

Software Defined Networking



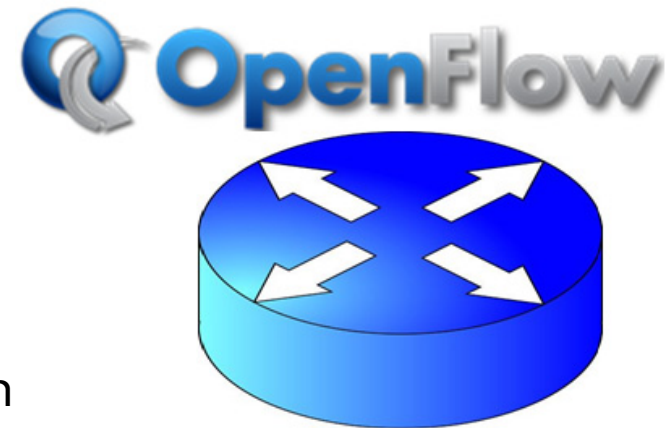
TECHNISCHE
UNIVERSITÄT
DARMSTADT

Network Virtualization and Slicing

David Hausheer

Department of Electrical Engineering
and Information Technology
Technische Universität Darmstadt

E-Mail: hausheer@ps.tu-darmstadt.de
<http://www.ps.tu-darmstadt.de/teaching/sdn>



*Original slides for this lecture provided by Xenofontas Dimitropoulos (ETH Zürich)

- ❖ Sharing physical hardware or software resources by multiple users and/or use cases
- ❖ Examples
 - Operating system shares hardware resources between multiple, e.g., processes
 - Virtual machine shares a physical machine with diverse and multiple operating systems
 - Multiplexing shares a physical channel with multiple communication flows

- ❖ Share physical network resources to form multiple diverse virtual networks
- ❖ Examples
 - 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).
- ❖ Benefits:
 - Create multiple VNs
 - Simplifies management
 - Increases utilization of resources

❖ Part I: Network Virtualization in Data Centers

- Problems: Isolation, Connectivity
- Solution: Network Virtualization
- Network Tunnels
- A Network Virtualization Architecture
- Open vSwitch Design

❖ Part II: Network Virtualization in Campus Networks

- Problem
- Solution Overview: Network Slicing
- Network Slicing Architecture
- FlowVisor



Part I: Network Virtualization in Data Centers

Input from Ben Pfaff, Nicira Inc.

Data Centers

Front of a rack



Rear of a rack



"Top of Rack"
switch



A data center has many racks.

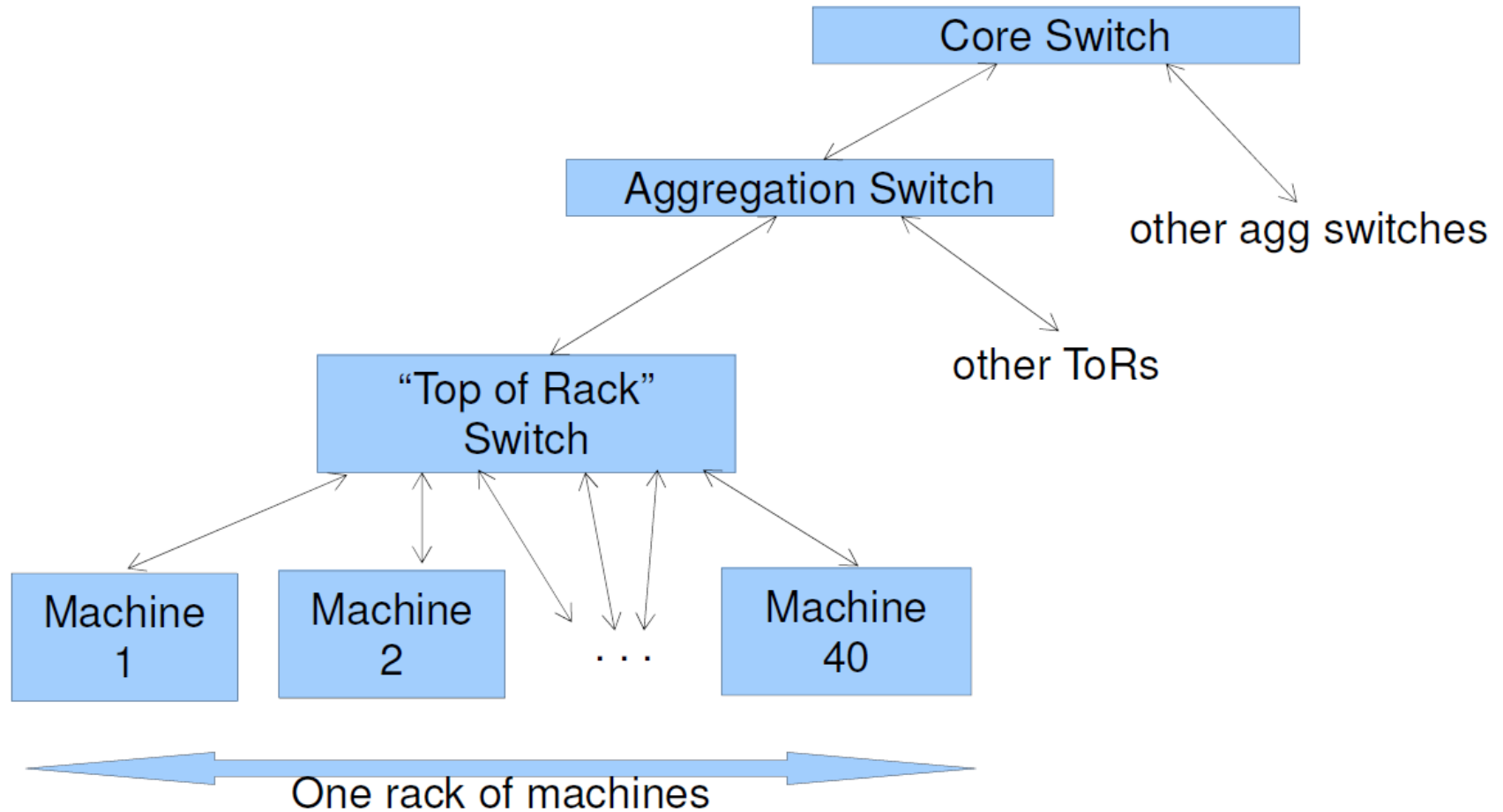
A rack has 20-40 servers.

The servers in a rack connect to a 48-port "top of rack" (ToR) switch.

Data centers buy the cheapest ToR switches they can find. They are pretty dumb devices.

Data centers do not rewire their networks without a really good reason.

Data Center Network Design before VMs



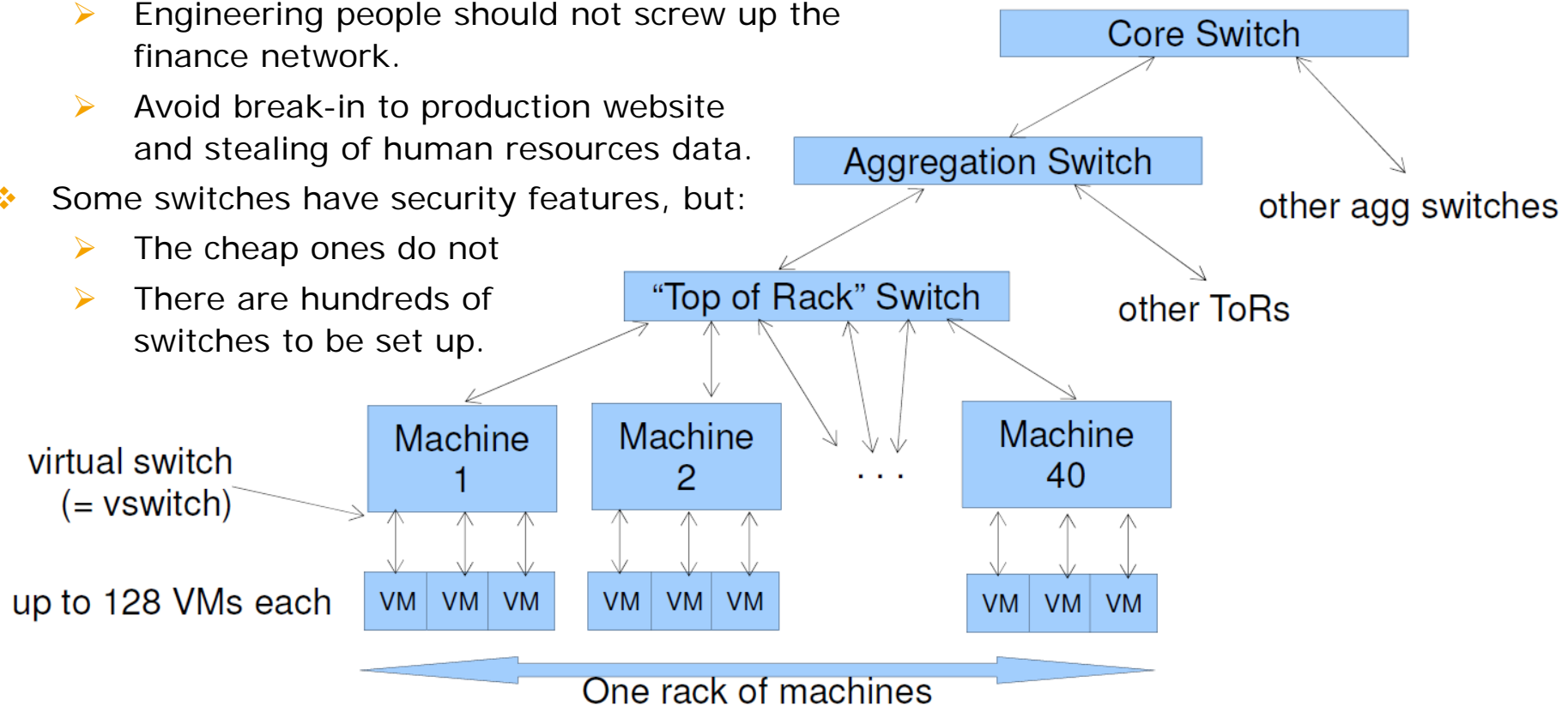
Data Center Network Design with VMs

❖ Problem: Isolation

- By default all VMs can talk to each other
- Engineering people should not screw up the finance network.
- Avoid break-in to production website and stealing of human resources data.

❖ Some switches have security features, but:

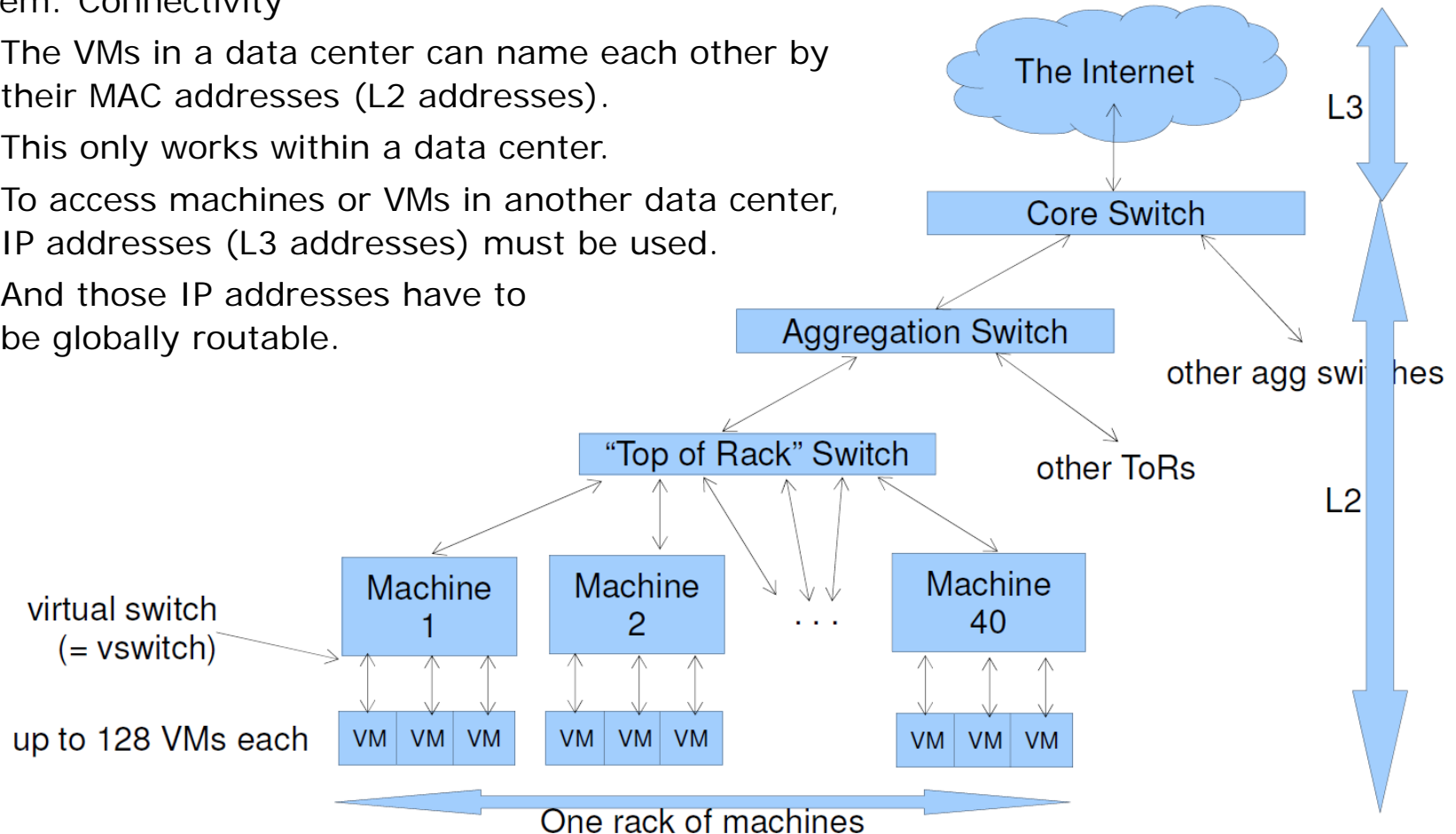
- The cheap ones do not
- There are hundreds of switches to be set up.



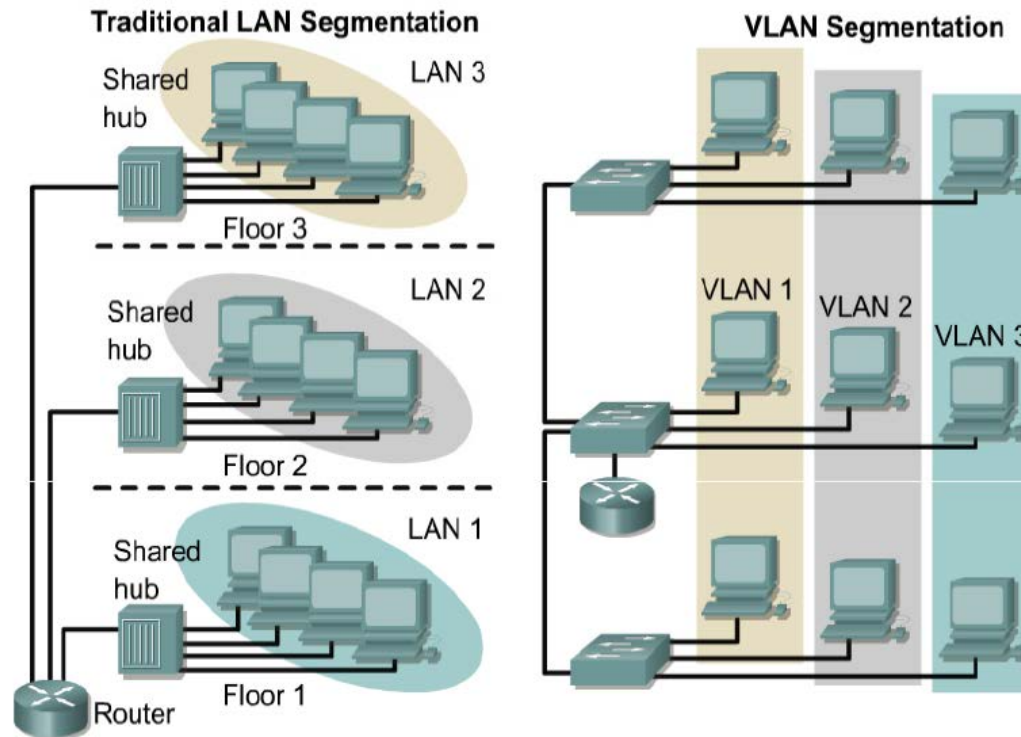
Data Center Network Design with VMs

❖ Problem: Connectivity

- The VMs in a data center can name each other by their MAC addresses (L2 addresses).
- This only works within a data center.
- To access machines or VMs in another data center, IP addresses (L3 addresses) must be used.
- And those IP addresses have to be globally routable.

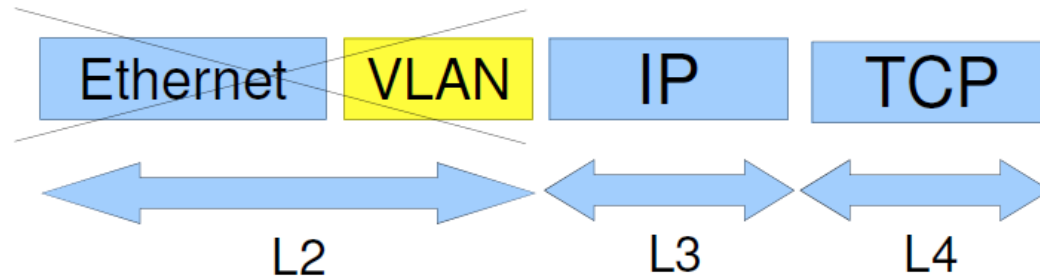


Non-Solution: VLANs



- ❖ VLANs partition a physical Ethernet network into isolated virtual Ethernet networks
- ❖ All workstations and servers used by a particular workgroup share the same VLAN, regardless of the physical connection or location.

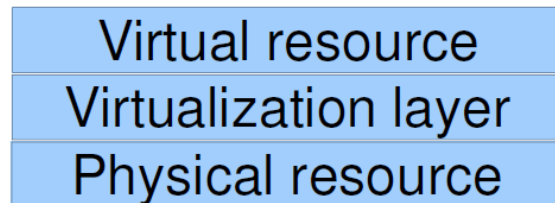
Non-Solution: VLANs



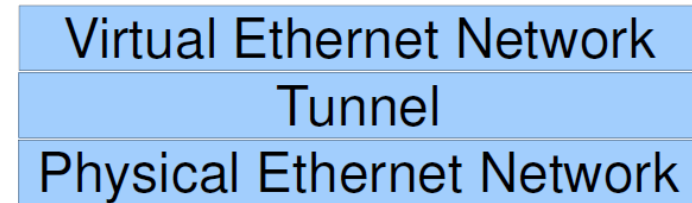
- ❖ The Internet is an L3 network.
 - When a packet crosses the Internet, it loses all its L2 headers, including the VLAN tag.
 - All the isolation is lost when the traffic crosses the Internet.
- ❖ Other problems
 - Limited number of VLANs
 - Static allocation

Solution: Network Virtualization using Encapsulation/Tunneling

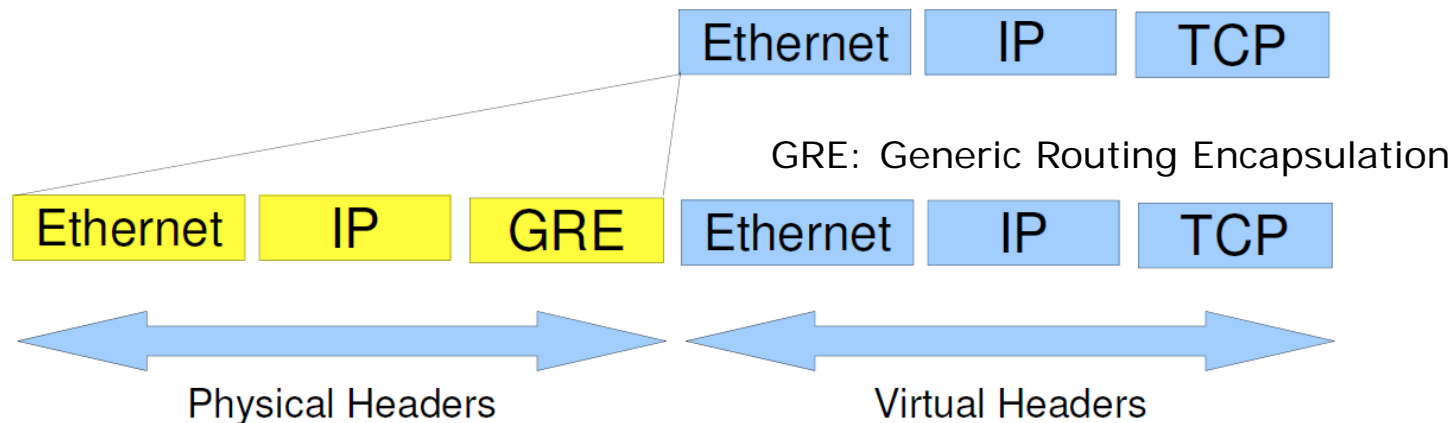
Virtualization Layering



Network Virtualization

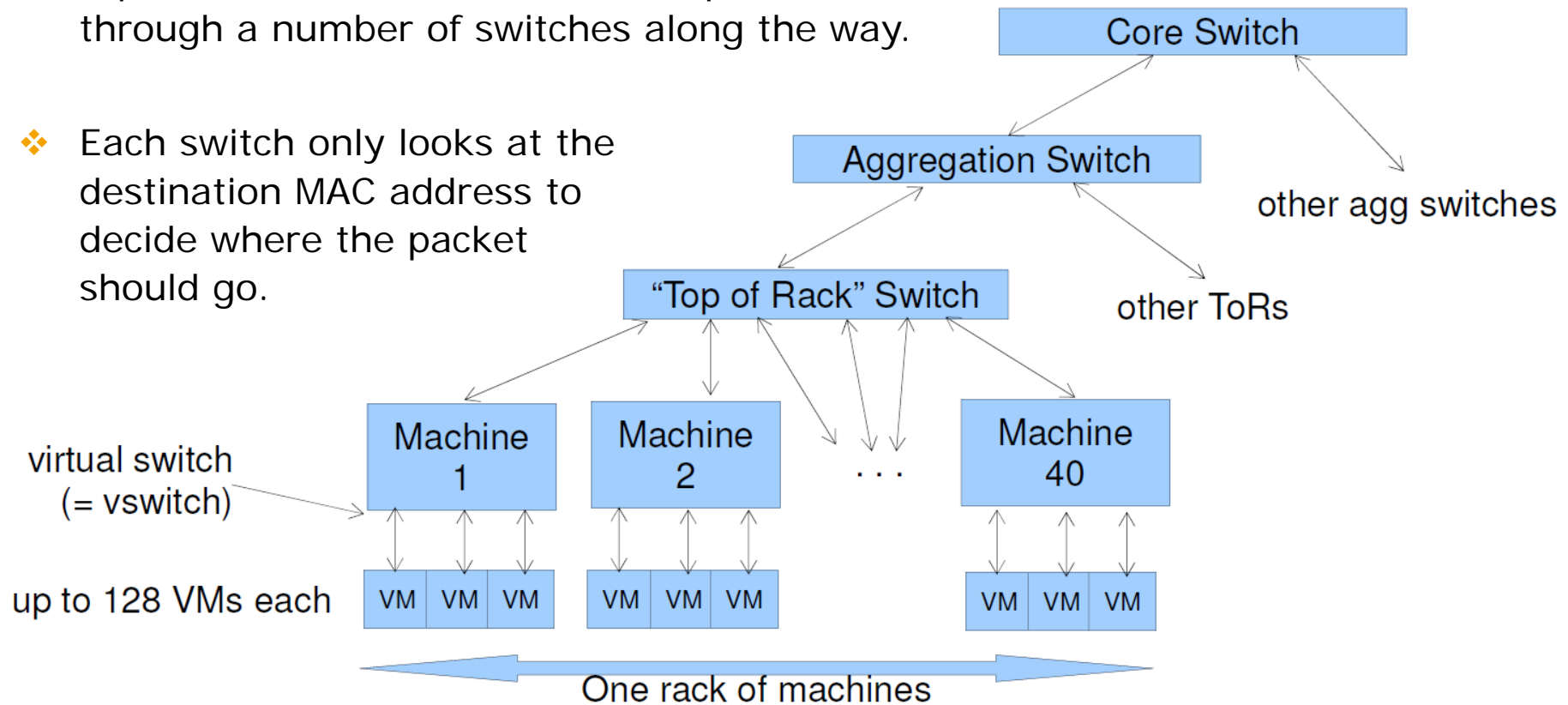


Tunneling: Separating Virtual and Physical Network

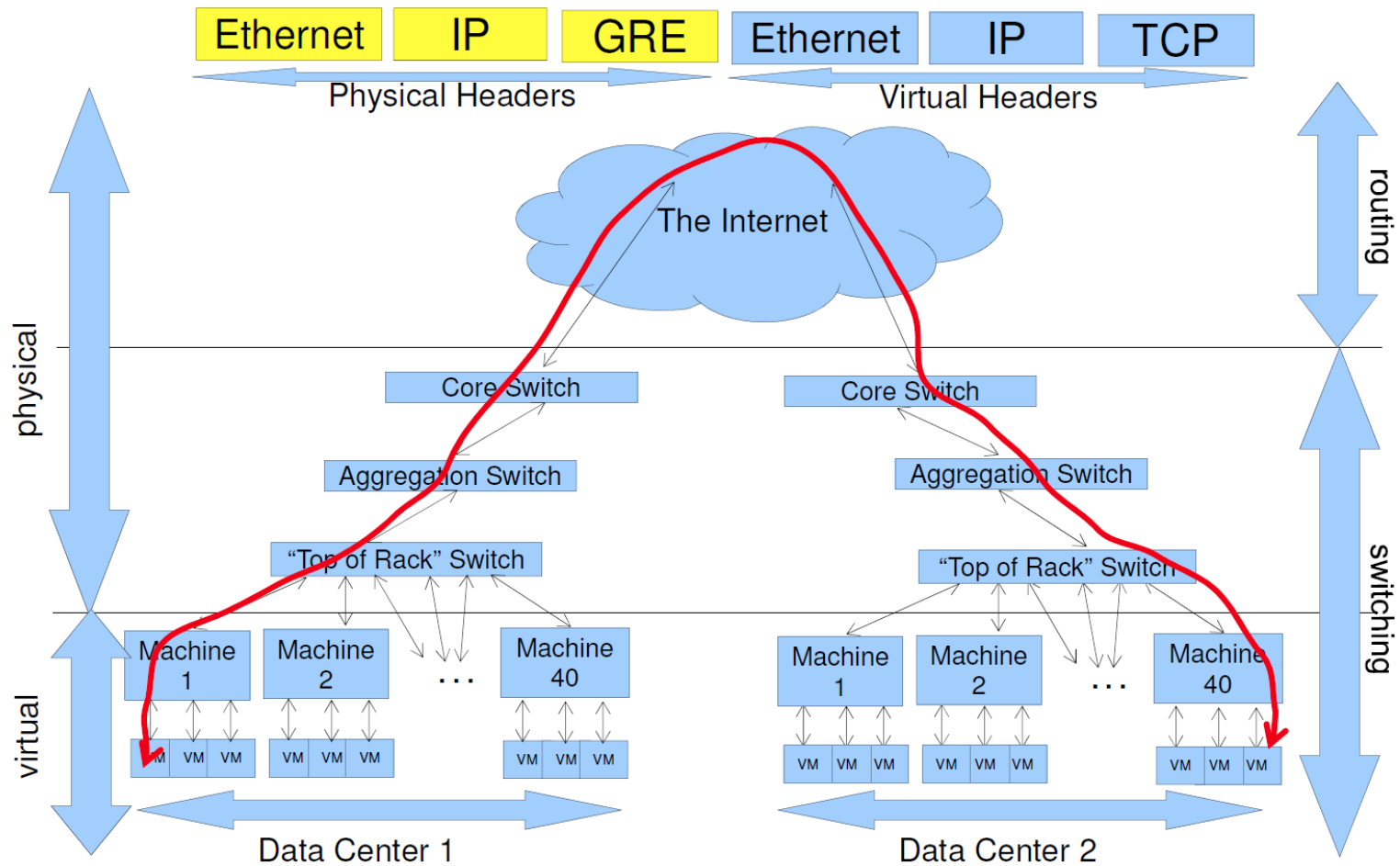


Path of a Packet (No Tunnel)

- ❖ A packet from one VM to another passes through a number of switches along the way.
- ❖ Each switch only looks at the destination MAC address to decide where the packet should go.



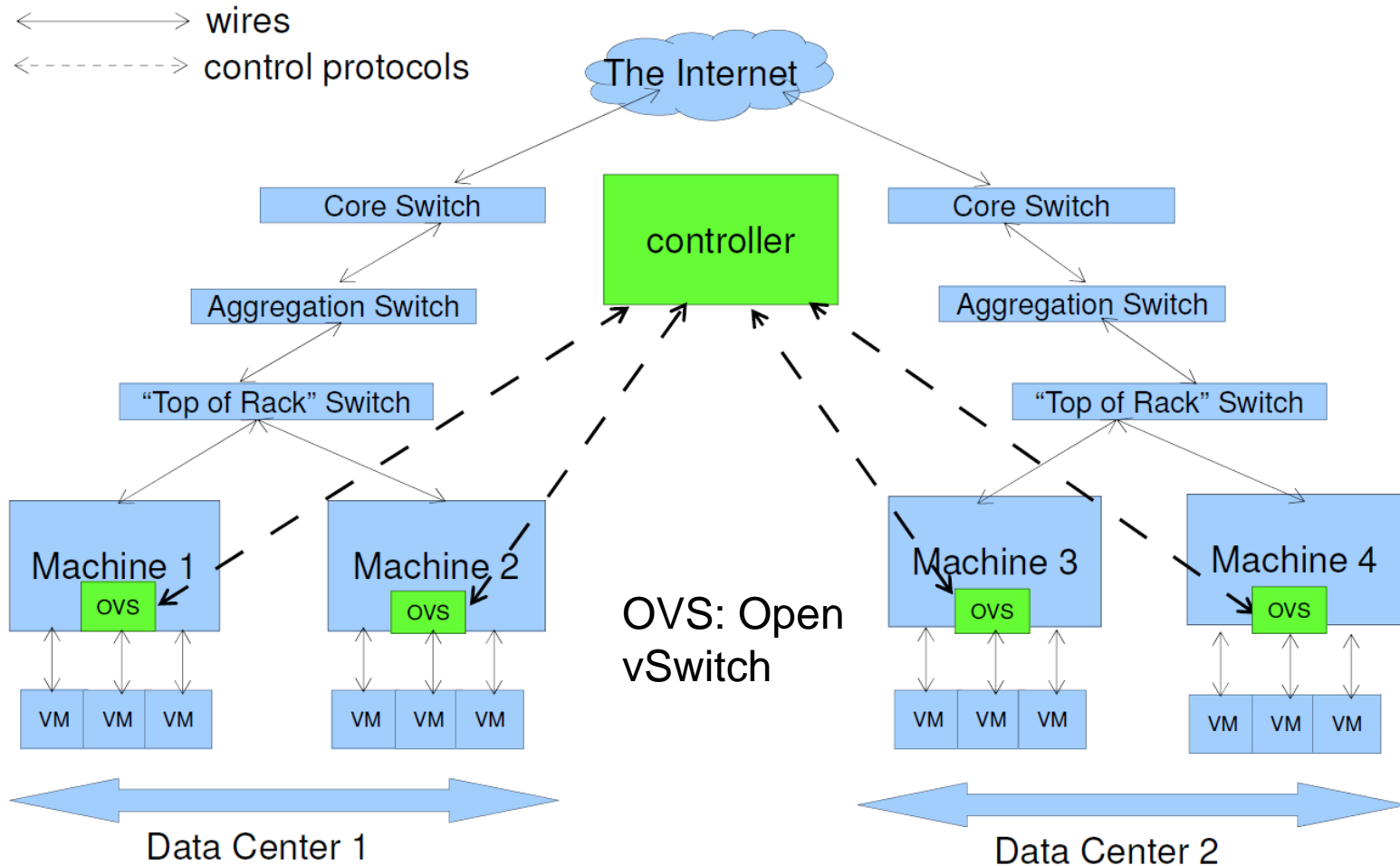
Path of a Packet (Via Tunnel)



- ❖ Setting up the tunnels:
 - After VM startup/shutdown
 - After VM migration
- ❖ Handling network failures
- ❖ Monitoring
- ❖ Administration

 Use a central controller to set up the tunnels.

A Network Virtualization Distributed System



❖ Monitor:

- Physical network
- VM locations, states

❖ Control:

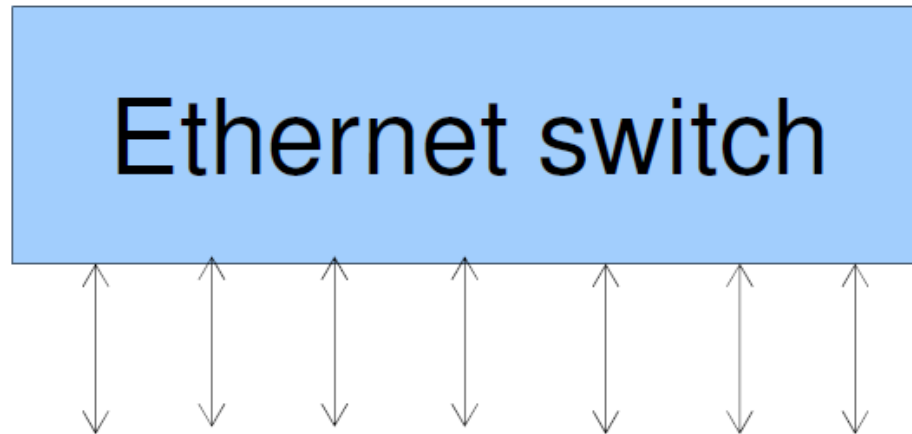
- Tunnel setup
- All packets on virtual and physical network
- Virtual/physical mapping

❖ Tells OVS what to do

Open vSwitch (OVS)

- ❖ Ethernet switch implemented in software
- ❖ Can be remotely controlled
- ❖ Tunnels (GRE and others)
- ❖ Integrates with virtual machine managers (VMMs), e.g. XenServer, KVM
- ❖ Free and open source: www.openvswitch.org

Open vSwitch: OpenFlow Protocol



Flow table = ordered list of "if-then" rules:


"If this packet comes from VM A and going to VM B, then send it out via tunnel 42."

(No rule: send to controller.)

OVS Management: OVSDB protocol

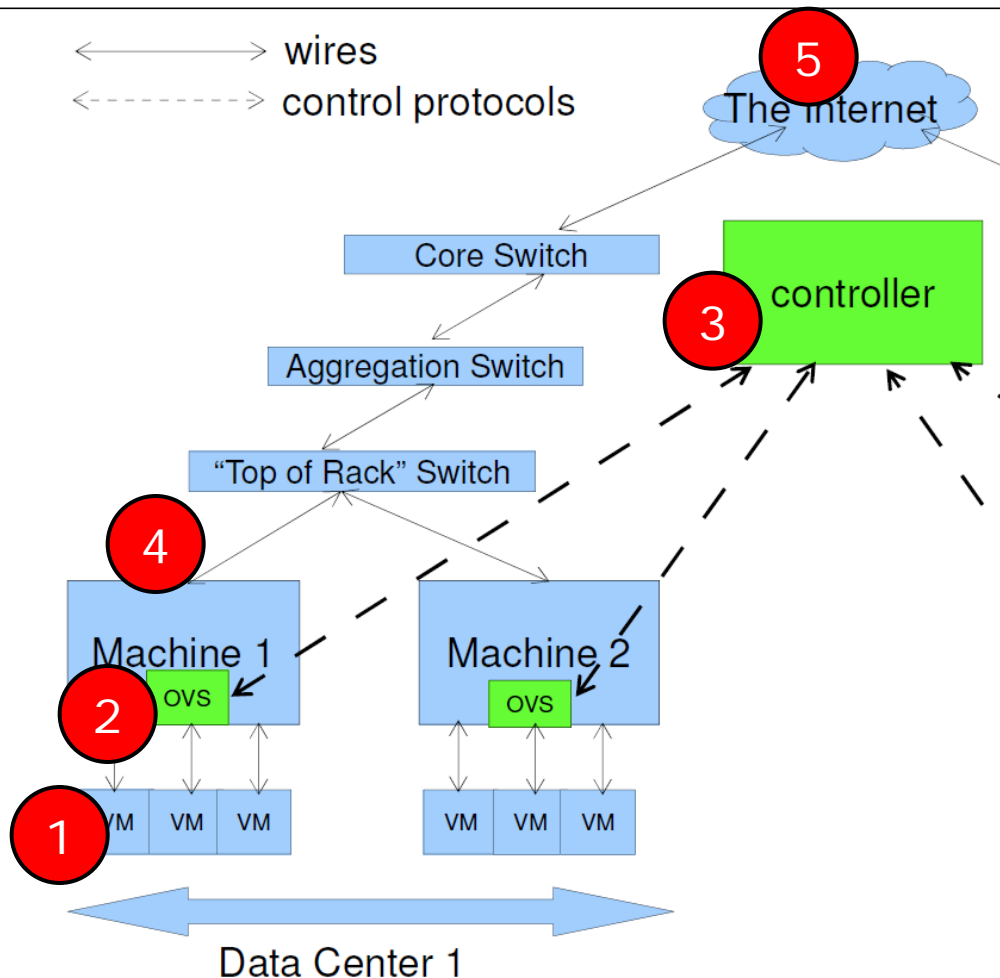
- ❖ Manage slow-moving state:
 - VM placement (via VMM integration)
 - Tunnel setup

- ❖ Other management tasks:
 - Create many virtual switch instances
 - Attach interfaces to virtual switches
 - Set QoS policies on interfaces

 OpenFlow does not solve all configuration and management problems

- ❖ Further reading about OVSDB protocol:
<http://networkheresy.com/tag/ovsdb/>

OpenFlow in the Data Center (One Possibility)



1. VM sends packet.
2. Open vSwitch checks flow table – no match. Sends packet to controller.
3. Controller tells OVS to set up a tunnel to the destination and send the packet on that tunnel.
4. OVS sends packet on the new tunnel.
5. Normal switching and routing carry the packet to its destination in the usual way.

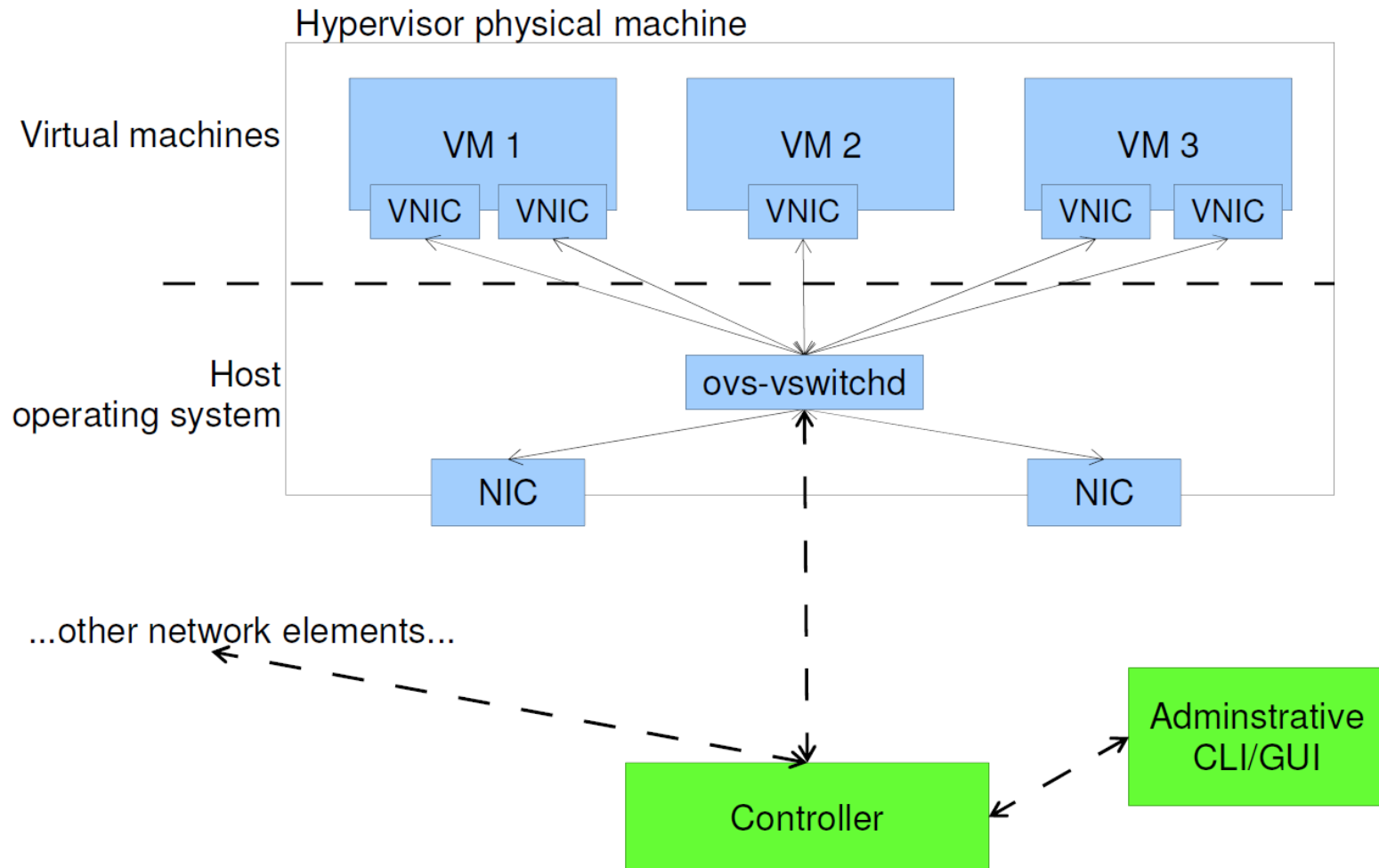
The same process repeats on the other end to send the reply back.

This is done at most on a per-"flow" basis, and other optimizations keep it from happening too frequently.

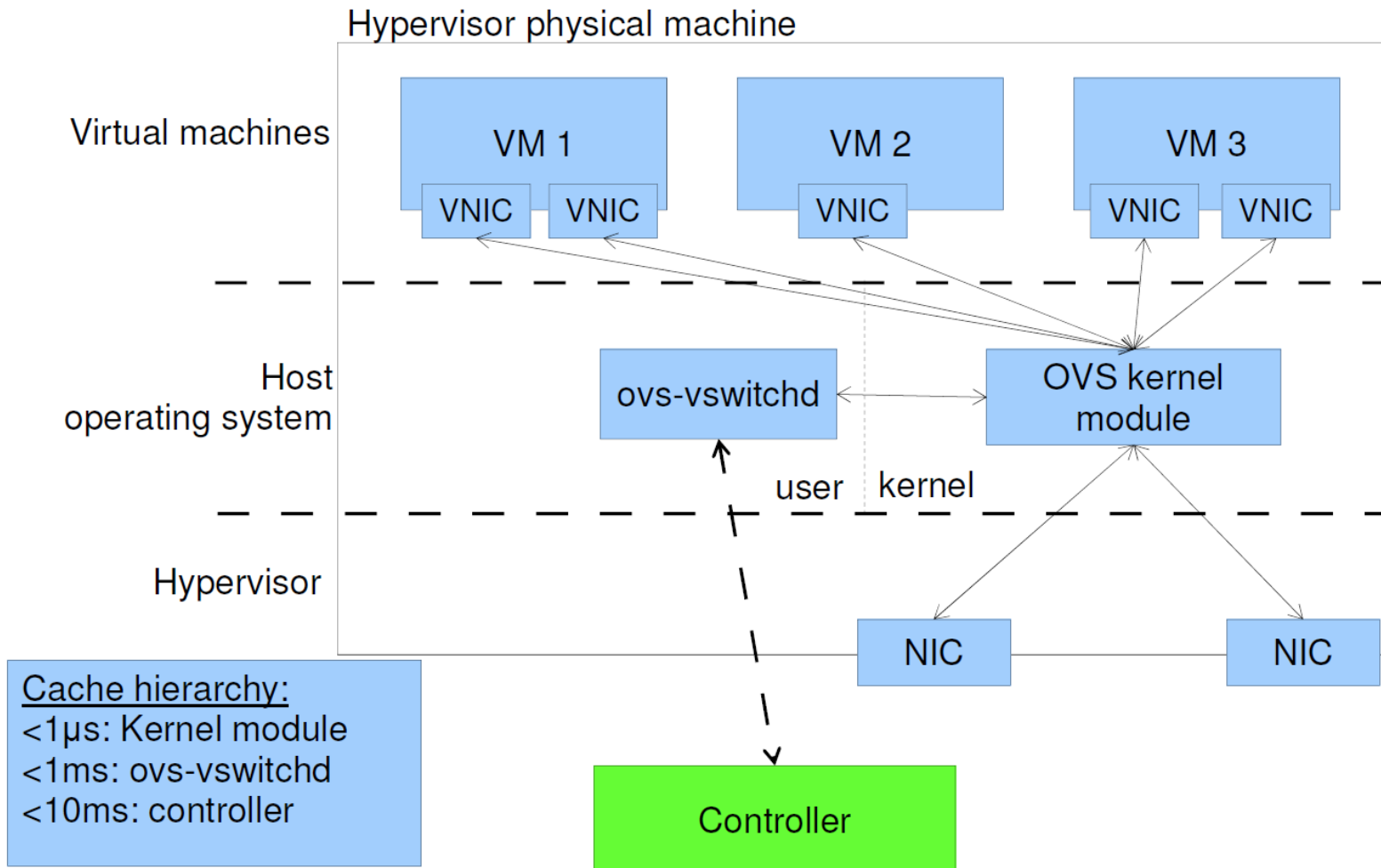
Open vSwitch: Design Overview



TECHNISCHE
UNIVERSITÄT
DARMSTADT

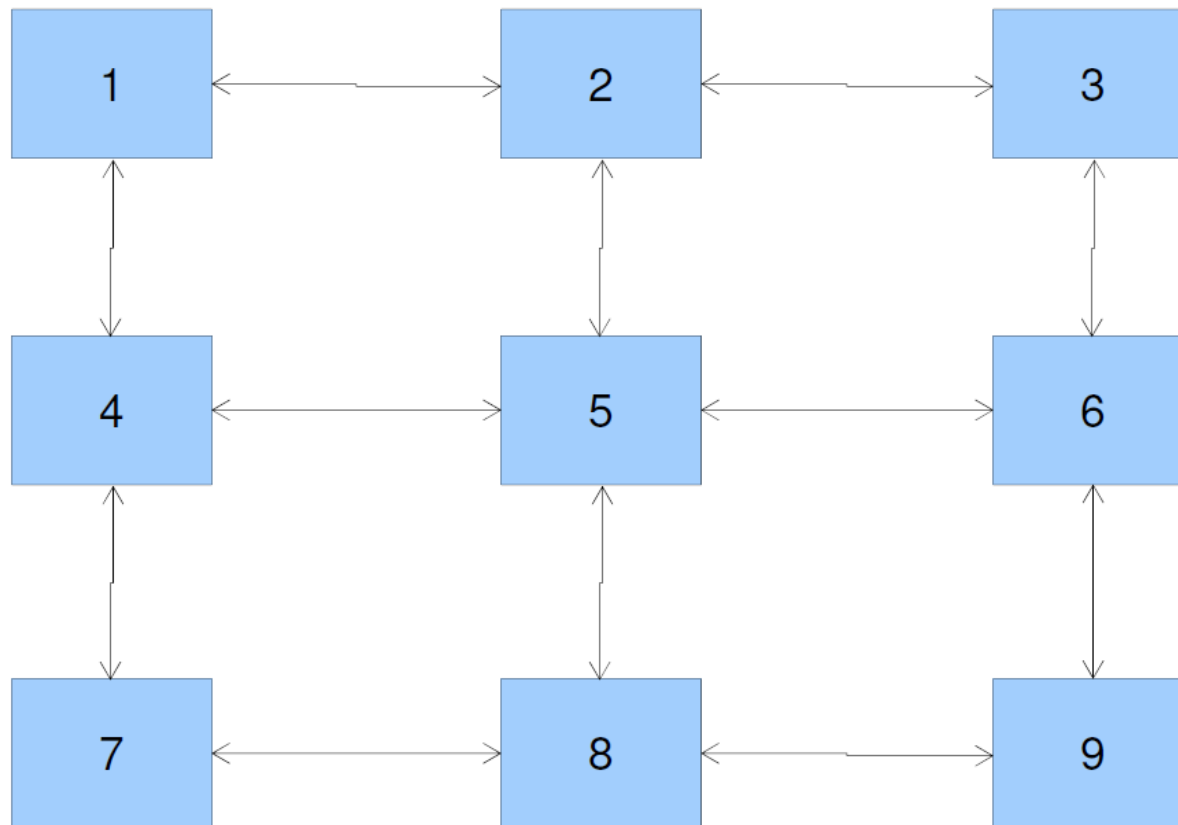


Open vSwitch: Design Details



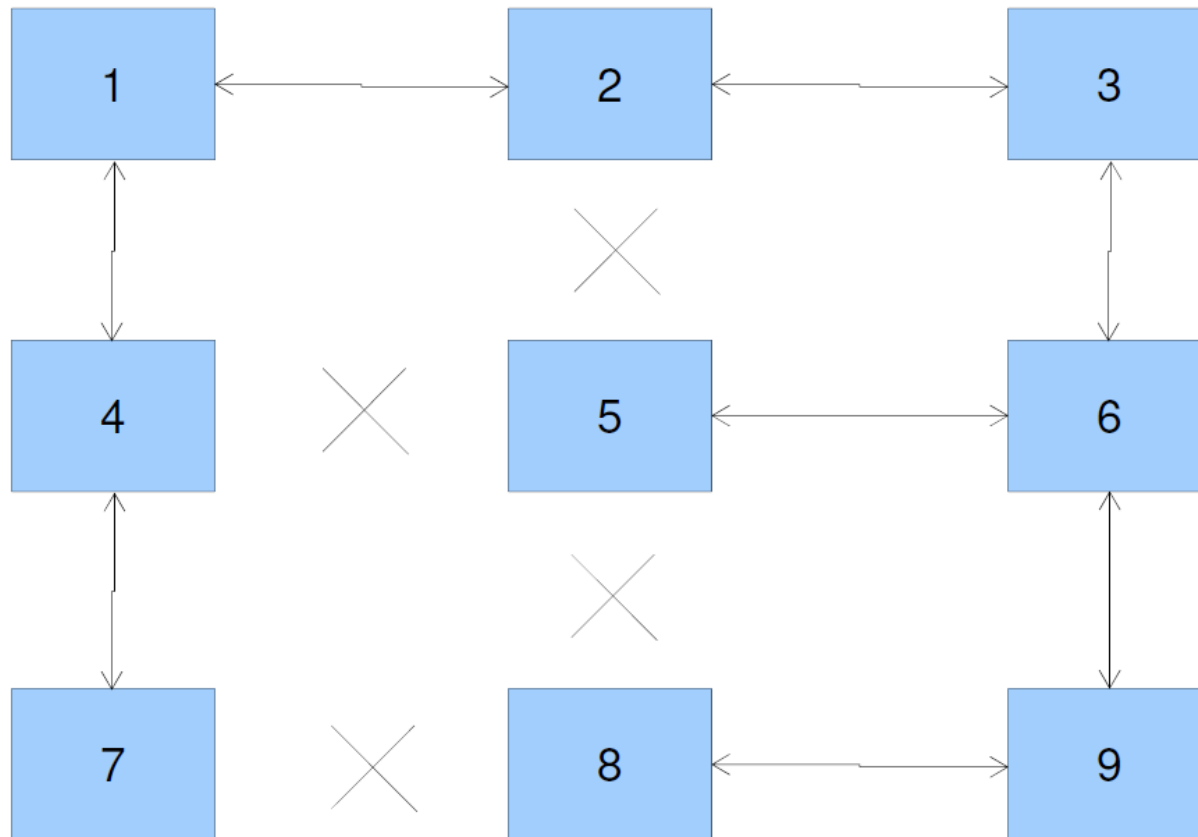
OpenFlow: Another Use

Redundant Wiring
Between Switches:



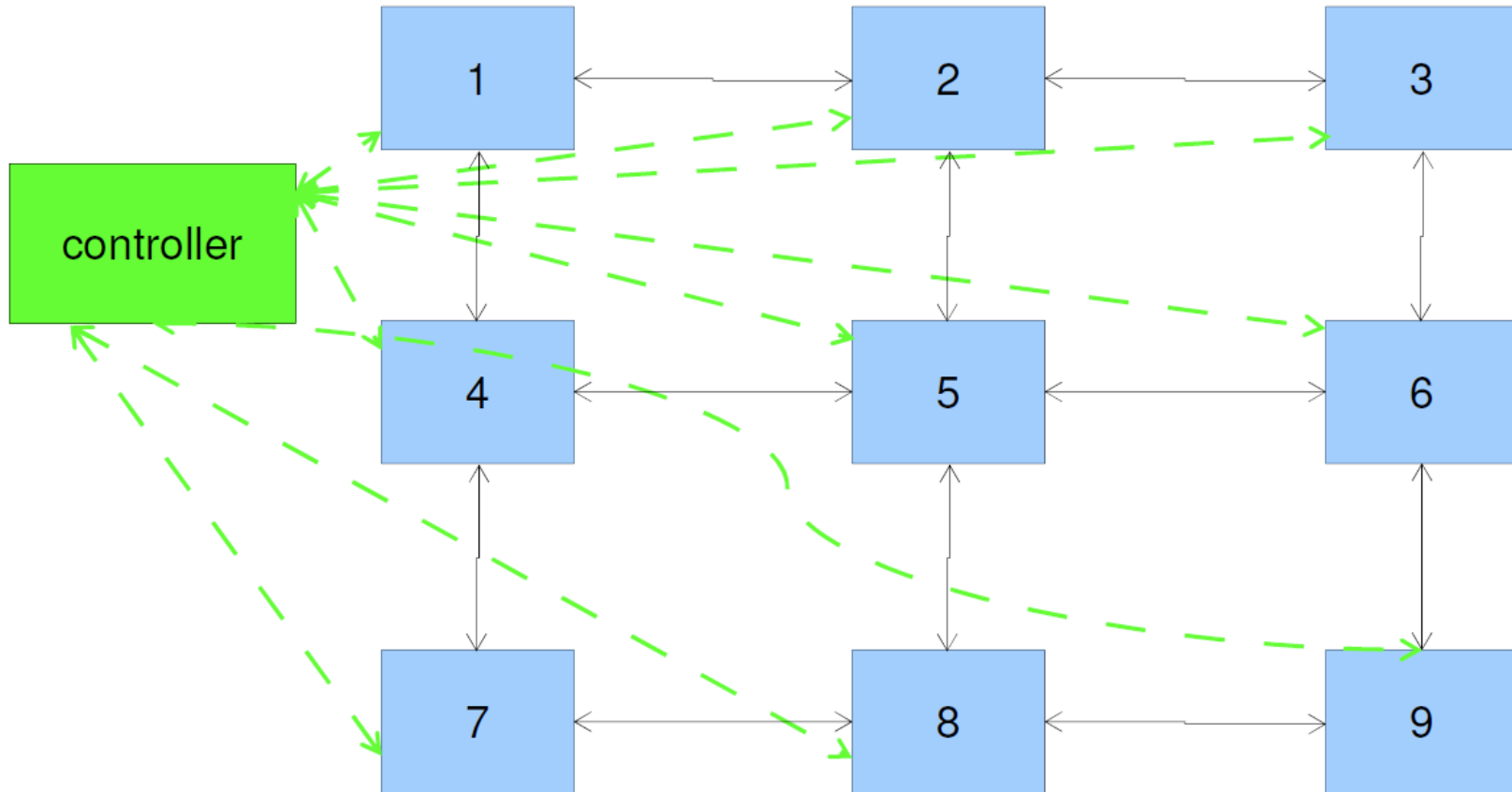
OpenFlow: Another Use

Effective topology with
normal L2 switching:



OpenFlow: Another Use

L2 routing managed by controller:



(Requires all switches to support OpenFlow)

Part I Summary

- ❖ Companies spread VMs across data centers.
- ❖ Need for VM isolation and virtual connectivity to VMs in remote data centers
- ❖ Solution: dynamic tunneling
- ❖ A controller and OpenFlow switches at the edge of the network can set up and maintain the tunnels.



Part II: Network Virtualization in Campus Networks

Can the Production Network be the Testbed?

Input from: Rob Sherwood, Big Switch Inc.

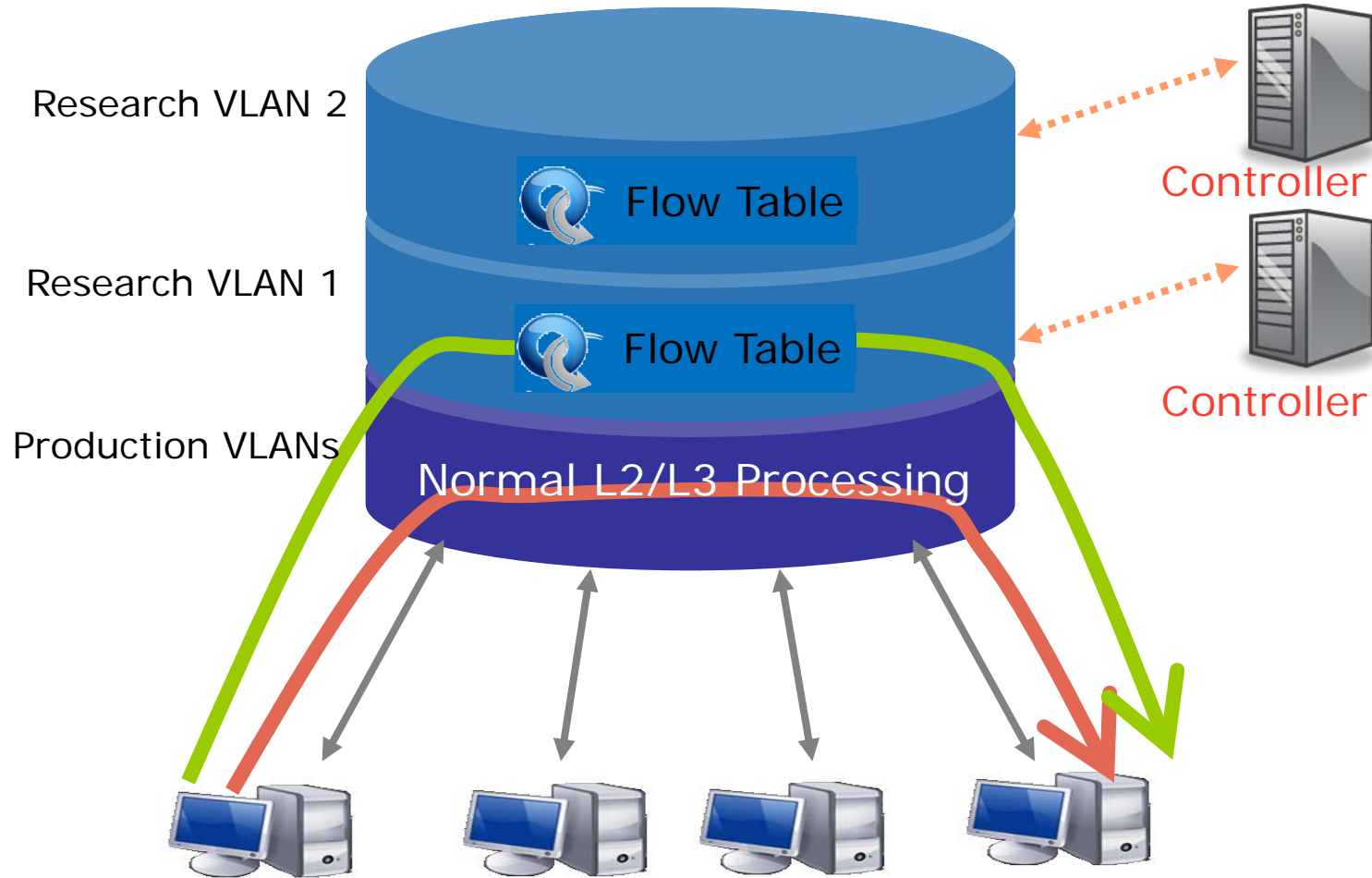
Problem

- ❖ Realistically evaluating new network services is hard
 - services that require changes to switches and routers, e.g.,
 - routing protocols
 - traffic monitoring services
 - IP mobility
- ❖ Many good ideas don't get deployed
- ❖ Many deployed services still have bugs

Solution Overview: Network Slicing

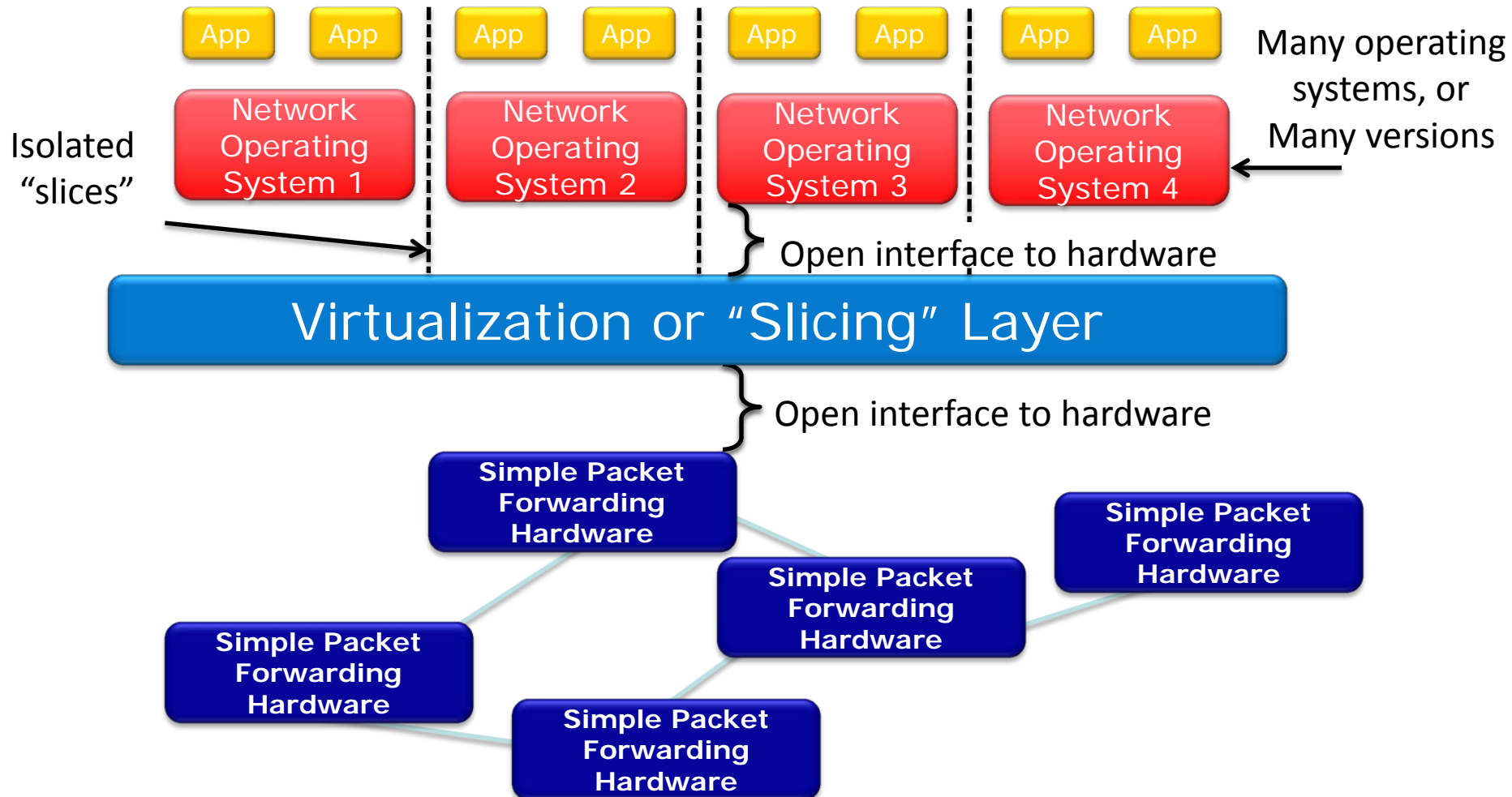
- ❖ Divide the production network into logical slices
- ❖ Enforce strong isolation between slices
- ❖ Allow the (logical) testbed to mirror the production network

Switch Based Virtualization



- ❖ How network slicing works?
FlowSpace, Opt-In
- ❖ Prototype implementation: FlowVisor

Virtualization or “Slicing” Layer



- ❖ A network slice is a collection of sliced switches/routers
 - Data plane is unmodified
 - Packets forwarded with no performance penalty
 - Slicing with existing ASIC

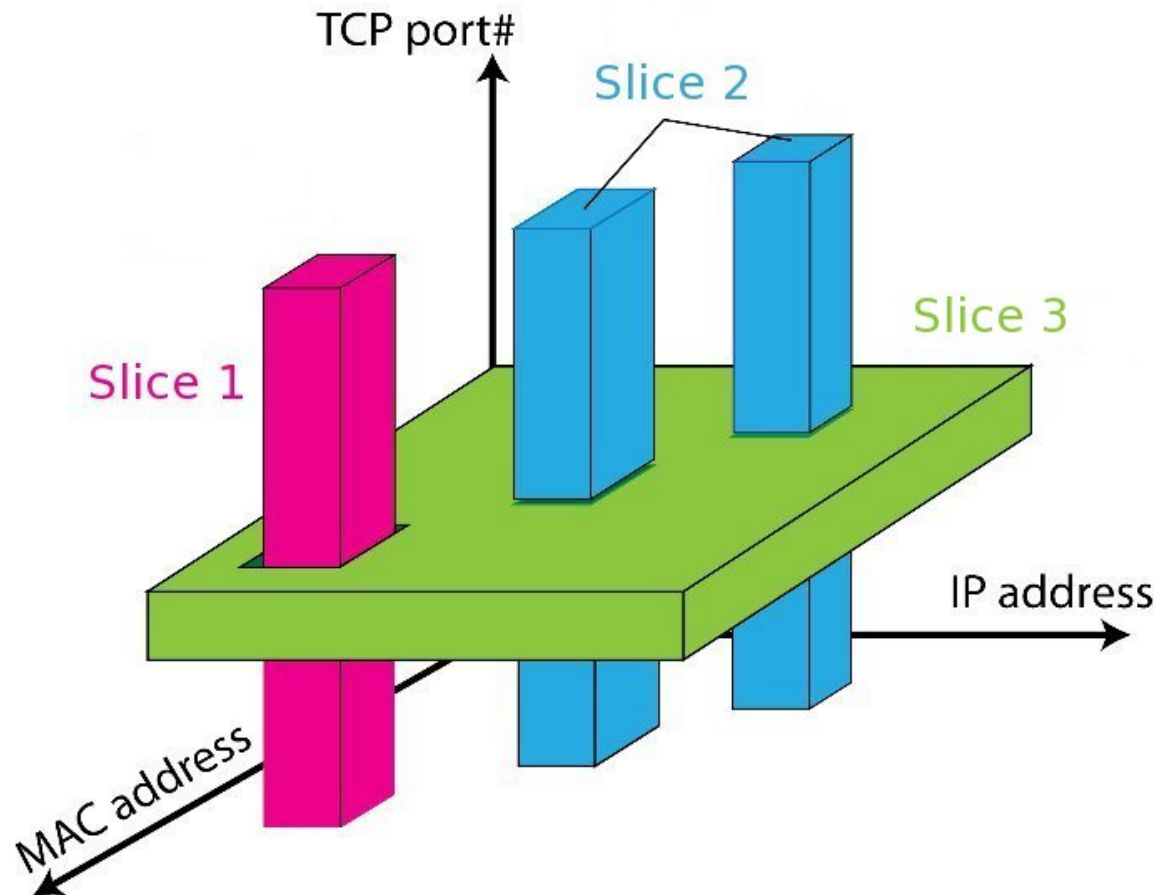
- ❖ Slicing layer
 - each slice believes it owns the data path
 - enforce isolation between slices
 - i.e., rewrites, drops rules to adhere to slice policy
 - forwards Packet-Ins to correct slice(s)

- ❖ For each slice, the policy specifies resource limits for:
 - FlowSpace: which packets does the slice control?
 - Link bandwidth
 - Number of forwarding rules
 - Fraction of switch/router CPU
 - Topology

FlowSpace: Maps Packets to Slices



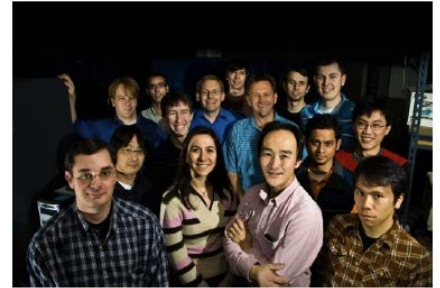
TECHNISCHE
UNIVERSITÄT
DARMSTADT



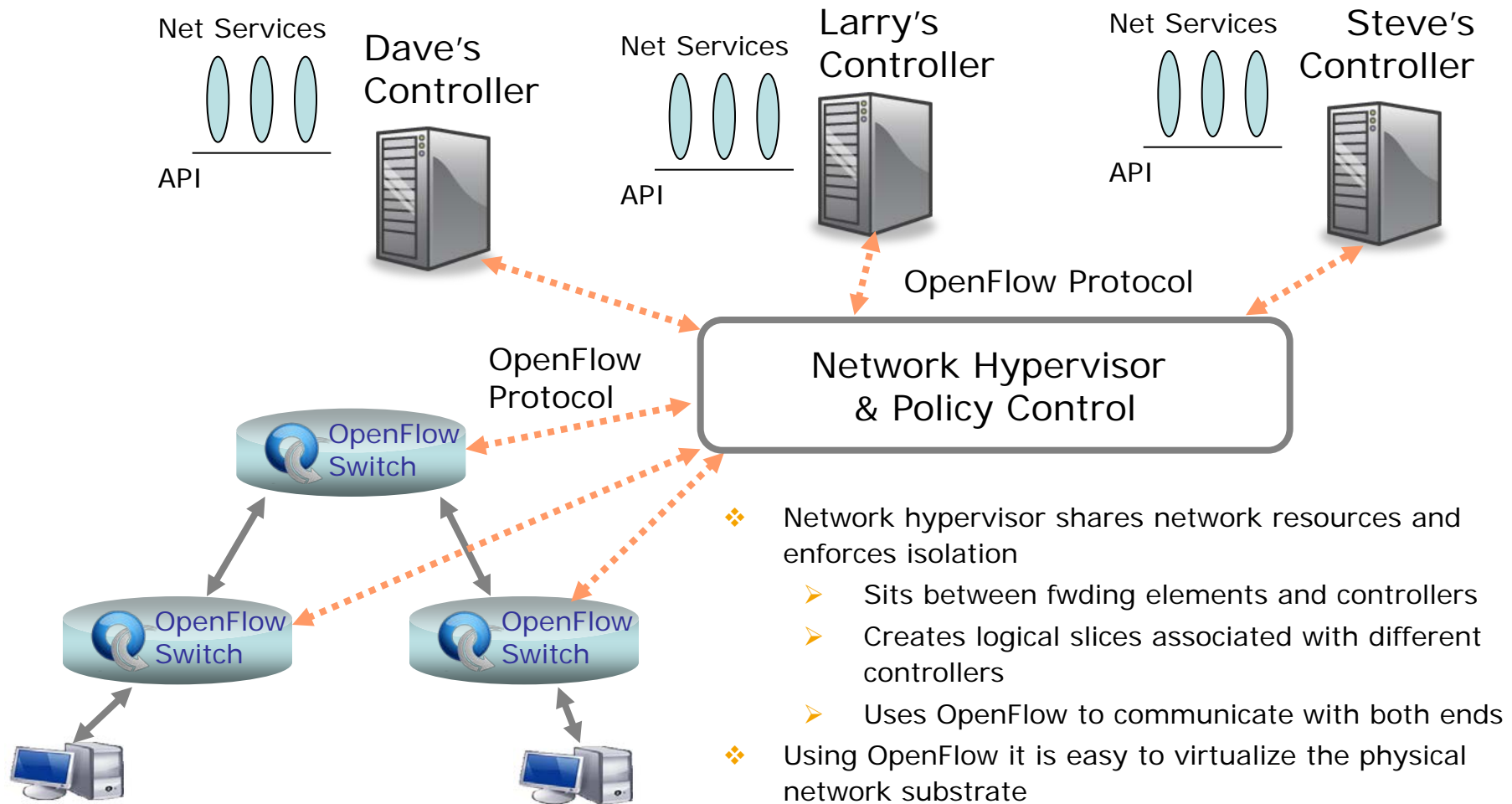
Real User Traffic: Opt-In

- ❖ 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

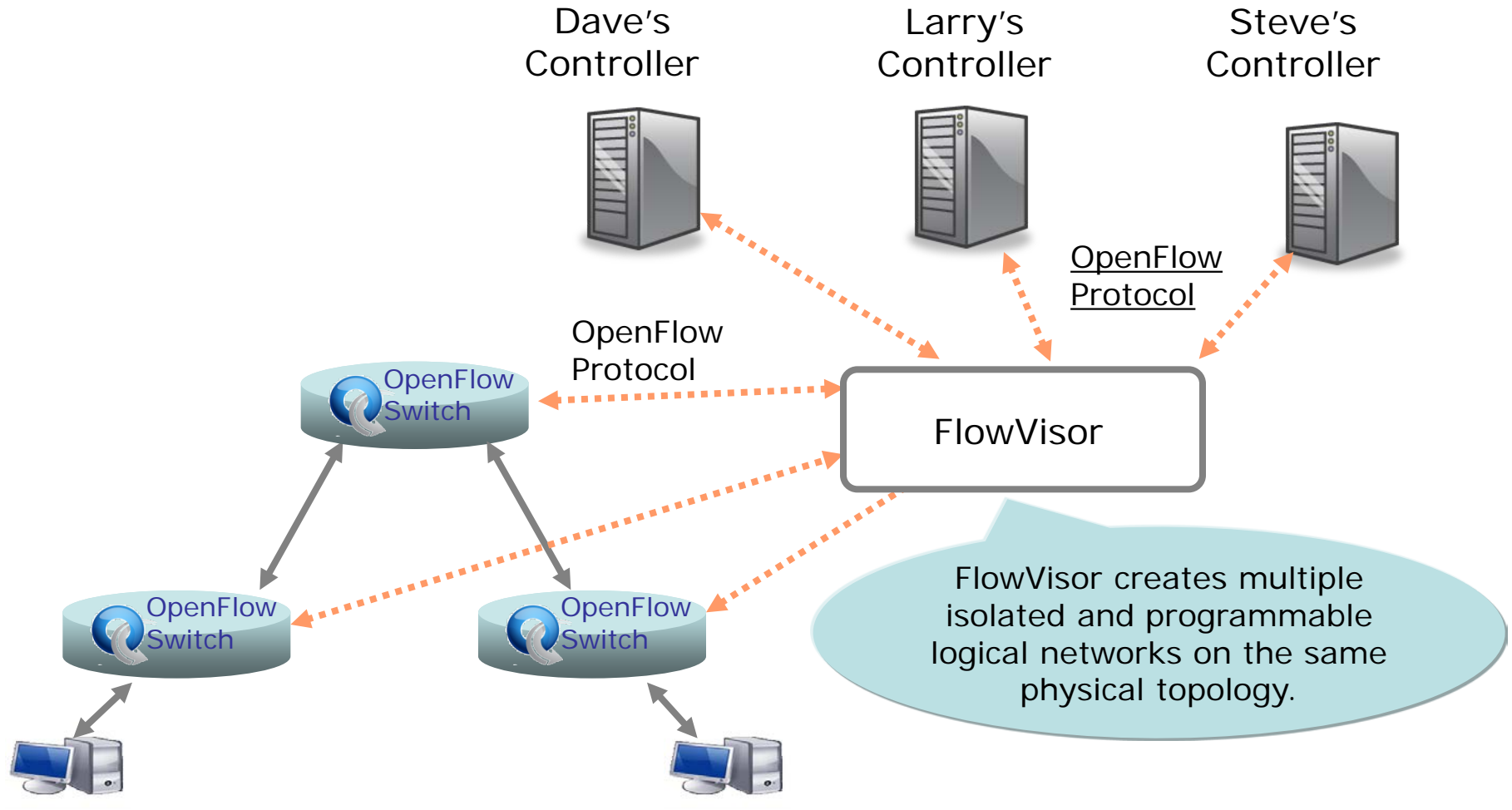
- ❖ Example:
 - "Slice 1 will handle my HTTP traffic"
 - "Slice 2 will handle my VoIP traffic"
 - "Slice 3 will handle everything else"



Virtualized OpenFlow Substrate



FlowVisor Slices OpenFlow Networks





Use Case: VLAN Based Partitioning

- ❖ Basic Idea: Partition Flows based on Ports and VLAN Tags
 - Traffic entering system (e.g. from end hosts) is tagged
 - VLAN tags consistent throughout substrate

	Switch Port	MAC src	MAC dst	Eth type	VLAN ID	IP Src	IP Dst	IP Prot	TCP sport	TCP dport
Dave	*	*	*	*	1,2,3	*	*	*	*	*
Larry	*	*	*	*	4,5,6	*	*	*	*	*
Steve	*	*	*	*	7,8,9	*	*	*	*	*

Use Case: New CDN - Turbo Coral ++

- ❖ Basic Idea: Build a CDN where you control the entire network
 - All traffic to or from Coral IP space controlled by Experimenter
 - All other traffic controlled by default routing
 - Topology is the entire network
 - End hosts are automatically added (no opt-in)

Switch Port	MAC src	MAC dst	Eth type	VLAN ID	IP Src	IP Dst	IP Prot	TCP sport	TCP dport
-------------	---------	---------	----------	---------	--------	--------	---------	-----------	-----------

Turbo
Coral

*	*	*	*	*	84.65.*	*	*	*	*
*	*	*	*	*	*	84.65.*	*	*	*

Default

*	*	*	*	*	*	*	*	*	*
---	---	---	---	---	---	---	---	---	---

Use Case: Your Internet Protocol

- ❖ A new layer 3 protocol
- ❖ Replaces IP
- ❖ Defined by a new Ether Type

Switch Port	MAC src	MAC dst	Eth type	VLAN ID	IP Src	IP Dst	IP Prot	TCP sport	TCP dport
-------------	---------	---------	----------	---------	--------	--------	---------	-----------	-----------

Your IP * * * YourIP * * * * * *

Rest * * * !YourIP * * * * * *

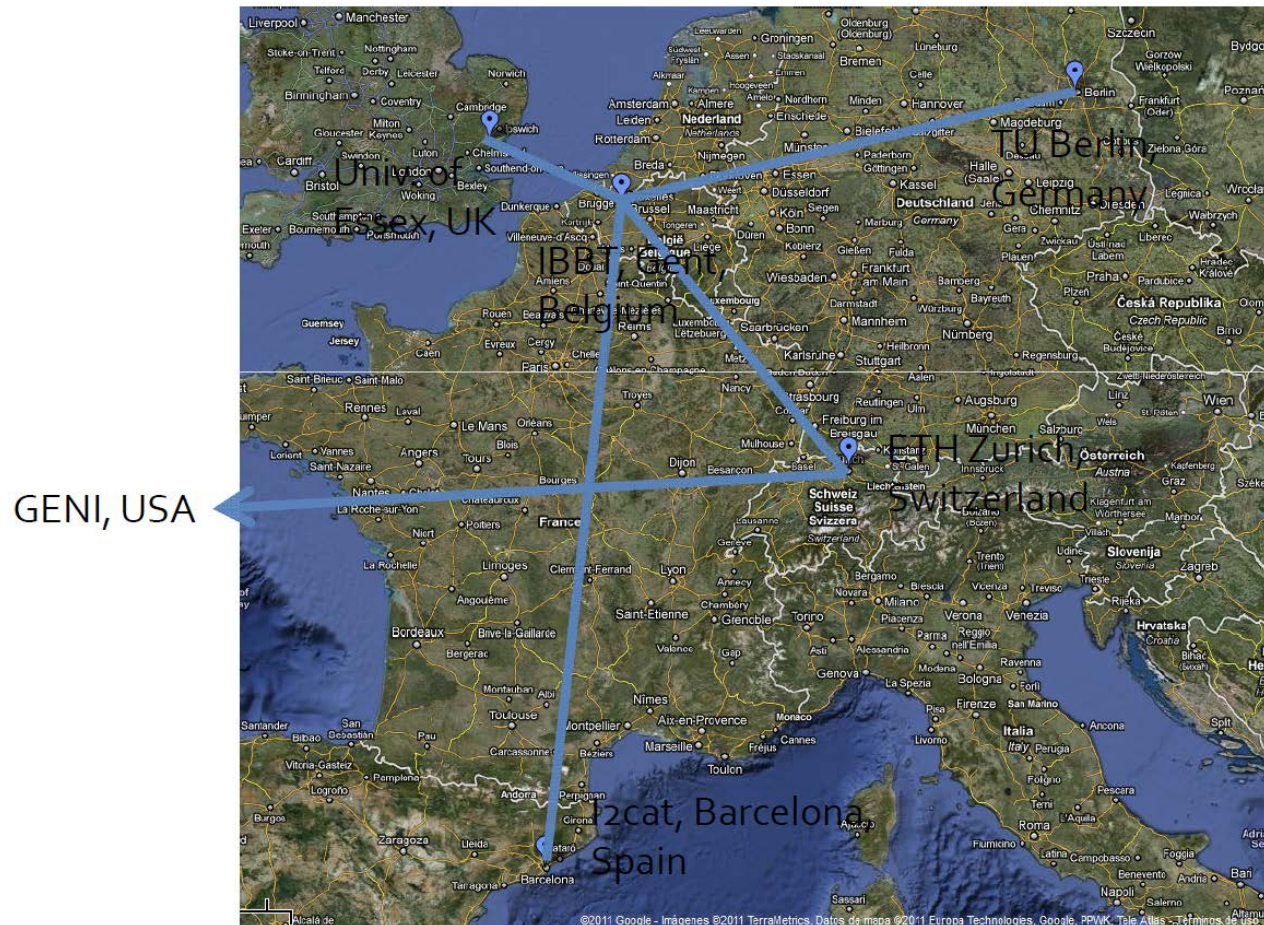
FlowVisor Deployment: Stanford

- ❖ Real production network
 - 15 switches, 35 APs
 - 25+ users
 - 1+ year of use
 - Personal email and web-traffic!

- ❖ Same physical network hosts Stanford demos
 - 7 different demos

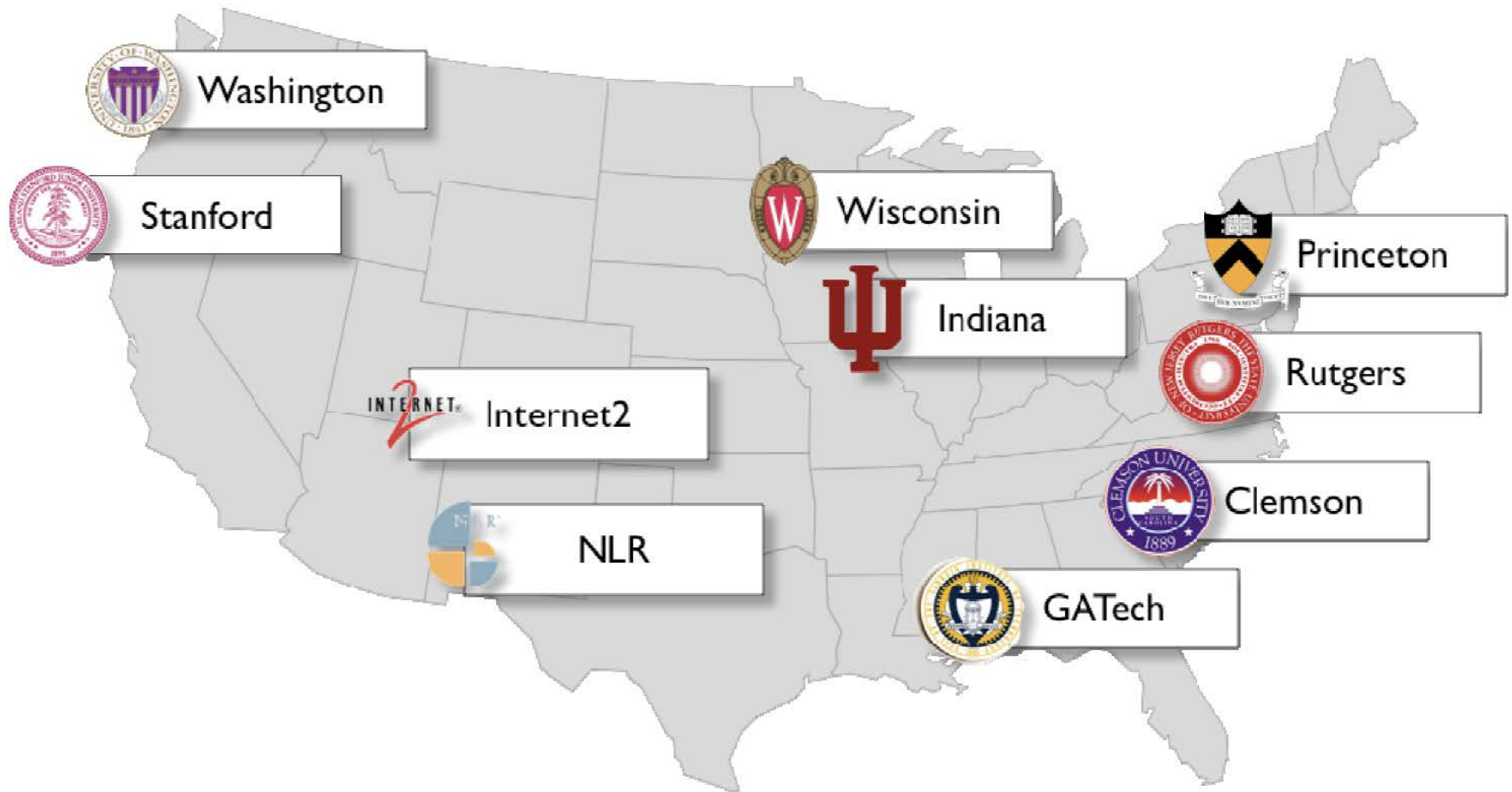


Flow Visor Deployments: OFELIA Testbed



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

FlowVisor Deployments: GENI



Part II Summary

- ❖ Network slicing can help perform more realistic evaluations
- ❖ FlowVisor allows experiments to run concurrently but safely on the production network