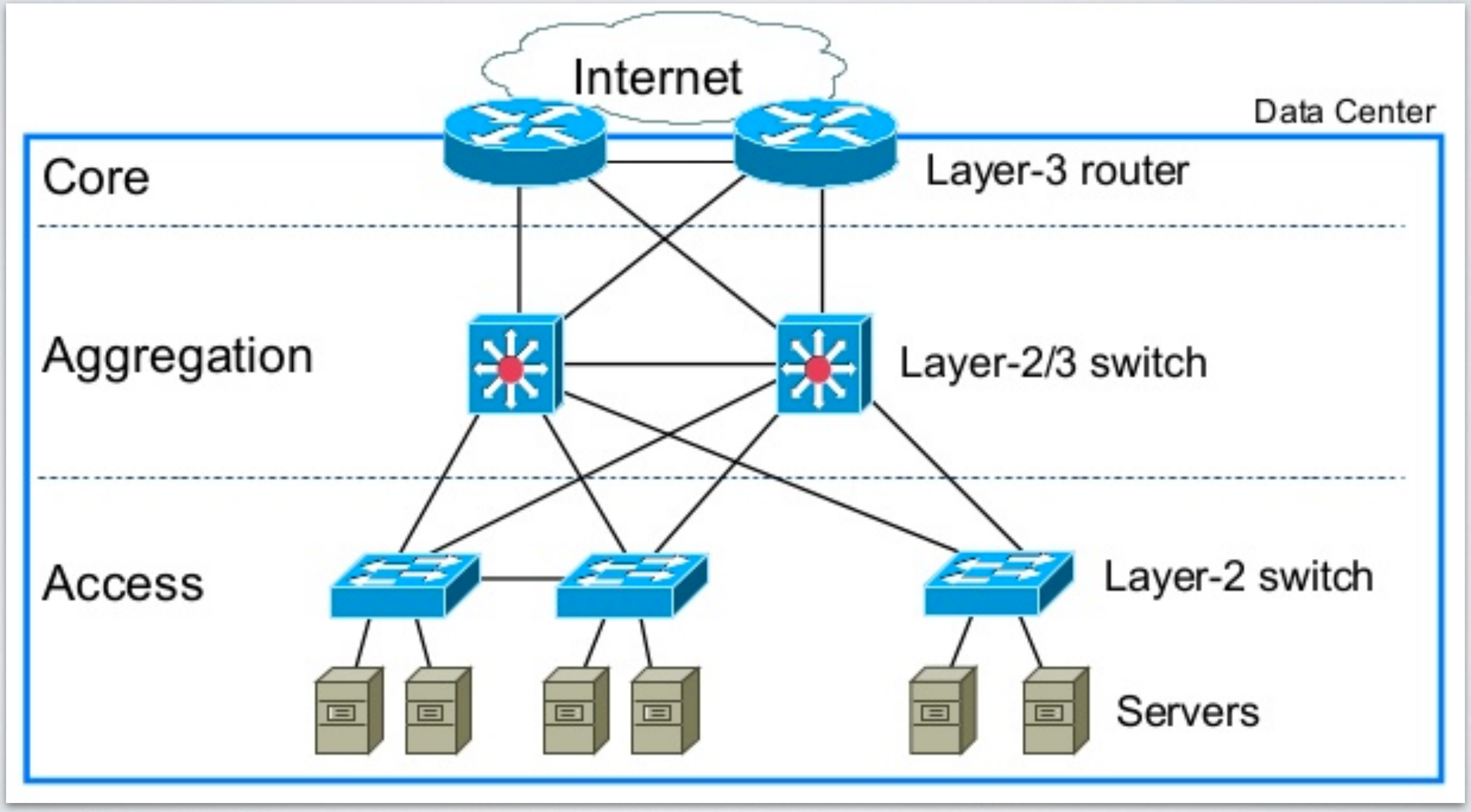
NETWORK VIRTUALIZATION

Ethan J. Jackson

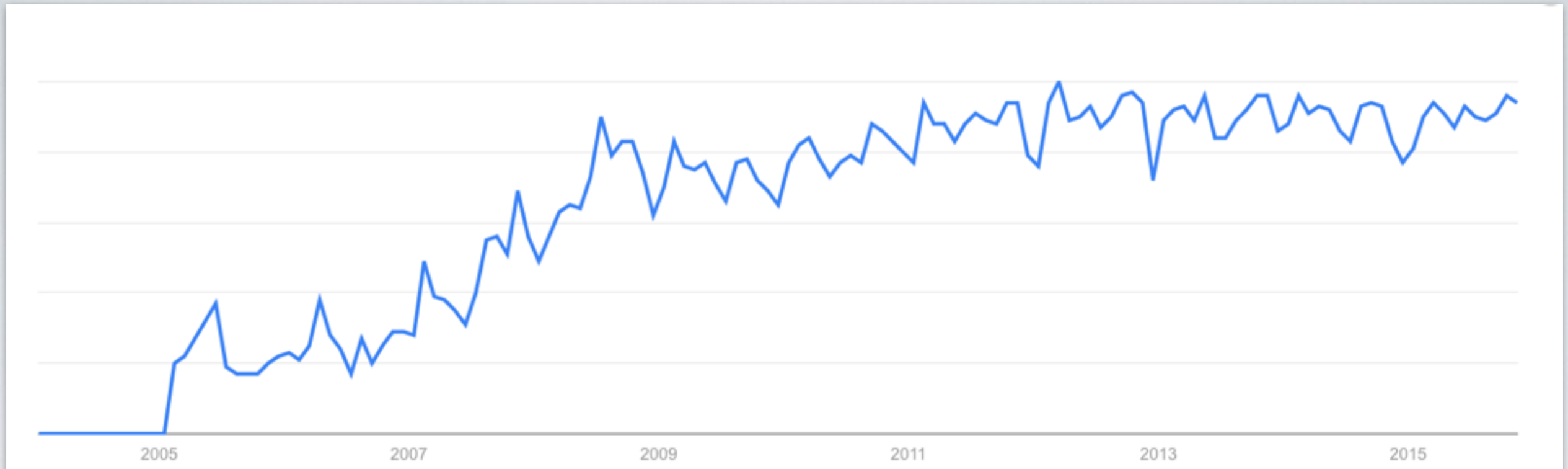
THE YEAR 2000

ENTERPRISE DATACENTER



ENTERPRISE DATACENTER

- Static Hosts
- Single Tenant
- Manual configuration
- Networking == Hardware



GOOGLE TRENDS

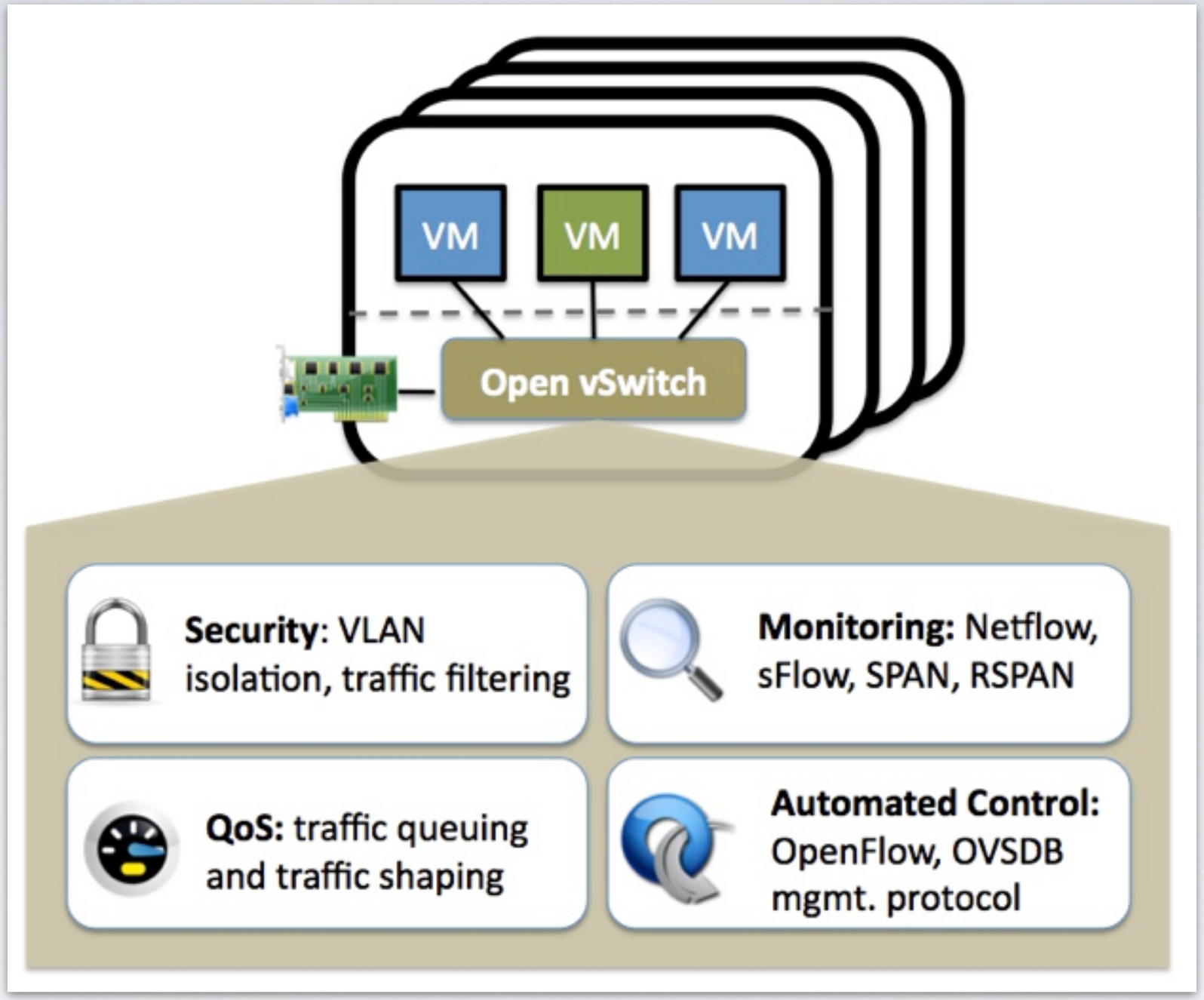
"Hypervisor"

VIRTUALIZED DATACENTER

- Static Hosts <-> Dynamic Hosts
  - Virtual machines boot/die quickly
  - Live migration
  - What does this do to routing?

VIRTUALIZED DATACENTER

- Single Tenant <-> Multi-tenant
  - Public Clouds (Amazon)
  - Private Clouds
    - Unified infrastructure for all applications
- How do we secure this?



VIRTUALIZED DATACENTER

The Hypervisor vSwitch

VIRTUALIZED DATACENTER

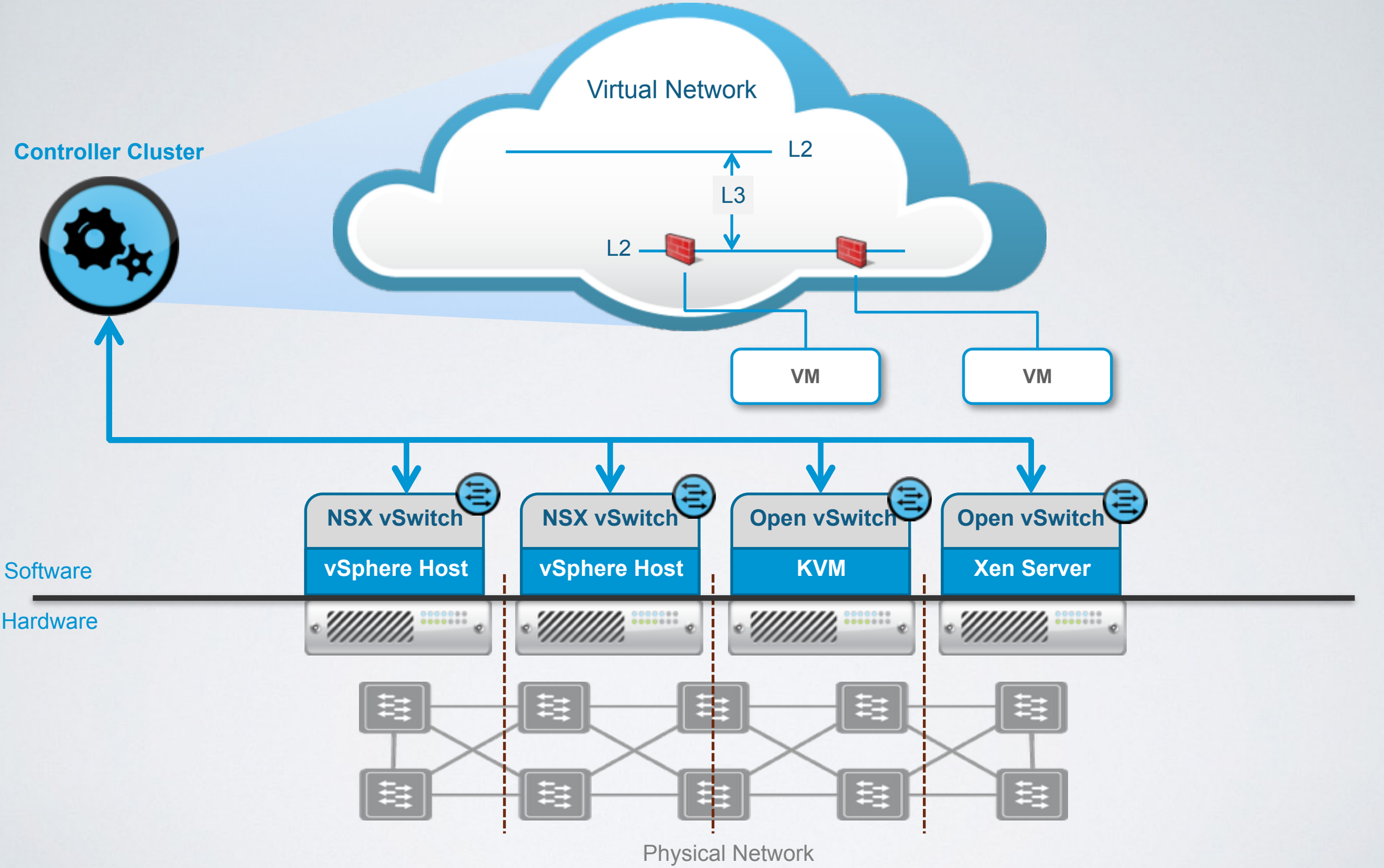
- Networks were broken
  - A high dynamic virtualized cloud
- Managed manually by network admins

THE PROBLEM

Network Policy is Chained to Network Hardware

THE SOLUTION

Network Virtualization


Controller Cluster
Virtual Network
L2
L3
L2
VM
VM
NSX vSwitch
vSphere Host
NSX vSwitch
vSphere Host
Open vSwitch
KVM
Open vSwitch
Xen Server
Software
Hardware
Physical Network

ABSTRACTION

- Physical network does connectivity (only)
- Virtual network does policy (only)
- Each evolves independently

SIMPLICITY

- Do complex stuff at the edge
  - In software
- Handle global state in a controller
  - Instead of distributed algorithms
- Use cheap, commodity, hardware
  - Instead of Cisco

ADVANTAGES

- Automation (humans cause bugs)
- Deployment Velocity
- Micro-segmentation
- Cloud Bursting
- Inter-DC Migration

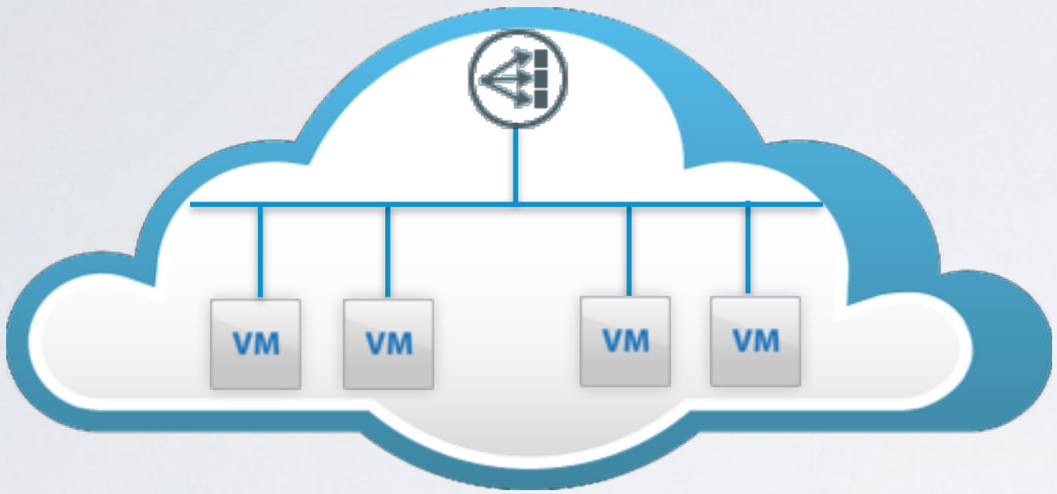
THE ARCHITECTURE

THREE PILLARS

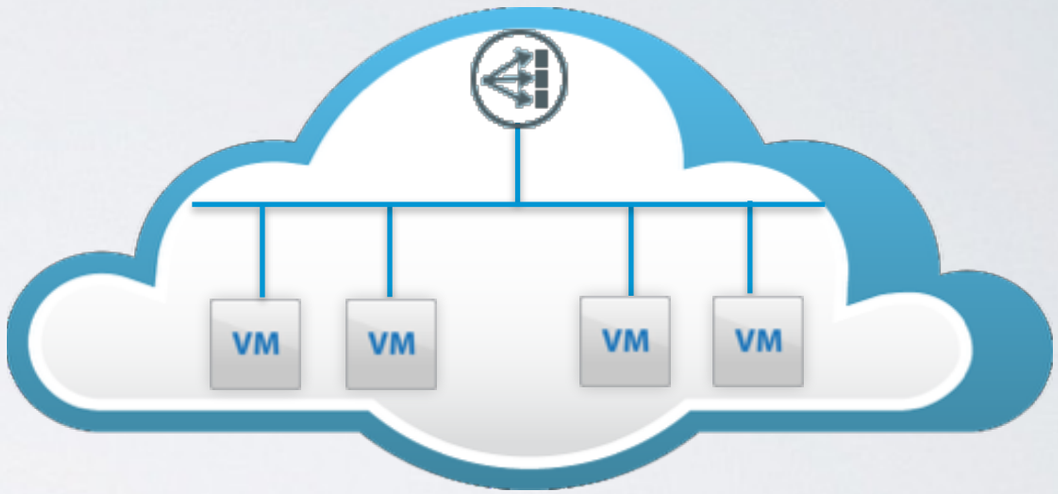
- Tunnels
- Virtual Switches
- Controllers

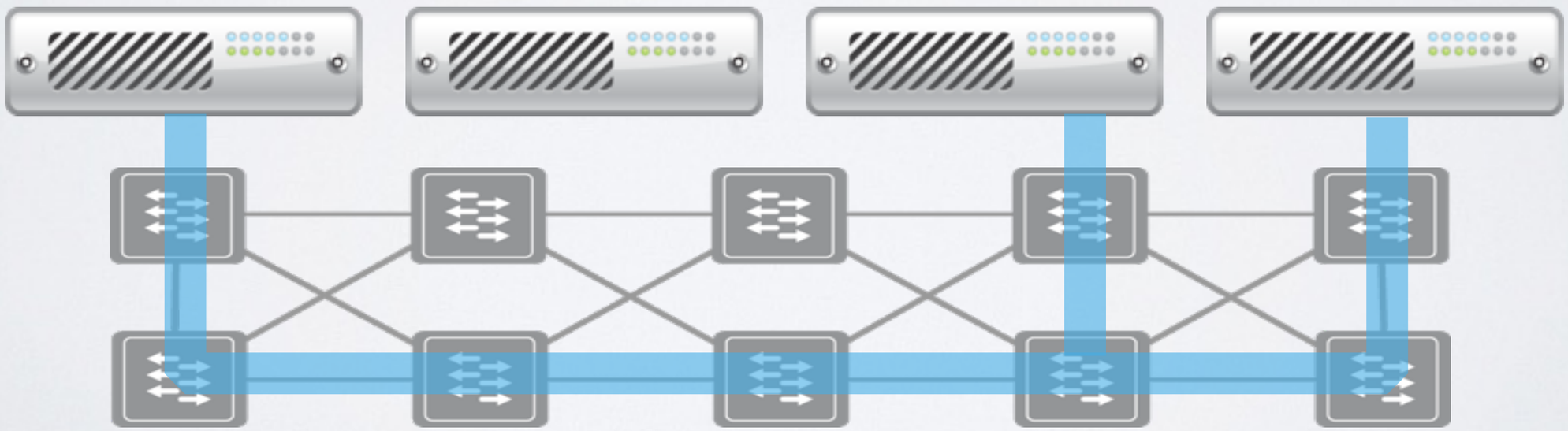
TUNNELS

10.0.0.0/8

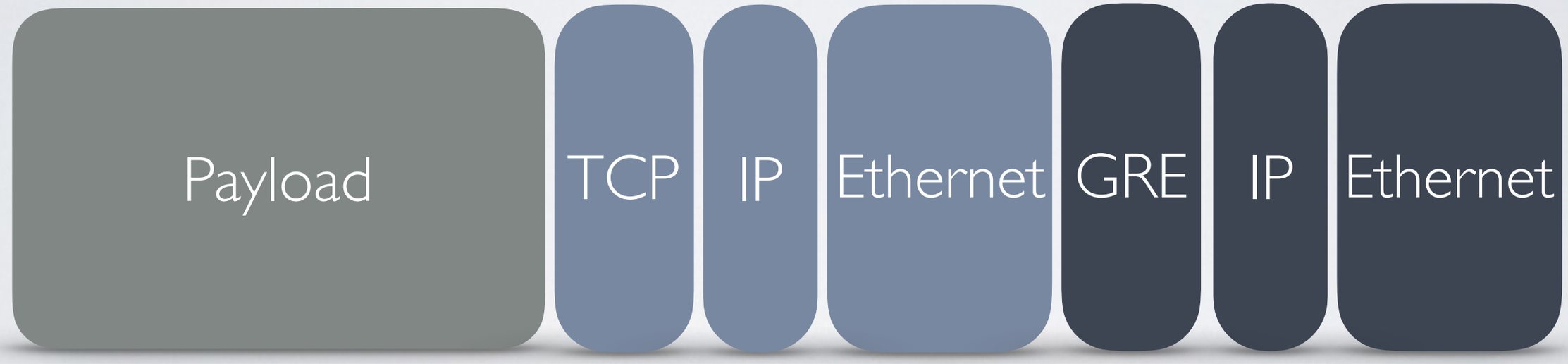


10.0.0.0/8



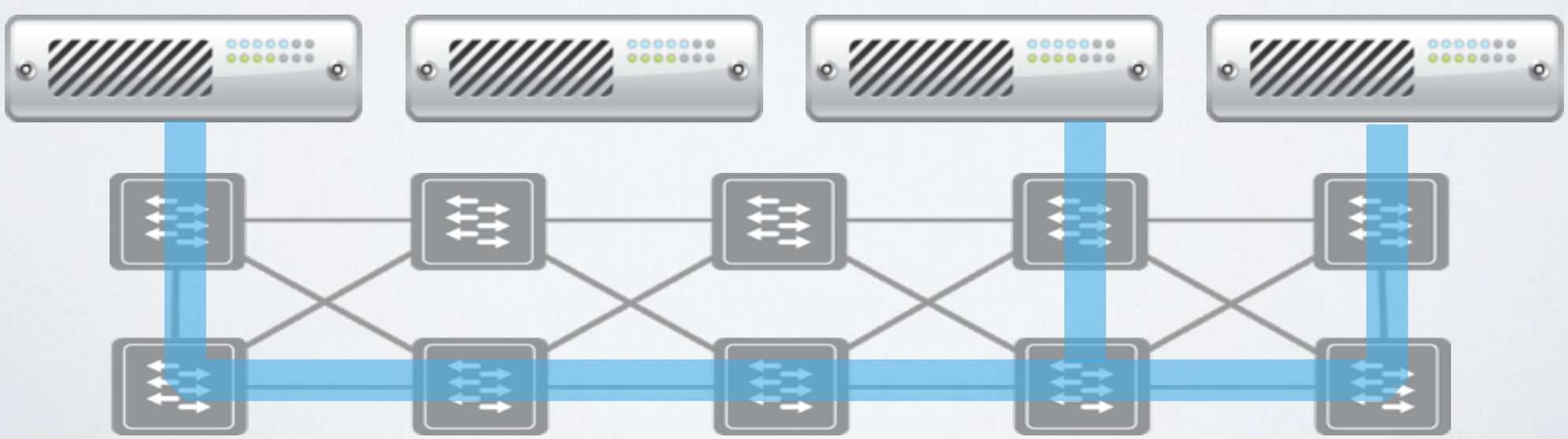


TUNNELS


Payload
TCP
IP
Ethernet
GRE
IP
Ethernet
10.0.0.1 -> 10.0.0.2
169.229.48.50 -> 217.12.1.125

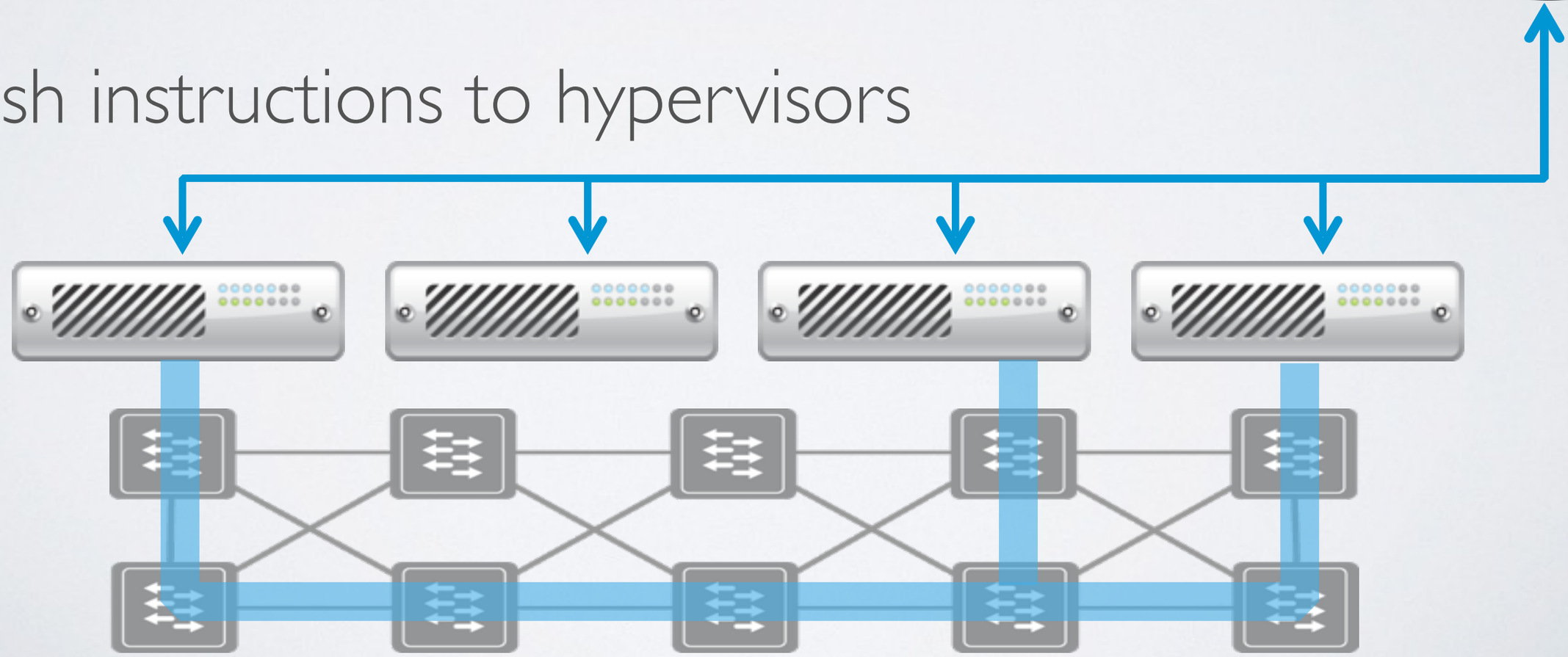
TUNNELS

- Mesh of tunnels
- Which one to use?

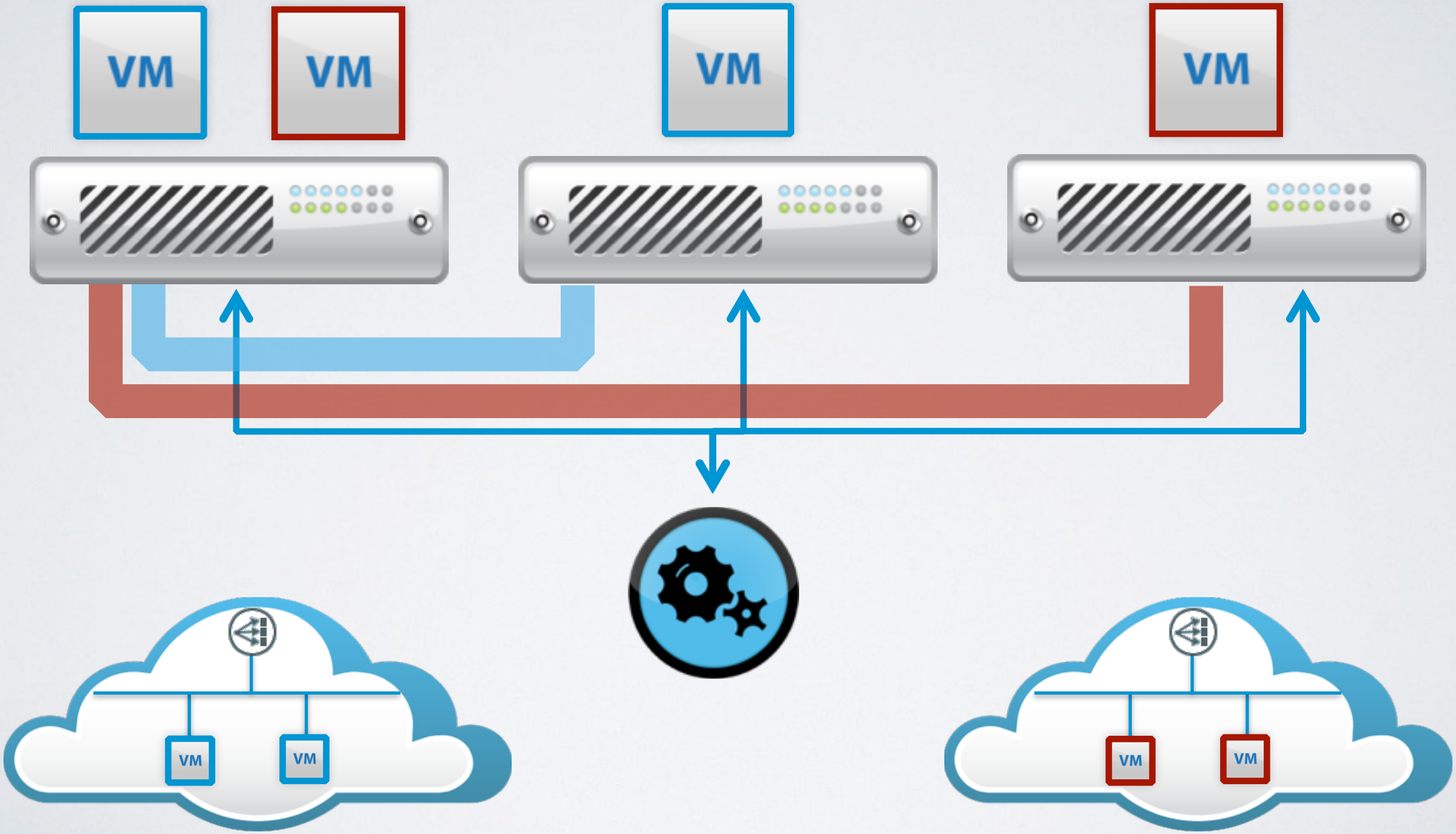


CONTROLLERS

- Receive virtual network configuration
- Monitor hypervisor state
- Push instructions to hypervisors

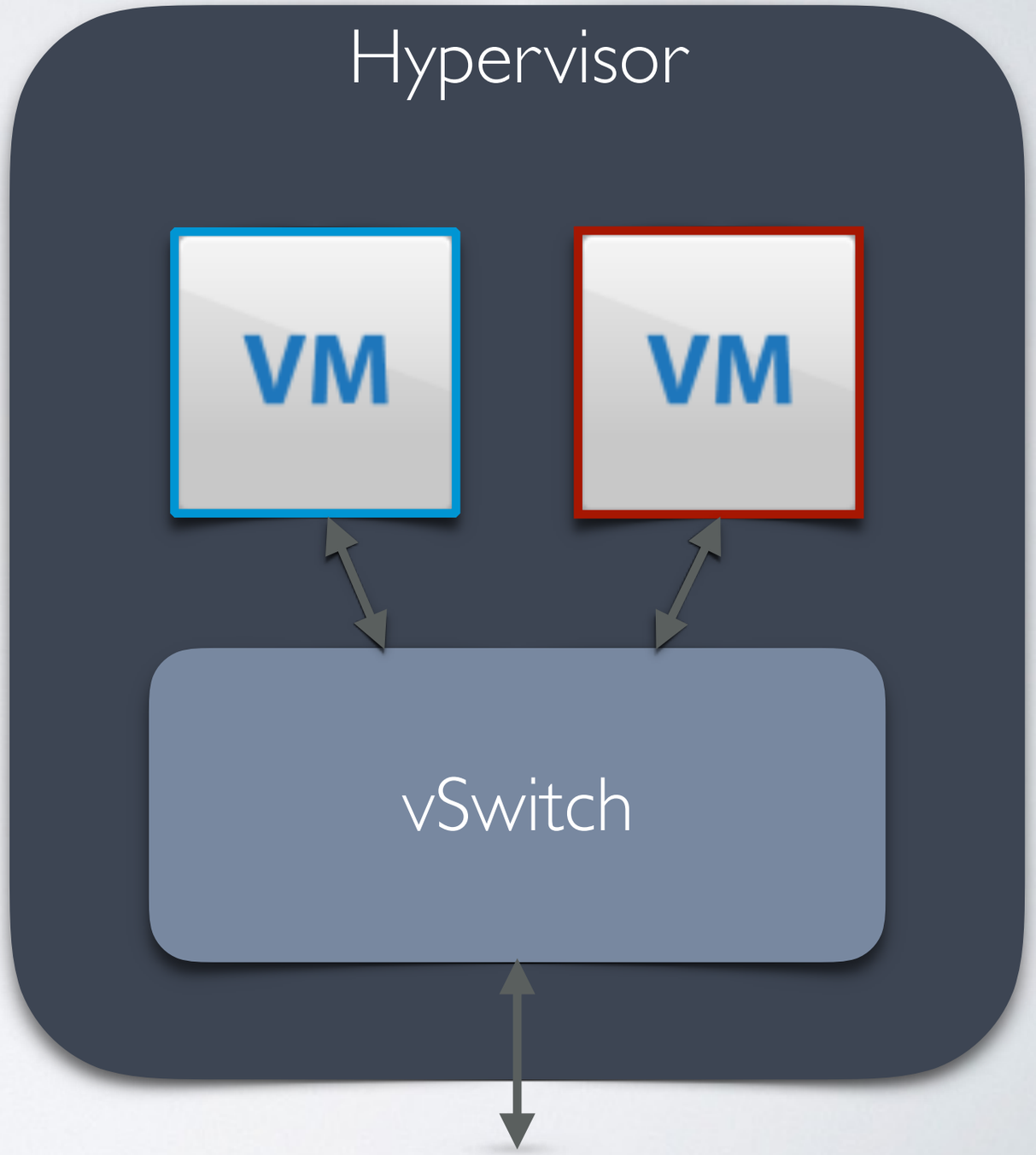


CONTROLLERS



TRADITIONAL **v**SWITCHES

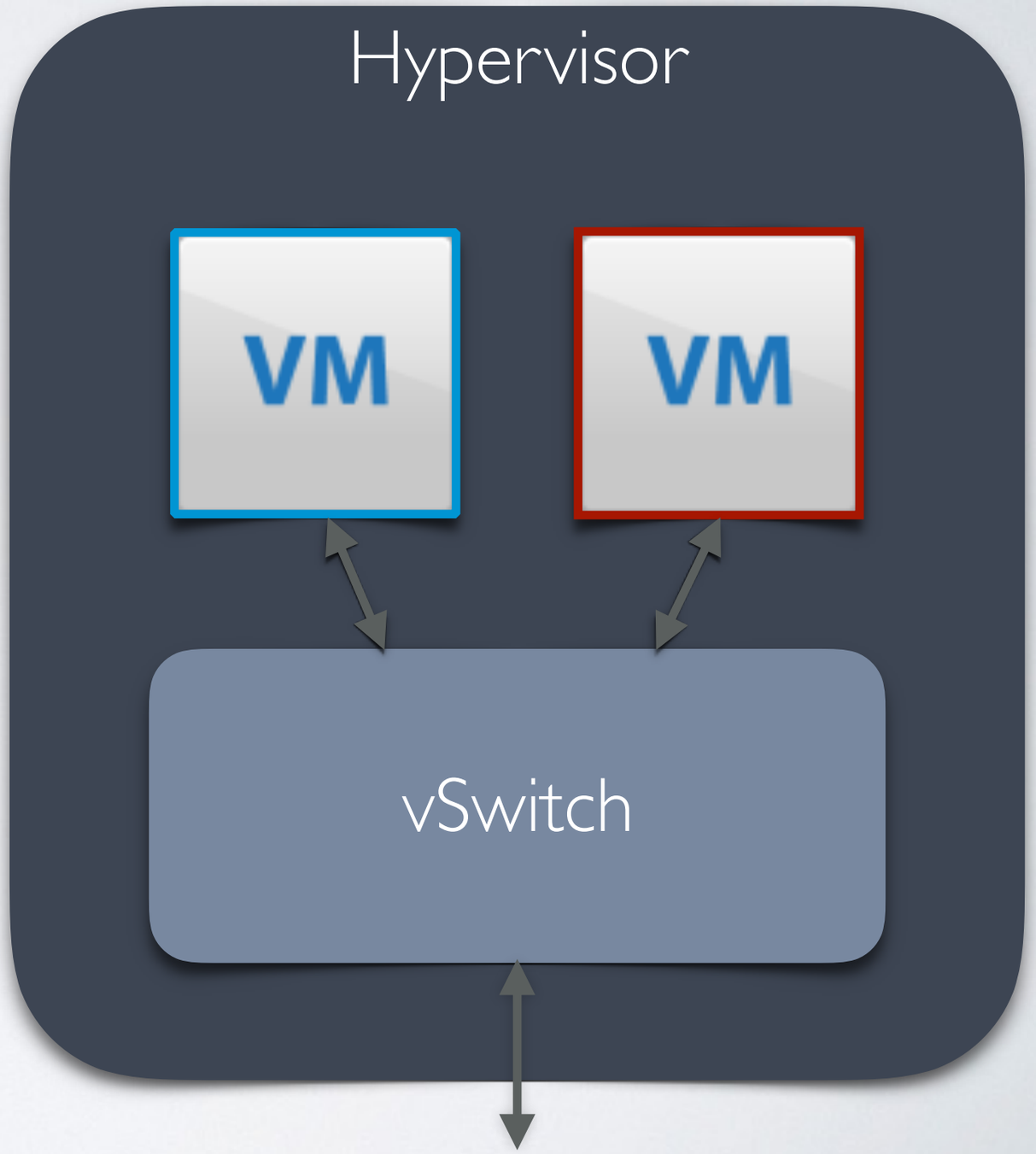
- Core component of hypervisors
- Connect VMs to each other
  - And the physical network
- Traditionally simple L2
  - Linux Bridge



MODERN **v**SWITCHES

- Implement Controller Policy
  - Create Tunnels
  - Implement Logical Pipeline
- Monitor Remote Hypervisors

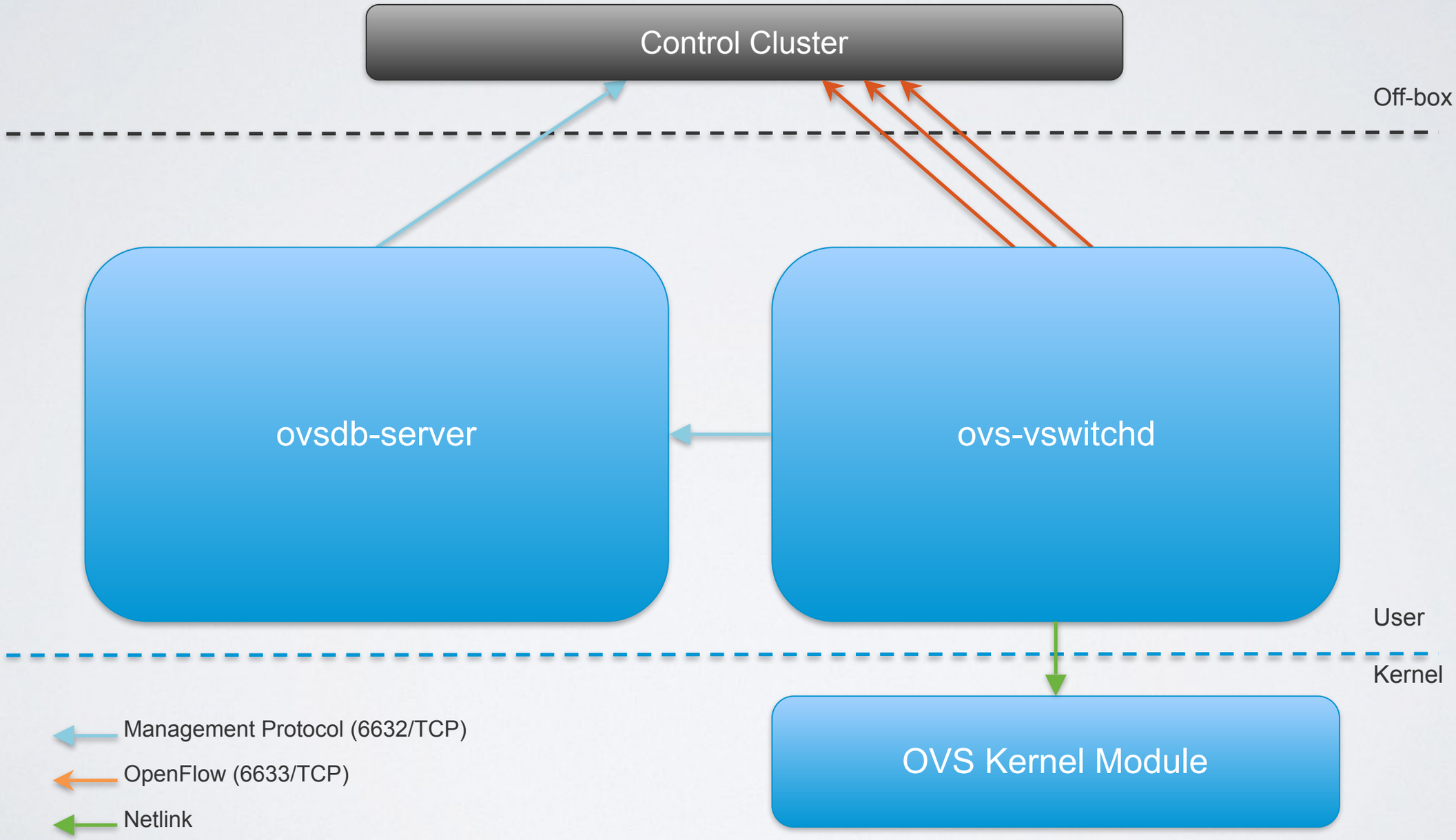
Programmable



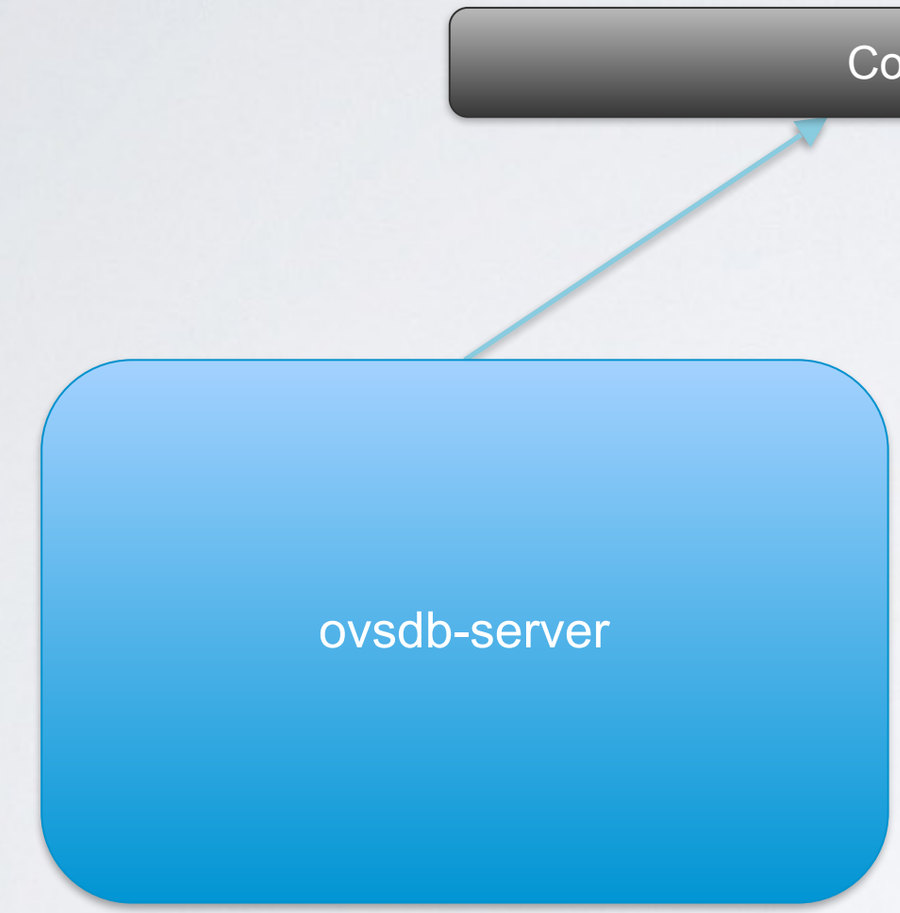
OPEN VSWITCH

- A programmable vSwitch designed for Network Virtualization
  - (And Software Defined Networking)
- Extremely successful
  - Bundled with the Linux Kernel
  - Default switch for KVM deployments

OPEN VSWITCH



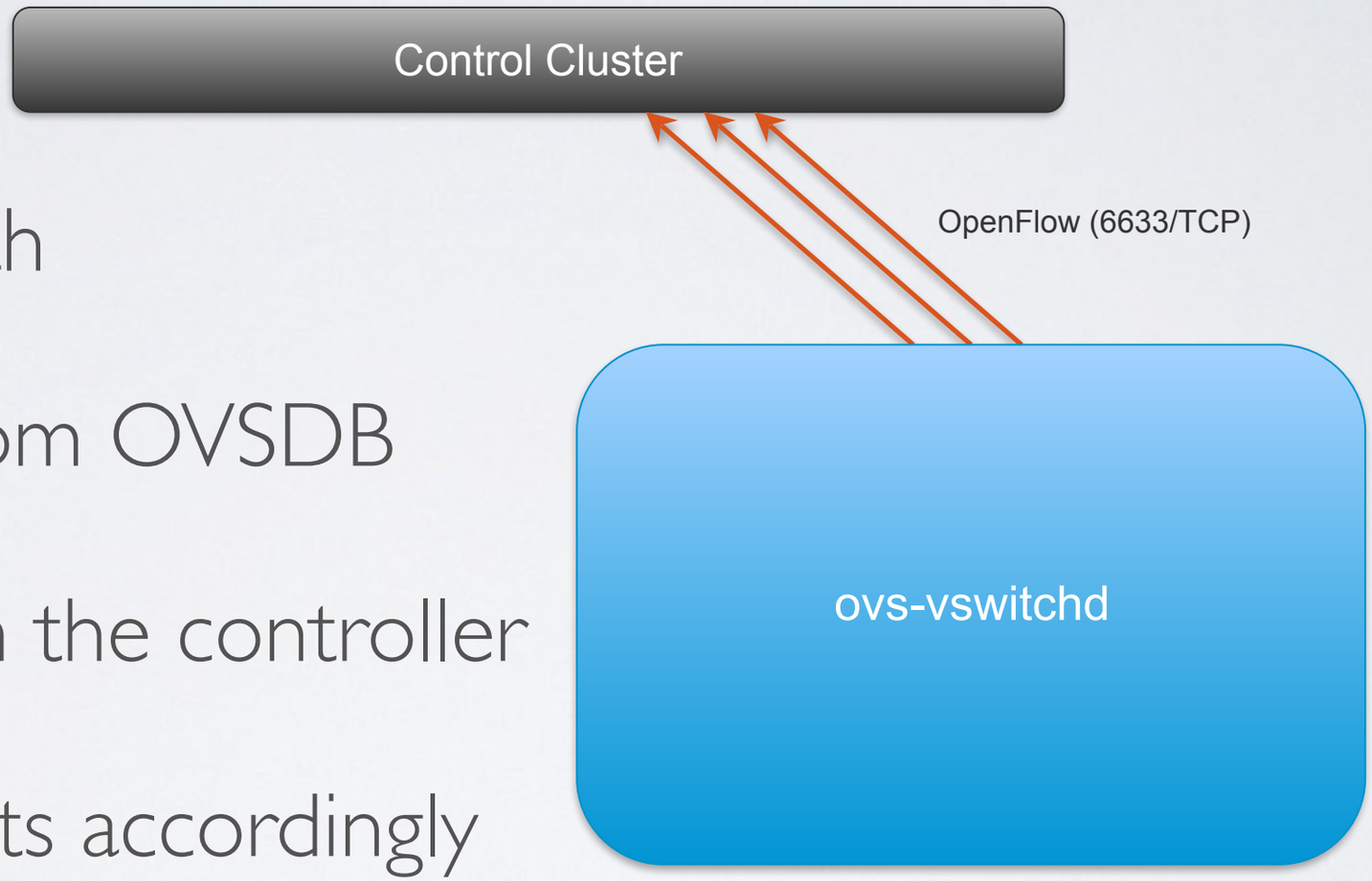
OVSDB



- Transactional database
- Speaks JSON RPC
- Stores slow-changing config
  - Ports
  - Tunnels

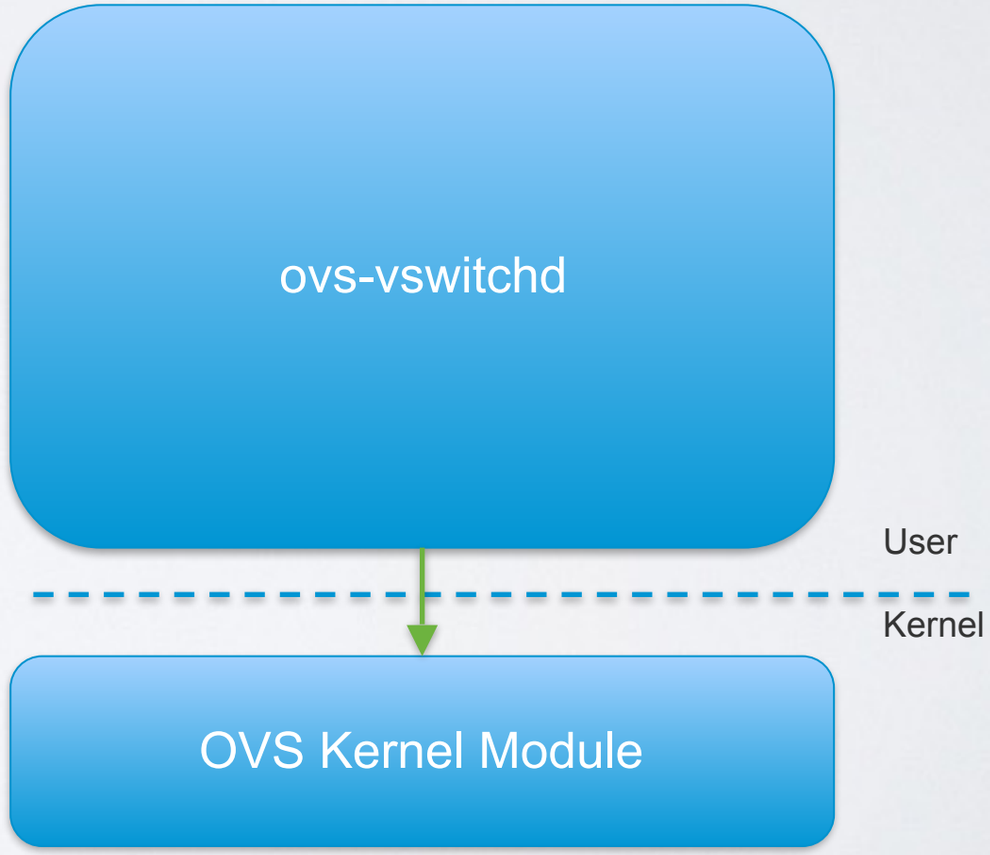
OVS VSWITCHD

- Switch Slow Path
- Reads config from OVSDB
- OpenFlow from the controller
- Forwards packets accordingly

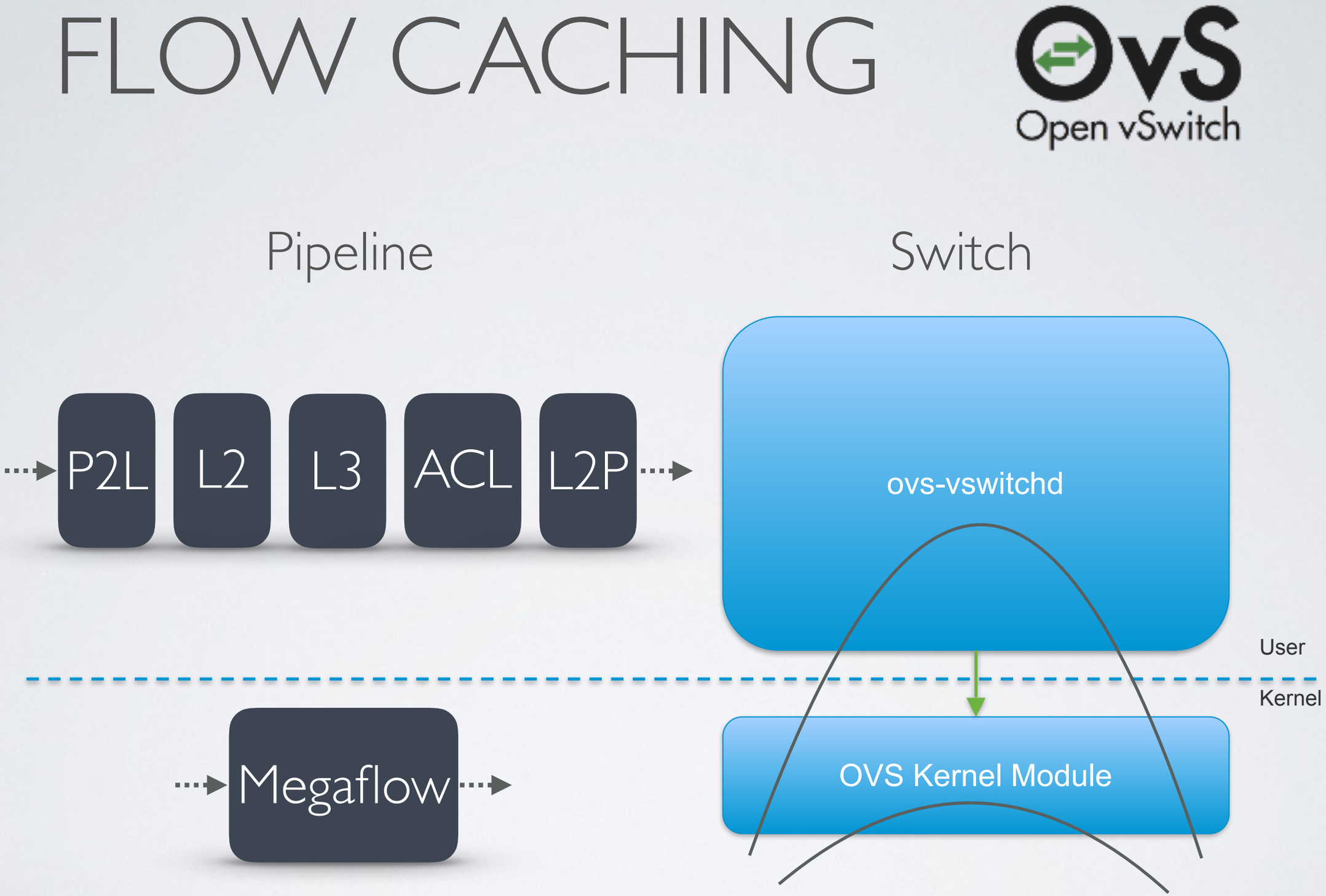


DATAPATH

- Switch fast path
  - Typically implemented in Kernel
  - Receives Packets from the NIC and forwards them
  - Punts them up if confused

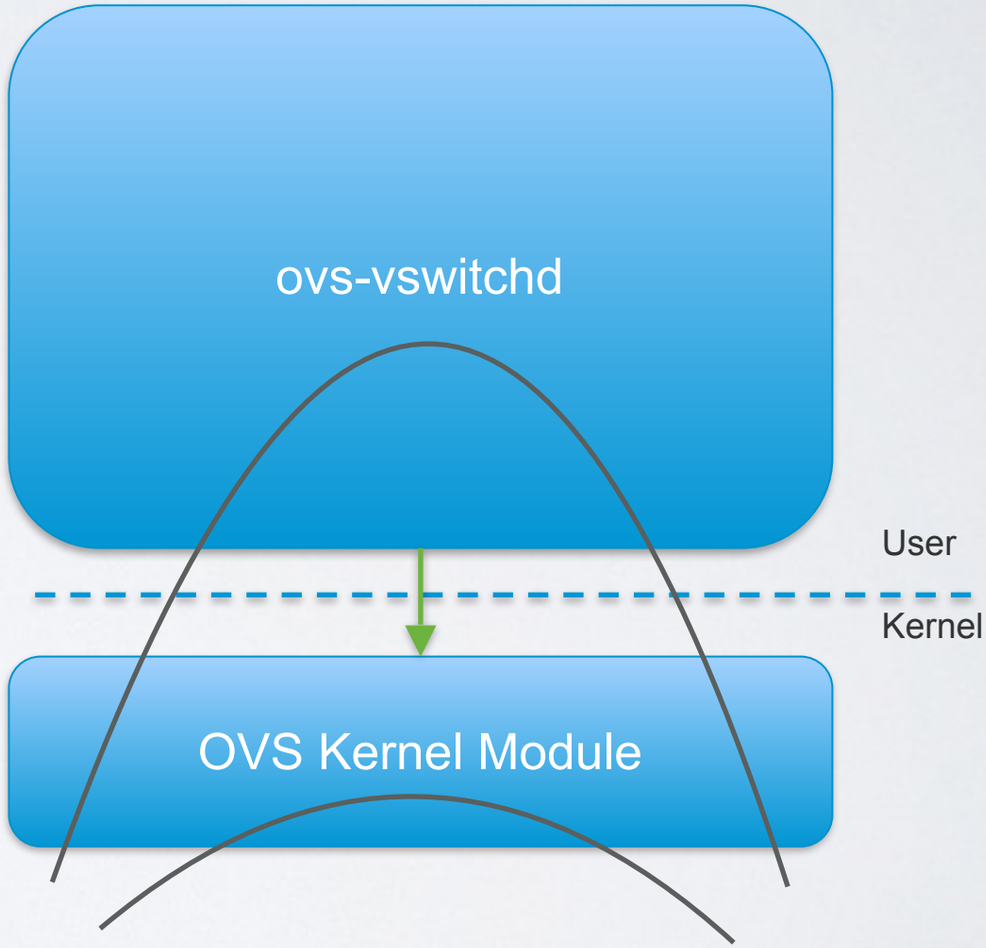


FLOW CACHING



FLOW CACHING

- Complex processing happens in slow path
- Forwarding happens in datapath
- Balances speed …
- And programmability



ARCHITECTURE SUMMARY

- Tunnels
  - Allow physical network to forward packets, ignorant of network policy
- Controller
  - Manages global configuration
- vSwitch
  - Highly programmable dataplane does the heavy lifting

THE FUTURE

MIDDLEBOXES

Network Function Virtualization

- Middleboxes (NATS, Firewalls, IDS, IPS, etc)

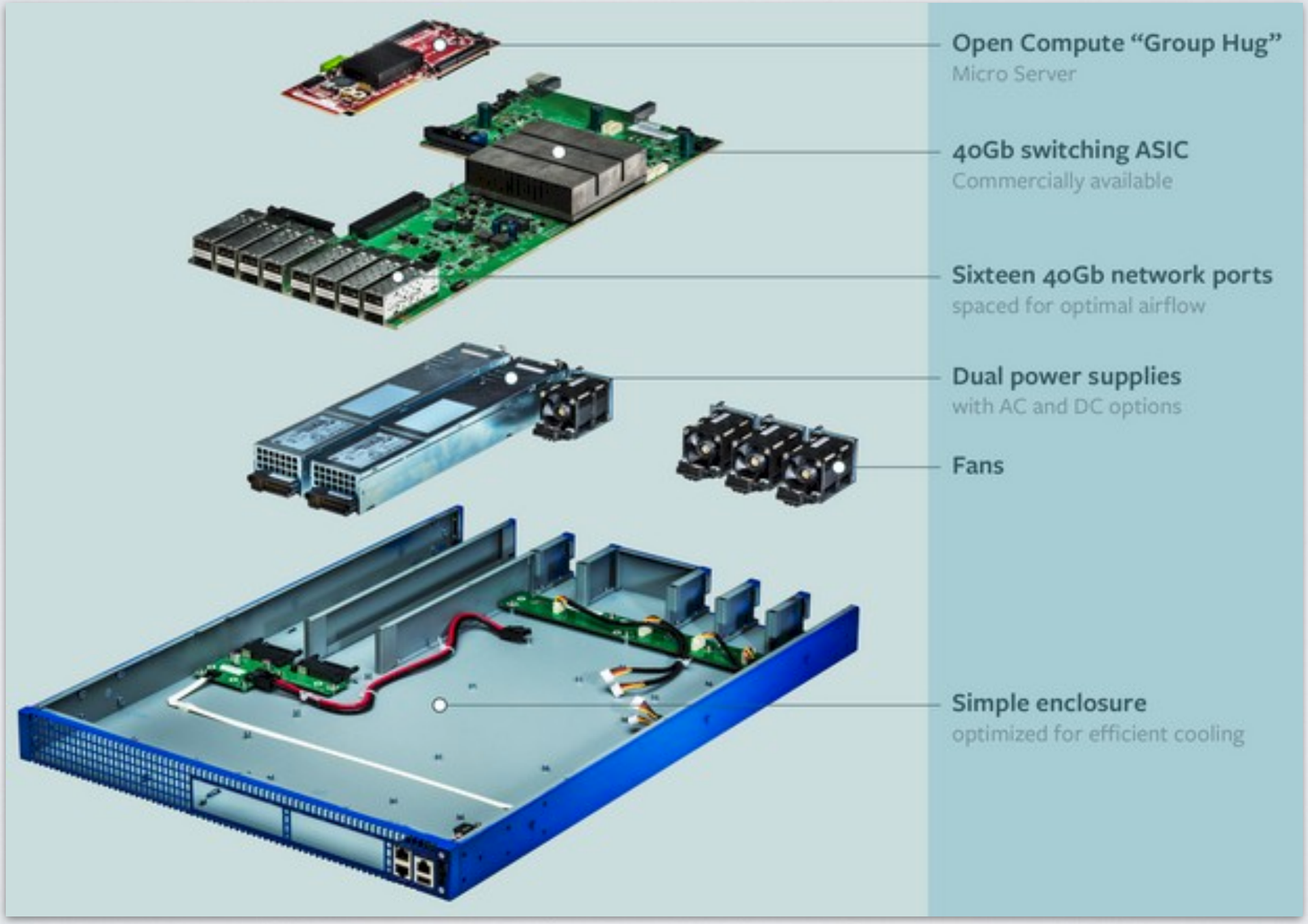
- Fit Badly into the model (state management is hard)

- Network Function Virtualization

  - Shove middleboxes into virtual machines

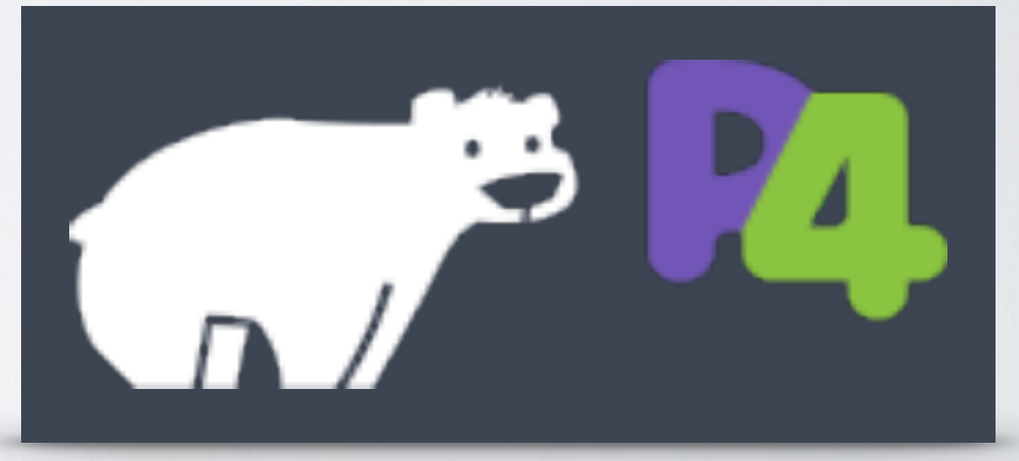
  - Use Netmap/DPDK to make it fast

WHITE-BOX SWITCHING



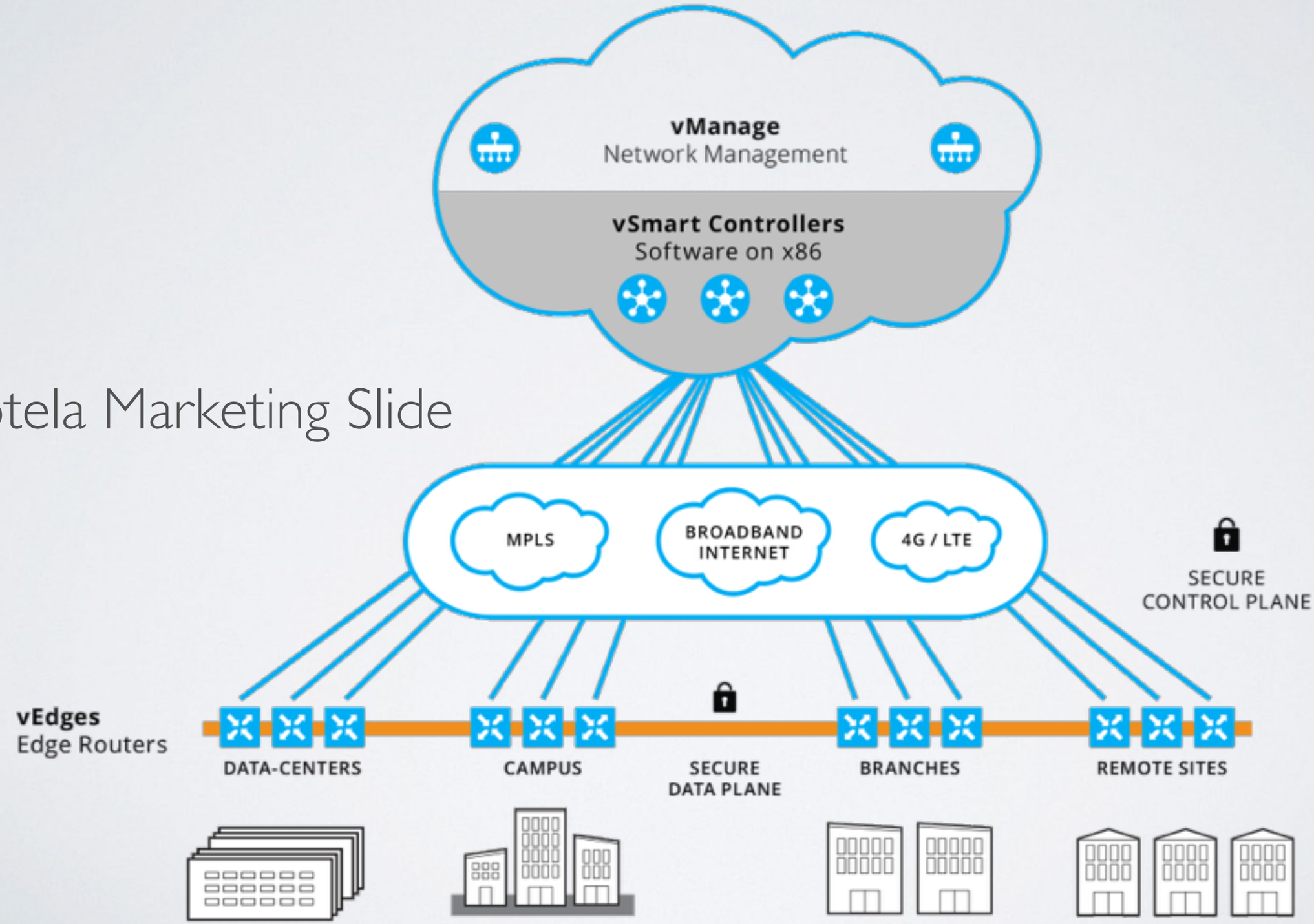
PROGRAMMABLE HARDWARE

- OpenFlow Failed
  - (For hardware)
- ASICS are inflexible
- Enter P4 …



SD-WAN

Viptela Marketing Slide



WILD SPECULATION

SHAMELESS PLUG

- Found this interesting?
- Good Hacker?
- Find me ….
  - ejj@eecs.berkeley.edu
  - ejj.github.io

THANK YOU!