Networking in the Clouds: overview, state-of-the-art and research directions

Prof. Marinho Pilla Barcellos

Escuela de Computación de Venezuela
Venezuela–Brazil
Mérida, 30th September 2013

INSTITUTO DE INFORMÁTICA UFRGS

Acknowledgements

- Co-authors
  Daniel Stefani and Rodrigo Ruas Oliveira

What is this tutorial all about?

This course involves three main subjects:

- Cloud Computing
- Network Virtualization and Software-Defined Networks
- Cloud Networks

And their combination so that we can have efficient networking in the cloud.

Cloud Computing

Three new aspects

1. Elastic growth: appearance of infinite computing resources available on demand
2. Elimination of an up-front commitment by cloud users
3. Pay-per-use model: ability to pay for computing resources on a short-term basis as needed

[Armbrust et al., 2010]

Cloud Computing

Three new aspects

1. Elastic growth: appearance of infinite computing resources available on demand
2. Elimination of an up-front commitment by cloud users
3. Pay-per-use model: ability to pay for computing resources on a short-term basis as needed

Three use cases

- When demand varies with time
- When demand cannot be dimensioned in advance
- Run batch analytics faster at no extra cost, exploiting cost associativity

[Armbrust et al., 2010]

Key enablers for Cloud Computing

Operating System Virtualization is the foundation of cloud computing

[Abts and Felderman, 2012; Armbrust et al., 2010; Zhang et al., 2010]

Key enablers for Cloud Computing

Operating System Virtualization is the foundation of cloud computing

Warehouse-scale commodity-computer datacenters at low-cost locations

- **Very large scale** allows factors of 5-7 decrease in costs
- Savings in electricity, network bandwidth, operations, software, and hardware

[Abts and Felderman, 2012; Armbrust et al., 2010; Zhang et al., 2010]

Key enablers for Cloud Computing

Operating System Virtualization is the foundation of cloud computing

Warehouse-scale commodity-computer datacenters at low-cost locations

- **Very large scale** allows factors of 5-7 decrease in costs
- Savings in electricity, network bandwidth, operations, software, and hardware

Statistical multiplexing used to increase utilization compared to "production" data centers

Result: large-scale virtualization of resources allowed services to be offered at lower cost by a cloud provider with increased revenue

[Abts and Felderman, 2012; Armbrust et al., 2010; Zhang et al., 2010]

Virtualization: three kinds

- **Server**: multiple platforms into a single hardware, allows specifically tailored environments, increases availability
  - Full virtualization
  - Container-based
  - Paravirtualization
- **Network**: allows address isolation, coexistence of multiple protocol stacks, experimentation with real traffic etc.
- **Cloud**: virtualize the entire infrastructure containing servers, switches and links

Virtualization: three kinds

- **Server**: multiple platforms into a single hardware, allows specifically tailored environments, increases availability
  - Full virtualization
  - Container-based
  - Paravirtualization
- **Network**: allows address isolation, coexistence of multiple protocol stacks, experimentation with real traffic etc.
- **Cloud**: virtualize the entire infrastructure containing servers, switches and links

Our focus is on **network virtualization** and **cloud virtualization**

Questions

1. What are the three new, business aspects in cloud computing?
2. Name the economical factor(s) that allowed cloud computing to emerge and become so important.
3. Provide two technical factors that enabled cloud computing.
4. List and compare the three kinds of virtualization.

Outline

Outline

Key Points

- Definition
- Deployment models
- Types of services
- Main **private** clouds
- Main **public** clouds
- Applications
- Trends and research challenges

Definition

- There are many definitions of "cloud computing"
- Not a new technology
- Actually a "new operations model that brings together a set of existing technologies to run business in a different way"

NIST definition

Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction.

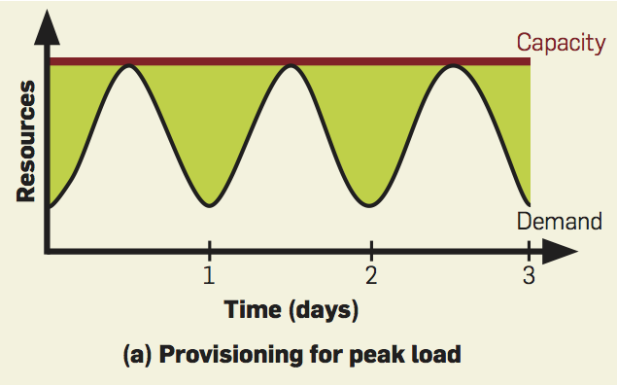
[Badger et al., 2009; Zhang et al., 2010]

Defining characteristics

- **Resource pooling**: serves multiple customers using a **multi-tenant** model
- On-demand **automatic provision** of resources
- Ubiquitous network access: access through standard mechanisms
- **Utility computing**: packaging of computing resources as a metered service, pay-per-use model
- **Elasticity**: capabilities can be dynamically scaled up or down

[Armbrust et al., 2010; Badger et al., 2009; Zhang et al., 2010]

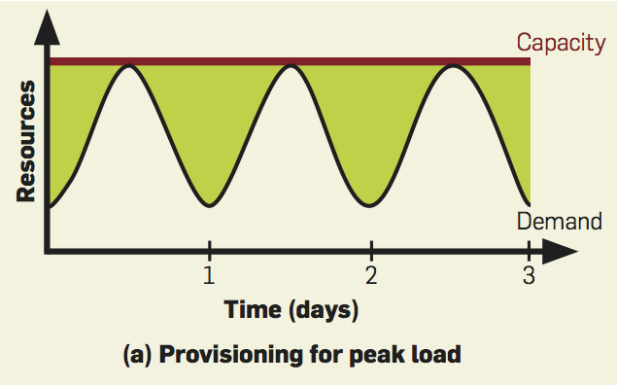
Utility Computing and Elasticity



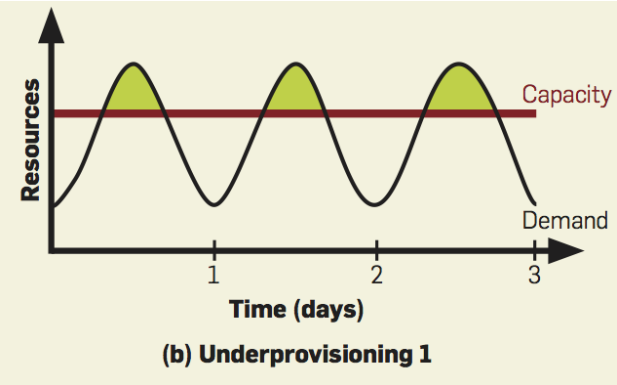
Even if the load is known, capacity is wasted

[Armbrust et al., 2010]

Utility Computing and Elasticity



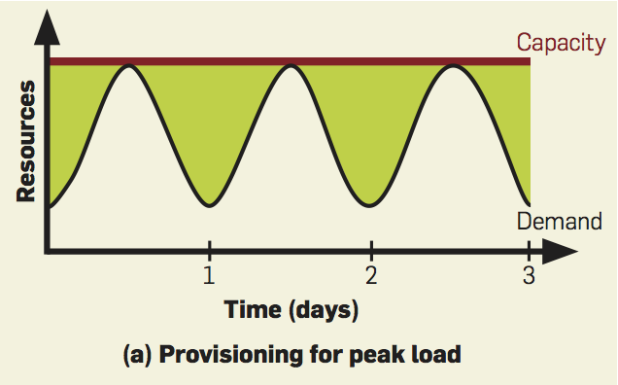
Even if the load is known, capacity is wasted



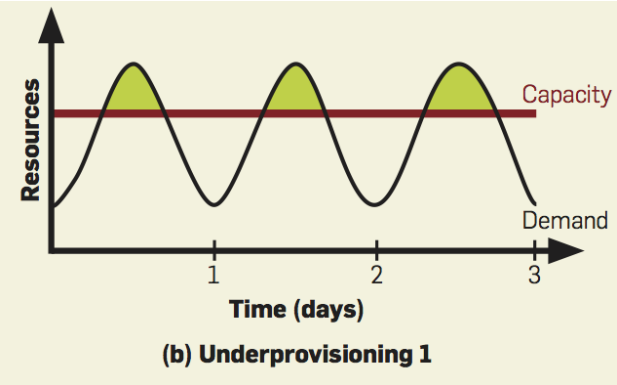
Understimating the peak load provides poor Quality of Experience

[Armbrust et al., 2010]

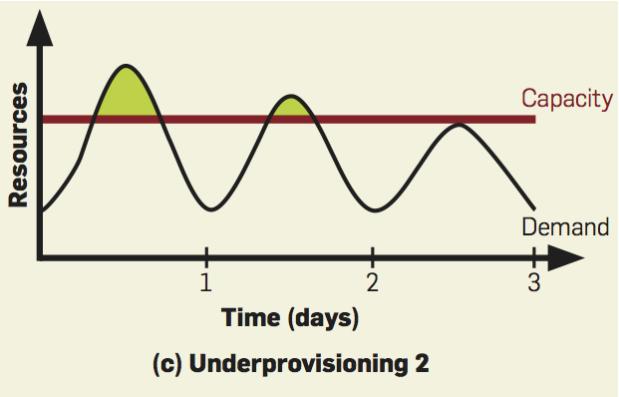
Utility Computing and Elasticity



Even if the load is known, capacity is wasted



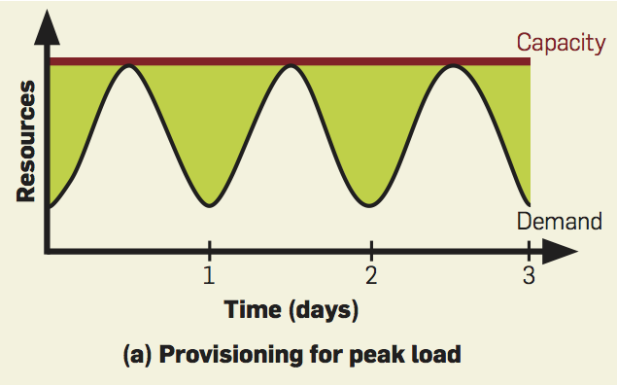
Understimating the peak load provides poor Quality of Experience



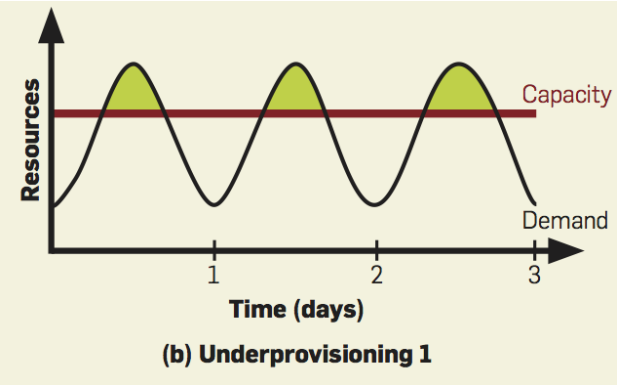
Bad Quality of Experience at peaks turns away users, which causes revenue loss (users defect)

[Armbrust et al., 2010]

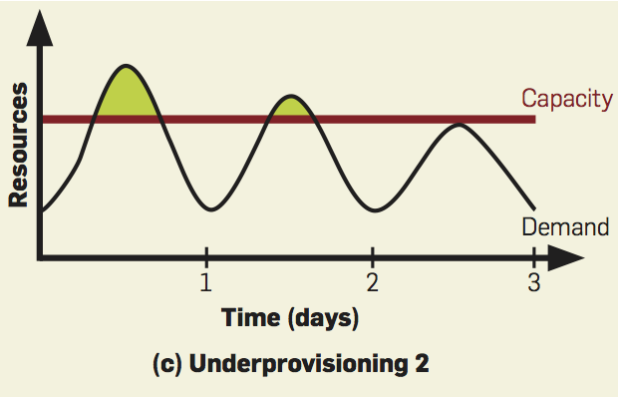
Utility Computing and Elasticity



Even if the load is known, capacity is wasted



Understimating the peak load provides poor Quality of Experience



Bad Quality of Experience at peaks turns away users, which causes revenue loss (users defect)

Question: is this question relevant to all kinds of clouds? which types exist?

[Armbrust et al., 2010]

Types of Clouds: deployment models

- **Private**: on-site or outsourced, exclusive for a single organization

[Badger et al., 2009; Zhang et al., 2010]

Types of Clouds: deployment models

- **Private**: on-site or outsourced, exclusive for a single organization
- **Community**: on-site or outsourced, shared among two or more mutually trusting organizations

[Badger et al., 2009; Zhang et al., 2010]

Types of Clouds: deployment models

- **Private**: on-site or outsourced, exclusive for a single organization
- **Community**: on-site or outsourced, shared among two or more mutually trusting organizations
- **Public**: at data centers owned by provider with unknown physical location, open to all registered users (pay-per-use)

[Badger et al., 2009; Zhang et al., 2010]

Types of Clouds: deployment models

- **Private**: on-site or outsourced, exclusive for a single organization
- **Community**: on-site or outsourced, shared among two or more mutually trusting organizations
- **Public**: at data centers owned by provider with unknown physical location, open to all registered users (pay-per-use)
- **Hybrid**: combination of multiple clouds, typically private and public

[Badger et al., 2009; Zhang et al., 2010]

Types of Clouds: deployment models

- **Private**: on-site or outsourced, exclusive for a single organization
- **Community**: on-site or outsourced, shared among two or more mutually trusting organizations
- **Public**: at data centers owned by provider with unknown physical location, open to all registered users (pay-per-use)
- **Hybrid**: combination of multiple clouds, typically private and public
- **Virtual Private**: a section of a public cloud is separated and dedicated to a tenant, allowing tenant to configure it, and potentially connected through a VPN to the corporate data center

[Badger et al., 2009; Zhang et al., 2010]

Types of Clouds: deployment models

- **Private**: on-site or outsourced, exclusive for a single organization
- **Community**: on-site or outsourced, shared among two or more mutually trusting organizations
- **Public**: at data centers owned by provider with unknown physical location, open to all registered users (pay-per-use)
- **Hybrid**: combination of multiple clouds, typically private and public
- **Virtual Private**: a section of a public cloud is separated and dedicated to a tenant, allowing tenant to configure it, and potentially connected through a VPN to the corporate data center

All these cloud deployments can offer one or more **service models**

[Badger et al., 2009; Zhang et al., 2010]

Types of Clouds: service models

- **IaaS**: Infrastructure as a Service
  - closest level to the hardware
  - services are resources such as CPU, memory, and storage
  - examples: Amazon EC2, GoGrid and Flexiscale

[Zhang et al., 2010]

Types of Clouds: service models

- **IaaS**: Infrastructure as a Service
  - closest level to the hardware
  - services are resources such as CPU, memory, and storage
  - examples: Amazon EC2, GoGrid and Flexiscale
- **PaaS**: Platform as a Service
  - uses resources from IaaS
  - provides specific environments to develop cloud applications
  - examples: Google App Engine, MS Win Azure and Force.com

[Zhang et al., 2010]

Types of Clouds: service models

- **IaaS**: Infrastructure as a Service
  - closest level to the hardware
  - services are resources such as CPU, memory, and storage
  - examples: Amazon EC2, GoGrid and Flexiscale
- **PaaS**: Platform as a Service
  - uses resources from IaaS
  - provides specific environments to develop cloud applications
  - examples: Google App Engine, MS Win Azure and Force.com
- **SaaS**: Software as a Service
  - uses the environment from PaaS
  - provides applications for customers
  - examples: Salesforce.com, Rackspace and SAP Business ByDesign

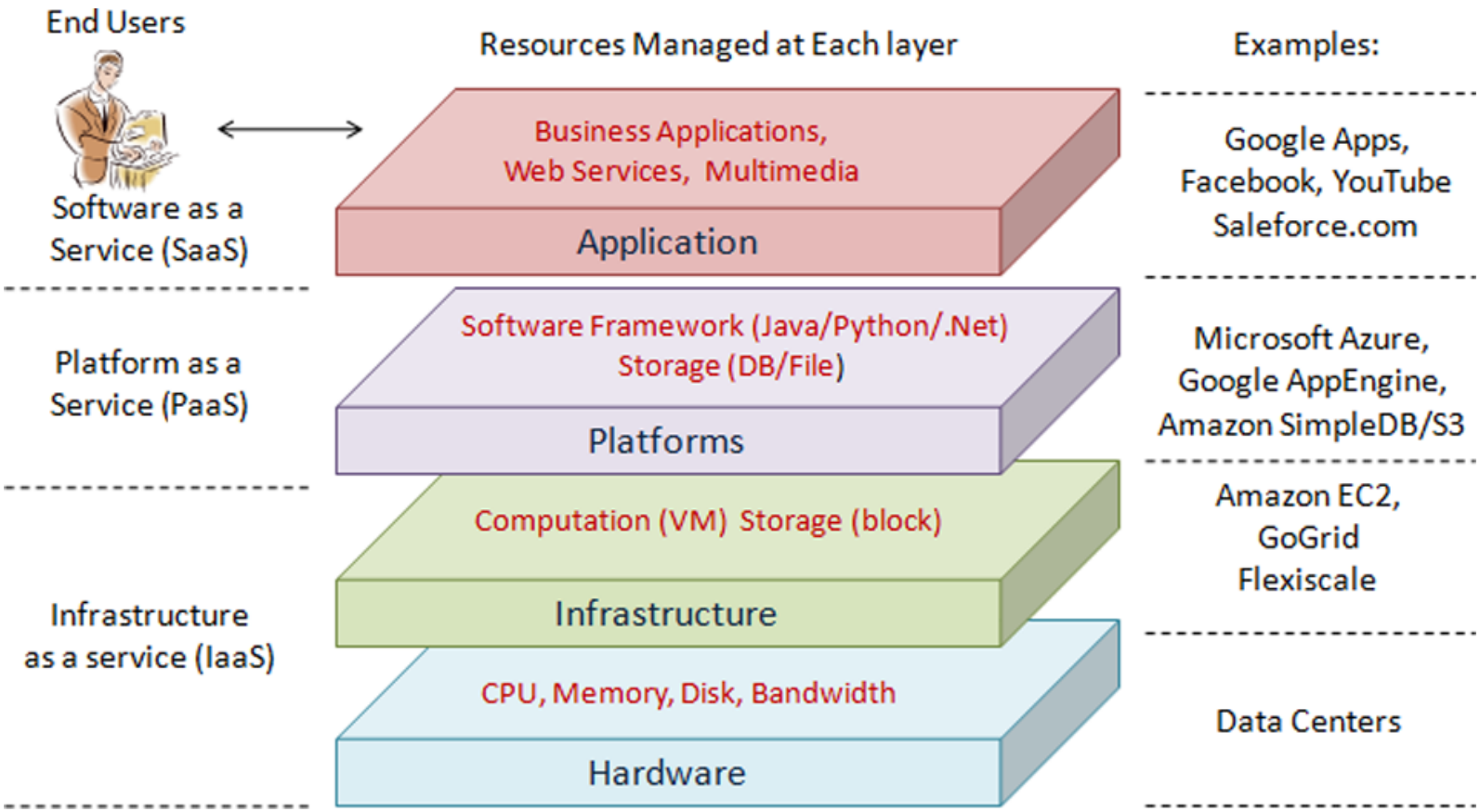
[Zhang et al., 2010]

Types of Clouds: service models

- **IaaS**: Infrastructure as a Service
  - closest level to the hardware
  - services are resources such as CPU, memory, and storage
  - examples: Amazon EC2, GoGrid and Flexiscale
- **PaaS**: Platform as a Service
  - uses resources from IaaS
  - provides specific environments to develop cloud applications
  - examples: Google App Engine, MS Win Azure and Force.com
- **SaaS**: Software as a Service
  - uses the environment from PaaS
  - provides applications for customers
  - examples: Salesforce.com, Rackspace and SAP Business ByDesign
- **DaaS**: Data as a service, **STaaS**: Storage as a service...

[Zhang et al., 2010]

A Layered Model



[Zhang et al., 2010]

Private Clouds

In private clouds, infrastructure layer (virtualization layer):

- **Eucalyptus**
- OpenNebula
- OpenStack
- Nimbus
- OpenQRM

http://www.eucalyptus.com/

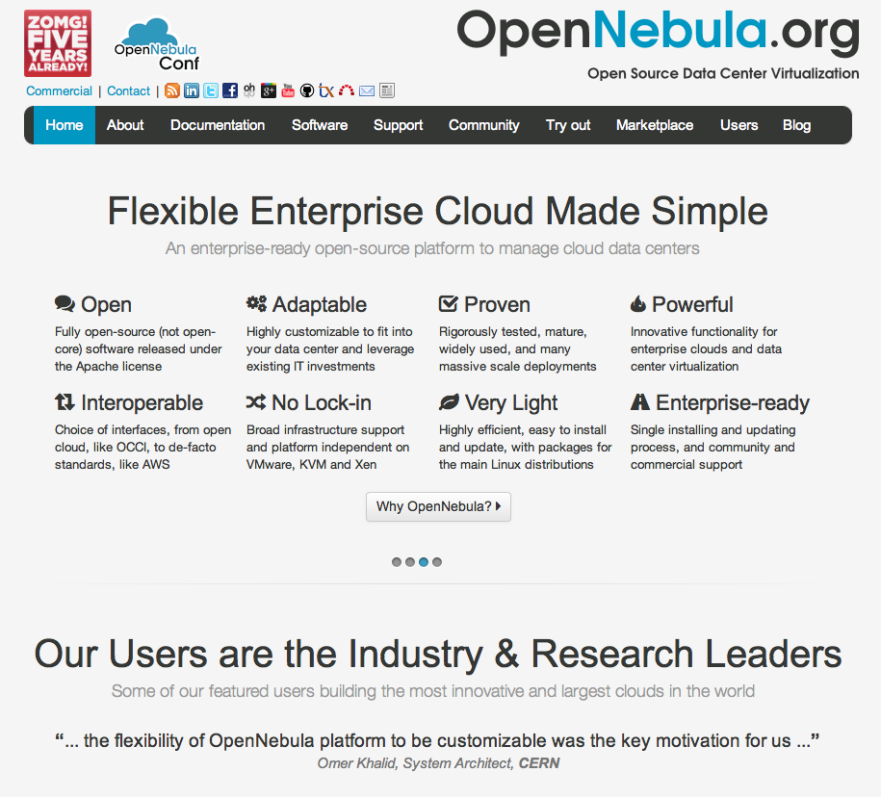
[Nurmi et al., 2009; OpenNebula, 2012; Sempolinski and Thain, 2010]

[Nimbus, 2012; OpenQRM, 2012; OpenStack, 2012]

Private Clouds

In private clouds, infrastructure layer (virtualization layer):

- Eucalyptus
- **OpenNebula**
- OpenStack
- Nimbus
- OpenQRM



http://www.opennebula.org/

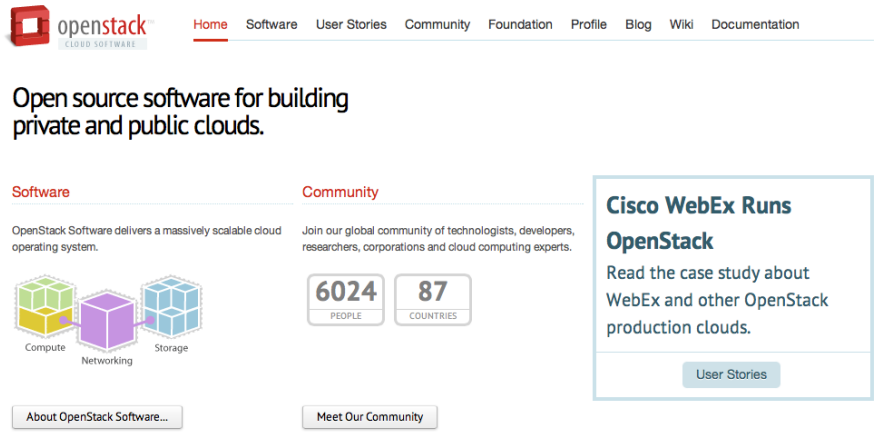
[Nurmi et al., 2009; OpenNebula, 2012; Sempolinski and Thain, 2010]

[Nimbus, 2012; OpenQRM, 2012; OpenStack, 2012]

Private Clouds

In private clouds, infrastructure layer (virtualization layer):

- Eucalyptus
- OpenNebula
- **OpenStack**
- Nimbus
- OpenQRM



http://www.openstack.org/

[Nurmi et al., 2009; OpenNebula, 2012; Sempolinski and Thain, 2010]

[Nimbus, 2012; OpenQRM, 2012; OpenStack, 2012]

Private Clouds

In private clouds, infrastructure layer (virtualization layer):

- Eucalyptus
- OpenNebula
- OpenStack
- **Nimbus**
- OpenQRM

http://www.nimbusproject.org/

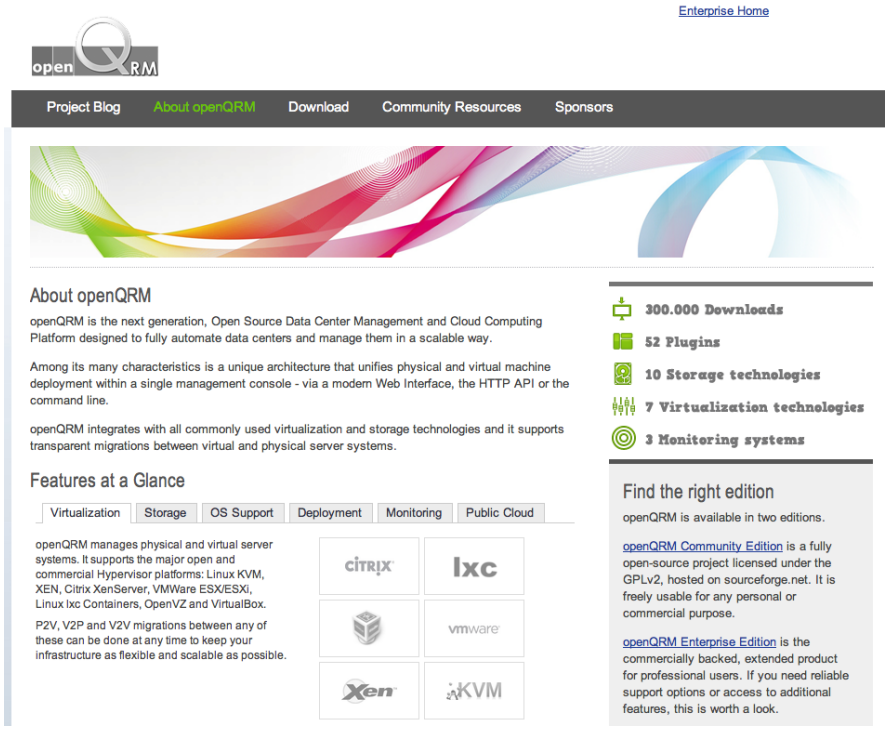
[Nurmi et al., 2009; OpenNebula, 2012; Sempolinski and Thain, 2010]

[Nimbus, 2012; OpenQRM, 2012; OpenStack, 2012]

Private Clouds

In private clouds, infrastructure layer (virtualization layer):

- Eucalyptus
- OpenNebula
- OpenStack
- Nimbus
- **OpenQRM**



http://www.openqrm.org/

[Nurmi et al., 2009; OpenNebula, 2012; Sempolinski and Thain, 2010]

[Nimbus, 2012; OpenQRM, 2012; OpenStack, 2012]

Public Clouds

- Amazon Web Services (AWS)
  - infrastructure located at NA, SA, EU and AS
  - set of cloud services (EC2, S3, Glacier, SimpleDB. . .)
  - accessible over HTTP REST and SOAP protocols

Public Clouds

- Amazon Web Services (AWS)
  - infrastructure located at NA, SA, EU and AS
  - set of cloud services (EC2, S3, Glacier, SimpleDB. . .)
  - accessible over HTTP REST and SOAP protocols
- Microsoft Windows Azure Platform
  - data centers located at NA, EU and AS
  - three components: Windows Azure, SQL Azure and .NET Services

Public Clouds

- Amazon Web Services (AWS)
  - infrastructure located at NA, SA, EU and AS
  - set of cloud services (EC2, S3, Glacier, SimpleDB. . .)
  - accessible over HTTP REST and SOAP protocols
- Microsoft Windows Azure Platform
  - data centers located at NA, EU and AS
  - three components: Windows Azure, SQL Azure and .NET Services
- Google AppEngine (GAE)
  - data centers located somewhere in NA (?)
  - platform for web applications

Public Clouds

- Amazon Web Services (AWS)
  - infrastructure located at NA, SA, EU and AS
  - set of cloud services (EC2, S3, Glacier, SimpleDB. . .)
  - accessible over HTTP REST and SOAP protocols
- Microsoft Windows Azure Platform
  - data centers located at NA, EU and AS
  - three components: Windows Azure, SQL Azure and .NET Services
- Google AppEngine (GAE)
  - data centers located somewhere in NA (?)
  - platform for web applications
- Rackspace Cloud
  - data centers located at NA and EU
  - set of cloud services (servers, platform, storage, load balancers, . . .)

Cloud Applications

Examples

- Storage-hungry: Dropbox, iCloud
- Web: search, email, document collaboration
- Video streaming: Netflix, Hulu
- High Performance Computing: MapReduce, Scientific computing

[Abts and Felderman, 2012; Armbrust et al., 2010]

Cloud Applications

Examples

- Storage-hungry: Dropbox, iCloud
- Web: search, email, document collaboration
- Video streaming: Netflix, Hulu
- High Performance Computing: MapReduce, Scientific computing

Diversity

- Own specific requirements: storage, network, cpu, memory
- Can be divided in two major classes: "user-facing" and "inward"

[Abts and Felderman, 2012; Armbrust et al., 2010]

Cloud Applications

User-facing computation

- Frequent message exchange with users
- Some require strict performance guarantees
- Web services, authentication services

Inward computation

- VMs from the same application are tightly coupled
- Orchestrated communication
- Usually require predictable intra-cloud network performance
- MapReduce, scientific computing

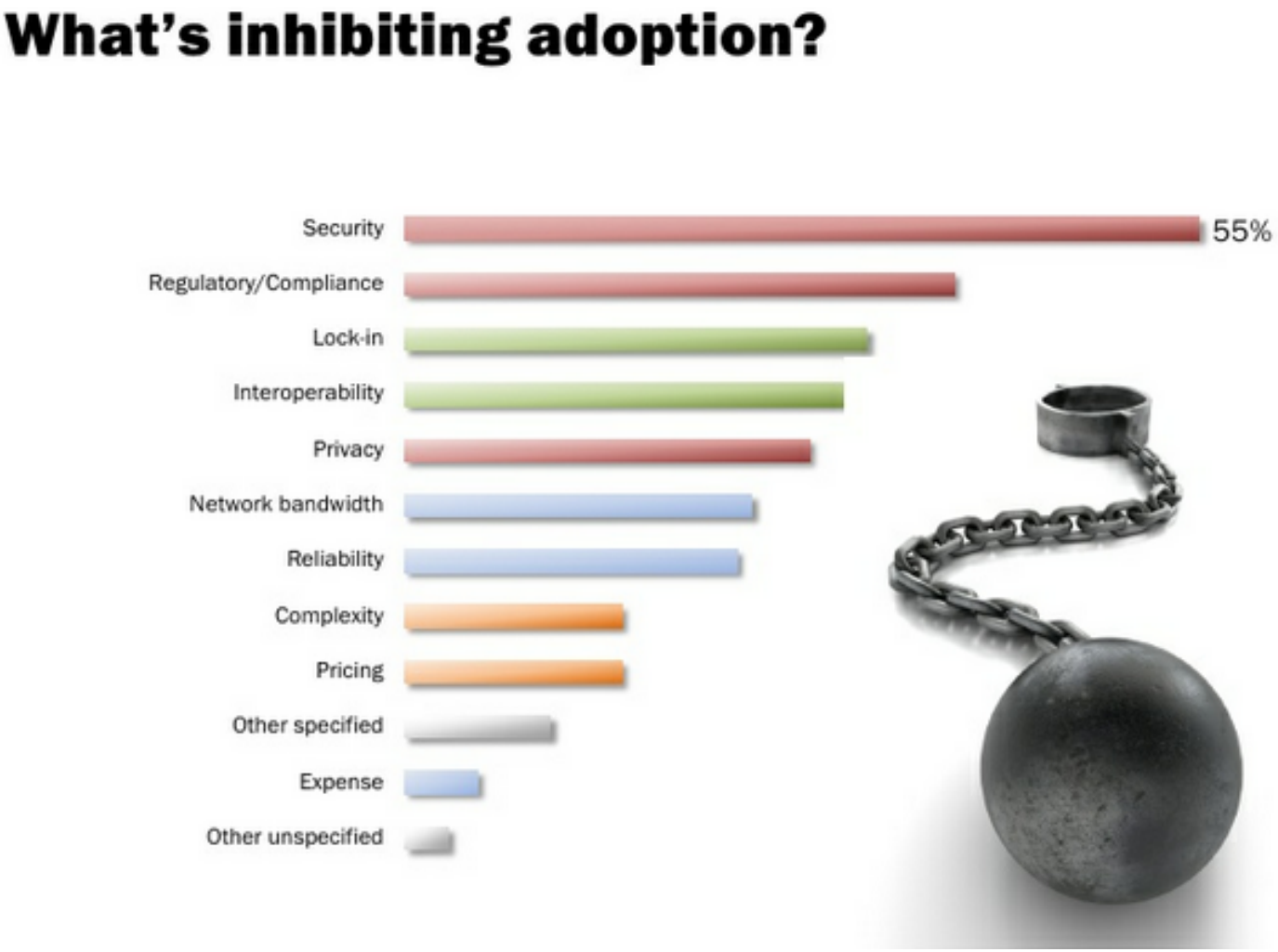
[Abts and Felderman, 2012; Benson et al., 2010]

Trends in Cloud Computing

- Cloud-centric application design
- Hybrid cloud utilization and management
- Service Level Agreement from providers becomes a key buying criterion
- Cloud services brokerage (CSB)
- Management of network resources (in addition to computational ones), application performance
- Addressing of privacy, compliance and governance issues

[Hurwitz, 2011; Pettey, 2012; Popa et al., 2012]

Cloud Adoption



[Nusca, 2012]

Research Challenges

Some examples

- Security: VM co-residence vulnerabilities, data privacy and integrity, virtual machine image integrity
- Virtual machine migration: VM state transmission from one physical machine to another
- Mutual auditability: provider audits customer's activities, customer audits what a provider provides

[Ballani et al., 2011a; Bowers et al., 2009, 2011; Bugiel et al., 2011; Chen et al., 2010; Popa et al., 2012]
[Ristenpart et al., 2009; Subashini and Kavitha, 2011; Xie et al., 2012; Zhang et al., 2011, 2010]

Questions

1. What are the most important properties of clouds?
2. In which ways clouds can be classified?
3. Provide three examples of public clouds and their services.
4. What is the difference between the services offered by Amazon EC2 and Google AppEngine?
5. What are the two main classes of cloud applications? Provide one example of each.
6. List three open challenges in cloud computing.

Outline

Key Points

- Definition
- Major benefits
- VLANs
- VPNs
- Full Network Virtualization
- Application-layer Overlays
- Trends and research challenges

Definition

- Consider a **physical network** of forwarding devices interconnected by different link technologies
- Forwarding devices offer some flavor of virtualization support

[Belbekkouche et al., 2012; Chowdhury and Boutaba, 2010]

Definition

- Consider a **physical network** of forwarding devices interconnected by different link technologies
- Forwarding devices offer some flavor of virtualization support
- **Virtual networks** are placed on top of physical network elements
  - virtual nodes mapped to forwarding devices
  - virtual links mapped to paths between these devices

[Belbekkouche et al., 2012; Chowdhury and Boutaba, 2010]

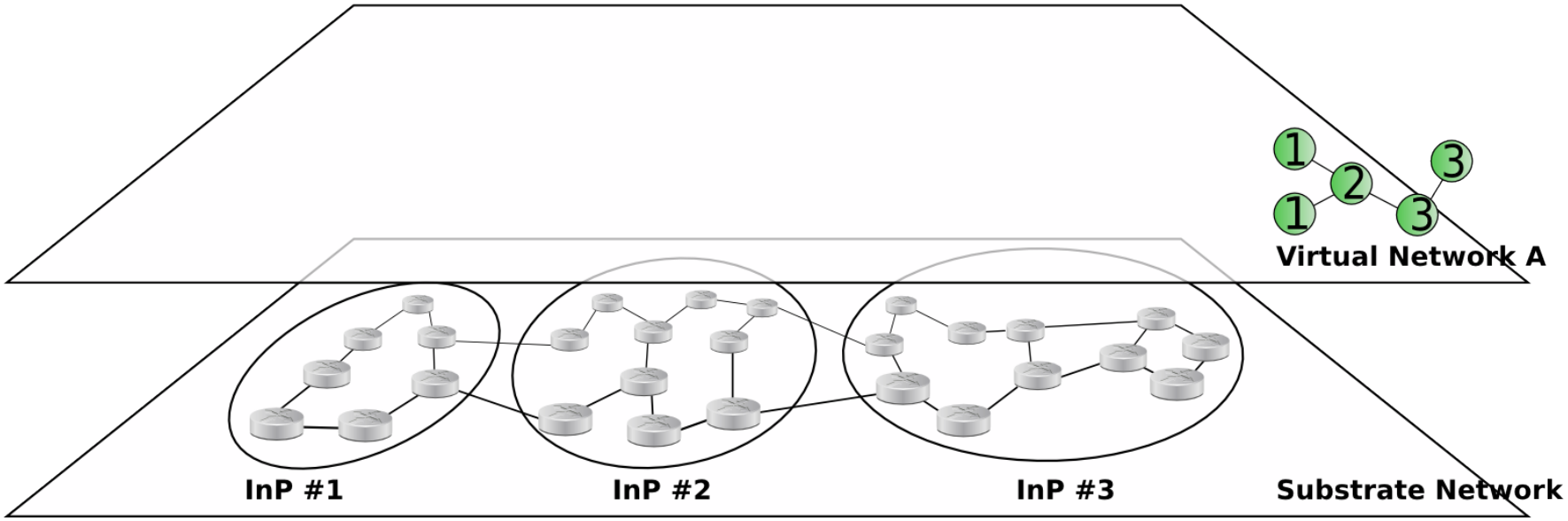
Definition

- Consider a **physical network** of forwarding devices interconnected by different link technologies
- Forwarding devices offer some flavor of virtualization support
- **Virtual networks** are placed on top of physical network elements
  - virtual nodes mapped to forwarding devices
  - virtual links mapped to paths between these devices

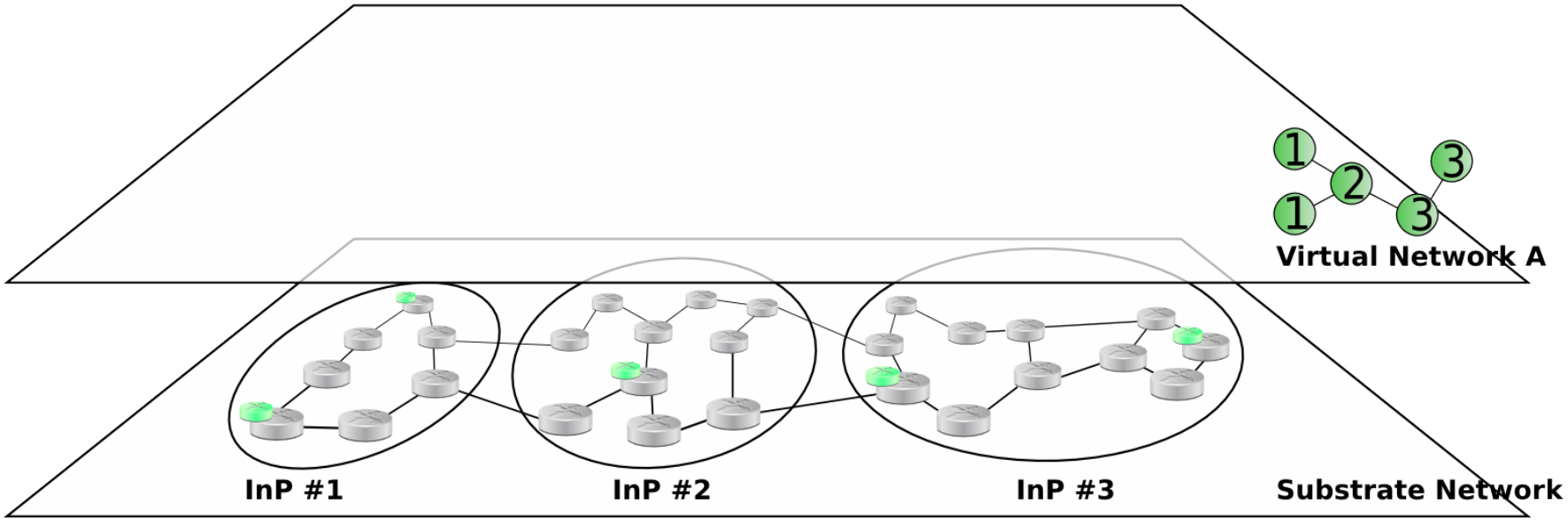
- Physical and virtual nets can be provided by different entities
  - **InP**: Infrastructure Providers lease physical network resources
  - **SPs**: Service Providers build and manage their virtual networks based on specific application requirements
  - **VNO**: Virtual Network Operators (a broker)

[Belbekkouche et al., 2012; Chowdhury and Boutaba, 2010]

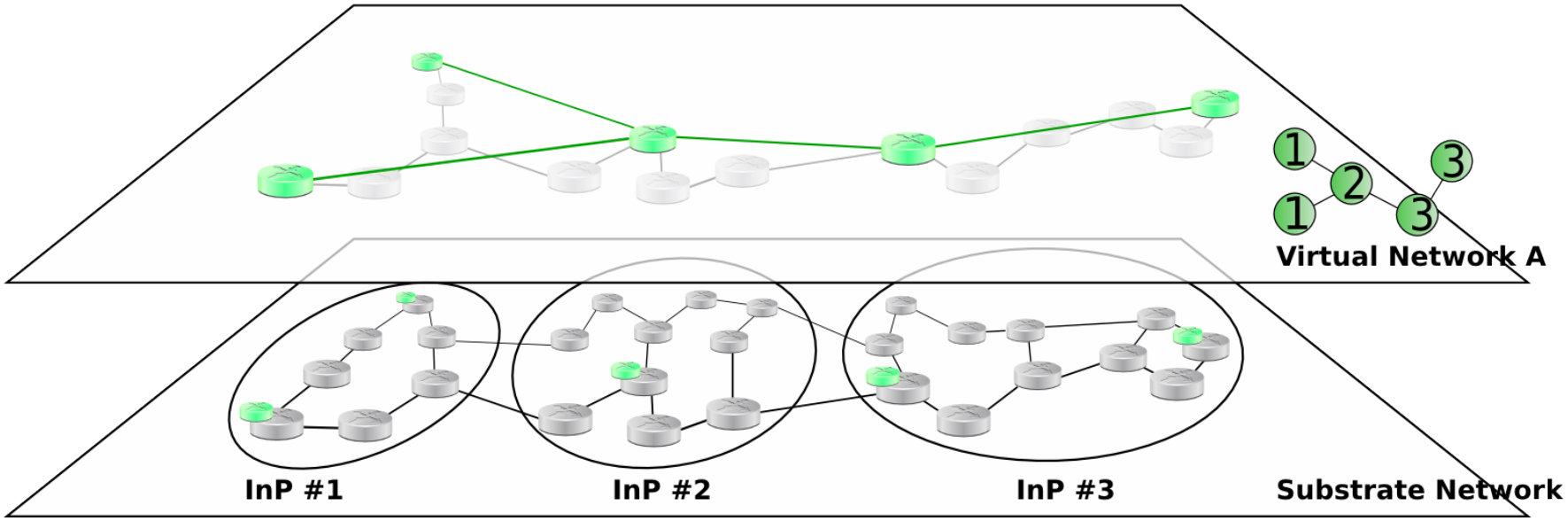
Definition: a simple scenario



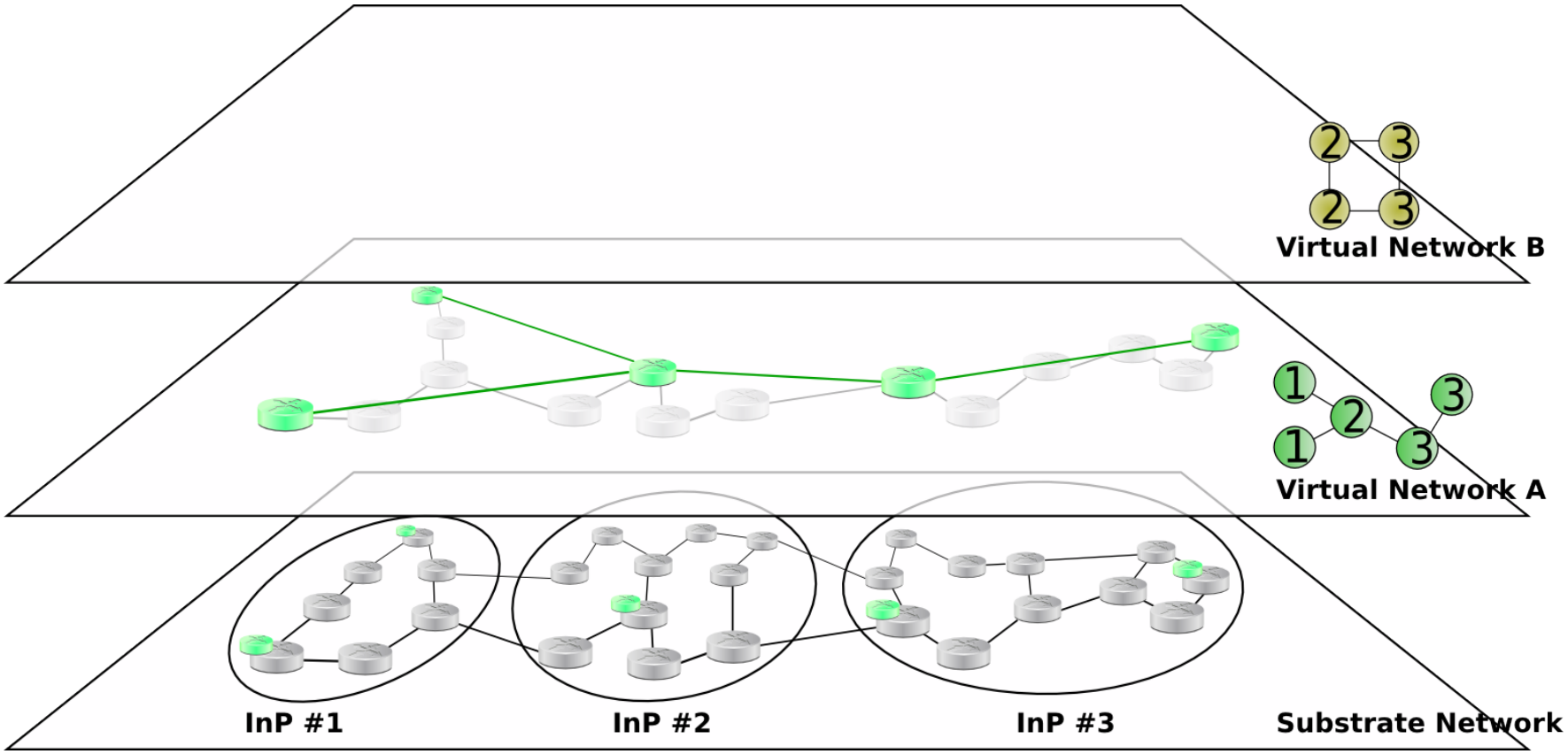
Definition: a simple scenario



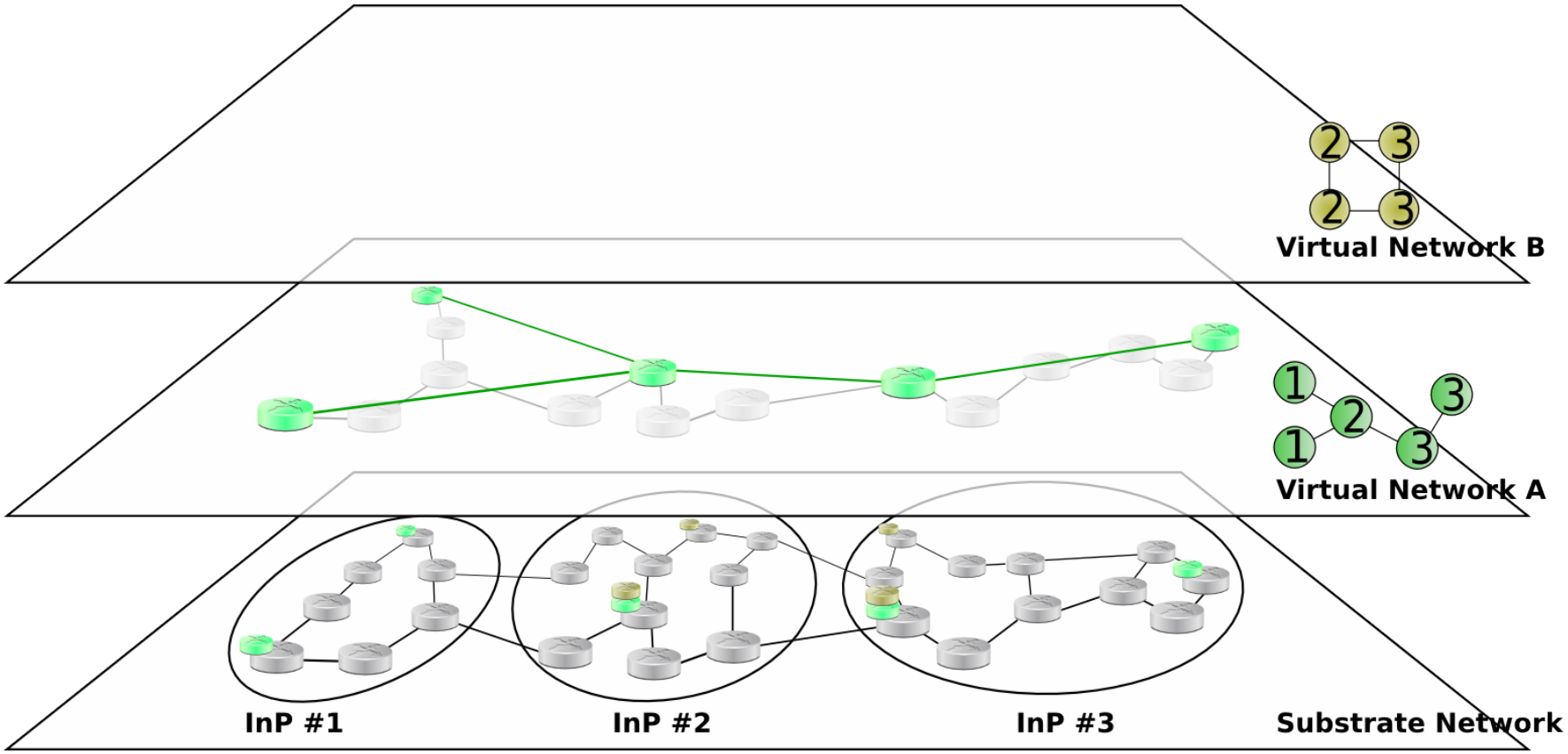
Definition: a simple scenario



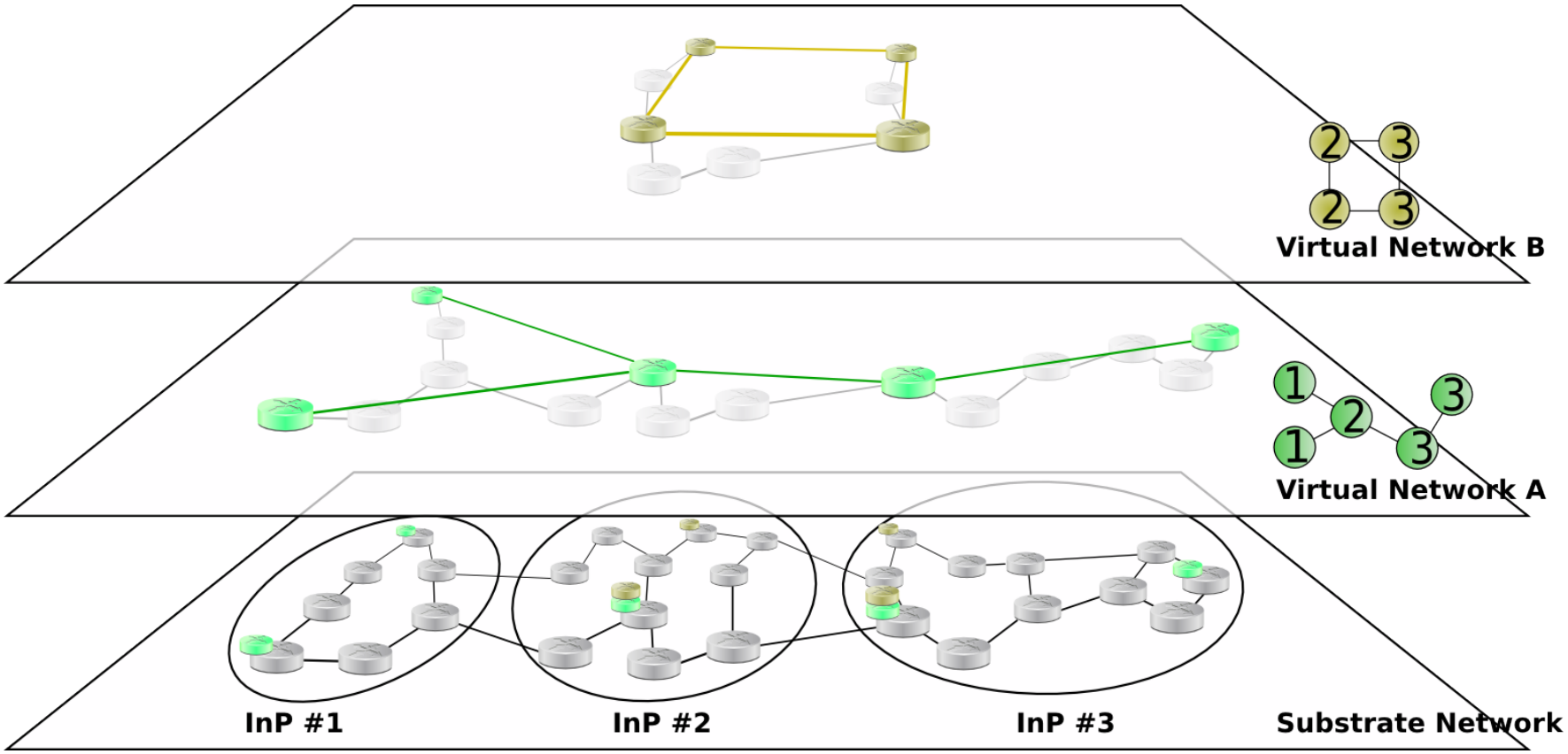
Definition: a simple scenario



Definition: a simple scenario



Definition: a simple scenario



Major (potential) benefits

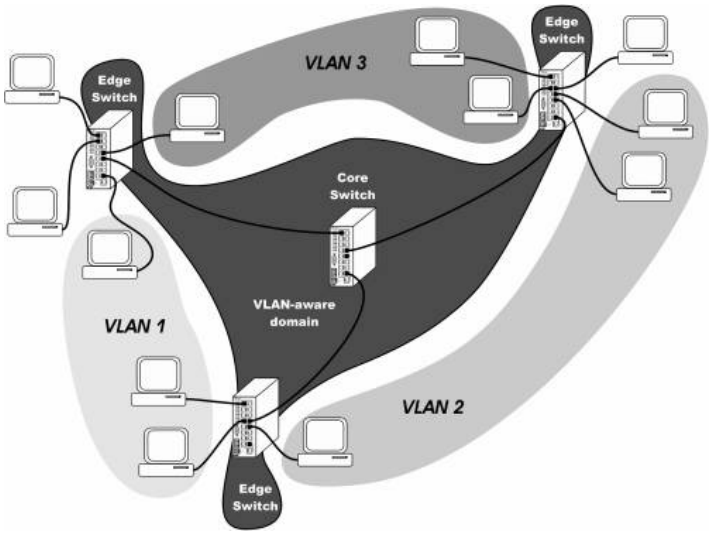
- **Protection**: provides isolation among different types of traffic (e.g., banking, VoIP, LiveTV, web)
- **Coexisting networks**: allows deployment of specially tailored (and possibly conflicting) architectures
- **Research**: allows development, debugging and experimentation of disruptive architectures over production networks
- **Better resource utilization**: through dynamic allocation of virtual nodes and links in respect to traffic variations
- **New business models**: allows separation of roles (e.g. InP/SPs), creating new opportunities for companies

Network virtualization is an abstract concept. Concrete examples follow.

[Belbekkouche et al., 2012; Khan et al., 2012]

VLANs – Virtual Local Area Networks

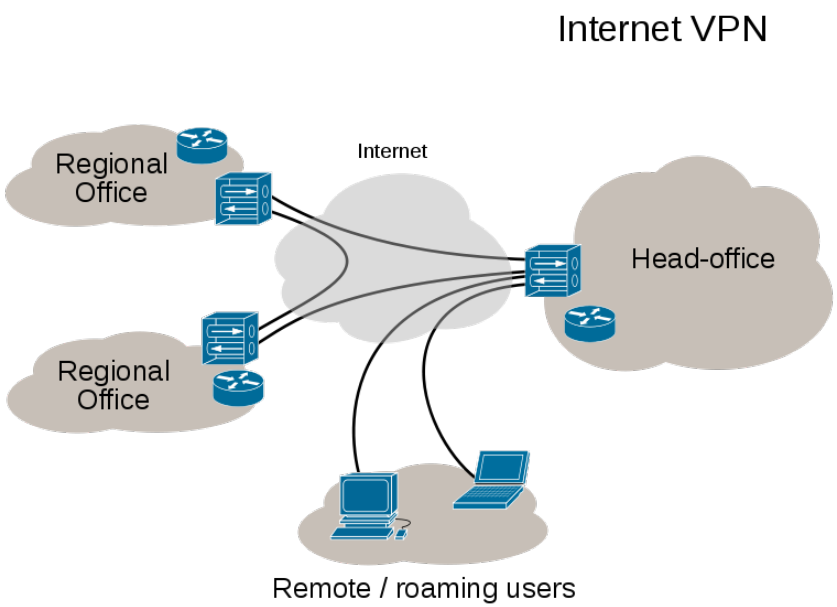
- Technology introduced in the 80s
- Became a standard – IEEE 802.1Q
- MAC headers in frames are labeled with VLAN tag
- Switches are configured to associate ports with VLAN tags
- VLAN bridges provide interconnection among different networks



[Chowdhury and Boutaba, 2010; LAN/MAN Standards Committee, 2011]

VPNs – Virtual Private Networks

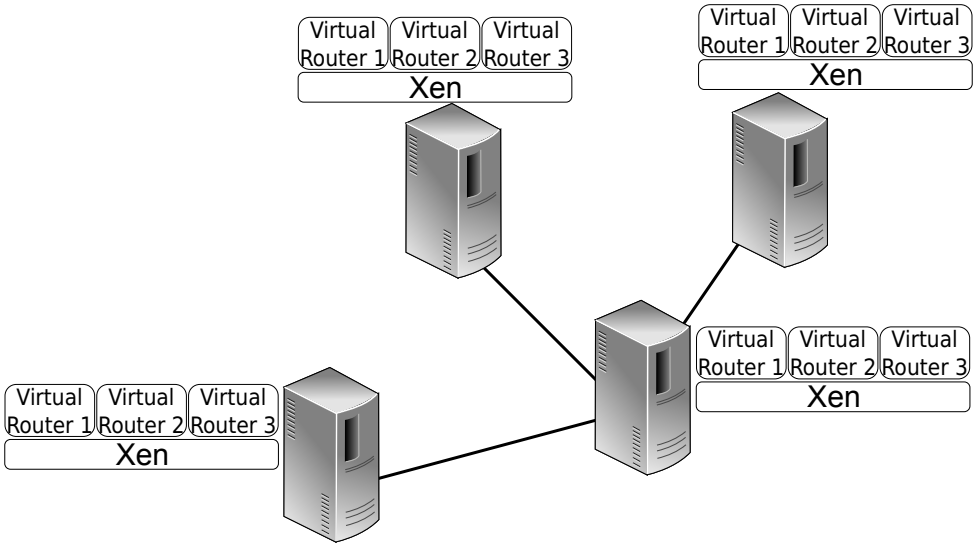
- Virtual Private Networks allow **remote** networks to be connected
- Standardized by RFC 4364
- Tunneling: edge routers do encapsulation using
  - wrapping: GRE, PPTP, L2TP
  - encrypted and authenticated channels: IPsec, TLS



[Chowdhury and Boutaba, 2010; Rosen et al., 2006]

Full Network Virtualization

- Based on off-the-shelf PCs (or special hardware)
- Allows **full** network stack virtualization: each VM runs entire protocol stack
- PCs slower than dedicated forwarding devices
- Requires tunnels if not all devices are compliant



[Egi et al., 2010; Fernandes and Duarte, 2011]

Application-layer Network Virtualization

- Virtual topologies built (typically) on top of IP networks
- Edges represent transport-level communication, e.g. TCP connections

- P2P multimedia communications (Skype)
- Privacy-preserving *overlays* (TOR network, Onion routing)
- Experimentation environments (PlanetLab)

- There is (almost) no control over underlaying network

[Chowdhury and Boutaba, 2010; Lua et al., 2005]

Trends and Research Challenges

Research challenges for network virtualization include

- Improving resource allocation and scheduling
- More suitable economic models
- Heterogeneity and diversity of the physical infrastructure
- Better understanding of security and privacy

[Belbekkouche et al., 2012; Chowdhury and Boutaba, 2010; Stezenbach et al., 2012]

Trends and Research Challenges

Research challenges for network virtualization include

- Improving resource allocation and scheduling
- More suitable economic models
- Heterogeneity and diversity of the physical infrastructure
- Better understanding of security and privacy

- Network virtualization seems to be moving to **SDN**
- SDN is being pushed by Industry and researchers (e.g., ONF, NDDI)

[Belbekkouche et al., 2012; Chowdhury and Boutaba, 2010; Stezenbach et al., 2012]

Questions

1. List at least three benefits of virtual networks.
2. Does network virtualization present any disadvantages? If so, which ones?
3. Name at least three NV broad categories.
4. Identify areas in which further research work is needed for network virtualization.

Outline

Key Points

- Motivation
- Definition
- Layered Architecture
- SDN Controller
- SDN Instantiation
- SDN application to datacenters
- Trends and research challenges

Motivation

- Lack of innovation
  - networks have become part of the critical infrastructure
  - new network requirements may take years to be addressed
- Complexity
  - networks built from a large number of devices
  - devices are typically "black-boxes"
  - heterogeneity of applications
  - operators must configure policies for a large number of events

Network management, including in datacenters, can be challenging and error-prone

[Koponen et al., 2010; Lara et al., 2013; McKeown et al., 2008; Mendonça et al., 2013]

Definition

- Software-Defined Networking (**SDN**) is a new paradigm based on the separation of data and control planes
- **Data plane** responsible for packet forwarding action:
  check packet header, lookup forwarding table, apply action
- **Control plane** populates forwarding tables based on an objective:
  routing, traffic engineering, access control...

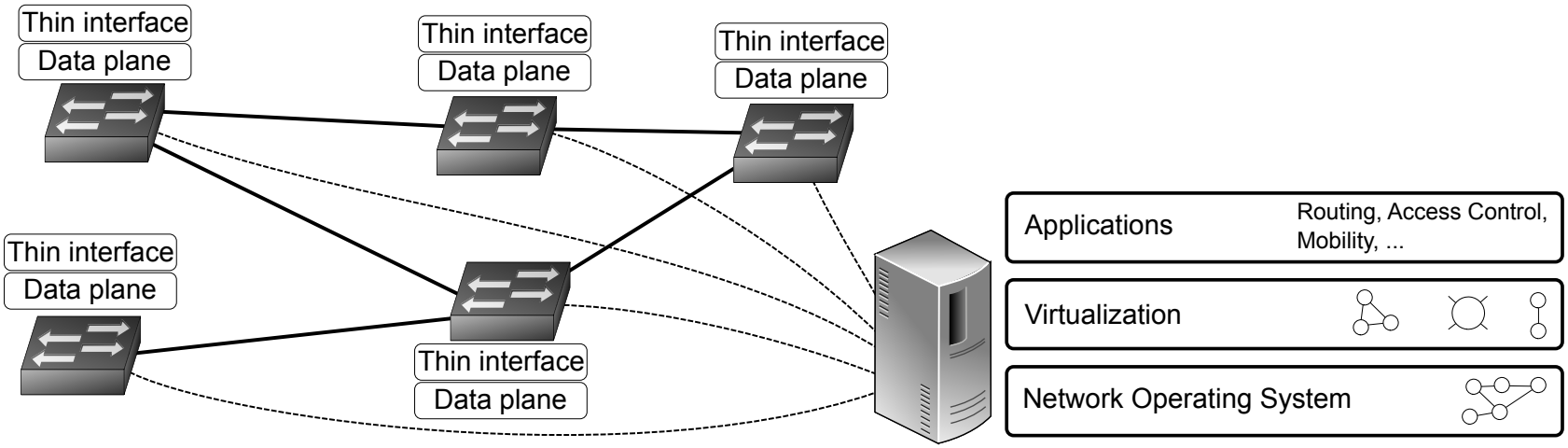
[McKeown et al., 2008; Sherwood et al., 2010]

Definition

- Software-Defined Networking (**SDN**) is a new paradigm based on the separation of data and control planes
- **Data plane** responsible for packet forwarding action:
  check packet header, lookup forwarding table, apply action
- **Control plane** populates forwarding tables based on an objective:
  routing, traffic engineering, access control...
- Control plane applications can be developed and deployed using high-level languages
- Network virtualization is implemented as a software running on a logically-centralized **controller**

[McKeown et al., 2008; Sherwood et al., 2010]

Layered Architecture



Applications, Virtualization, Network Operating System, Physical layer

[McKeown et al., 2008; Sherwood et al., 2010]

Layered Architecture

- **Applications**: several isolated instances of control software
- **Virtualization**: allows isolation among applications and abstractions of the network topology
- **Network Operating System**: interfaces with devices and provides full view and control of the network as abstractions for upper layers
- **Physical layer**: simple forwarding devices with support to remote data plane programmability

[McKeown et al., 2008; Sherwood et al., 2010]

Layered Architecture

- **Applications**: several isolated instances of control software
- **Virtualization**: allows isolation among applications and abstractions of the network topology
- **Network Operating System**: interfaces with devices and provides full view and control of the network as abstractions for upper layers
- **Physical layer**: simple forwarding devices with support to remote data plane programmability

But in practice things are far from this elegant set of layers.

[McKeown et al., 2008; Sherwood et al., 2010]

SDN Controller

- Centralized vs. distributed
- Control granularity
- Reactive vs. proactive

[Mendonça et al., 2013]

SDN Controller: centralized vs. distributed

- Logically and physically centralized
  - single controller instance for the entire network (e.g., NOX)
  - backup replicas required to allow the control plane to survive failures
- Logically centralized but physically distributed
  - network-wide controller running on top of several servers (e.g., Onix, HyperFlow)
  - increases scalability and reliability
- Hybrid (or logically distributed)
  - local controllers synchronize only for decisions that require global network state (e.g., Kandoo)
  - reduces the overhead in maintaining consistency when distributing state

[Koponen et al., 2010; McKeown et al., 2008; Mendonça et al., 2013; Tootoonchian and Ganjali, 2010]

SDN Controller: control granularity

- Traditional network elements process individual packets
- In SDN, Controller is remote to the forwarding element
- Controller would have to receive and make a decision for each arriving packet

[Lara et al., 2013; Mendonça et al., 2013]

SDN Controller: control granularity

- Traditional network elements process individual packets
- In SDN, Controller is remote to the forwarding element
- Controller would have to receive and make a decision for each arriving packet
- Instead, **flow** is the basic unit to be controlled
  - identified by matching one or more fields of packet headers
  - applications send data as a flow of many individual packets
  - decision made for the first packet of the flow can be applied to all subsequent packets of the same flow
- Flows can be aggregated, associating an operation that is common to multiple flows

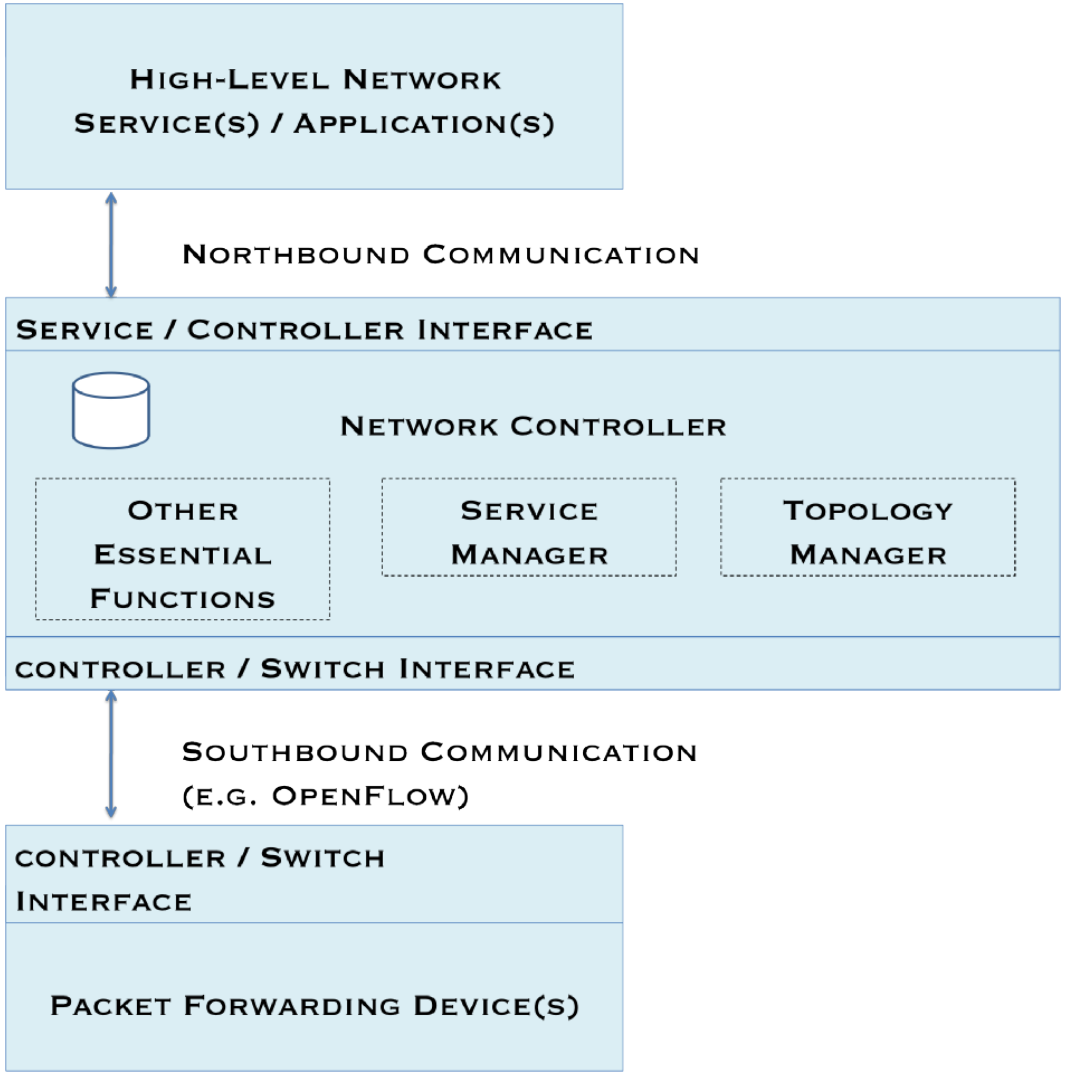
[Lara et al., 2013; Mendonça et al., 2013]

SDN Controller: reactive vs. proactive

- Reactive
  - forwarding elements must consult a controller each time a decision must be made
  - performance delay as the first packet of each new flow is forwarded to the controller (e.g., when a controller is geographically remote)
- Proactive
  - push policy rules out from the controller to switches (e.g., DIFANE)
  - controller needs to be consulted only about some new flows

[Mendonça et al., 2013; Yu et al., 2010]

SDN: Architecture, II



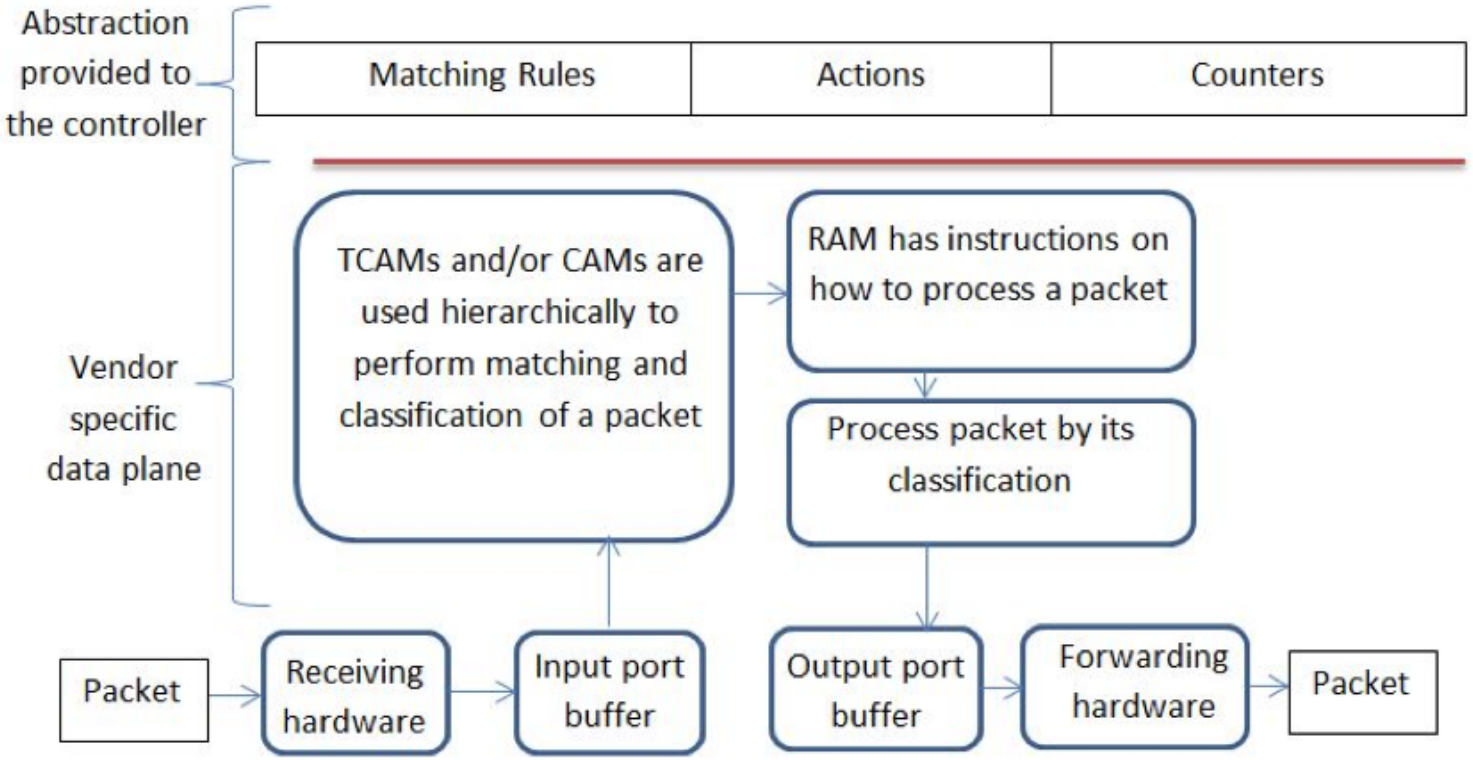
[Mendonça et al., 2013]

SDN Instantiation: OpenFlow

- Ethane, 4D
- Original motivation: network testbeds & experimentation
- OpenFlow protocol standardizes the communication between controller and forwarding devices
- Devices can be *OpenFlow-enabled* or *native*
  Examples: Cisco (cat6k, catalyst 3750, 6500), Juniper (MX-240, T-640), HP, Dell...
- Controller implementations: NOX, Beacon, Maestro, Floodlight...
- Specification (versions): 1.0.0, 1.1.0, 1.2, 1.3.0
- OpenFlow vs SDN

[Lara et al., 2013; McKeown et al., 2008]

SDN Instantiation: OpenFlow-compliant switch



[Lara et al., 2013]

SDN application to datacenters

- Datacenter networks (see next chapter) are subject to
  - dynamic nature of the network (tenant, app churn)
  - rapid change of traffic demands
- Traffic management and policy enforcement are critical

[Al-Fares et al., 2010; Benson et al., 2010; Curtis et al., 2011a; Heller et al., 2010]

[Lara et al., 2013; Mendonça et al., 2013; Popa et al., 2013; Rotsos et al., 2012]

SDN application to datacenters

- Datacenter networks (see next chapter) are subject to
  - dynamic nature of the network (tenant, app churn)
  - rapid change of traffic demands
- Traffic management and policy enforcement are critical
- SDN can provide a centralized way of control, implement network virtualization
  - provide bandwidth guarantees for tenants
  - meet efficiency, agility, scalability and simplicity

[Al-Fares et al., 2010; Benson et al., 2010; Curtis et al., 2011a; Heller et al., 2010]

[Lara et al., 2013; Mendonça et al., 2013; Popa et al., 2013; Rotsos et al., 2012]

SDN application to datacenters

- Datacenter networks (see next chapter) are subject to
  - dynamic nature of the network (tenant, app churn)
  - rapid change of traffic demands
- Traffic management and policy enforcement are critical
- SDN can provide a centralized way of control, implement network virtualization
  - provide bandwidth guarantees for tenants
  - meet efficiency, agility, scalability and simplicity
- Network accounts for 10-20% of the total energy cost

[Al-Fares et al., 2010; Benson et al., 2010; Curtis et al., 2011a; Heller et al., 2010]

[Lara et al., 2013; Mendonça et al., 2013; Popa et al., 2013; Rotsos et al., 2012]

SDN application to datacenters

- Datacenter networks (see next chapter) are subject to
  - dynamic nature of the network (tenant, app churn)
  - rapid change of traffic demands
- Traffic management and policy enforcement are critical
- SDN can provide a centralized way of control, implement network virtualization
  - provide bandwidth guarantees for tenants
  - meet efficiency, agility, scalability and simplicity
- Network accounts for 10-20% of the total energy cost
- Save energy: find the minimum-power network subset which satisfies current traffic conditions and migrate

[Al-Fares et al., 2010; Benson et al., 2010; Curtis et al., 2011a; Heller et al., 2010]

[Lara et al., 2013; Mendonça et al., 2013; Popa et al., 2013; Rotsos et al., 2012]

Trends and Research Challenges

- Availability, security & privacy
- Performance (delay needed to establish new rules for flows)
- Scalability for data center networks
  - over 16 million flows per second
  - adjust rate limits at a high frequency
  - VMs compete for bandwidth over an arbitrary set of congested links
- Global-scale SDN
- Ensuring network consistency

[Lara et al., 2013; Popa et al., 2013]

Questions

1. Why was SDN proposed?
2. What is the most commonly deployed SDN technology?
3. What are the advantages and disadvantages of centralized, distributed and hybrid controllers?
4. How can proactive controllers help improve scalability?

Outline

Key Points

- Clouds and data centers
- Data center topology
- Novel topologies
- Network expansion
- Types of traffic
- Routing
- Addressing

Clouds and Data Centers

Data centers are at the heart of cloud computing – clouds are powered by warehouse-scale data centers

Adapted from [Abts and Felderman, 2012]

Types of Data Centers

Not all data centers are clouds!

Production data center

- 1 tenant, or low rate of tenant arrivals/departures
- Runs data-analytics jobs from multiple groups/services
- Demands vary relatively less
- 100~10k servers

Cloud data center

- High rate of tenant arrival and departure ("churn")
- Runs both user-facing app and inward computation
- Elasticity required
- 50k~300k servers

[Ballani et al., 2011b]

Examples of Cloud Data Centers

- Guesses on the size of global infrastructures of companies
  - Google: ~900,000 servers
  - Amazon: ~450,000 servers
  - Microsoft: ~1,000,000 servers
  - Rackspace: 98,884 servers (*officially disclosed*)

- List of known data center locations (*officially disclosed*)
  - North-America: 6 Google, 13 Amazon, 8 Microsoft and 5 Rackspace
  - Europe: 3 Google, 3 Amazon, 7 Microsoft and 2 Rackspace
  - Asia: 3 Google, 7 Amazon, 5 Microsoft and 1 Rackspace
  - Latin America: 1 Google, 1 Amazon and 2 Microsoft
  - Australia: 1 Rackspace

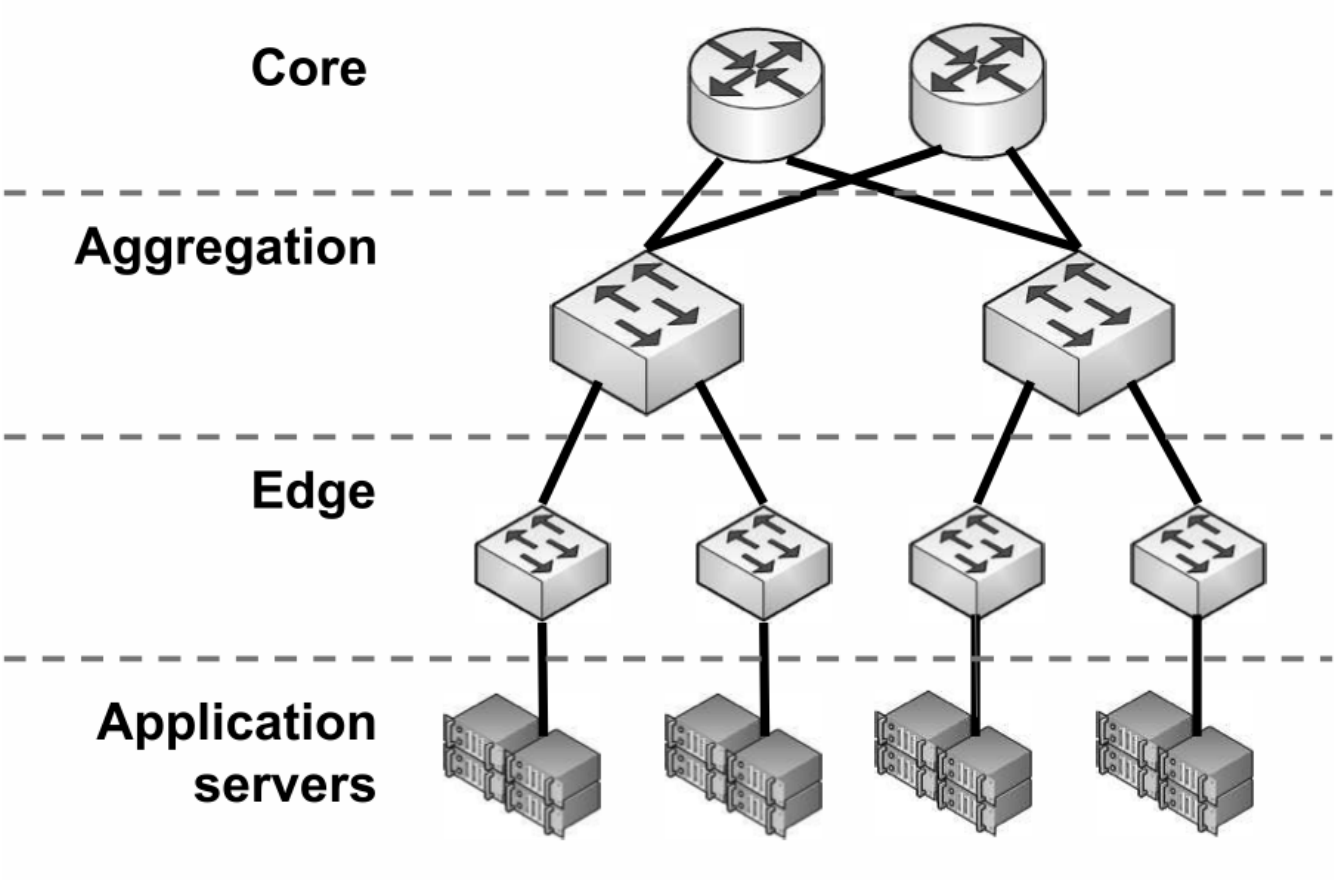
Examples of Cloud Data Centers

- Guesses on the size of global infrastructures of companies
  - Google: ~900,000 servers
  - Amazon: ~450,000 servers
  - Microsoft: ~1,000,000 servers
  - Rackspace: 98,884 servers (*officially disclosed*)

- List of known data center locations (*officially disclosed*)
  - North-America: 6 Google, 13 Amazon, 8 Microsoft and 5 Rackspace
  - Europe: 3 Google, 3 Amazon, 7 Microsoft and 2 Rackspace
  - Asia: 3 Google, 7 Amazon, 5 Microsoft and 1 Rackspace
  - Latin America: 1 Google, 1 Amazon and 2 Microsoft
  - Australia: 1 Rackspace

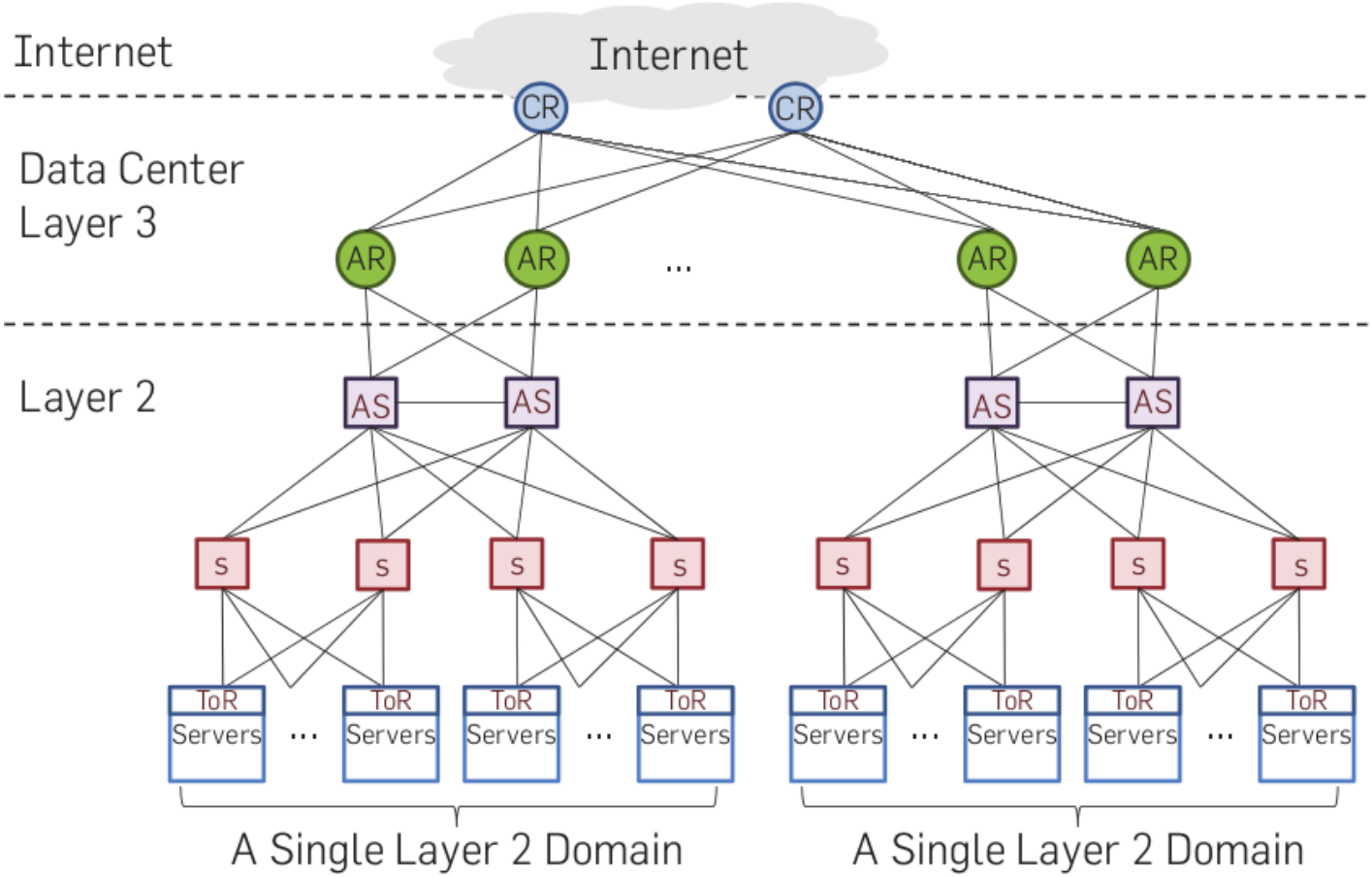
- What about the **network topology** of a single of these warehouse-scale data centers?

Typical Topology of a Data Center



Canonical 3-Tier data center topology, adapted from [Benson et al., 2010]

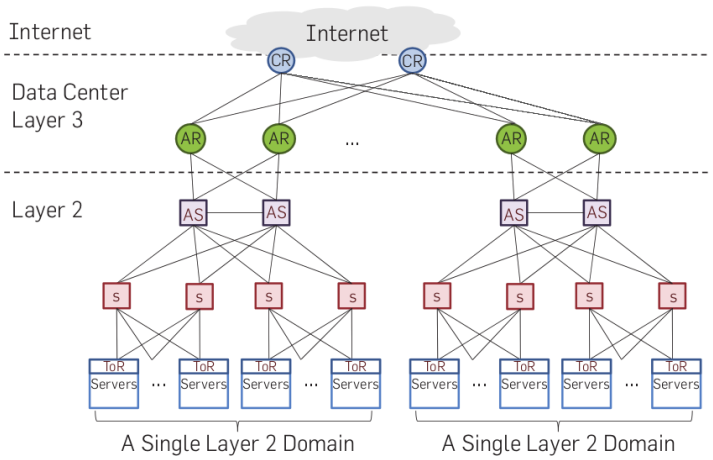
Typical Topology of a Data Center



Adapted from [Greenberg et al., 2011]

Typical Topology of a Data Center

- Data center switching infrastructure with two or three tiers
- Redundancy at different levels
- Multiple interconnected domains:
  - core routers
  - access routers
  - aggregation switches
  - intermediate switches
  - top-of-rack (**ToR**) switches
- Guaranteed bandwidth at ToR level
- The higher, the more **oversubscribed**
- Core oversubscription can be 1:240



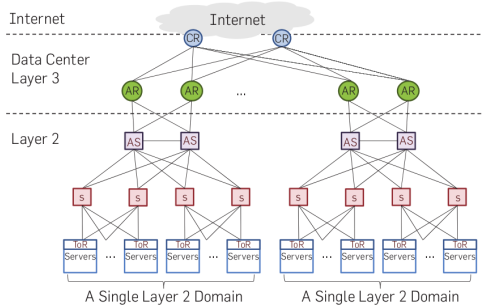
Conventional data center topology (adapted from [Greenberg et al., 2011])

[Abts and Felderman, 2012; Greenberg et al., 2011; Zhang and Ansari, 2013]

Limitations with Current Topologies

Major issues

- Poor server-to-server capacity, capped by oversubscription
- Limited bisection bandwidth (overloaded network core)
- Resources eventually fragmented
- Poor exploitation of multiple paths
- Incremental expansion hindered by rigid structure



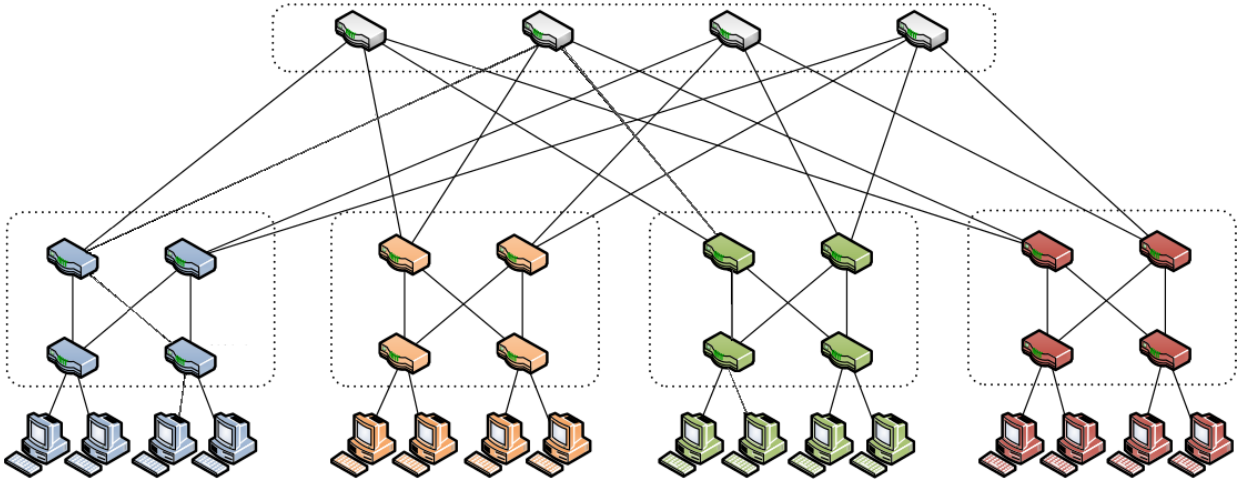
Conventional data center topology (adapted from [Greenberg et al., 2011])

Novel topologies organized in three classes:
Switch-oriented, Hybrid switch/server, and Server only

[Abu-Libdeh et al., 2010; Guo et al., 2008, 2009; Popa et al., 2010; Singla et al., 2012]
[Zhang and Ansari, 2013]

Switch-oriented: Clos-based trees (Fat-tree)

- Every level is fully connected to lower and upper levels
- Provides higher fault-tolerance and richer connectivity
- Theoretically achievable 1:1 oversubscription factor with multipath-aware routing
- Downside: increases the number of links



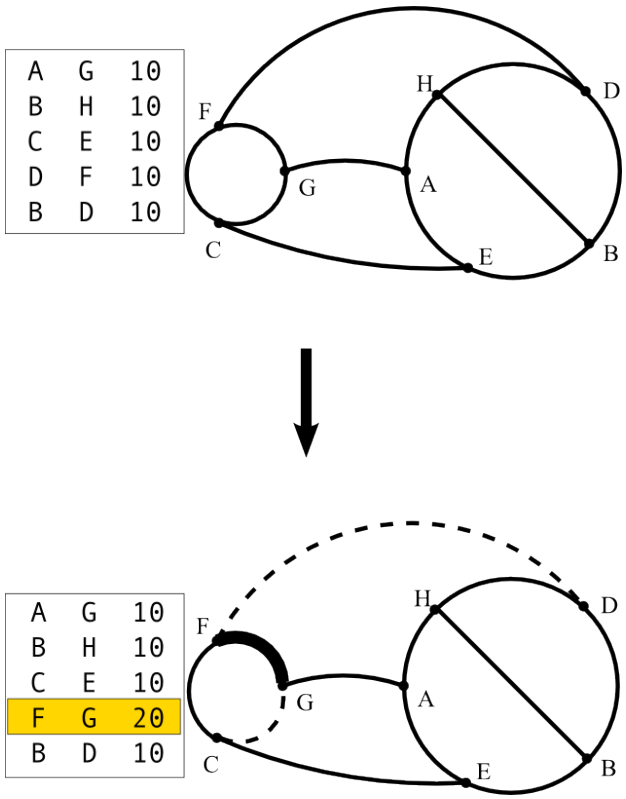
Fat-Tree topology (adapted from [Al-Fares et al., 2008])

[Al-Fares et al., 2008; Greenberg et al., 2009, 2011]

Switch-oriented: Optical Switching Architecture

- Leverages runtime reconfigurable **optical devices**
- Dynamically changes physical topology and link capacities (within 10 ms)
- Flexibility to adapt to dynamic traffic patterns
- Achieves high bisection bandwidth. But...

- Small, latency-sensitive flows may be affected by reconfiguration delays
- Scalability: OSA is designed to connect few thousands of servers in a container



OSA example (adapted from [Chen et al., 2012])

[Chen et al., 2012]

Hybrid switch/server topologies

- Principle: server performs routing, while low-end mini-switches interconnect a fixed number of hosts

- Can provide higher fault-tolerance and richer connectivity
- Complexity at servers, not switches (can use COTS switches)
- Different types of topologies proposed to increase scalability up to millions of servers
- Improve innovation since servers are easier to customize than COTS switches

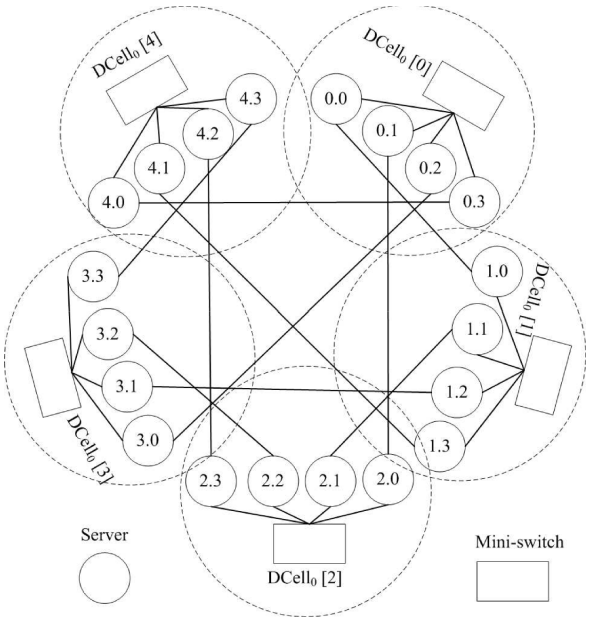
- For example, DCell and BCube

[Guo et al., 2008, 2009]

Hybrid: DCell

- Cell hierarchy:
  $DCells_0[1..n]$ composed of servers,
  $DCells_1$ composed of $DCells_0$
- Servers are interconnected by a COTS switch
- DCells are interconnected in a full mesh with servers-to-servers connections
- Scales up to millions (3.2M with 6 hosts per $DCell_0$ and 4 DCell levels). But...

- Increases overhead and number of NIC ports at end-hosts
- Unbalanced if the number of servers and connections is not close to full capacity



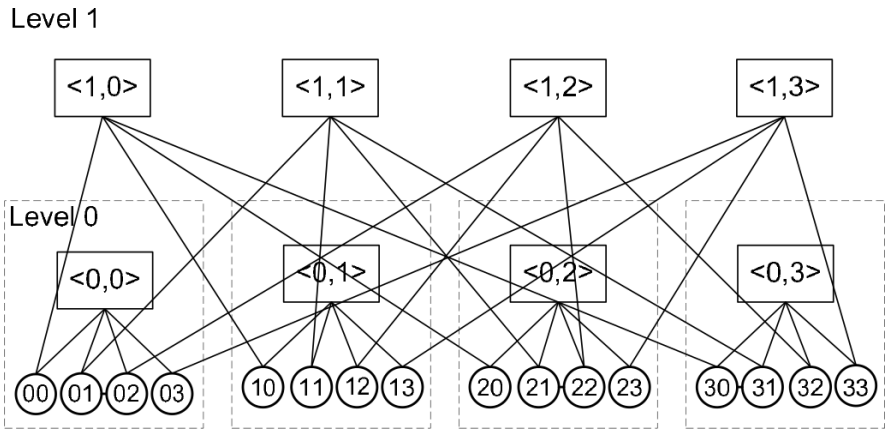
DCell topology

[Guo et al., 2008]

Hybrid: BCube

- Servers within a cluster interconnected by a COTS switch
- Clusters interconnected by COTS switch in hypercube-based topology (every edge connected to every other edge)
- Provides low latency and graceful degradation. But...

- Requires a higher number of links
- Increases overhead and number of NIC ports at end-hosts



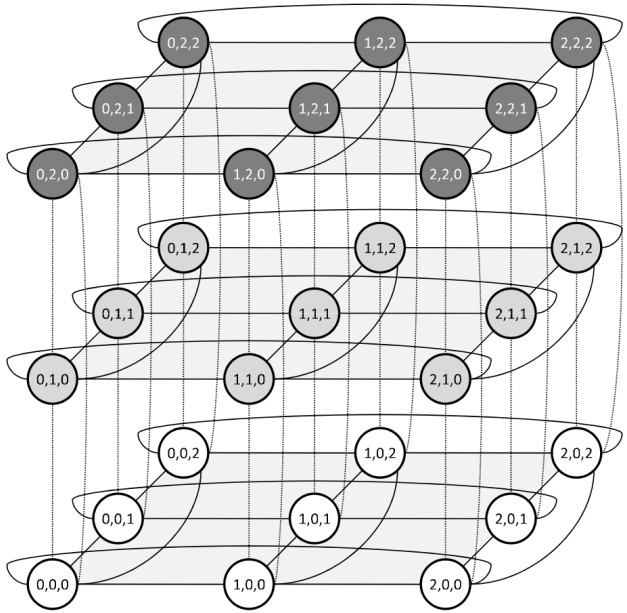
BCube topology

[Guo et al., 2009]

Server-oriented: CamCube

- Key-based routing (CAN)
- Unlikely to partition even with 50% server or link failures
- Does not hide topology from applications
- Improves innovation with server-to-server routing. But...

- Higher network diameter $O(\sqrt[3]{N})$, resulting increased latency/traffic
- Increased overhead and number of NIC ports at end-hosts
- Network expansion?



CamCube topology

[Abu-Libdeh et al., 2010; Costa et al., 2009]

Network Expansion

- Cloud data center networks can become under-provisioned
- There exist strict constraints on network topology
- Expensive to increase bisection bandwidth for applications
- Most physical data centers designs are unique:
  - expansions and upgrades must be custom designed
  - need to maximize network performance while
  - minimizing costs

[Curtis et al., 2010, 2012]

Network Expansion: recent proposals

Legup [Curtis et al., 2010]

- Attempts to keep legacy equipment, minimize rewiring
- Facilitates network upgrades and expansions
- Based on heterogeneous Clos networks

Network Expansion: recent proposals

Legup [Curtis et al., 2010]

- Attempts to keep legacy equipment, minimize rewiring
- Facilitates network upgrades and expansions
- Based on heterogeneous Clos networks

JellyFish [Singla et al., 2012]

- Degree-bounded random graph topology among ToR switches
- Random graph topology leads to flexibility in capacity and extensibility
- More cost-efficient than a fat-tree

Network Expansion: recent proposals

Rewire [Curtis et al., 2012]

- Design framework to create/upgrade/expand data center net
- Optimization algorithm
  - minimize end-to-end latency
  - user-defined constraints
  - maximize bisection bandwidth

Network Expansion: recent proposals

Rewire [Curtis et al., 2012]

- Design framework to create/upgrade/expand data center net
- Optimization algorithm
  - minimize end-to-end latency
  - user-defined constraints
  - maximize bisection bandwidth

These optimizations should consider the **nature of traffic**

Data Center Traffic: properties

Bimodal and Skewed

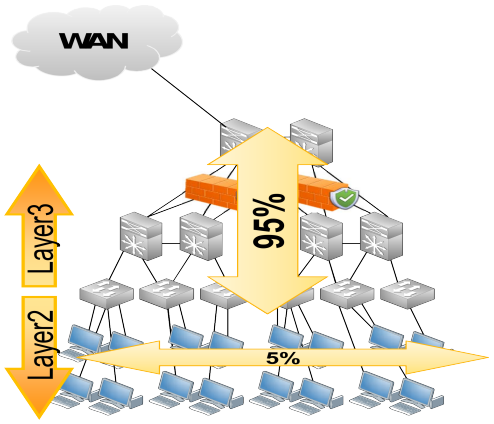
- Elephant flows: long-lived, bandwidth hungry, and scarce/bursty
  - less than 1% of all flows
  - generate more than half of the data volume
- Mice flows: very small and short-lived

- Mixing types of traffic may cause adverse effects
- Elephants create hot-spots, dropping several mice packets

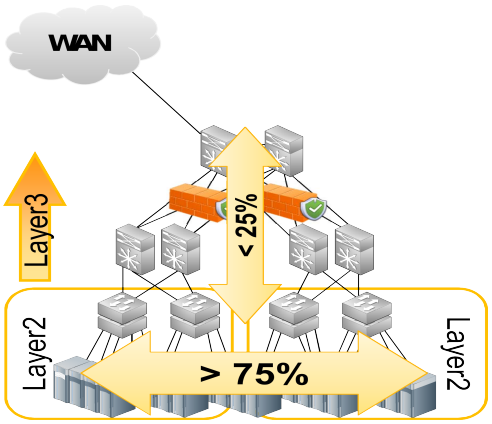
[Abts and Felderman, 2012; Kandula et al., 2009]

Data Center Traffic: properties

- **North-South**: extra-cloud communication (to/from the Internet)
- **East-West**: intra-cloud communication (inter-VMs)
- Depends on the kind of data center/mix of applications
- North-South traffic is increasing, but the East-West portion of overall traffic is getting much larger
- Inter-data center (D2D) traffic is a growing concern



Typical campus network



Cloud data center network

[Benson et al., 2010; Bodík et al., 2012; Chen et al., 2011; Recio, 2012]

Data Center Traffic: case studies

Public, Private, Academic data centers [Benson et al., 2010]

- Intra-cloud vs Extra-cloud
  - Public: 80% of traffic stays within the cloud
  - Private/Academic: 40-90% leaves the cloud
- Packet losses are due to momentary bursts
- Link utilization is on average low in all layers but the core
- Cloud networks may behave poorly with current models of statistical multiplexing

Data Center Traffic: case studies

Bing data centers [Bodík et al., 2012]

- **Intra**-service communication (represents 45% of all traffic)
  - 1% of services generate 64% of the traffic
  - 18% of services generate 99% of the traffic
  - services that do not require a lot of bandwidth can be spread out
- **Inter**-service communication (represents the remaining 55%)
  - only 2% of service pairs exchange data
  - of which 5% accounts for 99% of inter-service traffic
  - several highly-connected components, with a few services connected to hundreds of others in star-like topologies
- Applications with high bandwidth demands may overload core if placement is location-unaware

Routing in data center Networks

Load can be distributed if routing explores path diversity

Layer 3

- Equal-Cost Multi-Path (ECMP): spreads traffic among multiple paths that have the same cost calculated by the routing protocol
- Valiant Load Balancing (VLB): selects a random intermediate switch that will be responsible for forwarding an incoming flow to its corresponding destination

[Abts and Felderman, 2012; Wu et al., 2012]

Routing in data center Networks

Load can be distributed if routing explores path diversity

Layer 3

- Equal-Cost Multi-Path (ECMP): spreads traffic among multiple paths that have the same cost calculated by the routing protocol
- Valiant Load Balancing (VLB): selects a random intermediate switch that will be responsible for forwarding an incoming flow to its corresponding destination

Layer 2

- Multiple spanning trees
- TRansparent Interconnect of Lots of Links (TRILL)
  RFC 5556, RFC 6325
- Link Aggregation Control Protocol (LACP)

[Abts and Felderman, 2012; Wu et al., 2012]

Addressing in Data Center Networks

Using LAN technology in data centers

- Each VM instance requires a unique address
- Number of entries in switches is limited (~64K)
- Single address space to be sliced among tenants
- VLANs can isolate address spaces, but restricted to 4096 VLANs

[Abts and Felderman, 2012; Wu et al., 2012]

Addressing in Data Center Networks

Using LAN technology in data centers

- Each VM instance requires a unique address
- Number of entries in switches is limited (~64K)
- Single address space to be sliced among tenants
- VLANs can isolate address spaces, but restricted to 4096 VLANs

New technologies to achieve isolation

- Amazon's Virtual Private Cloud (VPC)
- Microsoft's Hyper-V
- Virtual Extensible LAN (VXLAN) framework (RFC draft)

[Abts and Felderman, 2012; Wu et al., 2012]

Addressing: separation of name and locator

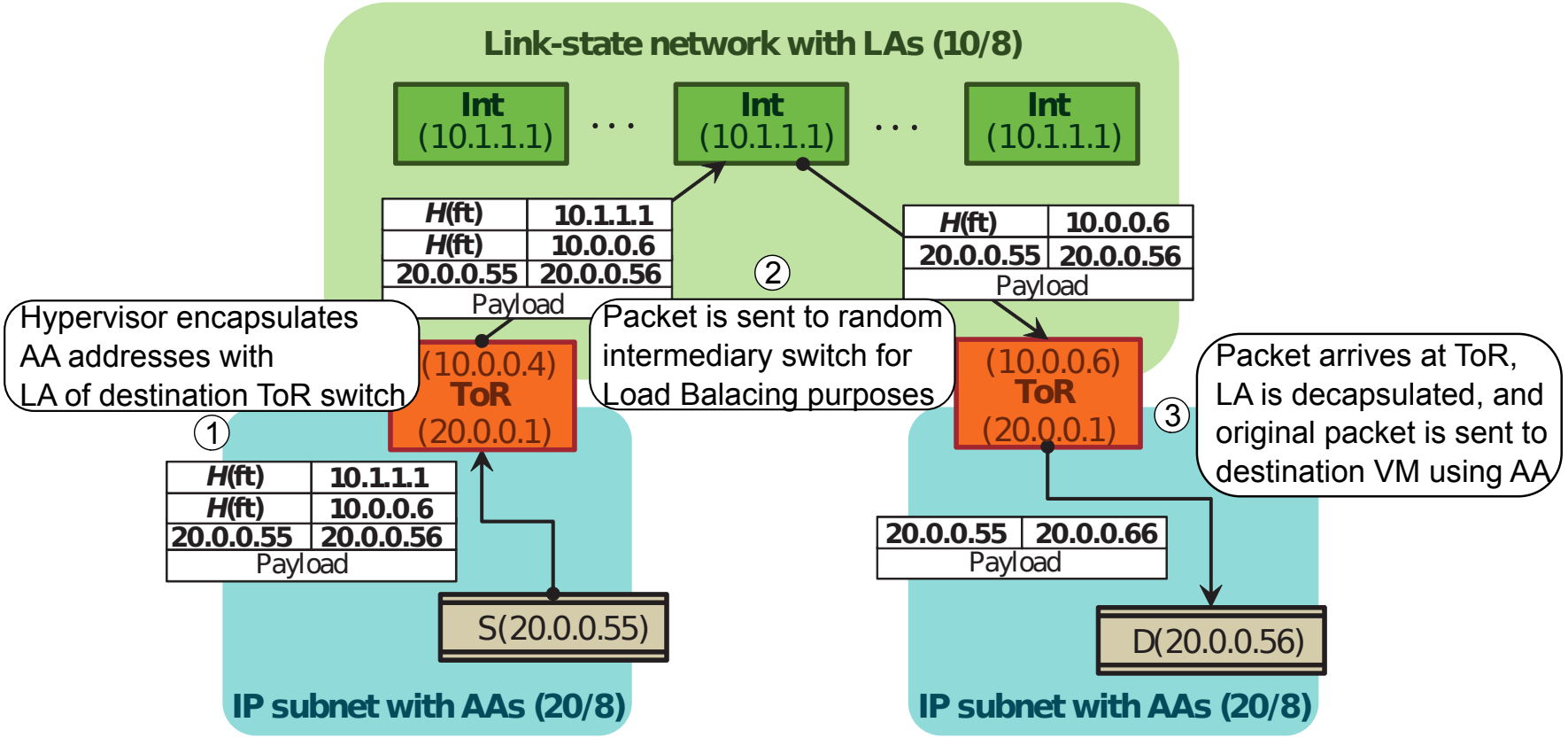
VL2 [Greenberg et al., 2009, 2011]

- Two types of addresses
  - **location-specific** address (LA): actual address, used for routing
  - **application-specific** address (AA): permanent address assigned to VM, remain unaltered even after migrations
- Directory system performs the mapping between AAs and LAs
- VL2 directory system enforces isolation (access control)

- Requirements:
  - switches have to support IP-over-IP
  - hypervisors need VL2 directory system and IP-over-IP

Addressing: separation of name and locator

VL2 [Greenberg et al., 2009, 2011]



Architecture for address translation in VL2, adapted from [Greenberg et al., 2011]

Addressing: separation of name and locator

CrossRoads [Mann et al., 2012]

- Extends the idea of location independence based on pseudo addresses
- CrossRoads is a network fabric to provide live/offline VM mobility across multiple data centers
- Solution is based on Software Define Networking
- Control plane overlay of OpenFlow network controllers in various data centers

Questions

1. Compare "production" and "cloud" data centers.
2. Provide a generic example of a production and a cloud data center.
3. What is "oversubscription" in DCNs? What are its implications?
4. Identify two limitations with current data center topologies.
5. Why do we say the nature of traffic in data center networks is skewed and bimodal?
6. To which types of topologies ECMP is applicable?

Outline

Key Points

- Major Improvements
- Adapting Conventional Protocols
- Full Address Space Virtualization
- Hypervisor Rate-limiting
- Software-Defined Cloud Network

Major Improvements

- **Performance isolation**: helps providing strict QoS requirements in terms of latency and throughput of tenants in data centers
- **Security**: improves protection against insider attacks performed by malicious tenants, e.g. performance interference
- **Deployability**: application-specific protocols/address spaces are hard to migrate to the cloud without isolation
- **Management flexibility**: application owners allowed to control/manage network fabric
- **Inherits advantages** of network virtualization, e.g. support for innovation, new business models, finer-grained resource utilization
- **Cost** can be reduced substantially by better network sharing

[Ballani et al., 2011b; Bari et al., 2012; Popa et al., 2012; Shieh et al., 2011]

Adapting Conventional Protocols

- VLAN and VPN technologies can be used to incorporate virtual networking into data centers
- Two major drawbacks:
  - lack of scalability of VLANs (up to 4k VLANs)
  - lack of interoperability between VLANs and VPNs

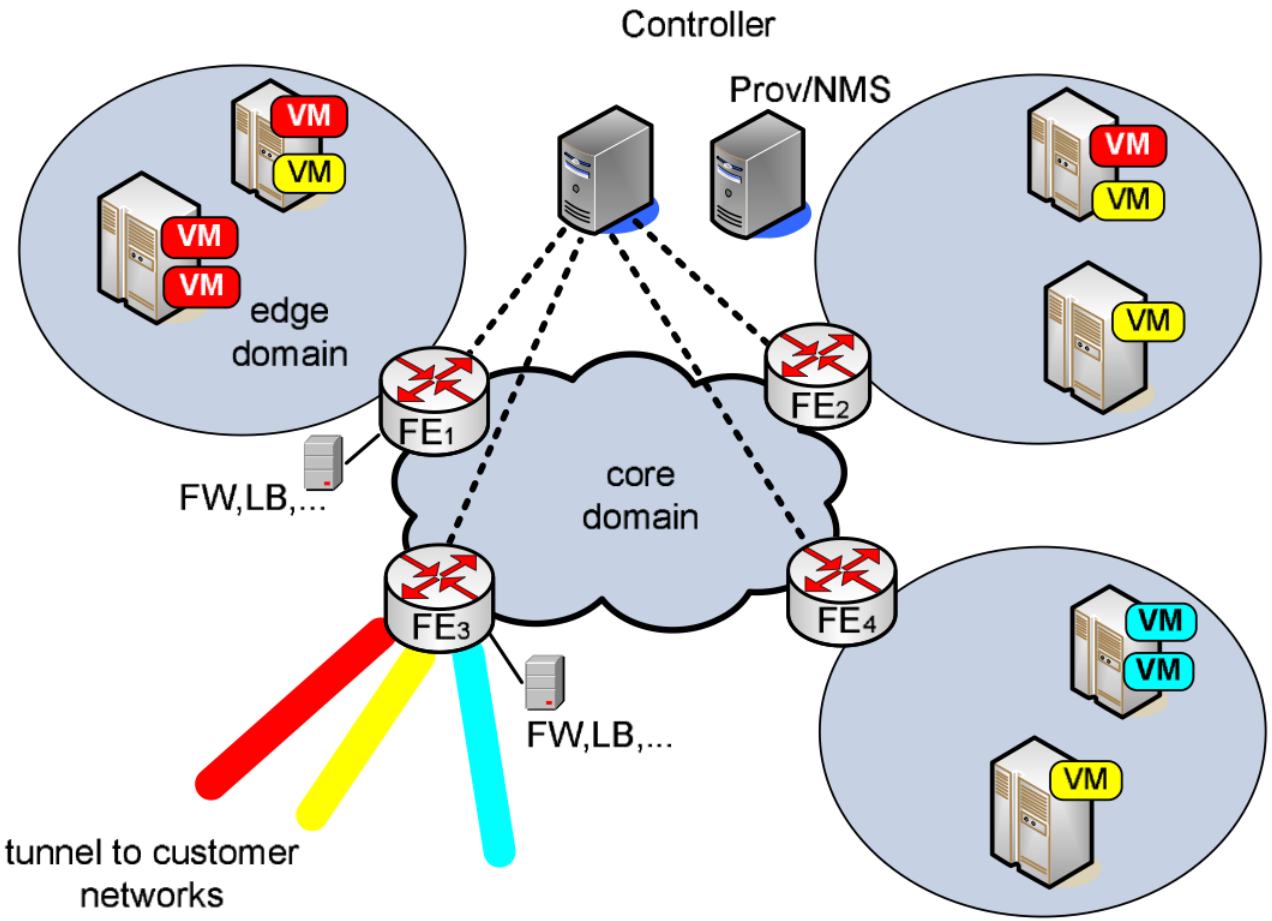
Approach

- Scale up VLANs by creating multiple **independent domains**
- Domains are implemented through SDNs
- Controller is used to coordinate inter-domain and inter-data center communication

Scaling up VLANs

VICTOR/SEC2 [Hao et al., 2009, 2010]

A core domain and multiple edge ones, with edge forwarding elements

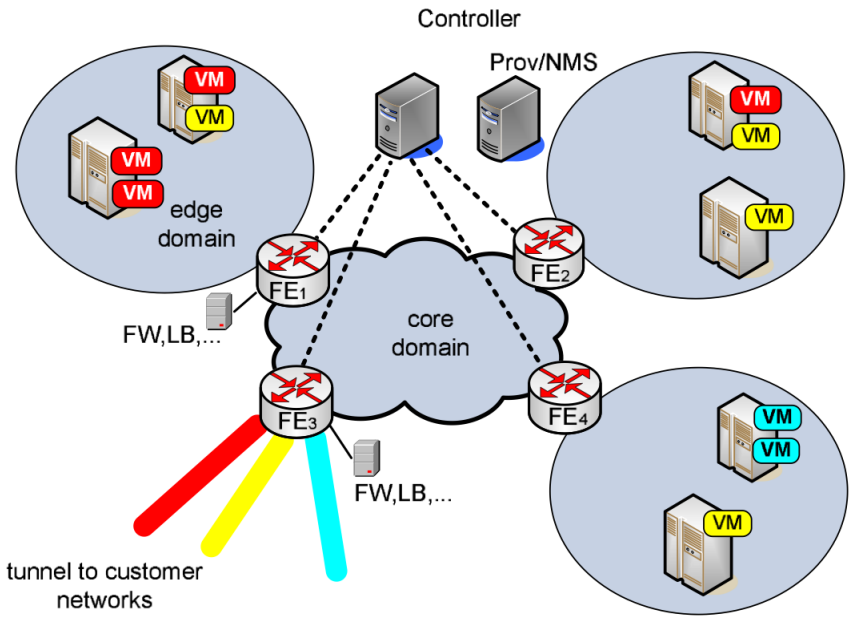


SEC2 architecture [Hao et al., 2010]

Scaling up VLANs

VICTOR/SEC2 [Hao et al., 2009, 2010]

- Architecture:
  - Forwarding Element (**FE**): OpenFlow edge switches
  - Central Controller (**CC**): defines access control policies
  - Core and Edge domains: COTS switches



- Communication:
  - intra-domain: isolated by (up to 4096) VLANs
  - inter-domain: mapping between VLANs of distinct domains via FEs
  - extra-cloud: obtained by mapping tenants' VLANs to VPNs

Novel Addressing Approach

- As already mentioned, current schemes are too constrained for data centers
  - limited support for multi-tenancy
  - may require complex configurations (e.g., manually setting up IP subnets)
  - tenants cannot design their own L2 and L3 addresses
  - hampers VM migration as IP offers poor support for mobility

Approach

- Full address space virtualization

[Mudigonda et al., 2011]

Full Address Space Virtualization

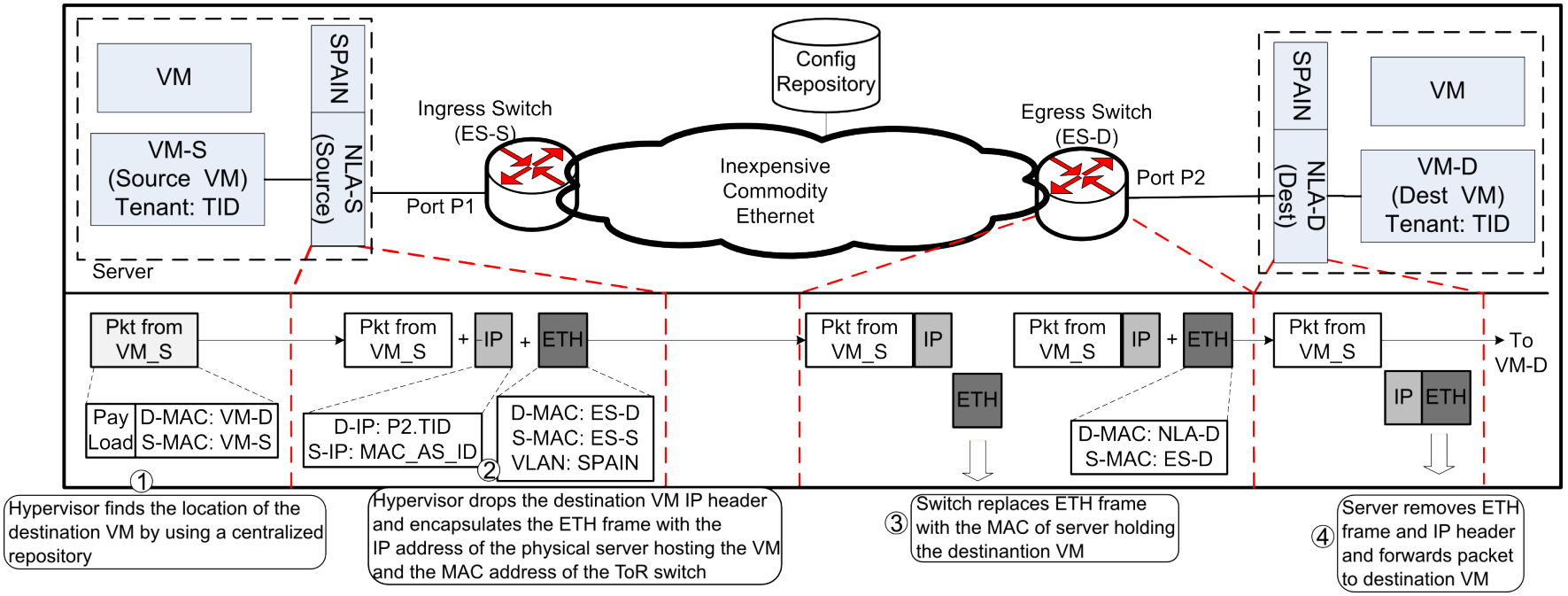
NetLord [Mudigonda et al., 2011]

- Allows tenants to design their own address spaces
- Employs a central configuration repository
- Mixes L2/L3 encapsulation and transfers use L2 fabric
- Multipath based on SPAIN

- Requirements:
  - hypervisors need Mac-in-IP encapsulation and SPAIN
  - aggregation switches require support for IP forwarding
  - packet fragmentation not allowed
  - use of a central configuration repository

Full Address Space Virtualization

NetLord [Mudigonda et al., 2011]



Architecture for address translation in NetLord, adapted from [Mudigonda et al., 2011]

Hypervisor Rate-limiting

- Typically high number of tenants in cloud data centers
- Explicit bandwidth reservations at switches does not scale
- Limited number of reservation classes on commodity switches

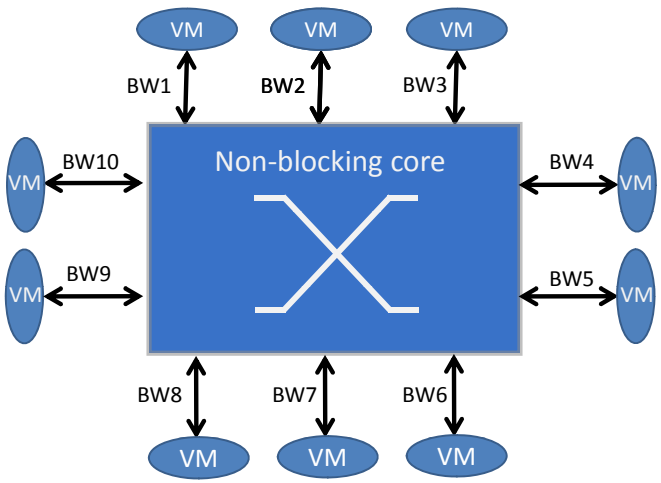
Approaches

- Bandwidth capping: limitation applied to each VM/cluster
- Bandwidth profiling: used to understand/predict specific application demands

Hypervisor Rate-limiting: bandwidth capping

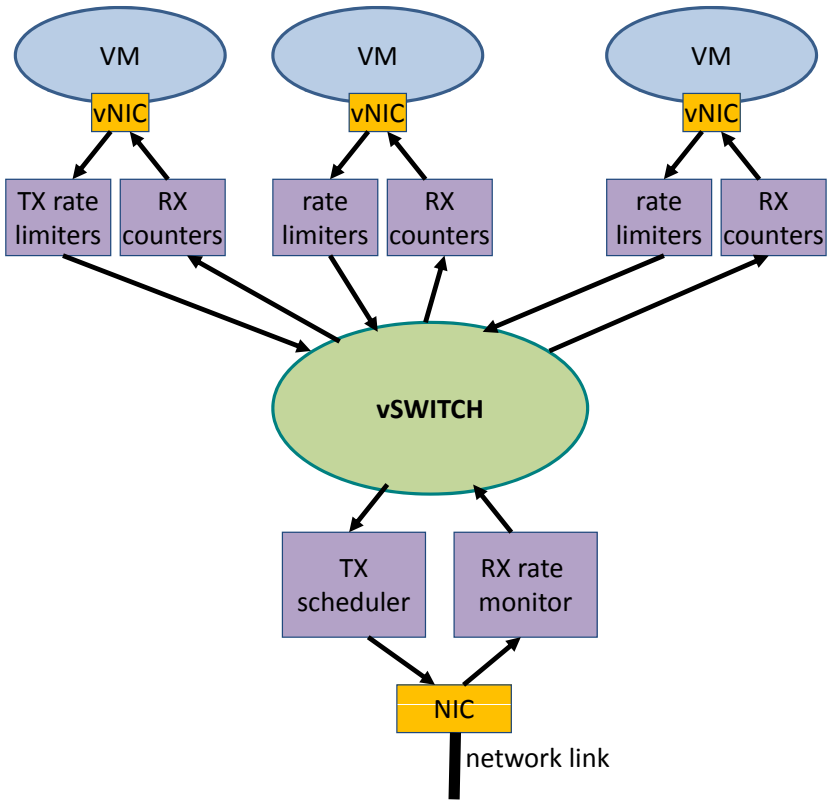
Gatekeeper [Rodrigues et al., 2011]

- Tenants are given an abstract "virtual switch" topology
- Bandwidth specified in min,max interval
- Rate-limiting at hypervisors
- Rate limiter monitors VM traffic



Hypervisor Rate-limiting: bandwidth capping

Gatekeeper Architecture



[Rodrigues et al., 2011]

Hypervisor Rate-limiting: bandwidth capping

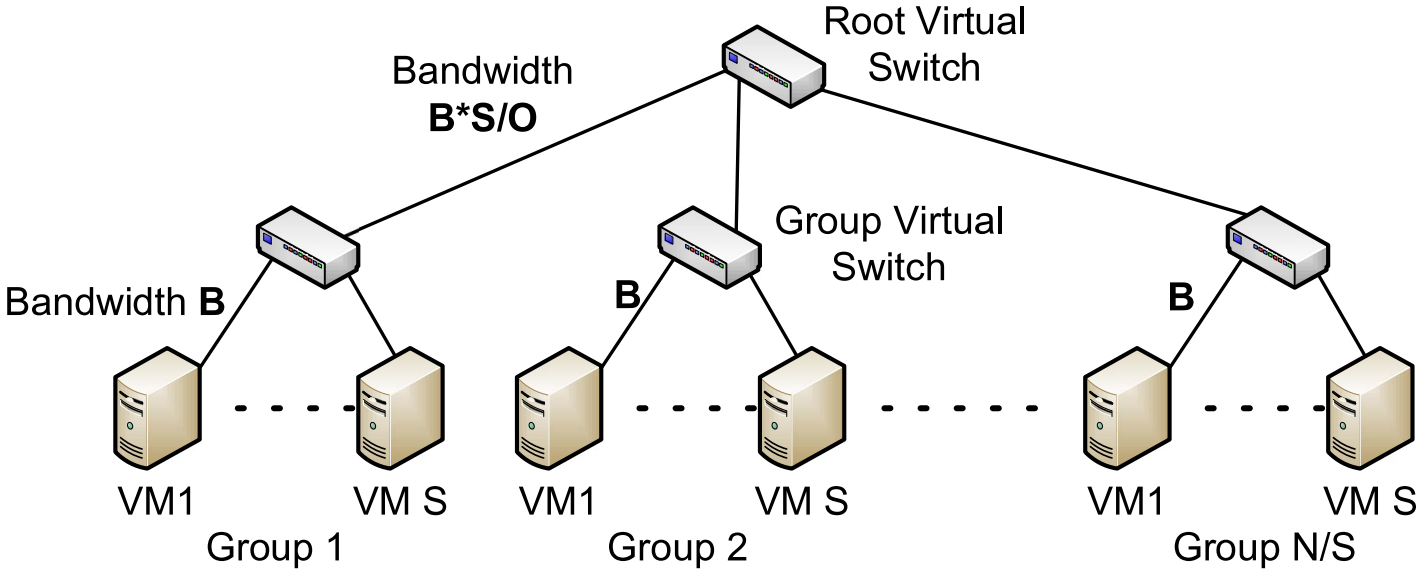
Oktopus [Ballani et al., 2011b]

- Abstract topology for tenants
  - "virtual cluster"
  - "virtual **oversubscribed** cluster"
- Hypervisor implements rate limiters
  - traffic at sources capped according to receiver capacity
  - warn senders that exceed capped capacity

- Requirements:
  - modified hypervisor
  - one rate-limiter per tenant with global knowledge of current traffic

Hypervisor Rate-limiting: bandwidth capping

Oktopus [Ballani et al., 2011b]



Request <N, S, B, O>

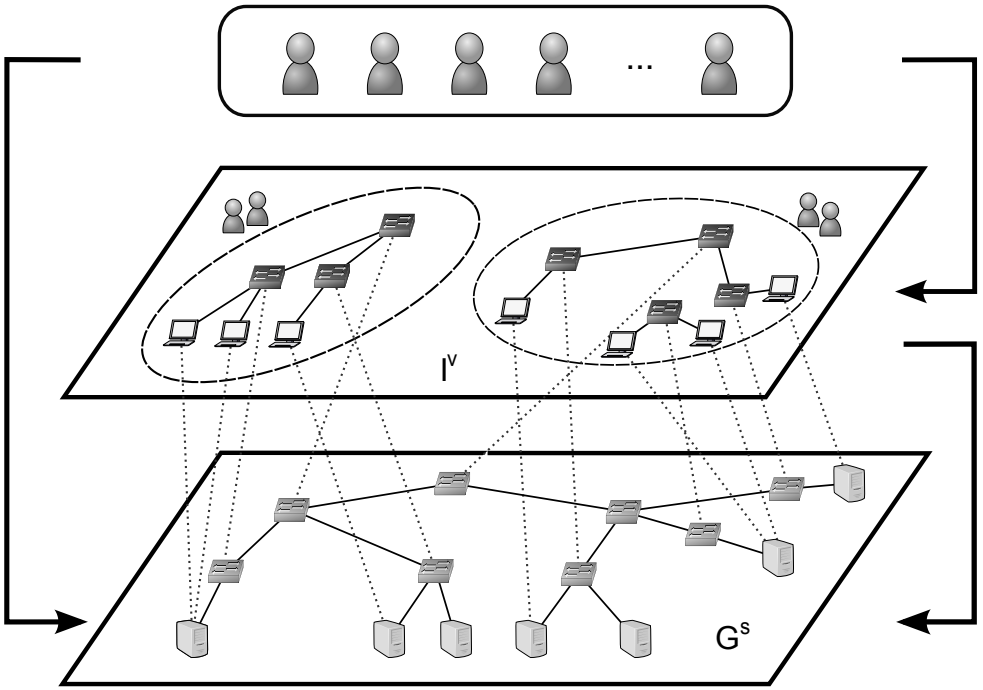
N VMs in groups of size S, Oversubscription factor O

Group switch bandwidth = S*B, Root switch bandwidth = N*B/O

Hypervisor Rate-limiting: bandwidth capping

Application grouping in virtual infrastructures [Marcon et al., 2013]

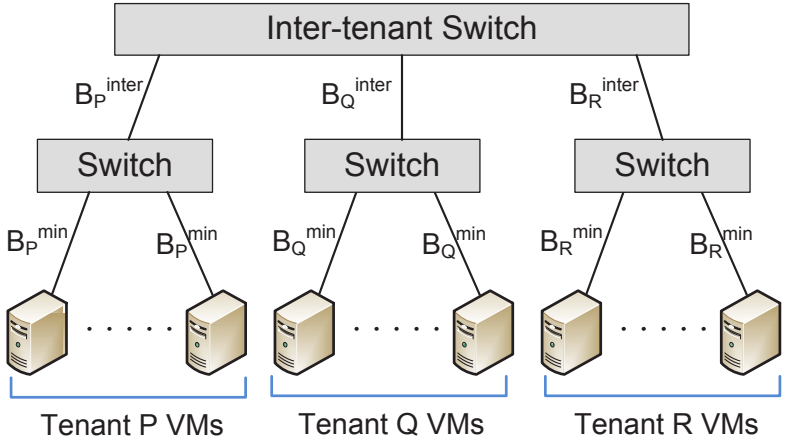
- Objectives:
  - increase security for tenants
  - improve network performance for applications
- Allocation process in two steps:
  - virtual infrastructure embedding on the cloud substrate
  - application mapping in virtual infrastructures



Hypervisor Rate-limiting: bandwidth capping

Hadrian [Ballani et al., 2013]

- Focuses on **inter-tenant** communication
- Addresses the "chatty" tenants problem
- Provides
  - minimum bandwidth and hierarchical guarantees
  - upper bound proportionality and high utilization

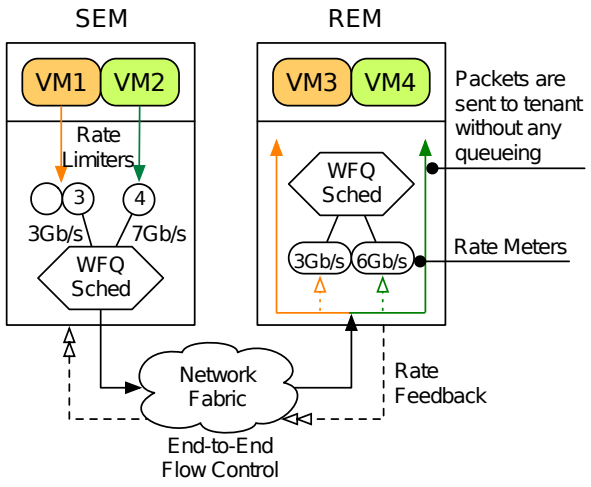


Hierarchical hose model ([Ballani et al., 2013])

Hypervisor Rate-limiting: bandwidth capping

EyeQ [Jeyakumar et al., 2013]

- Min, max bandwidth for each VM
- Enforces admission control on traffic
- Partitions bandwidth in a distributed manner at the edge
  - uses server-to-server congestion control
  - ensures bandwidth guarantees even in the presence of highly volatile traffic patterns
- Fully implemented

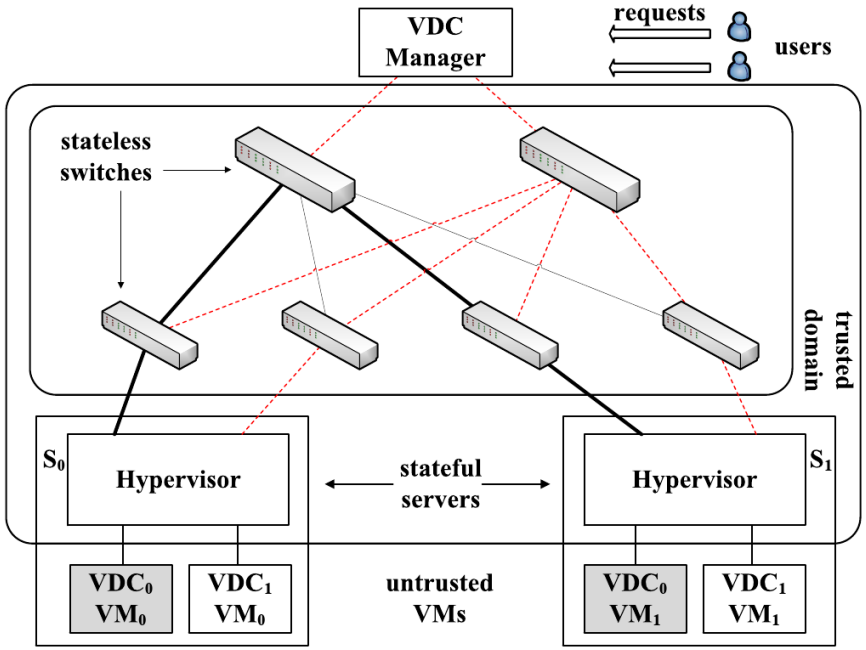


EyeQ sender and receiver design ([Jeyakumar et al., 2013])

Hypervisor Rate-limiting: bandwidth capping

SecondNet [Guo et al., 2010]

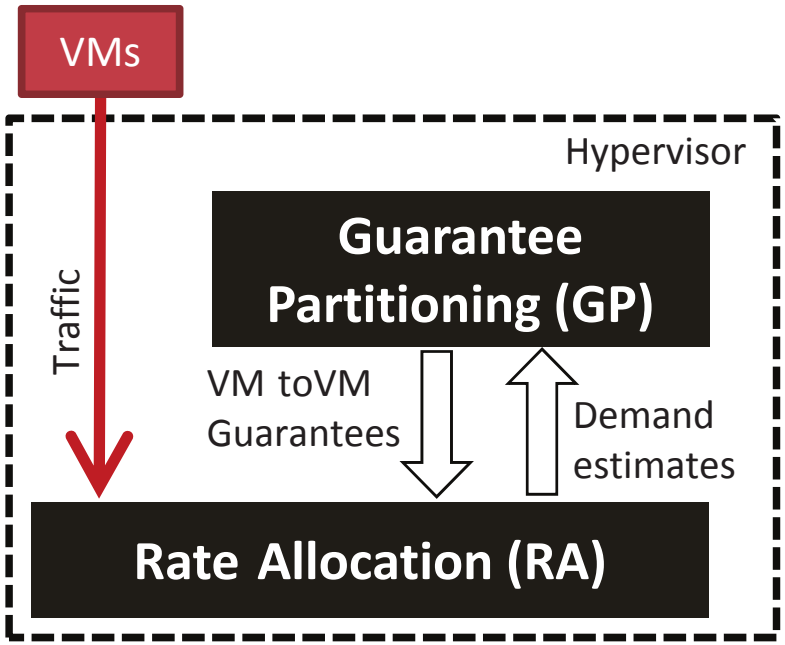
- Allocation of Virtual Data Centers (VDCs)
- Distributes all the virtual-to-physical mapping, routing, and bandwidth reservation state in hypervisors
  - Stateful hypervisors
  - Stateless switches
  - Port-switching based source routing (PSSR)
- Three basic service types
  - High priority end-to-end guarantee
  - Better than best-effort
  - Best-effort



Hypervisor Rate-limiting: bandwidth capping

ElasticSwitch [Popa et al., 2013]

- Does not require a specific topology or support from switches
- Provides minimum bandwidth guarantees
- **Work-conserving**: uses spare bandwidth
- Two layers in hypervisors:
  - guarantee partitioning
  - rate allocation



Hypervisor Rate-limiting: bandwidth capping

Seawall [Shieh et al., 2010, 2011]

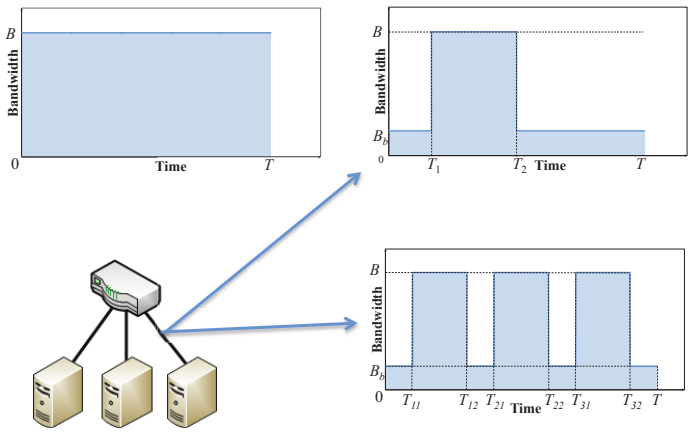
- Assign weights to network entities generating traffic (VMs, processes. . .)
- Uses congestion-controlled, hypervisor-to-hypervisor, **tunnels** between pairs of network entities
- "Shim layer" at the sender (hypervisor) runs in the virtualization or platform network stack

- Monitors and enforce bandwidth sharing policies
- Provides network proportionality
- Addresses DoS and performance interference attacks

Hypervisor Rate-limiting: bandwidth profiling

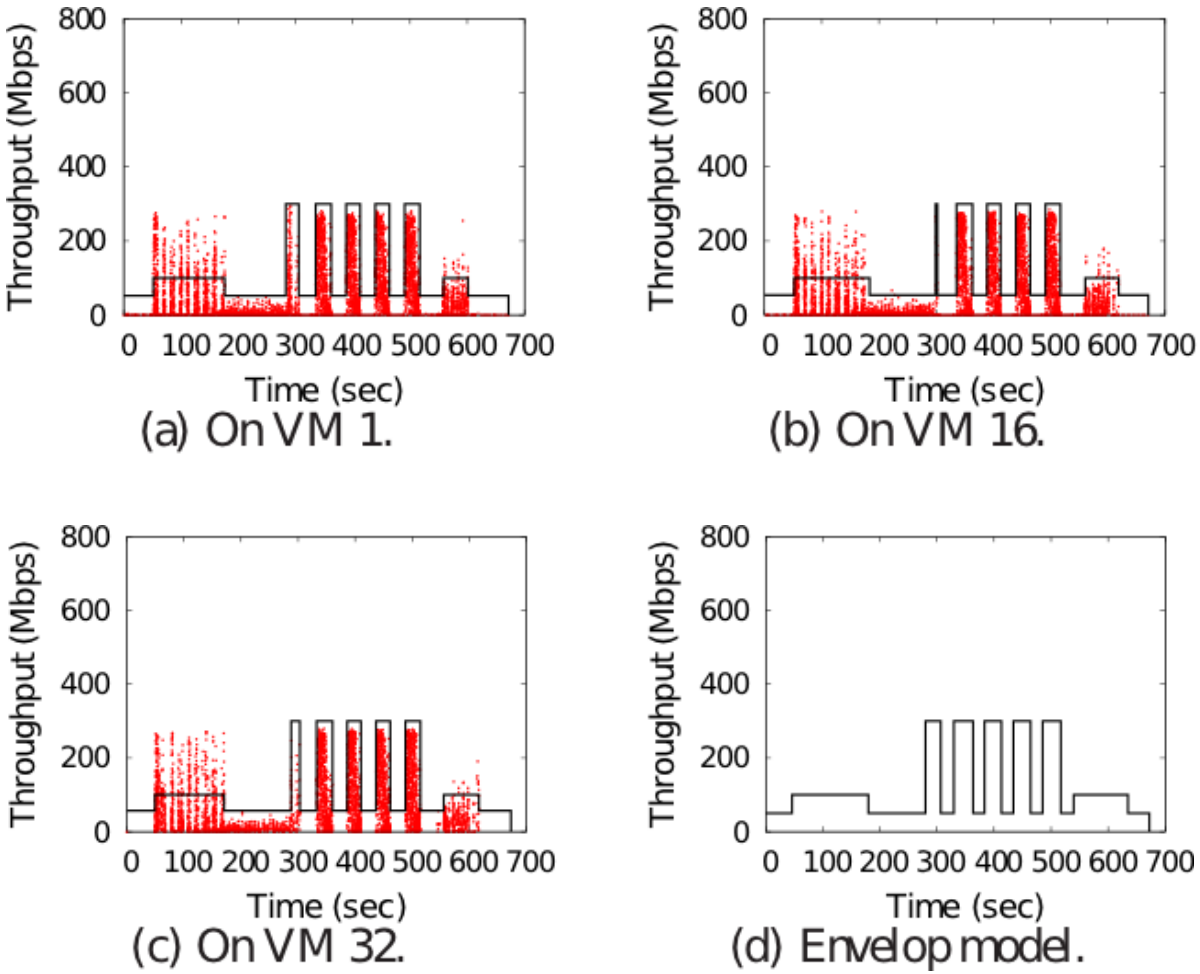
Proteus [Xie et al., 2012]

- Time-interleaved virtual clusters (TIVC)
- Measurement study of application traffic patterns
  1. single peak
  2. repeated fixed-width peaks
  3. varying-width peaks
  4. varying height and width peaks
- For the first three above schemes a single base bandwidth is reserved
- Now recall the advantages of elasticity and utilitary computing (as in Introduction)



TIVC models ([Xie et al., 2012])

Hypervisor Rate-limiting: bandwidth profiling

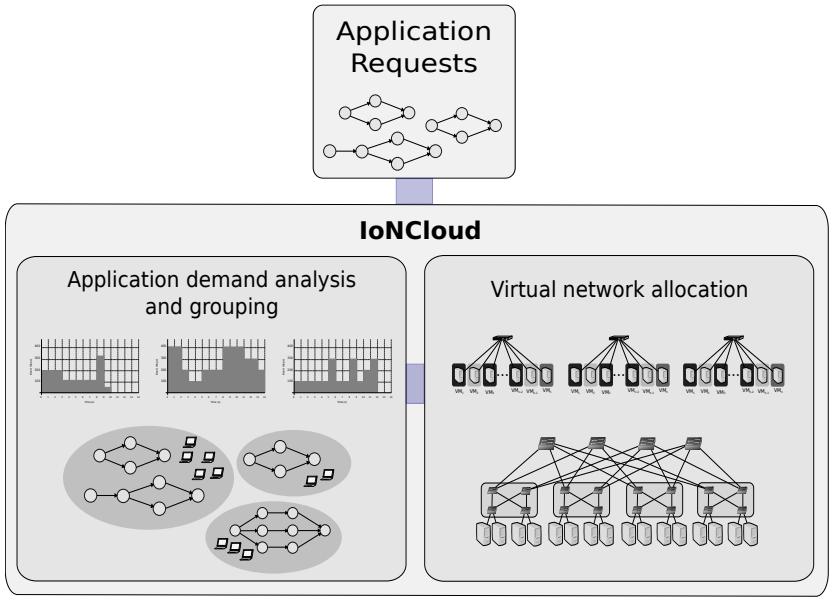


Bandwidth profiles and models in Proteus ([Xie et al., 2012])

Hypervisor Rate-limiting: bandwidth profiling

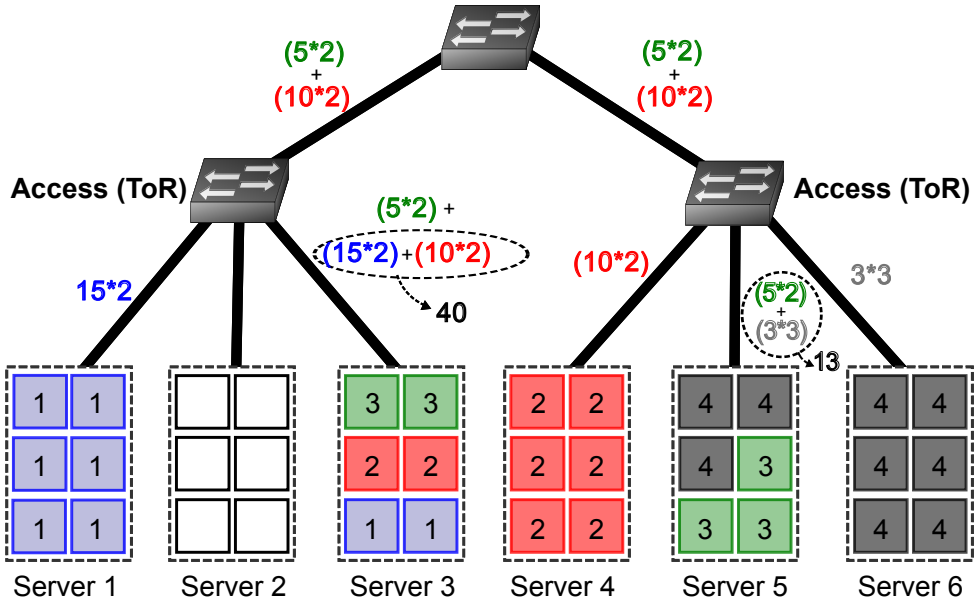
IoNCloud [Neves et al., 2013]

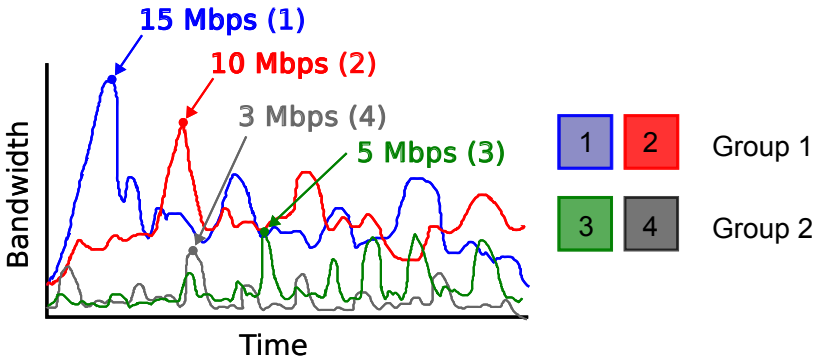
- Focuses on:
  - minimizing network underutilization
  - predictably share bandwidth among applications
- Groups applications in virtual networks:
  - temporal bandwidth demands
  - complementary bandwidth requirements



Hypervisor Rate-limiting: bandwidth profiling

IoNCloud [Neves et al., 2013]





Software-Defined Cloud Network

- Cloud deployability is still hard for several reasons:
  - tenants are unable to specify network QoS requirements
  - middleboxes (e.g. firewalls, caches and load balancers) cannot be installed in the cloud network
  - applications may need to rewritten to adapt to the cloud environment

Approach

- SDN used to control the entire cloud network

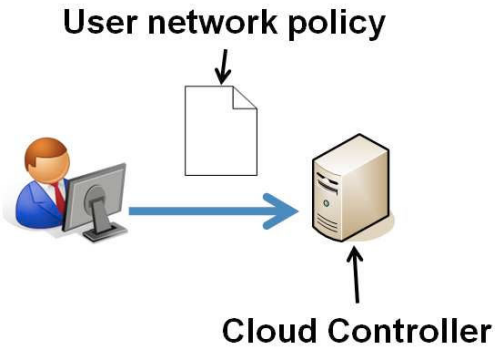
Software-Defined Cloud Network

CloudNaaS [Benson et al., 2011]

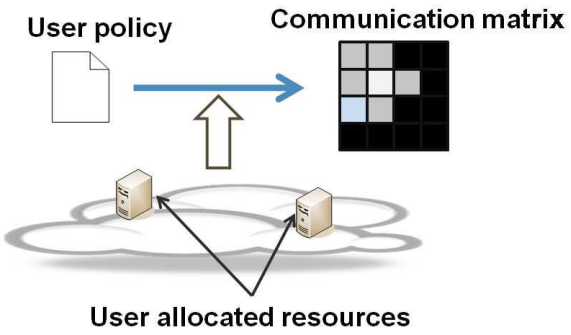
- VN architecture for deploying and managing applications in clouds
- Includes application-specific address spaces, middlebox traversal, network broadcasting, VM grouping, and bandwidth reservation
- Cloud Controller installs VMs on servers and calculates **communication matrix**
- Network controller calculates paths in accordance to the communication matrix and tenant policies
- Allocation is performed by converting the calculated paths in OpenFlow rules and installing them into the switches

- Requires all network devices to support (OpenFlow-based) SDN

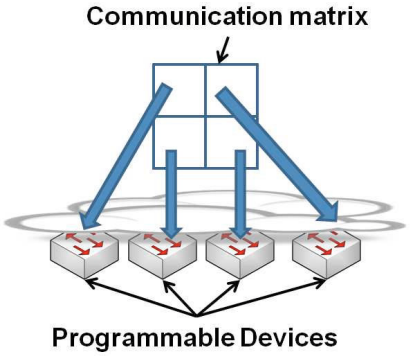
CloudNaaS framework



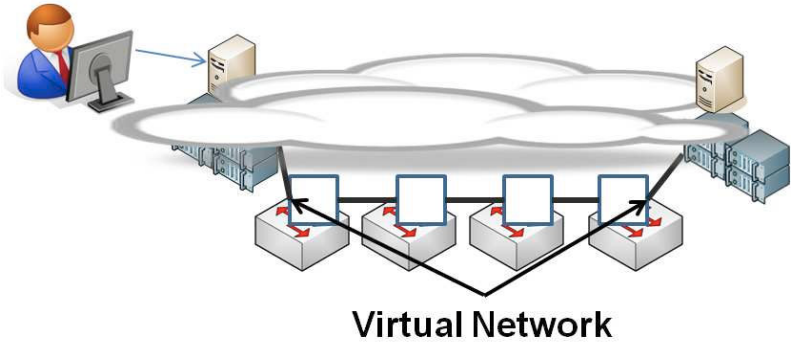
(a) Specify tenant requirements



(b) Convert requirements into a communication matrix



(c) Run network allocator and compile matrix entries into network-level rules



(d) Install rules and configure paths into devices

adapted from [Benson et al., 2011]

Questions

1. Name three potential improvements made possible by DCN virtualization.
2. Discuss how the scalability of VLANs can be improved.
3. How can virtualization overcome the limitations of current addressing schemes in data center networks?
4. What is the main purpose of NetLord?
5. What is the difference between bandwidth capping and bandwidth profiling?
6. SDNs are a trend in data center networks. In your opinion, why is that so?

Outline

Key (Research) Points

- Virtual data-center embedding
- Energy optimization
- Virtualized edge data centers
- Network programmability: flexibility
- Network performance guarantees
- Data center monitoring & fault handling
- Security
- Pricing

[Bari et al., 2012]

Virtual data center embedding

- Efficient mapping of virtual resources and services to physical ones
- Resource demand for data center applications can change over time
- Hardware configuration can change through expansion or failures

[Ballani et al., 2011b; Bari et al., 2012; Bodík et al., 2012]

[Esteves et al., 2013; Giurgiu et al., 2012; Guo et al., 2010; Rabbani et al., 2013; Zhani et al., 2013]

Virtual data center embedding

- Efficient mapping of virtual resources and services to physical ones
- Resource demand for data center applications can change over time
- Hardware configuration can change through expansion or failures

Points

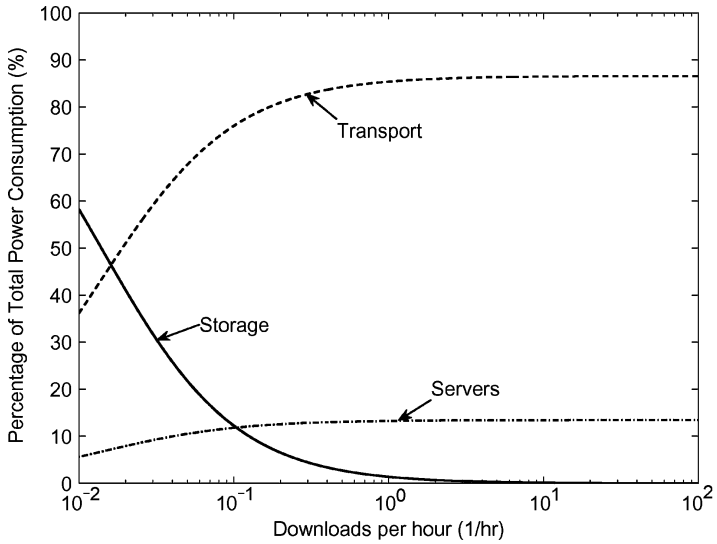
- Design of a **re-embedding algorithm** – should consider reconfiguration cost of VMs and virtual topologies
- Consider **network components** (not only servers) when embedding

[Ballani et al., 2011b; Bari et al., 2012; Bodík et al., 2012]

[Esteves et al., 2013; Giurgiu et al., 2012; Guo et al., 2010; Rabbani et al., 2013; Zhani et al., 2013]

Energy Optimization

- Plethora of trade-offs: energy consumption (kWh), carbon footprint, traffic surges, electricity cost, fault tolerance, QoE
- Network has a prominent weight in energy consumption and soon becomes the dominant cost



Transport x Storage x Processing [Baliga et al., 2011]

[Abts et al., 2010; Baliga et al., 2011; Gao et al., 2012; Heller et al., 2010; Vasić et al., 2011]

Energy Optimization

- Network virtualization can decrease energy costs by reducing the number of active physical elements
- Dynamically powering off switches and links
- Consolidating virtual resources onto smaller number of physical ones
- Carbon footprint is not obvious, decisions may not be really green

[Bari et al., 2012; Gao et al., 2012; Heller et al., 2010; Singh et al., 2013; Vasić et al., 2011]

Energy Optimization

- Network virtualization can decrease energy costs by reducing the number of active physical elements
- Dynamically powering off switches and links
- Consolidating virtual resources onto smaller number of physical ones
- Carbon footprint is not obvious, decisions may not be really green

Points

- Add energy cost to existing embedding algorithms
- Design better green virtual data center embedding algorithms
- Achieve energy-proportionality, when consumption is determined by server/network utilization

[Bari et al., 2012; Gao et al., 2012; Heller et al., 2010; Singh et al., 2013; Vasić et al., 2011]

Virtualized Edge Data Centers

- Scale economy leads to few, large data centers
- Position of data centers according to energy costs
- Higher communication costs and sub-optimal delay, jitter and throughput
- Proposal of smaller data centers located at the edge, closer to users
- Future cloud infrastructures will be multi-tiered

[Ahlgren et al., 2011; Bari et al., 2012; Goiri et al., 2011; Valancius et al., 2009]

Virtualized Edge Data Centers

- Scale economy leads to few, large data centers
- Position of data centers according to energy costs
- Higher communication costs and sub-optimal delay, jitter and throughput
- Proposal of smaller data centers located at the edge, closer to users
- Future cloud infrastructures will be multi-tiered

Points

- Optimal service placement problem in multi-tiered cloud
- Efficient monitoring of services hosted in multiple data centers

[Ahlgren et al., 2011; Bari et al., 2012; Goiri et al., 2011; Valancius et al., 2009]

Network Programmability

- Ordinary virtualization requires the use of the same L2/L3 protocols (e.g. IPv4 and Ethernet) by all tenants
- Decomposing network functions increases **flexibility** and **innovation**
  - facilitate the introduction of new protocols, services, and architectures
  - run third-party code on network device in both control/data planes
  - modular interface that separates physical topologies from virtual ones
- SDN offers simple API for programming the network control plane

[Bari et al., 2012; Curtis et al., 2011b; Porras et al., 2012; Tootoonchian and Ganjali, 2010]

Network Programmability

- Ordinary virtualization requires the use of the same L2/L3 protocols (e.g. IPv4 and Ethernet) by all tenants
- Decomposing network functions increases **flexibility** and **innovation**
  - facilitate the introduction of new protocols, services, and architectures
  - run third-party code on network device in both control/data planes
  - modular interface that separates physical topologies from virtual ones
- SDN offers simple API for programming the network control plane

Points

- Mechanisms to determine how much control delegate to tenants
- Flexibility programming network devices vs. ensuring safe and secured co-existence of multiple tenants

[Bari et al., 2012; Curtis et al., 2011b; Porras et al., 2012; Tootoonchian and Ganjali, 2010]

Network Performance Guarantees

- Applications have diverse performance requirements
- DCN performance needs to be predictable under very different application requirements
- Many recent mechanisms to provide guaranteed bandwidth, bandwidth proportionality
- TCP incast collapse and TCP outcast unfairness problems

[Ballani et al., 2011b; Bari et al., 2012; Lam et al., 2012; Popa et al., 2012; Rodrigues et al., 2011]
[Jeyakumar et al., 2012; Mogul and Popa, 2012; Wilson et al., 2011; Wu et al., 2010; Xie et al., 2012]
[Prakash et al., 2012; Wu et al., 2010; Zhang and Ansari, 2013]

Network Performance Guarantees

- Applications have diverse performance requirements
- DCN performance needs to be predictable under very different application requirements
- Many recent mechanisms to provide guaranteed bandwidth, bandwidth proportionality
- TCP incast collapse and TCP outcast unfairness problems

Points

- Scalability, efficiency, predictability, work-conserving and intra-tenant fairness

---

[Ballani et al., 2011b; Bari et al., 2012; Lam et al., 2012; Popa et al., 2012; Rodrigues et al., 2011]

[Jeyakumar et al., 2012; Mogul and Popa, 2012; Wilson et al., 2011; Wu et al., 2010; Xie et al., 2012]

[Prakash et al., 2012; Wu et al., 2010; Zhang and Ansari, 2013]

Monitoring & Fault Handling

- InPs responsible for managing physical resources of their data centers
- SPs responsible for managing virtual resources of their virtual data centers
- Monitoring is a challenging task
  - large number of resources in production data centers
  - centralized monitoring suffer low scalability and resilience

[Bari et al., 2012; Heller et al., 2010; Wuhib et al., 2009; Xu and Wang, 2008; Yuan et al., 2010]

Monitoring & Fault Handling

- InPs responsible for managing physical resources of their data centers
- SPs responsible for managing virtual resources of their virtual data centers
- Monitoring is a challenging task
  - large number of resources in production data centers
  - centralized monitoring suffer low scalability and resilience

Points

- Define interplay between the monitoring systems InPs and SPs
- Minimize negative impact of management on network performance
- Aggregate relevant information without hurting accuracy

[Bari et al., 2012; Heller et al., 2010; Wuhib et al., 2009; Xu and Wang, 2008; Yuan et al., 2010]

Monitoring & Fault Handling

- Failure detection, mitigation and recovery are fundamental requirements to any data center architecture
- Reactive failure handing may require a long response time
- Proactive fault management is often ensured by redundancy
- Early attempts focus on mitigating the impact of failures

[Bari et al., 2012; Bodík et al., 2012; Wu et al., 2012; Zhou et al., 2012]

Monitoring & Fault Handling

- Failure detection, mitigation and recovery are fundamental requirements to any data center architecture
- Reactive failure handing may require a long response time
- Proactive fault management is often ensured by redundancy
- Early attempts focus on mitigating the impact of failures

Points

- Offering high reliability without incurring excessive costs is an interesting research problem
- Fast and efficient recovery mechanisms also require future exploration

[Bari et al., 2012; Bodík et al., 2012; Wu et al., 2012; Zhou et al., 2012]

Security

- Shared nature of data center creates new attack vectors
- Network virtualization can help improve the security by reducing information leakage and interference

- Virtualization technology can be exploited both in hosts (VMs) and programmable network devices
- An attack can propagate from a compromised VM to the hypervisor, and eventually to all virtual networks sharing that server

[Bari et al., 2012; Ristenpart et al., 2009; Shieh et al., 2010, 2011]

Security

- Shared nature of data center creates new attack vectors
- Network virtualization can help improve the security by reducing information leakage and interference

- Virtualization technology can be exploited both in hosts (VMs) and programmable network devices
- An attack can propagate from a compromised VM to the hypervisor, and eventually to all virtual networks sharing that server

Points

- Designing secure virtualization architectures immune to these security vulnerabilities

[Bari et al., 2012; Ristenpart et al., 2009; Shieh et al., 2010, 2011]

Security

- Provide monitoring and auditing infrastructures
- Detect malicious activities from both tenants and infrastructure providers
- Data-center network traffic exhibits different characteristics than the traffic in traditional data networks
- Harder to detect network anomalies in virtualized data-center networks

[Bari et al., 2012; Shieh et al., 2010, 2011]

Security

- Provide monitoring and auditing infrastructures
- Detect malicious activities from both tenants and infrastructure providers
- Data-center network traffic exhibits different characteristics than the traffic in traditional data networks
- Harder to detect network anomalies in virtualized data-center networks

Points

- Novel designs of scalable and efficient mechanisms for monitoring and auditing virtualized data centers

[Bari et al., 2012; Shieh et al., 2010, 2011]

Security

- Different tenants may desire different levels of security
- This introduces the additional complexity of managing heterogeneous security mechanisms and policies

[Bari et al., 2012]

Security

- Different tenants may desire different levels of security
- This introduces the additional complexity of managing heterogeneous security mechanisms and policies

Points

- The co-existence and interaction of multiple security systems expected in a multi-tenant data center is an issue that has not been addressed before
- Potential conflicts between firewalls and intrusion detection systems policies of infrastructure providers and service providers, need to be detected and solved

[Bari et al., 2012]

Pricing

- Directly affects the income of the infrastructure provider
- Provides incentives for tenants

[Ballani et al., 2011a,b; Bari et al., 2012; Zaheer et al., 2010]

Pricing

- Directly affects the income of the infrastructure provider
- Provides incentives for tenants

Points

- Well-designed pricing scheme should be both fair and efficient
- Fairness means that identical good should be sold at identical price
- Efficiency means the price of the good should lead to efficient outcomes (e.g. matching supply and demand)

[Ballani et al., 2011a,b; Bari et al., 2012; Zaheer et al., 2010]

Questions

1. How can virtual data centers embedding be improved?
2. Why energy optimization is an important research topic?
3. Discuss the importance of Edge Data Centers to the current service provisioning model.
4. Name two research topics for Network Programmability.
5. Which network resource sharing scheme do tenants need? Is it good for providers?
6. Is there room for improvement on monitoring and fault handing techniques in data centers?
7. Does all tenants have the same security requirements? Why?
8. Why are pricing models important in cloud computing?

Outline

Review

- Introduction (what's new?)
- The rise of clouds
- Network virtualization
- Software Defined Network
- Cloud networking
- Virtualization of cloud networks
- Research challenges

Review

- Introduction (what's new?)
- **The rise of clouds**
- Network virtualization
- Software Defined Network
- Cloud networking
- Virtualization of cloud networks
- Research challenges

Definition, benefits, types of clouds, challenges.

Review

- Introduction (what's new?)
- The rise of clouds
- **Network virtualization**
- Software Defined Network
- Cloud networking
- Virtualization of cloud networks
- Research challenges

Definition, benefits, types of virtualization, challenges.

Review

- Introduction (what's new?)
- The rise of clouds
- Network virtualization
- **Software Defined Network**
- Cloud networking
- Virtualization of cloud networks
- Research challenges

Motivation, Definition, Architecture, Controller, OpenFlow, use in Datacenters, challenges.

Review

- Introduction (what's new?)
- The rise of clouds
- Network virtualization
- Software Defined Network
- **Cloud networking**
- Virtualization of cloud networks
- Research challenges

Data centers, topologies, expansion, traffic, routing, addressing, challenges.

Review

- Introduction (what's new?)
- The rise of clouds
- Network virtualization
- Software Defined Network
- Cloud networking
- **Virtualization of cloud networks**
- Research challenges

Benefits, adapting conventional approaches, addressing, hypervisor rate-limiting, SDNs.

Review

- Introduction (what's new?)
- The rise of clouds
- Network virtualization
- Software Defined Network
- Cloud networking
- Virtualization of cloud networks
- **Research challenges**

VDC embedding, energy optimization, virtualized edge DCs, network programmability, performance guarantees, monitoring & faults, security, pricing.

Parting Thoughts

Network virtualization offers several benefits, including great flexibility, and can become a **key element for** the **evolution** of **cloud** data-center networks.

Lots of open challenges and **interesting research** topics in this field.

Outline

INSTITUTO DE INFORMÁTICA UFRGS

Opportunities

Prospective PhD candidates are welcome in our Research Group. Contact me here for more info.

Postgraduate students **regardless of nationality** can receive a brazilian grant, which covers tuition *and* living expenses

UFRGS is ranked as the **best** University in Brazil according to the official evaluation.

Areas

- Performance & Security of...
- Peer-to-Peer = Large Scale Descentralized Systems
- Data Center Networks
- Software-Defined Networks
- Information-Centric Networks

PPGC – Postgraduation Program in Computing

- Areas: Computer Networks, Parallel and Distributed Processing, Multimedia, Microelectronics, Embedded Systems, Artificial Intelligence, Robotics, Information Systems, Databases, Software Engineering, Formal Methods, Computer Graphics, Bioinformatics...
- Supervisors: 53
- Current PhDs: 104 (virtually all with grant) – 204 already graduated
- Current MSc: 147 (part and full-time) – 1,294 MScs already graduated
- Labs: 11 racks, 60 switches and 2,800+ points, 24 printers, 34 HiRes projectors
- Research projects/funding: ~49
- POSCOMP Exam (equivalent to GRE on CS) may be required
- http://ppgc.inf.ufrgs.br/en/

INF – Institute of Informatics

- Created: November 9th, 1989 (23 year-old)
- Academic staff: ~73
- Technical staff: ~28
- Current postgraduate students: ~251
- Current undergraduate students: ~752
- Sites: http://www.inf.ufrgs.br/, http://www.inf.ufrgs.br/en/

UFRGS – Federal University of Rio Grande do Sul

- Created: 1895
- Departments: 27
- Postgraduate Programs: 143
- Academic staff: 2,250 full-time faculty members
- Postgraduate students: 8,900
- Undergraduate students: 24,900

Obrigado! Thanks! Gracias!

Rodrigo, Daniel and Marinho.

We are in debt to CNPq (under Project SecFuNet grant), RNP CTIC (Projects ReVir and AltoStratus) and FAPERGS PRONEM (NPRV), which funded this work.

Contact information
Prof. Marinho P. Barcellos
marinho@inf.ufrgs.br

References I

Dennis Abts and Bob Felderman. A guided tour of data-center networking. *Commun. ACM*, 55(6):44–51, June 2012. ISSN 0001-0782. doi: 10.1145/2184319.2184335. URL http://doi.acm.org/10.1145/2184319.2184335.

Dennis Abts, Michael R. Marty, Philip M. Wells, Peter Klausler, and Hong Liu. Energy proportional datacenter networks. *SIGARCH Comput. Archit. News*, 38(3):338–347, June 2010. ISSN 0163-5964. doi: 10.1145/1816038.1816004. URL http://doi.acm.org/10.1145/1816038.1816004.

Hussam Abu-Libdeh, Paolo Costa, Antony Rowstron, Greg O'Shea, and Austin Donnelly. Symbiotic routing in future data centers. In *Proceedings of the ACM SIGCOMM 2010 conference*, SIGCOMM '10, pages 51–62, New York, NY, USA, 2010. ACM. ISBN 978-1-4503-0201-2. doi: 10.1145/1851182.1851191. URL http://doi.acm.org/10.1145/1851182.1851191.

B. Ahlgren, P.A. Aranda, P. Chemouil, S. Oueslati, L.M. Correia, H. Karl, M. Sö andllner, and A. Welin. Content, connectivity, and cloud: ingredients for the network of the future. *Communications Magazine, IEEE*, 49(7):62 –70, july 2011. ISSN 0163-6804. doi: 10.1109/MCOM.2011.5936156.

Mohammad Al-Fares, Alexander Loukissas, and Amin Vahdat. A scalable, commodity data center network architecture. In *Proceedings of the ACM SIGCOMM 2008 conference on Data communication*, SIGCOMM '08, pages 63–74, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-175-0. doi: 10.1145/1402958.1402967. URL http://doi.acm.org/10.1145/1402958.1402967.

Mohammad Al-Fares, Sivasankar Radhakrishnan, Barath Raghavan, Nelson Huang, and Amin Vahdat. Hedera: dynamic flow scheduling for data center networks. In *Proceedings of the 7th USENIX conference on Networked systems design and implementation*, NSDI'10, pages 19–19, Berkeley, CA, USA, 2010. USENIX Association. URL http://dl.acm.org/citation.cfm?id=1855711.1855730.

Michael Armbrust, Armando Fox, Rean Griffith, Anthony D. Joseph, Randy Katz, Andy Konwinski, Gunho Lee, David Patterson, Ariel Rabkin, Ion Stoica, and Matei Zaharia. A view of cloud computing. *Commun. ACM*, 53:50–58, April 2010. doi: http://doi.acm.org/10.1145/1721654.1721672.

References II

Lee Badger, Tim Grance, Robert Patt-Corner, and Jeff Voas. The NIST Definition of Cloud Computing. Technical report, National Institute of Standards and Technology, 2009. URL http://csrc.nist.gov/publications/drafts/800-145/Draft-SP-800-145_cloud-definition.pdf.

J. Baliga, R.W.A. Ayre, K. Hinton, and R.S. Tucker. Green cloud computing: Balancing energy in processing, storage, and transport. *Proceedings of the IEEE*, 99(1):149 –167, jan. 2011. ISSN 0018-9219. doi: 10.1109/JPROC.2010.2060451.

Hitesh Ballani, Paolo Costa, Thomas Karagiannis, and Ant Rowstron. The price is right: towards location-independent costs in datacenters. In *ACM HotNets*, 2011a.

Hitesh Ballani, Paolo Costa, Thomas Karagiannis, and Ant Rowstron. Towards predictable datacenter networks. In *ACM SIGCOMM*, 2011b.

Hitesh Ballani, Keon Jang, Thomas Karagiannis, Changhoon Kim, Dinan Gunawardena, and Greg O'Shea. Chatty tenants and the cloud network sharing problem. In *Proceedings of the 10th USENIX conference on Networked Systems Design and Implementation*, NSDI'13, pages 1–14, Berkeley, CA, USA, 2013. USENIX Association.

M. Bari, R. Boutaba, R. Esteves, L. Granville, M. Podlesny, M. Rabbani, Q. Zhang, and M. Zhani. Data center network virtualization: A survey. *Communications Surveys Tutorials, IEEE*, PP(99):1–20, 2012. ISSN 1553-877X. doi: 10.1109/SURV.2012.090512.00043.

A. Belbekkouche, M. Hasan, and A. Karmouch. Resource discovery and allocation in network virtualization. *Communications Surveys & Tutorials, IEEE*, PP(99):1–15, 2012. ISSN 1553-877X. doi: 10.1109/SURV.2011.122811.00060.

Theophilus Benson, Aditya Akella, and David A. Maltz. Network traffic characteristics of data centers in the wild. In *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement*, IMC '10, pages 267–280, New York, NY, USA, 2010. ACM. ISBN 978-1-4503-0483-2. doi: 10.1145/1879141.1879175. URL http://doi.acm.org/10.1145/1879141.1879175.

References III

Theophilus Benson, Aditya Akella, Anees Shaikh, and Sambit Sahu. Cloudnaas: a cloud networking platform for enterprise applications. In *ACM Symposium on Cloud Computing (SOCC)*, 2011.

Peter Bodík, Ishai Menache, Mosharaf Chowdhury, Pradeepkumar Mani, David A. Maltz, and Ion Stoica. Surviving failures in bandwidth-constrained datacenters. In *Proceedings of the ACM SIGCOMM 2012 conference on Applications, technologies, architectures, and protocols for computer communication*, SIGCOMM '12, pages 431–442, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1419-0. doi: 10.1145/2342356.2342439. URL http://doi.acm.org/10.1145/2342356.2342439.

Kevin D. Bowers, Ari Juels, and Alina Oprea. Hail: a high-availability and integrity layer for cloud storage. In *Proceedings of the 16th ACM conference on Computer and communications security*, CCS '09, pages 187–198, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-894-0.

Kevin D. Bowers, Marten van Dijk, Ari Juels, Alina Oprea, and Ronald L. Rivest. How to tell if your cloud files are vulnerable to drive crashes. In *Proceedings of the 18th ACM conference on Computer and communications security*, CCS '11, pages 501–514, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0948-6.

Sven Bugiel, Stefan Nürnberger, Thomas Pöppelmann, Ahmad-Reza Sadeghi, and Thomas Schneider. Amazonia: when elasticity snaps back. In *Proceedings of the 18th ACM conference on Computer and communications security*, CCS '11, pages 389–400, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0948-6.

Kai Chen, Ankit Singlay, Atul Singhz, Kishore Ramachandranz, Lei Xuz, Yueping Zhangz, Xitao Wen, and Yan Chen. Osa: an optical switching architecture for data center networks with unprecedented flexibility. In *Proceedings of the 9th USENIX conference on Networked Systems Design and Implementation*, NSDI'12, pages 18–18, Berkeley, CA, USA, 2012. USENIX Association. URL http://dl.acm.org/citation.cfm?id=2228298.2228323.

Yanpei Chen, Vern Paxson, and Randy H. Katz. What's new about cloud computing security? Technical Report UCB/EECS-2010-5, EECS Department, University of California, Berkeley, Jan 2010. URL http://www.eecs.berkeley.edu/Pubs/TechRpts/2010/EECS-2010-5.html.

References IV

Yingying Chen, S. Jain, V.K. Adhikari, Zhi-Li Zhang, and Kuai Xu. A first look at inter-data center traffic characteristics via yahoo! datasets. In *INFOCOM, 2011 Proceedings IEEE*, pages 1620–1628, april 2011. doi: 10.1109/INFCOM.2011.5934955.

N.M. Mosharaf Kabir Chowdhury and Raouf Boutaba. A Survey of Network Virtualization. *Computer Networks*, 54(5): 862–876, 2010. ISSN 1389-1286. doi: 10.1016/j.comnet.2009.10.017.

Paolo Costa, Thomas Zahn, Ant Rowstron, Greg O'Shea, and Simon Schubert. Why should we integrate services, servers, and networking in a data center? In *Proceedings of the 1st ACM workshop on Research on enterprise networking*, WREN '09, pages 111–118, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-443-0. doi: 10.1145/1592681.1592699. URL http://doi.acm.org/10.1145/1592681.1592699.

Andrew R. Curtis, S. Keshav, and Alejandro Lopez-Ortiz. Legup: using heterogeneity to reduce the cost of data center network upgrades. In *Proceedings of the 6th International COnference*, Co-NEXT '10, pages 14:1–14:12, New York, NY, USA, 2010. ACM. ISBN 978-1-4503-0448-1.

Andrew R. Curtis, Jeffrey C. Mogul, Jean Tourrilhes, Praveen Yalagandula, Puneet Sharma, and Sujata Banerjee. Devoflow: scaling flow management for high-performance networks. In *Proceedings of the ACM SIGCOMM 2011 conference*, SIGCOMM '11, pages 254–265, New York, NY, USA, 2011a. ACM. ISBN 978-1-4503-0797-0. doi: 10.1145/2018436.2018466. URL http://doi.acm.org/10.1145/2018436.2018466.

Andrew R. Curtis, Jeffrey C. Mogul, Jean Tourrilhes, Praveen Yalagandula, Puneet Sharma, and Sujata Banerjee. Devoflow: scaling flow management for high-performance networks. In *Proceedings of the ACM SIGCOMM 2011 conference*, SIGCOMM '11, pages 254–265, New York, NY, USA, 2011b. ACM. ISBN 978-1-4503-0797-0. doi: 10.1145/2018436.2018466. URL http://doi.acm.org/10.1145/2018436.2018466.

A.R. Curtis, T. Carpenter, M. Elsheikh, A. Lopez-Ortiz, and S. Keshav. Rewire: An optimization-based framework for unstructured data center network design. In *INFOCOM, 2012 Proceedings IEEE*, pages 1116 –1124, march 2012.

References V

Norbert Egi, Adam Greenhalgh, Mark Handley, Mickael Hoerdt, Felipe Huici, Laurent Mathy, and Panagiotis Papadimitriou. Forwarding path architectures for multicore software routers. In *Proceedings of the Workshop on Programmable Routers for Extensible Services of Tomorrow*, PRESTO '10, pages 3:1–3:6, New York, NY, USA, 2010. ACM. ISBN 978-1-4503-0467-2. doi: 10.1145/1921151.1921155.

Rafael Esteves, Lisandro Zambenedetti Granville, Hadi Bannazadeh, and Raouf Boutaba. Paradigm-based adaptive provisioning in virtualized data centers. In *Proceedings of the 13th IFIP/IEEE Integrated Network Management Symposium (IM 2013)*, Ghent, Belgium, may 2013.

N.C. Fernandes and O.C.M.B. Duarte. Xnetmon: A network monitor for securing virtual networks. In *Communications (ICC), 2011 IEEE International Conference on*, pages 1–5, june 2011. doi: 10.1109/icc.2011.5963443.

Peter Xiang Gao, Andrew R. Curtis, Bernard Wong, and Srinivasan Keshav. It's not easy being green. In *Proceedings of the ACM SIGCOMM 2012 conference on Applications, technologies, architectures, and protocols for computer communication*, SIGCOMM '12, pages 211–222, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1419-0. doi: 10.1145/2342356.2342398. URL http://doi.acm.org/10.1145/2342356.2342398.

Ioana Giurgiu, Claris Castillo, Asser N. Tantawi, and Malgorzata Steinder. Enabling efficient placement of virtual infrastructures in the cloud. In *Middleware*, pages 332–353, 2012.

I. Goiri, Kien Le, J. Guitart, J. Torres, and R. Bianchini. Intelligent placement of datacenters for internet services. In *Distributed Computing Systems (ICDCS), 31st International Conference on*, pages 131–142, june 2011. doi: 10.1109/ICDCS.2011.19.

Albert Greenberg, James R. Hamilton, Navendu Jain, Srikanth Kandula, Changhoon Kim, Parantap Lahiri, David A. Maltz, Parveen Patel, and Sudipta Sengupta. Vl2: a scalable and flexible data center network. In *Proceedings of the ACM SIGCOMM 2009 conference on Data communication*, SIGCOMM '09, pages 51–62, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-594-9. doi: 10.1145/1592568.1592576. URL http://doi.acm.org/10.1145/1592568.1592576.

References VI

Albert Greenberg, James R. Hamilton, Navendu Jain, Srikanth Kandula, Changhoon Kim, Parantap Lahiri, David A. Maltz, Parveen Patel, and Sudipta Sengupta. Vl2: a scalable and flexible data center network. *Commun. ACM*, 54 (3):95–104, March 2011. ISSN 0001-0782. doi: 10.1145/1897852.1897877. URL http://doi.acm.org/10.1145/1897852.1897877.

Chuanxiong Guo, Haitao Wu, Kun Tan, Lei Shi, Yongguang Zhang, and Songwu Lu. Dcell: a scalable and fault-tolerant network structure for data centers. In *Proceedings of the ACM SIGCOMM 2008 conference on Data communication*, SIGCOMM '08, pages 75–86, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-175-0. doi: 10.1145/1402958.1402968. URL http://doi.acm.org/10.1145/1402958.1402968.

Chuanxiong Guo, Guohan Lu, Dan Li, Haitao Wu, Xuan Zhang, Yunfeng Shi, Chen Tian, Yongguang Zhang, and Songwu Lu. Bcube: a high performance, server-centric network architecture for modular data centers. In *Proceedings of the ACM SIGCOMM 2009 conference on Data communication*, SIGCOMM '09, pages 63–74, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-594-9. doi: 10.1145/1592568.1592577. URL http://doi.acm.org/10.1145/1592568.1592577.

Chuanxiong Guo, Guohan Lu, Helen J. Wang, Shuang Yang, Chao Kong, Peng Sun, Wenfei Wu, and Yongguang Zhang. Secondnet: a data center network virtualization architecture with bandwidth guarantees. In *ACM CoNEXT*, 2010.

Fang Hao, T. V. Lakshman, Sarit Mukherjee, and Haoyu Song. Enhancing dynamic cloud-based services using network virtualization. In *Proceedings of the 1st ACM workshop on Virtualized infrastructure systems and architectures*, VISA '09, pages 37–44, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-595-6. doi: 10.1145/1592648.1592655.

Fang Hao, T. V. Lakshman, Sarit Mukherjee, and Haoyu Song. Secure cloud computing with a virtualized network infrastructure. In *Proceedings of the 2nd USENIX conference on Hot topics in cloud computing*, HotCloud'10, pages 16–16, Berkeley, CA, USA, 2010. USENIX Association.

References VII

Brandon Heller, Srini Seetharaman, Priya Mahadevan, Yiannis Yiakoumis, Puneet Sharma, Sujata Banerjee, and Nick McKeown. Elastictree: saving energy in data center networks. In *Proceedings of the 7th USENIX conference on Networked systems design and implementation*, NSDI'10, pages 17–17, Berkeley, CA, USA, 2010. USENIX Association. URL http://dl.acm.org/citation.cfm?id=1855711.1855728.

Judith Hurwitz. 5 Big Cloud Trends For 2012, December 2011. Available at: http://www.informationweek.com/cloud-computing/infrastructure/5-big-cloud-trends-for-2012/232200551.

Vimalkumar Jeyakumar, Mohammad Alizadeh, David Mazières, Balaji Prabhakar, and Changhoon Kim. Eyeq: practical network performance isolation for the multi-tenant cloud. In *Proceedings of the 4th USENIX conference on Hot Topics in Cloud Ccomputing*, HotCloud'12, pages 8–8, Berkeley, CA, USA, 2012. USENIX Association. URL http://dl.acm.org/citation.cfm?id=2342763.2342771.

Vimalkumar Jeyakumar, Mohammad Alizadeh, David Mazières, Balaji Prabhakar, Changhoon Kim, and Albert Greenberg. Eyeq: Practical network performance isolation at the edge. In *Proceedings of the 10th USENIX conference on Networked Systems Design and Implementation*, NSDI'13, pages 1–15, Berkeley, CA, USA, 2013. USENIX Association.

Srikanth Kandula, Sudipta Sengupta, Albert Greenberg, Parveen Patel, and Ronnie Chaiken. The nature of data center traffic: measurements & analysis. In *Proceedings of the 9th ACM SIGCOMM conference on Internet measurement conference*, IMC '09, pages 202–208, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-771-4. doi: 10.1145/1644893.1644918. URL http://doi.acm.org/10.1145/1644893.1644918.

A. Khan, A. Zugenmaier, D. Jurca, and W. Kellerer. Network virtualization: a hypervisor for the internet? *Communications Magazine, IEEE*, 50(1):136–143, january 2012. ISSN 0163-6804. doi: 10.1109/MCOM.2012.6122544.

Teemu Koponen, Martin Casado, Natasha Gude, Jeremy Stribling, Leon Poutievski, Min Zhu, Rajiv Ramanathan, Yuichiro Iwata, Hiroaki Inoue, Takayuki Hama, et al. Onix: A distributed control platform for large-scale production networks. In *OSDI*, volume 10, pages 1–6, 2010.

References VIII

Vinh The Lam, Sivasankar Radhakrishnan, Rong Pan, Amin Vahdat, and George Varghese. Netshare and stochastic netshare: predictable bandwidth allocation for data centers. *ACM SIGCOMM CCR*, 42(3), 2012.

LAN/MAN Standards Committee. Ieee standard for local and metropolitan area networks – virtual bridged local area networks. IEEE Std 802.1Q-2011 (incorporates 802.1Q1998, 802.1u-2001, 802.1v-2001, 802.1s-2002, and 802.1Q-2005), august 2011.

A. Lara, A. Kolasani, and B. Ramamurthy. Network innovation using openflow: A survey, 2013. ISSN 1553-877X.

Eng Keong Lua, J. Crowcroft, M. Pias, R. Sharma, and S. Lim. A survey and comparison of peer-to-peer overlay network schemes. *Communications Surveys Tutorials, IEEE*, 7(2):72–93, quarter 2005. ISSN 1553-877X. doi: 10.1109/COMST.2005.1610546.

V Mann, A Vishnoi, K Kannan, and S Kalyanaraman. CrossRoads: Seamless VM mobility across data centers through software defined networking. *Network Operations and Management Symposium (NOMS), 2012 IEEE*, pages 88–96, 2012.

Daniel Stefani Marcon, Rodrigo Ruas Oliveira, Miguel Cardoso Neves, Luciana Salete Buriol, Luciano Paschoal Gaspary, and Marinho Pilla Barcellos. Trust-based grouping for cloud datacenters: improving security in shared infrastructures. In *Proceedings of the 12th international IFIP TC 6 conference on Networking*, IFIP'13, Berlin, Heidelberg, 2013. Springer-Verlag.

Nick McKeown, Tom Anderson, Hari Balakrishnan, Guru Parulkar, Larry Peterson, Jennifer Rexford, Scott Shenker, and Jonathan Turner. Openflow: enabling innovation in campus networks. *SIGCOMM Computer Communication Review*, 38:69–74, March 2008. ISSN 0146-4833. doi: http://doi.acm.org/10.1145/1355734.1355746.

Marc Mendonça, Bruno Nunes Astuto, Xuan Nam Nguyen, Katia Obraczka, and Thierry Turletti. A Survey of Software-Defined Networking: Past, Present, and Future of Programmable Networks, June 2013. URL http://hal.inria.fr/hal-00825087. In Submission.

References IX

Jeffrey C. Mogul and Lucian Popa. What we talk about when we talk about cloud network performance. *SIGCOMM Comput. Commun. Rev.*, 42(5):44–48, September 2012. ISSN 0146-4833. doi: 10.1145/2378956.2378964. URL http://doi.acm.org/10.1145/2378956.2378964.

Jayaram Mudigonda, Praveen Yalagandula, Jeff Mogul, Bryan Stiekes, and Yanick Pouffary. Netlord: a scalable multi-tenant network architecture for virtualized datacenters. In *Proceedings of the ACM SIGCOMM 2011 conference on SIGCOMM*, SIGCOMM '11, pages 62–73, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0797-0.

Miguel Cardoso Neves, Daniel Stefani Marcon, Rodrigo Ruas Oliveira, Leonardo Richter Bays, Tiago de Almeida, Luciano Paschoal Gaspary, and Marinho Pilla Barcellos. Ioncloud: approaching high utilization and network predictability with shared bandwidth guarantees. 2013.

Nimbus, 2012. Nimbus Home Page, 2012. Available at: http://www.nimbusproject.org/.

Daniel Nurmi, Rich Wolski, Chris Grzegorczyk, Graziano Obertelli, Sunil Soman, Lamia Youseff, and Dmitrii Zagorodnov. The eucalyptus open-source cloud-computing system. In *Proceedings of the 2009 9th IEEE/ACM International Symposium on Cluster Computing and the Grid*, CCGRID '09, pages 124–131, Washington, DC, USA, 2009. IEEE Computer Society. ISBN 978-0-7695-3622-4. doi: 10.1109/CCGRID.2009.93. URL http://dx.doi.org/10.1109/CCGRID.2009.93.

Andrew Nusca. The future of cloud computing: 9 trends for 2012, June 2012. Available at: http://www.zdnet.com/blog/btl/the-future-of-cloud-computing-9-trends-for-2012/80511.

OpenNebula, 2012. OpenNebula Home Page, 2012. Available at: http://www.opennebula.org/.

OpenQRM, 2012. OpenQRM Home Page, 2012. Available at: www.openqrm.org.

OpenStack, 2012. OpenStack Home Page, 2012. Available at: http://www.openstack.org/.

References X

Christy Pettey. Gartner Outlines Five Cloud Computing Trends That Will Affect Cloud Strategy Through 2015, 2012. Available at: http://www.gartner.com/it/page.jsp?id=1971515.

Lucian Popa, Sylvia Ratnasamy, Gianluca Iannaccone, Arvind Krishnamurthy, and Ion Stoica. A cost comparison of datacenter network architectures. In *Proceedings of the 6th International COnference*, Co-NEXT '10, pages 16:1–16:12, New York, NY, USA, 2010. ACM. ISBN 978-1-4503-0448-1. doi: 10.1145/1921168.1921189. URL http://doi.acm.org/10.1145/1921168.1921189.

Lucian Popa, Gautam Kumar, Mosharaf Chowdhury, Arvind Krishnamurthy, Sylvia Ratnasamy, and Ion Stoica. Faircloud: sharing the network in cloud computing. In *ACM SIGCOMM*, 2012. ISBN 978-1-4503-1419-0.

Lucian Popa, Praveen Yalagandula, Sujata Banerjee, and Jeffrey C. Mogul. ElasticSwitch: Practical Work-Conserving Bandwidth Guarantees for Cloud Computing. In *Proceedings of the ACM SIGCOMM 2013 conference*, SIGCOMM'13, New York, NY, USA, 2013. ACM.

Philip Porras, Seungwon Shin, Vinod Yegneswaran, Martin Fong, Mabry Tyson, and Guofei Gu. A security enforcement kernel for openflow networks. In *Proceedings of the first workshop on Hot topics in software defined networks*, HotSDN '12, pages 121–126, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1477-0. doi: 10.1145/2342441.2342466. URL http://doi.acm.org/10.1145/2342441.2342466.

Pawan Prakash, Advait Dixit, Y. Charlie Hu, and Ramana Kompella. The tcp outcast problem: exposing unfairness in data center networks. In *Proceedings of the 9th USENIX conference on Networked Systems Design and Implementation*, NSDI'12, pages 30–30, Berkeley, CA, USA, 2012. USENIX Association.

Md Rabbani, Rafael Esteves, Maxim Podlesny, Gwendal Simon, Lisandro Zambenedetti Granville, and Raouf Boutaba. On tackling virtual data center embedding problem. In *Proceedings of the 13th IFIP/IEEE Integrated Network Management Symposium (IM 2013)*, Ghent, Belgium, may 2013.

Renato Recio. The coming decade of data center networking discontinuities. ICNC, august 2012. keynote speaker.

References XI

Thomas Ristenpart, Eran Tromer, Hovav Shacham, and Stefan Savage. Hey, you, get off of my cloud: exploring information leakage in third-party compute clouds. In *ACM CCS*, 2009.

Henrique Rodrigues, Jose Renato Santos, Yoshio Turner, Paolo Soares, and Dorgival Guedes. Gatekeeper: supporting bandwidth guarantees for multi-tenant datacenter networks. In *Proceedings of the 3rd conference on I/O virtualization*, WIOV'11, pages 6–6, Berkeley, CA, USA, 2011. USENIX Association.

E. Rosen, Inc. Cisco Systems, Y. Rekhter, and Inc. Juniper Networks. Rfc 4364: Bgp/mpls ip virtual private networks (vpns). Internet Engineering Task Force (IETF). http://www.ietf.org/rfc/rfc4364.txt, February 2006.

C. Rotsos, R. Mortier, A. Madhavapeddy, B. Singh, and A.W. Moore. Cost, performance amp; flexibility in openflow: Pick three. In *Communications (ICC), 2012 IEEE International Conference on*, pages 6601–6605, 2012. doi: 10.1109/ICC.2012.6364690.

Peter Sempolinski and Douglas Thain. A comparison and critique of eucalyptus, opennebula and nimbus. In *Proceedings of the 2010 IEEE Second International Conference on Cloud Computing Technology and Science*, CLOUDCOM '10, pages 417–426, Washington, DC, USA, 2010. IEEE Computer Society. ISBN 978-0-7695-4302-4. doi: 10.1109/CloudCom.2010.42. URL http://dx.doi.org/10.1109/CloudCom.2010.42.

Rob Sherwood, Glen Gibb, Kok-Kiong Yap, Guido Appenzeller, Martin Casado, Nick McKeown, and Guru Parulkar. Can the production network be the testbed? In *Proceedings of the 9th USENIX conference on Operating systems design and implementation*, OSDI'10, pages 1–6, Berkeley, CA, USA, 2010. USENIX Association.

Alan Shieh, Srikanth Kandula, Albert Greenberg, and Changhoon Kim. Seawall: performance isolation for cloud datacenter networks. In *Proceedings of the 2nd USENIX conference on Hot topics in cloud computing*, HotCloud'10, pages 1–1, Berkeley, CA, USA, 2010. USENIX Association.

Alan Shieh, Srikanth Kandula, Albert Greenberg, Changhoon Kim, and Bikas Saha. Sharing the data center network. In *USENIX NSDI*, 2011.

References XII

Rahul Singh, David Irwin, Prashant Shenoy, and K.K. Ramakrishnan. Yank: Enabling green data centers to pull the plug. In *Proceedings of the 10th USENIX conference on Networked Systems Design and Implementation*, NSDI'13, pages 1–13, Berkeley, CA, USA, 2013. USENIX Association.

Ankit Singla, Chi-Yao Hong, Lucian Popa, and P. Brighten Godfrey. Jellyfish: networking data centers randomly. In *Proceedings of the 9th USENIX conference on Networked Systems Design and Implementation*, NSDI'12, pages 17–17, Berkeley, CA, USA, 2012. USENIX Association. URL http://dl.acm.org/citation.cfm?id=2228298.2228322.

D. Stezenbach, M. Hartmann, and K. Tutschku. Parameters and challenges for virtual network embedding in the future internet. In *Network Operations and Management Symposium (NOMS), 2012 IEEE*, pages 1272–1278, april 2012. doi: 10.1109/NOMS.2012.6212063.

S. Subashini and V. Kavitha. A survey on security issues in service delivery models of cloud computing. *Journal of Network and Computer Applications*, 34(1):1–11, 2011. doi: 10.1016/j.jnca.2010.07.006.

Amin Tootoonchian and Yashar Ganjali. Hyperflow: a distributed control plane for openflow. In *Proceedings of the 2010 internet network management conference on Research on enterprise networking*, INM/WREN'10, pages 3–3, Berkeley, CA, USA, 2010. USENIX Association. URL http://dl.acm.org/citation.cfm?id=1863133.1863136.

Vytautas Valancius, Nikolaos Laoutaris, Laurent Massoulié, Christophe Diot, and Pablo Rodriguez. Greening the internet with nano data centers. In *Proceedings of the 5th international conference on Emerging networking experiments and technologies*, CoNEXT '09, pages 37–48, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-636-6. doi: 10.1145/1658939.1658944. URL http://doi.acm.org/10.1145/1658939.1658944.

Nedeljko Vasić, Prateek Bhurat, Dejan Novaković, Marco Canini, Satyam Shekhar, and Dejan Kostić. Identifying and using energy-critical paths. In *Proceedings of the Seventh COnference on emerging Networking EXperiments and Technologies*, CoNEXT '11, pages 18:1–18:12, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-1041-3. doi: 10.1145/2079296.2079314. URL http://doi.acm.org/10.1145/2079296.2079314.

References XIII

Christo Wilson, Hitesh Ballani, Thomas Karagiannis, and Ant Rowtron. Better never than late: meeting deadlines in datacenter networks. In *Proceedings of the ACM SIGCOMM 2011 conference*, SIGCOMM '11, pages 50–61, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0797-0. doi: 10.1145/2018436.2018443. URL http://doi.acm.org/10.1145/2018436.2018443.

Haitao Wu, Zhenqian Feng, Chuanxiong Guo, and Yongguang Zhang. Ictcp: Incast congestion control for tcp in data center networks. In *Proceedings of the 6th International COnference*, Co-NEXT '10, pages 13:1–13:12, New York, NY, USA, 2010. ACM. ISBN 978-1-4503-0448-1. doi: 10.1145/1921168.1921186. URL http://doi.acm.org/10.1145/1921168.1921186.

Xin Wu, Daniel Turner, Chao-Chih Chen, David A. Maltz, Xiaowei Yang, Lihua Yuan, and Ming Zhang. Netpilot: automating datacenter network failure mitigation. In *Proceedings of the ACM SIGCOMM 2012 conference on Applications, technologies, architectures, and protocols for computer communication*, SIGCOMM '12, pages 419–430, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1419-0. doi: 10.1145/2342356.2342438. URL http://doi.acm.org/10.1145/2342356.2342438.

F. Wuhib, M. Dam, R. Stadler, and A. Clemm. Robust monitoring of network-wide aggregates through gossiping. *IEEE Transactions on Network and Service Management*, 6(2):95–109, 2009.

Di Xie, Ning Ding, Y. Charlie Hu, and Ramana Kompella. The only constant is change: Incorporating time-varying network reservations in data centers. In *ACM SIGCOMM*, 2012.

Kuai Xu and Feng Wang. Cooperative monitoring for internet data centers. In *Performance, Computing and Communications Conference, 2008. IPCCC 2008. IEEE International*, pages 111–118, dec. 2008. doi: 10.1109/PCCC.2008.4745129.

Minlan Yu, Jennifer Rexford, Michael J. Freedman, and Jia Wang. Scalable flow-based networking with difane. In *Proceedings of the ACM SIGCOMM 2010 conference*, SIGCOMM '10, pages 351–362, New York, NY, USA, 2010. ACM. ISBN 978-1-4503-0201-2. doi: 10.1145/1851182.1851224.

References XIV

Hang Yuan, C.-C. Jay Kuo, and Ishfaq Ahmad. Energy efficiency in data centers and cloud-based multimedia services: An overview and future directions. In *Proceedings of the International Conference on Green Computing*, GREENCOMP '10, pages 375–382, Washington, DC, USA, 2010. IEEE Computer Society. ISBN 978-1-4244-7612-1. doi: 10.1109/GREENCOMP.2010.5598292. URL http://dx.doi.org/10.1109/GREENCOMP.2010.5598292.

F.E. Zaheer, Jin Xiao, and R. Boutaba. Multi-provider service negotiation and contracting in network virtualization. In *Network Operations and Management Symposium (NOMS), 2010 IEEE*, pages 471–478, april 2010. doi: 10.1109/NOMS.2010.5488487.

Kehuan Zhang, Xiaoyong Zhou, Yangyi Chen, XiaoFeng Wang, and Yaoping Ruan. Sedic: privacy-aware data intensive computing on hybrid clouds. In *Proceedings of the 18th ACM conference on Computer and communications security*, CCS '11, pages 515–526, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0948-6.

Qi Zhang, Lu Cheng, and Raouf Boutaba. Cloud computing: state-of-the-art and research challenges. *Journal of Internet Services and Applications*, 1(1):7–18, May 2010. doi: 10.1007/s13174-010-0007-6.

Yan Zhang and Nirwan Ansari. On architecture design, congestion notification, tcp incast and power consumption in data centers. *Communications Surveys Tutorials, IEEE*, 15(1):39–64, 2013. ISSN 1553-877X. doi: 10.1109/SURV.2011.122211.00017.

Mohamed Faten Zhani, Qi Zhang, Gwendal Simon†, and Raouf Boutaba. VDC Planner: Dynamic Migration-Aware Virtual Data Center Embedding for Clouds. In *2011 IFIP/IEEE International Symposium on Integrated Network Management (IM)*, IM'13, Washington, DC, USA, 2013. IEEE Computer Society.

Xia Zhou, Zengbin Zhang, Yibo Zhu, Yubo Li, Saipriya Kumar, Amin Vahdat, Ben Y. Zhao, and Haitao Zheng. Mirror mirror on the ceiling: flexible wireless links for data centers. *SIGCOMM Comput. Commun. Rev.*, 42(4):443–454, August 2012. ISSN 0146-4833. doi: 10.1145/2377677.2377761. URL http://doi.acm.org/10.1145/2377677.2377761.