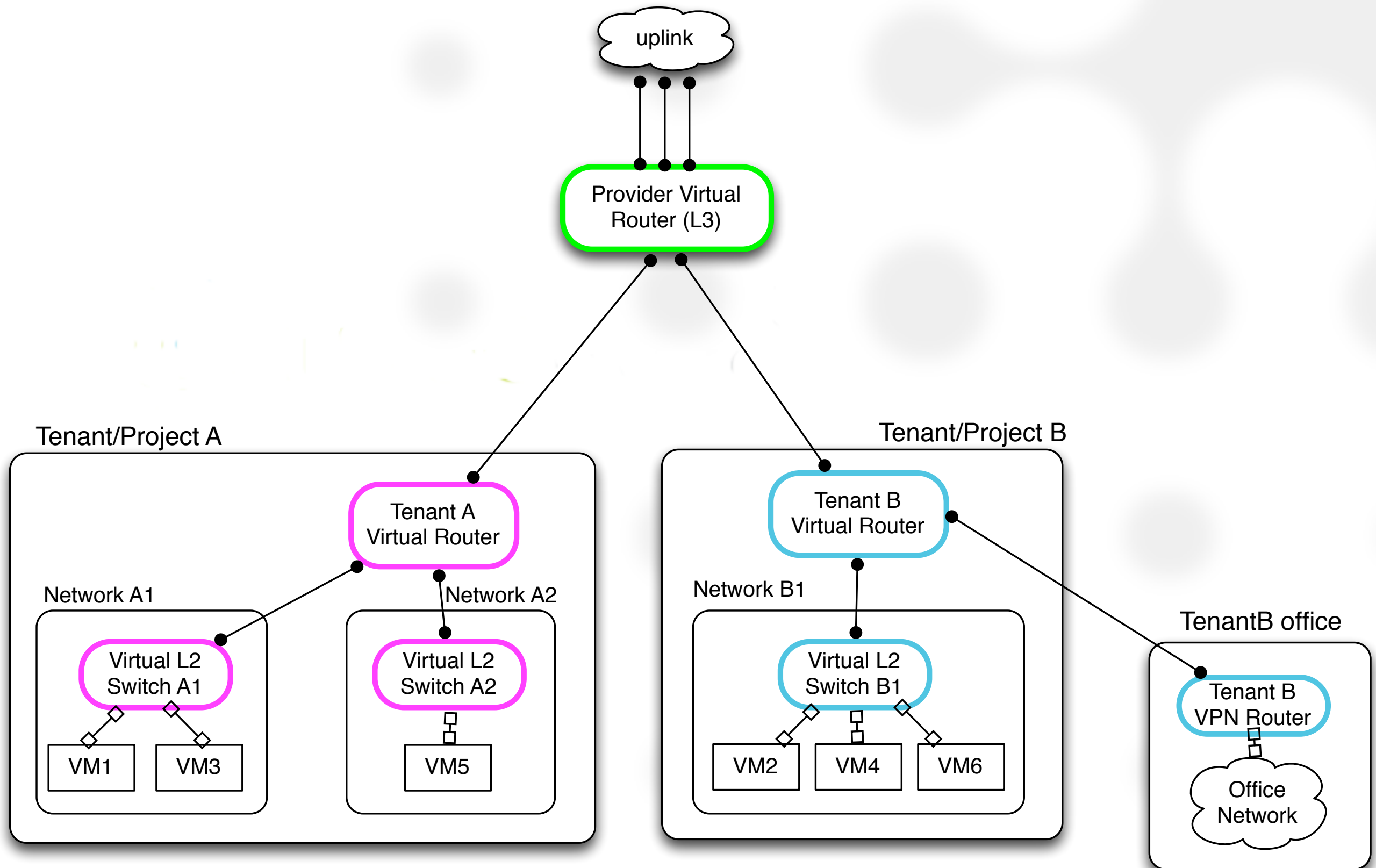
CTO, Midokura

KOSUG - December 4, 2012

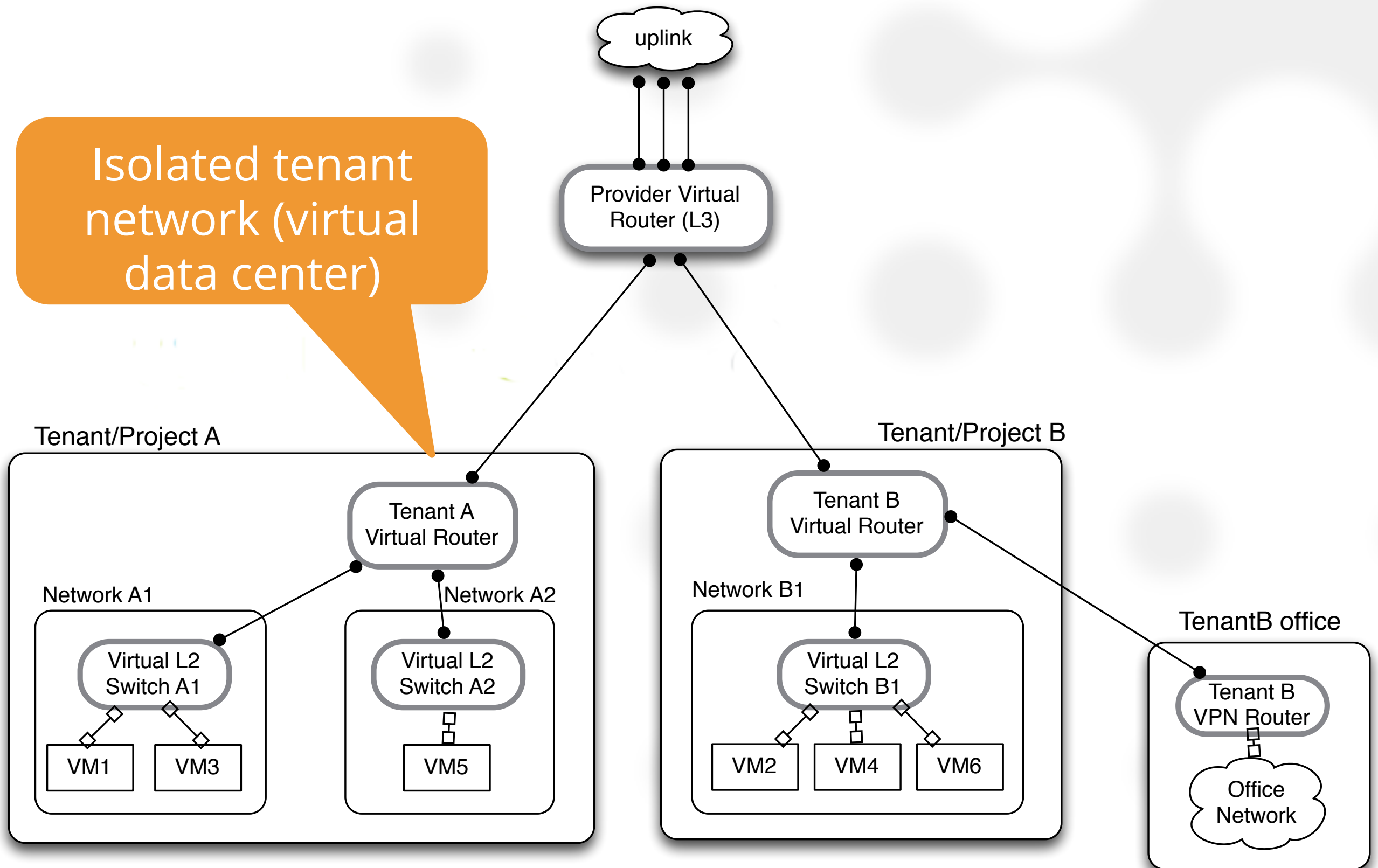


- Cloud tenant networking requirements
- How to build it:
 - ◆ Virtualized physical devices
 - ◆ OpenFlow switch fabric
 - ◆ IP overlays
- Choose overlays, but what about the control plane?
- MidoNet SDN solution
- Questions

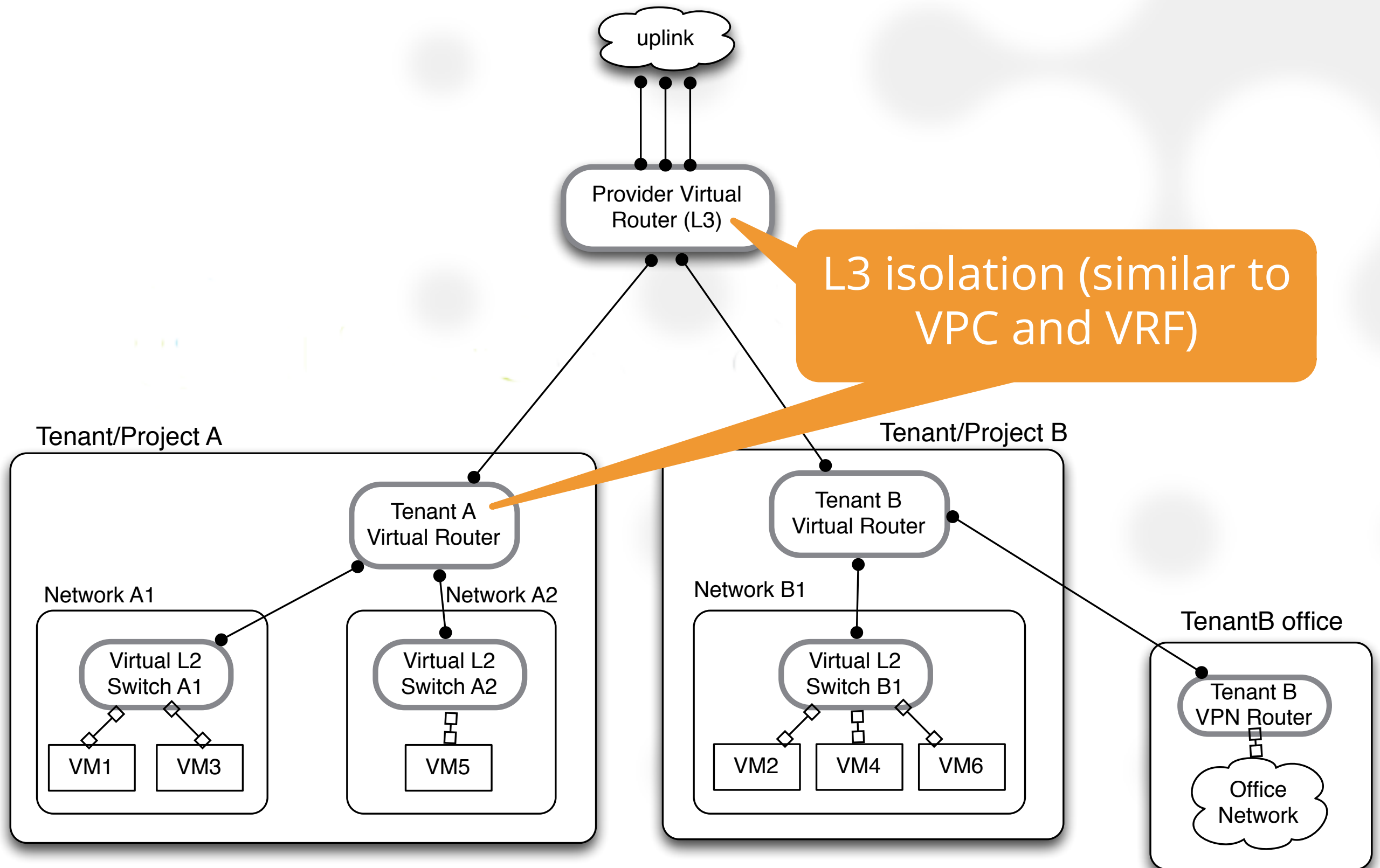
Requirements



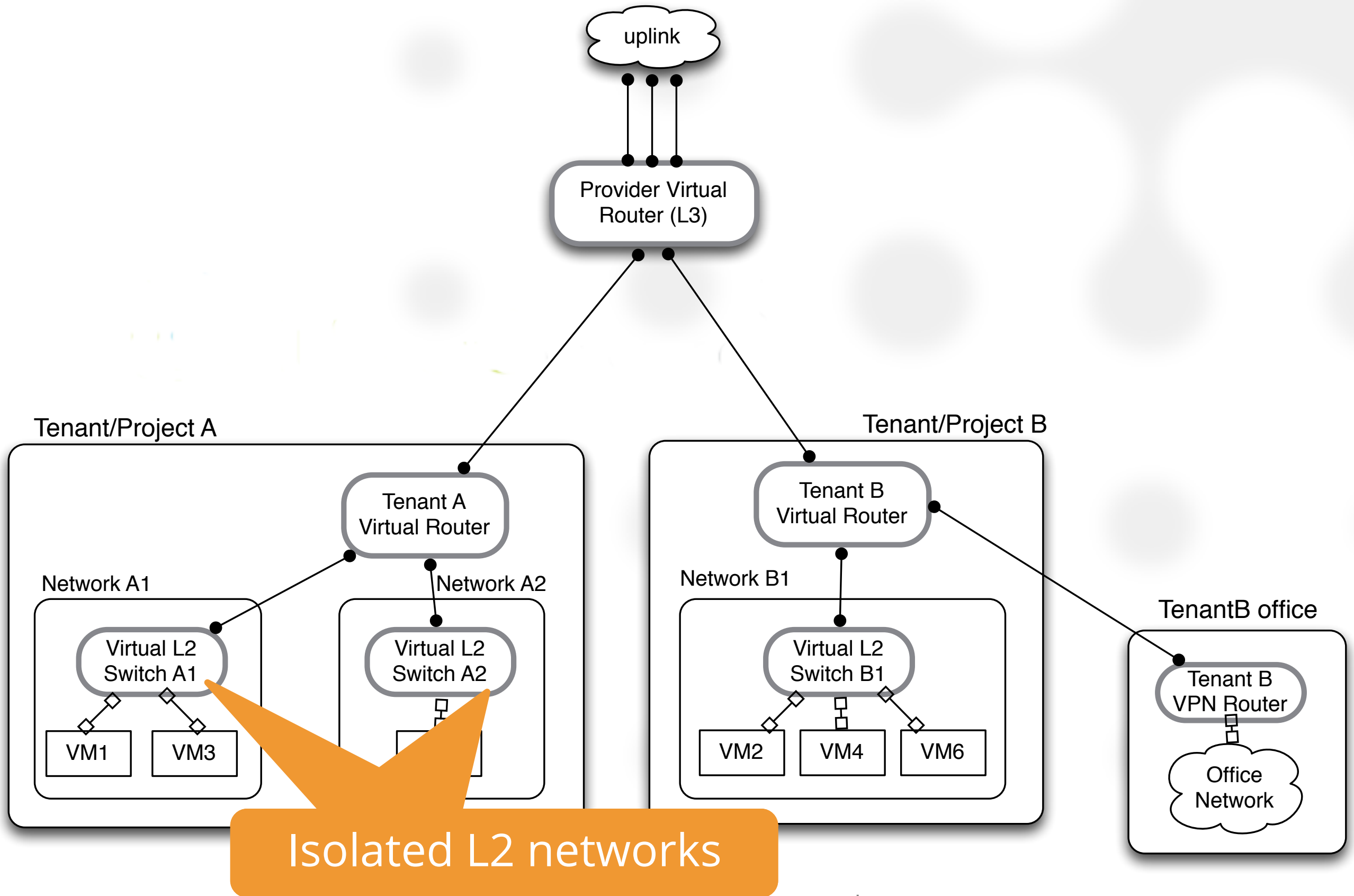
Requirements



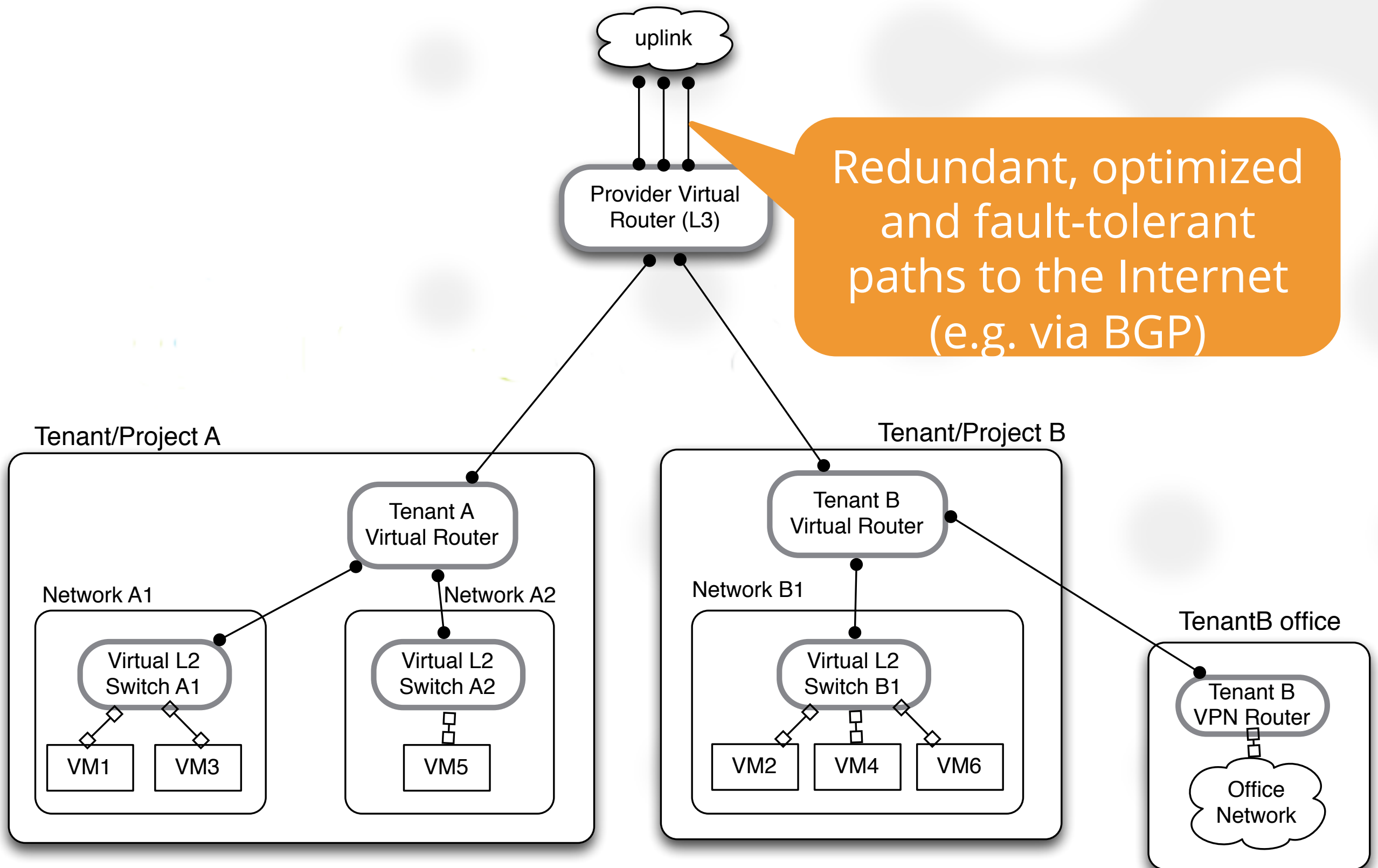
Requirements



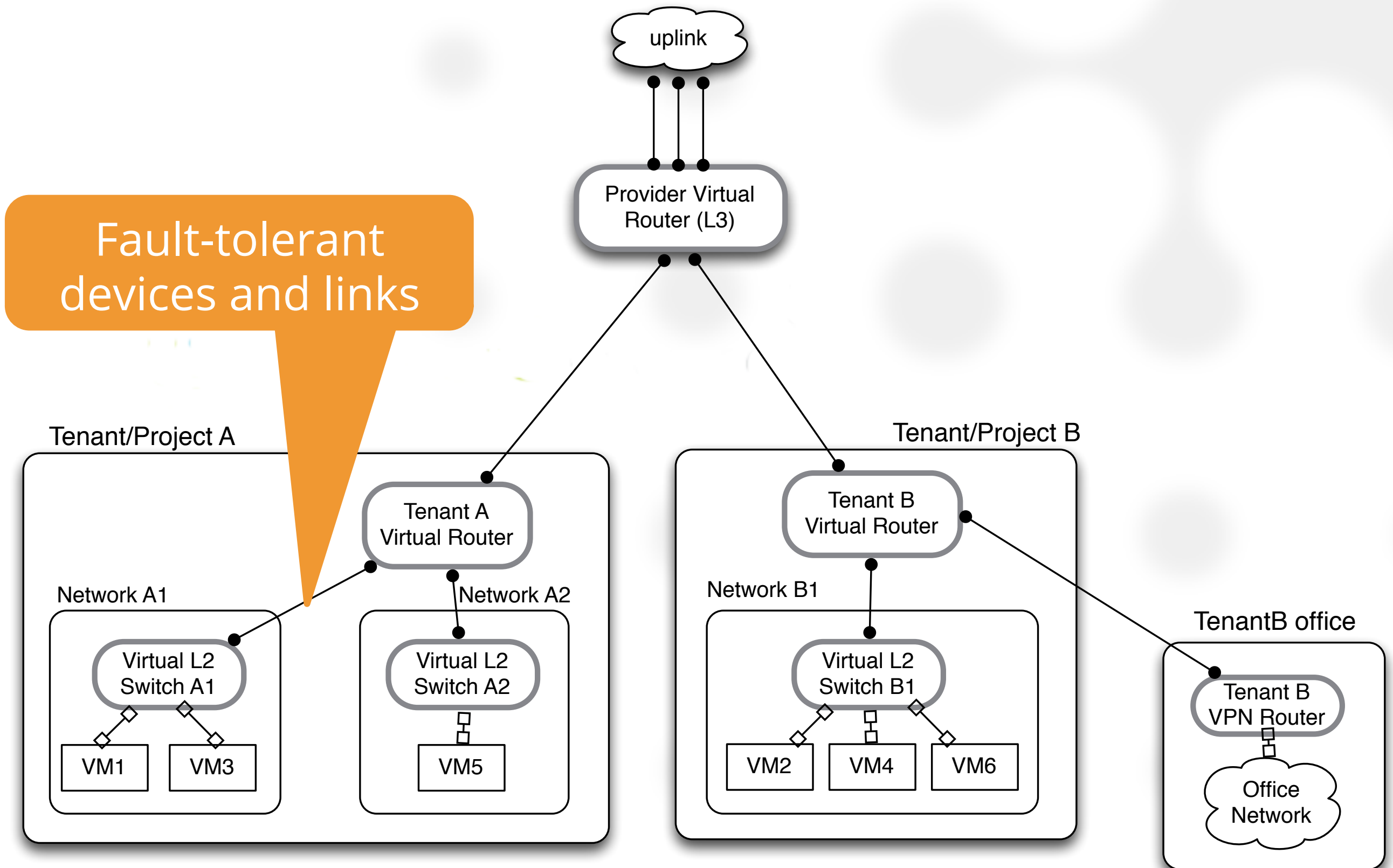
Requirements



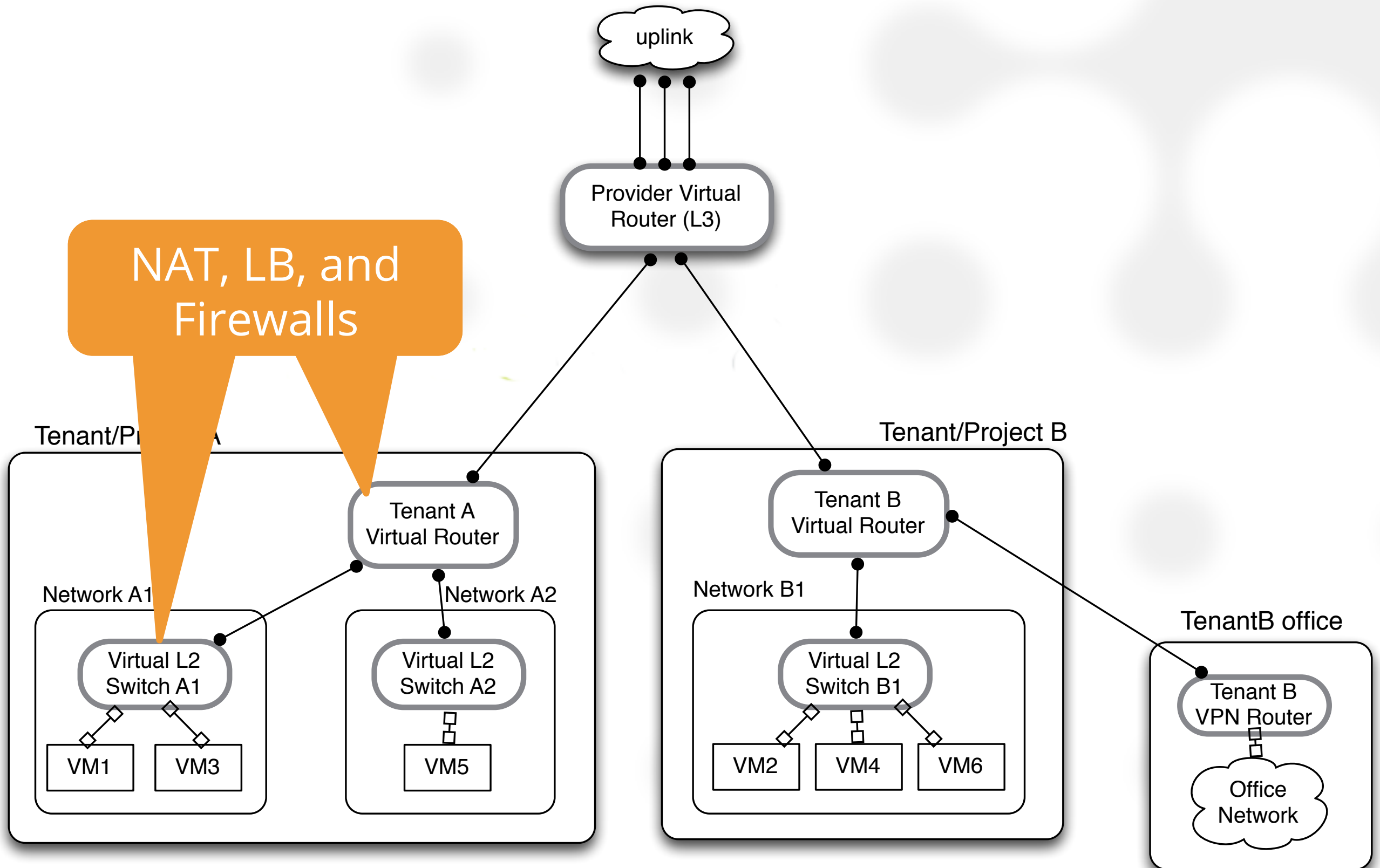
Requirements



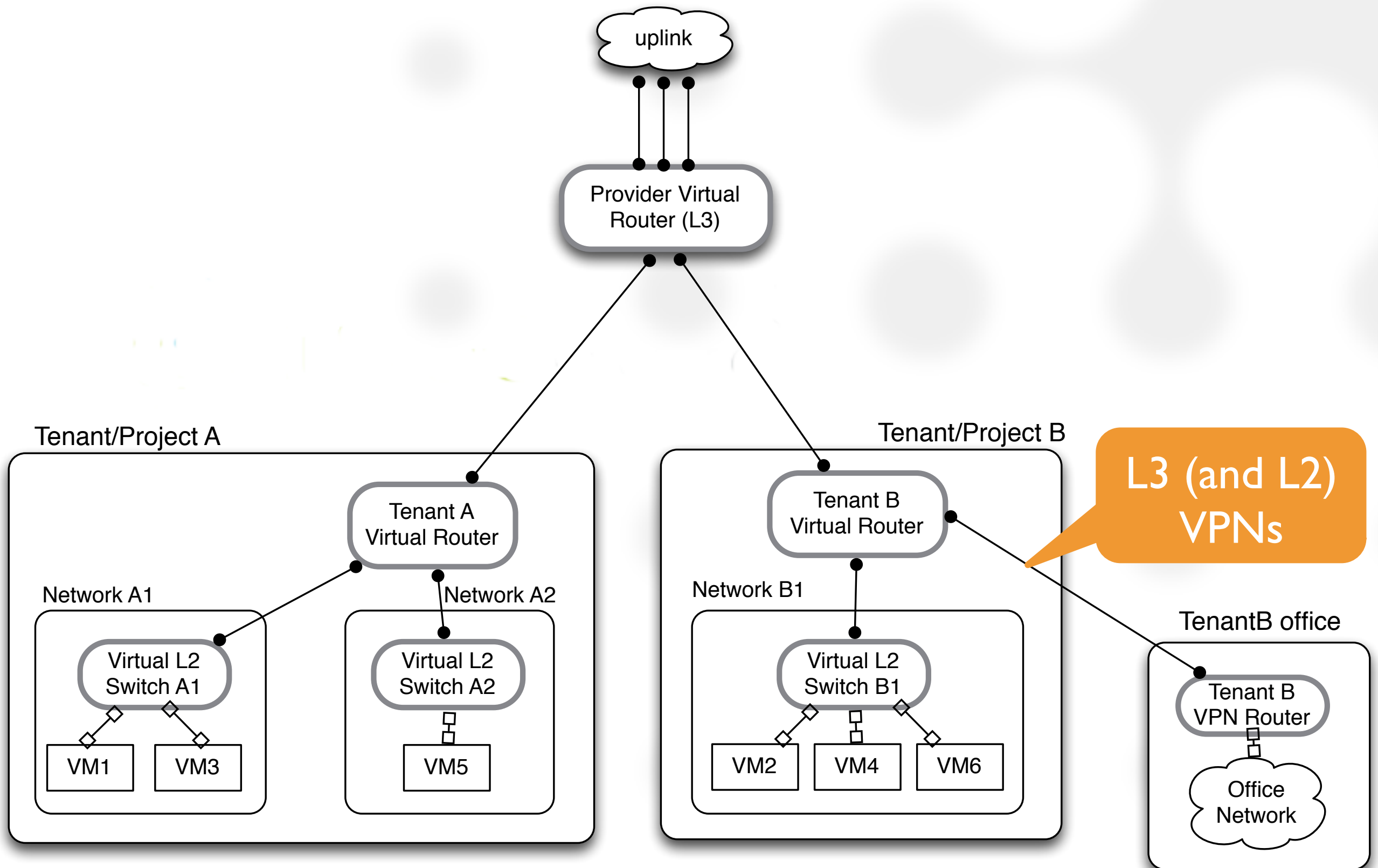
Requirements



Requirements



Requirements



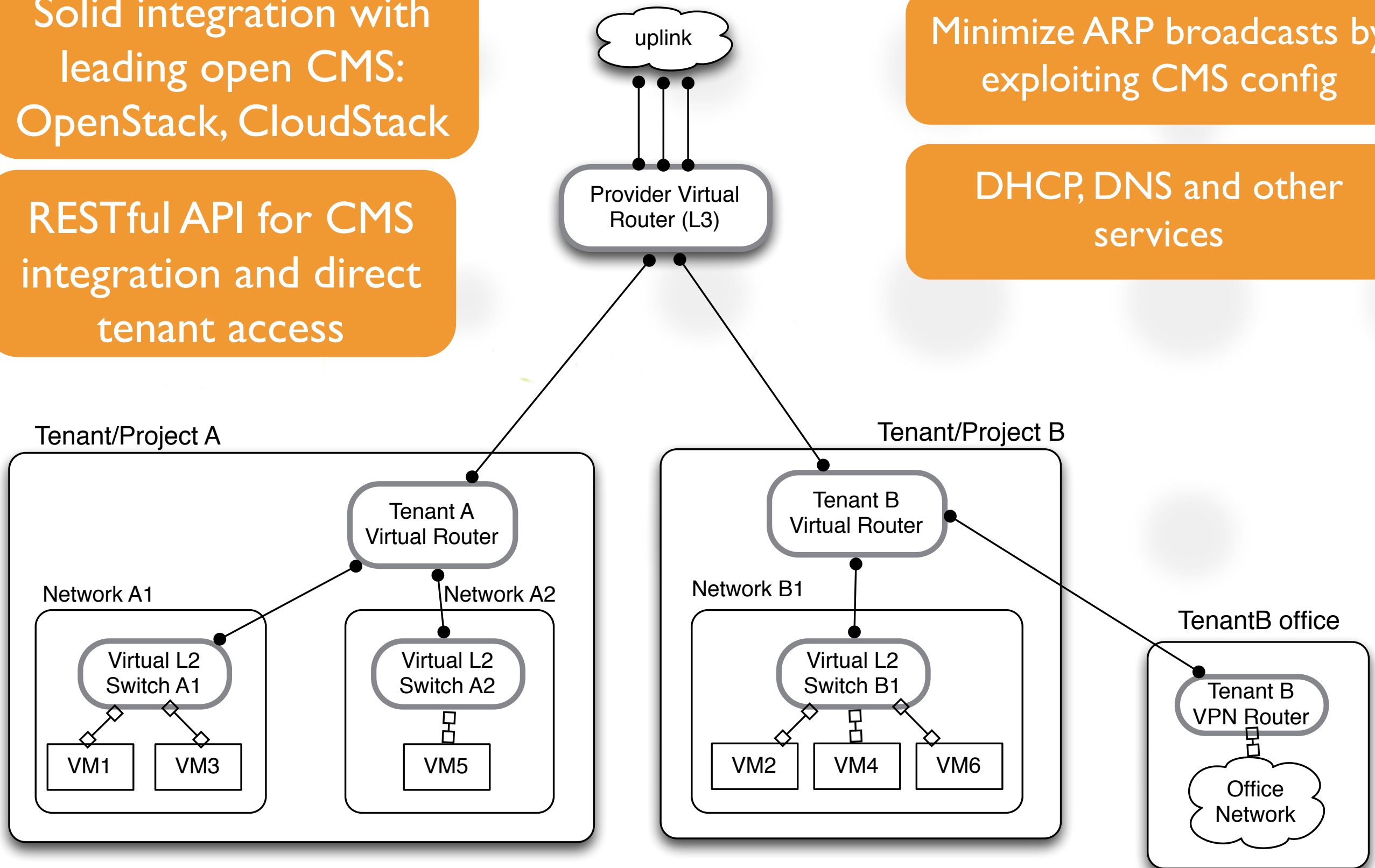
Requirements

Solid integration with leading open CMS: OpenStack, CloudStack

RESTful API for CMS integration and direct tenant access

Minimize ARP broadcasts by exploiting CMS config

DHCP, DNS and other services



Requirements: recap

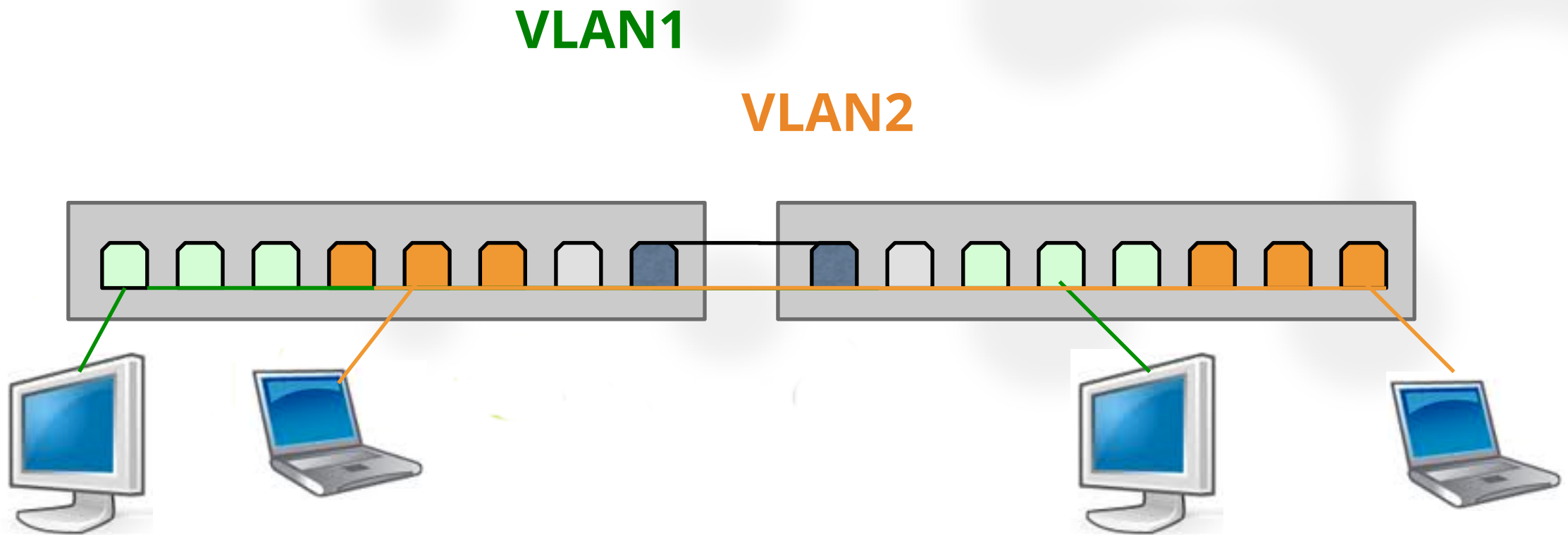
- Multi-tenancy
 - Scalable, fault-tolerant devices (or device-agnostic network services).
 - L2 isolation
 - L3 routing isolation
 - ◆ VPC
 - ◆ Like VRF (virtual routing and fwd-ing)
 - BGP gateway
 - Scalable control plane
 - ◆ ARP, DHCP, ICMP
 - Floating IP
- Stateful NAT
 - ◆ Port masquerading
 - ◆ DNAT
 - ACLs
 - Stateful (L4) Firewalls
 - ◆ Security Groups
 - LB health checks
 - VPNs at L2 and L3
 - ◆ IPSec
 - REST API
 - Integration with CMS
 - ◆ OpenStack
 - ◆ CloudStack

How to build it?

- 1.Virtualized physical devices
- 2.Centrally controlled OpenFlow-based hop-by-hop switching fabric
- 3.Edge to edge overlays

1 Virtualized physical devices

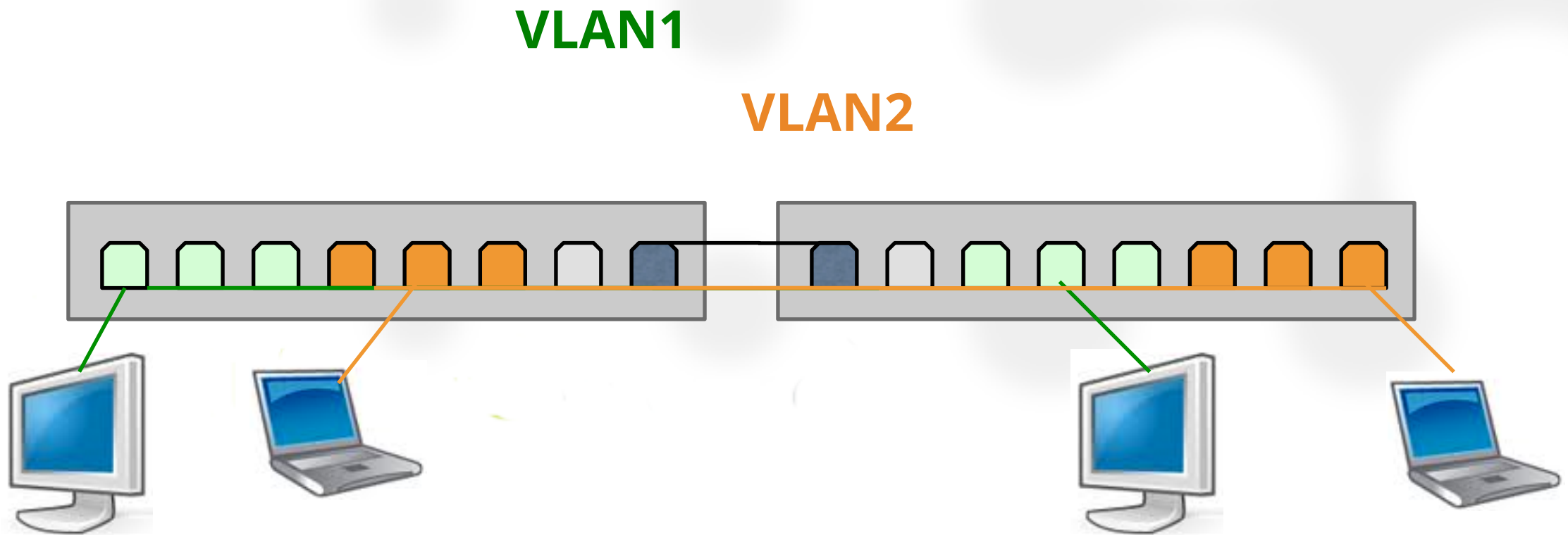
VLAN



- 4096 limit on number of unique tags
- Large spanning trees terminating on many hosts
- High churn in switch control planes due to MAC learning
- Need MLAG for L2 multi-path (vendor specific)

1 Virtualized physical devices

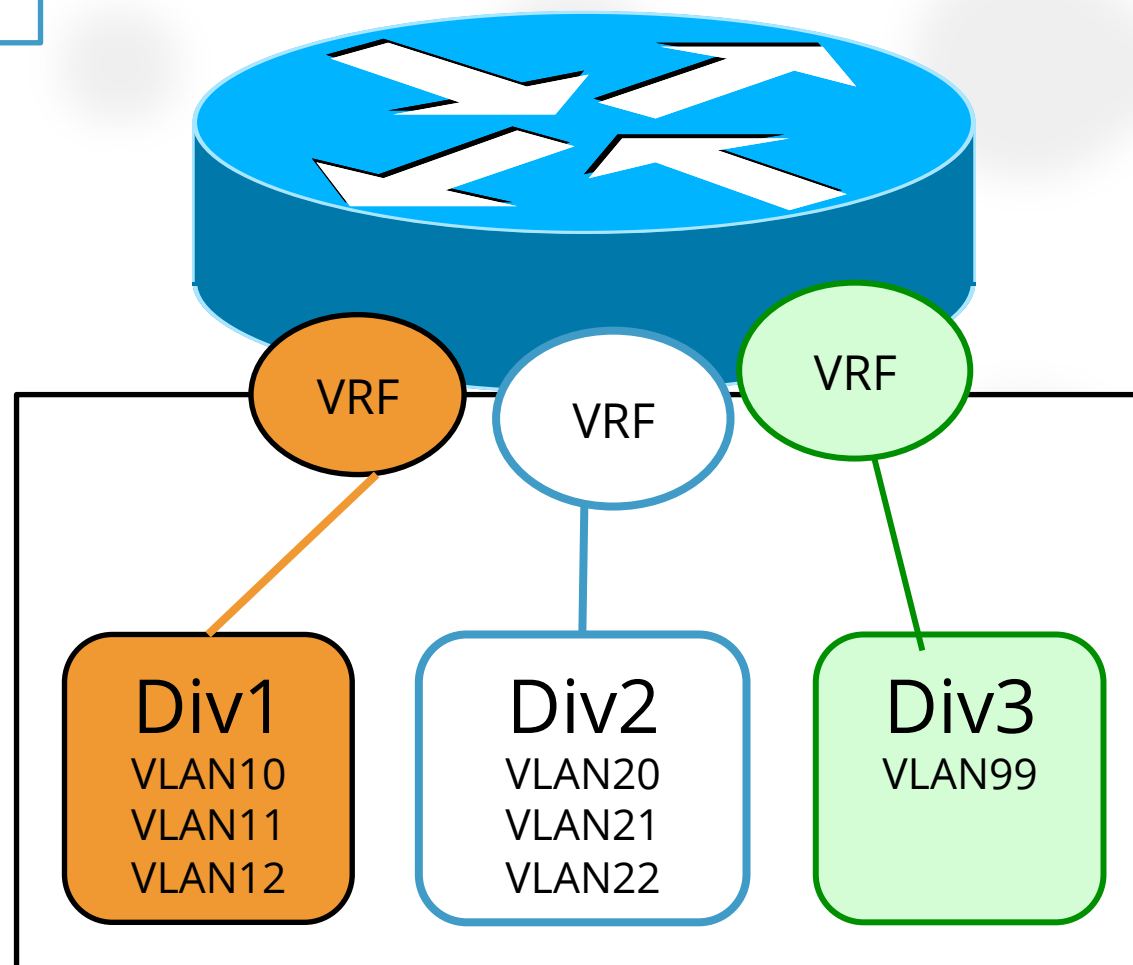
VLAN (more)



- L2 isolation
- What about L3 and Internet access?
- Use VRF or virtual appliances? Fault-tolerance?

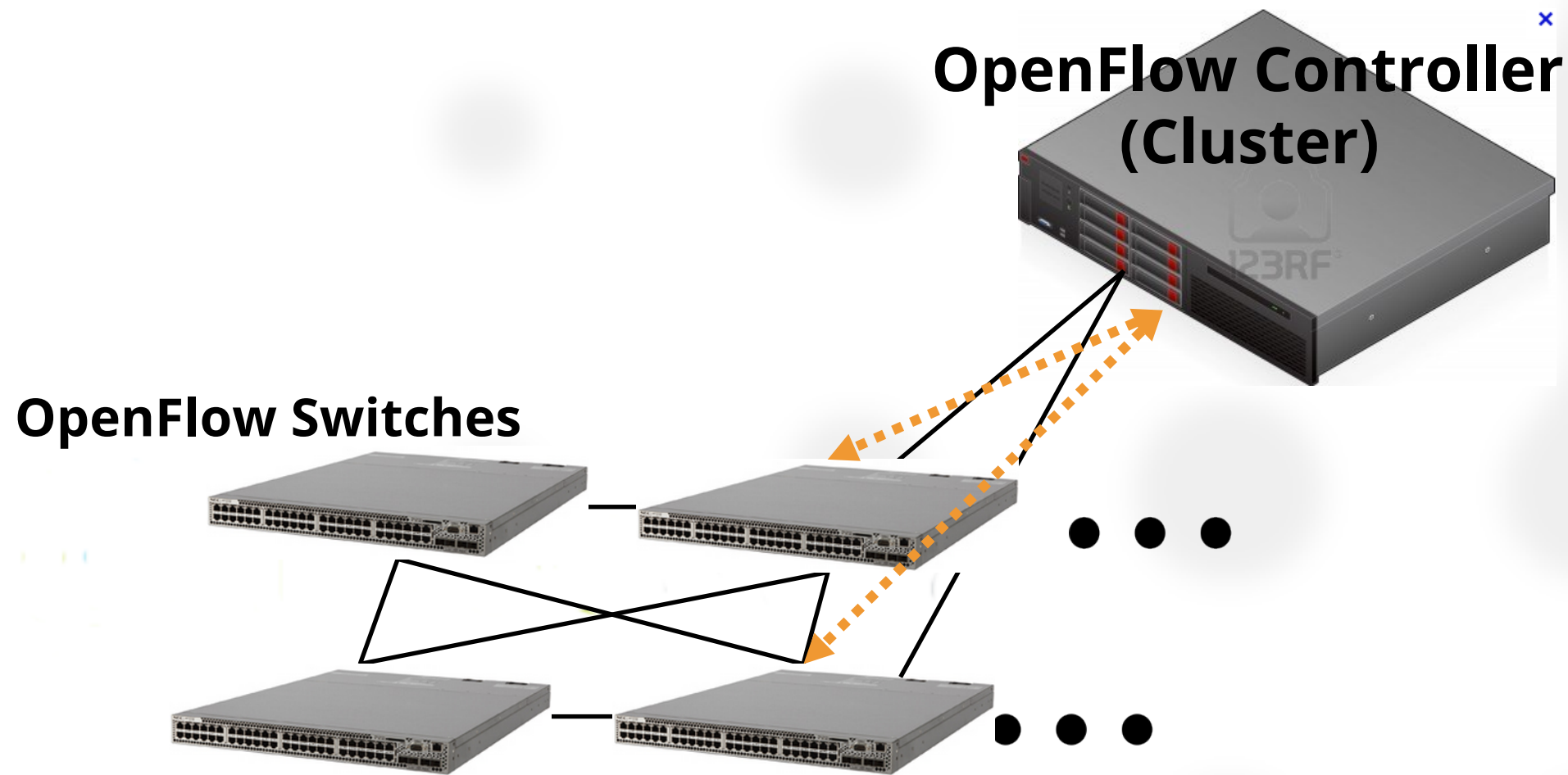
1 Virtualized physical devices

VRF

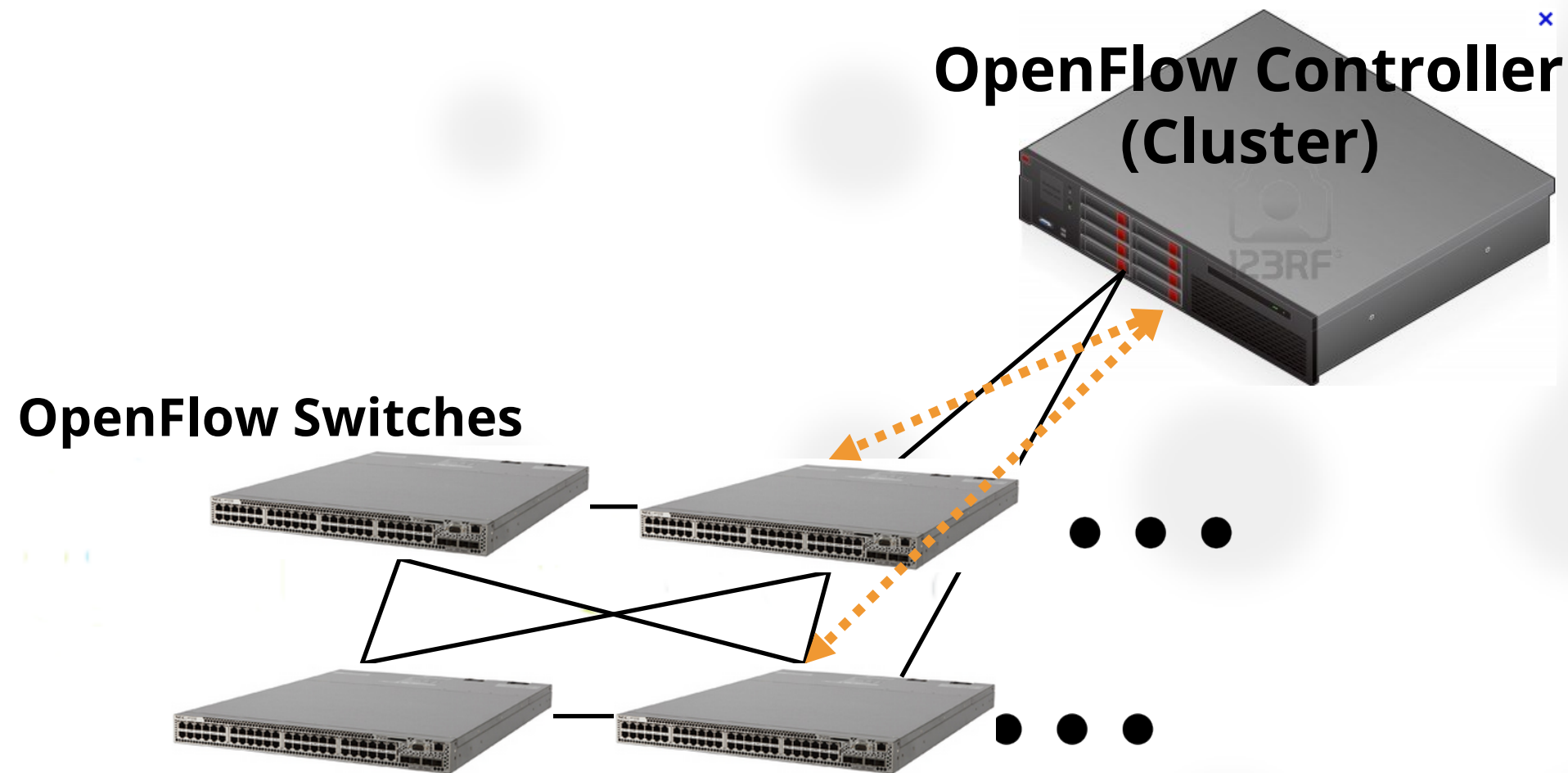


出典 : <http://infrastructureadventures.com/tag/vrf-lite/>

- Not scalable to cloud scale
- Expensive hardware
- Not fault tolerant (HSRP?)
- L2 and L3 isolation. What about NAT, LB, FW?



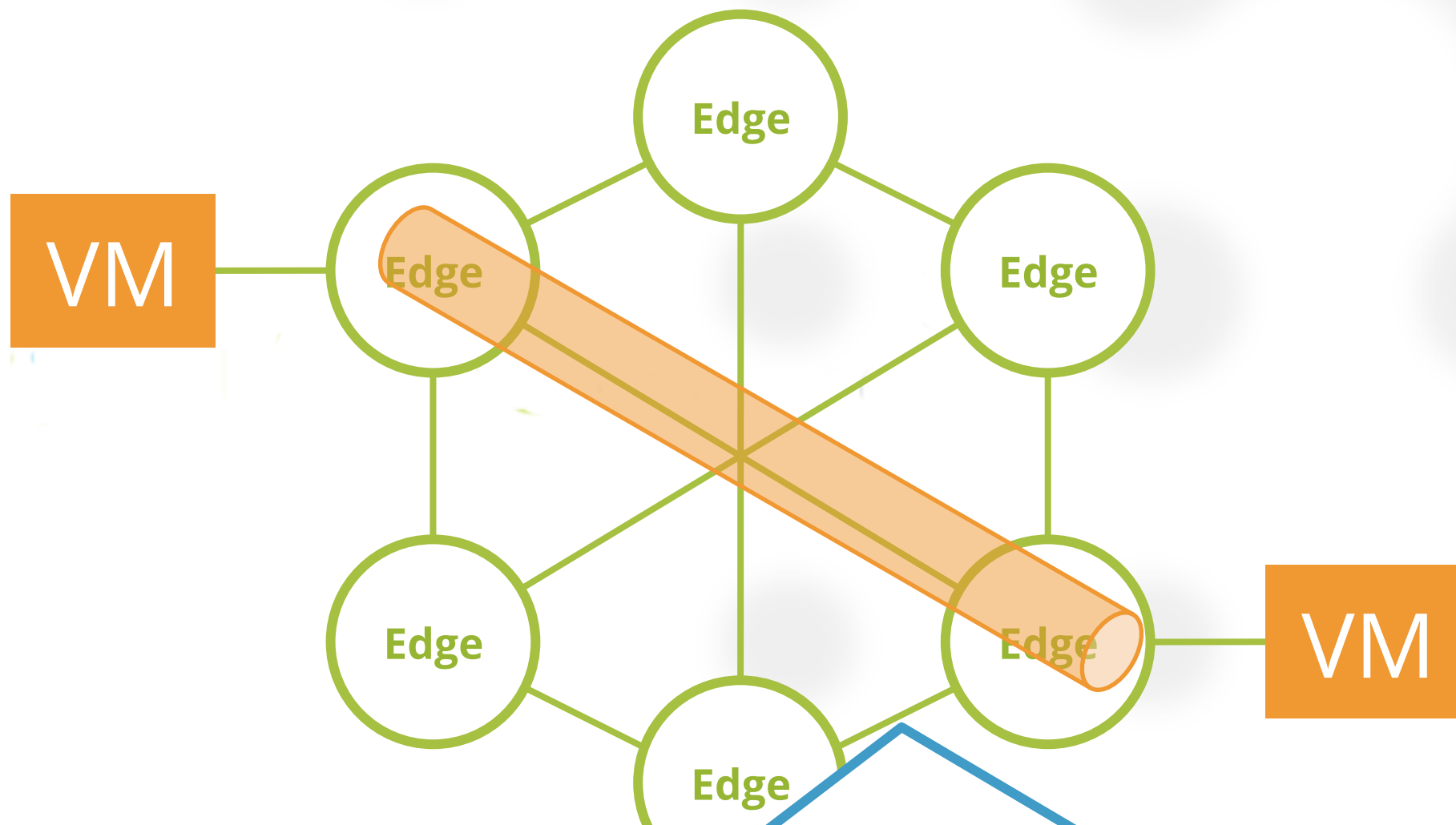
- Fabric extends to the compute host software switch?
 - State in each switch is proportional to the virtual network state
 - Need to update all switches in path when provisioning new virtual devices or updating them.
 - Not scalable, slow and non-atomic switch updates.



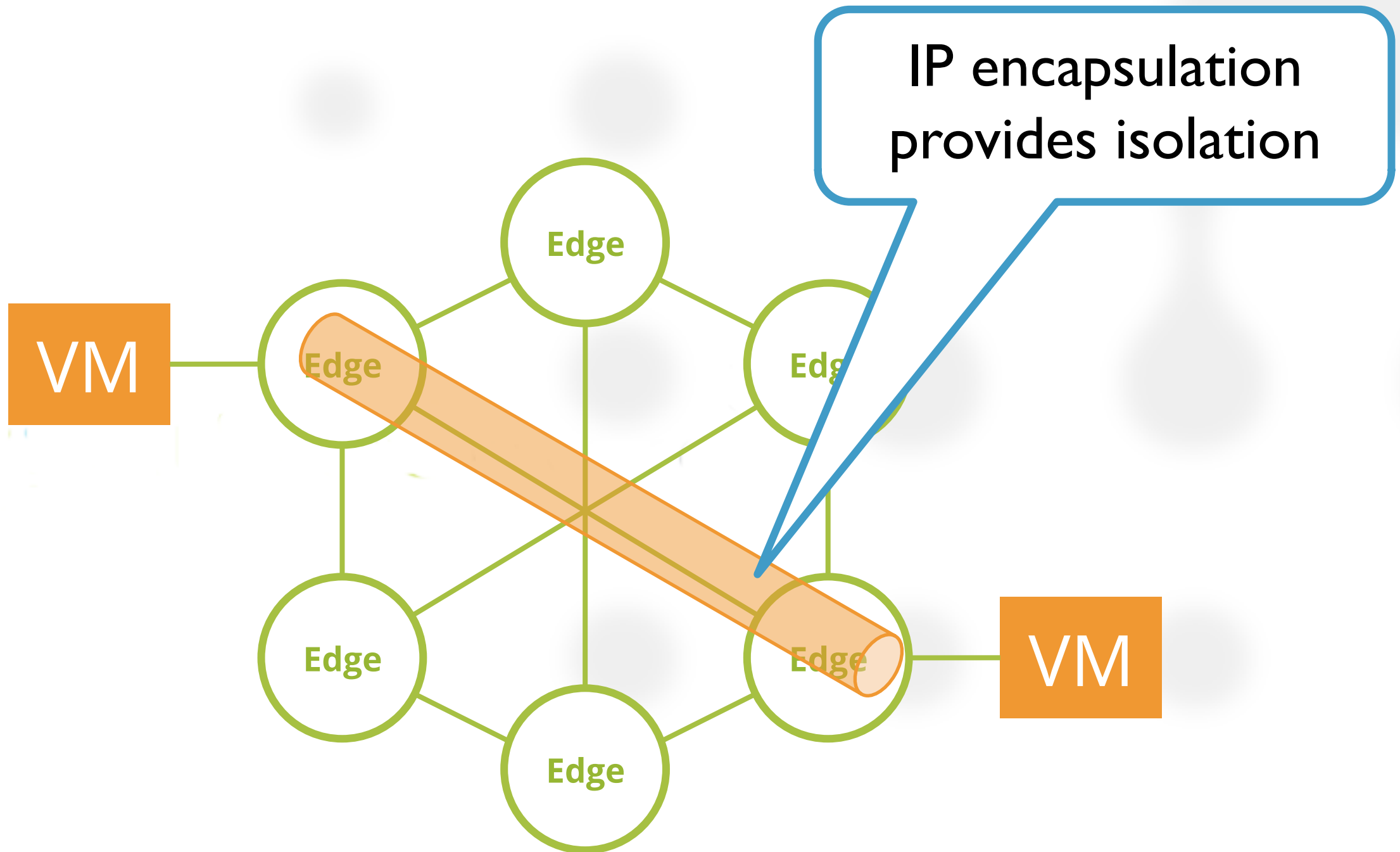
- Flow rules for VM flows (microflows)?
- Flow rules for virtual device simulation?

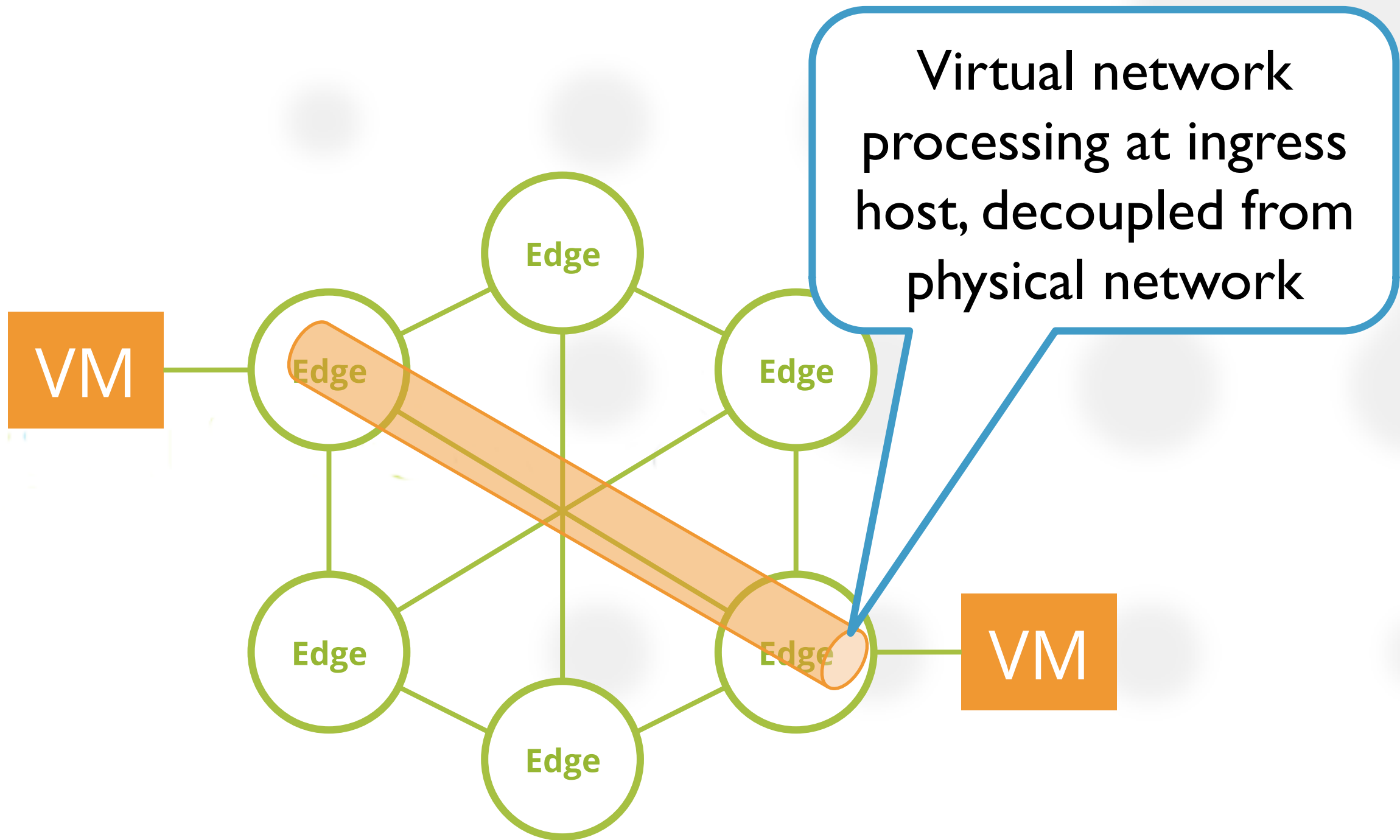
3

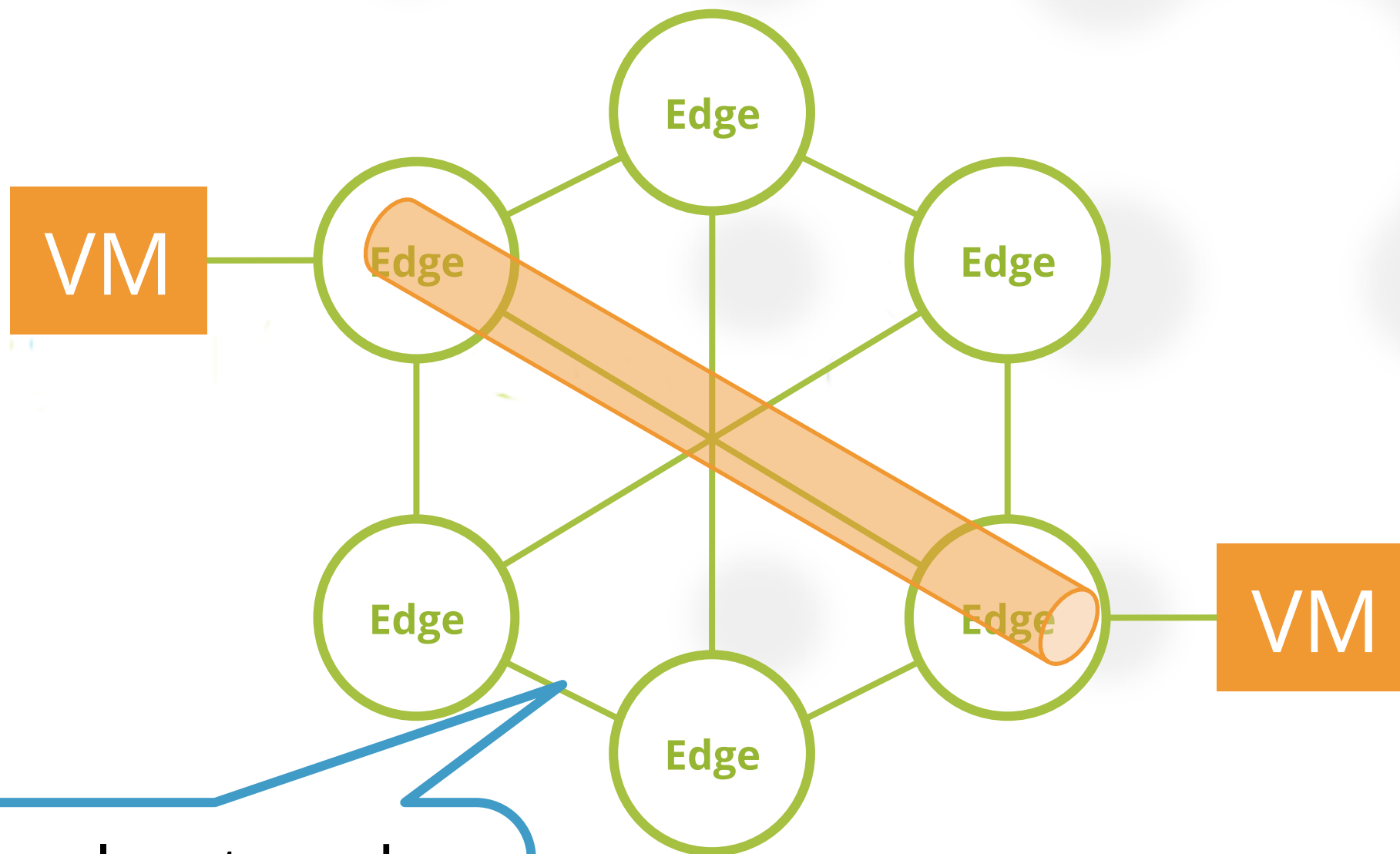
Edge-to-Edge Overlays



Use scalable IGP (iBGP, OSPF) to build multi-path underlay



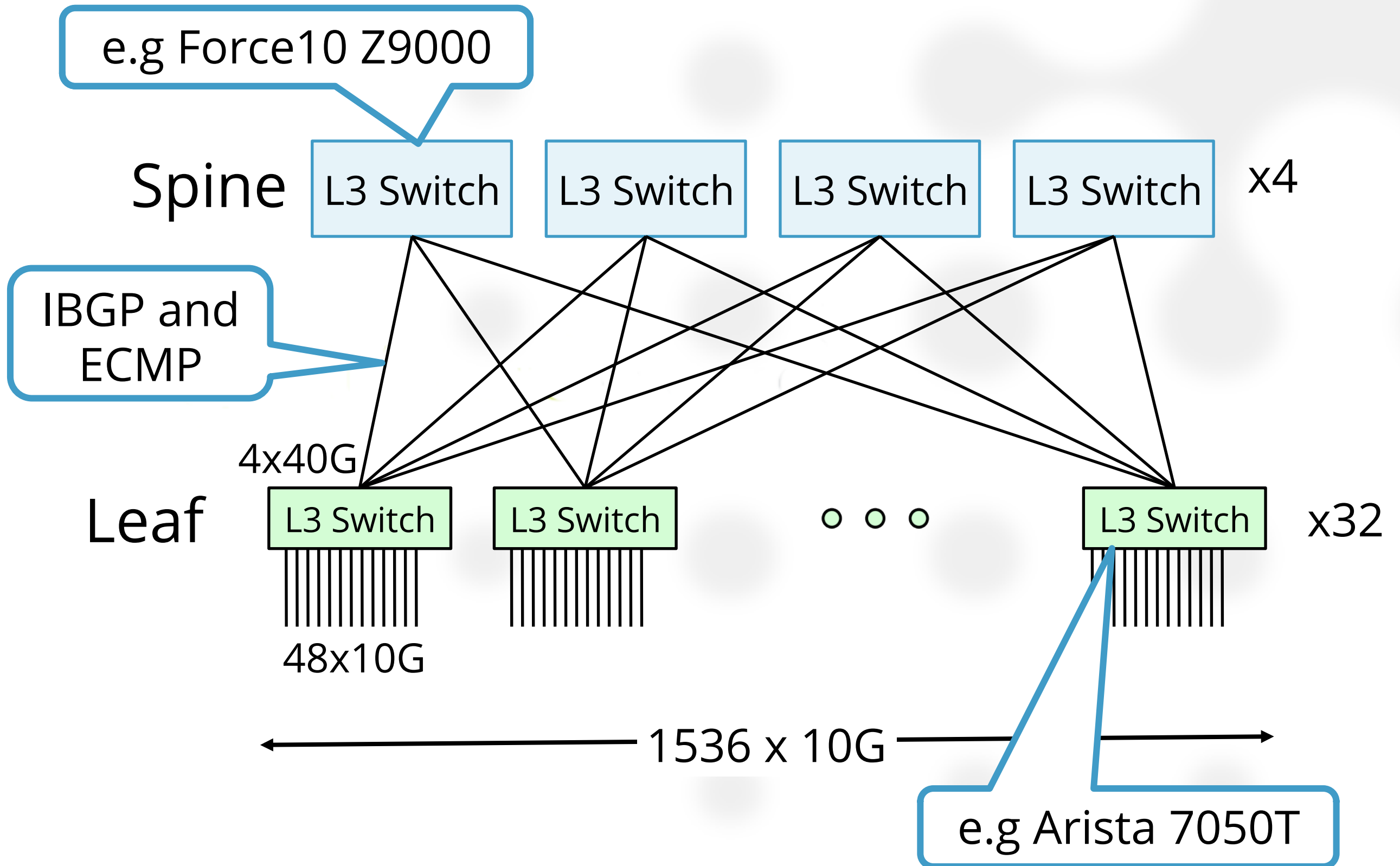




Virtual network
changes don't affect
underlay state

- Packet processing on x86 CPUs (at edge)
 - Intel DPDK facilitates packet processing
 - Number of cores in servers increasing fast
- Clos Networks (for underlay)
 - Spine and Leaf architecture with IP
 - Economical and high E-W bandwidth
- Merchant silicon (cheap IP switches)
 - Broadcom, Intel (Fulcrum Micro), Marvell
 - ODMs (Quanta, Accton) starting to sell directly
 - Switches are becoming just like Linux servers
- Optical intra-DC Networks

Spine and Leaf Network Architecture

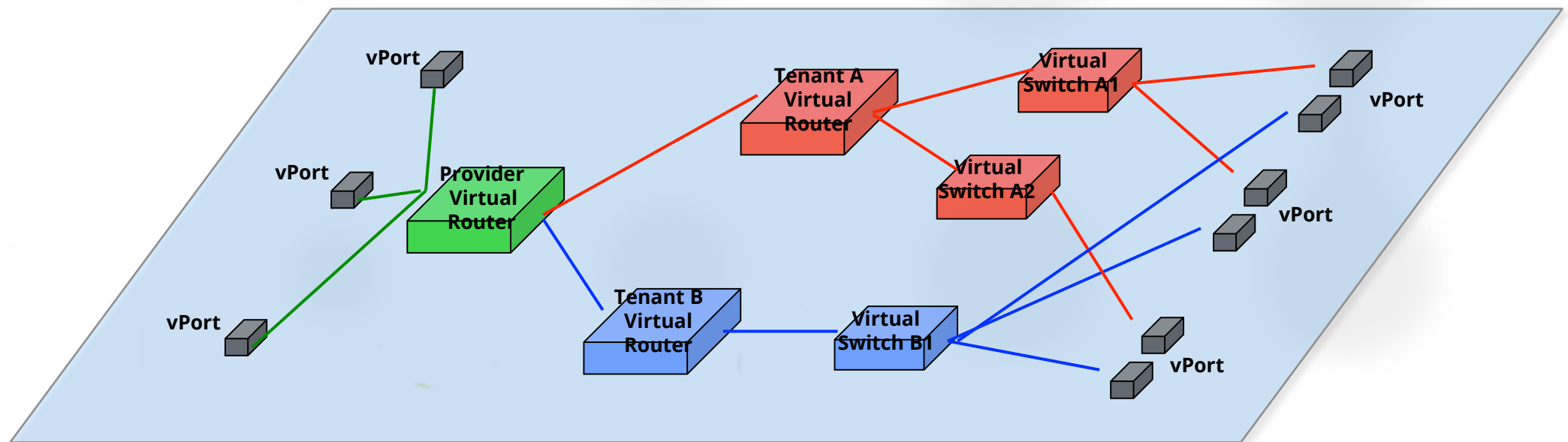


Overlays are the right approach!

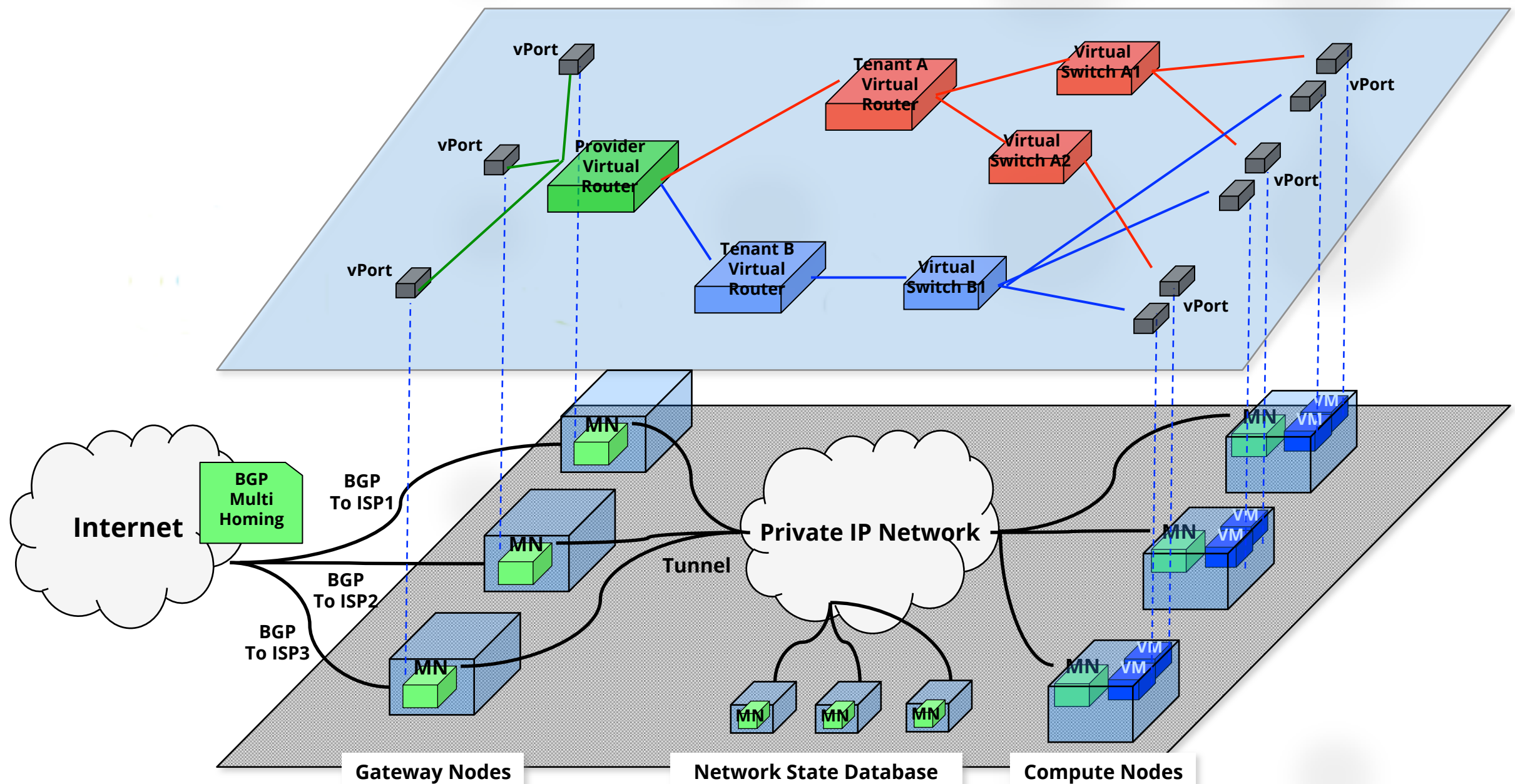
But not sufficient...

We still need a scalable control plane.

Logical Topology



Logical Topology



Physical Topology

MidoNet SDN Solution

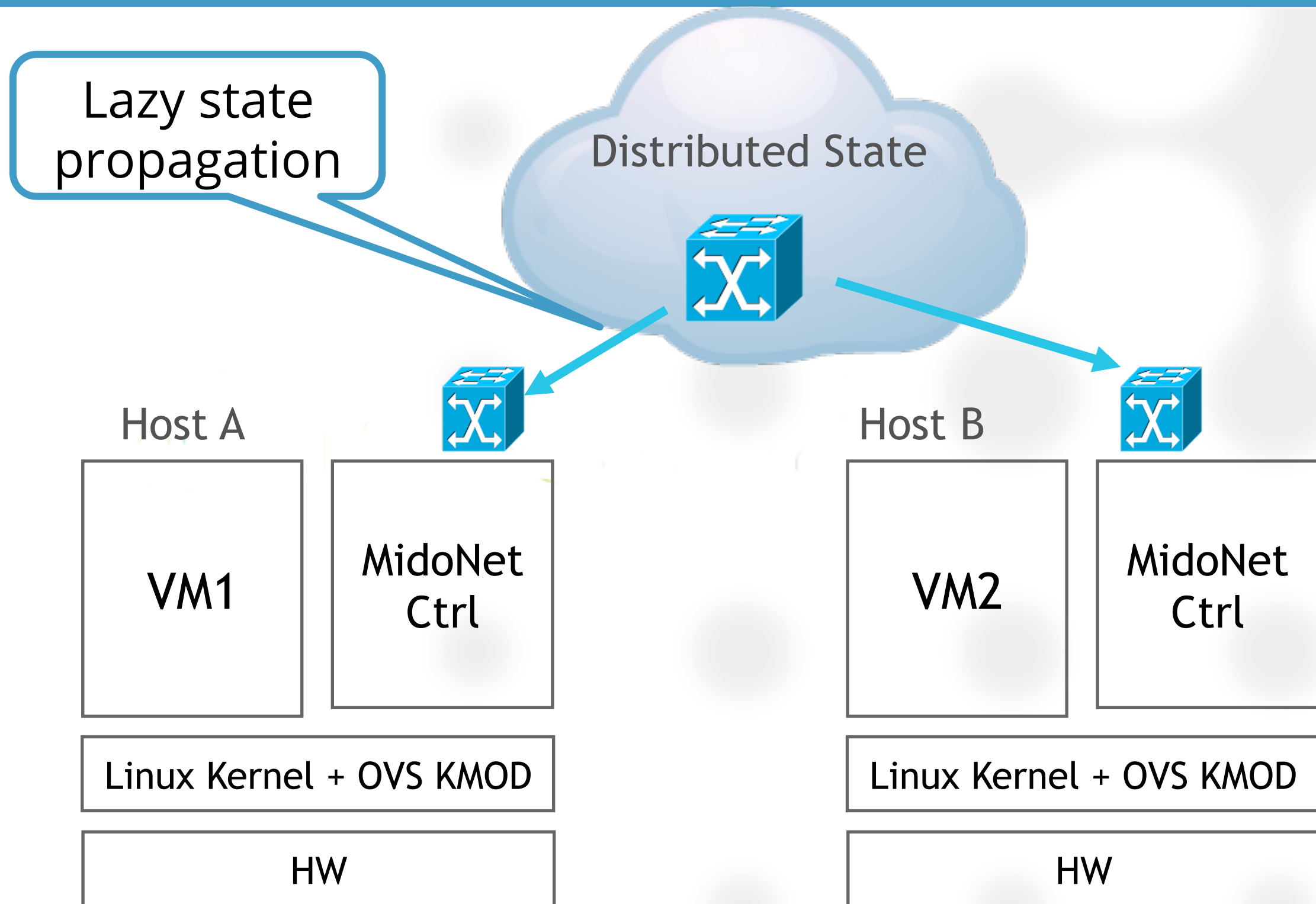
Distributed State



MidoNet REST API



MidoNet SDN Solution



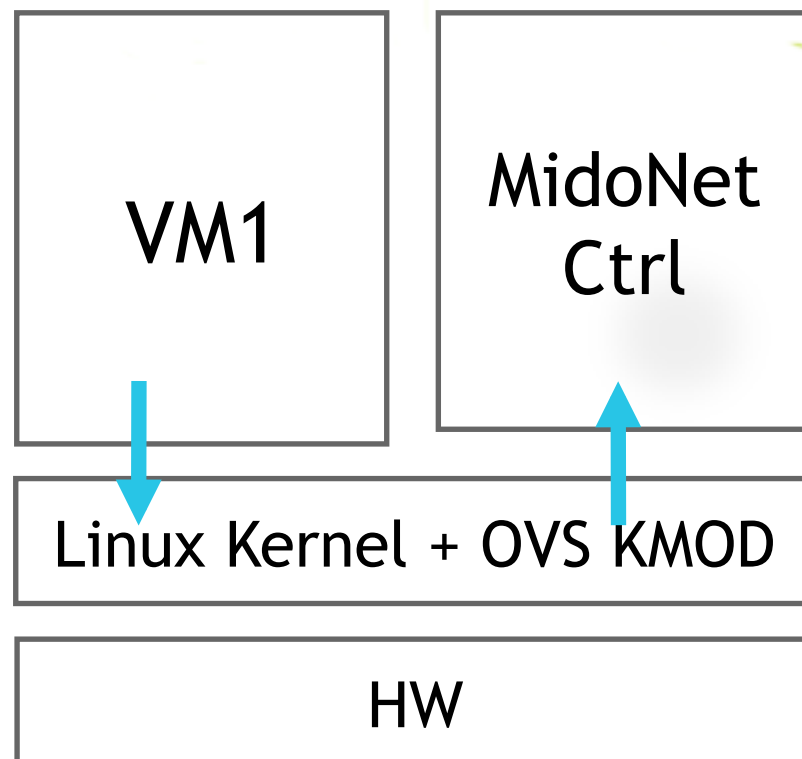
MidoNet SDN Solution

VM sends first packet; table miss; NetLink upcall to MidoNet

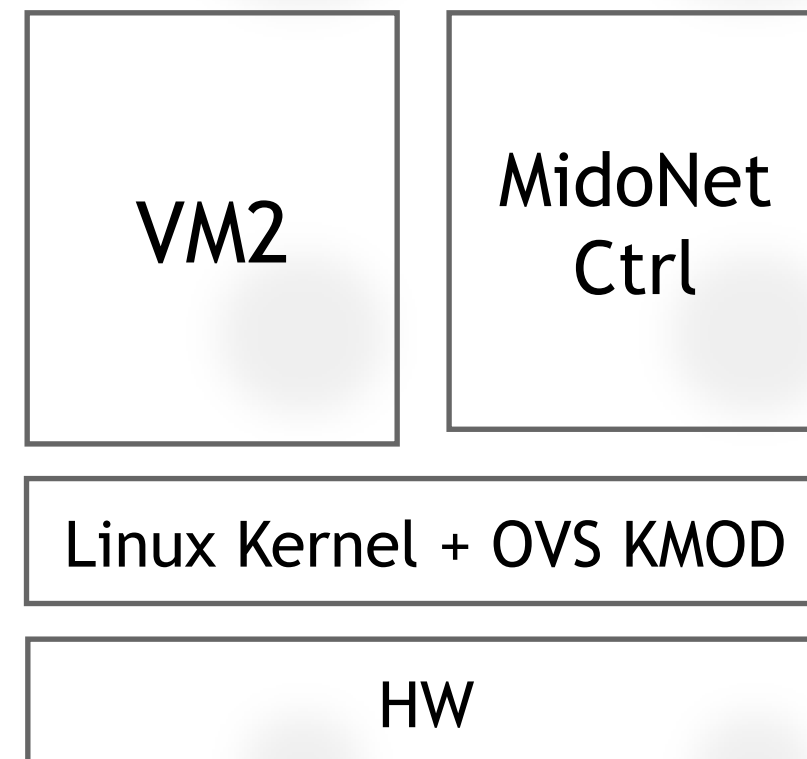
Distributed State



Host A



Host B



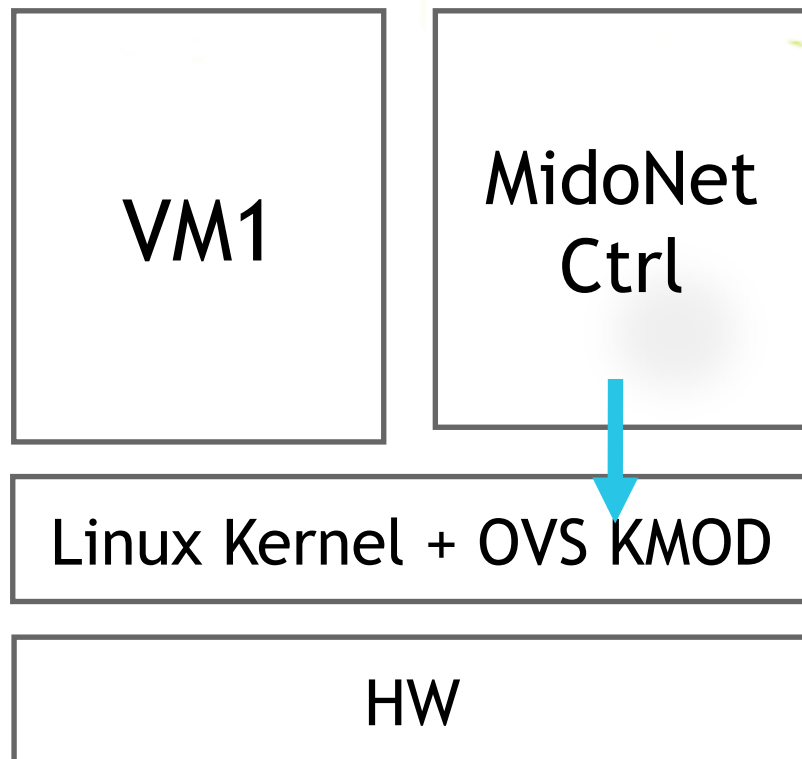
MidoNet SDN Solution

MidoNet agent locally processes packet (virtual layer simulation); installs local flow (drop/mod/fwd)

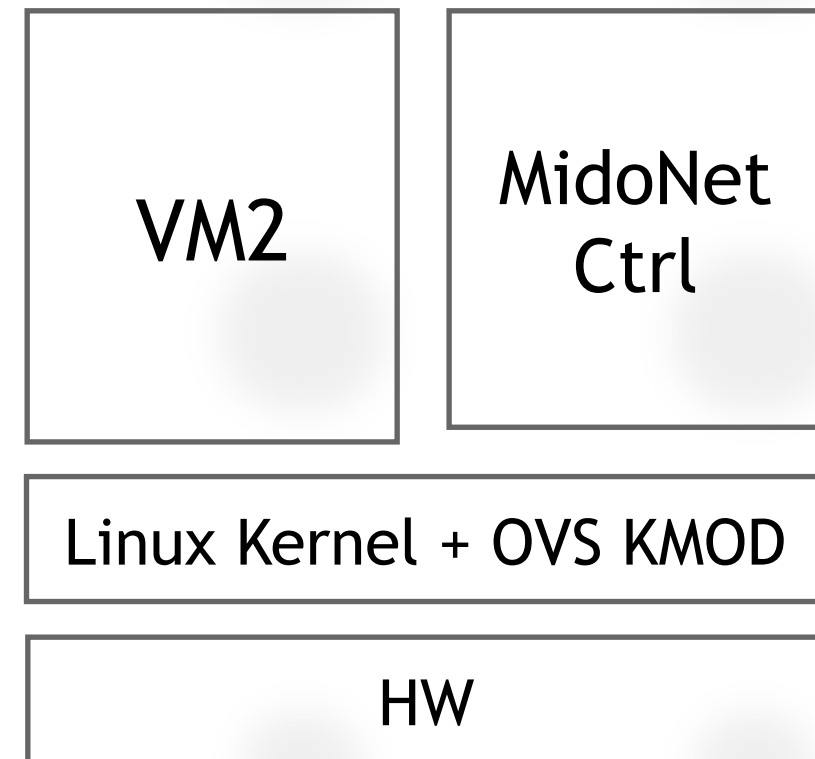
Distributed State



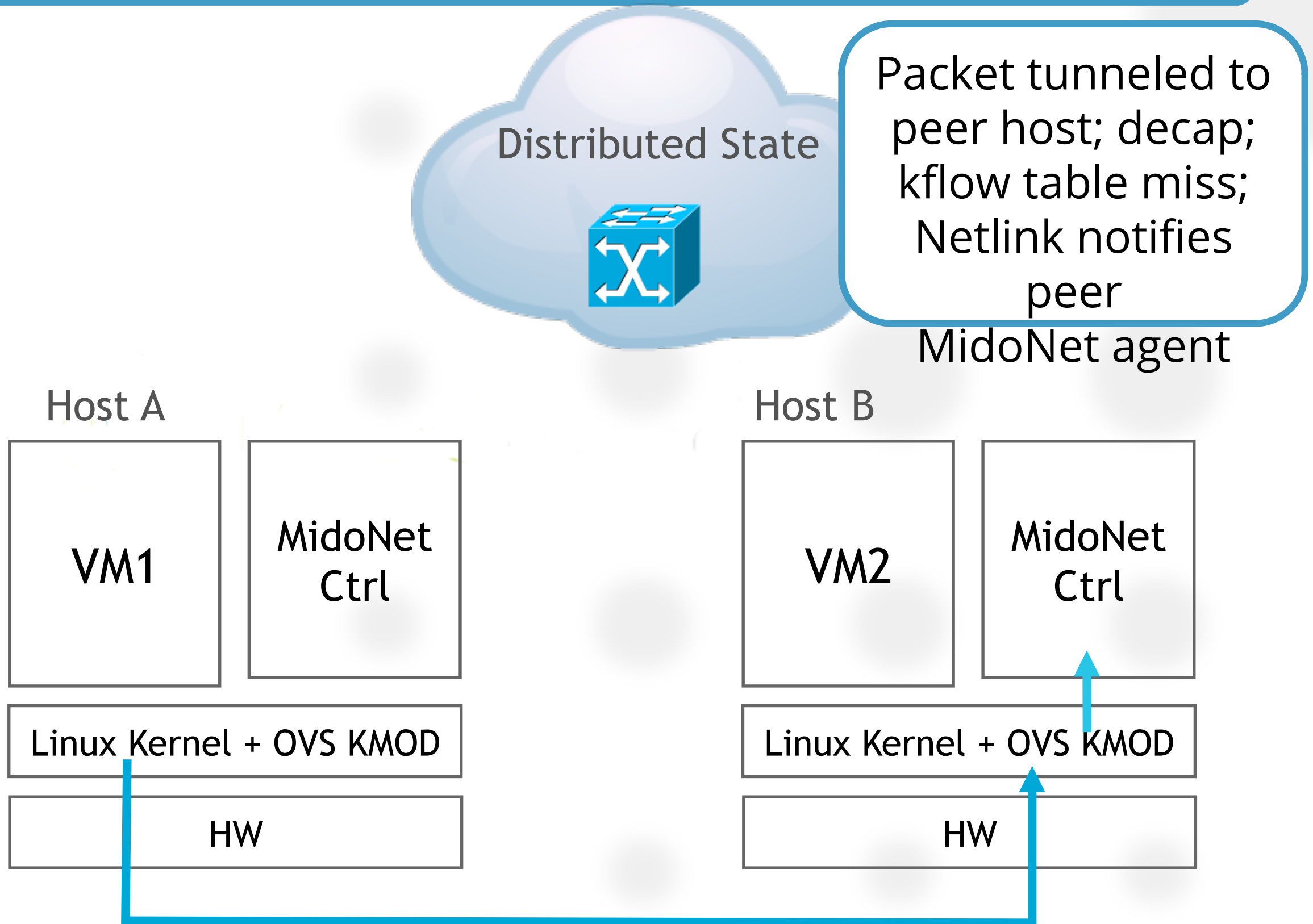
Host A



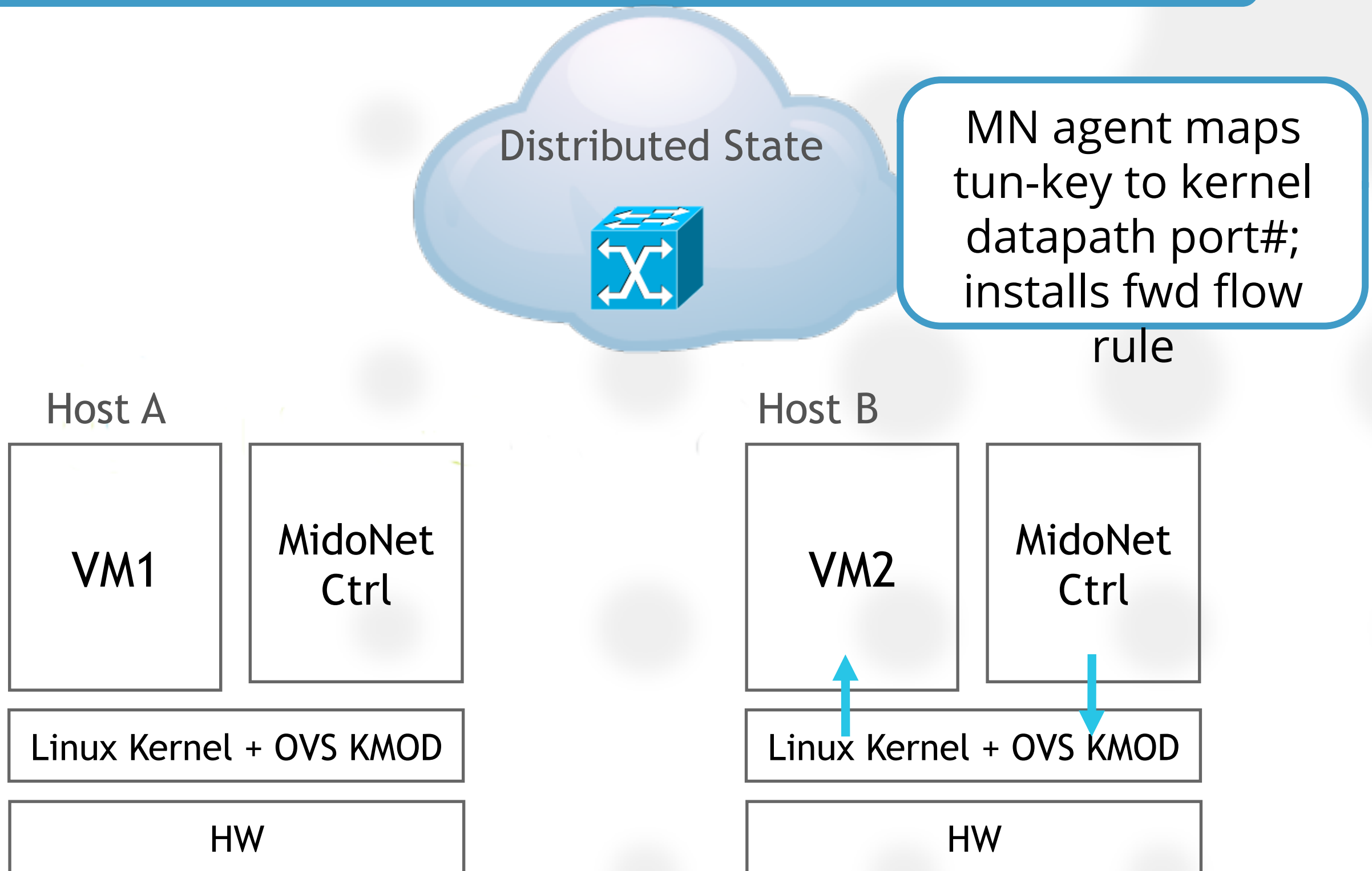
Host B



MidoNet SDN Solution



MidoNet SDN Solution



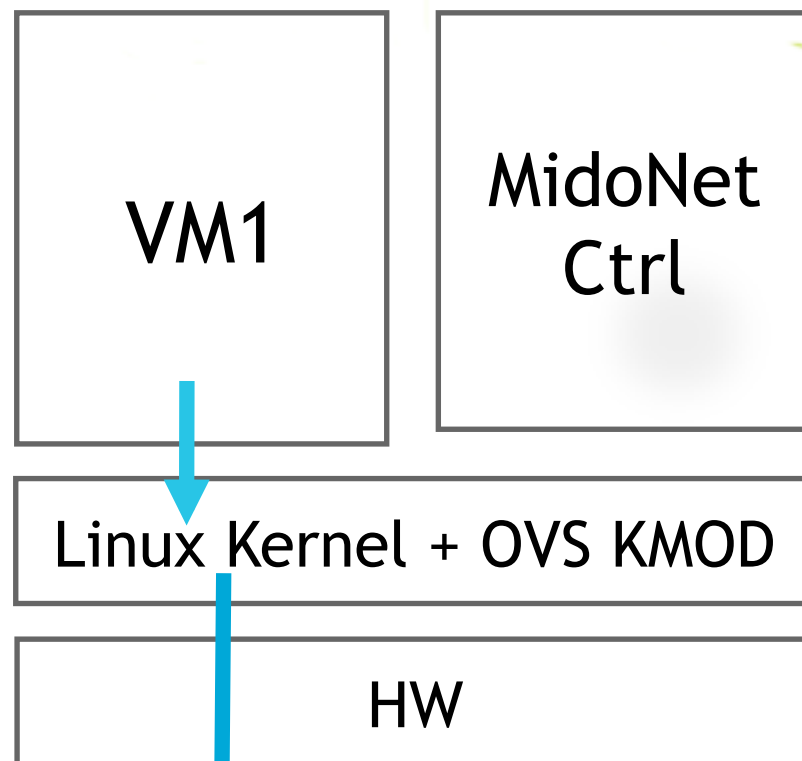
MidoNet SDN Solution

Subsequent packets
matched by flow rules
at both ingress and
egress hosts

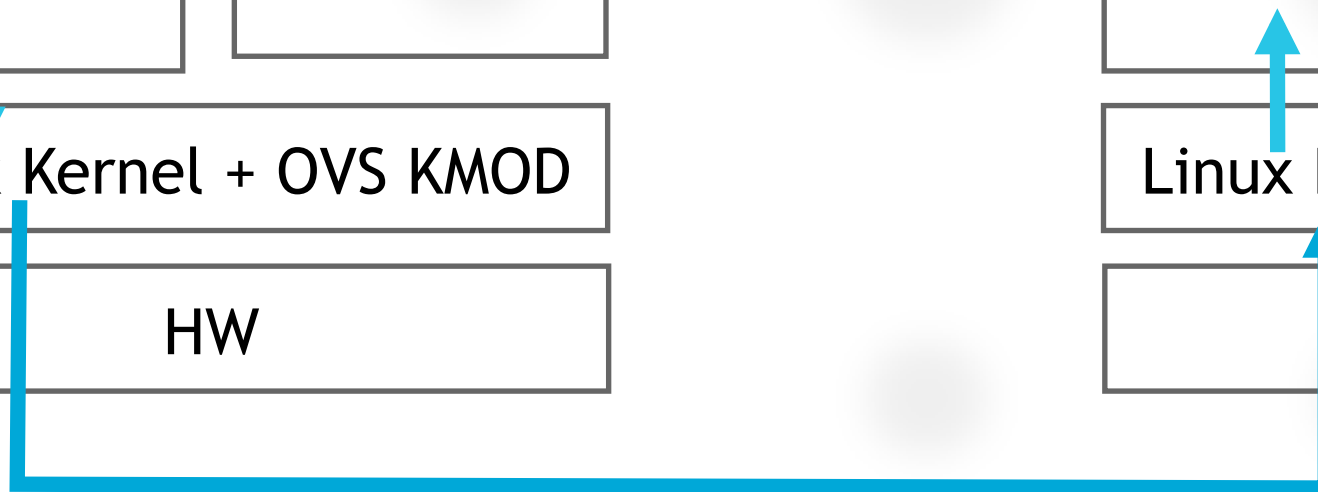
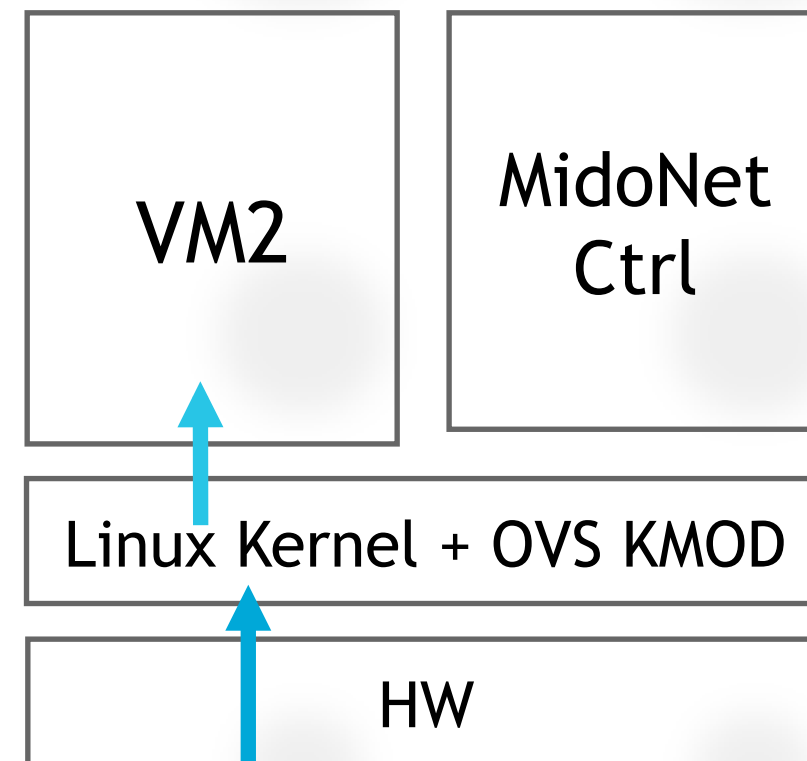
Distributed State



Host A

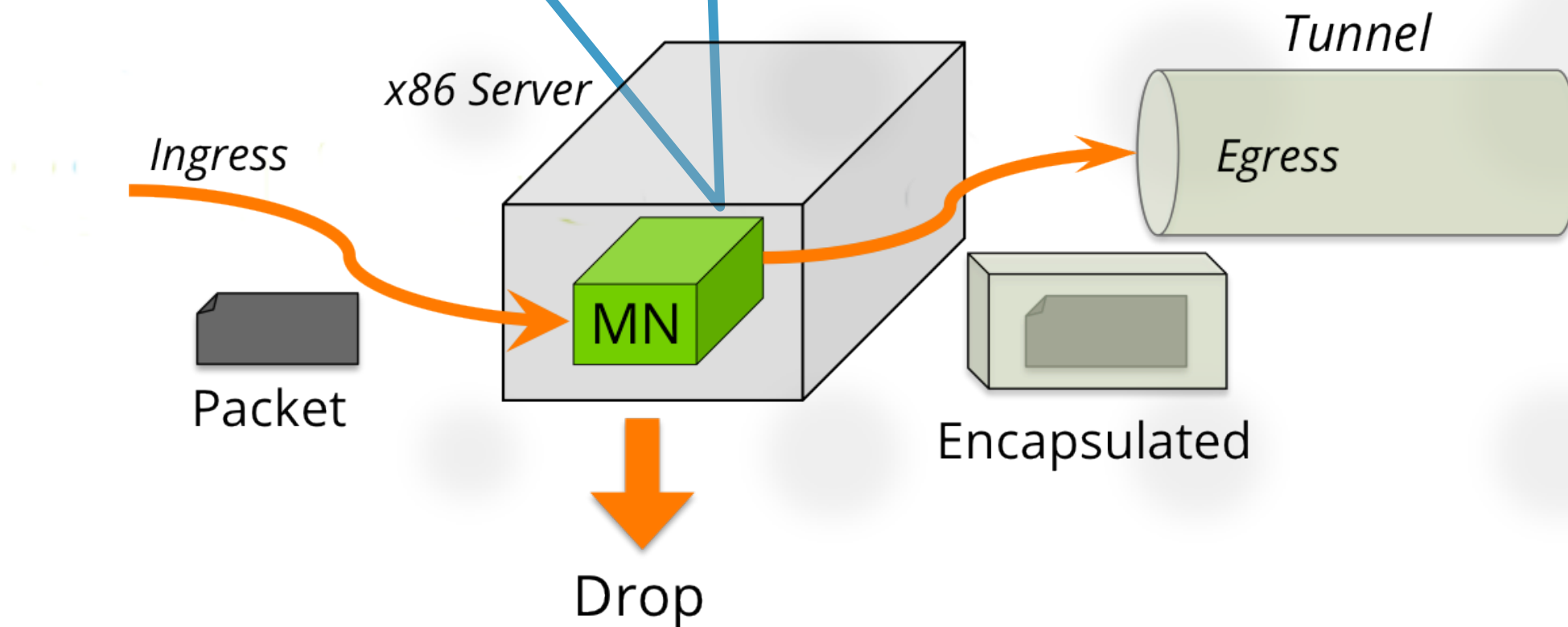


Host B

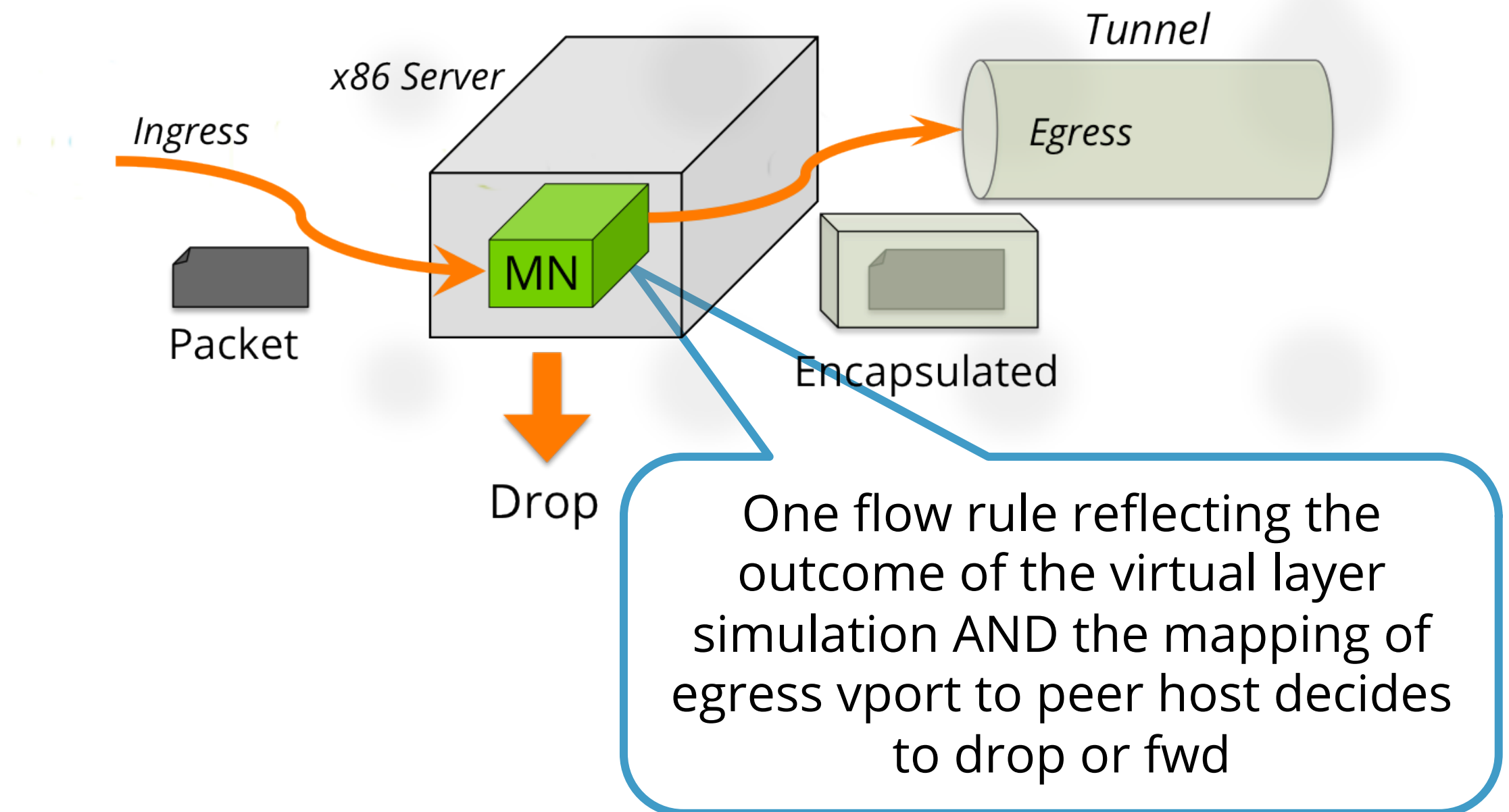


MidoNet SDN Solution

Packet from VM, VPN, or external BGP peer enters kernel datapath



MidoNet SDN Solution



- Distributed and scalable control plane
 - Handle all control packets at local MidoNet agent adjacent to VM
- Scalable and fault tolerant central database
 - Stores virtual network configuration
 - Dynamic network state
 - ✧ MAC learning, ARP cache, etc
 - Cached at edges on demand
- All packet modifications at ingress
 - One virtual hop
 - ✧ No travel through middle boxes
 - Drop at ingress

- Scalable edge gateway interface to external networks
 - Multihomed BGP to ISP
- REST API and GUI
- Integration with popular open source cloud stacks
 - OpenStack
 - Removes SPOF of network node
 - Scalable and fault tolerant NAT for floating IP
 - Implements security groups efficiently
 - CloudStack and Eucalyptus

Deep OpenStack Integration

- Quantum Plugin
 - L2 isolation, of course
- Also...
 - L3 isolation (without VM / appliance)
 - Security groups (stateful firewall)
 - Floating IP (NAT)
 - Load balancing (L4)

Future Directions

- Scalable L7 virtual appliances
- MPLS VPN termination
 - Interconnect with carrier backbones
- multiple data center federation
 - Virtual L2 between sites
- LISP
 - Global IP mobility between sites

Conclusions

- IaaS clouds require new networking
- Edge to edge overlays are the right approach
- Servers are good at packet processing
 - Can use them for edge gateways
- Multipath IP network fabric is cheap and easy to build

Questions?

info@midokura.com



Hiring!!!

Midokura is hiring!
in TYO, SFO, and BCN

careers@midokura.com

