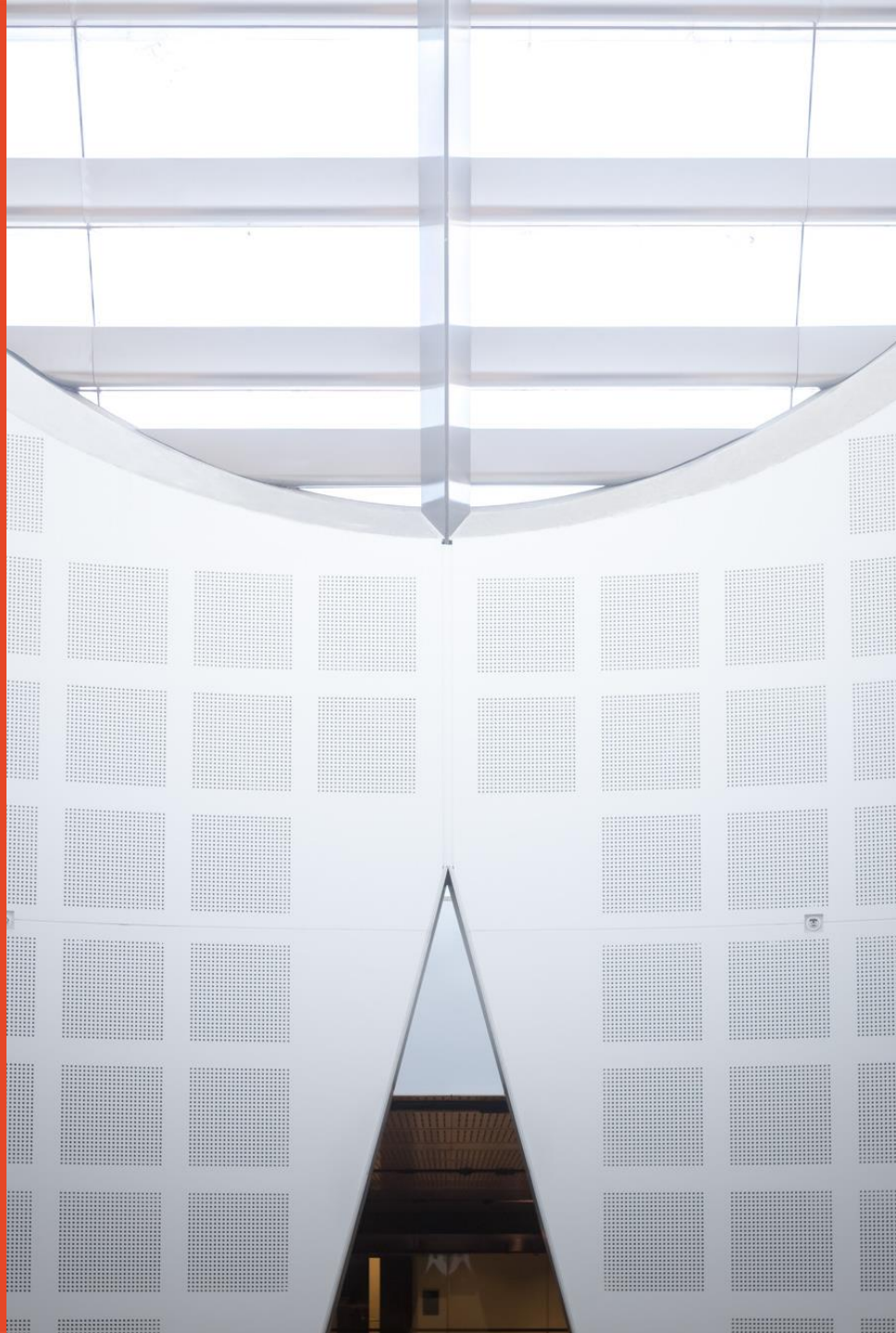
Topic 9: Network Virtualisation & NFV

Presented by
Dong YUAN

School of Electrical and Computer Engineering

dong.yuan@sydney.edu.au

THE UNIVERSITY OF SYDNEY



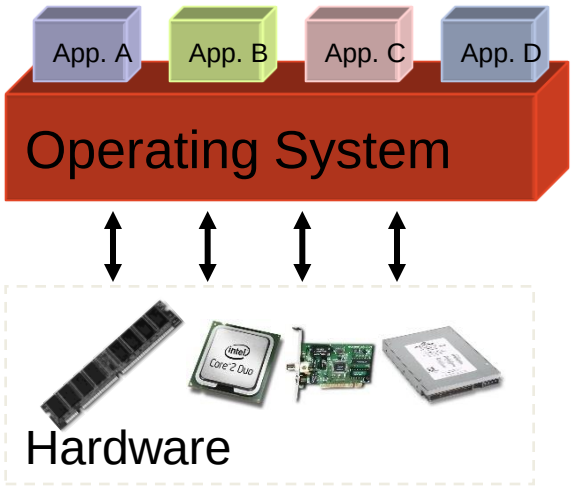
Contents

- Virtualisation Technologies
- Network Virtualisation
- Network Function Virtualisation and SDN

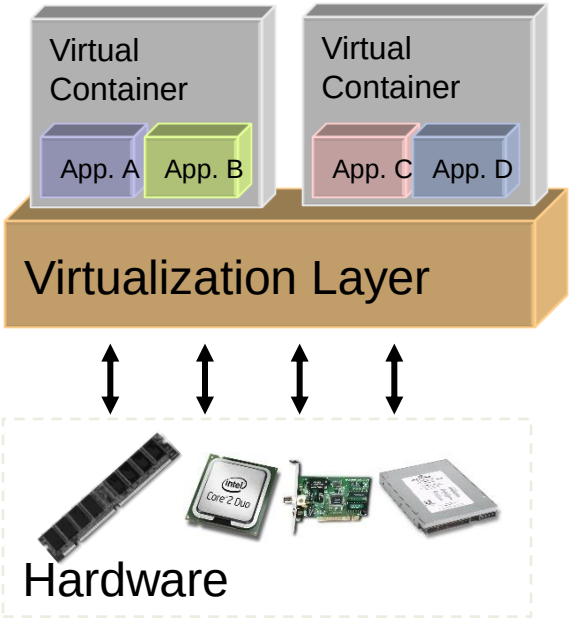
What is virtualization?

Virtualization is a broad term (virtual memory, storage, network, etc)

Virtualization basically allows one computer to do the job of multiple computers, by sharing the resources of a single hardware across multiple environments

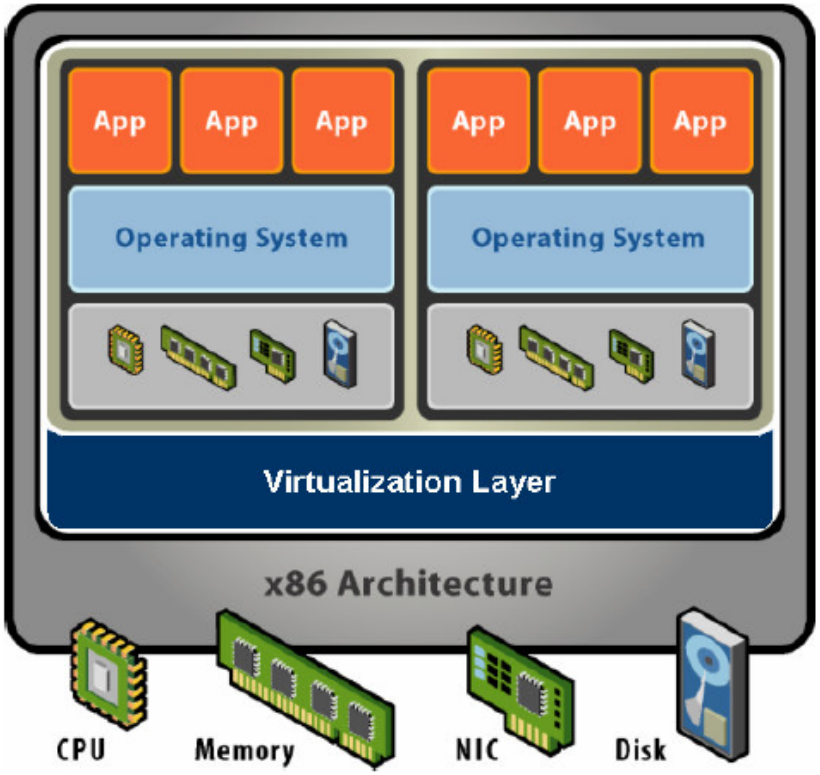


'Non-virtualized' system
A single OS controls all hardware platform resources



Virtualized system
It makes it possible to run multiple Virtual Containers on a single physical platform

What is a Virtual Machine?



Hardware-Level Abstraction

- Virtual hardware: processors, memory, chipset, I/O devices, etc.
- Encapsulates all OS and application state

Virtualization Software

- Extra level of indirection decouples hardware and OS
- Multiplexes physical hardware across multiple "guest" VMs
- Strong isolation between VMs
- Manages physical resources, improves utilization

How did it start?

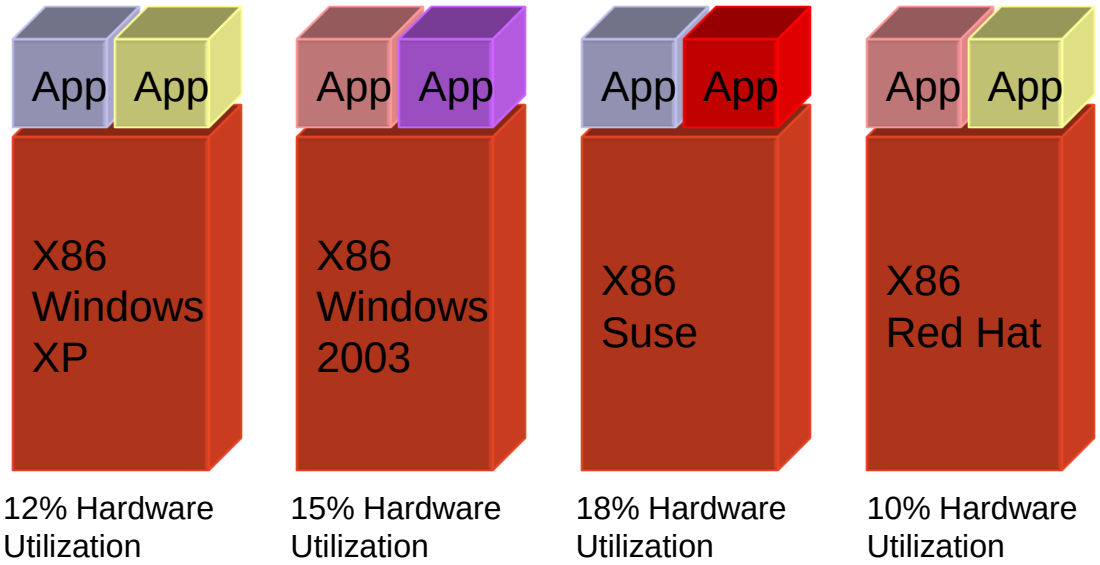
- Server virtualization has existed for several decades
  - IBM pioneered more than 40 years ago with the capability to “multitask”

- The inception was in specialized, proprietary, high-end server and mainframe systems

- By 1980/90 servers virtualization adoption initiated a reduction
  - Inexpensive x86 hardware platforms
  - Windows/Linux adopted as server OSs

Before 2000

- 1 machine → 1 OS → several applications
- Applications can affect each other
- Big disadvantage: machine utilization is very low, most of the times it is below 25%



Virtualization again…

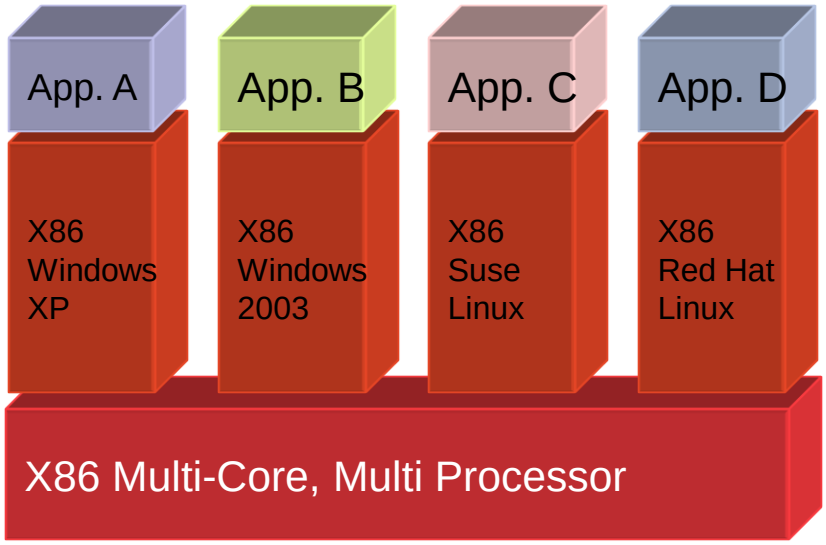
x86 server deployments introduced new IT challenges:

- Low server infrastructure utilization (10-18%)
- Increasing physical infrastructure costs (facilities, power, cooling, etc)
- Increasing IT management costs (configuration, deployment, updates, etc)
- Insufficient failover and disaster protection

The solution for all these problems was to virtualize x86 platforms

Computing Infrastructure - Virtualization

- It matches the benefits of high hardware utilization with running several operating systems (applications) in separated virtualized environments
  - Each application runs in its own operating system
  - Each operating system does not know it is sharing the underlying hardware with others

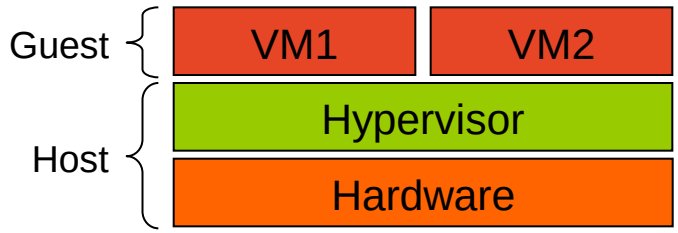


70% Hardware Utilization

Two types of hypervisors

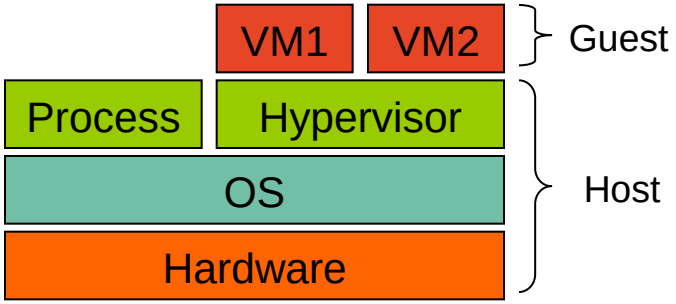
- Definitions
  - **Hypervisor** (or **VMM** – Virtual Machine Monitor) is a software layer that allows several **virtual machines** to **run** on a **physical machine**
  - The physical OS and hardware are called the **Host**
  - The virtual machine OS and applications are called the **Guest**

Type 1 (bare-metal)



VMware ESX, Microsoft Hyper-V, Xen

Type 2 (hosted)



VMware Workstation, Microsoft Virtual PC, Sun VirtualBox, QEMU, KVM

Bare-metal or hosted?

- **Bare-metal**
  - Has complete **control over hardware**
  - Doesn't have to **"fight" an OS**
- **Hosted**
  - Avoid **code duplication**: need not code a **process scheduler, memory management** system – the **OS already does** that
  - Can run native **processes alongside** VMs
  - Familiar environment – **how much CPU** and **memory** does a VM take? How big is the **virtual disk**?
  - Easy management – stop a VM? Sure, just kill it!
- A combination
  - Mostly hosted, but some parts are inside the OS kernel for performance reasons
  - E.g., **KVM**

I/O Virtualization

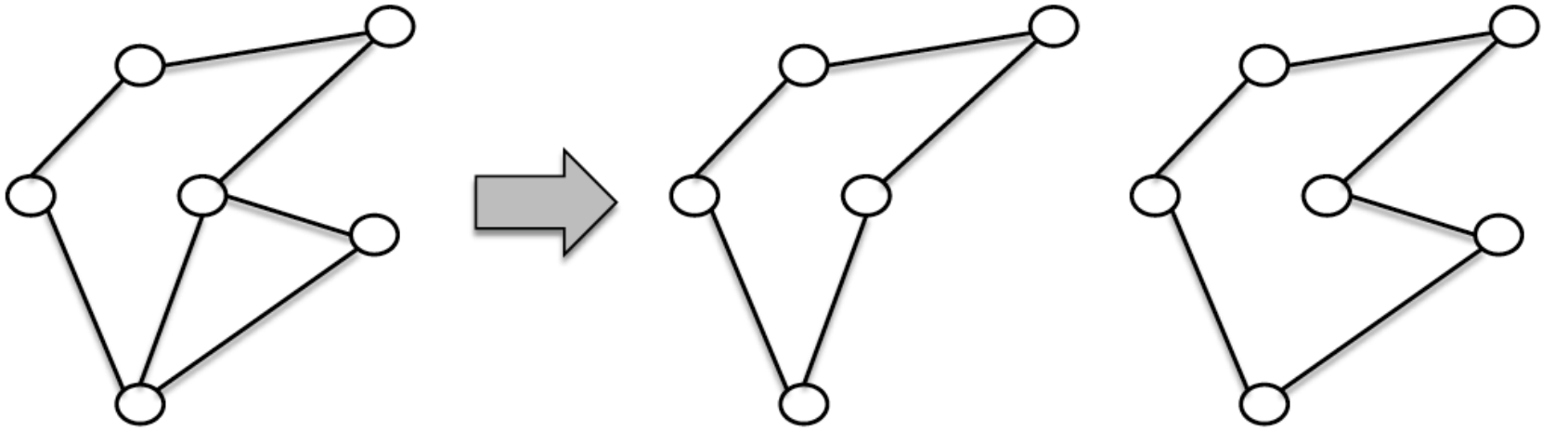
- We saw **methods** to **virtualize** the **CPU**
  - A computer is more than a CPU
  - Also need I/O!
- Types of I/O:
  - Block (e.g., hard disk), Network – most performance critical
  - Input (e.g., keyboard, mouse), Sound, Video
- Hypervisor implements virtual NIC (by the specification of a real NIC, e.g., Intel, Realtek, Broadcom)
  - Pro: Unmodified guest (guest already has drivers for Intel NICs…)
  - Cons: Slow – every access to every NIC register causes a VM exit (trap to hypervisor), and Hypervisor needs to emulate complex hardware

Contents

- Virtualisation Technologies
- Network Virtualisation
- Network Function Virtualisation and SDN

Network Virtualisation

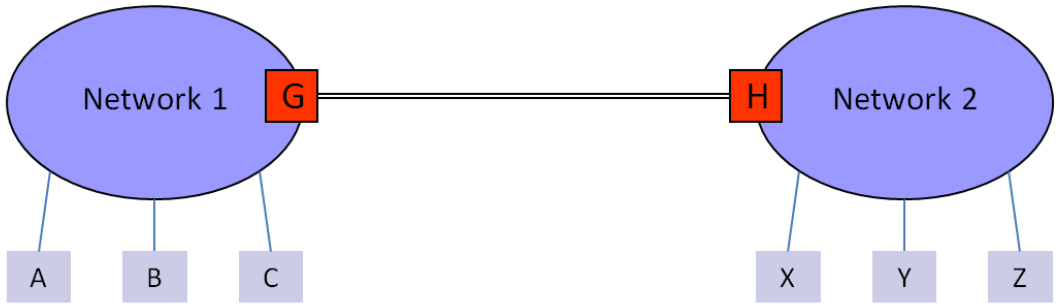
– Making a physical network appear as multiple logical ones



Why network virtualisation?

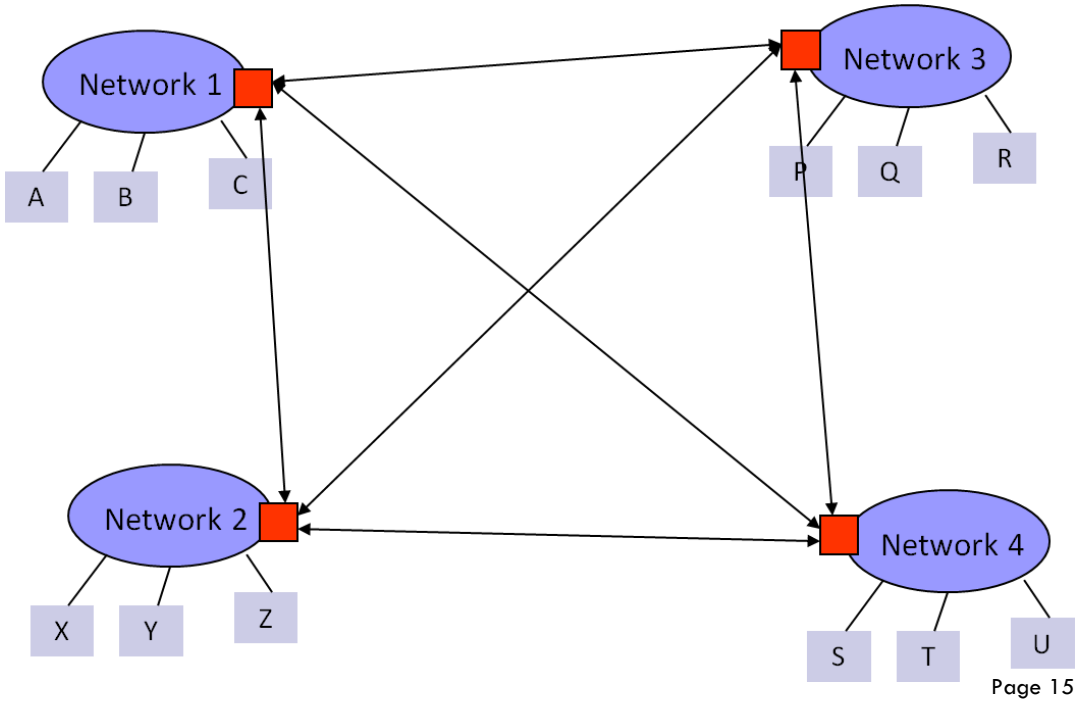
- Internet is almost ossified
  - Lots of band-aids and makeshift solutions (e.g. overlays)
  - A new architecture (aka clean-slate) is needed
- Hard to come up with a one-size-fits-all architecture
  - Almost impossible to predict what future might unleash
- Why not create an all-sizes-fit-into-one instead!
  - Open and expandable architecture
- Testbed for future networking architectures and protocols

Network Architecture

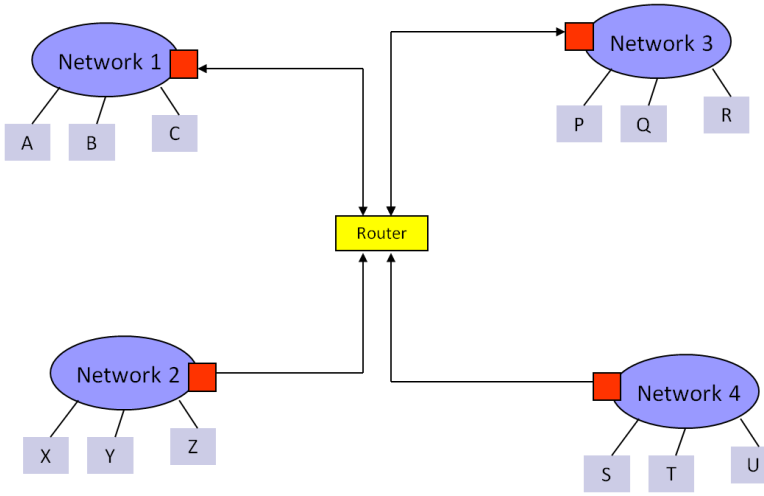


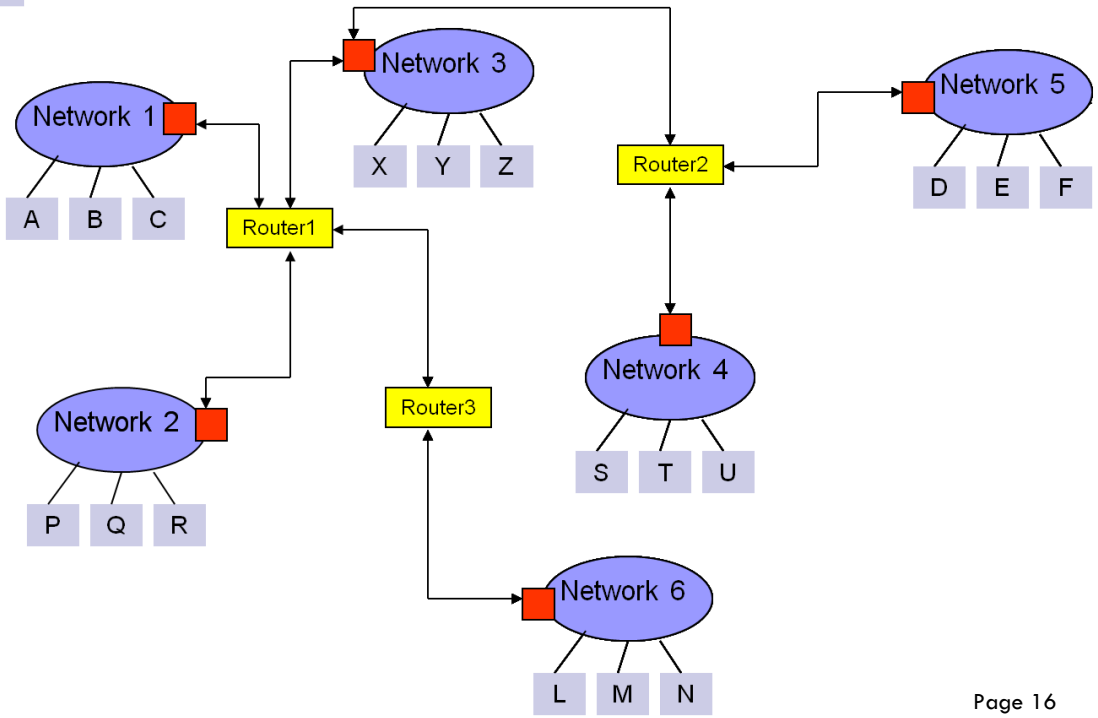
G
H

Gateway computers



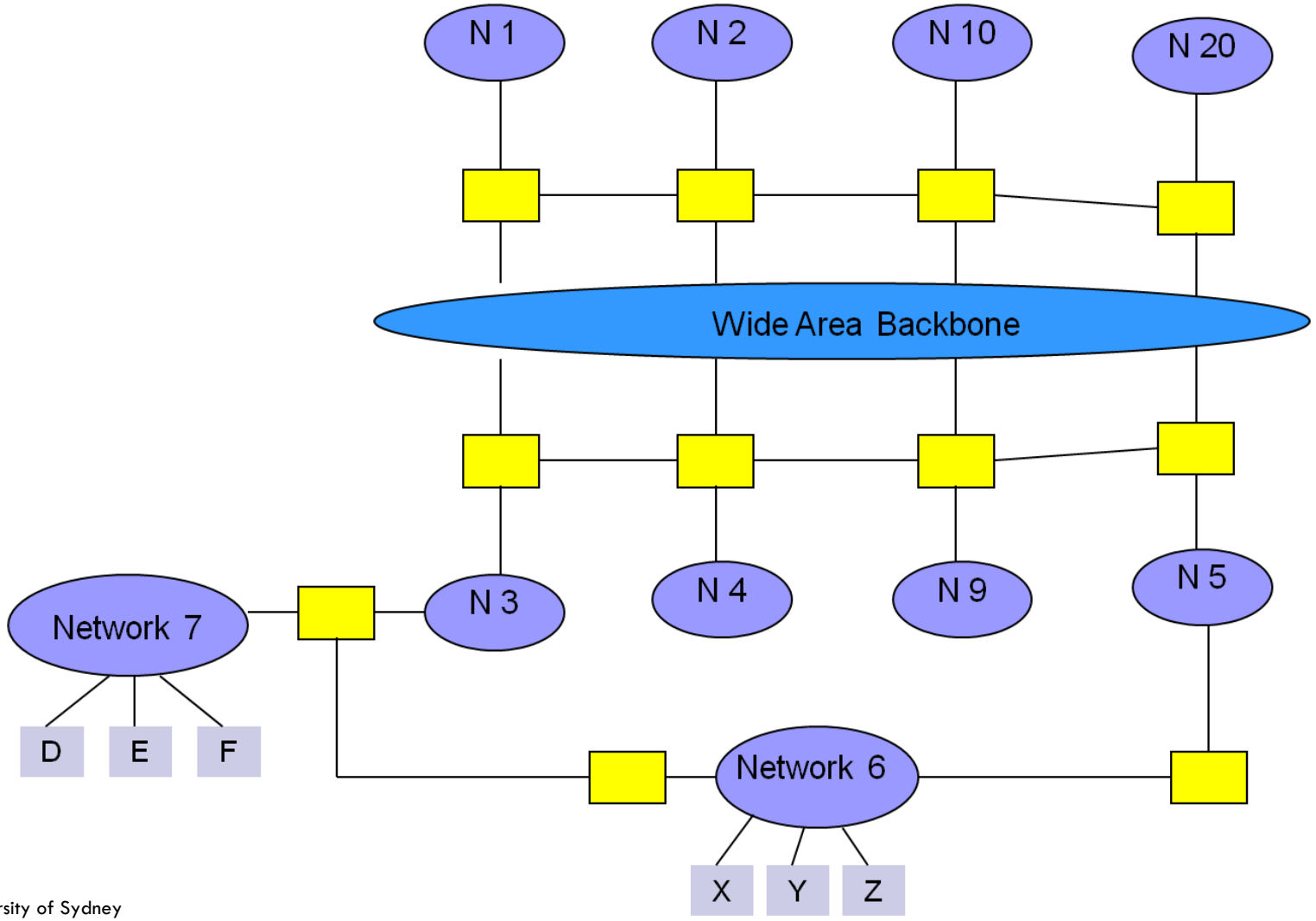
Network Architecture





Network Architecture

The simple view of Internet



What is network virtualization?

Network Virtualization is the process of combining hardware and software network resources and network functionality into a single, software-based administrative entity, a virtual network. --- In computing

- Two categories :
  - External network virtualization
    - Combining many networks, or parts of networks, into a virtual unit.
  - Internal network virtualization
    - Providing network-like functionality to the software containers on a single system.
    - This was before SDN and NFV

Desirable properties of network virtualization

- Scalability
  - Easy to extend resources in need
  - Administrator can dynamically create or delete virtual network connection
- Resilience
  - Recover from the failures
  - Virtual network will automatically redirect packets by redundant links
- Security
  - Increased path isolation and user segmentation
  - Virtual network should work with firewall software
- Availability
  - Access network resource anytime

External Network Virtualization

- Layer 1
  - Seldom virtualization implement in this physical data transmission layer.
- Layer 2
  - Use some tags in MAC address packet to provide virtualization.
  - Example, VLAN.
- Layer 3
  - Use some tunnel techniques to form a virtual network.
  - GRE (Generic Routing Encapsulation) by CISCO
  - Example, VPN.
- Layer 4 or higher
  - Build up some overlay network for some application.
  - Example, P2P.

Internal Network Virtualization

- Layer 1
  - Hypervisor usually do not need to emulate the physical layer.
- Layer 2
  - Implement virtual L2 network devices, such as switch, in hypervisor.
  - Example, Linux TAP driver + Linux bridge.
- Layer 3
  - Implement virtual L3 network devices, such as router, in hypervisor.
  - Example, Linux TUN driver + Linux bridge + iptables.
- Layer 4 or higher
  - Layer 4 or higher layers virtualization is usually implemented in guest OS.
  - Applications should make their own choice.

TUN/TAP driver

- TUN and TAP are virtual network kernel drivers :
  - TAP (as in network tap) simulates an Ethernet device and it operates with layer 2 packets such as Ethernet frames.
  - TUN (as in network TUNnel) simulates a network layer device and it operates with layer 3 packets such as IP.

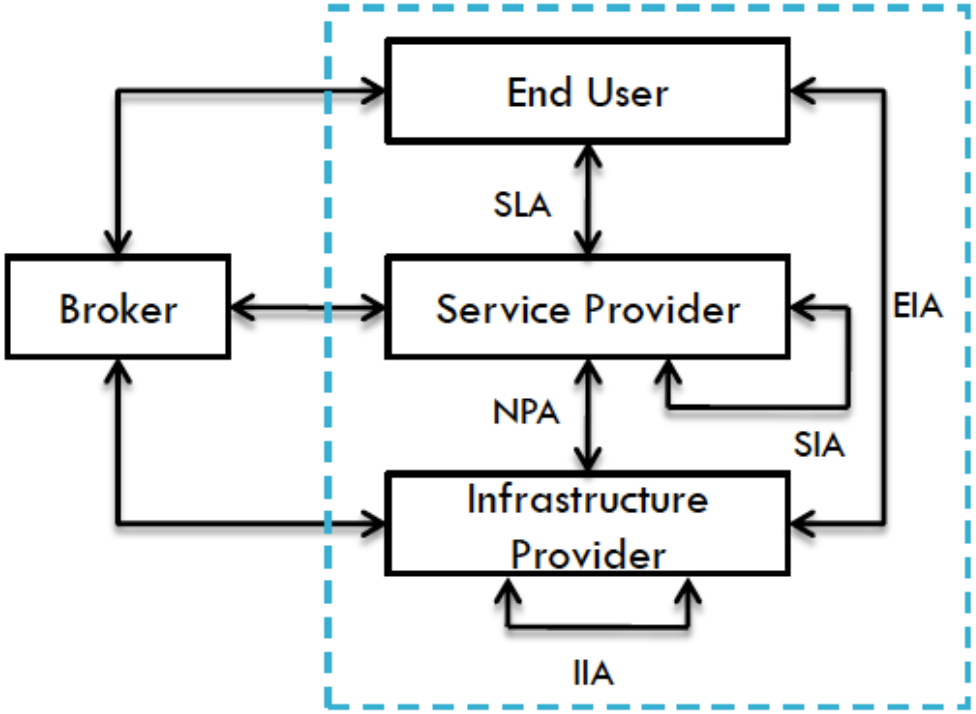
- Data flow of TUN/TAP driver
  - Packets sent by an operating system via a TUN/TAP device are delivered to a user-space program that attaches itself to the device.
  - A user-space program may pass packets into a TUN/TAP device. TUN/TAP device delivers (or "injects") these packets to the operating system network stack thus emulating their reception from an external source.

KVM system

- KVM focus on CPU and memory virtualization, so IO virtualization framework is completed by QEMU project.
  - In QEMU, network interface of virtual machines connect to host by TUN/TAP driver and Linux bridge. https://www.qemu.org/
- Work with TUN/TAP and Linux Bridge :
  - Virtual machines connect to host by a virtual network adapter, which is implemented by TUN/TAP driver.
  - Virtual adapters will connect to Linux bridges, which play the role of virtual switch.

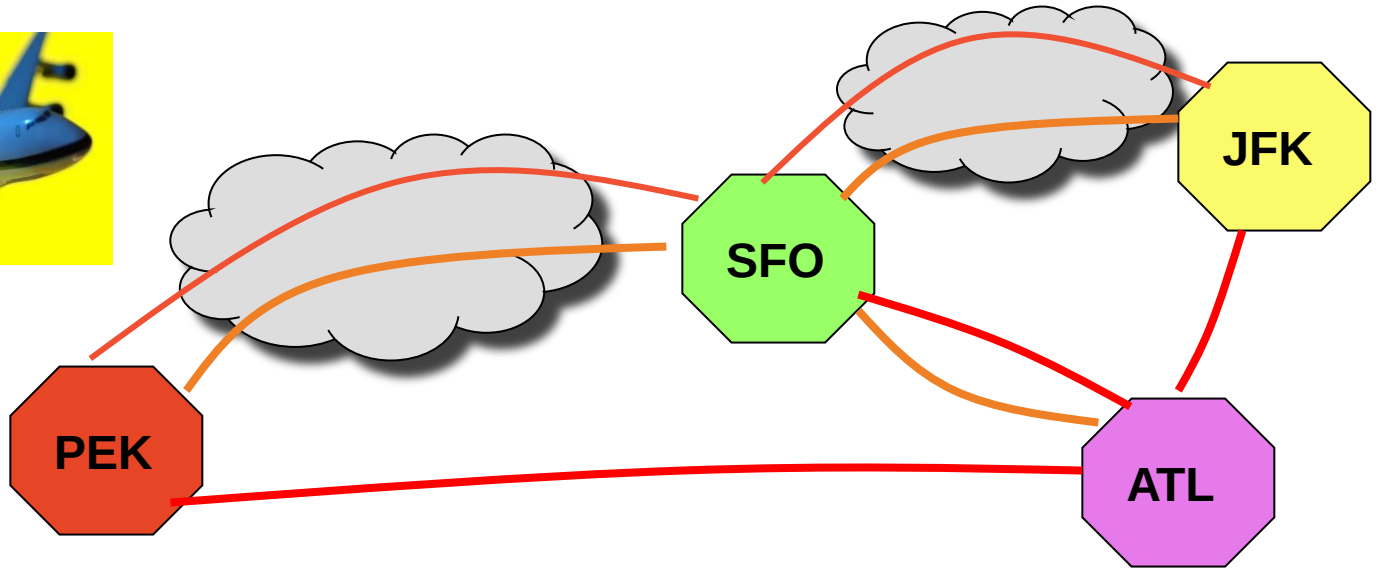
Business Model

- Infrastructure Providers (InPs)
  - Manage underlying physical networks
- Service Providers (SPs)
  - Create and manage virtual networks
  - Deploy customized end-to-end services
- End Users
  - Buy and use services from different service providers
- Brokers
  - Mediators/Arbiters



Similar Trends in Other Industries

- Commercial aviation
  - Infrastructure providers: Airports
  - Infrastructure: Gates, Terminals, other supports
  - Service providers: Airlines

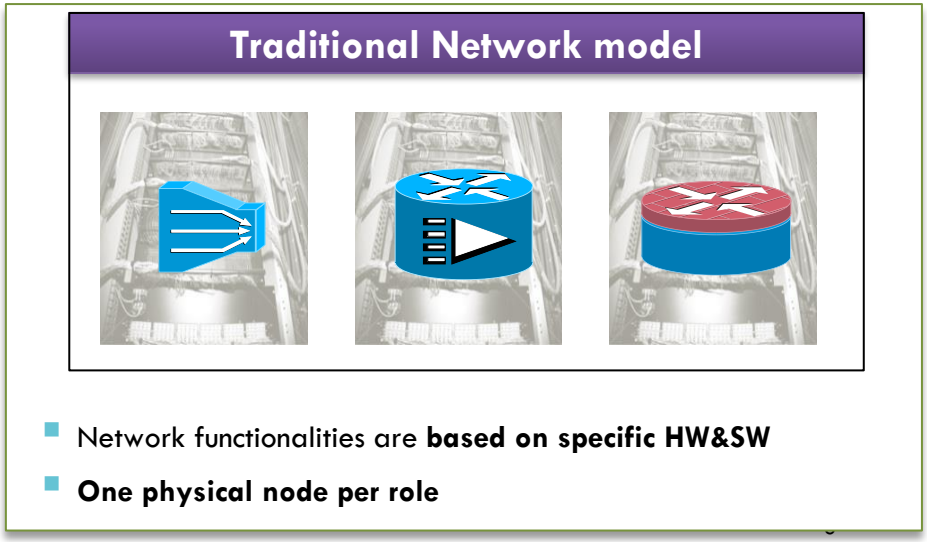


Contents

- Virtualisation Technologies
- Network Virtualisation
- Network Function Virtualisation and SDN

Motivation Problem Statement

- Complex carrier networks
  - with a large variety of proprietary nodes and hardware appliances.
- Launching new services is difficult and takes too long
  - Space and power to accommodate
  - requires just another variety of box, which needs to be integrated.
- Operation is expensive
  - due to existing procure-design, integrate and deploy cycle.



Middlebox

- A middlebox or network appliance is a computer networking device that transforms, inspects, filters, or otherwise manipulates traffic for purposes other than packet forwarding.
- Network functions: firewall, Intrusion detection system, DPI, VPN, gateways, WAN optimiser, etc.

- SDN is more about the forwarding parts of data plane, and promises a centralised management.
  - Can we also build a unified framework for functions of middleboxes?

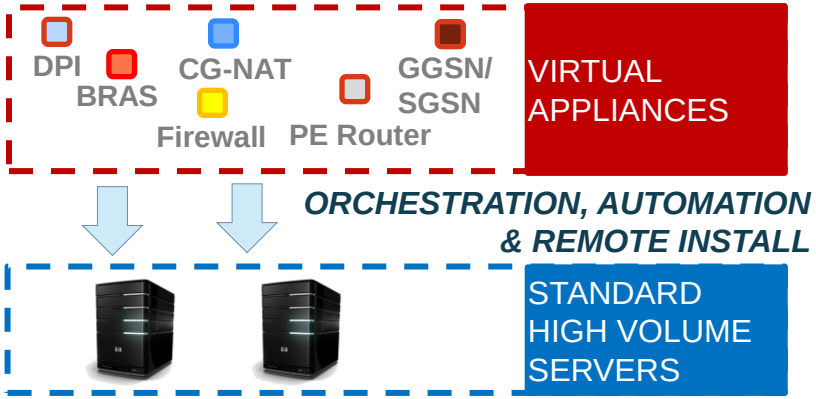
The NFV Concept

A means to make the network more flexible and simple by minimising dependence on HW constraints

Traditional Network Model: APPLIANCE APPROACH

- Network Functions are **based on specific HW&SW**
- **One physical node per role**

Virtualised Network Model: VIRTUAL APPLIANCE APPROACH



- Network Functions are **SW-based over well-known HW**
- **Multiple roles** over **same HW**

Source: Adapted from D. Lopez Telefonica I+D, NFV

Network Functions Virtualization

- Network Functions Virtualization is about implementing network functions in software - that today run on proprietary hardware - leveraging (high volume) standard servers and IT virtualization
- Supports multi-versioning and multi-tenancy of network functions, which allows use of a single physical platform for different applications, users and tenants
- Enables new ways to implement resilience, service assurance, test and diagnostics and security surveillance
- Provides opportunities for pure software players

Source: Adapted from D. Lopez Telefonica I+D, NFV

Network Functions Virtualization

- Facilitates innovation towards new network functions and services that are only practical in a pure software network environment
- Applicable to any data plane packet processing and control plane functions, in fixed or mobile networks
- NFV will only scale if management and configuration of functions can be automated
- NFV aims to ultimately transform the way network operators architect and operate their networks, but change can be incremental

Benefits & Promises of NFV

- Reduced equipment costs (CAPEX)
  - through consolidating equipment and economies of scale of IT industry.
- Increased speed of time to market
  - by minimising the typical network operator cycle of innovation.
- Availability of network appliance multi-version and multi-tenancy
  - allows a single platform for different applications, users and tenants.
- Enables a variety of eco-systems and encourages openness.
- Encouraging innovation to bring new services and generate new revenue streams.

Source: NFV

Benefits & Promises of NFV

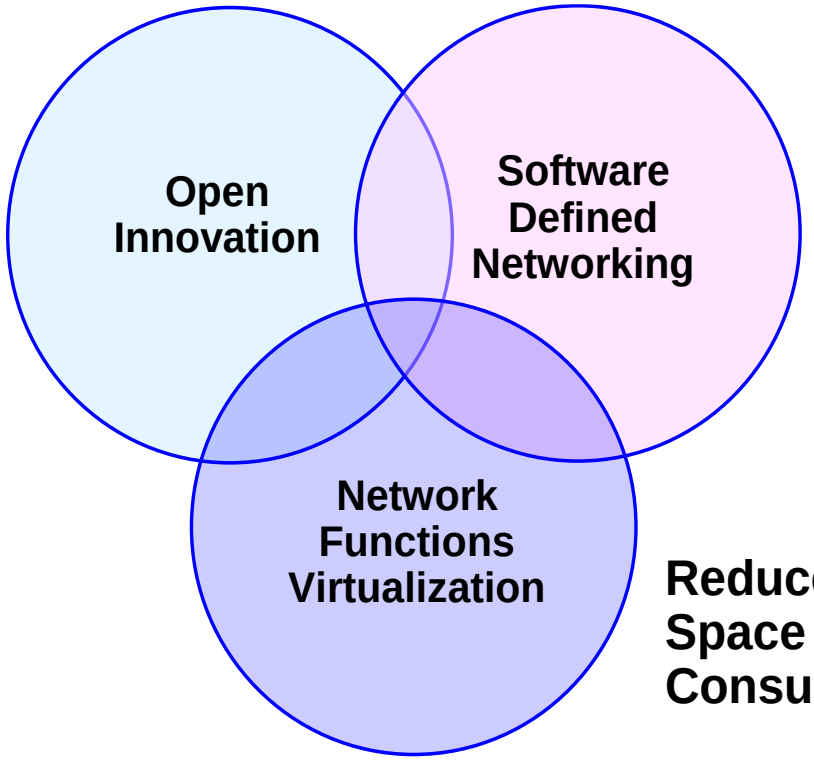
- Flexibility to easily, rapidly, dynamically provision and instantiate new services in various locations
- Improved operational efficiency
  - by taking advantage of the higher uniformity of the physical network platform and its homogeneity to other support platforms.
- Software-oriented innovation to rapidly prototype and test new services and generate new revenue streams
- More service differentiation & customization
- Reduced (OPEX) operational costs: reduced power, reduced space, improved network monitoring
- IT-oriented skillset and talent

Source: Adapted from D. Lopez Telefonica I+D, NFV

NFV and SDN

- NFV and SDN are highly complementary
- Both topics are mutually beneficial but not dependent on each other

Creates competitive supply of innovative applications by third parties



Creates network abstractions to enable faster innovation

Reduces CAPEX, OPEX, Space & Power Consumption

Source: NFV

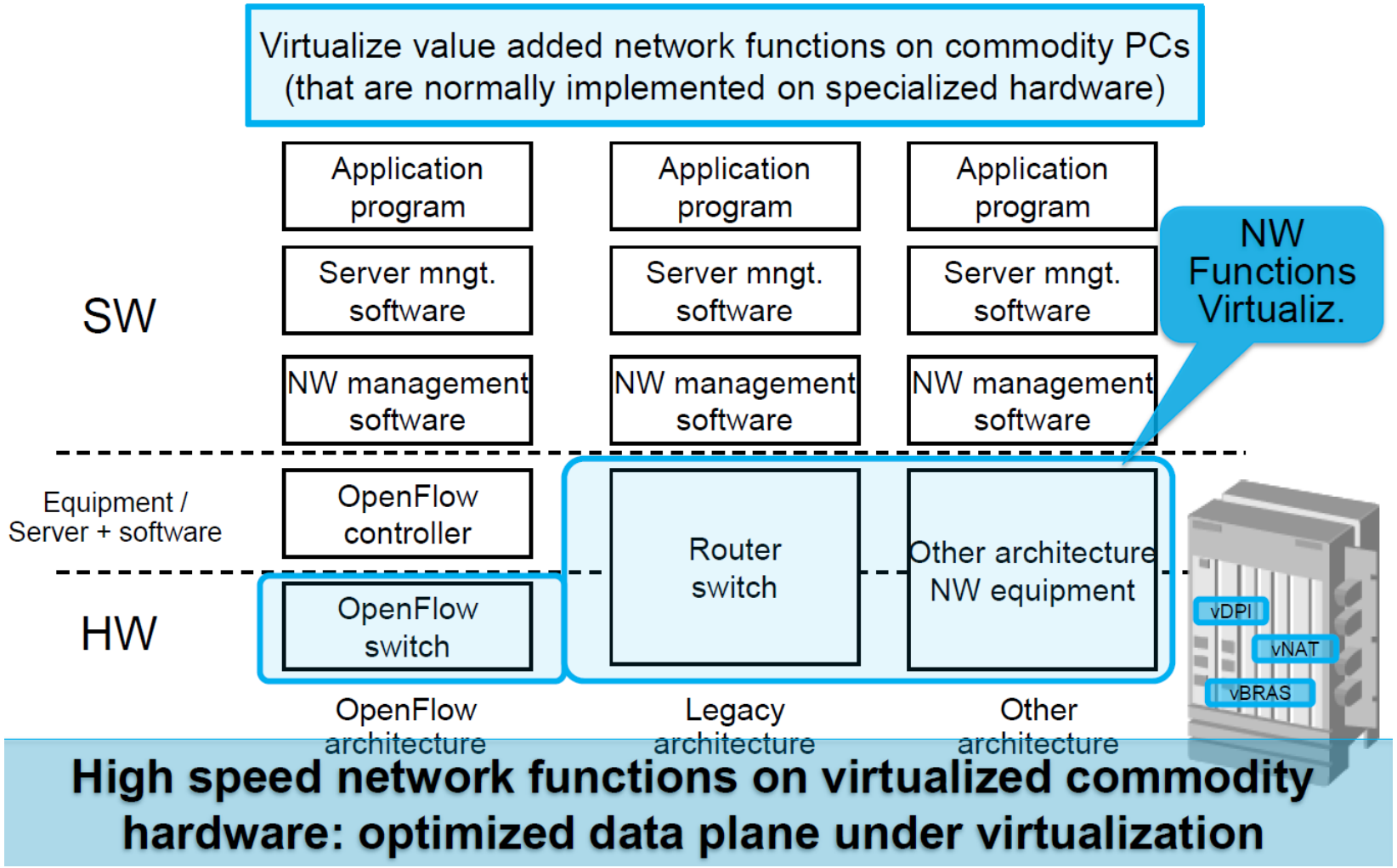
NFV vs SDN

- NFV: re-definition of **network equipment architecture**
- NFV was born to meet Service Provider (SP) needs:
  - Lower CAPEX by reducing/eliminating proprietary hardware
  - Consolidate multiple network functions onto industry standard platforms
- SDN: re-definition of **network architecture**
- SDN comes from the IT world:
  - Separate the data and control layers, while centralizing the control
  - Deliver the ability to program network behavior using well-defined interfaces

NFV vs SDN

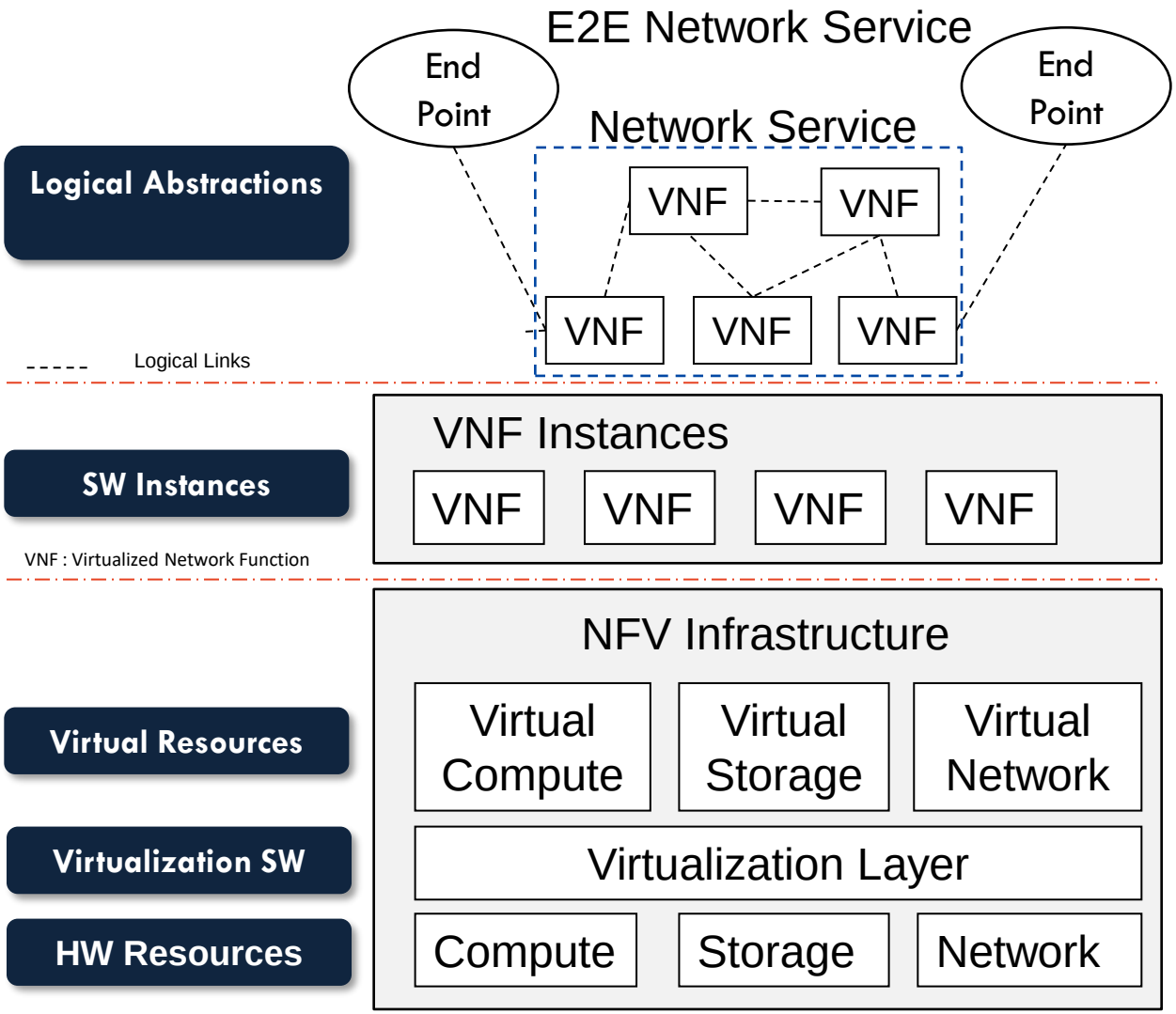
- NFV and SDN are complementary
  - One does not depend upon the other.
  - You can do SDN only, NFV only, or SDN and NFV.
- Both have similar goals but approaches are very different.
- SDN needs new interfaces, control modules, applications.
- NFV requires moving network applications from dedicated hardware to virtual containers on commercial-off-the-shelf (COTS) hardware
- NFV is present. SDN is the future.
- Virtualization alone provides many of the required features
- Not much debate about NFV.

Scope of NFV and OpenFlow/SDN



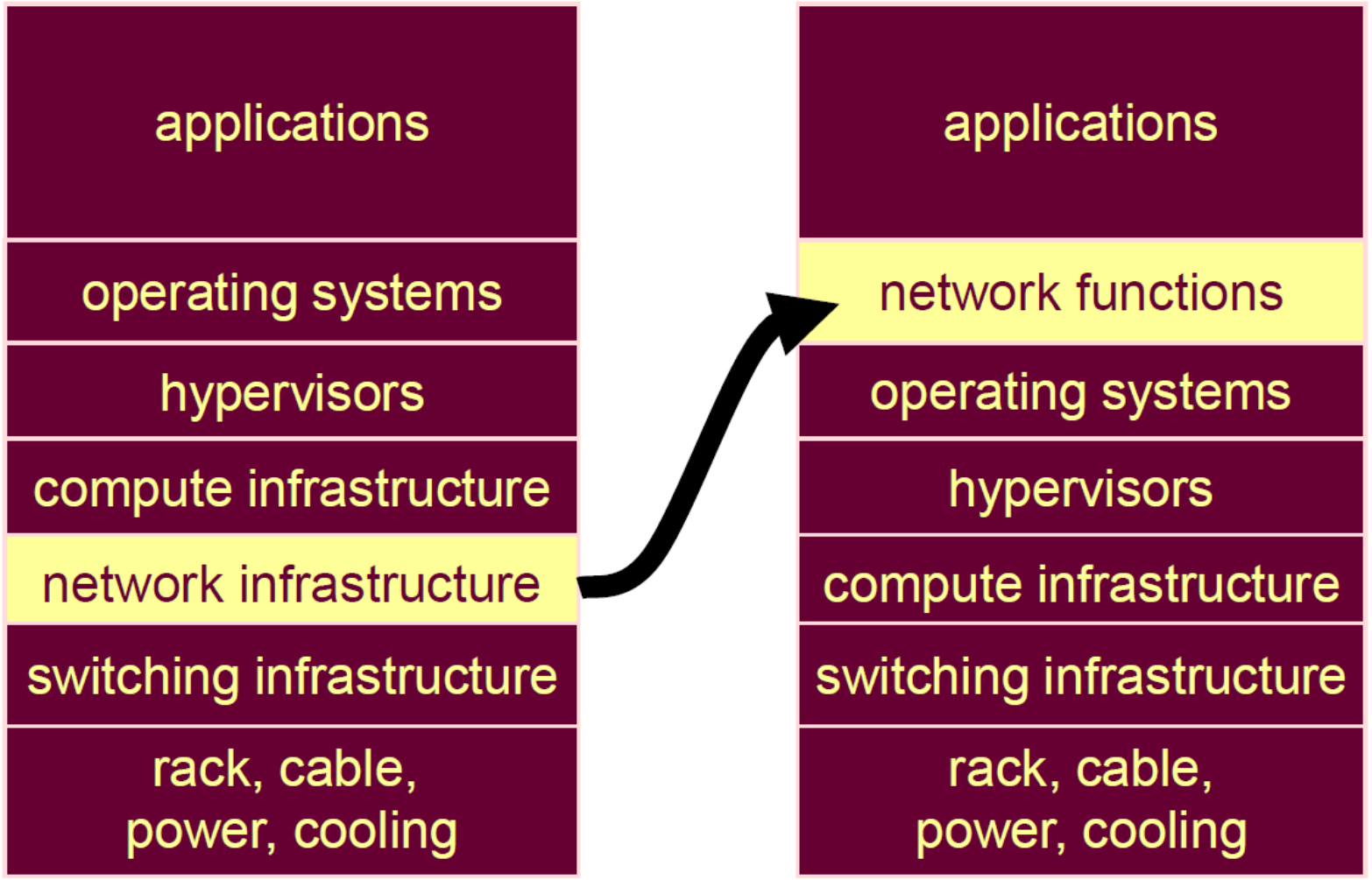
Source: NEC

NFV Layers



Source: Adapted from D. Lopez Telefonica I+D, NFV

Rethinking relayering



Use Case: Network as a Service (NaaS)

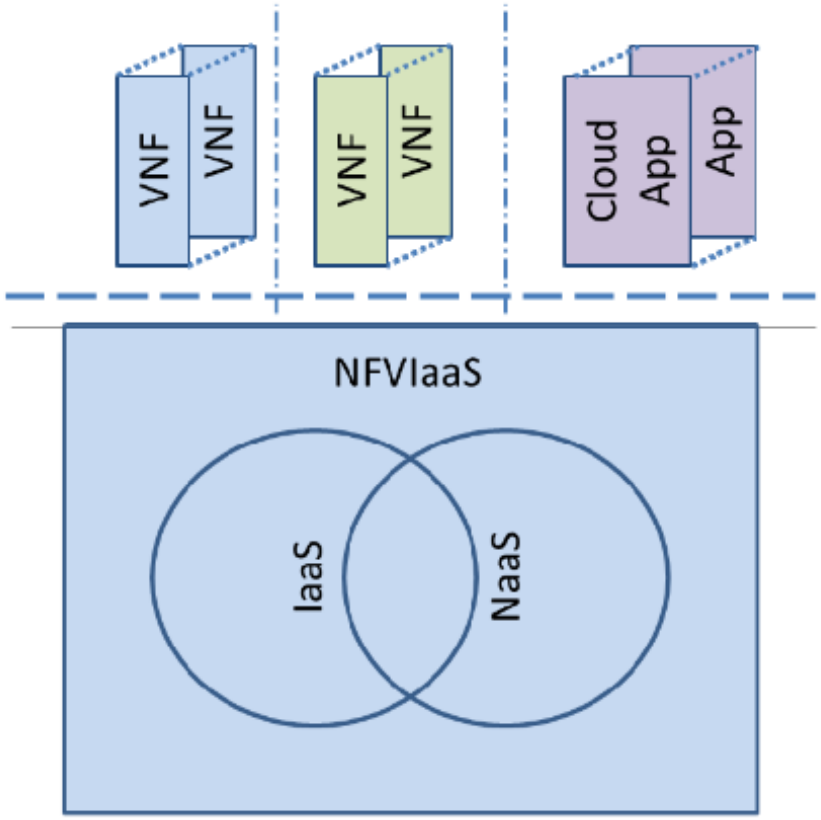
- NaaS is defined as a **service delivery model,** analogous to IaaS, for providing dynamic, scalable, secure, and isolated network access for multiple tenants
- Related to:
  - Network infrastructure provisioning
  - Enabling dynamic and scalable network services (NS)
  - Enabling multiple tenants to access NS
  - Keeping NS secure and isolated
- NaaS sets the basis for business models related to network infrastructure servicing.

NFV Infrastructure as a Service (NFVIaaS)

NFV Infrastructure :

- provide the capability or functionality of providing an environment in which Virtualized network functions (VNF) can execute

- **NFVIaaS** provides compute capabilities comparable to an **IaaS cloud computing service** as a run time execution environment **as well as support the dynamic network connectivity services** that may be considered as comparable to **NaaS**



A Few Challenges

- Achieving high performance virtualised network appliances
  - portable between different HW vendors, and with different hypervisors.
- Co-existence with bespoke HW based network platforms
  - enabling efficient migration paths to fully virtualised network platforms.
- Management and orchestration of virtual network appliances
  - ensuring security from attack and misconfiguration.
- NFV will only scale if all of the functions can be automated.
- Security

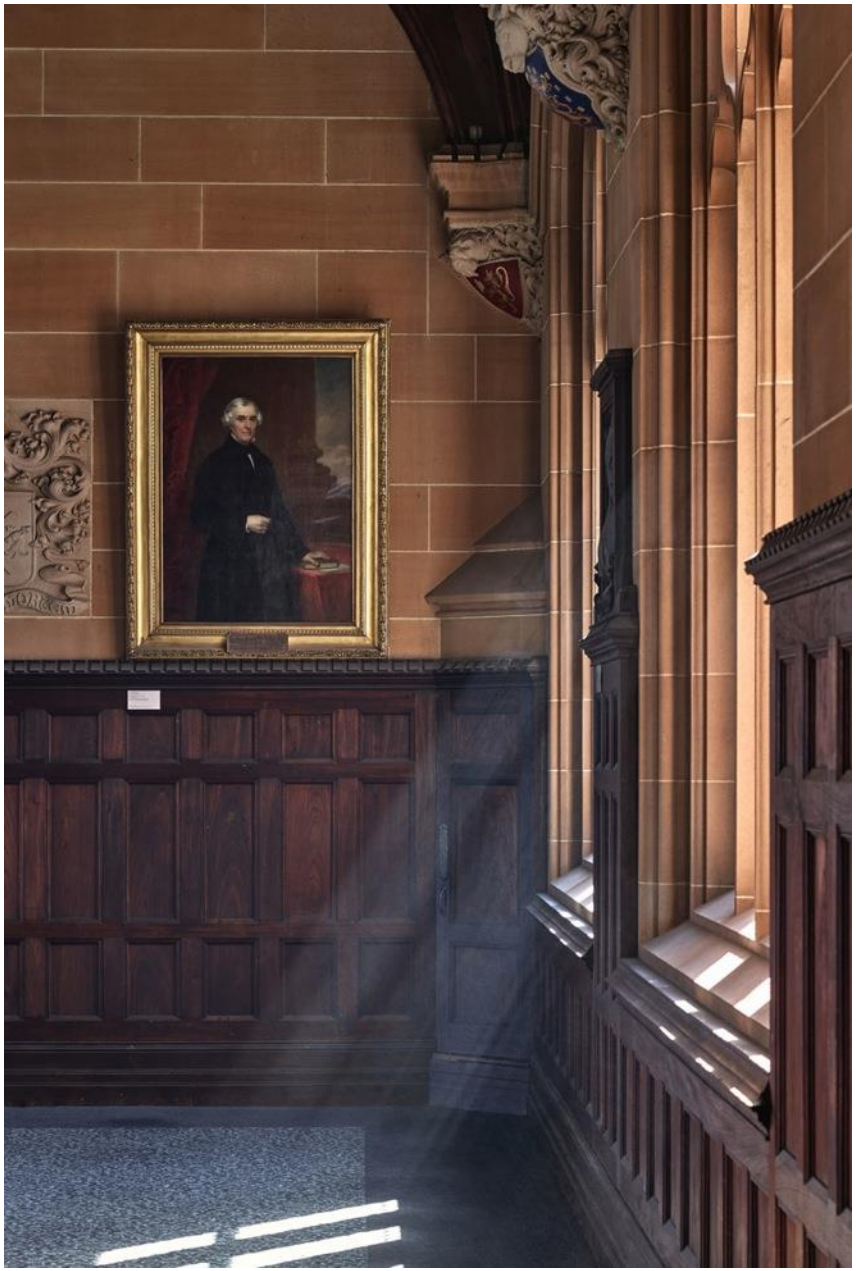
Thank you!

References:

https://www.scs.gatech.edu/news/195201/free-online-sdn-course

https://www.sdxcentral.com/sdn/?c_action=num_ball

https://www.opennetworking.org/



THE UNIVERSITY OF SYDNEY