ΕΛΛΗΝΙΚΗ ΔΗΜΟΚΡΑΤΙΑ
ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΡΗΤΗΣ

Δίκτυα Καθοριζόμενα από Λογισμικό

Ενότητα 2.1: Network Virtualization

Ξενοφώντας Δημητρόπουλος
Τμήμα Επιστήμης Υπολογιστών

HY436: Network Virtualization

20/10/2014

Xenofontas Dimitropoulos

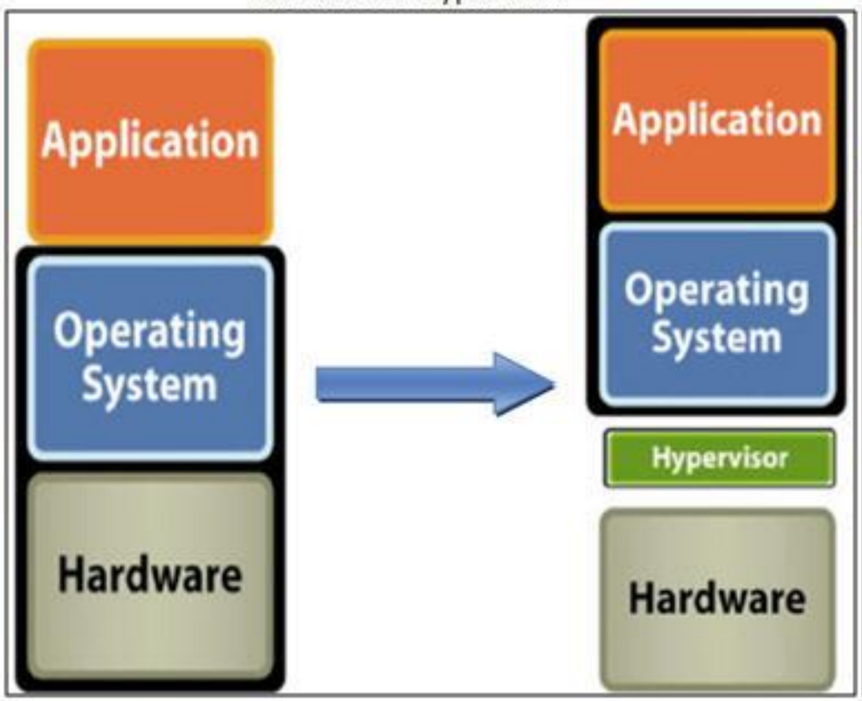
Credits: Bing Wang, Rob Sherwood, Ben Pfaff, Nick Feamster

Agenda

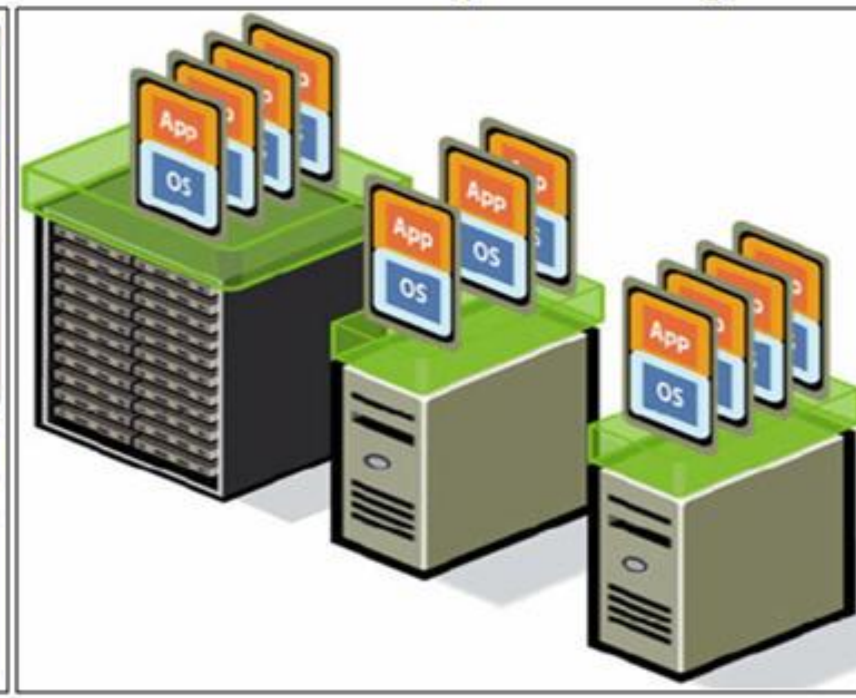
- Network virtualization basics
- Early Forms of Vnets
  - Overlay networks
  - VPNs
- Vnets:
  - External Vnets with FlowVisor/OpenVirteX
  - Internal Vnets with Open vSwitch

From Virtual Operating Systems

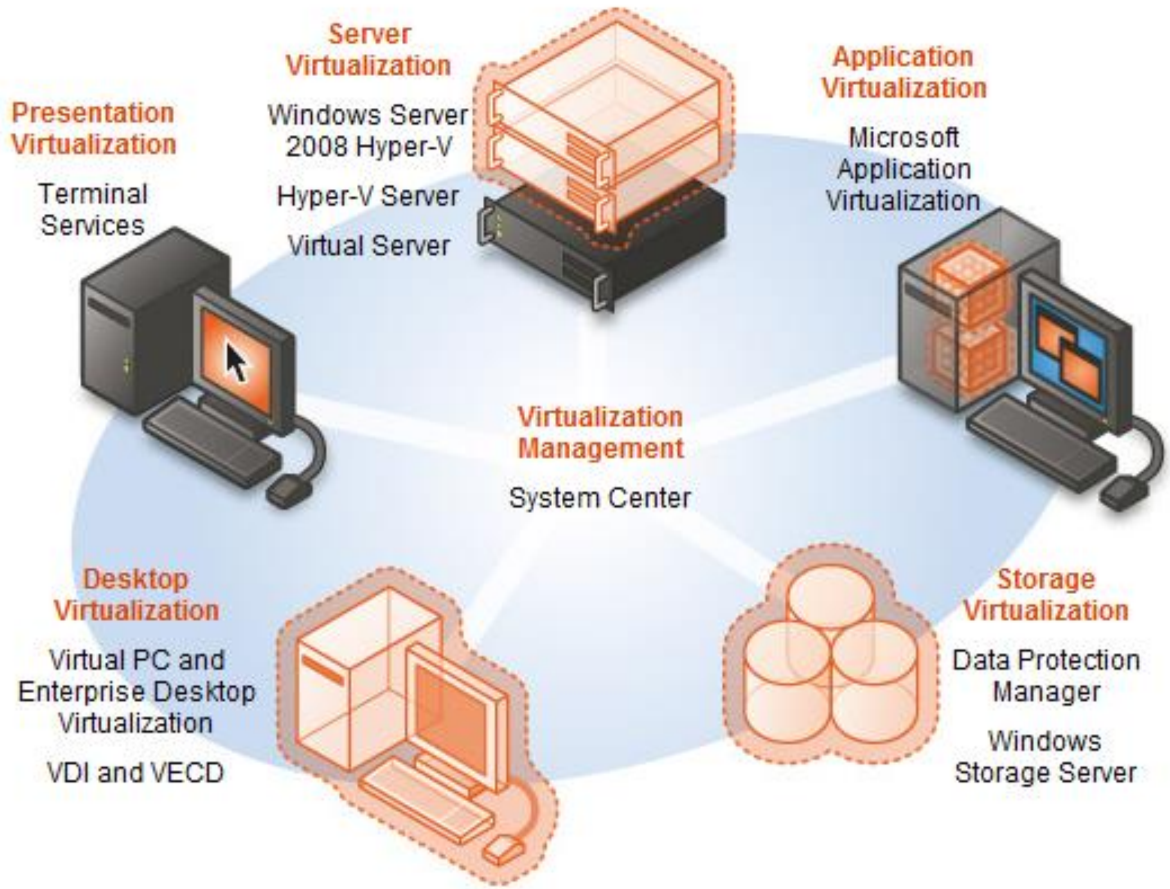
Insertion of hypervisor



Virtual machines run on any hardware configuration



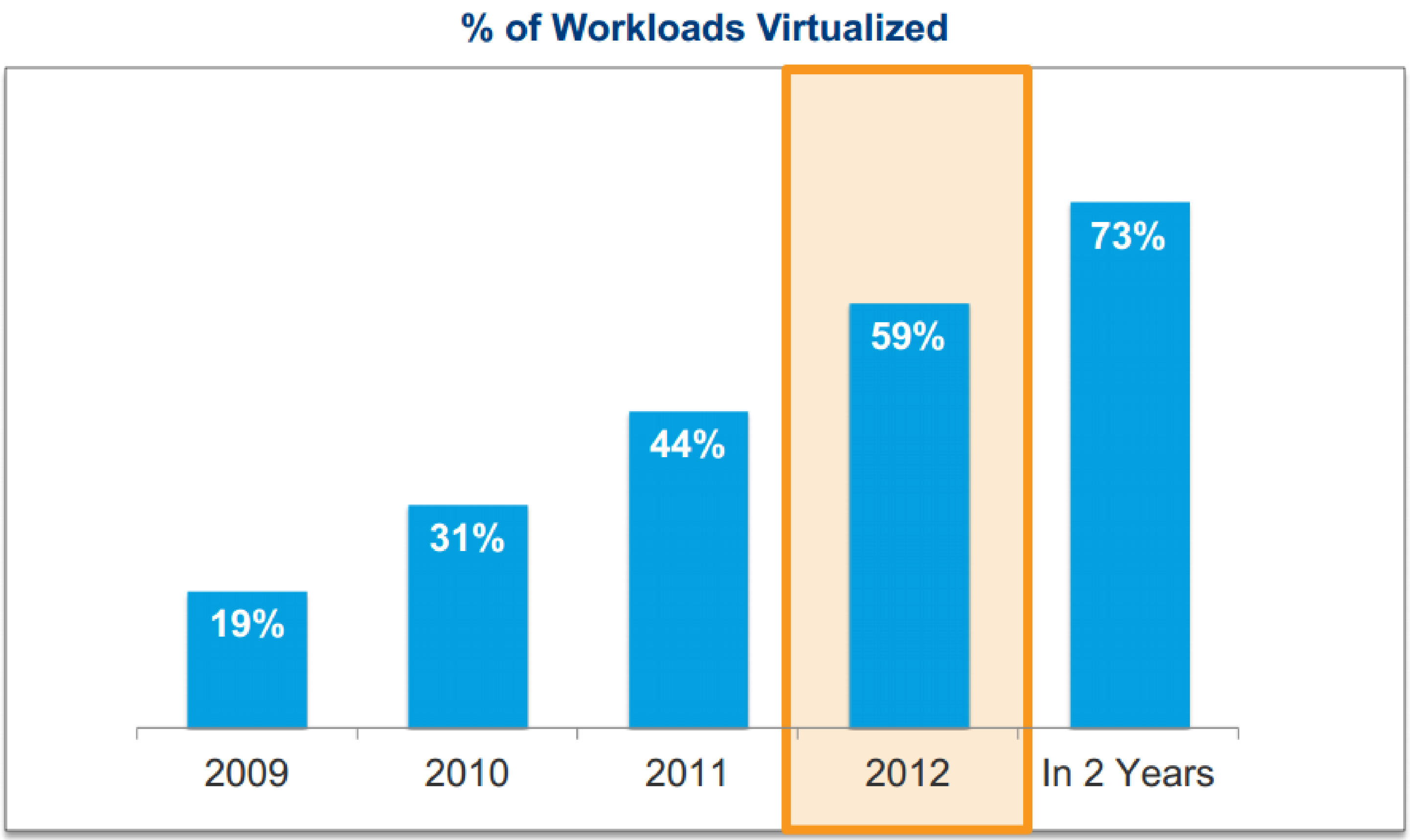
To Virtual Resources (in general)



Example: Microsoft's Virtualization Technologies

Further reading: http://en.wikipedia.org/wiki/Virtualization

The Rise of Virtualization Technologies



Source: VMware customer survey, Jan 2010, Jun 2011, Mar 2012
Question: Please indicate percentage of x86 server operating system instances (e.g., Windows, Linux) that run in virtual machines

Increase in adoption of virtualization technologies in the enterprise

What is network virtualization?

- Decoupling of the services provided by a (virtualized) network from the physical network
- Virtual network is a container of network services (L2-L7) provisioned by software
- Faithful reproduction of services provided by physical network

Credit for definition: Bruce Davie, VMware Principal Engineer

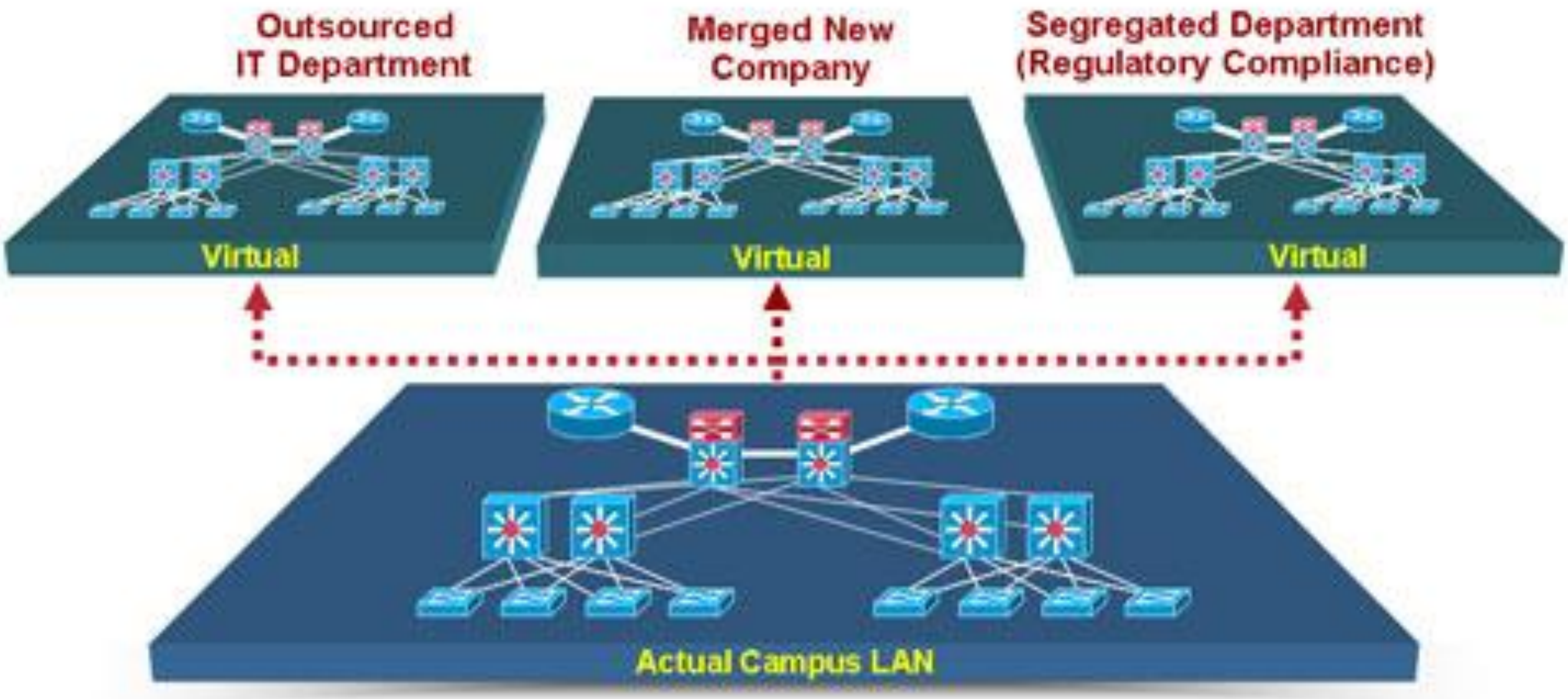
Types of Network Virtualization

- External network virtualization
  - Segment a physical network into multiple vnets
  - Combine many physical nets into a virtual unit

- Internal network virtualization
  - Providing network-like functionality within a system

External Net Virtualization

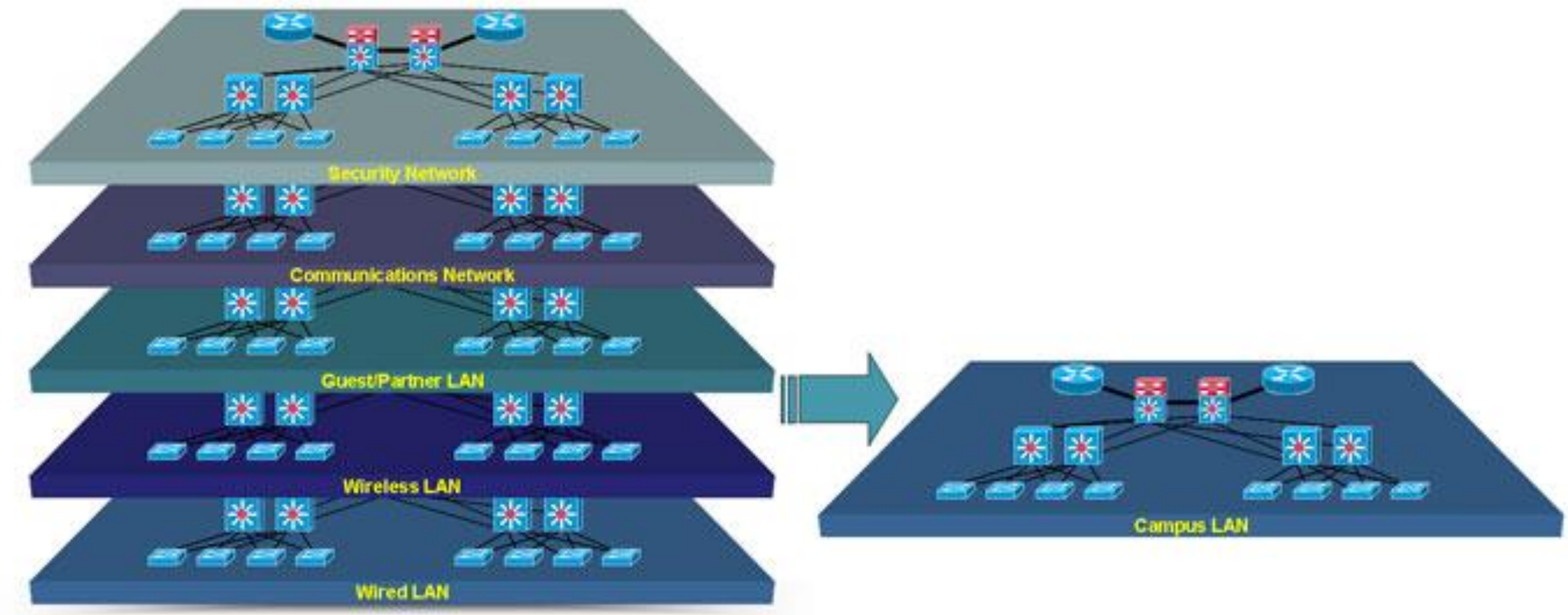
Configure systems physically attached to the same local network into separate virtual networks



Source: Cisco Net Virtualization Solutions

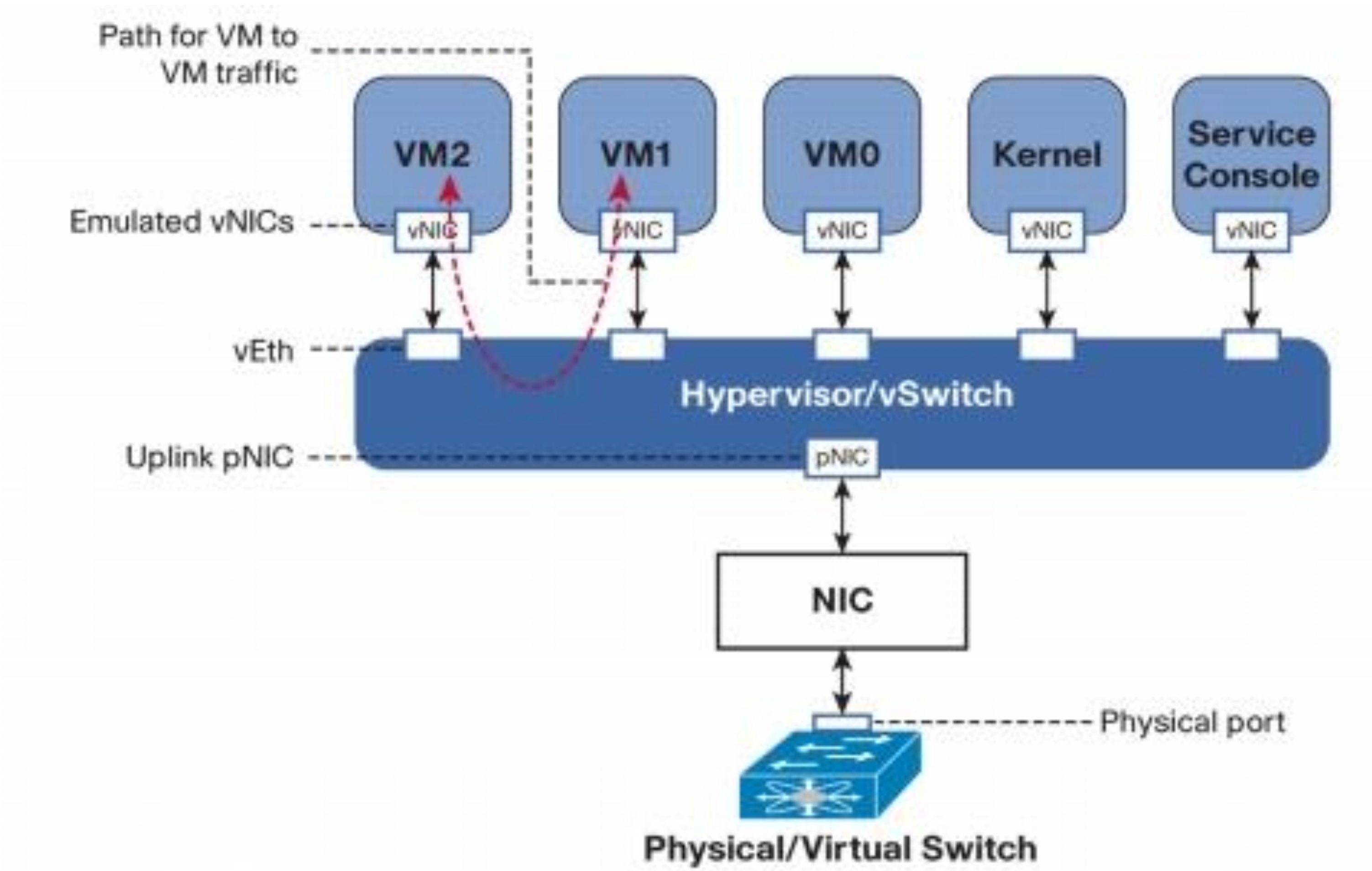
External Net Virtualization

Combine systems on separate local networks into a VLAN spanning the segments of a large network



Source: Cisco Net Virtualization Solutions

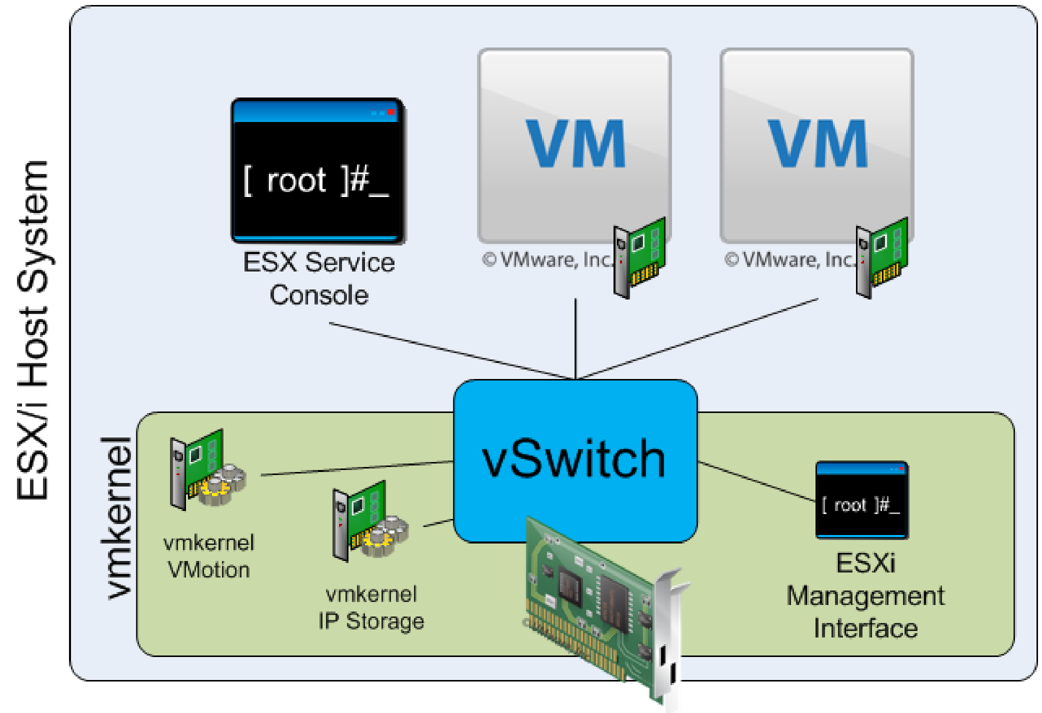
Internal Net Virtualization



Source: Cisco Virtual Interface Cards

Virtual Switches

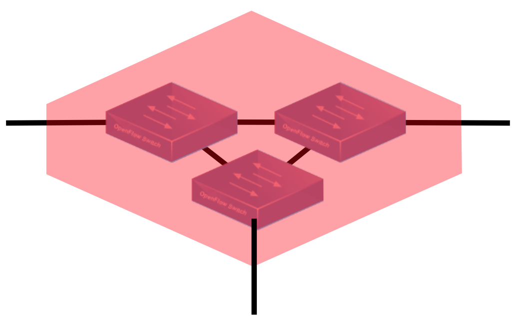
- Work much like physical Ethernet switches
- Detect VMs connected to virtual ports
- Forward traffic to the correct virtual ports
- Uses x86, not ASICs



VMware's vSwitch Overview

Vnets enable abstract topologies

- Applications see abstract topology, which may differ than the physical topology
- Common example “one big switch” topology:



- Promise: simplified programming and operations

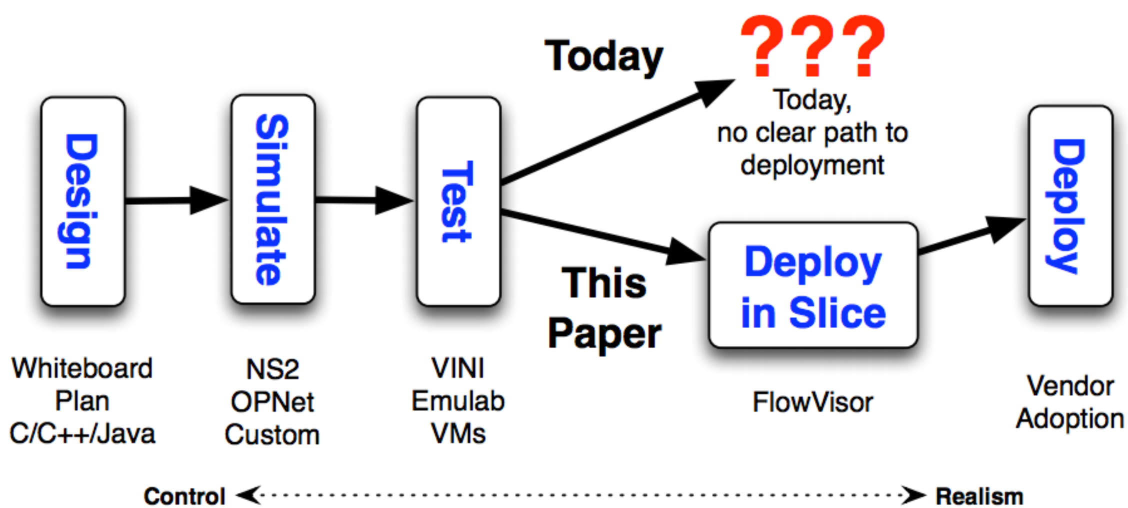
What led to Net Virtualization?

- **Path A**: Internet "ossification"
  - Mostly the path of the research community

- **Path B**: Natural extension of cloud computing to the network
  - Mostly the path of the industry

Internet Ossification

- Very difficult to experiment on real networks with new technologies for IP, routing, etc.
- Experimentation approaches:



Further reading: R. Sherwood et al. "Can the production network be the testbed?" OSDI 2010

Promise of Net Virtualization

- Rapid network innovation:
  - Network services delivered at software speed
  - New forms of network control
- Isolation allows experimental vnets deployments
- Vendor choice (hardware/software from different vendors)
- Simplified programming

Promise of (Net) Virtualization

- Re-use resources for multiple vnets
  - Reduce hardware costs
  - Increase resource utilization
  - Decrease energy costs
  - Dynamic resource scaling
- Fault and disaster recovery, i.e., decouple software from hardware faults
- Easier management of “logical” resources

→ Much like **cloud computing**

Vnets Design Goals?

- **Flexibility**: different topologies, routing and forwarding architectures; independent configuration
- **Manageability**: provide high-level abstractions
- **Scalability**: maximize the number of vnets that can coexist
- **Isolation**: Isolate vnets and resources
- **Heterogeneity**: support for different technologies

Further reading: Nick Feamster’s lecture
http://youtu.be/G1lCF5VALsc?list=PLpherdrLyny-OTgZzlLTcbMIDtLdNuXcT

Virtual Networks vs. SDN

- SDN separates data from control plane and "centralizes" control
- Virtual networks separate logical from physical networks
- SDN helps virtualize a network, but network virtualization predates SDN

Further reading: http://networkheresy.com/2013/04/29/netvirt-delivering/

Agenda

- This lecture:
  - Early Types of Vnets
  - External Vnets with FlowVisor
  - Internal Vnets with Open vSwtich

- "SDN in the Cloud" lecture:
  - Data center networking basics
  - Vnet applications in the cloud
  - Other SDN apps in the cloud

Some Early Types of Vnets

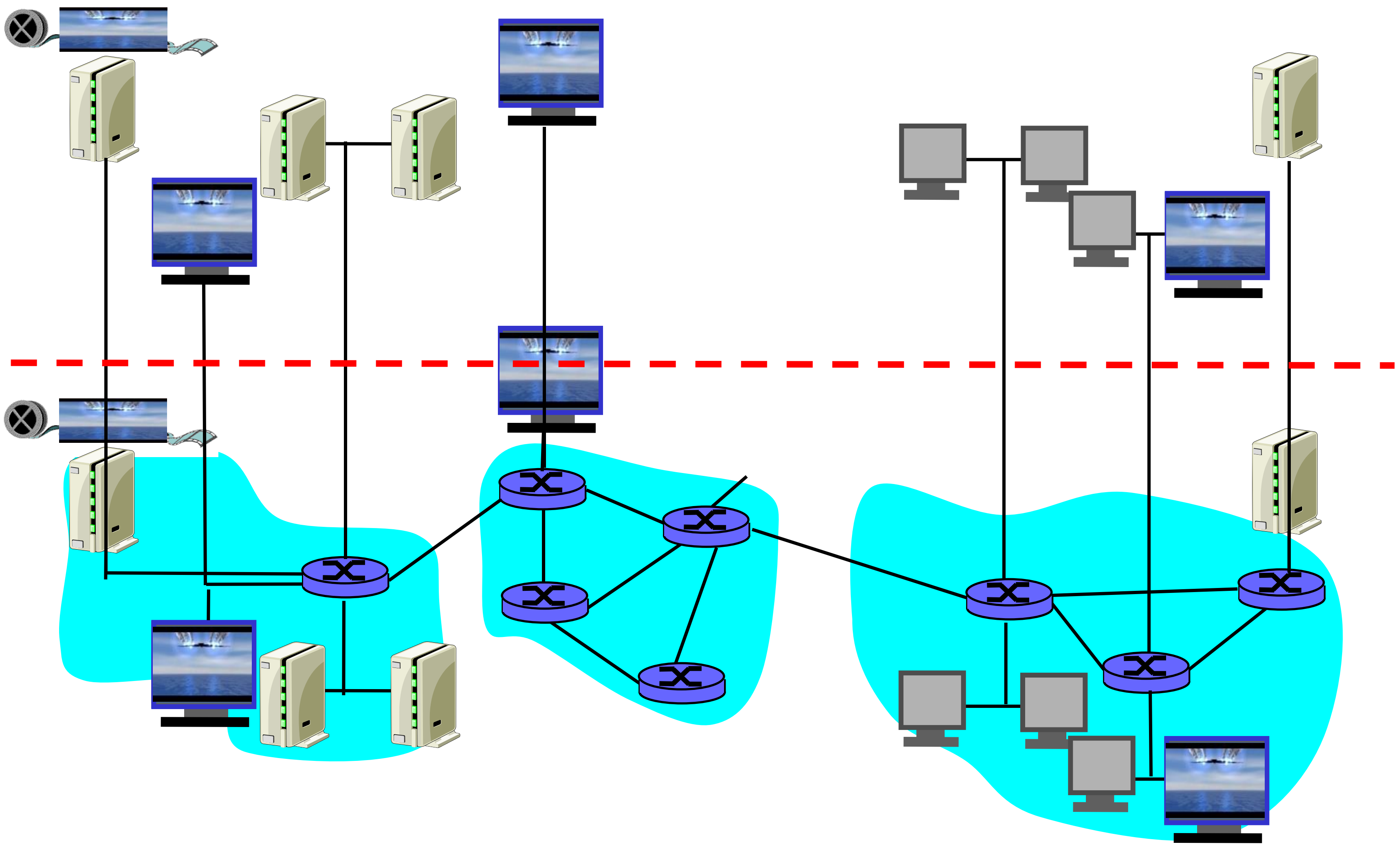
- Overlay and p2p networks
- Virtual Private Networks (VPN) provide remote access to company's network
- Group remote computers in the same Virtual Local Area Network (VLAN) (2nd lecture)

➔**They are also Vnets, but were designed for different goals**

Overlay Networks

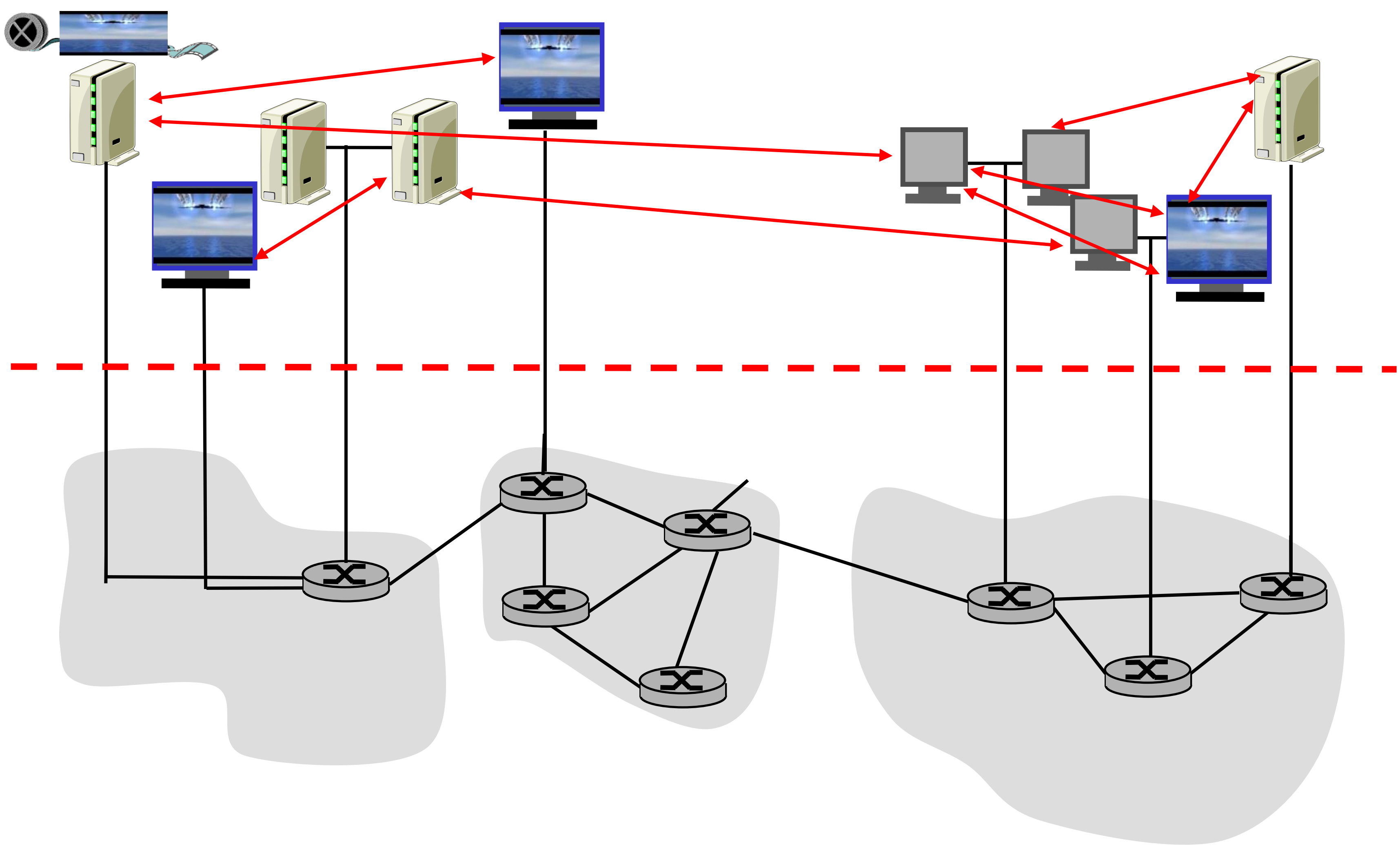
- applications, running at various sites as "nodes" on an application-level network
- create "logical" links (e.g., TCP or UDP connections) pairwise between each other
- each logical link: multiple physical links, routing defined by native Internet routing

Overlay network



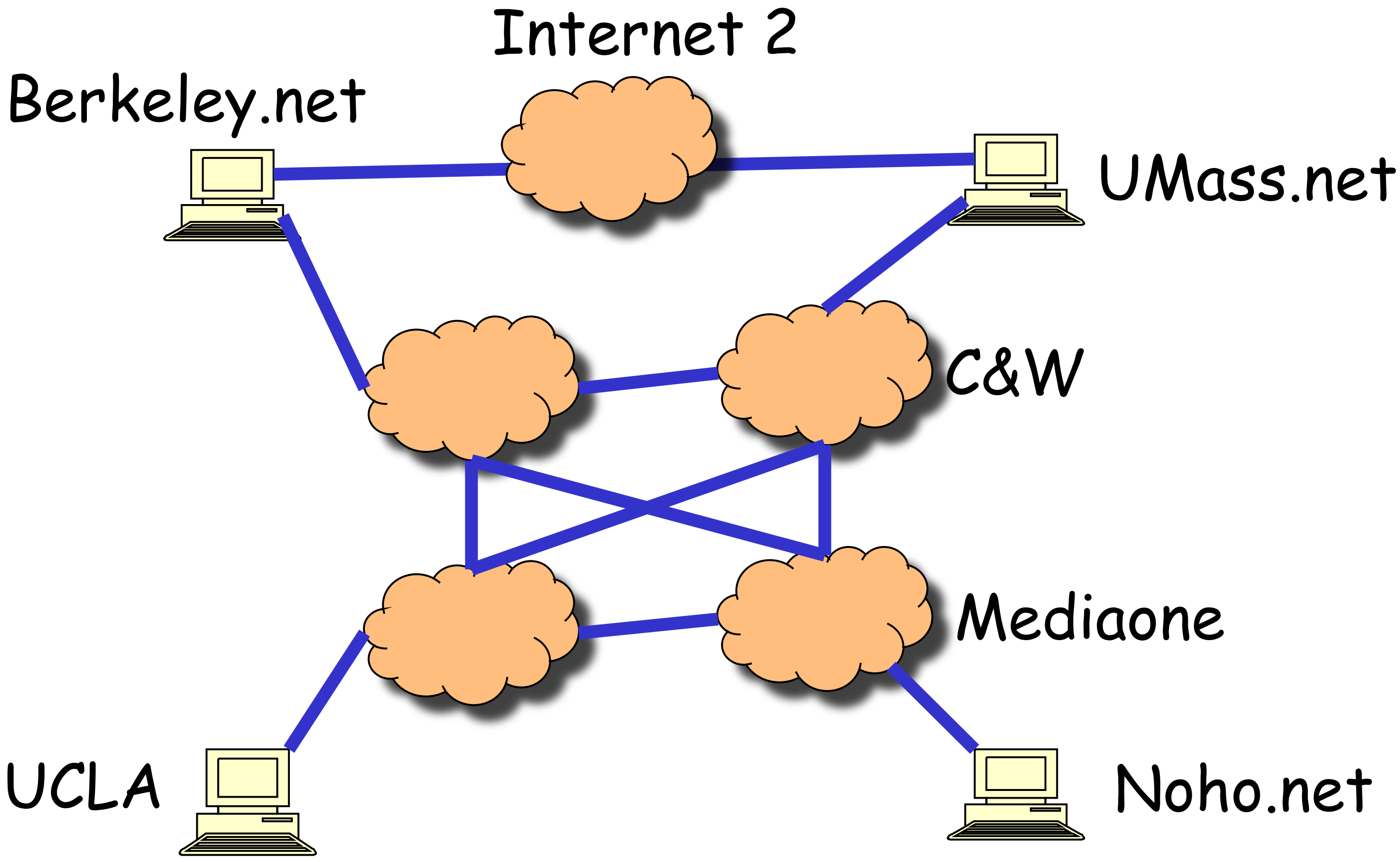
Overlay network

Focus at the application level



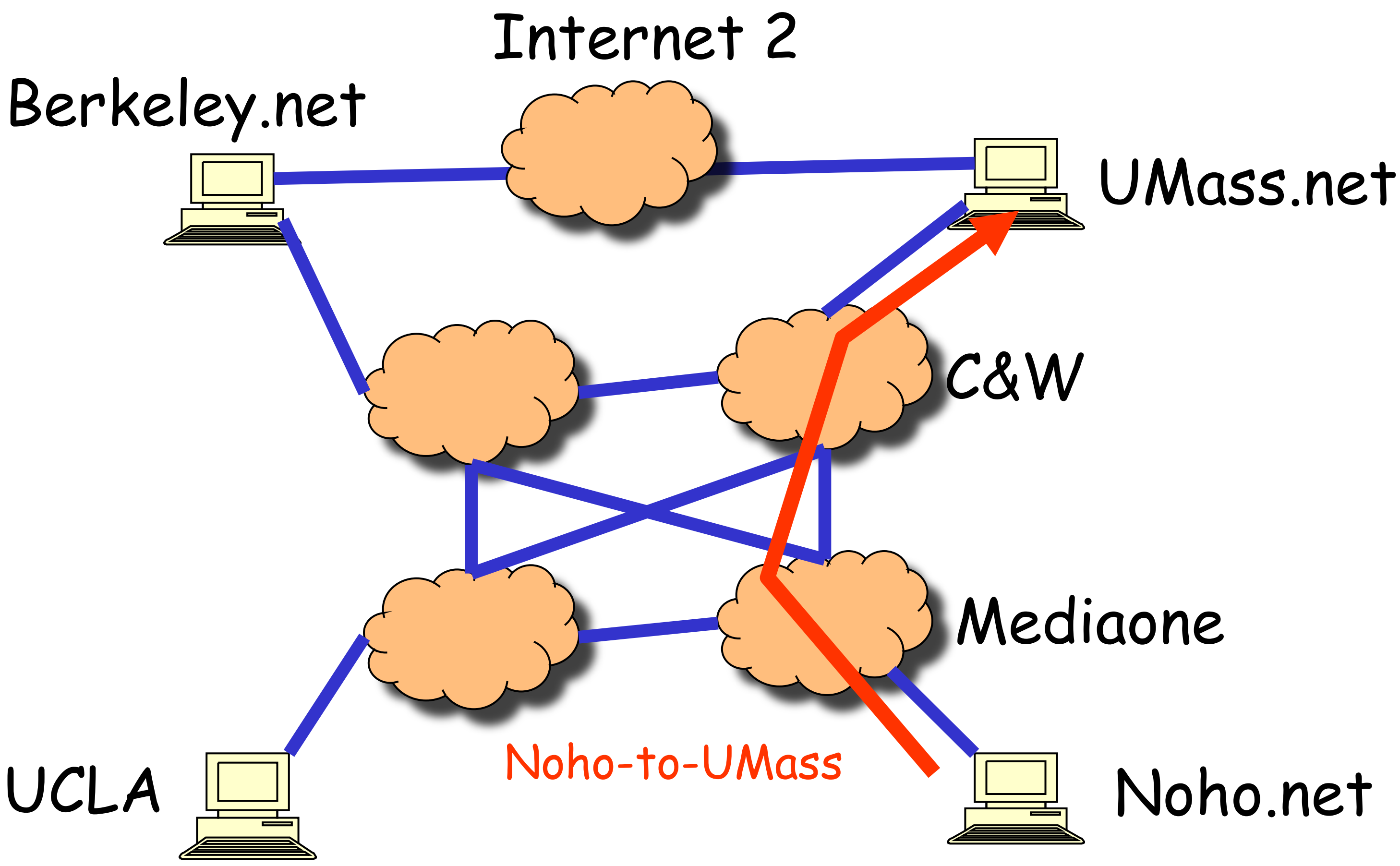
Internet Routing

- BGP defines routes between stub networks



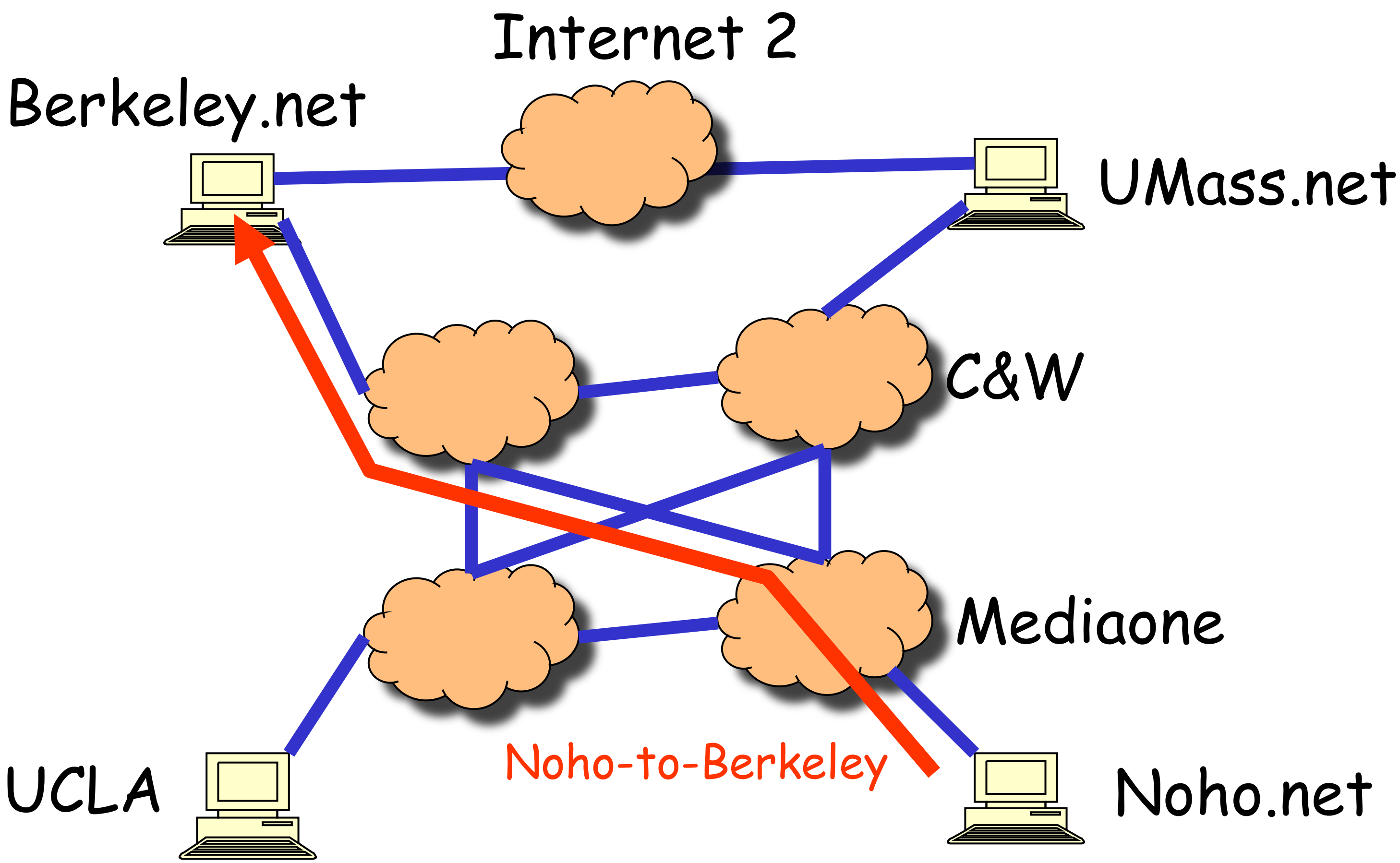
Internet Routing

- BGP defines routes between stub networks

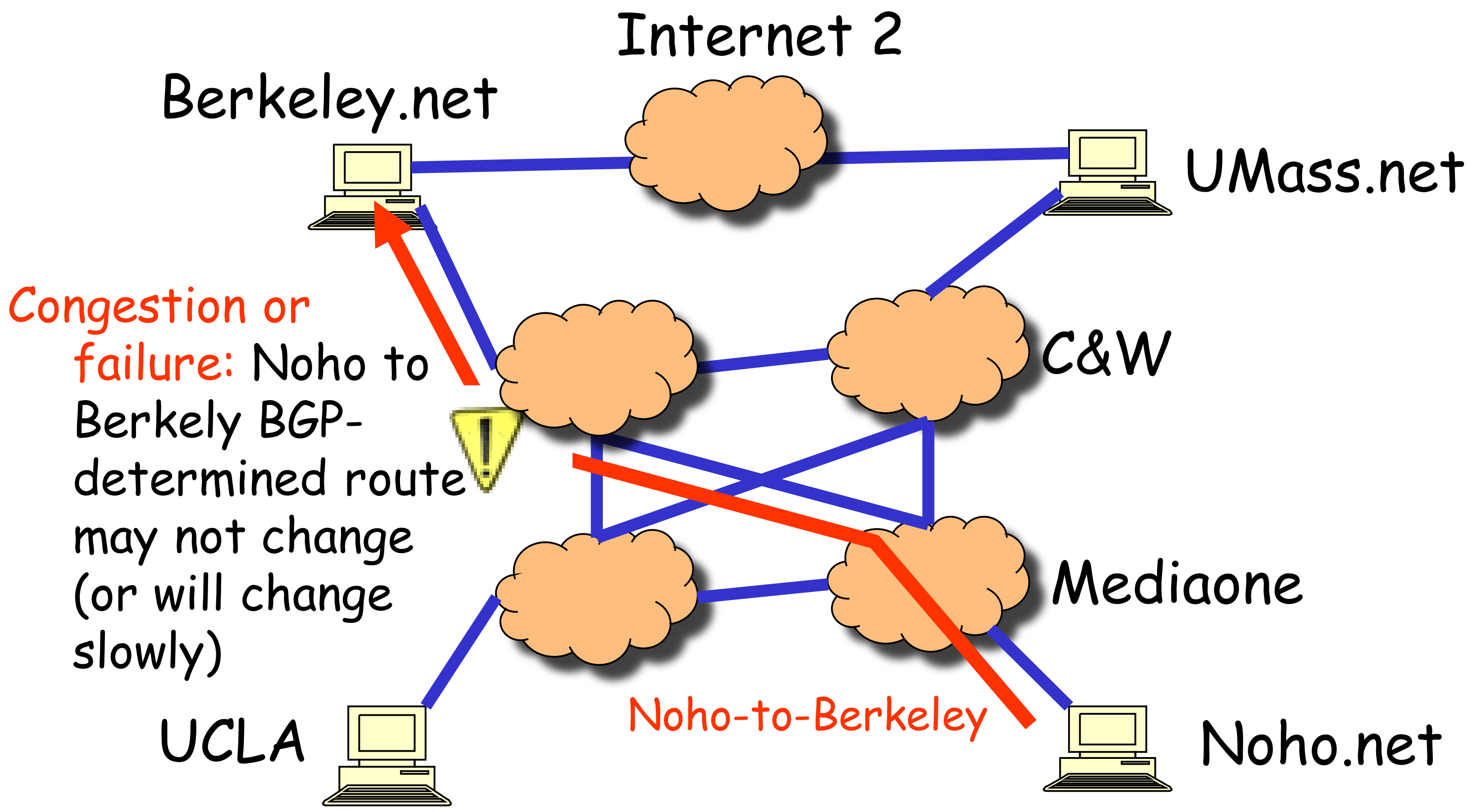


Internet Routing

- BGP defines routes between stub networks

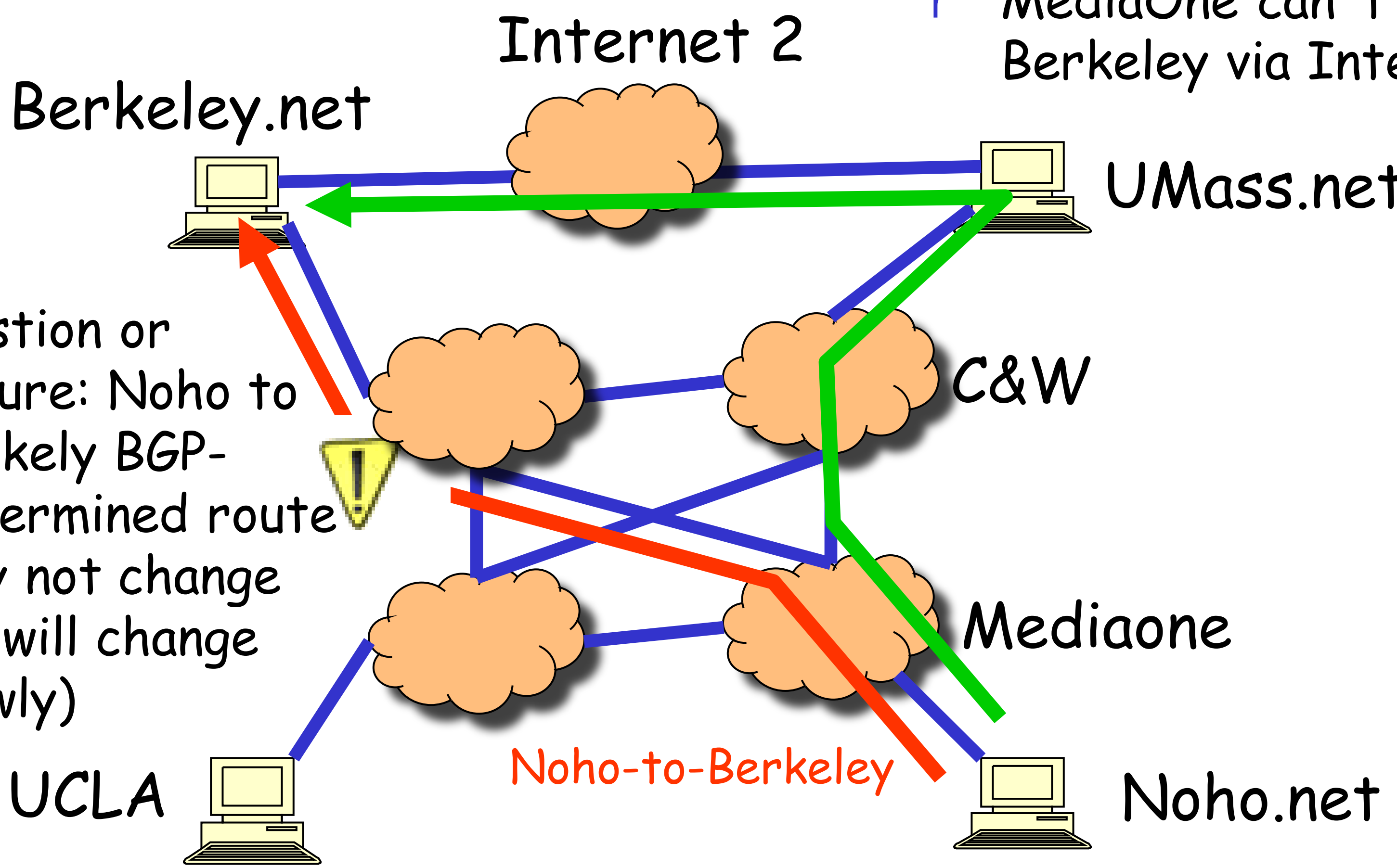


Internet Routing



Internet Routing

- route not visible or available via BGP!
- MediaOne can't route to Berkeley via Internet2

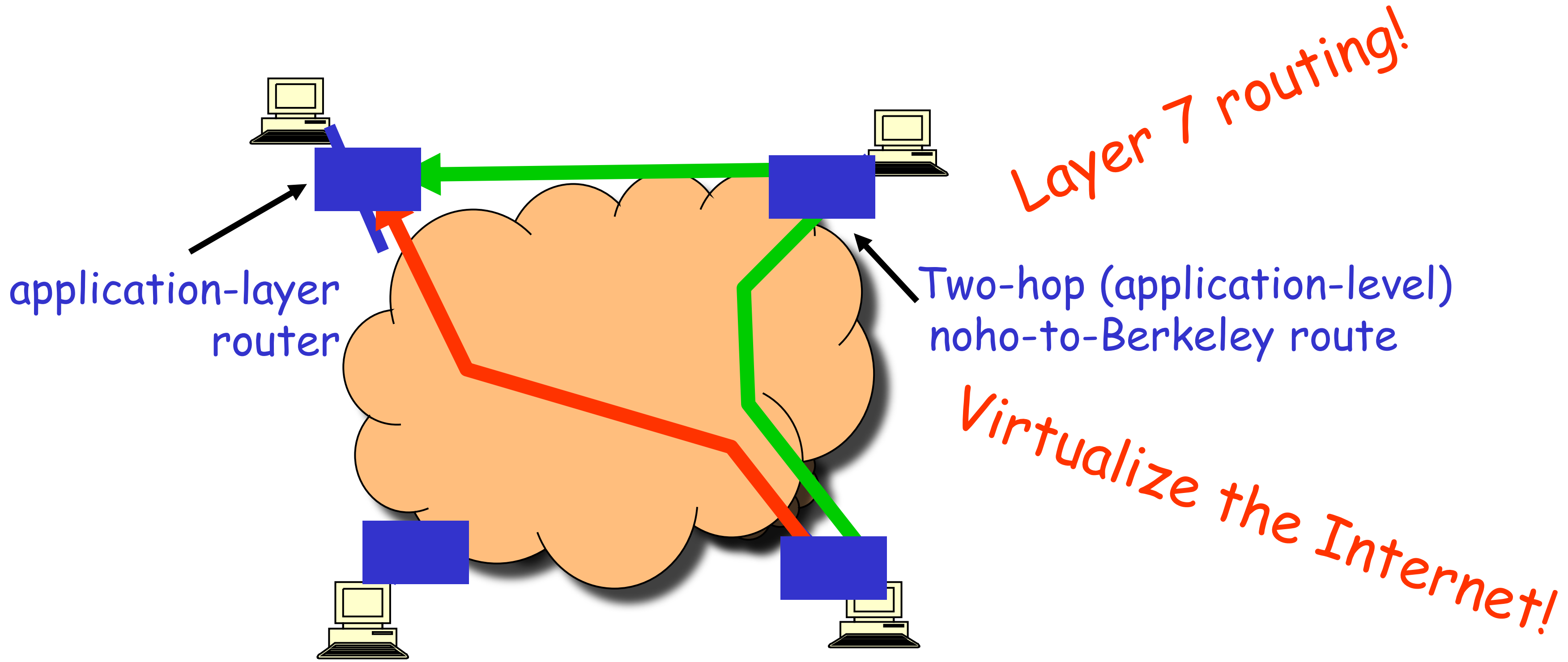


Congestion or failure: Noho to Berkely BGP-determined route may not change (or will change slowly)

RON: Resilient Overlay Networks

Further reading: http://nms.csail.mit.edu/ron/

Premise: by building application overlay network, can increase performance, reliability of routing



RON Experiments

- measure loss, latency, and throughput with and without RON
- 13 hosts in the US and Europe
- 3 days of measurements from data collected in March 2001
- 30-minute average loss rates
  - A 30 minute outage is very serious!

An order-of-magnitude fewer failures

30-minute average loss rates

Loss Rate	RON Better	No Change	RON Worse
10%	479	57	47
20%	127	4	15
30%	32	0	0
50%	20	0	0
80%	14	0	0
100%	10	0	0

6,825 “path hours” represented here
12 “path hours” of essentially complete outage
76 “path hours” of TCP outage
RON routed around all of these!
One indirection hop provides almost all the benefit!

RON Research Issues

- how to design overlay networks?
  - Measurement and self-configuration
  - Fast fail-over
  - Sophisticated metrics
  - application-sensitive (e.g., delay versus throughput) path selection
- effect of RON on underlying network
  - If everyone does RON, are we better off?
  - Interacting levels of control (network- and application-layer routing

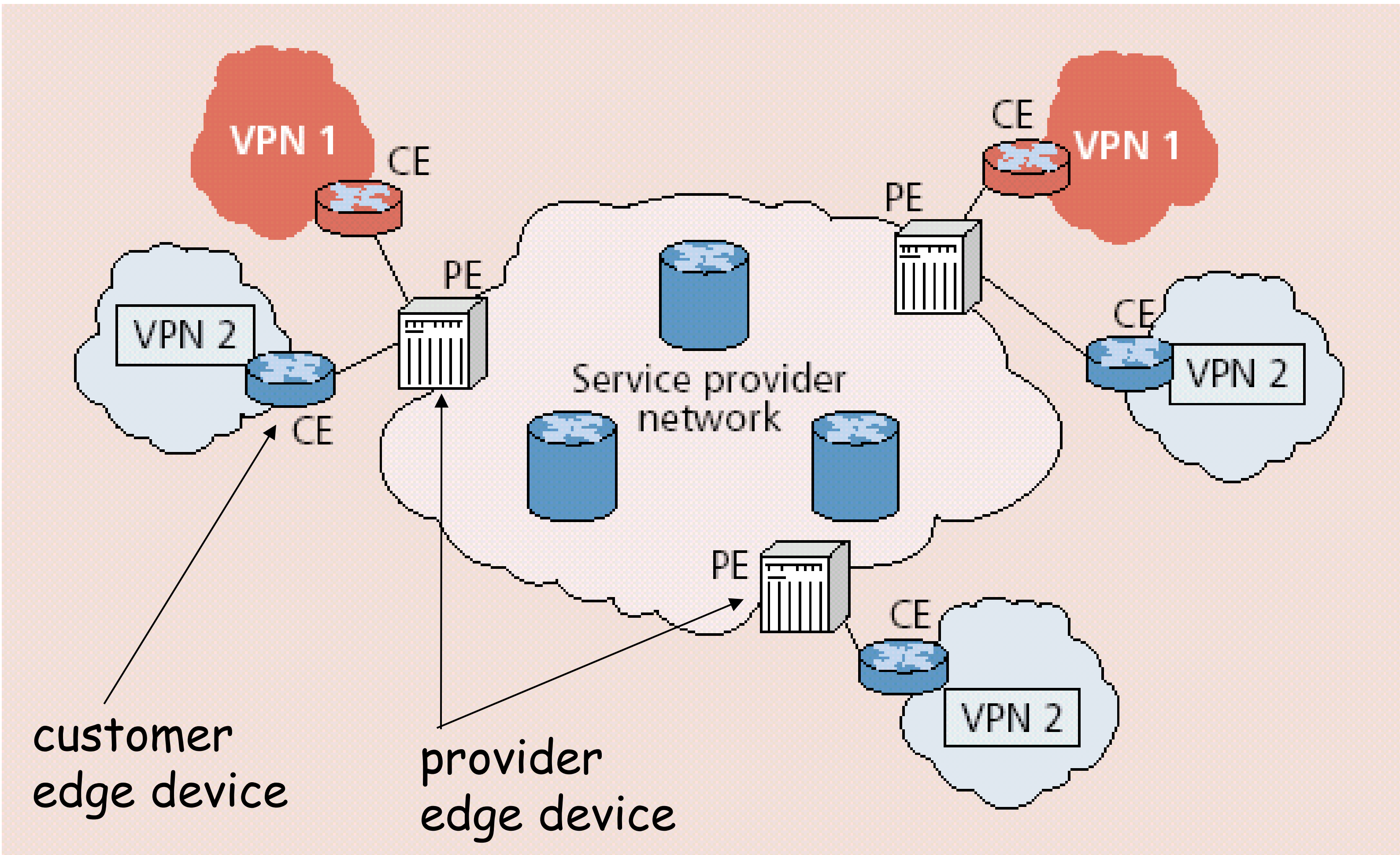
Virtual Private Networks (VPN)

VPNs

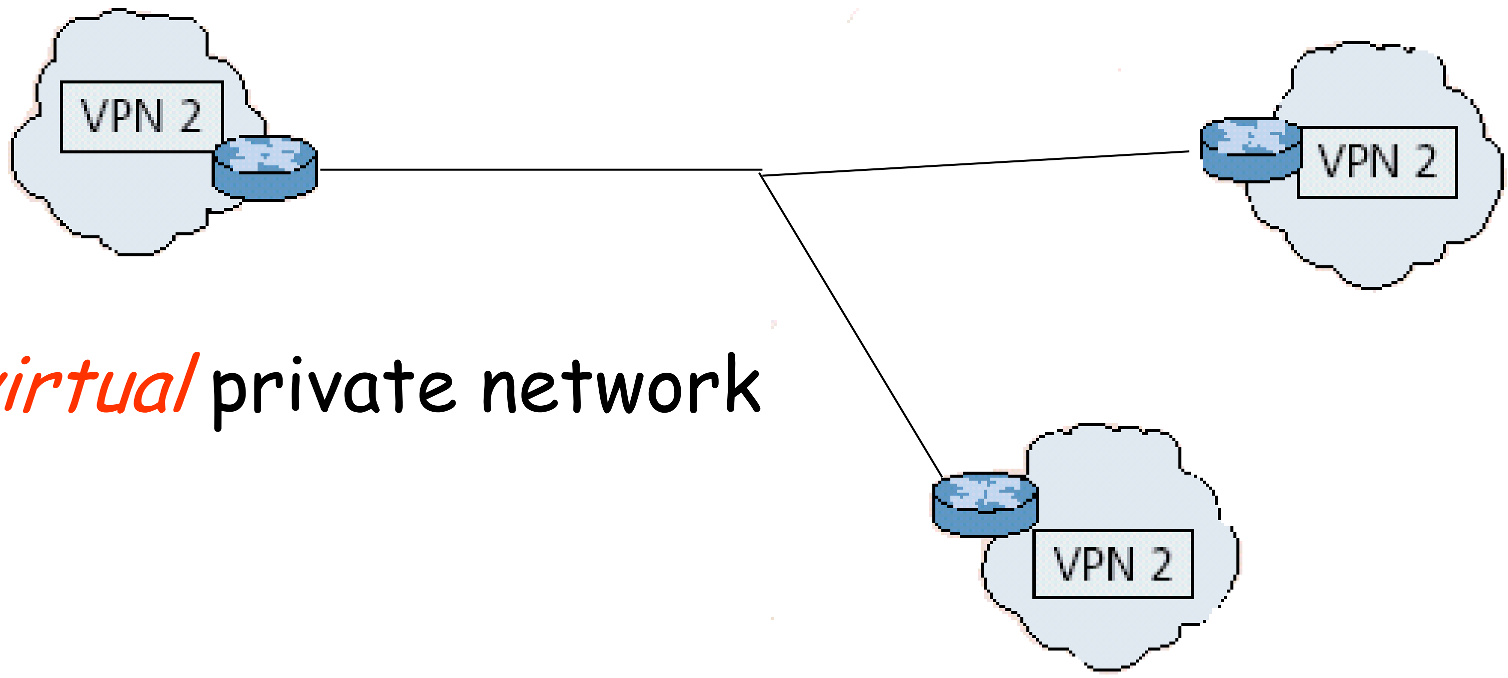
Networks perceived as being private networks by customers using them, but built over shared infrastructure owned by service provider (SP)

- SP infrastructure:
  - backbone
  - provider edge devices
- Customer:
  - customer edge devices (communicating over shared backbone)

VPN reference architecture

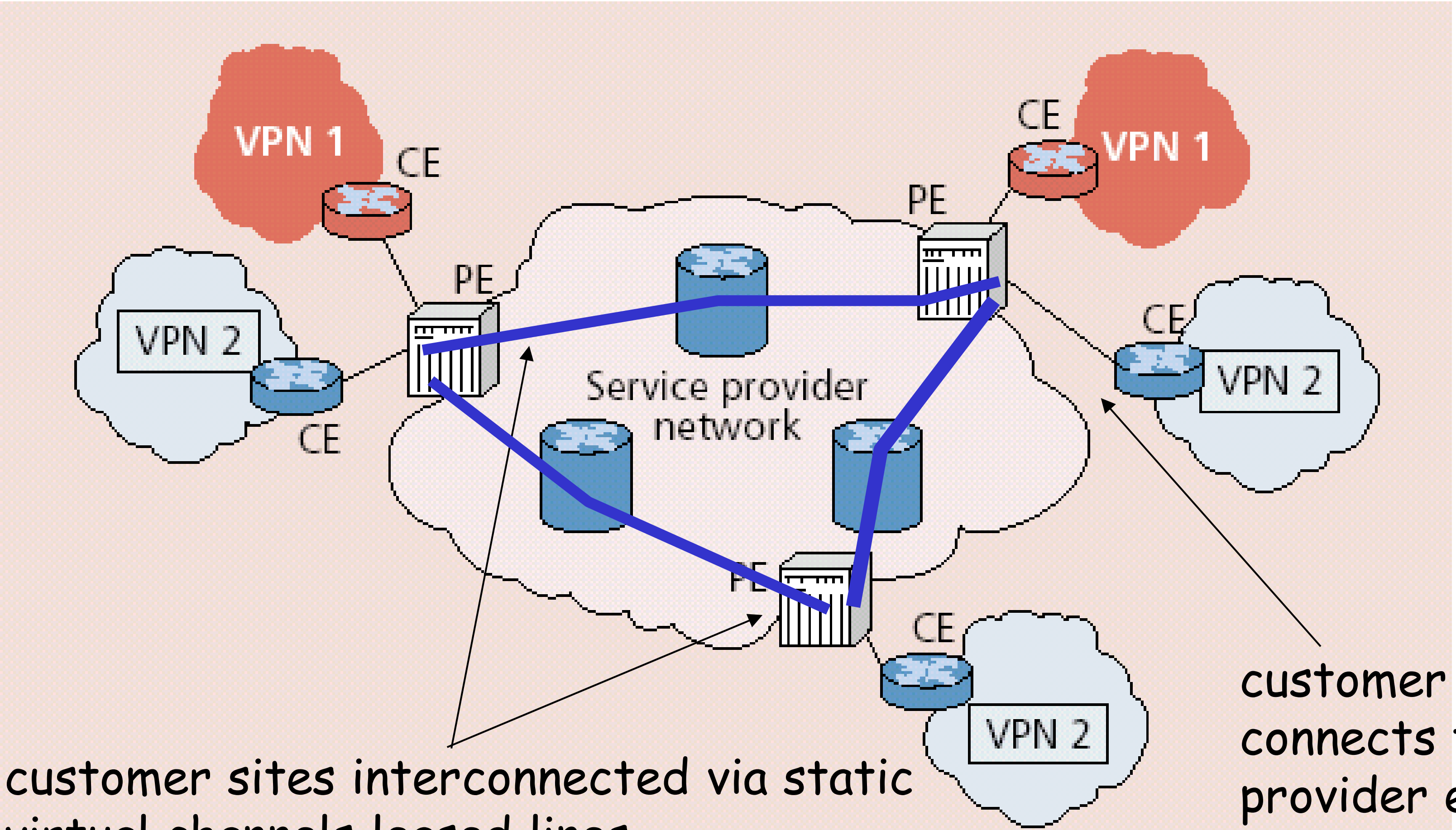


VPN: logical view



virtual private network

Leased-line VPN

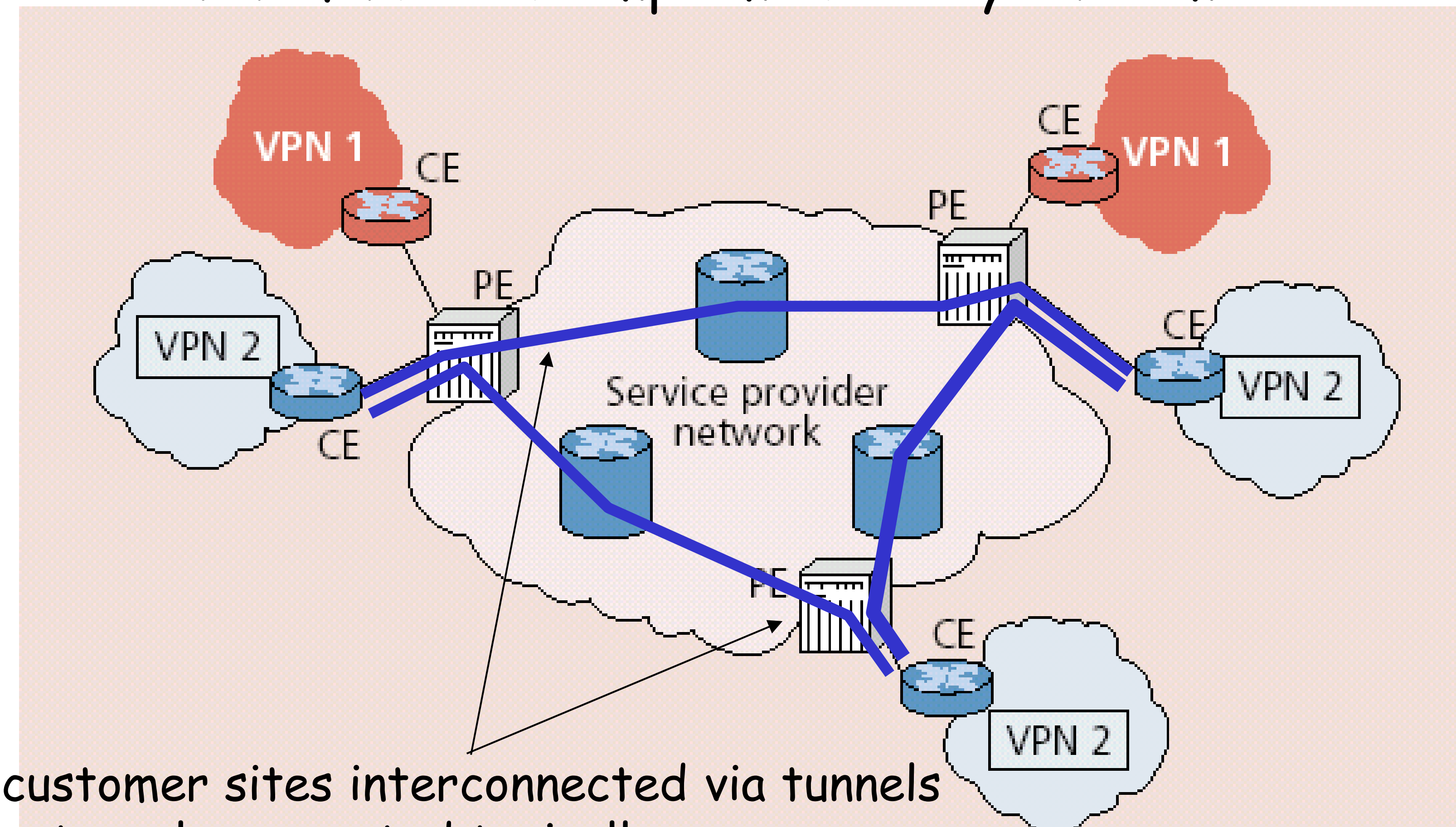


customer sites interconnected via static virtual channels leased lines

customer site connects to provider edge

Customer premise VPN

- All VPN functions implemented by customer



customer sites interconnected via tunnels

- tunnels encrypted typically
- SP treats VPN packets like all other packets

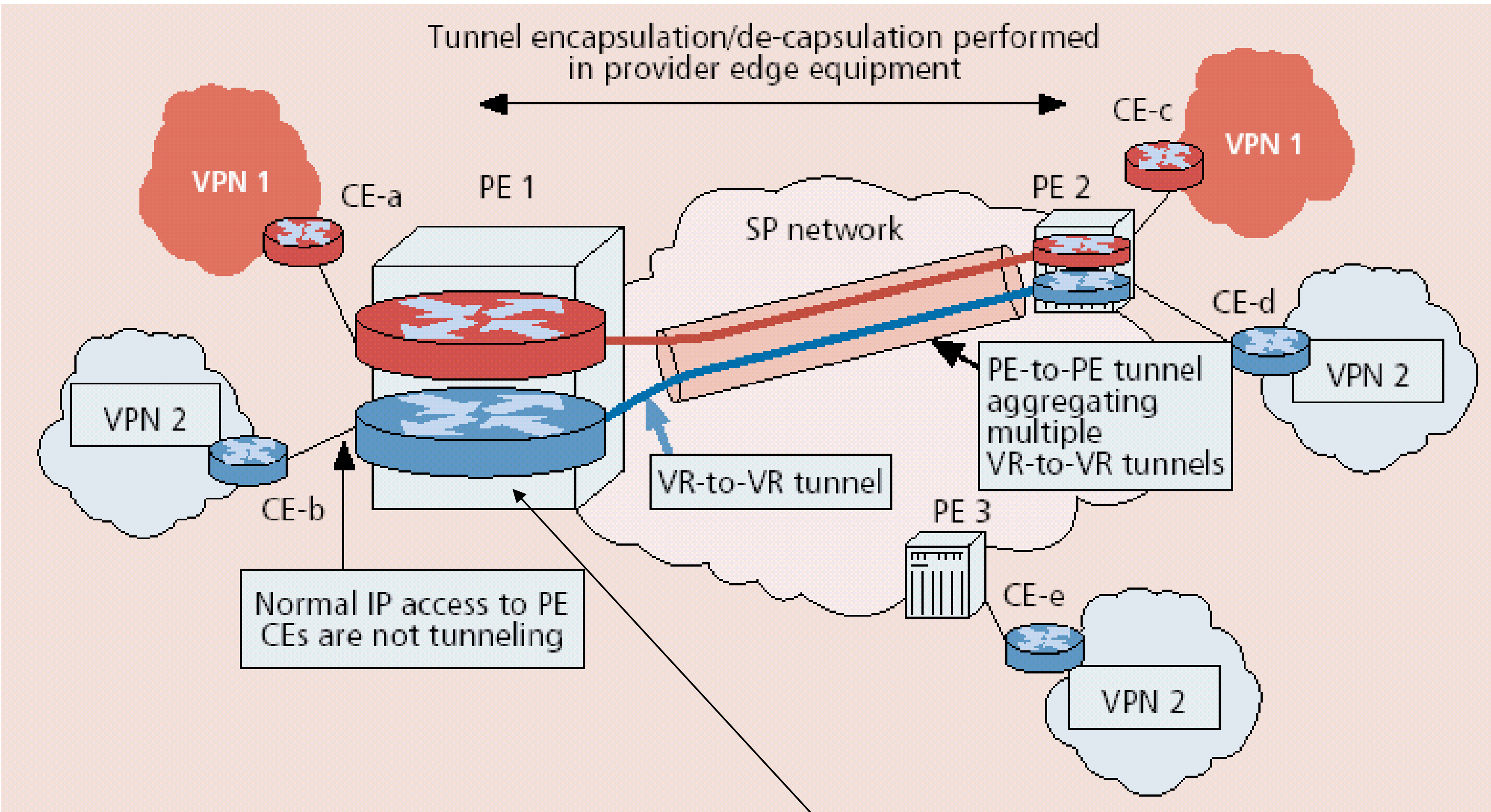
Drawbacks

- Leased-line VPN: configuration costs, maintainence by SP: long time, much manpower
- CPE-based VPN: expertise by customer to acquire, configure, manage VPN

Network-based VPN

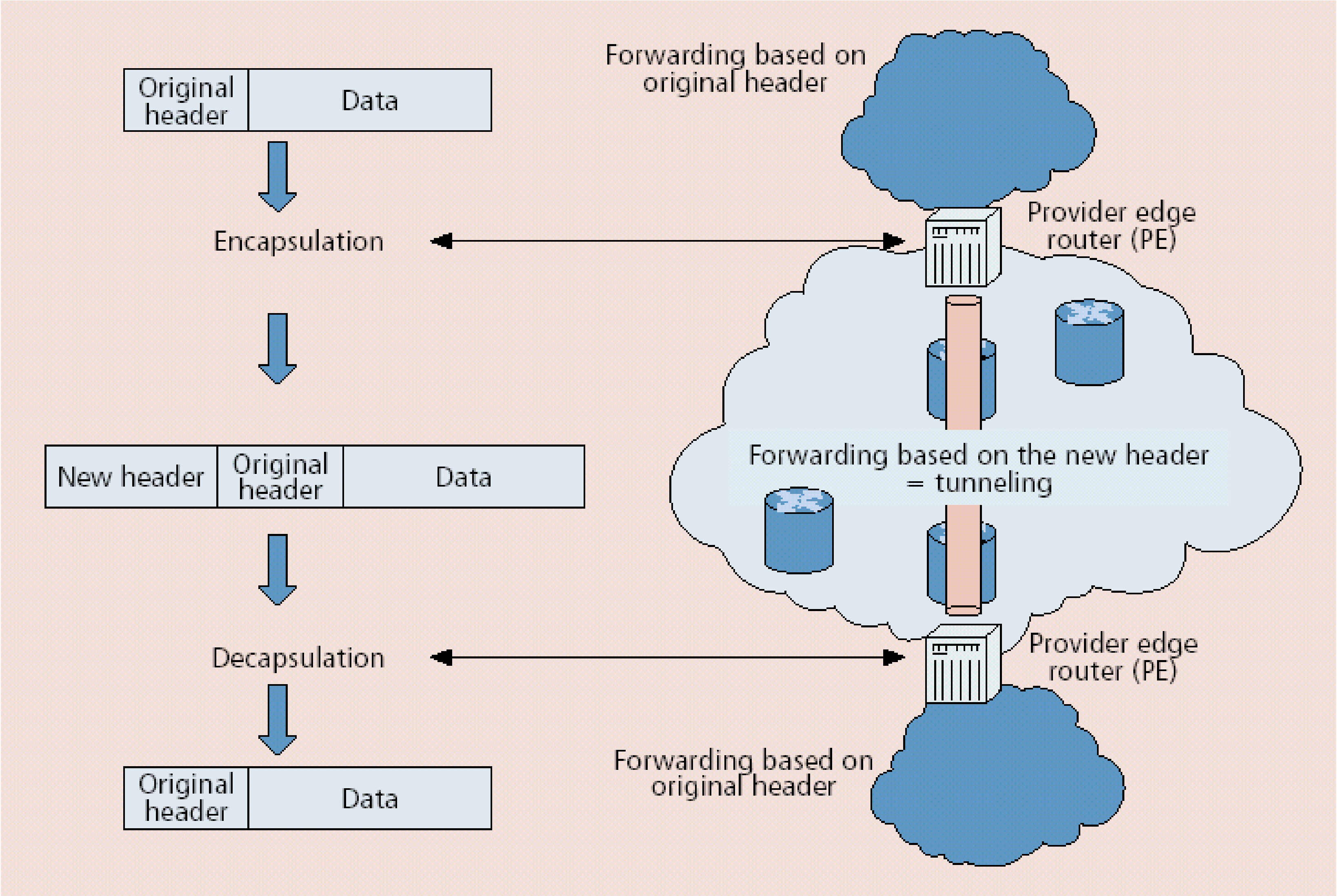
- customer's routers connect to SP routers
- SP routers maintain separate (independent) IP contexts for each VPN
  - sites can use private addressing
  - traffic from one vpn can not be injected into another

Network-based Layer 3 VPNs



multiple virtual routers
in single provider edge device

Tunneling



Original header
Data
Encapsulation
New header
Original header
Data
Decapsulation
Original header
Data
Forwarding based on original header
Provider edge router (PE)
Forwarding based on the new header = tunneling
Provider edge router (PE)
Forwarding based on original header

VPNs: why?

- Privacy
- security
- works well with mobility (looks like you are always at home)
- cost: many forms of newer VPNs are cheaper than leased line VPNs
  - ability to share at lower layers even though logically separate means lower cost
  - exploit multiple paths, redundancy, fault-recovery in lower layers
  - Need isolation mechanisms to ensure resources shared appropriately
- abstraction and manageability: all machines with addresses that are "in" are trusted no matter where they are

Agenda

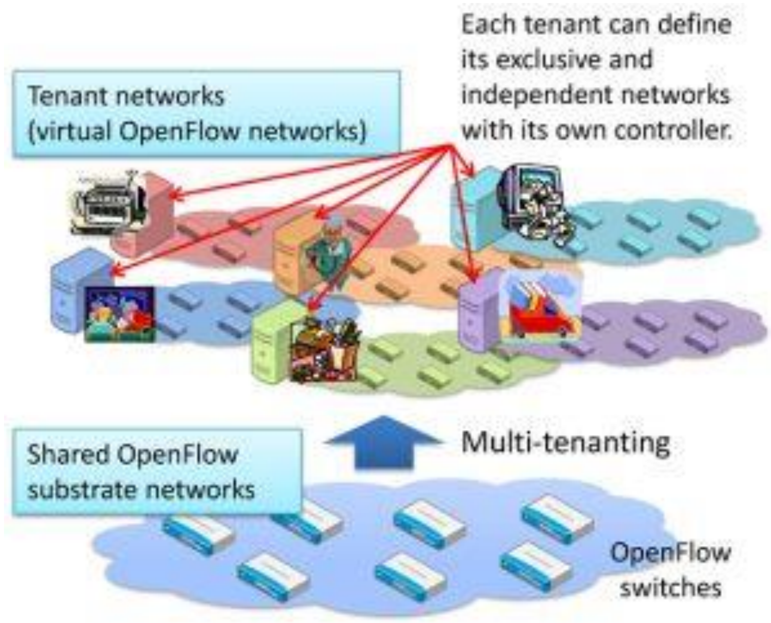
- Network virtualization basics
- Early Forms of Vnets
  - Overlay networks
  - VPNs
- Vnets:
  - External Vnets with FlowVisor/OpenVirteX
  - Internal Vnets with Open vSwtich

FlowVisor

https://github.com/OPENNETWORKINGLAB/flowvisor/wiki

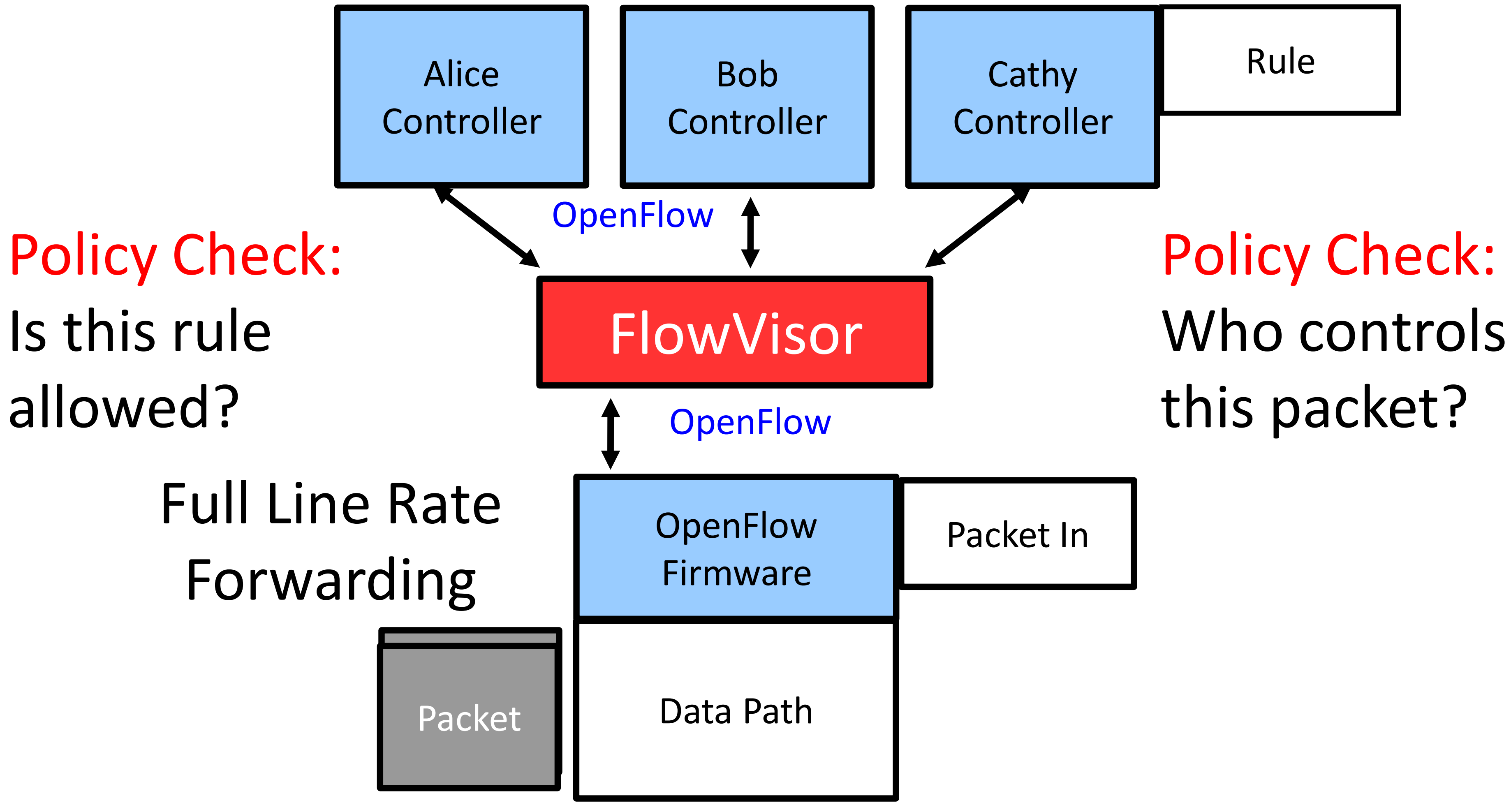
- Transparent OpenFlow proxy between switches and controllers
- Creates network "slices" which are managed by different controllers
- Enforces isolation between slices

Network "slices"



Source: http://www.nict.go.jp/en/press/2013/04/26-1.html

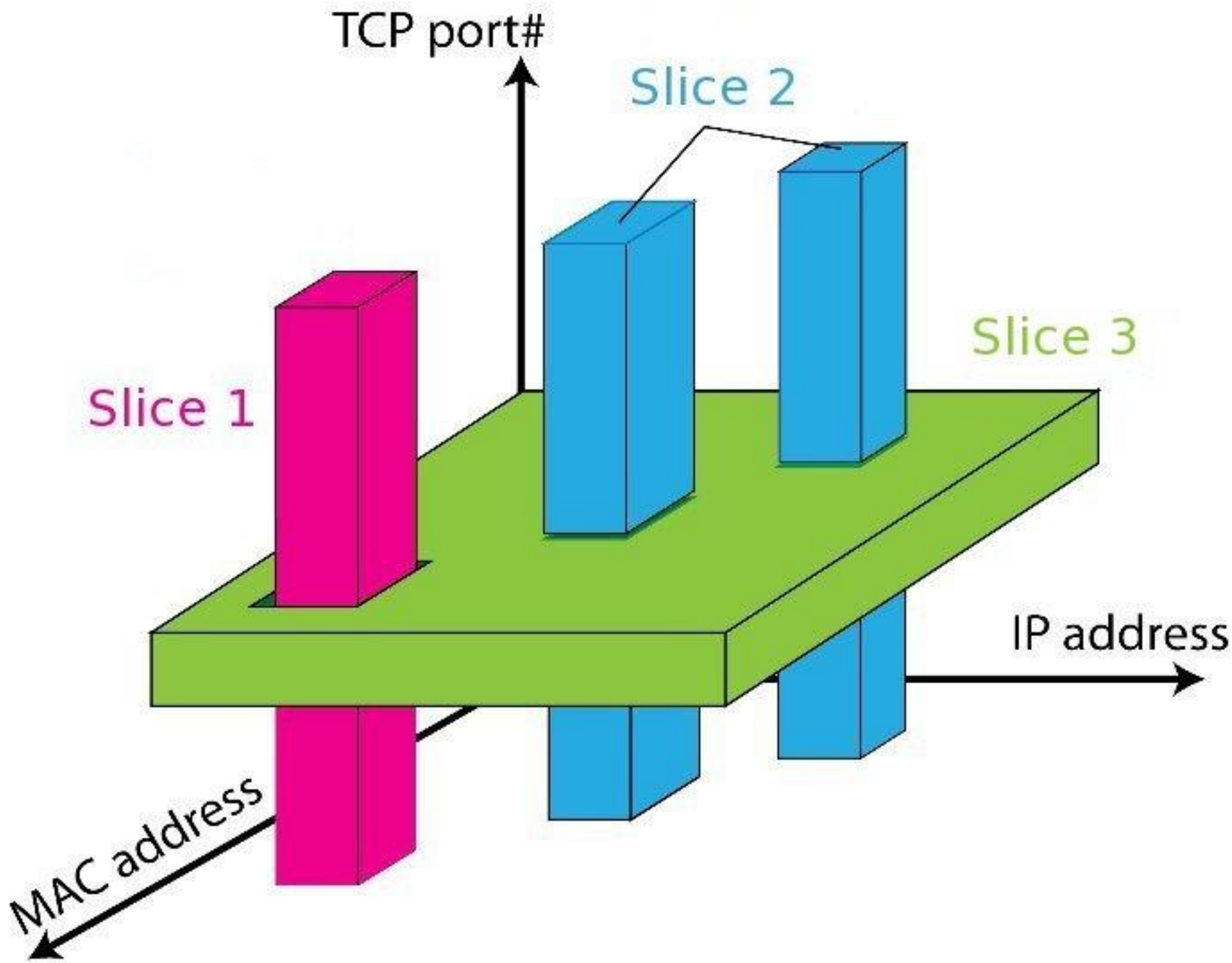
FlowVisor Message Handling



Policy: Limits Slice Resources

- FlowSpace: which packets does the slice control?
- Link bandwidth
- Number of flow table rules
- Fraction of switch/router CPU
- Topology (subgraph)

FlowSpace: Maps Packets to Slices



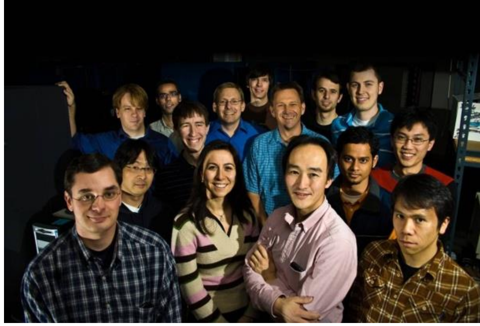
FlowVisor Deployment: Stanford

- Real production network
  - 15 switches, 35 APs
  - 25+ users
  - Several years of use

- Same physical network hosts Stanford demos
  - 7 different demos

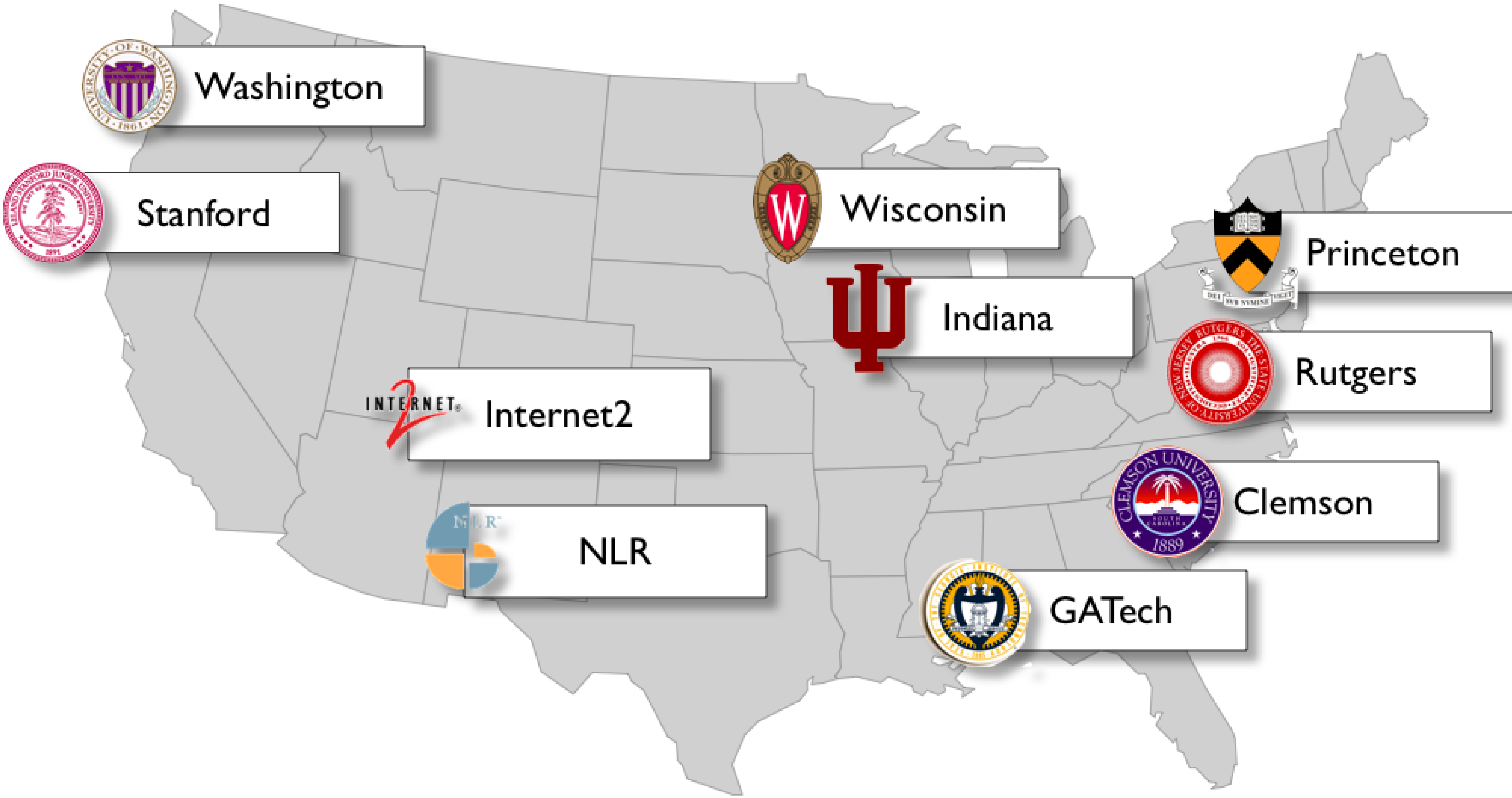
See demos in
http://archive.openflow.org/videos/

Real User Traffic: Opt-In



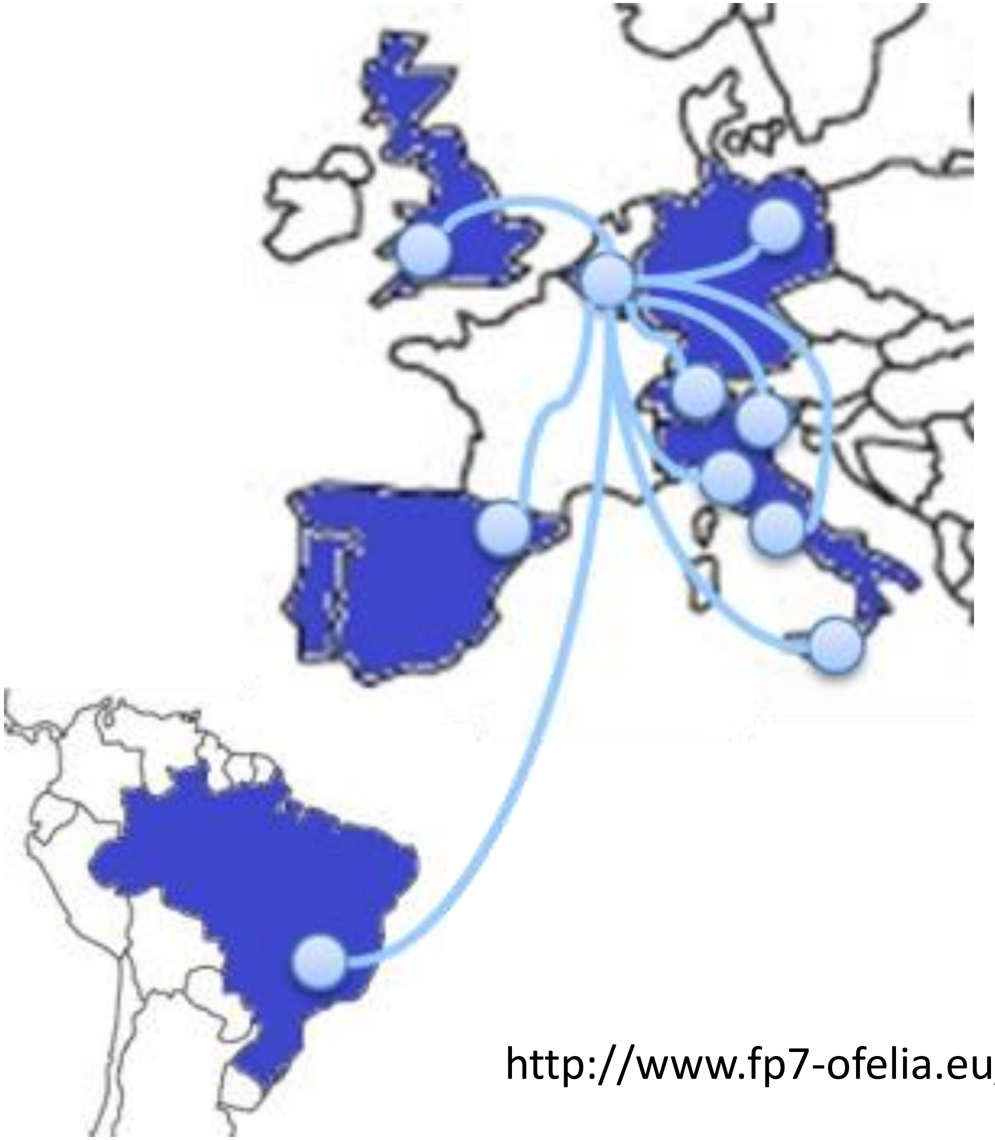
- Allow users to Opt-In to services in real-time
  - Users can delegate control of individual flows to slices
  - Add new FlowSpace to each slice's policy
- Example:
  - "Slice 1 will handle my HTTP traffic"
  - "Slice 2 will handle my VoIP traffic"
  - "Slice 3 will handle everything else"

FlowVisor Deployments: GENI Testbed



GENI stands for Global Environment for Network Innovations

OFELIA Testbed



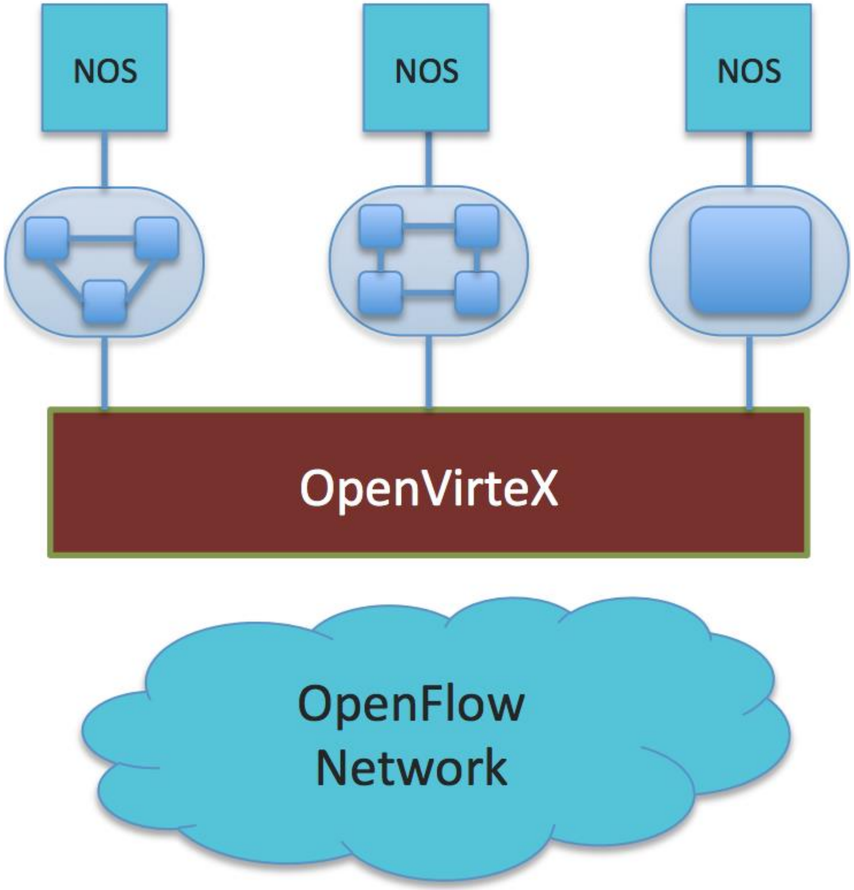
- TU Berlin
- IBBT, Belgium
- ETH Zurich
- i2CAT, Spain
- UNIVBRIS, UK
- CNIT, Italy
- CREATE-NET, Italy
- UFU, Brasil
- CTTC, Spain

http://www.fp7-ofelia.eu/

OpenVirteX (OVX)

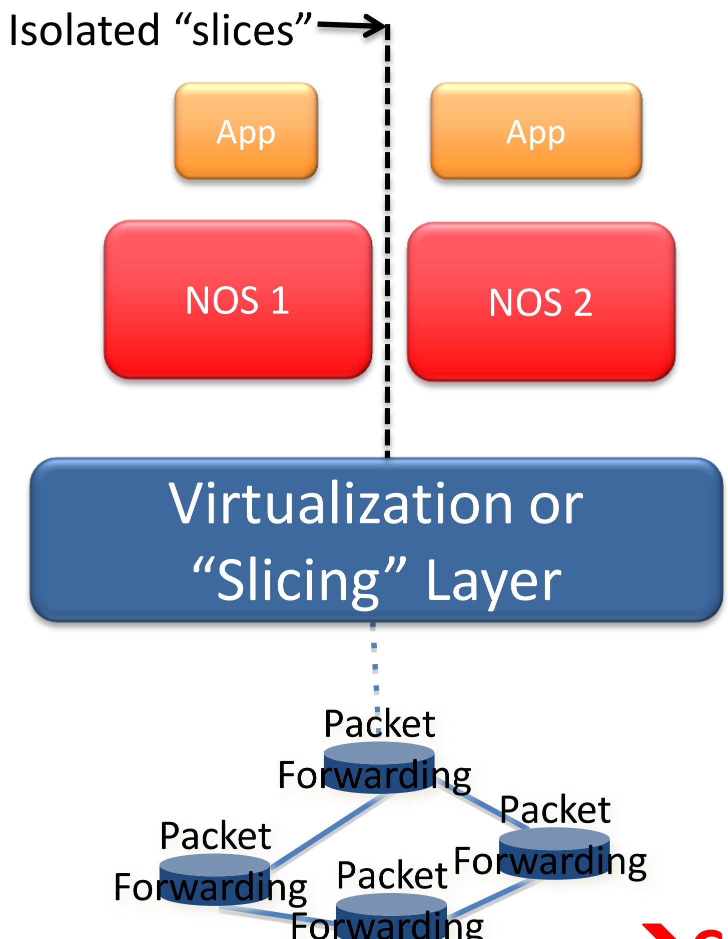
http://www.openvirtex.org/

- Slicing like FlowVisor
- Address space virtualization
  - vnets can use same addresses
  - inserts tags to identify slices
- Custom topologies

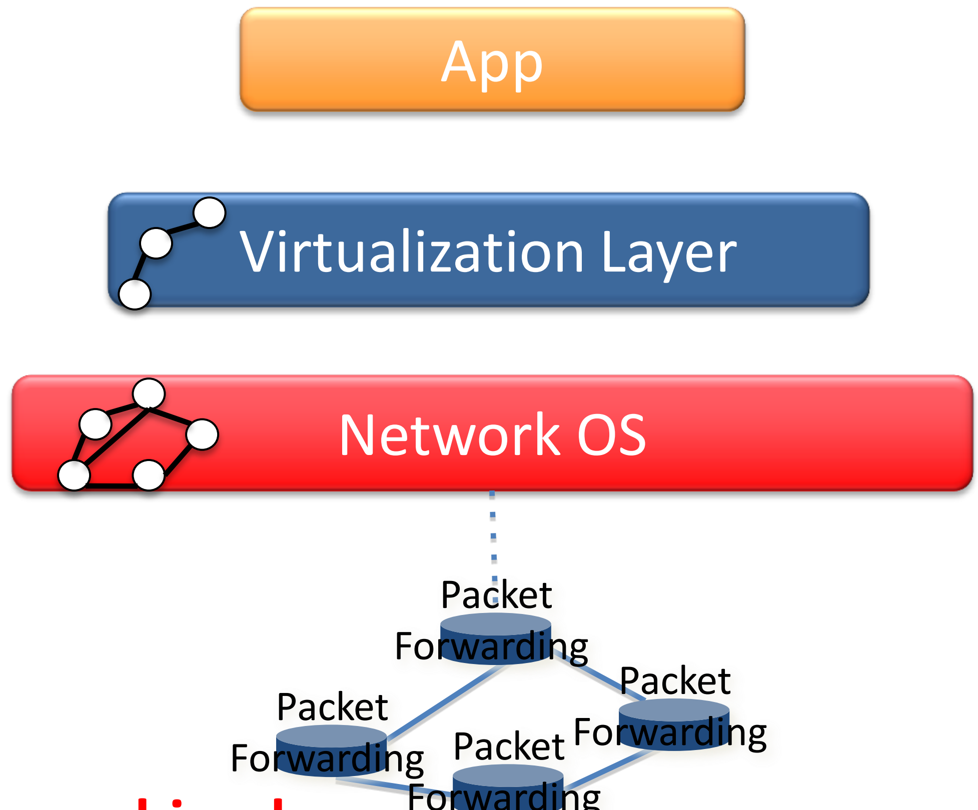


Underlay vs. Overlay Vnets

Underlay approach:
Slicing (e.g. FlowVisor)



Overlay approach:
App-specific topology abstraction



→ Can be combined

Agenda

- Network virtualization basics
- Early Forms of Vnets
  - Overlay networks
  - VPNs
- Vnets:
  - External Vnets with FlowVisor/OpenVirteX
  - Internal Vnets with Open vSwitch

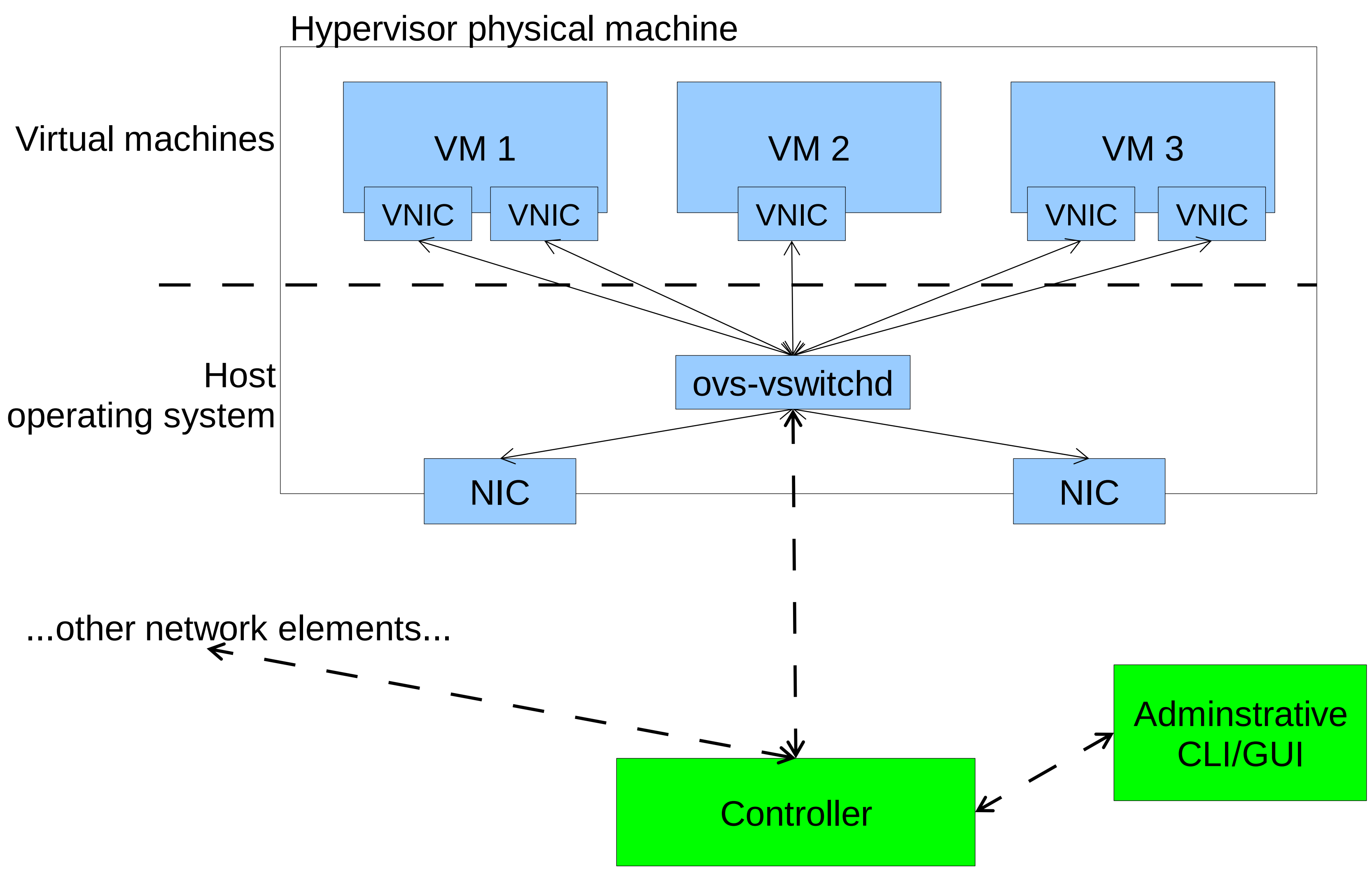
Open vSwitch (OVS)

http://openvswitch.org

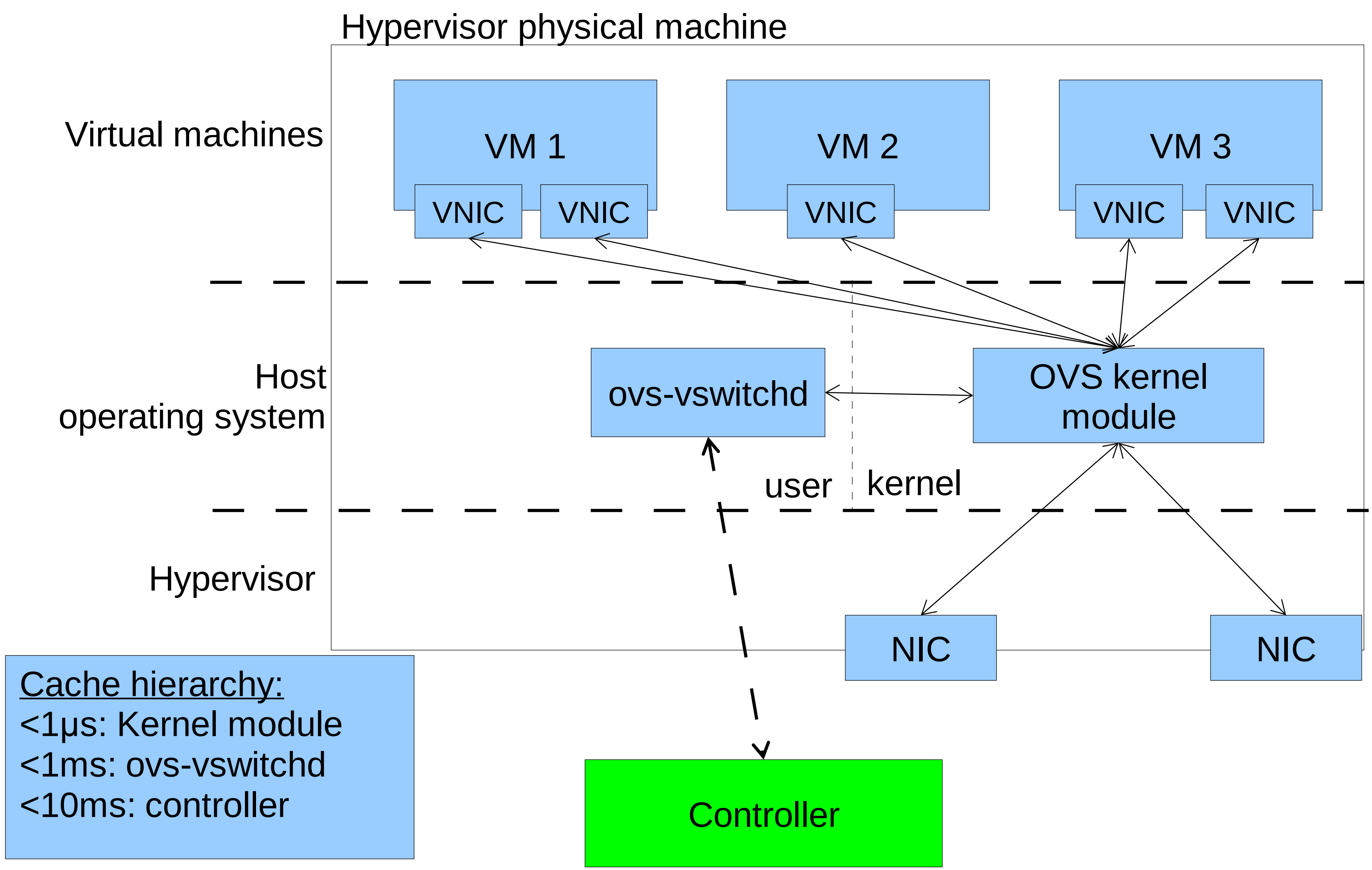
- Open source switch for hardware virtualization
- Supported by: Xen, KVM, VirtualBox, OpenStrack, OpenNebula, etc.
- Runs within the hypervisor or standalone
- Comes with Linux kernel

Open vSwitch: Design Overview

ovs-vswitchd: The Open vSwitch deamon manages and controls OVS instances on the local machine



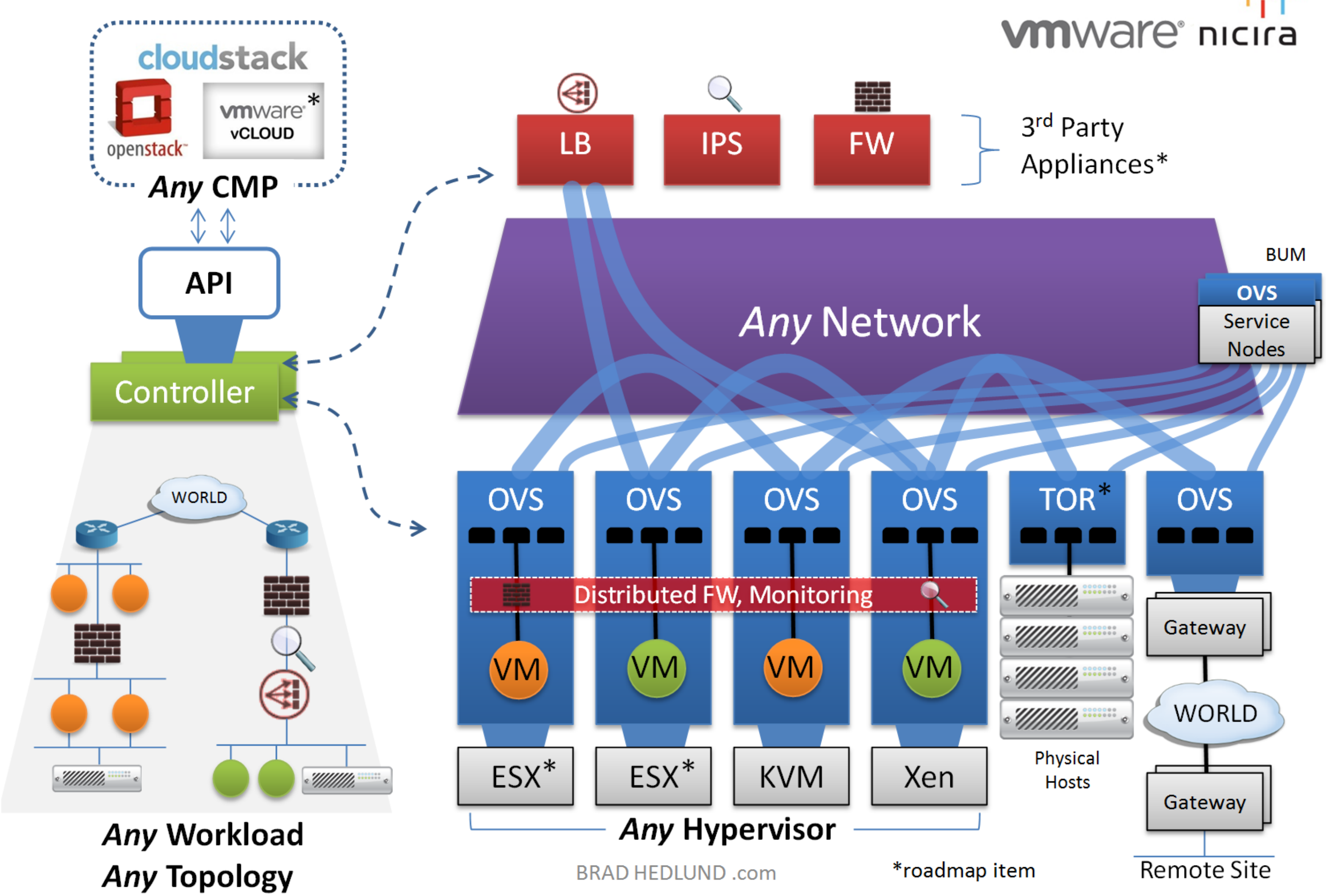
Open vSwitch: Cache Hierarchy



Challenges with Virtual Switches

- **Feature Heterogeneity**: Cannot use advanced hardware features for load balancing and traffic shaping of physical switches
- **Increased latency and decreased throughput**: The hypervisor adds overhead
- **More switches to manage**
- **Large broadcast domains** resulting from VLAN trunking

Network Virtualization Platform



OVS integral part of NVP solution:

- **Core does simple forwarding**
- **Edge does middlebox functions**

Agenda

- Network virtualization basics
- Early Forms of Vnets
  - Overlay networks
  - VPNs
- Vnets:
  - External Vnets with FlowVisor/OpenVirteX
  - Internal Vnets with Open vSwitch

Τέλος Ενότητας

Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο
ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΕΚΠΑΙΔΕΥΣΗ ΚΑΙ ΔΙΑ ΒΙΟΥ ΜΑΘΗΣΗ
επένδυση στην κοινωνία της γνώσης
ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ
ΕΣΠΑ 2007-2013
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ
Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

Χρηματοδότηση

- Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στα πλαίσια του εκπαιδευτικού έργου του διδάσκοντα.
- Το έργο «**Ανοικτά Ακαδημαϊκά Μαθήματα στο Πανεπιστήμιο Κρήτης**» έχει χρηματοδοτήσει μόνο τη αναδιαμόρφωση του εκπαιδευτικού υλικού.
- Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «Εκπαίδευση και Δια Βίου Μάθηση» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους.

Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο

ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΕΚΠΑΙΔΕΥΣΗ ΚΑΙ ΔΙΑ ΒΙΟΥ ΜΑΘΗΣΗ
επένδυση στην κοινωνία της γνώσης
ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

ΕΣΠΑ
2007-2013
πρόγραμμα για την ανάπτυξη
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

Σημειώματα

Σημείωμα αδειοδότησης

Σημείωμα Αναφοράς

Copyright Πανεπιστήμιο Κρήτης, Ξενοφώντας Δημητρόπουλος. «**Δίκτυα Καθοριζόμενα από Λογισμικό. 2.1 Network Visualization**». Έκδοση: 1.0. Ηράκλειο/Ρέθυμνο 2015. Διαθέσιμο από τη δικτυακή διεύθυνση: http://www.csd.uoc.gr/~hy436/