Overlay-based virtual networking
vs
OpenFlow-controlled switch fabrics
in IaaS Clouds

Pino de Candia, pino@midokura.com
Software dev/mgr, Midokura BCN
oVirt Workshop - November 8, 2012

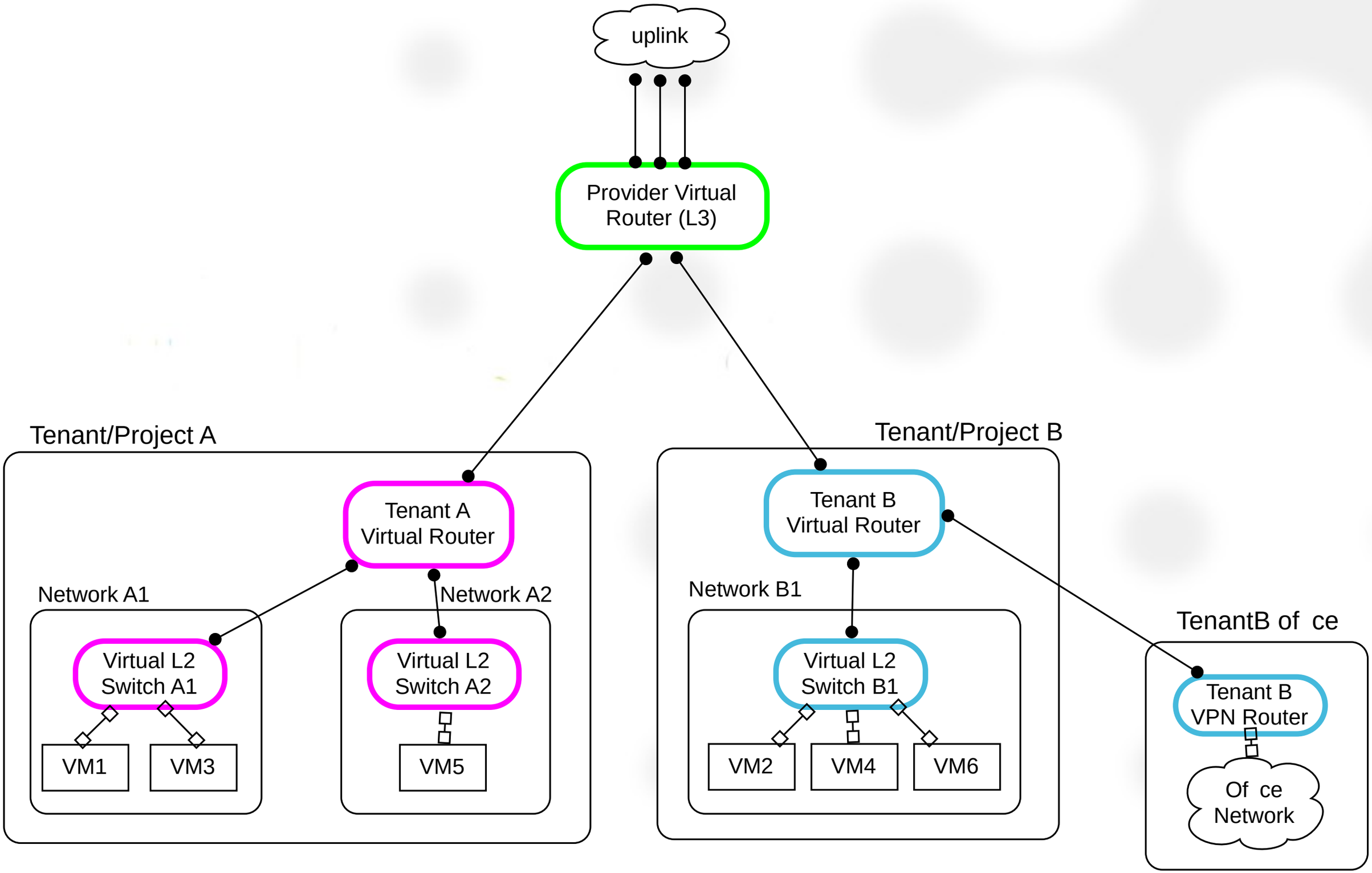
midokura

Overview

- 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

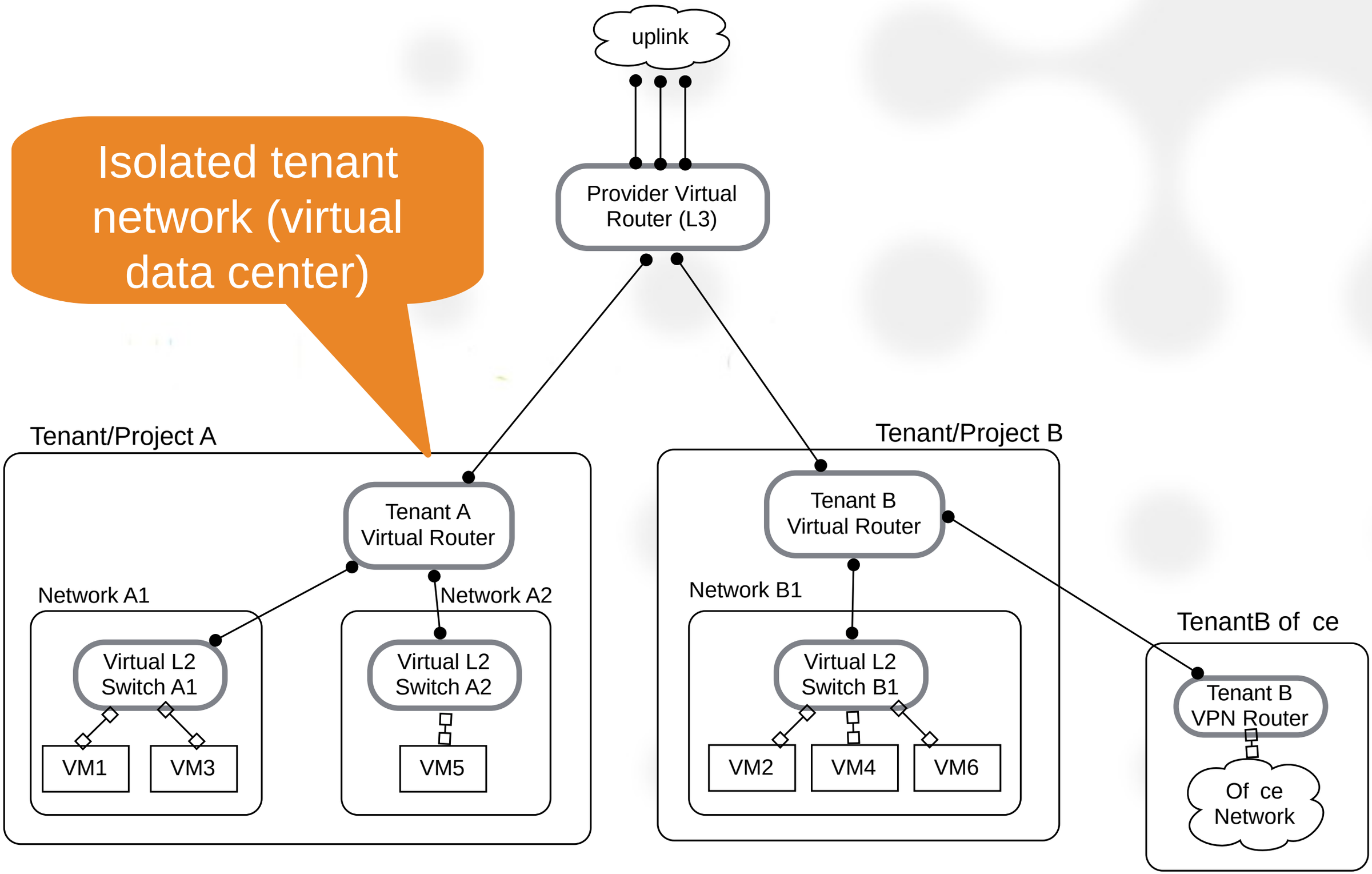
Copyright © 2012 Midokura All rights reserved

Requirements



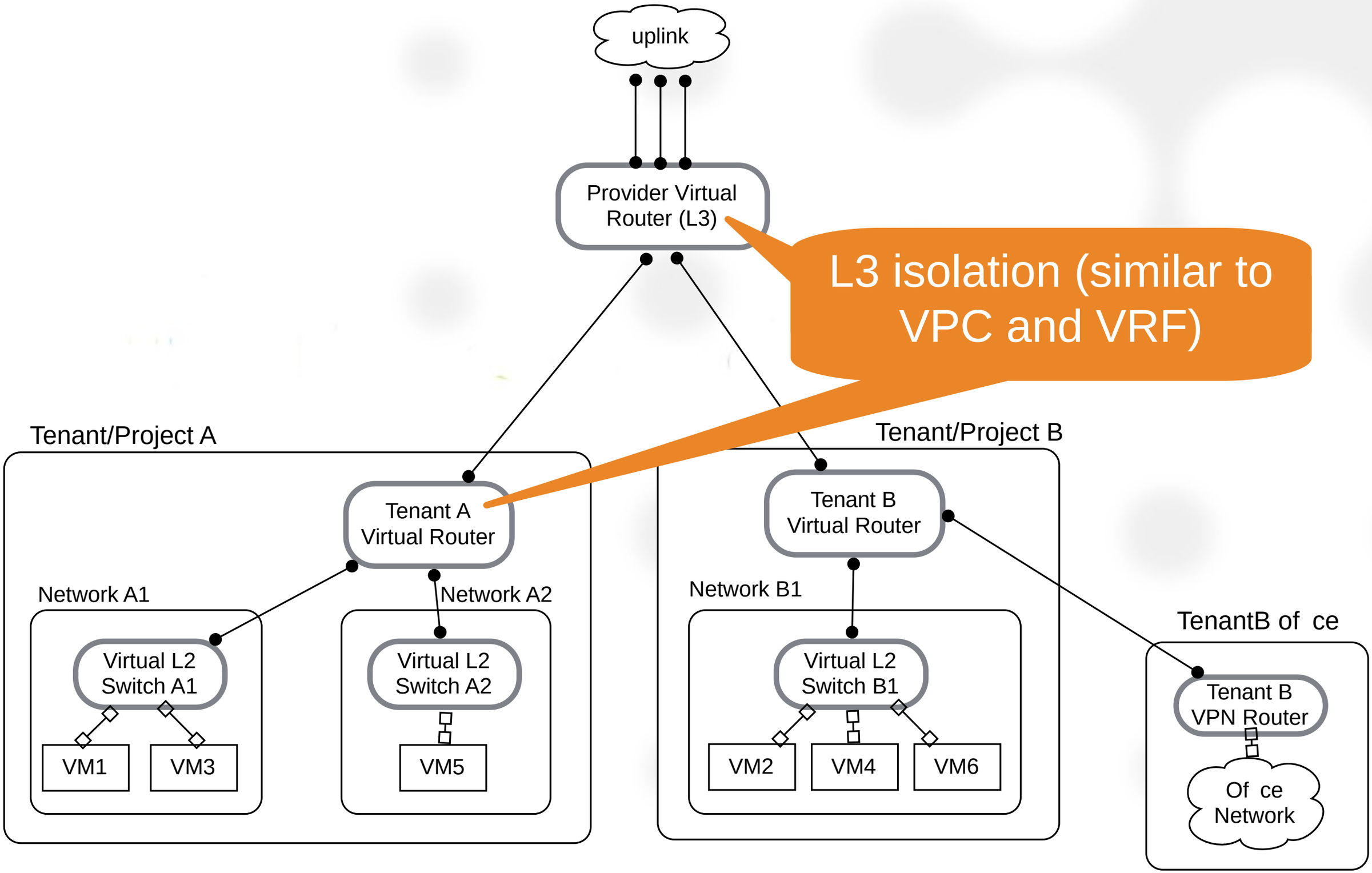
Copyright © 2012 Midokura All rights reserved

Requirements



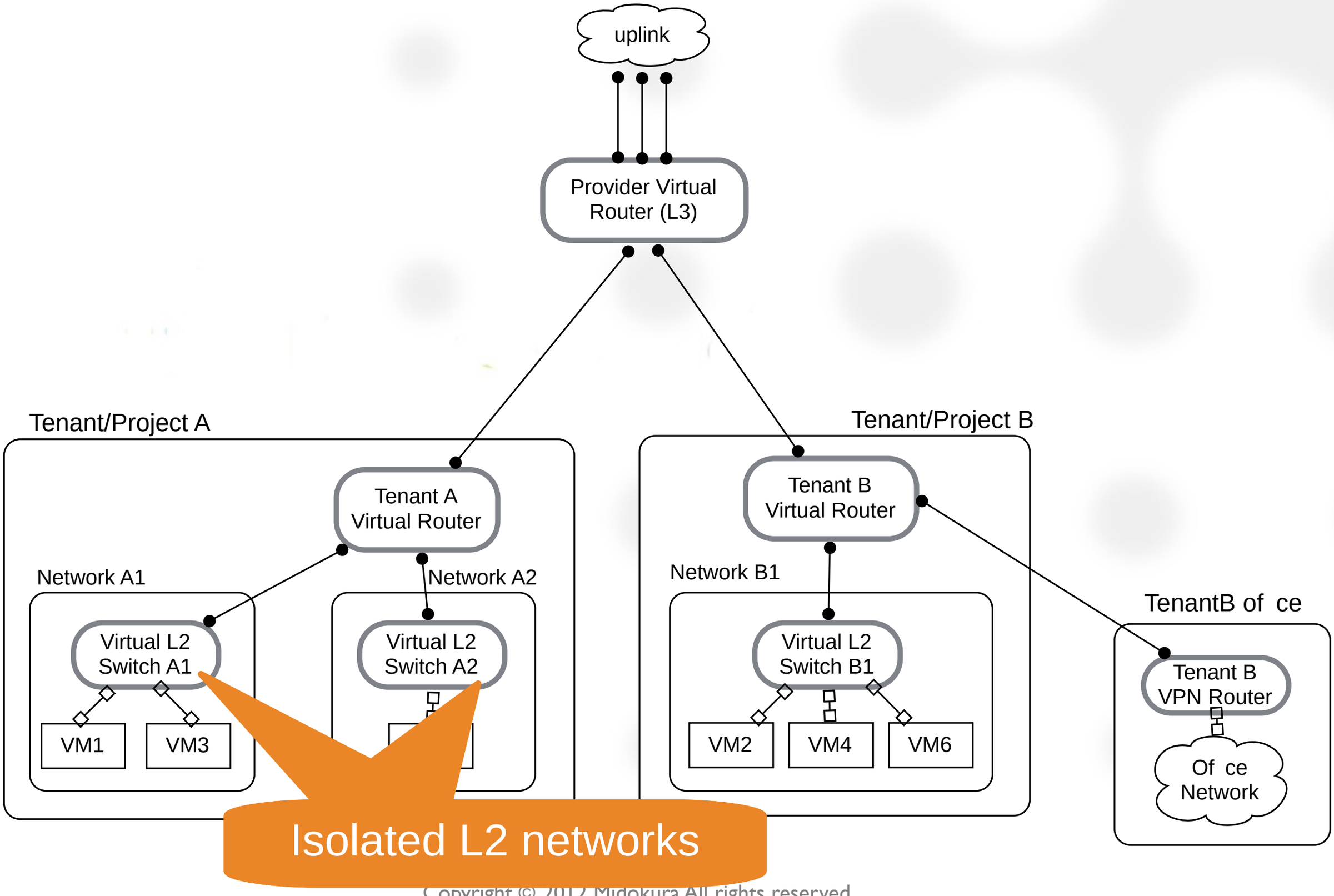
Copyright © 2012 Midokura All rights reserved

Requirements



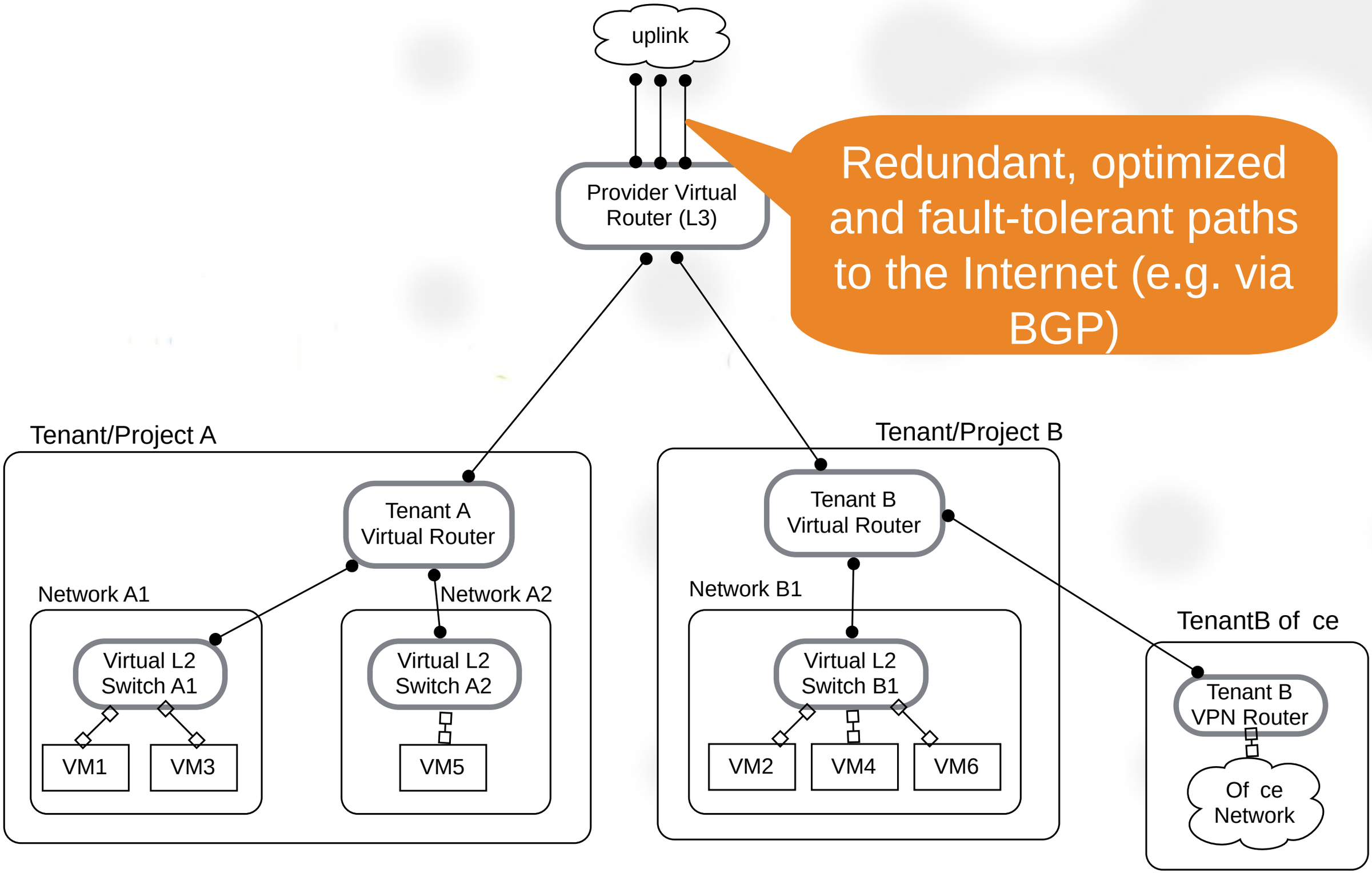
Copyright © 2012 Midokura All rights reserved

Requirements



Copyright © 2012 Midokura All rights reserved

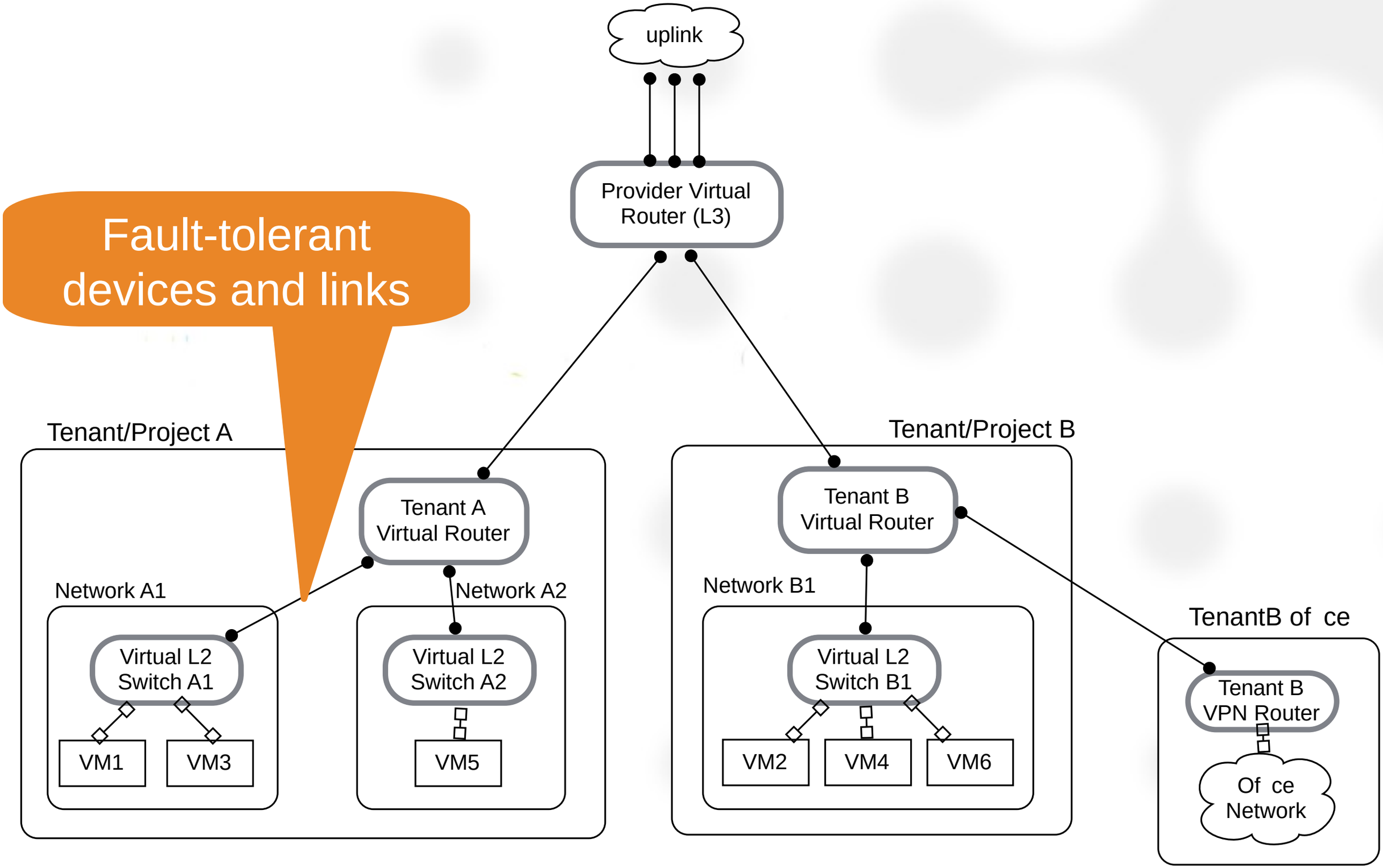
Requirements



Copyright © 2012 Midokura All rights reserved

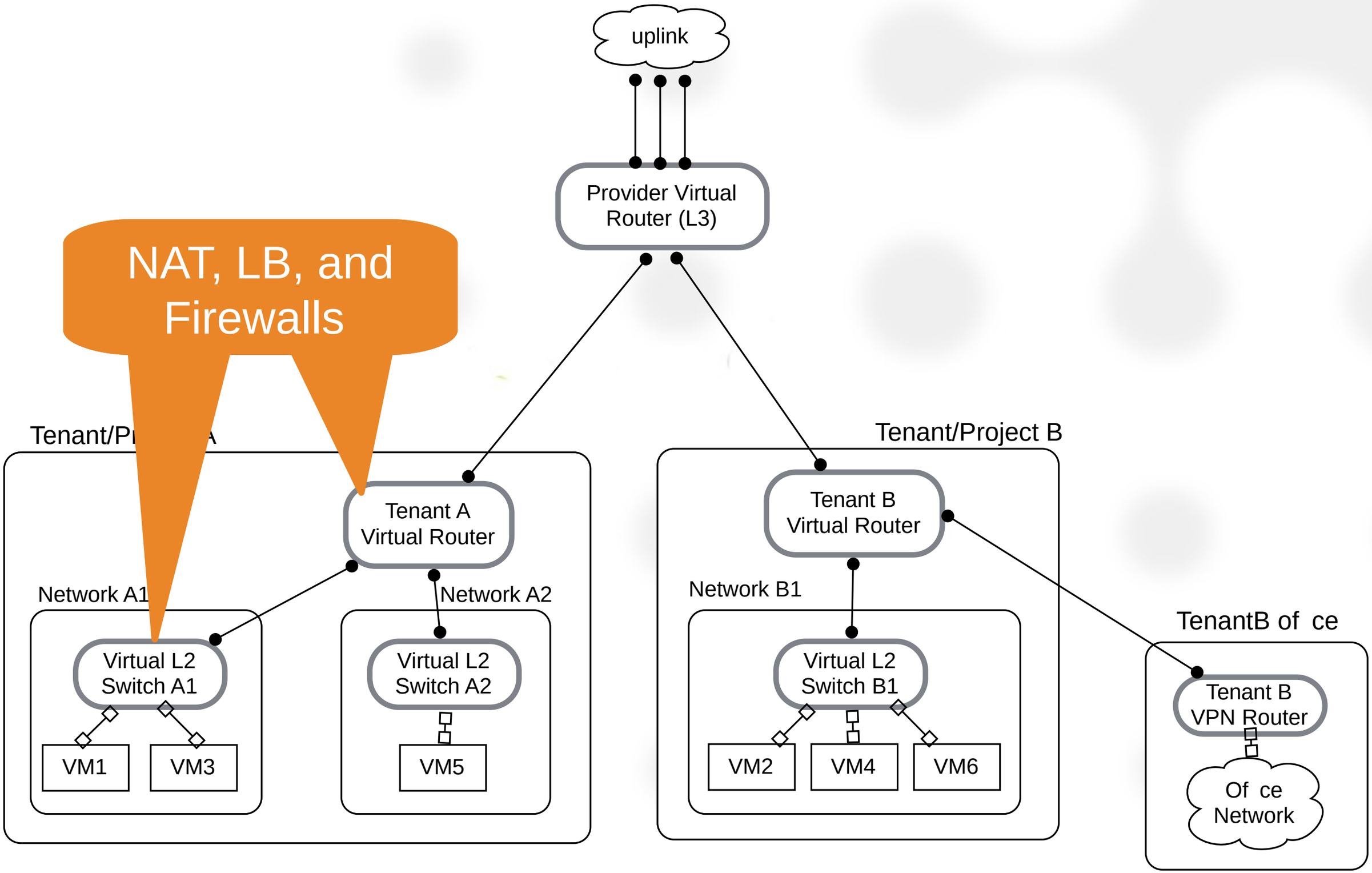
Requirements

Fault-tolerant devices and links



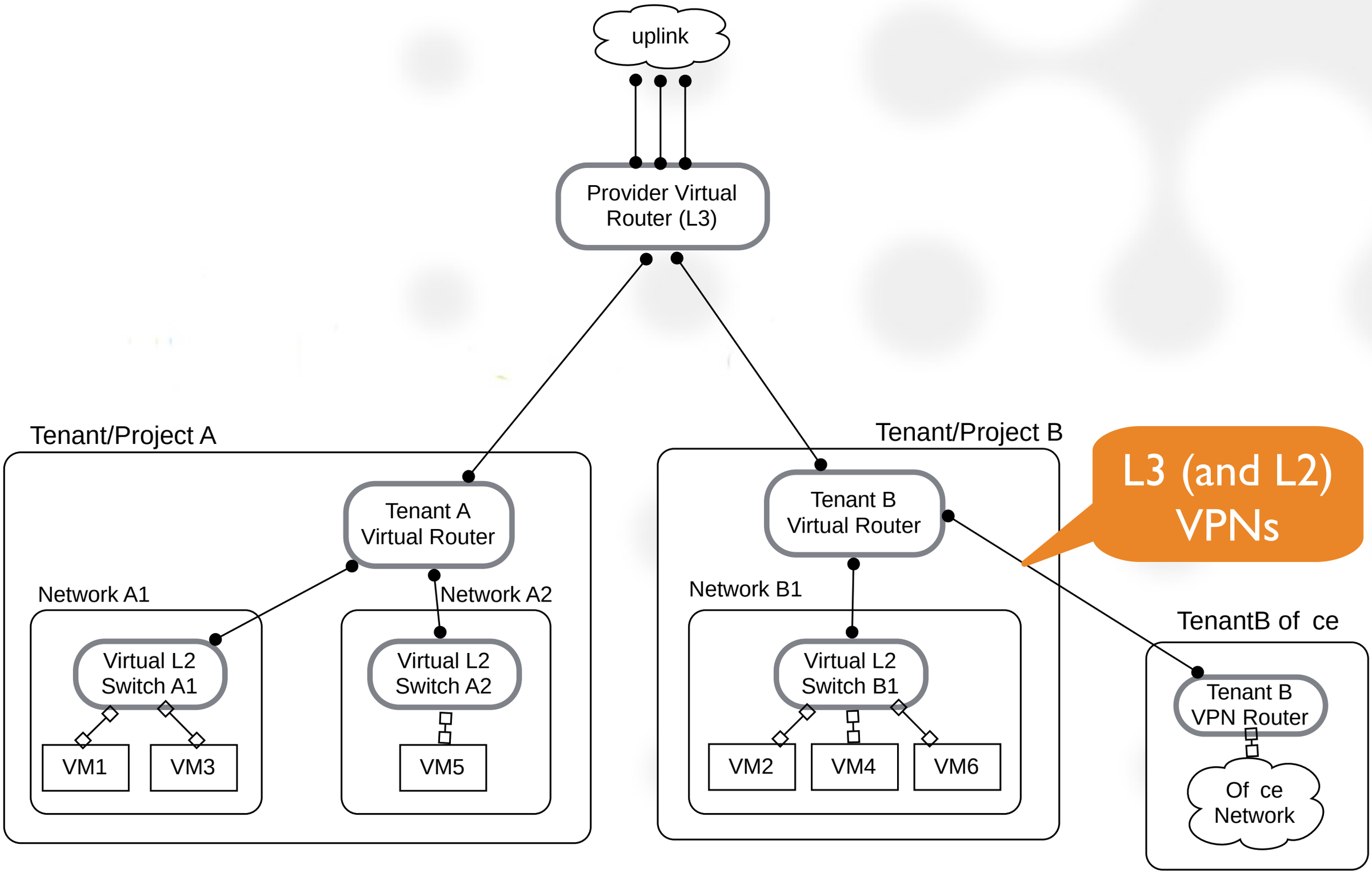
Copyright © 2012 Midokura All rights reserved

Requirements



Copyright © 2012 Midokura All rights reserved

Requirements

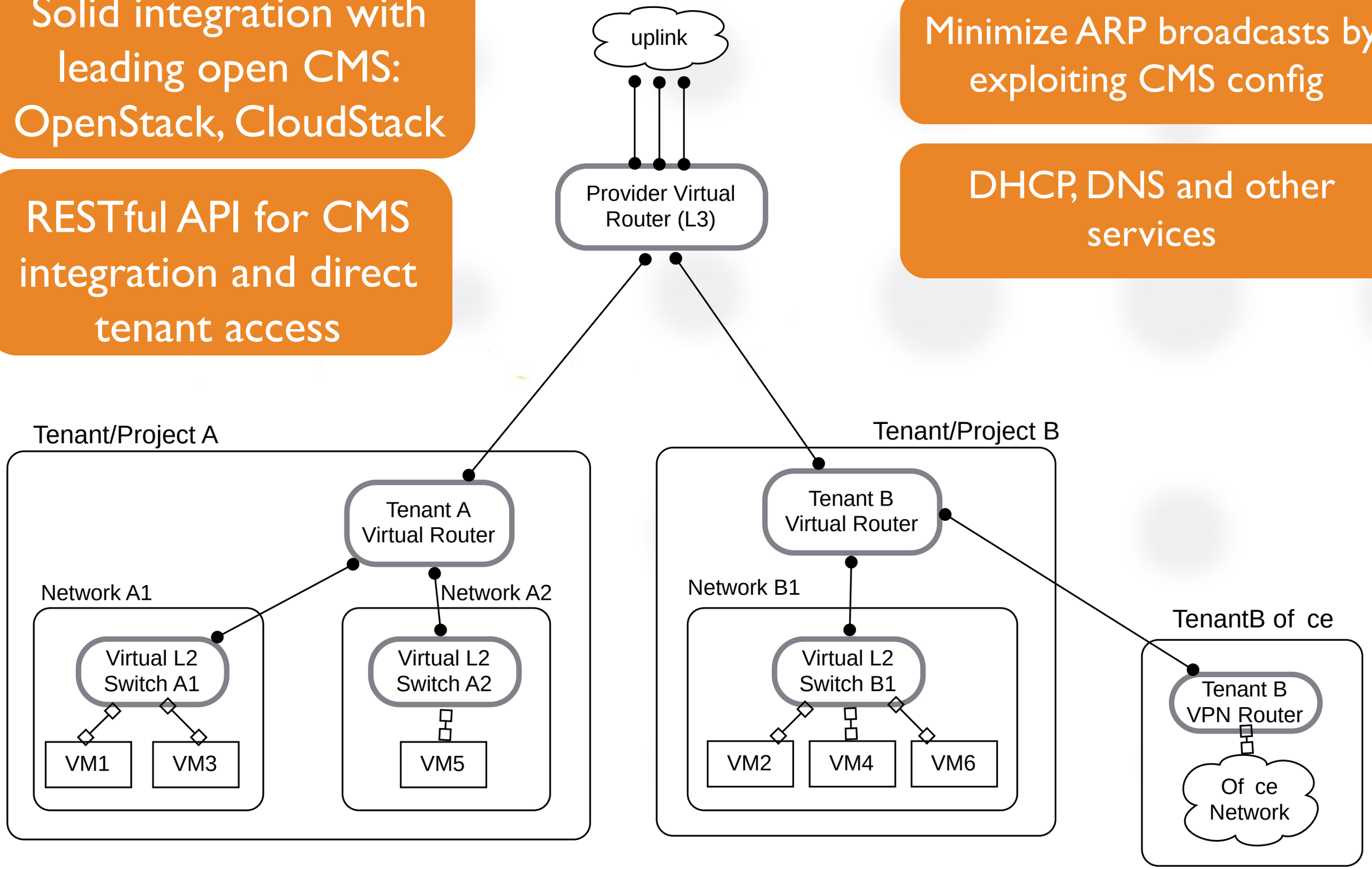


Copyright © 2012 Midokura All rights reserved

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

Copyright © 2012 Midokura All rights reserved

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

Copyright © 2012 Midokura All rights reserved

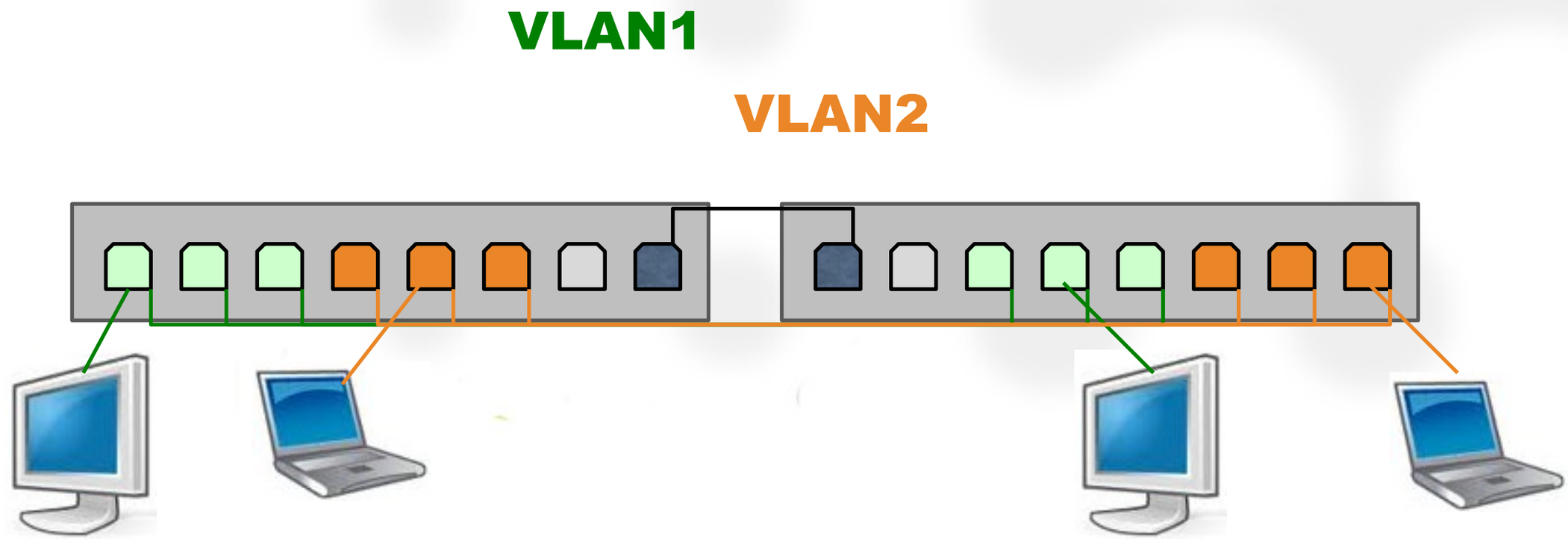
How to build it?

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

Copyright © 2012 Midokura All rights reserved

1 Virtualized physical devices

VLAN

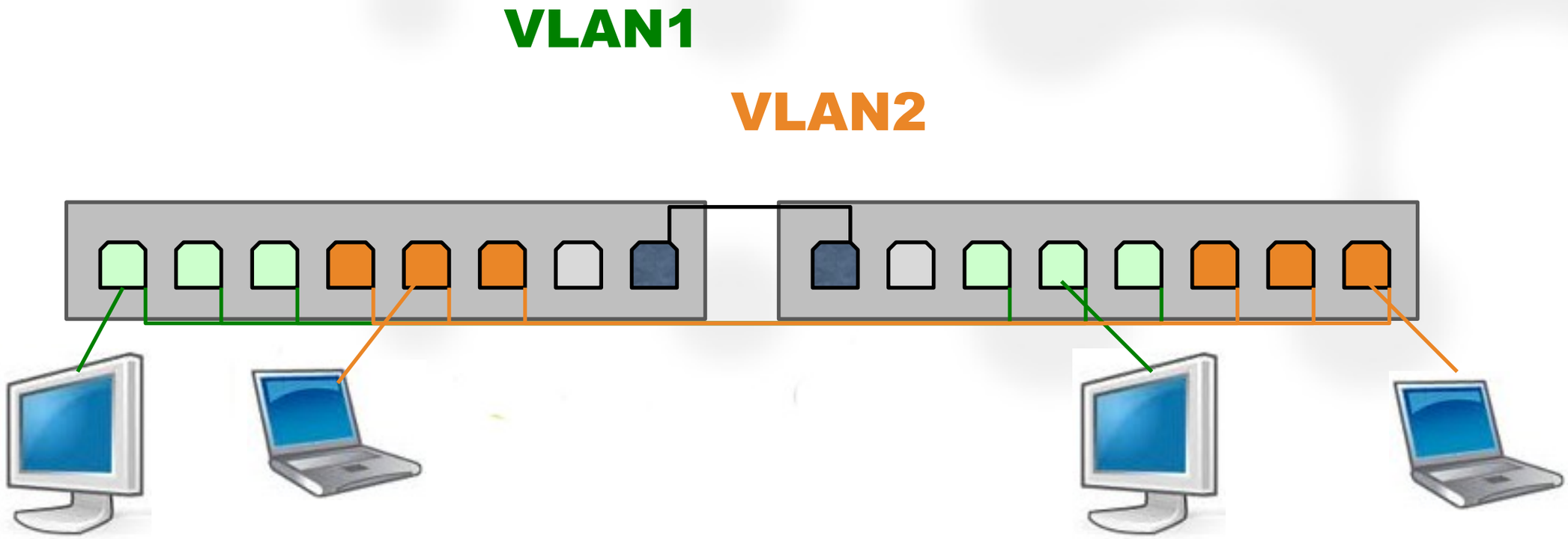


- 4096 limit on number of unique tags
- Large spanning trees terminating on many hosts
- High churn in switch control planes due to MAC learning
- Need MLAG for L2 multi-path (vendor specific)

Copyright © 2012 Midokura All rights reserved

1 Virtualized physical devices

VLAN (more)

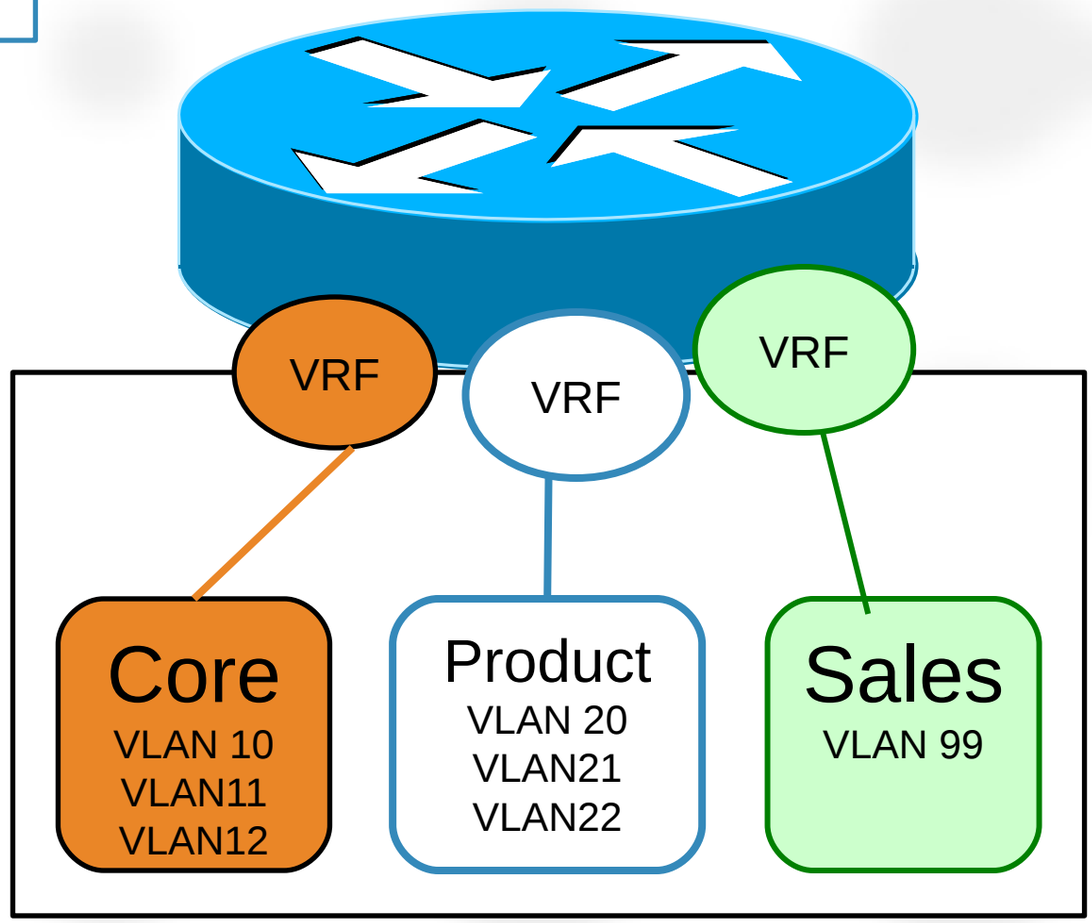


- L2 isolation
- What about L3 and Internet access?
- Use VRF or virtual appliances? Fault-tolerance?

Copyright © 2012 Midokura All rights reserved

1 Virtualized physical devices

VRF

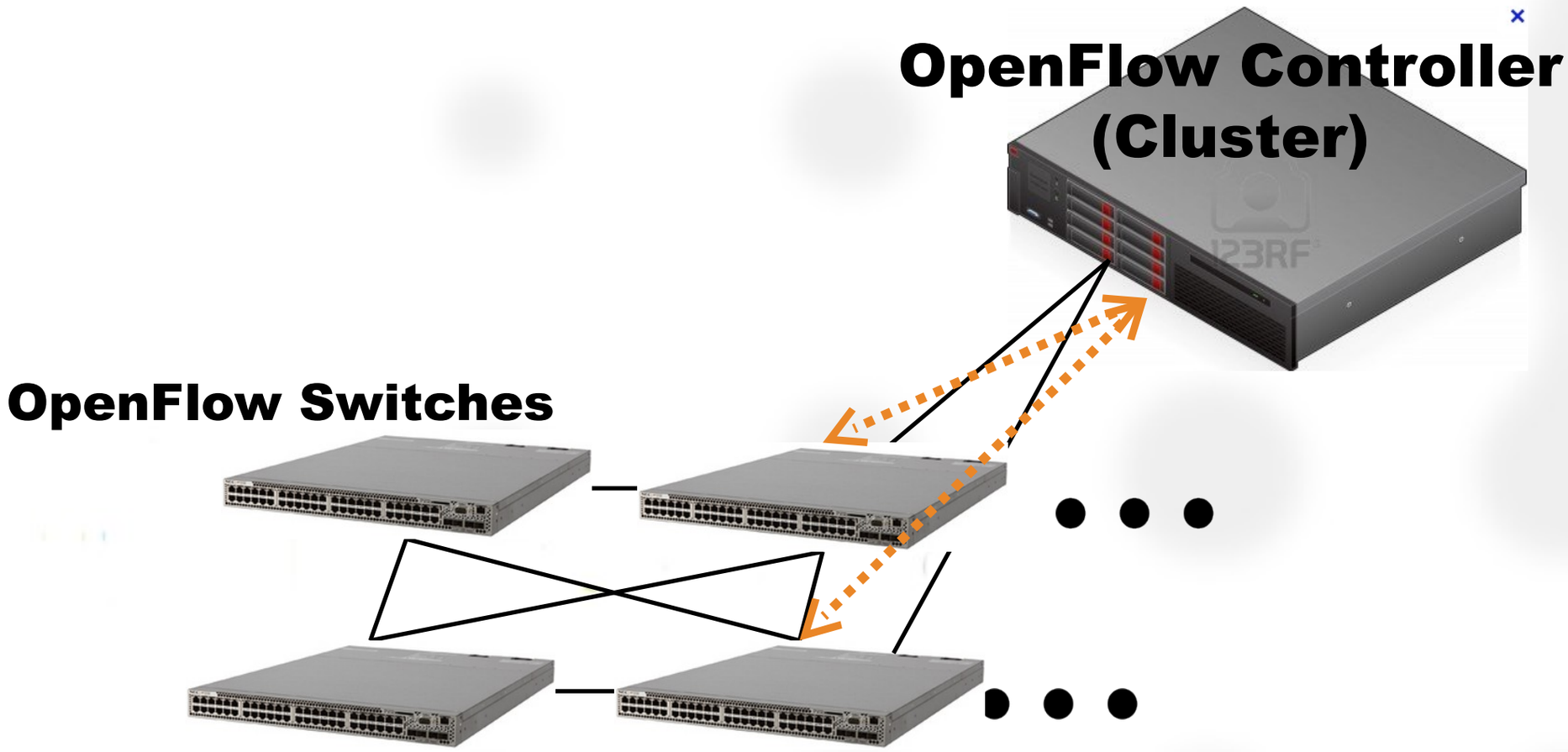


出典： http://infrastructureadventures.com/tag/vrf-lite/

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

Copyright © 2012 Midokura All rights reserved

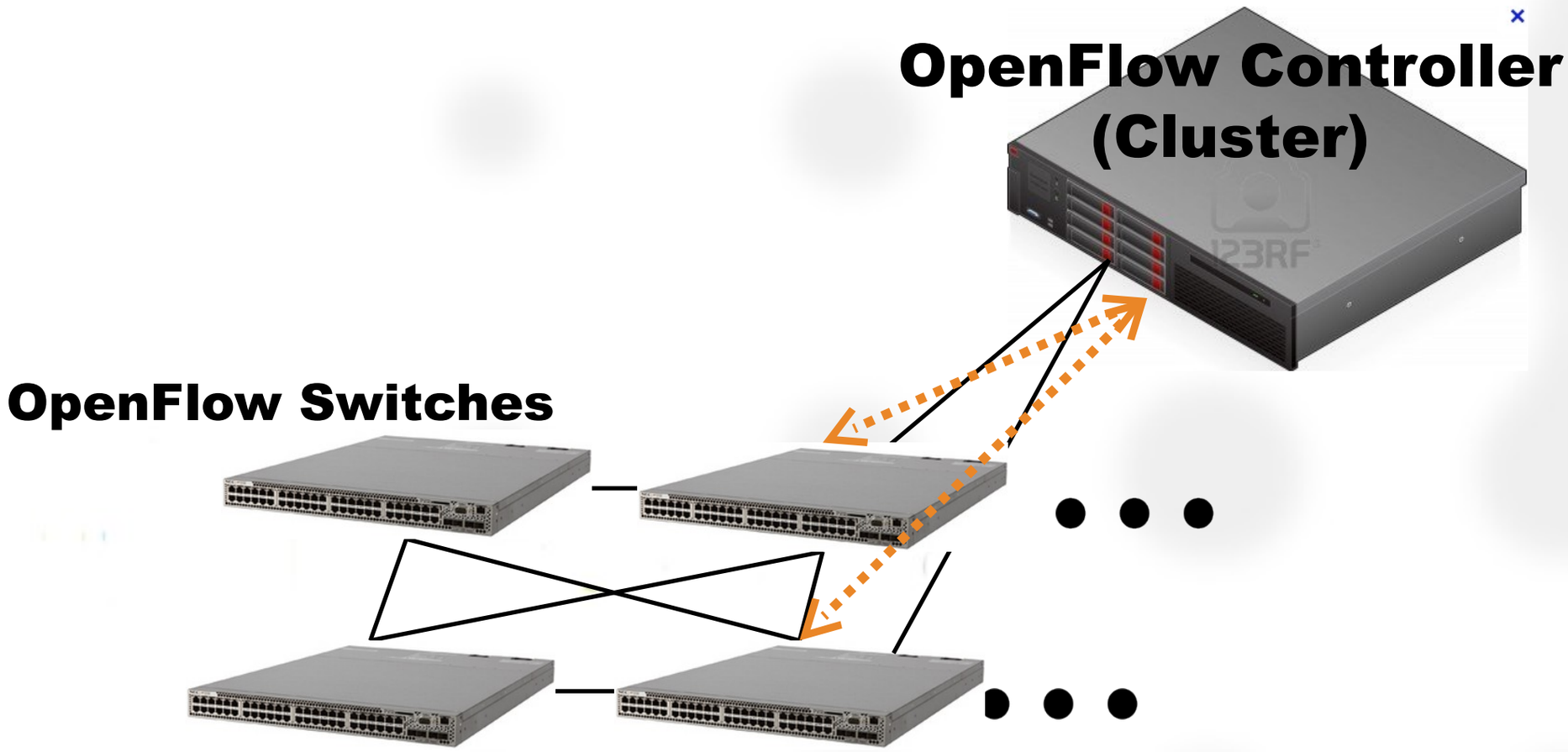
2 OpenFlow hop-by-hop switch fabric



- 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.

Copyright © 2012 Midokura All rights reserved

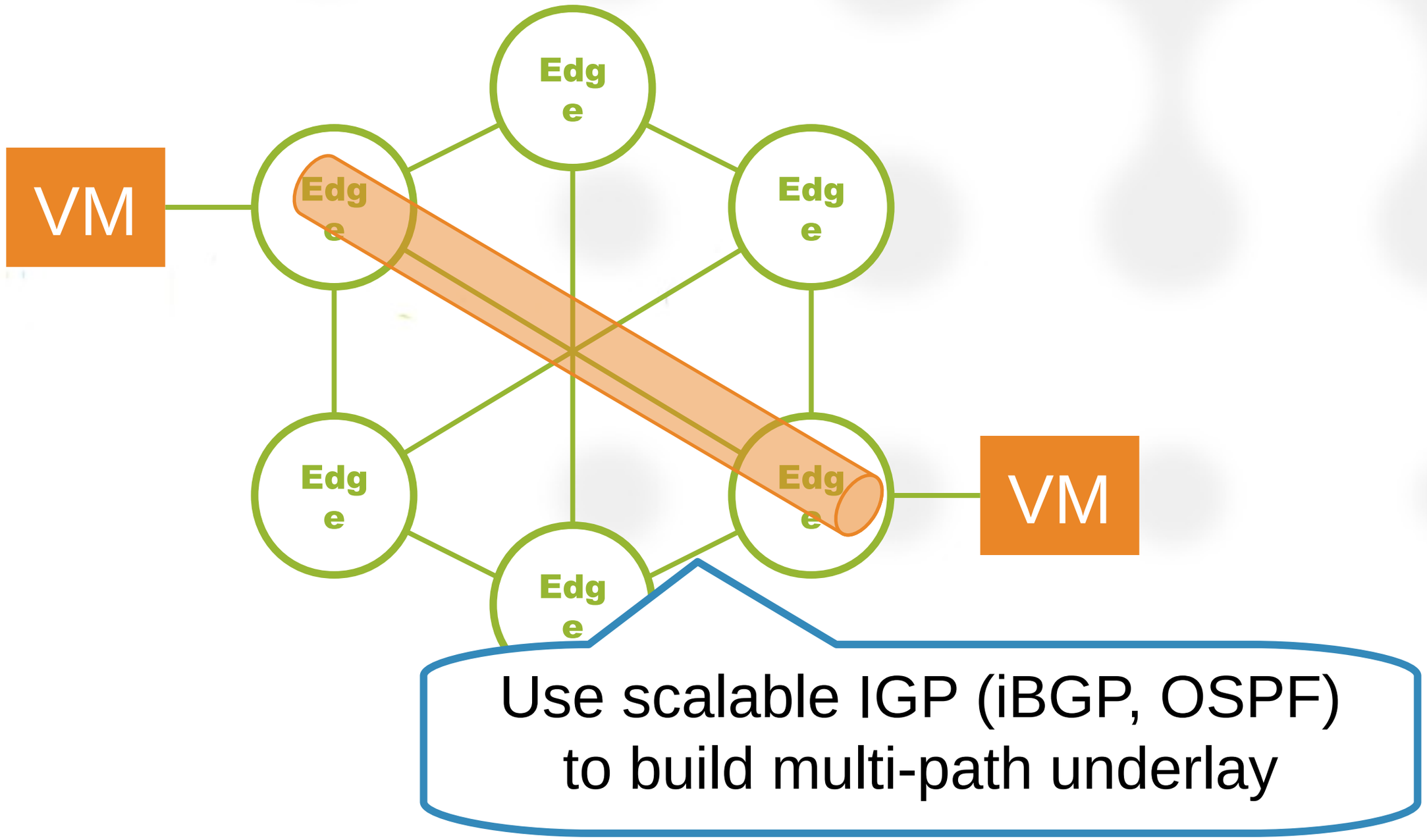
2 OpenFlow hop-by-hop switch fabric (more)



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

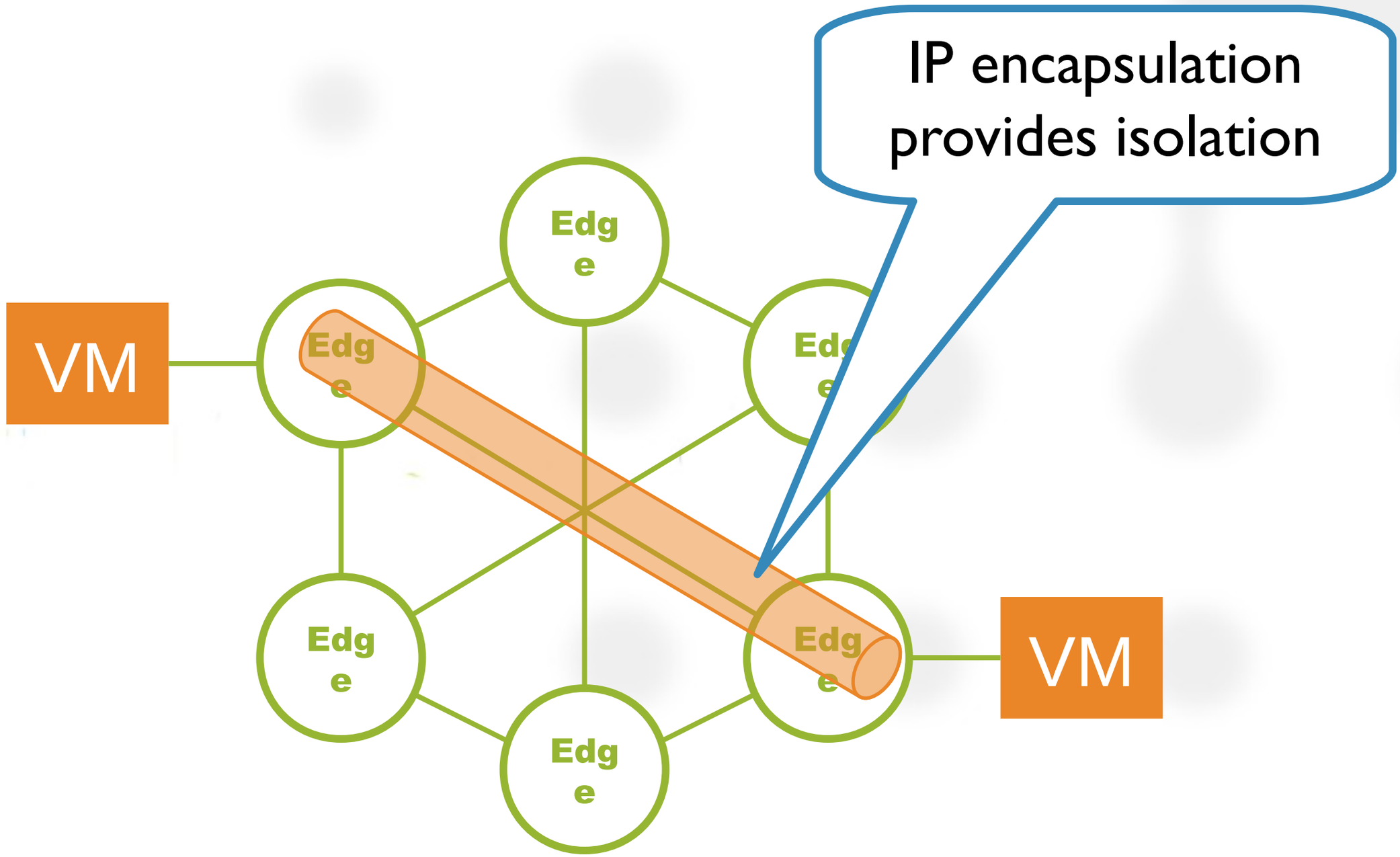
Copyright © 2012 Midokura All rights reserved

3 Edge-to-Edge Overlays



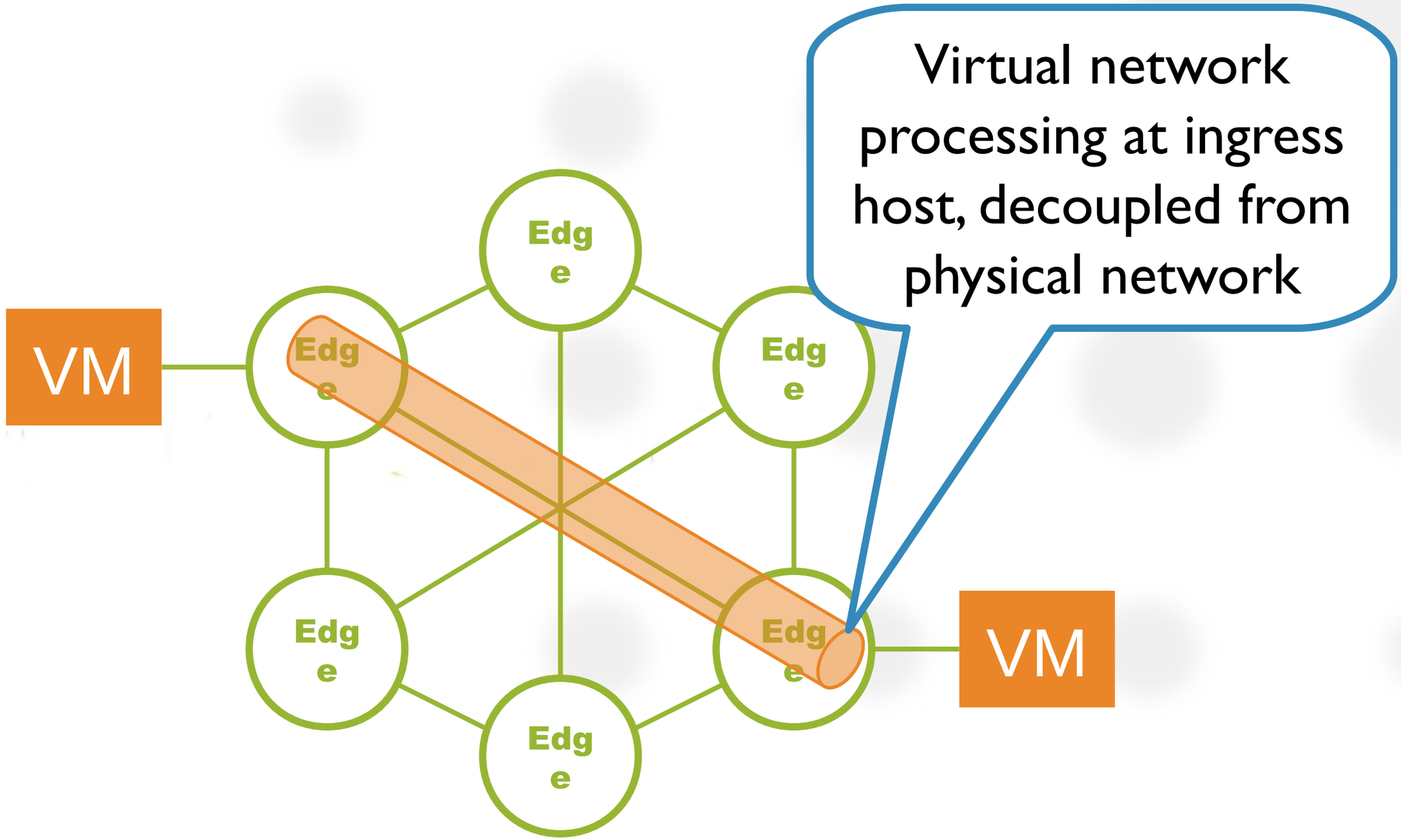
Copyright © 2012 Midokura All rights reserved

3 Edge-to-Edge Overlays



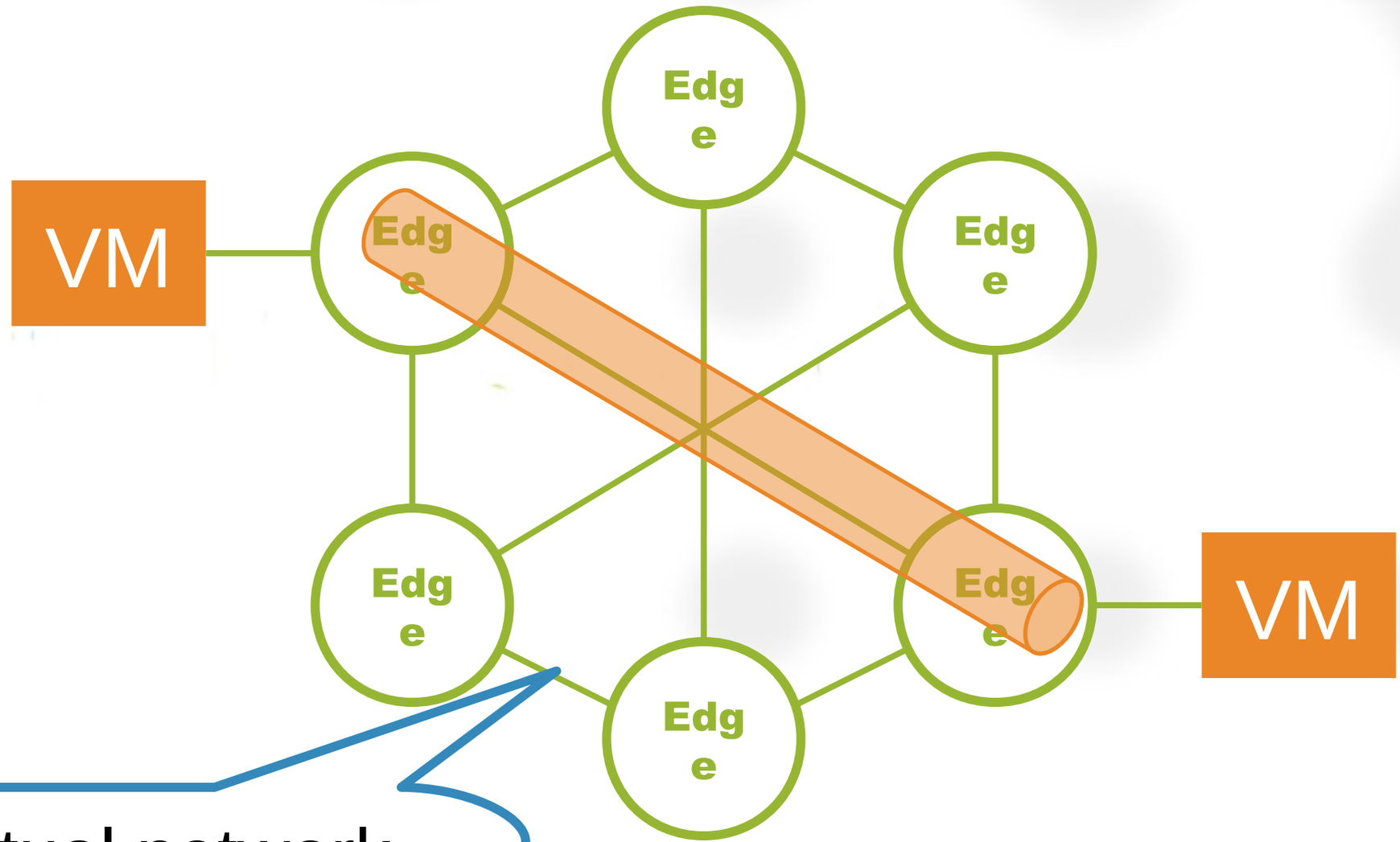
Copyright © 2012 Midokura All rights reserved

3 Edge-to-Edge Overlays



Copyright © 2012 Midokura All rights reserved

3 Edge-to-Edge Overlays



Virtual network changes don't affect underlay state

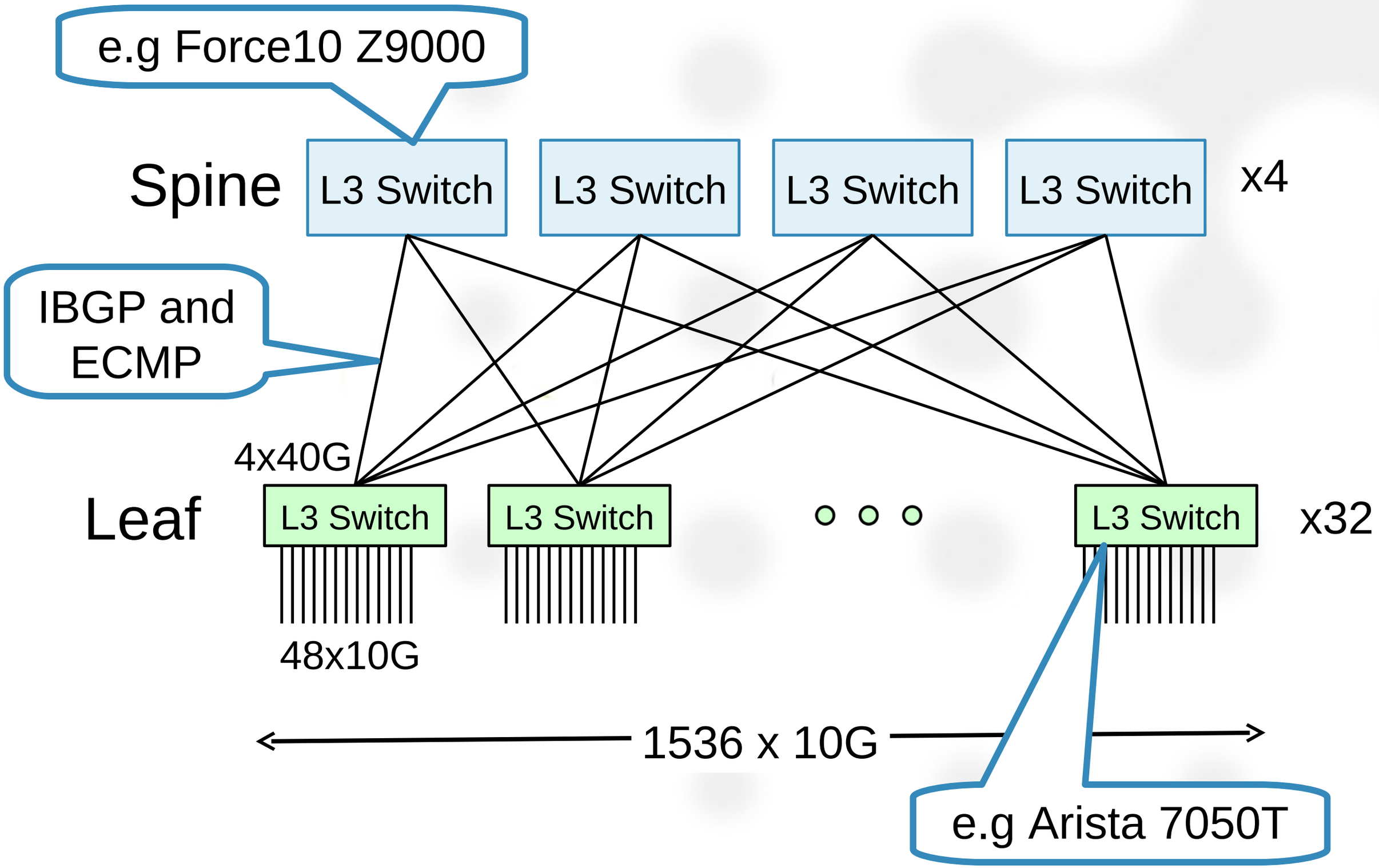
Copyright © 2012 Midokura All rights reserved

3 Edge-to-Edge Overlays

- 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

Copyright © 2012 Midokura All rights reserved

Spine and Leaf Network Architecture



Copyright © 2012 Midokura All rights reserved

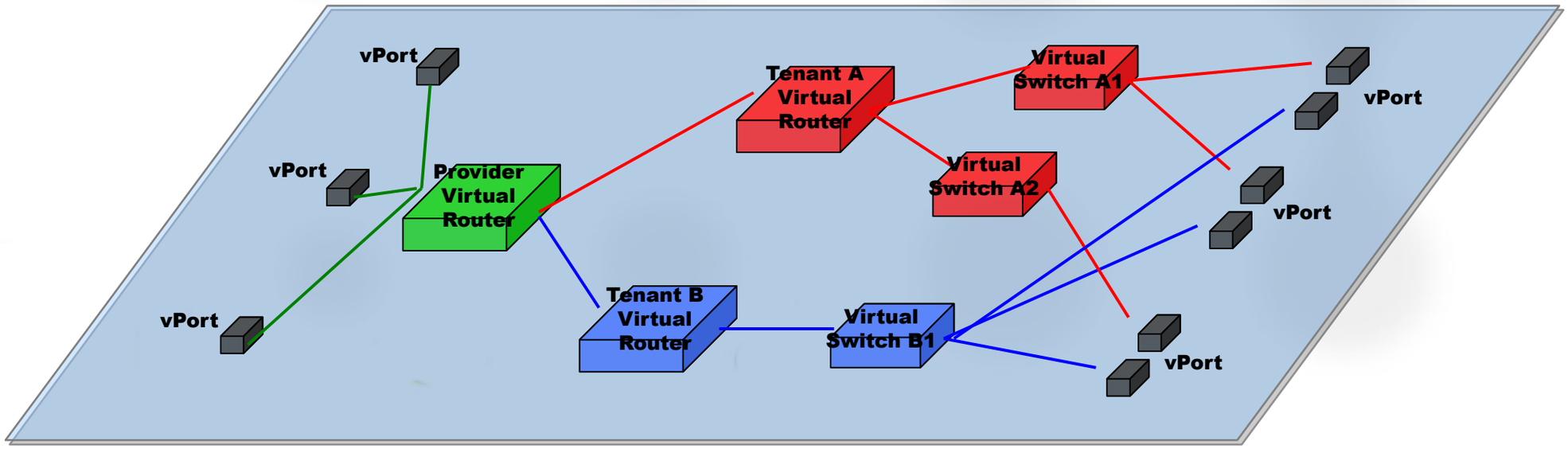
Overlays are the right approach!

But not sufficient...

We still need a scalable control plane.

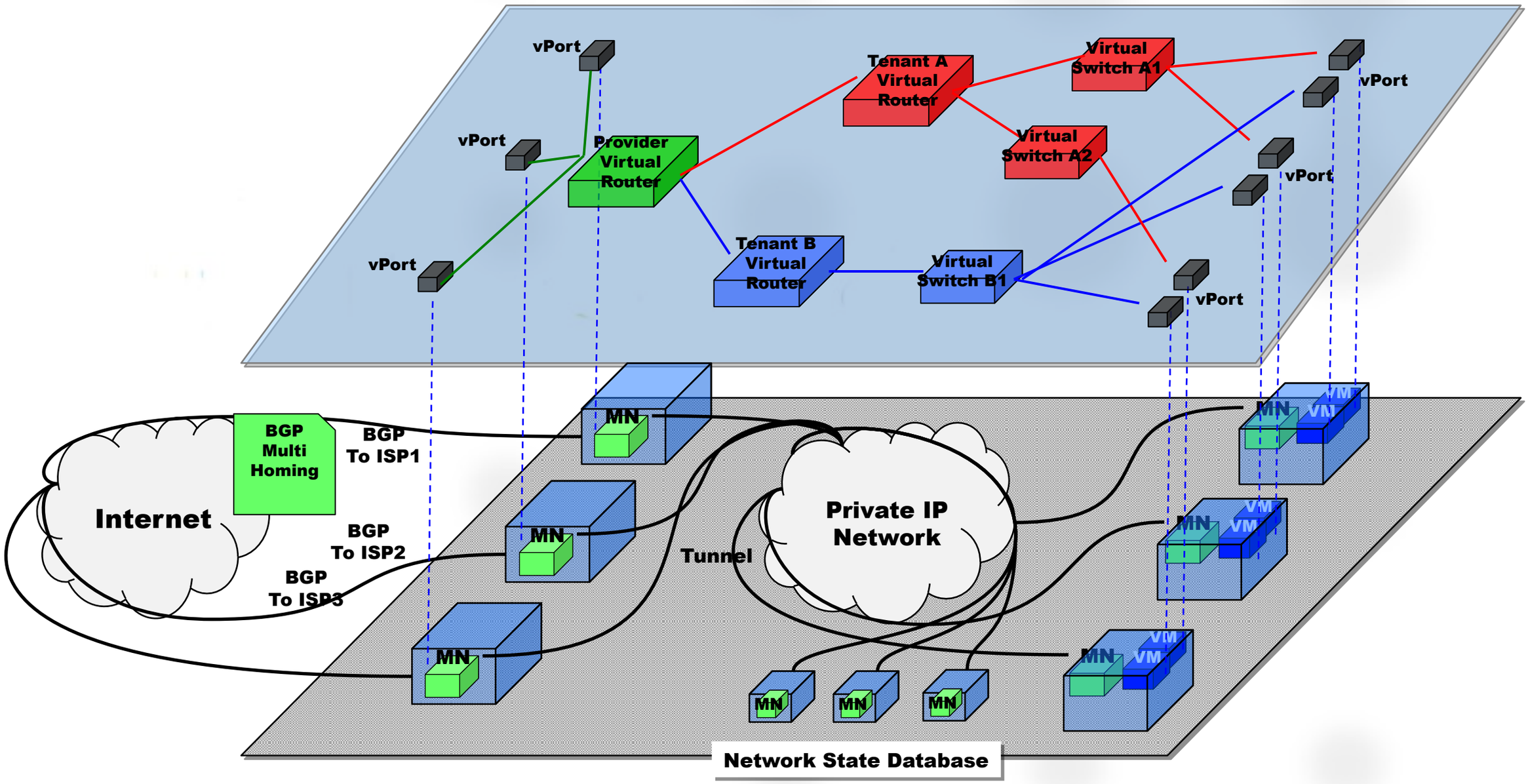
Copyright © 2012 Midokura All rights reserved

Logical Topology



Copyright © 2012 Midokura All rights reserved

Logical Topology



Physical Topology

Copyright © 2012 Midokura All rights reserved

MidoNet SDN Solution

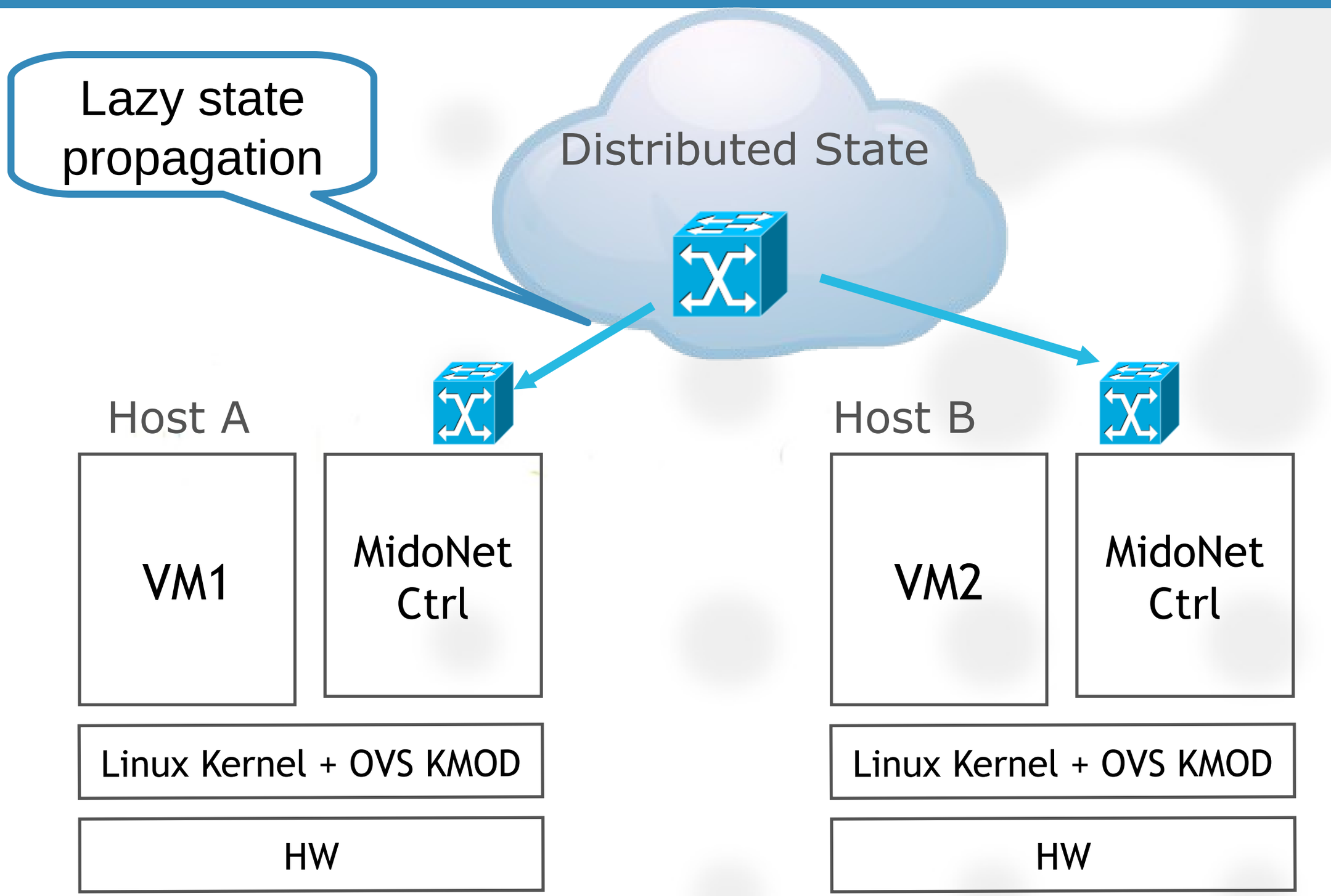
Dashboard

MidoNet REST API

Distributed State

Copyright © 2012 Midokura All rights reserved

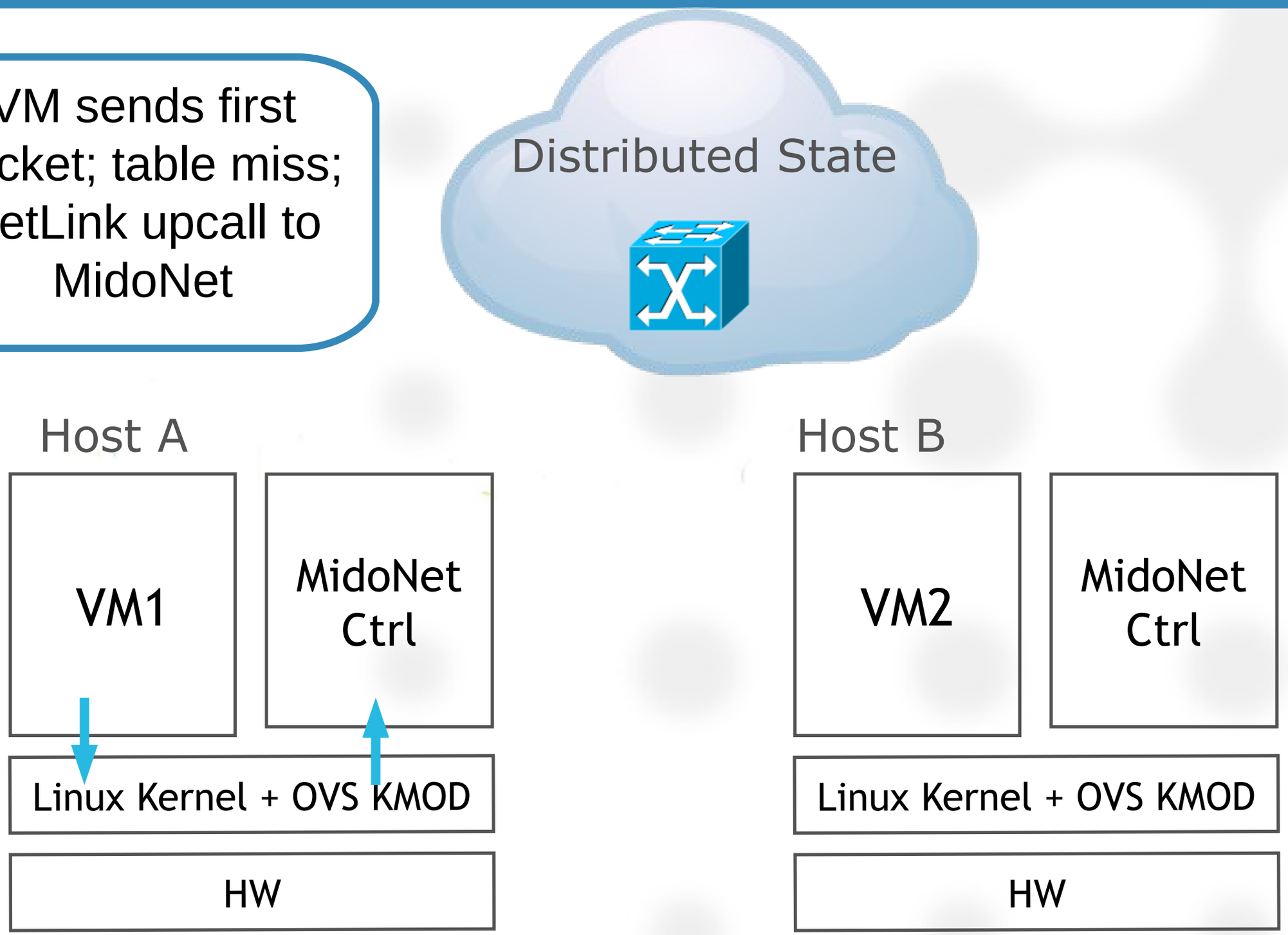
MidoNet SDN Solution



Copyright © 2012 Midokura All rights reserved

MidoNet SDN Solution

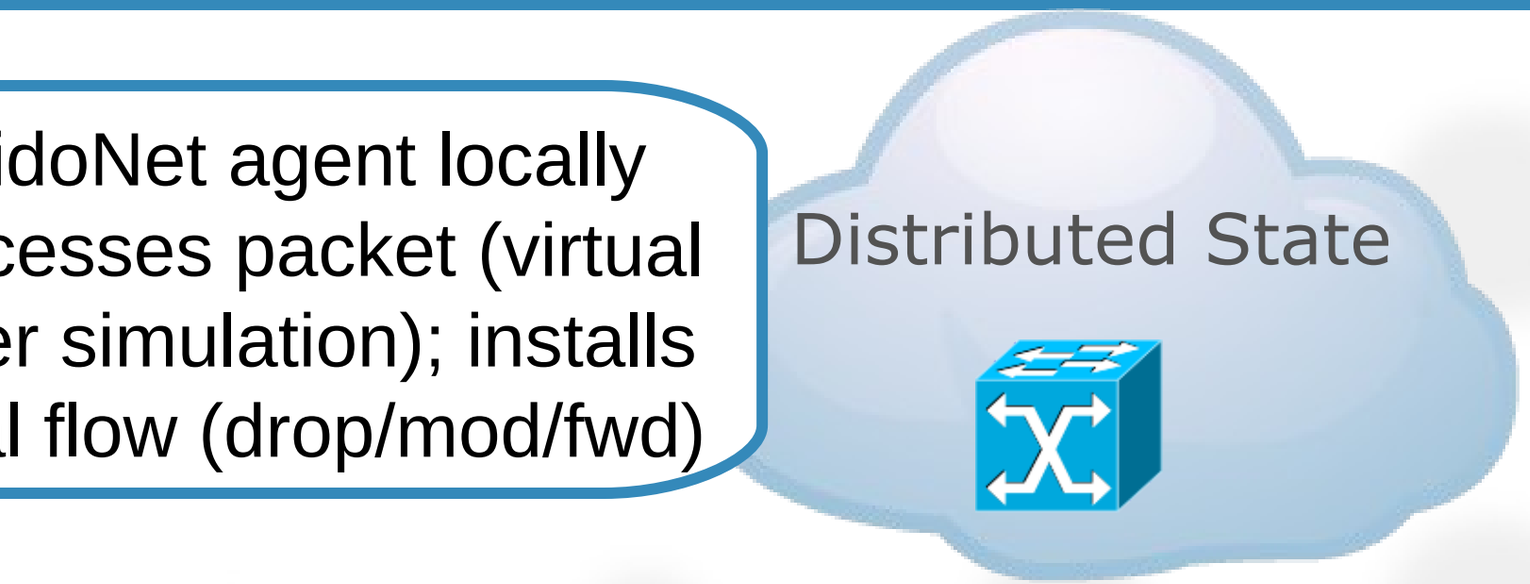
VM sends first packet; table miss; NetLink upcall to MidoNet



Copyright © 2012 Midokura All rights reserved

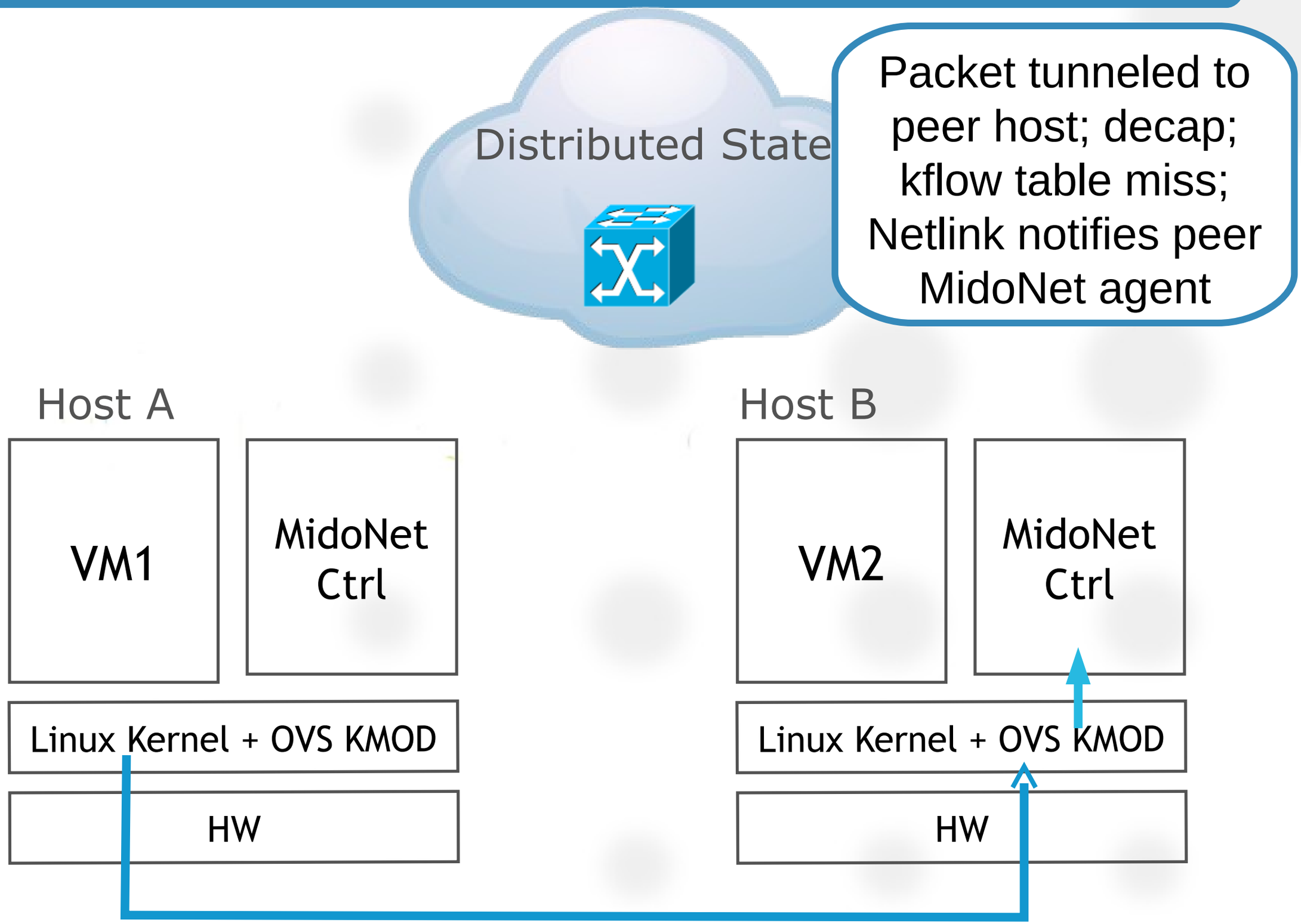
MidoNet SDN Solution

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



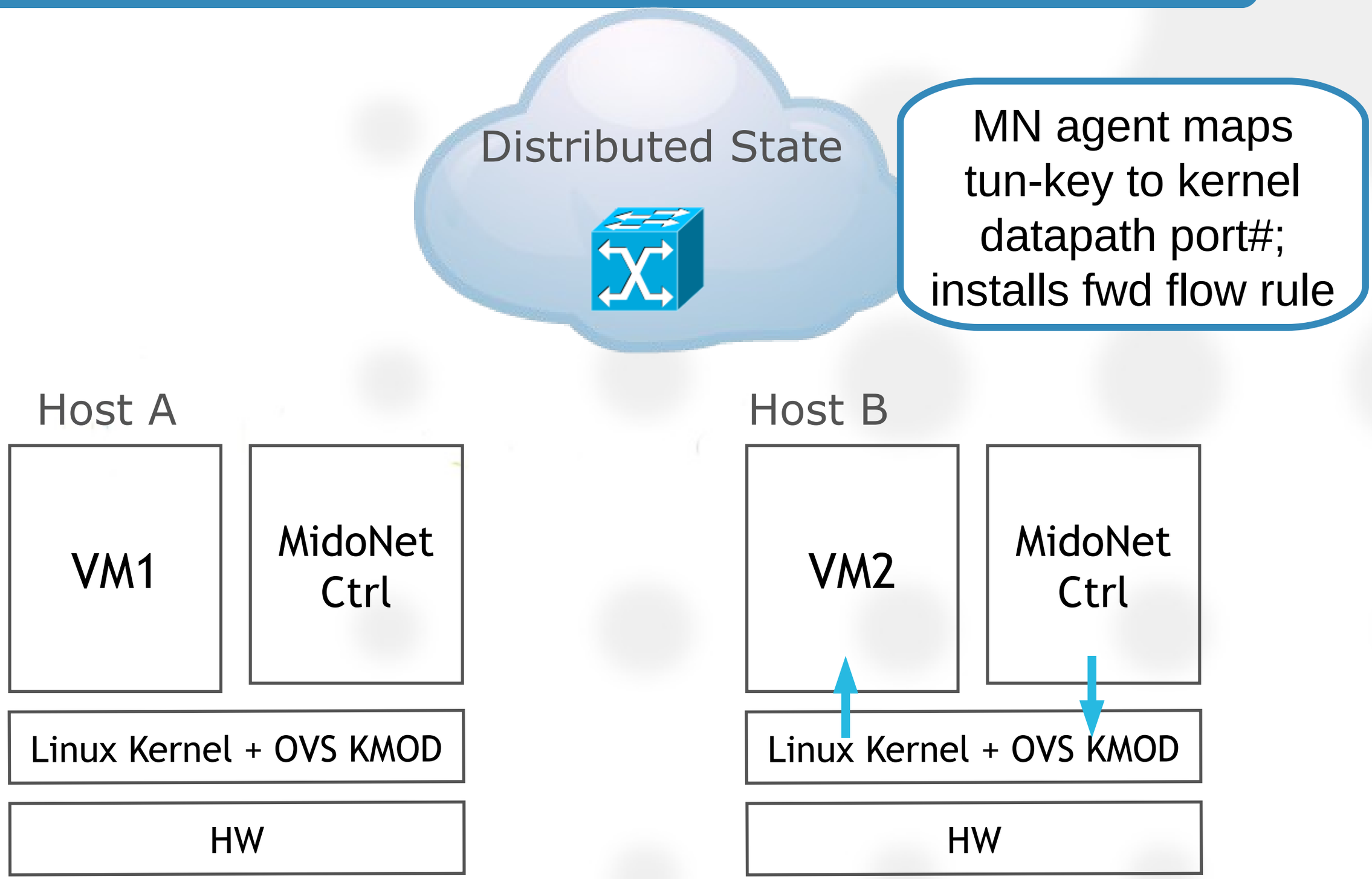
Copyright © 2012 Midokura All rights reserved

MidoNet SDN Solution



Copyright © 2012 Midokura All rights reserved

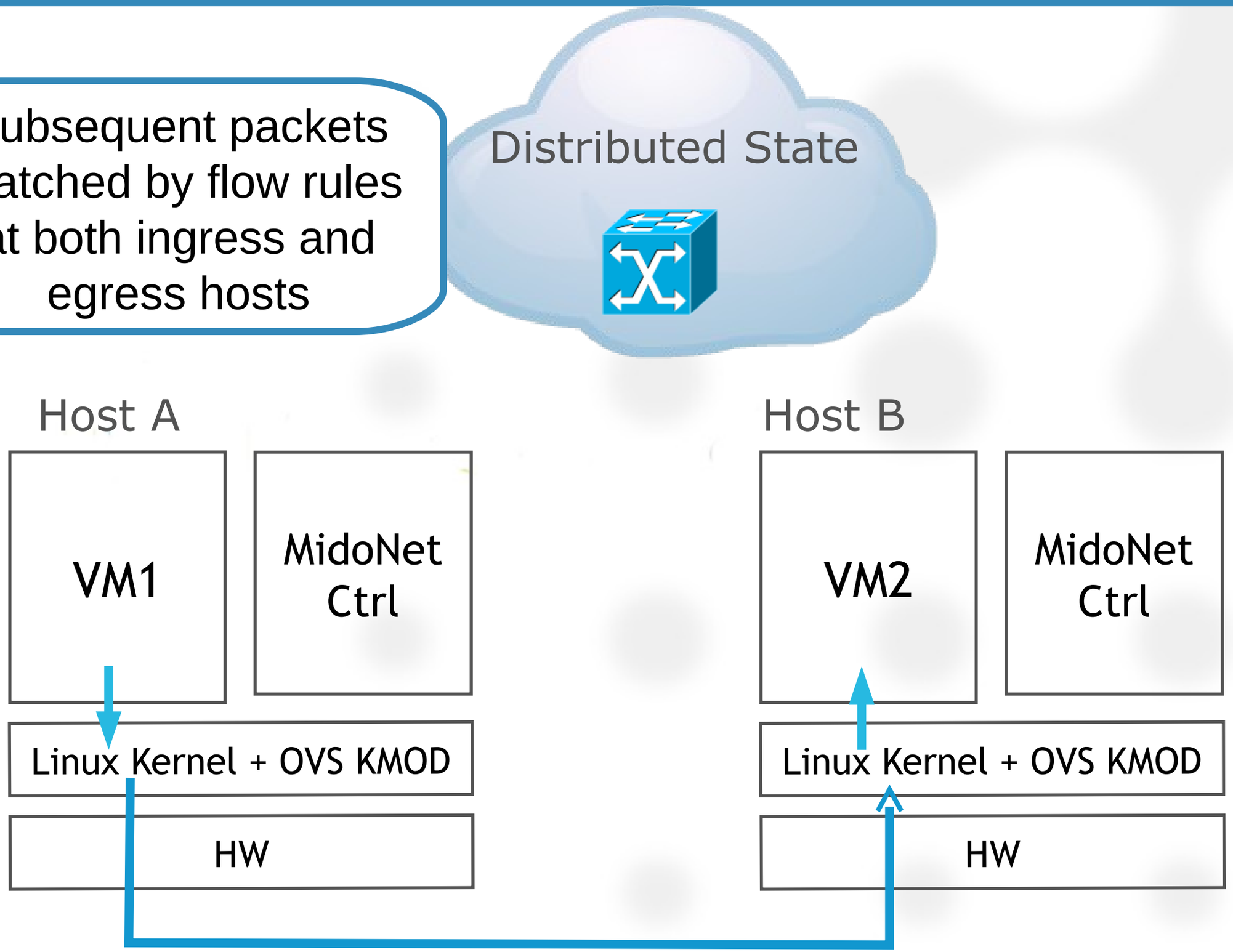
MidoNet SDN Solution



Copyright © 2012 Midokura All rights reserved

MidoNet SDN Solution

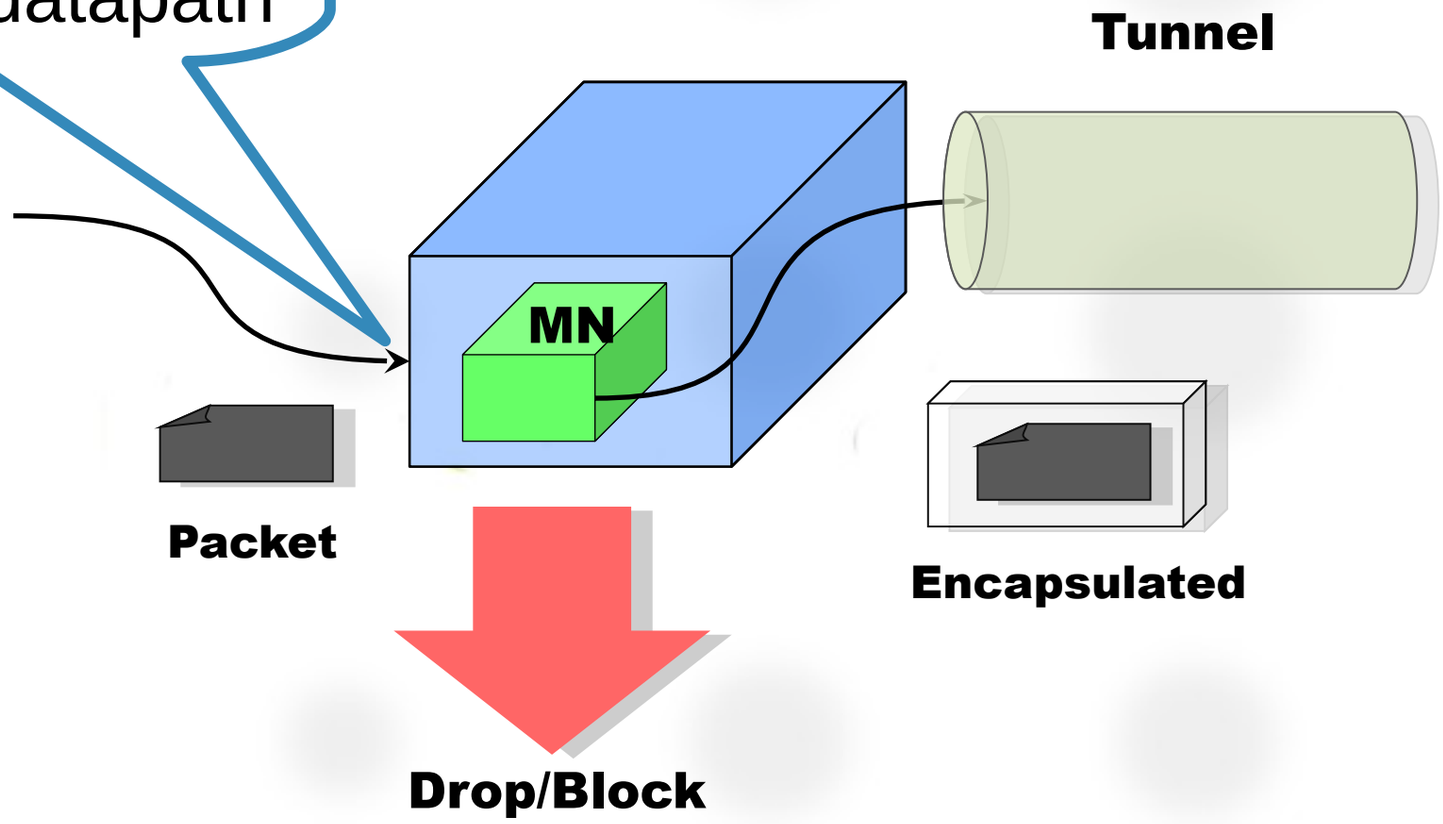
Subsequent packets matched by flow rules at both ingress and egress hosts



Copyright © 2012 Midokura All rights reserved

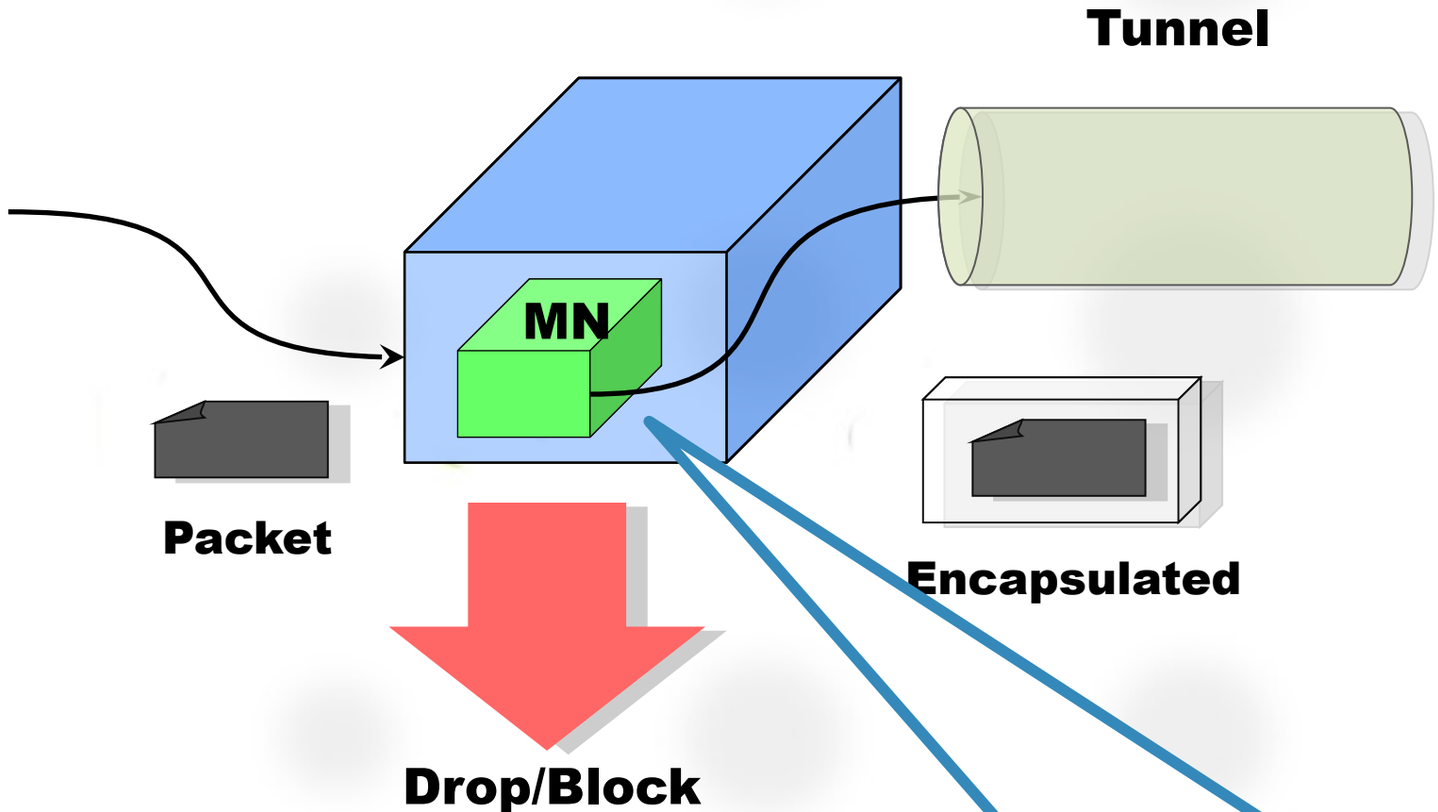
MidoNet SDN Solution

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



Copyright © 2012 Midokura All rights reserved

MidoNet SDN Solution



One flow rule reflecting the outcome of the virtual layer simulation AND the mapping of egress vport to peer host decides to drop or fwd

Copyright © 2012 Midokura All rights reserved

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

Copyright © 2012 Midokura All rights reserved

MidoNet SDN Solution

- 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

Copyright © 2012 Midokura All rights reserved

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)

Copyright © 2012 Midokura All rights reserved

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

Copyright © 2012 Midokura All rights reserved

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

Copyright © 2012 Midokura All rights reserved

Questions?

Midokura is hiring!

in TYO, SFO, and BCN

careers@midokura.com

midokura