Topic 10: CORD and Datacenter Network

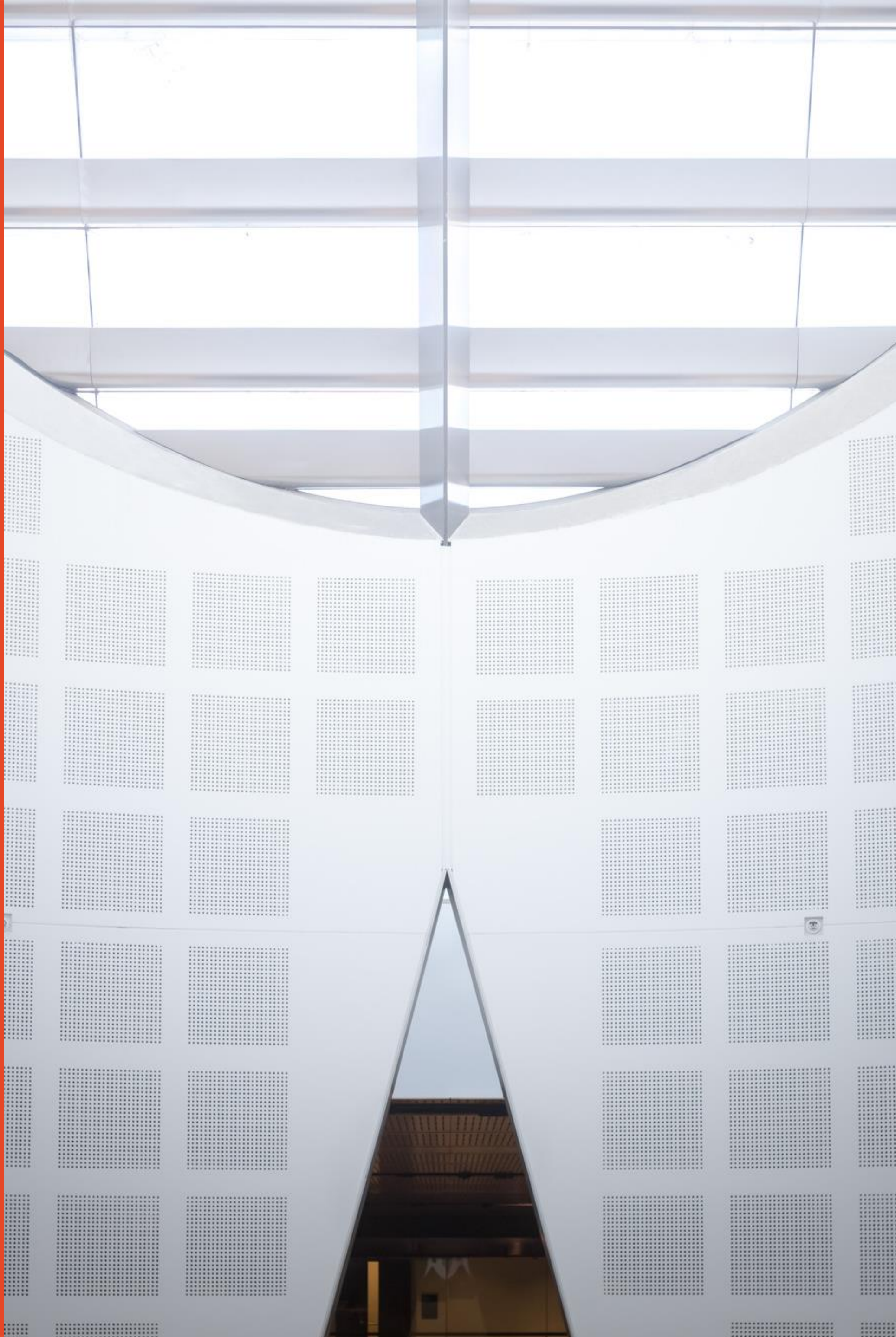
Presented by

Dong YUAN

School of Electrical and Computer Engineering

dong.yuan@sydney.edu.au

THE UNIVERSITY OF SYDNEY



Contents

– NFV and CORD

– Software Defined Data Centers (SDDC)
  – Modern data centers
  – Data center network topology
  – Network slicing
  – Multi-tenant Data center

NFV

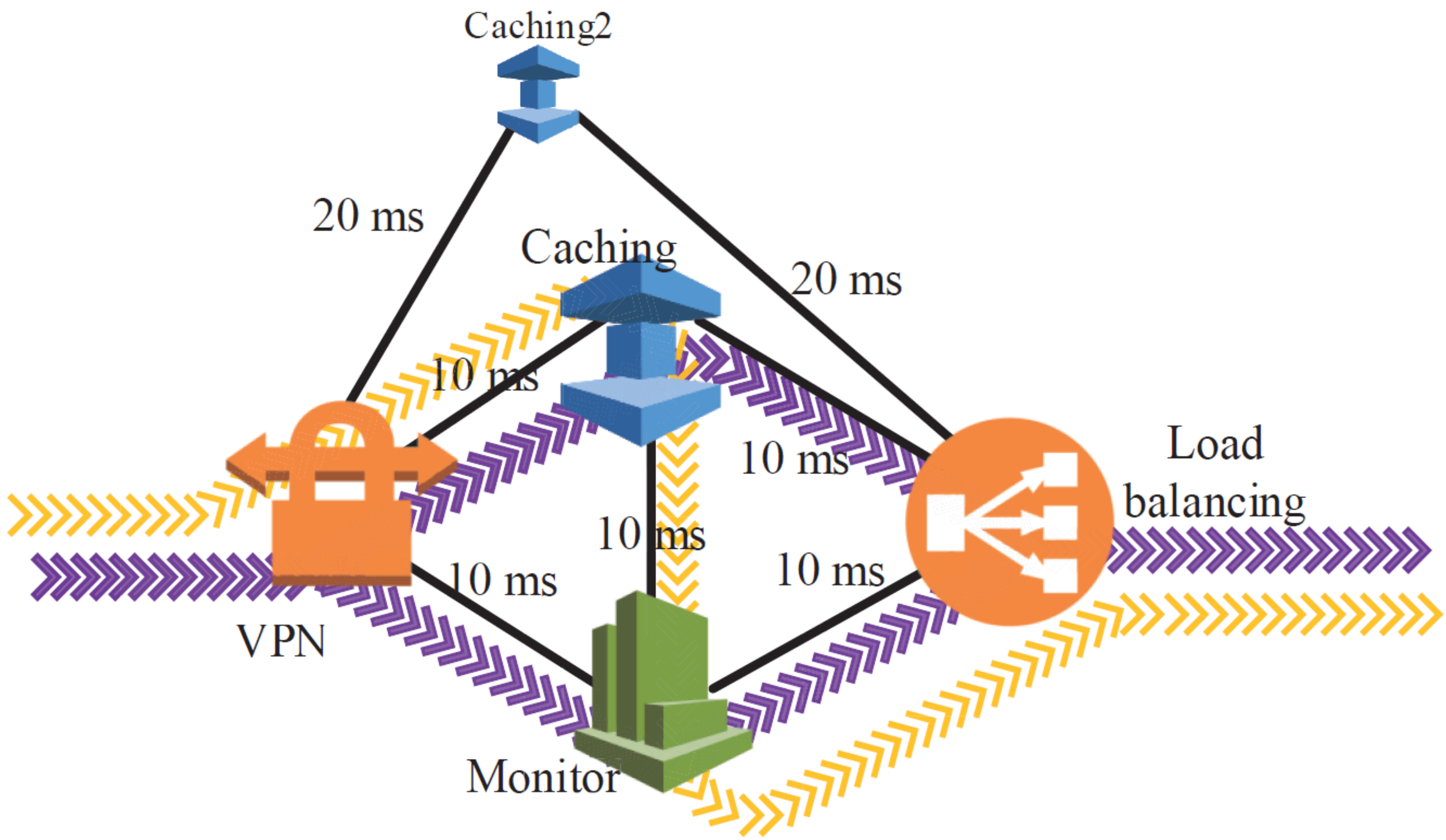
- Combining forwarding devices and middleboxes into a common control framework
- NFV enables network operators to implement network policies without worrying about:
  - Placement: Where to place the functions (middleboxes) in the network.
  - Steering: How to route traffic through these functions.

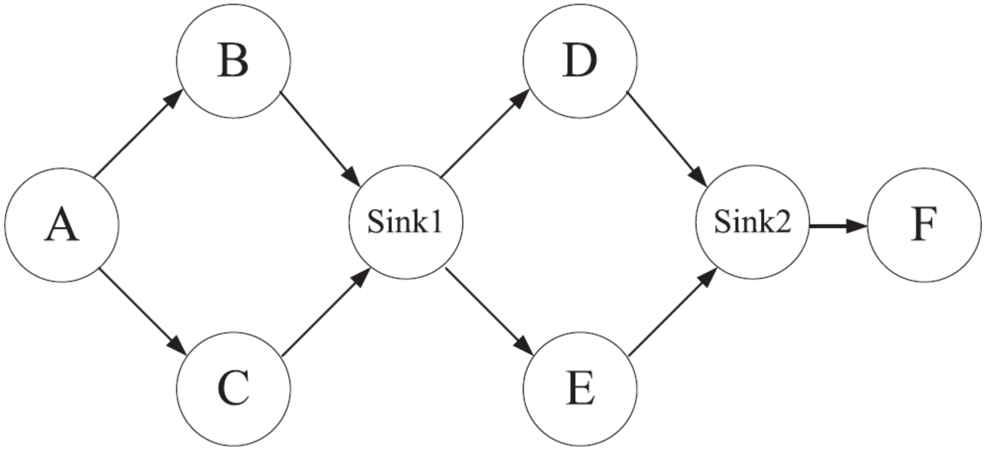
  - Placement and steering are two difficult problems for the traditional network with middleboxes.

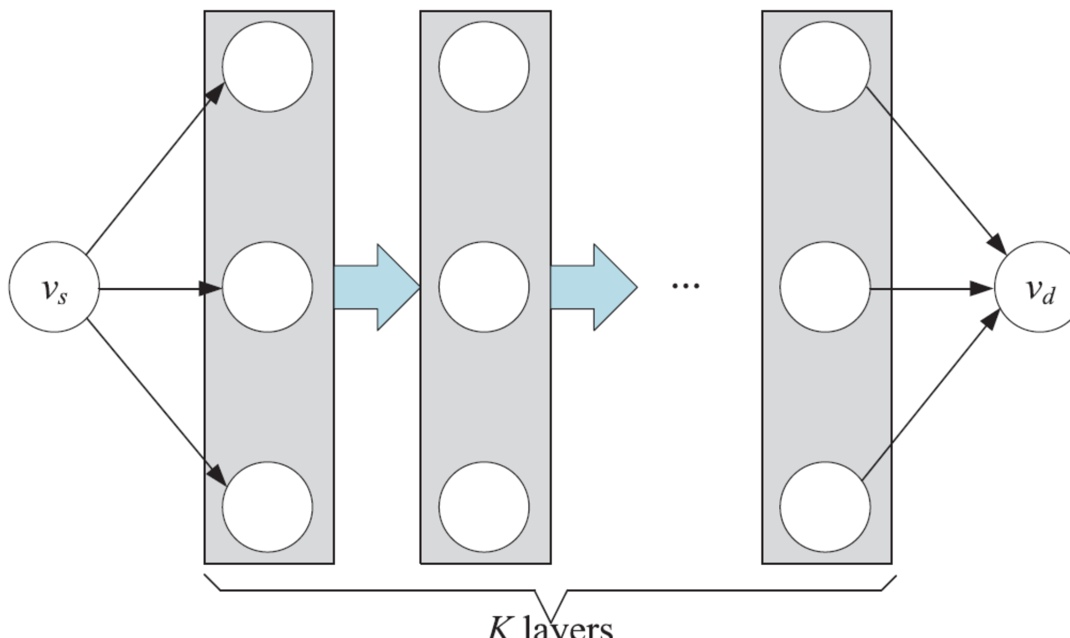
NFV

- In NFV, the functions of middleboxes are decoupled in to functions elements.
- E.g.,
  - WAN Optimizer = Caching + Deduplication + Compression + Encryption + Forward Error Correction + Rate Limiter
  - Application Firewall = IP Defragmenter + Application Detection Engine + Logger + Blocker
- Orchestration and Customisability
  - Enable network operators to implement modular network functions (elements) without worrying about how and where to install network functions.
  - Add custom middlebox functions inside network data plane.
- Require network programming!!!

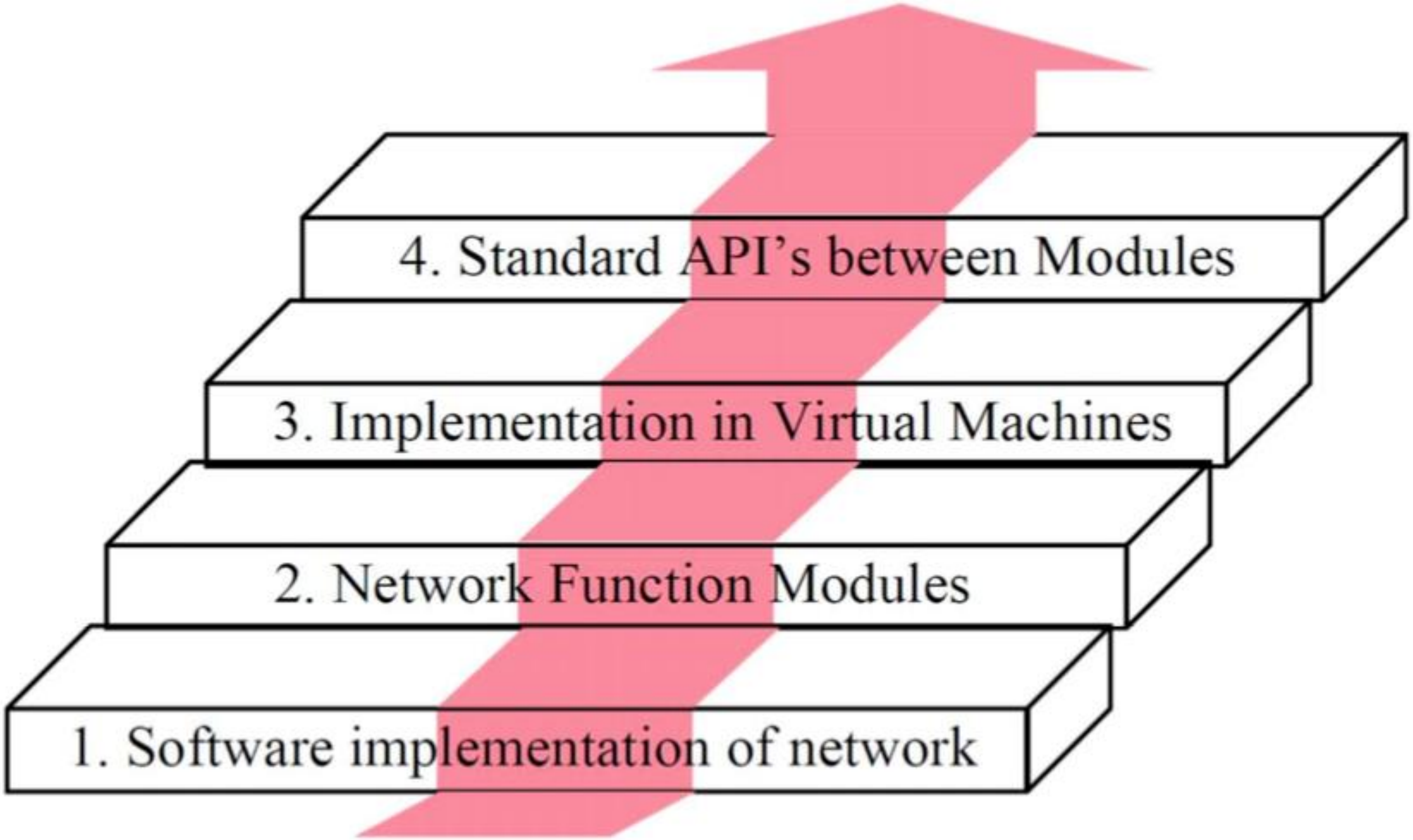
VNF Placement and Parallelism




B
D
A
Sink1
Sink2
F
C
E

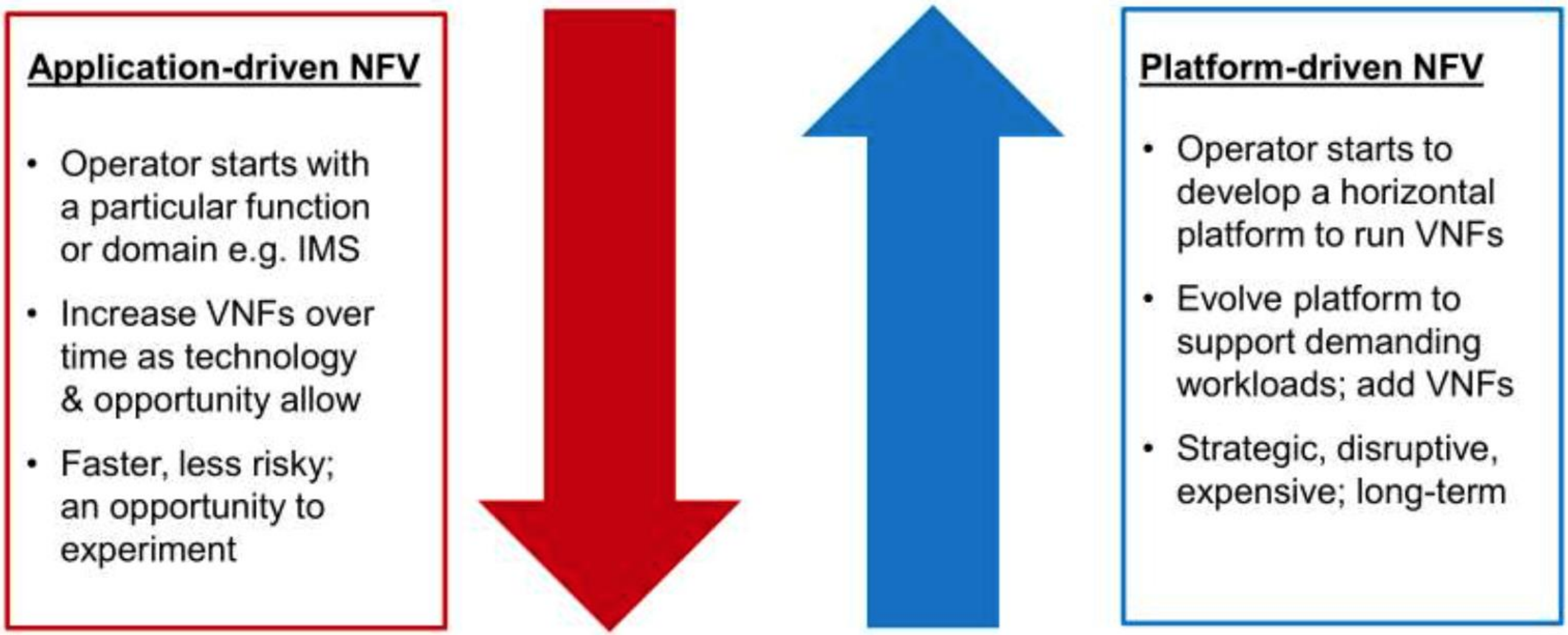

v_s
...
v_d
K layers

Four Innovations of NFV



Source: Adapted from Raj Jain

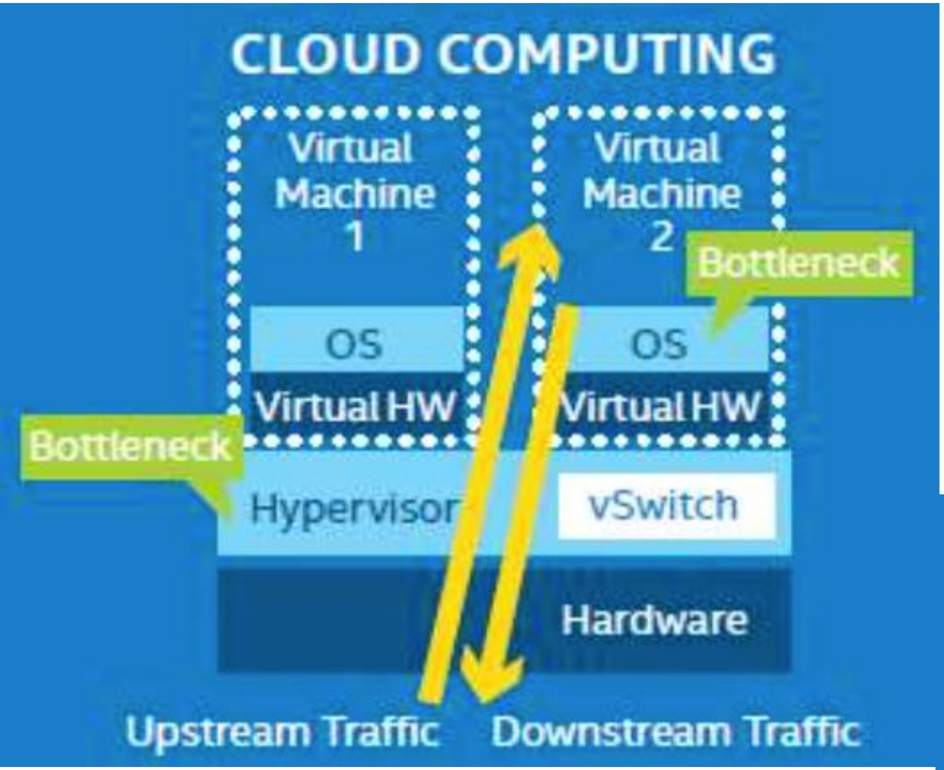
Two approaches to NFV



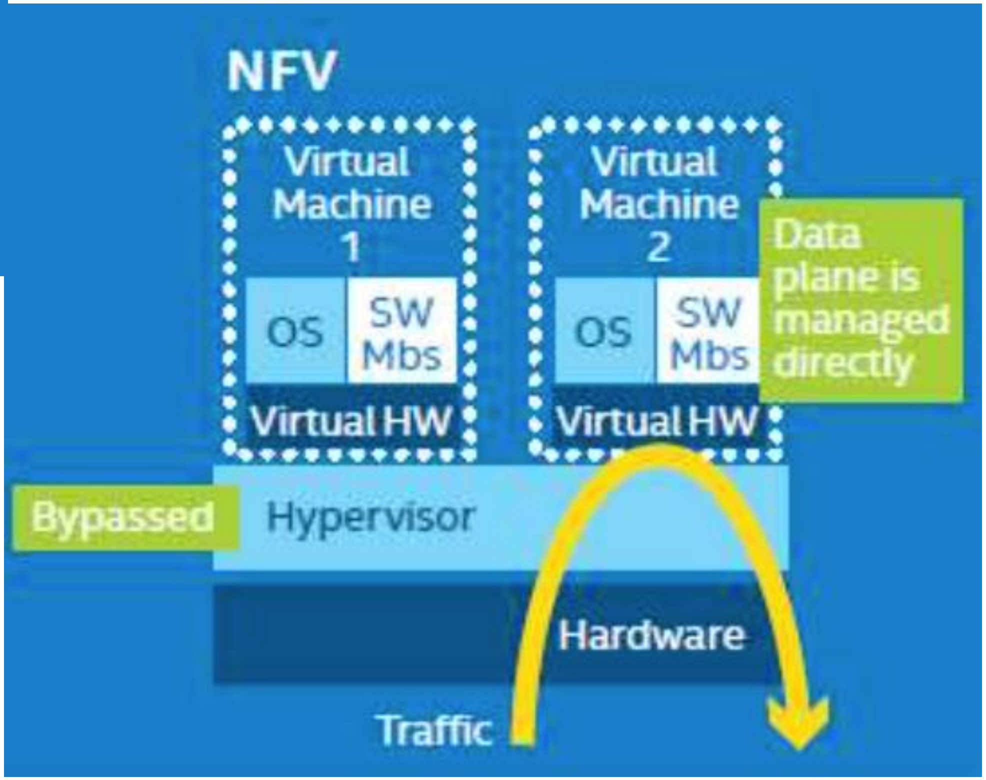
– Comparing to SDN, NFV can be implemented gradually.

Source: Gabriel Brown, Heavy Reading

NFV vs Cloud

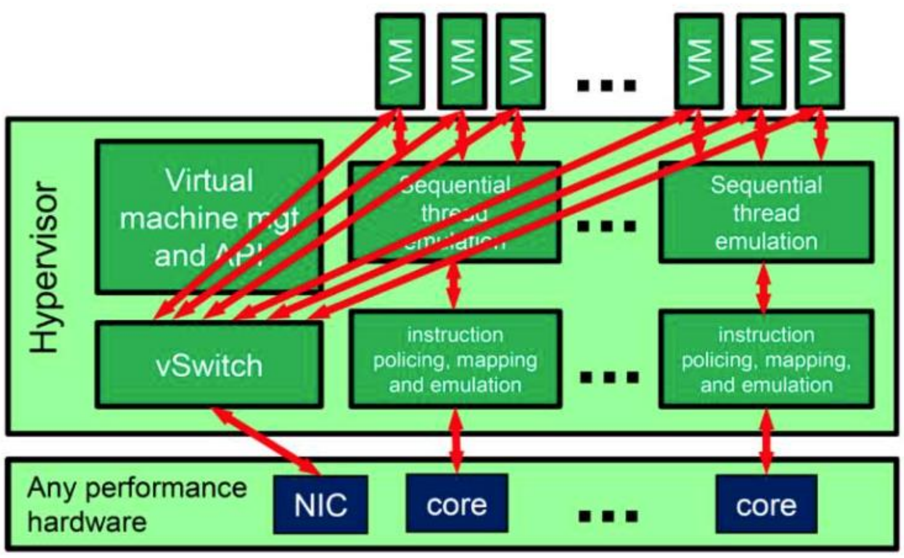


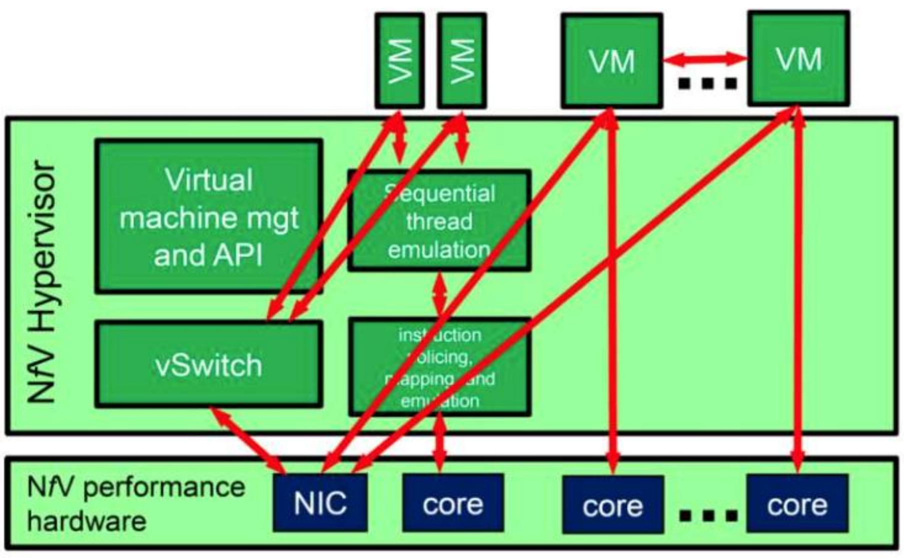
Data flow in the Virtualization Architecture



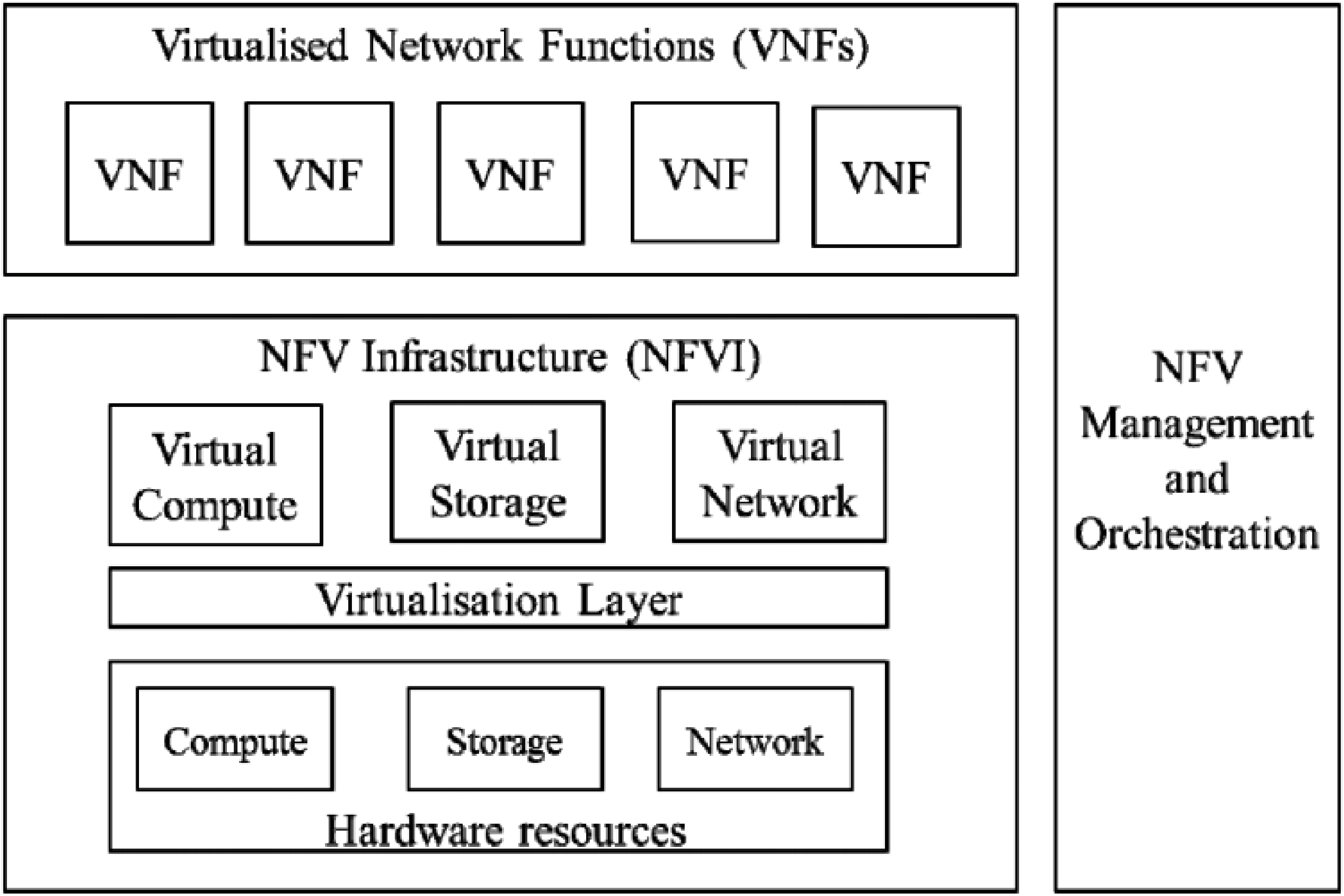
Performance Challenges of NFV

- Bottlenecks
  - TCP stack and Linux kernel in the NFV virtual machines
  - Hypervisor virtual switch / VLANs occupy the NIC
- Solutions (0 copy)
  - Optimised virtual switches (e.g., Intel DPDK)
  - Dedicate virtual NICs – bypass hypervisor
  - Dedicate packet processing CPU cores

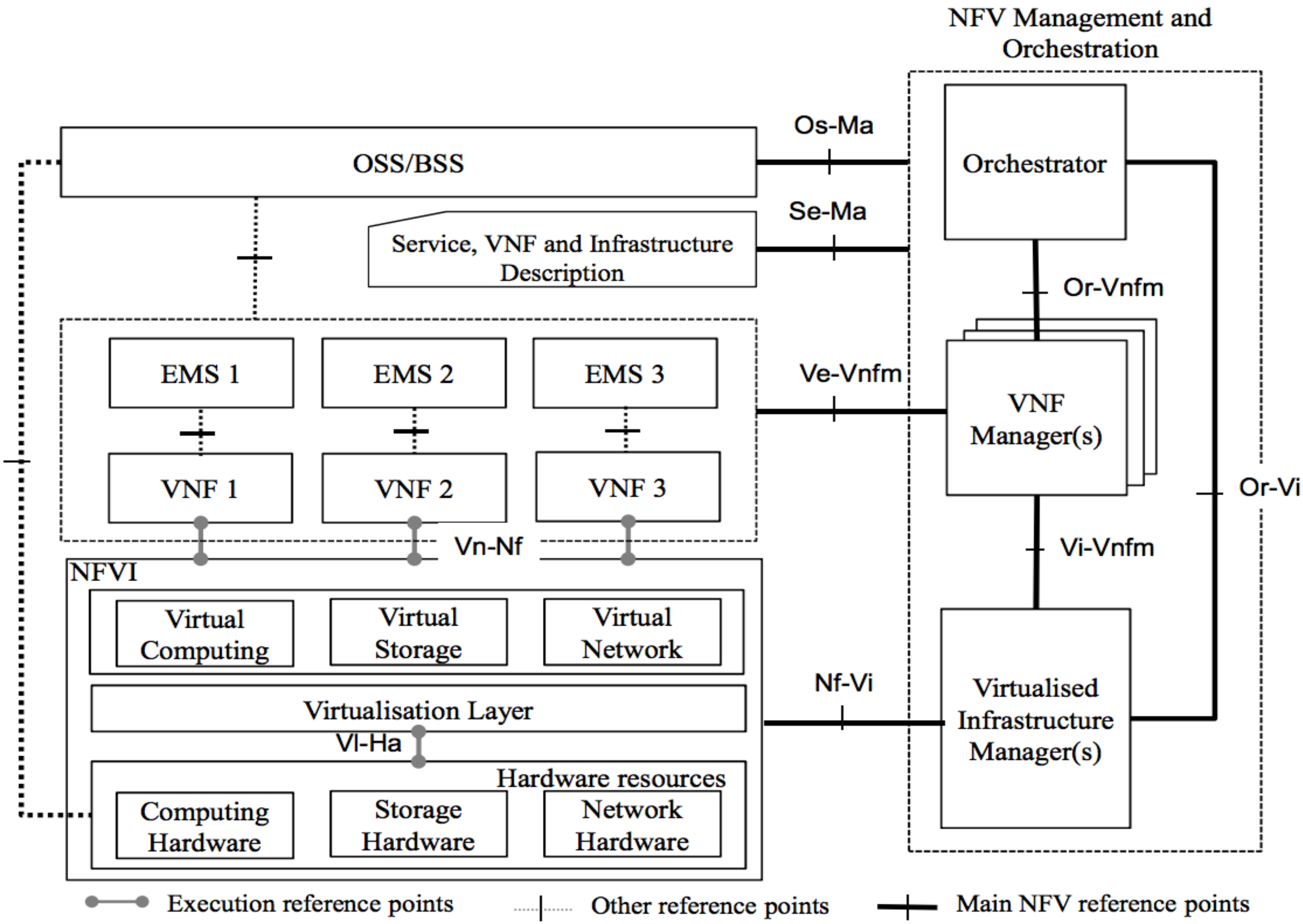




NFV architecture



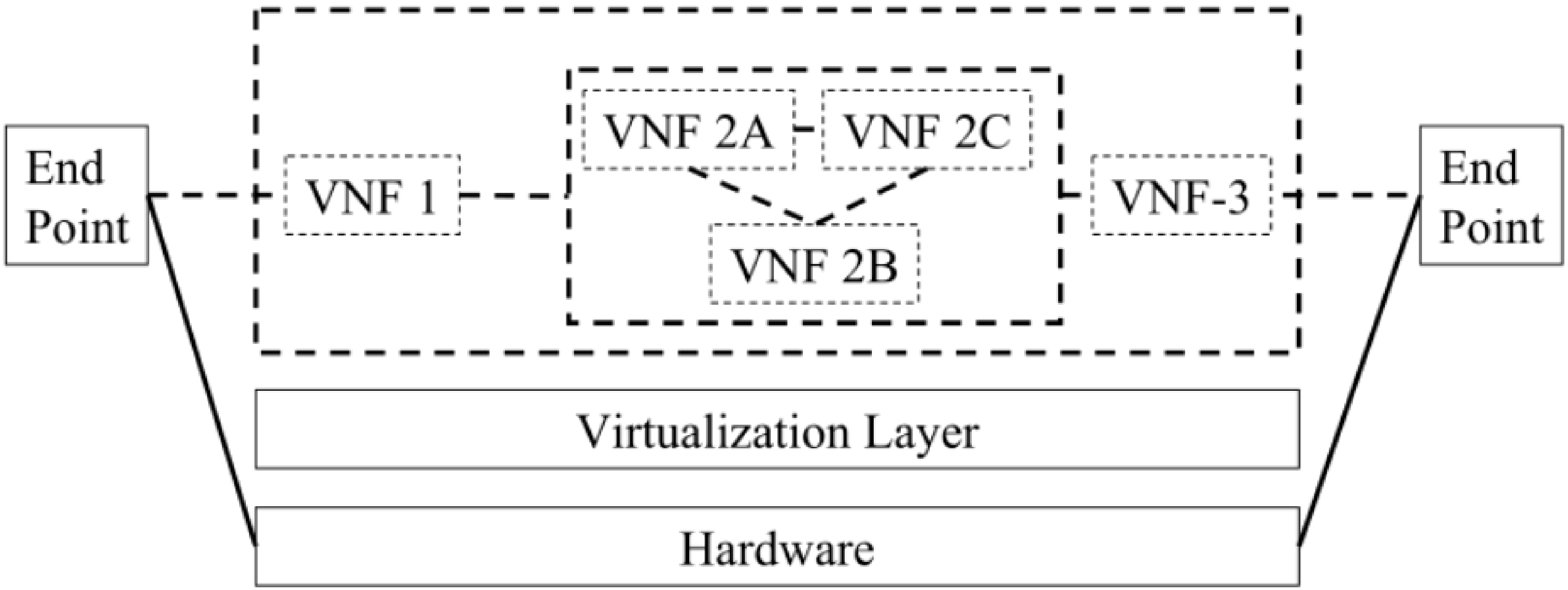
NFV reference architecture framework



https://www.etsi.org/deliver/etsi_gs/NFV/001_099/002/01.01.01_60/gs_NFV002v010101p.pdf

Orchestration

- An end-to-end perspective
  - May include nested forwarding graphs



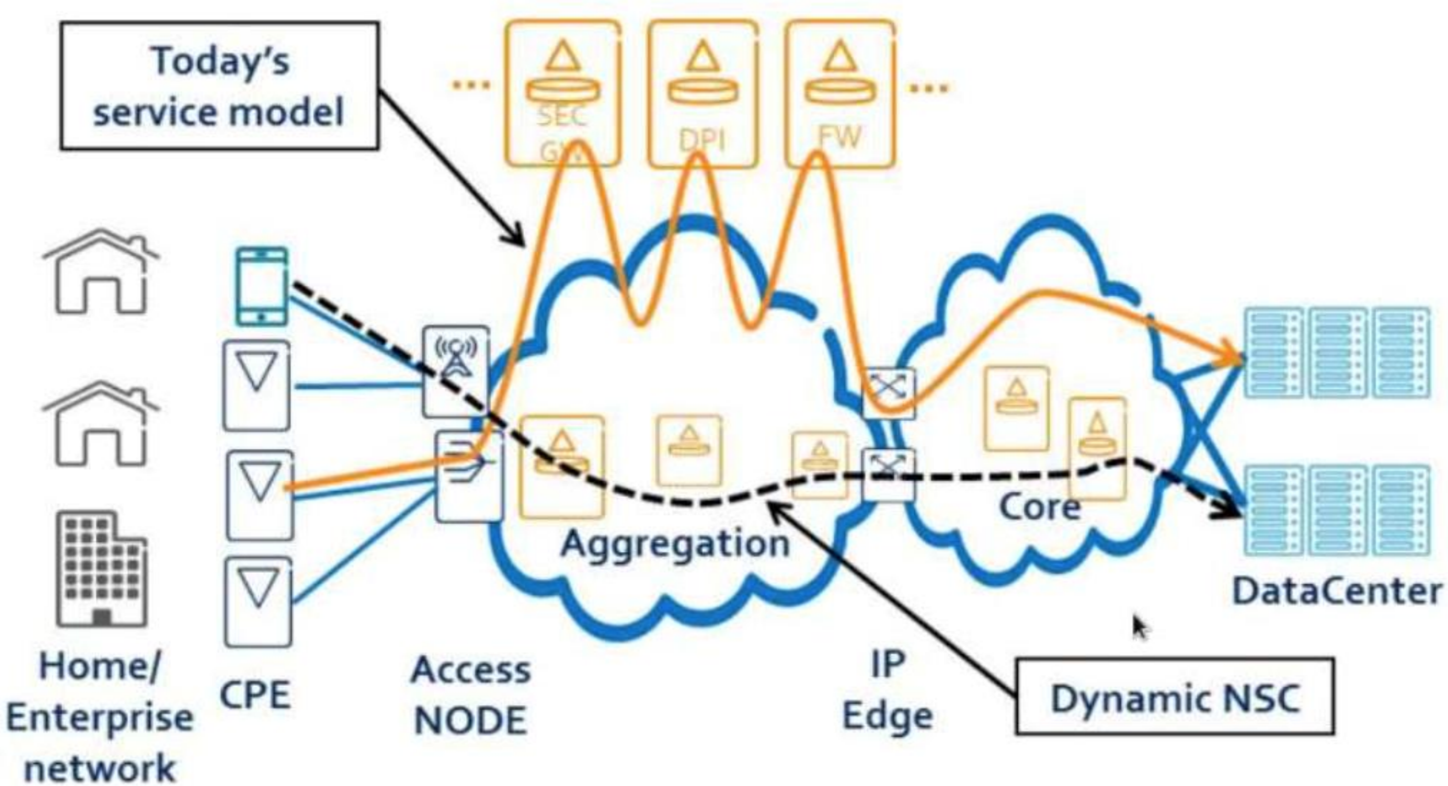
Orchestration

- Automated deployment of NFV applications
  - Orchestration console, Higher level carrier OSS
- Tools exist for automated cloud deployment
  - vSphere, Openstack, Cloudstack
  - Kubernetes
- NFV infrastructure profile for NFV application to
  - Select host, Configure host, Start VM(s)
- Application profile to specify
  - Service address assignment (mechanism)
  - Location specific configuration

Service Chains and VNF Forwarding Graphs

- VNF FGs are the analogue of connecting existing Physical Appliances via cables as described in the NFV
  - In other words, a VNF Forwarding Graph provides the logical connectivity between virtual appliances (i.e. VNFs).

Network Service Chaining

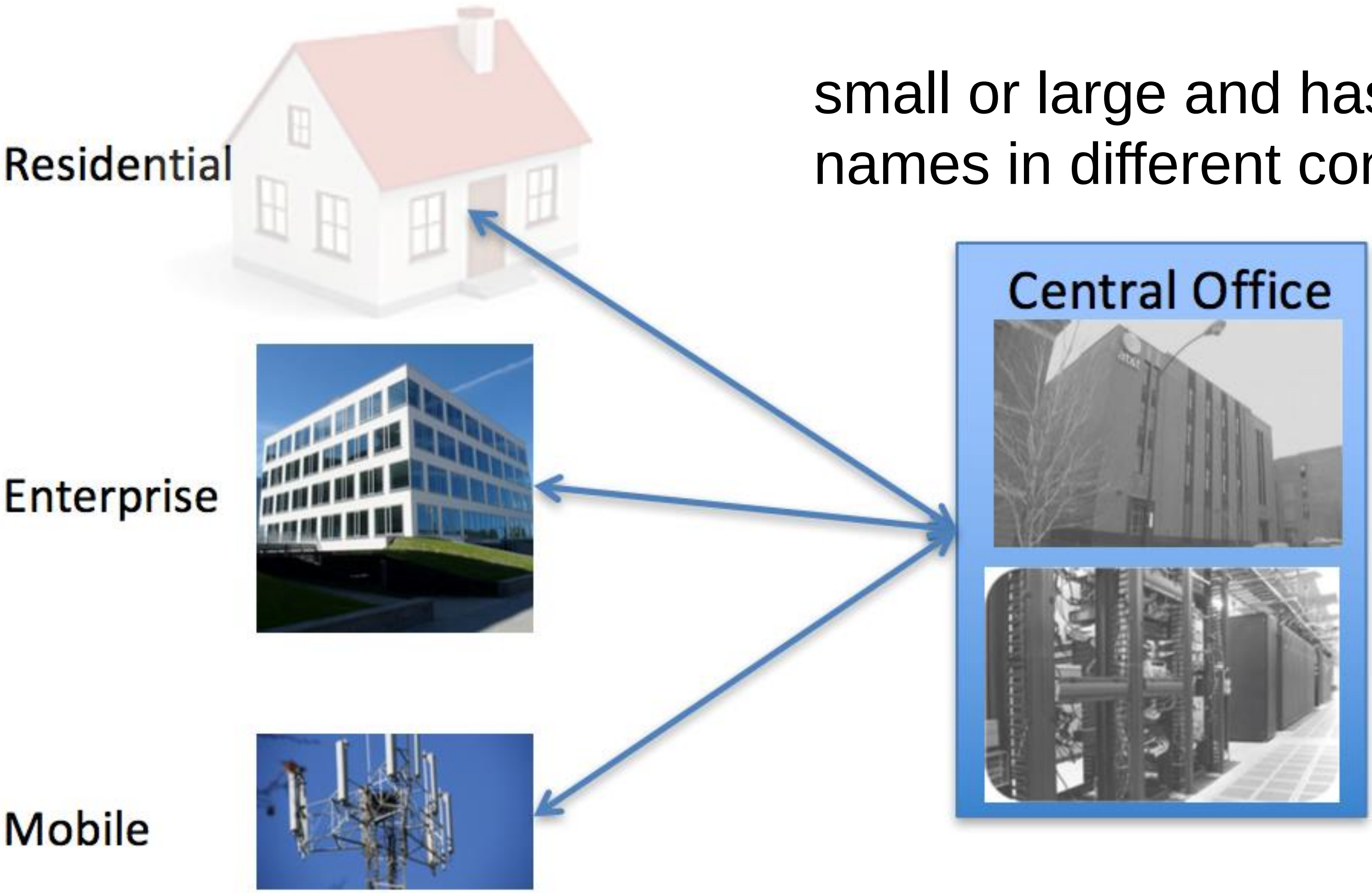


Source: Ericsson, EU UNIFY

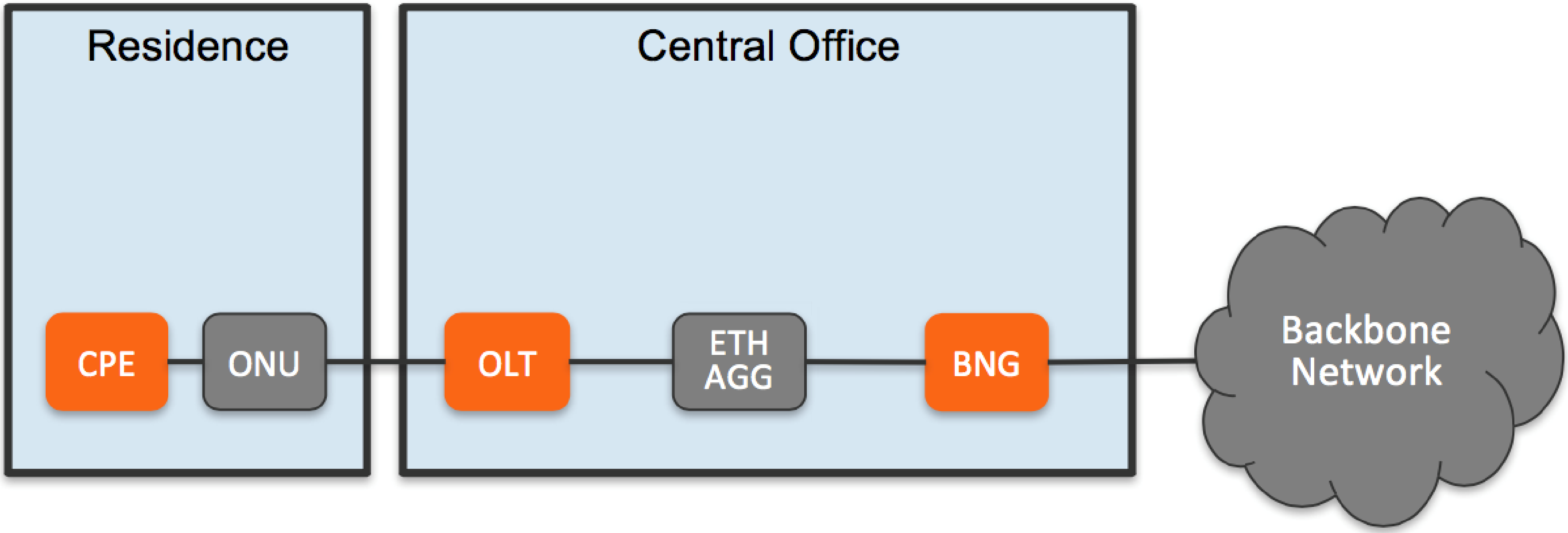
CORD

- CORD: Central Office Re-architected as a Datacenter
  - Central Office is a service provider's "gateway" to its customers

small or large and has different names in different contexts



Legacy Central Office



CPE – Customer Premises Equipment
OLT – Optical Line Termination
BNG – Broadband Network Gateway

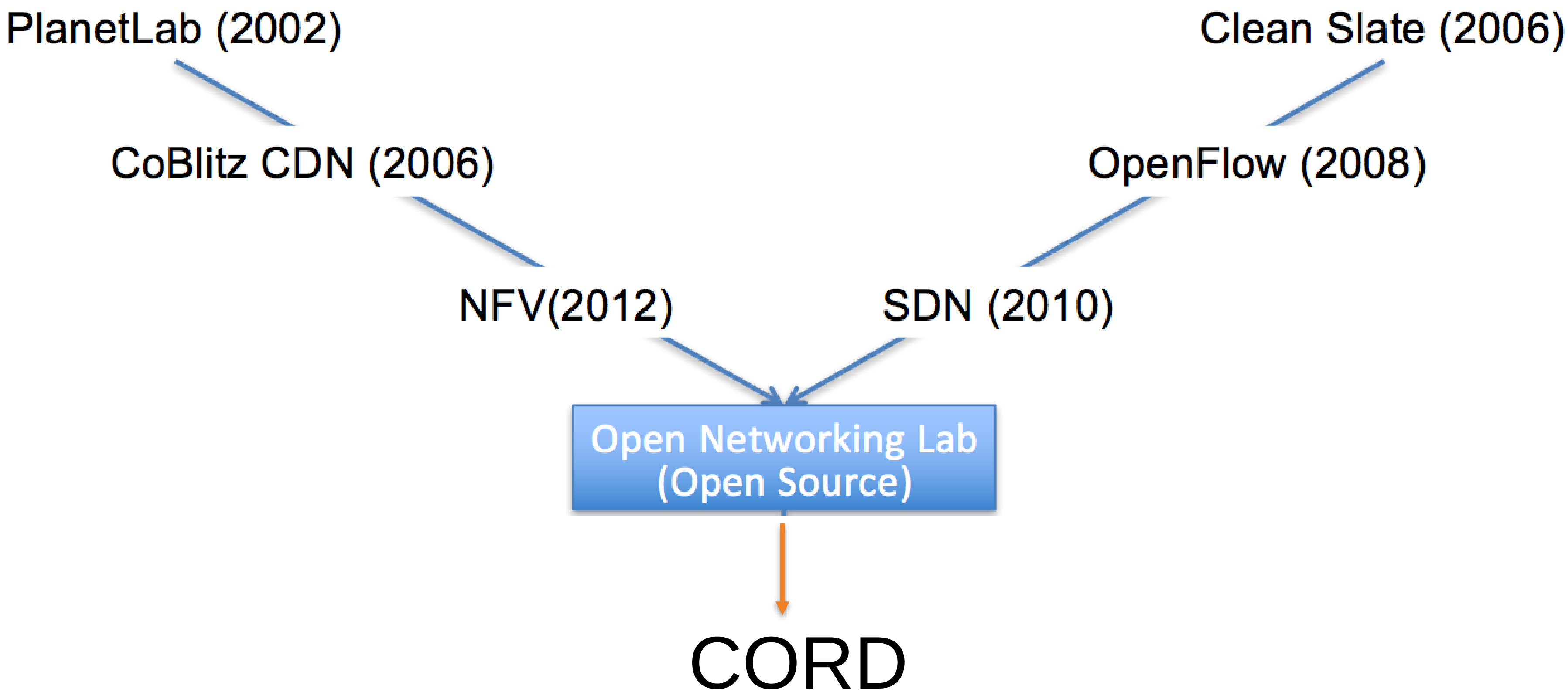
Central Office

- Challenges
  - High CAPEX and OPEX
  - Lack of programmability inhibits innovation
  - Limits ability to create new services and new revenue

- Economies of a datacenter
  - *Infrastructure built with a few commodity building blocks using open source software and white-box switches*

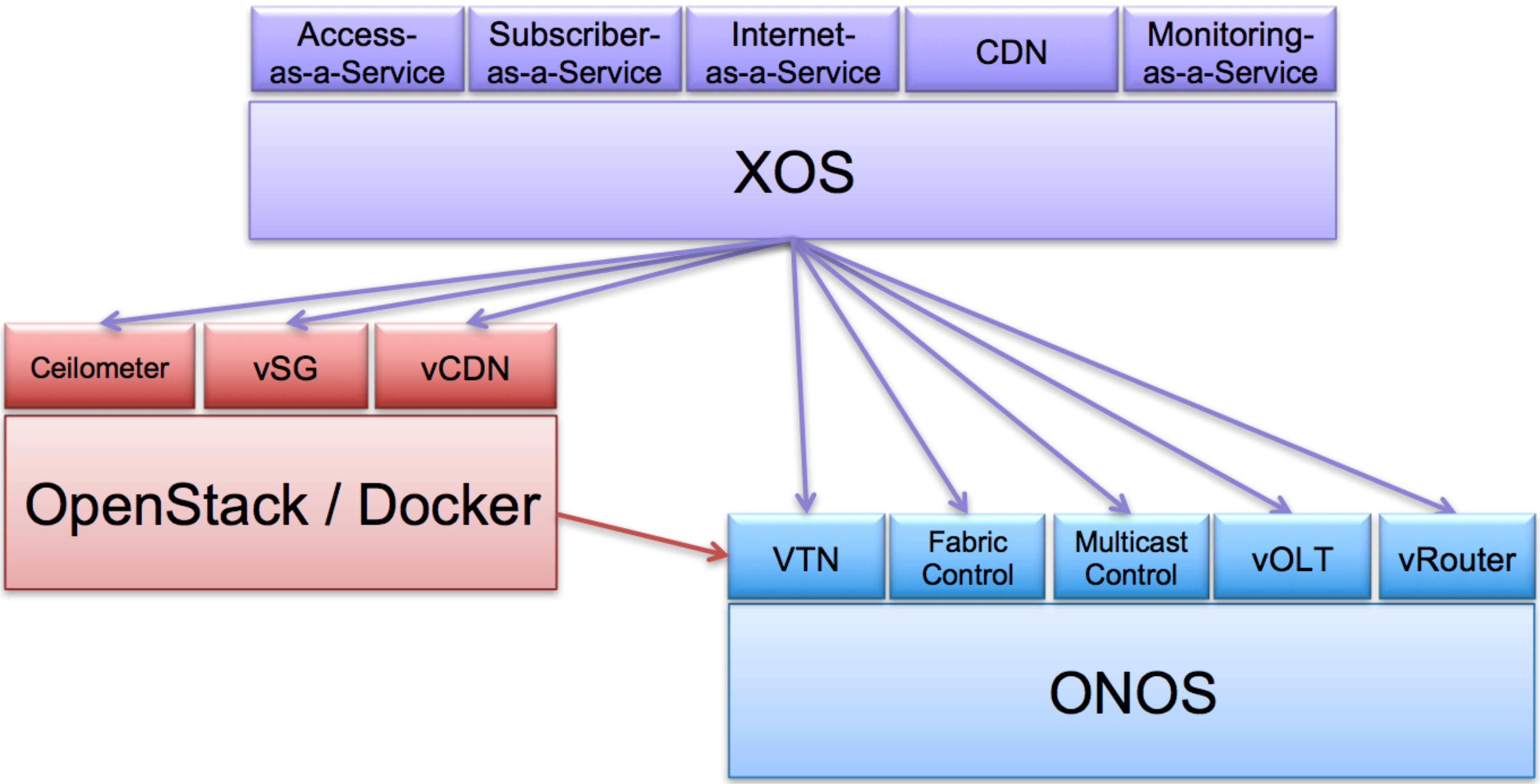
- Agility of a cloud provider
  - *Software platforms that enable rapid creation of new services*
  - *From Access-as-a-Service to Software-as-a-Service*

History



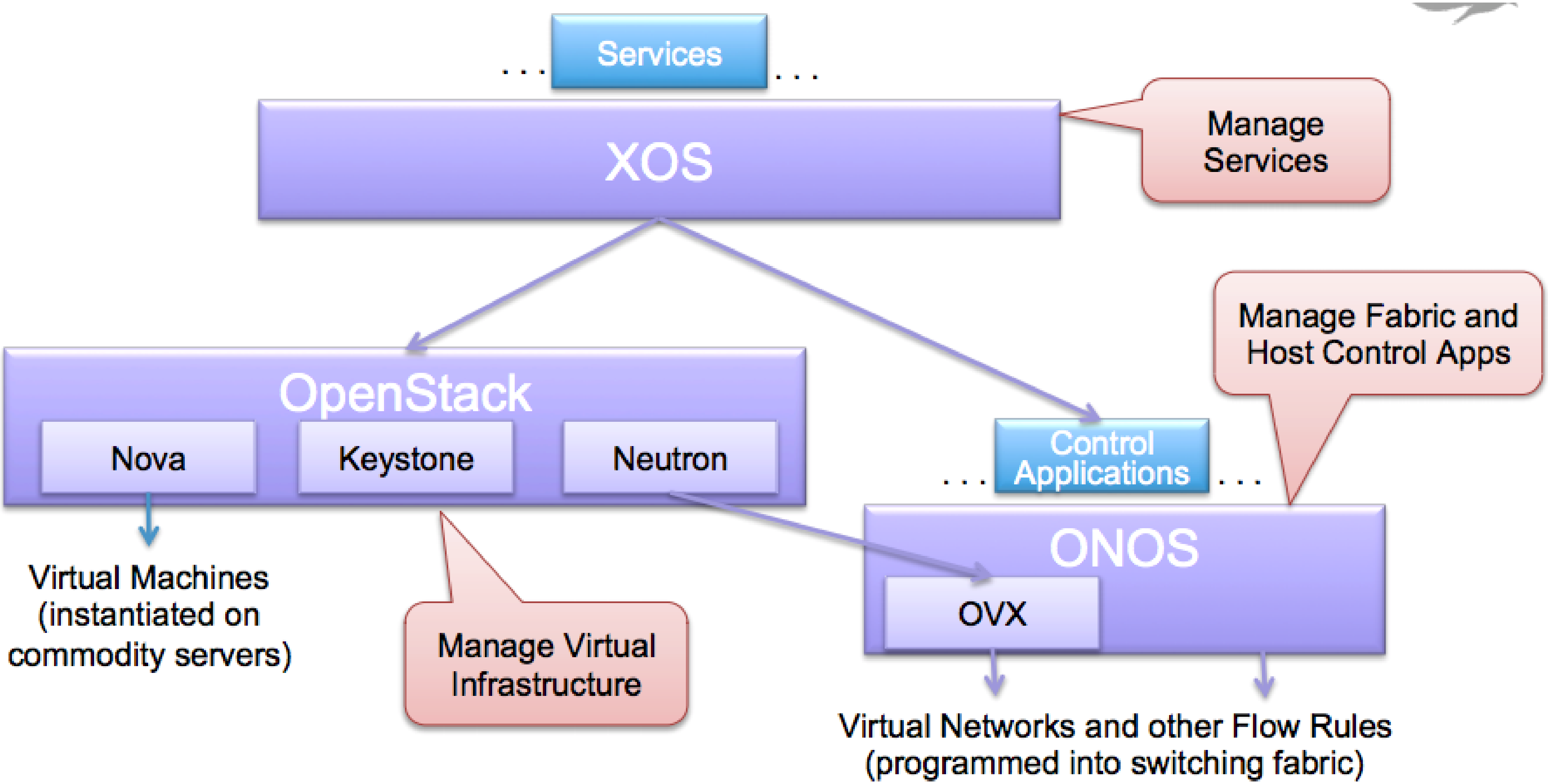
- **ON.LAB initiated many main stream SDN/NFV projects and put in the CORD.**
  - **Changed name to Open Networking Foundation in 2017**

Some key components of CORD



- After version 7.0
  - OpenStack has been replaced by Kubernetes

CORD



Services
XOS
Manage Services
OpenStack
Nova
Keystone
Neutron
Control Applications
Manage Fabric and Host Control Apps
ONOS
OVX
Virtual Machines (instantiated on commodity servers)
Manage Virtual Infrastructure
Virtual Networks and other Flow Rules (programmed into switching fabric)

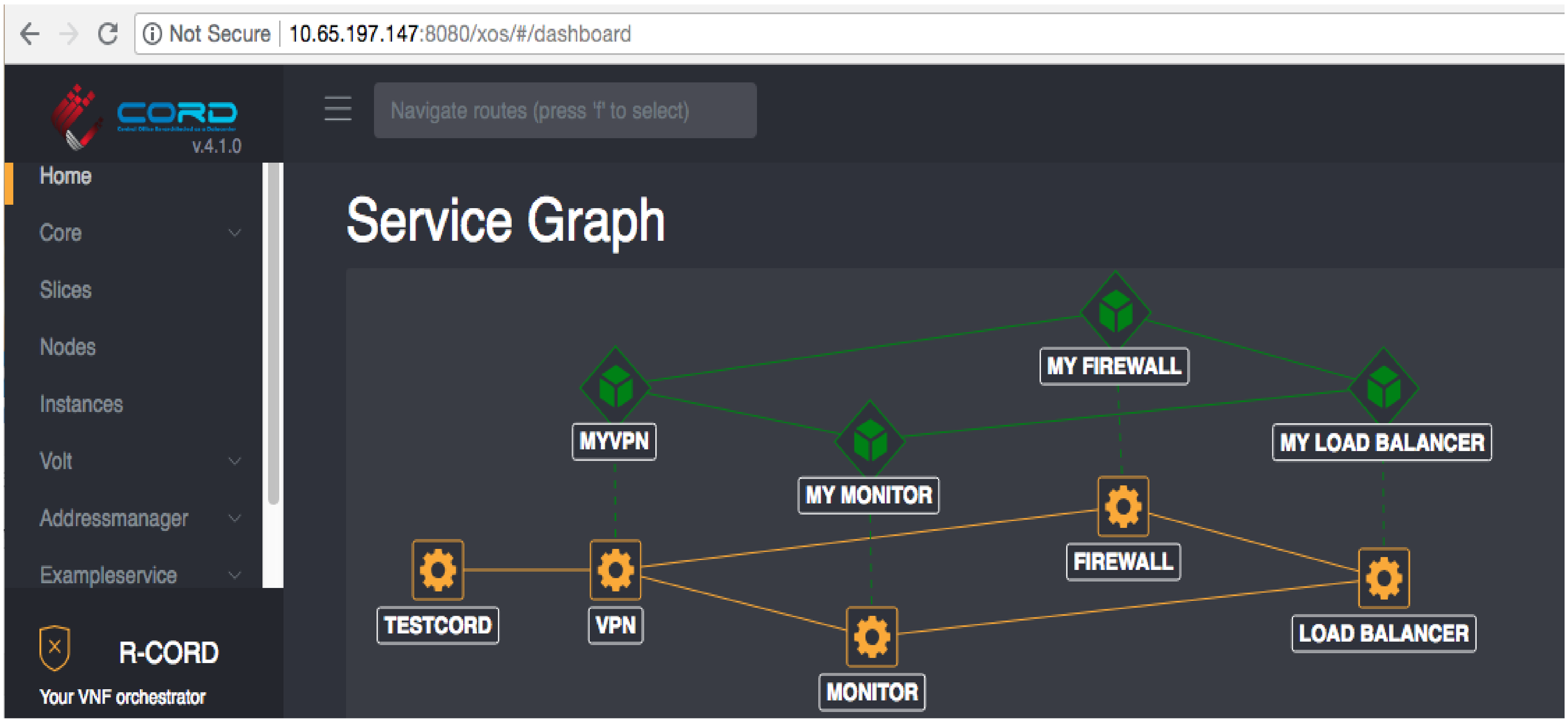
XOS

- XOS defines a coherent framework for combining SDN, NFV, and Cloud services, all running on commodity hardware, to build cost-effective and agile networks.

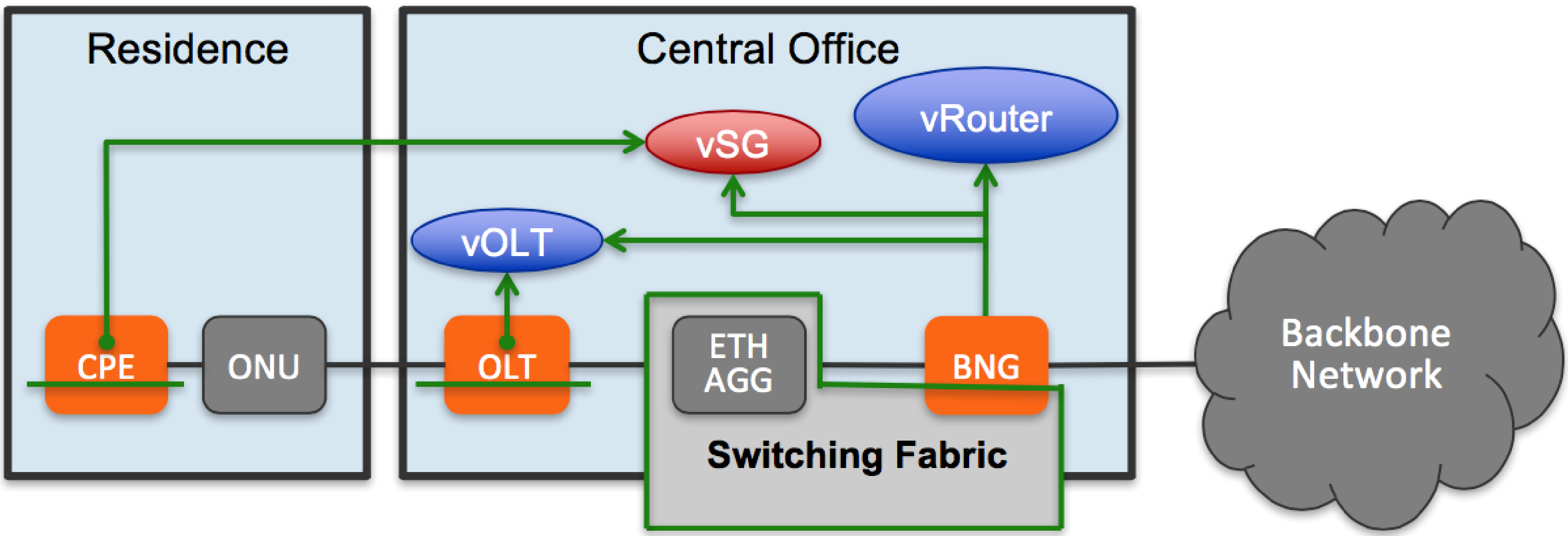
- XOS is from Open Networking Foundation.
  - Same family with ONOS

- A service orchestration layer built on top of OpenStack/Kubernetes and ONOS that manages scalable services running in a Central Office Re-architected as a Datacenter (CORD)

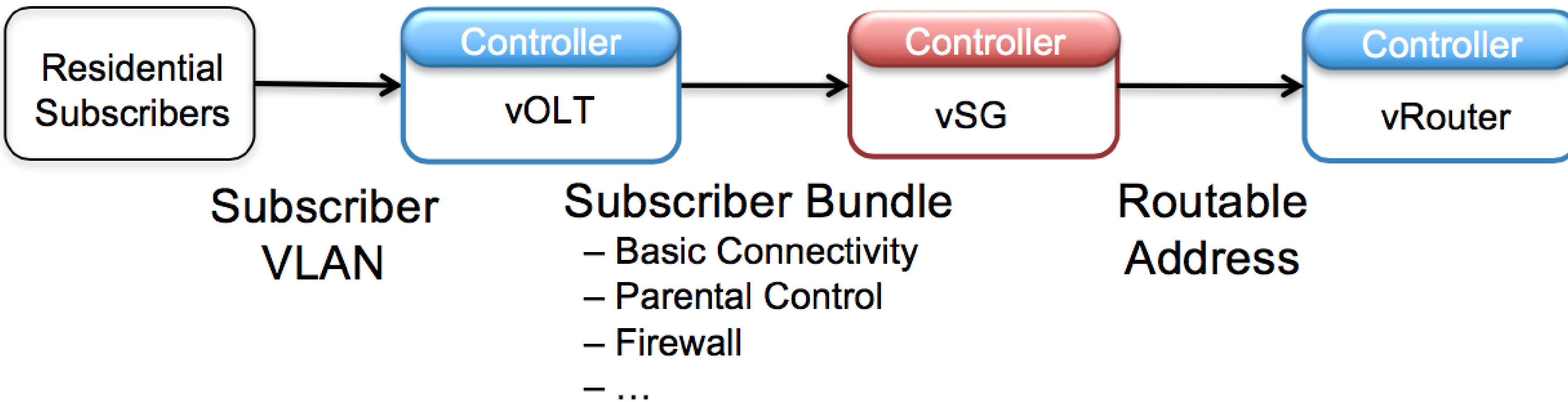
XOS GUI

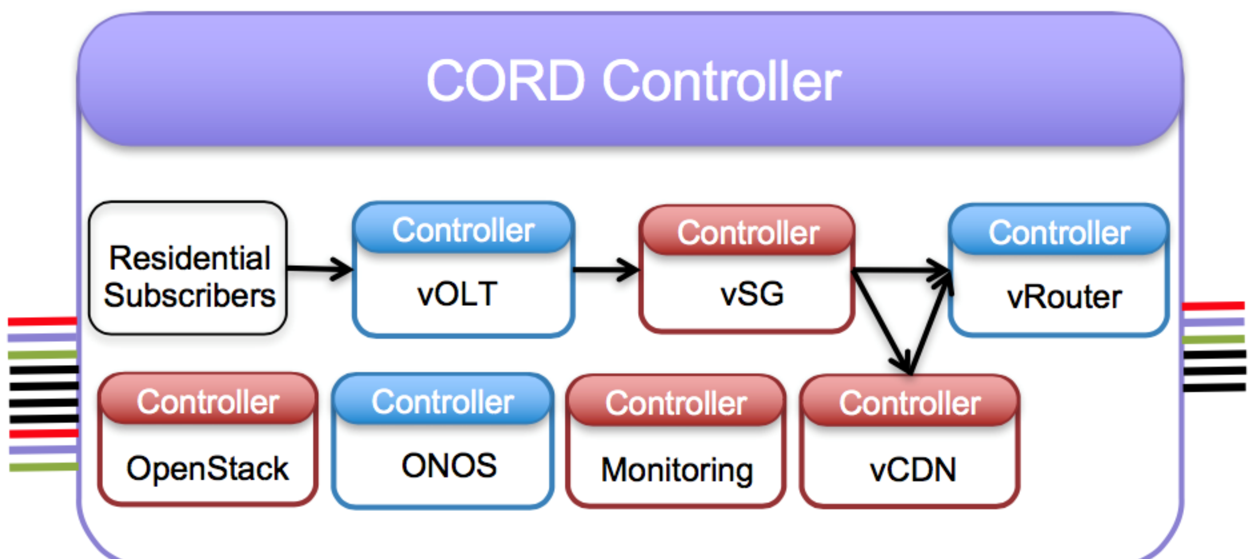

Not Secure
10.65.197.147:8080/xos/#/dashboard
CORD
v.4.1.0
Navigate routes (press 'f' to select)
Home
Core
Slices
Nodes
Instances
Volt
Addressmanager
Exampleservice
R-CORD
Your VNF orchestrator
Service Graph
MY FIREWALL
MYVPN
MY MONITOR
MY LOAD BALANCER
FIREWALL
TESTCORD
VPN
MONITOR
LOAD BALANCER

Virtualisation



Everything as a service in CORD





Contents

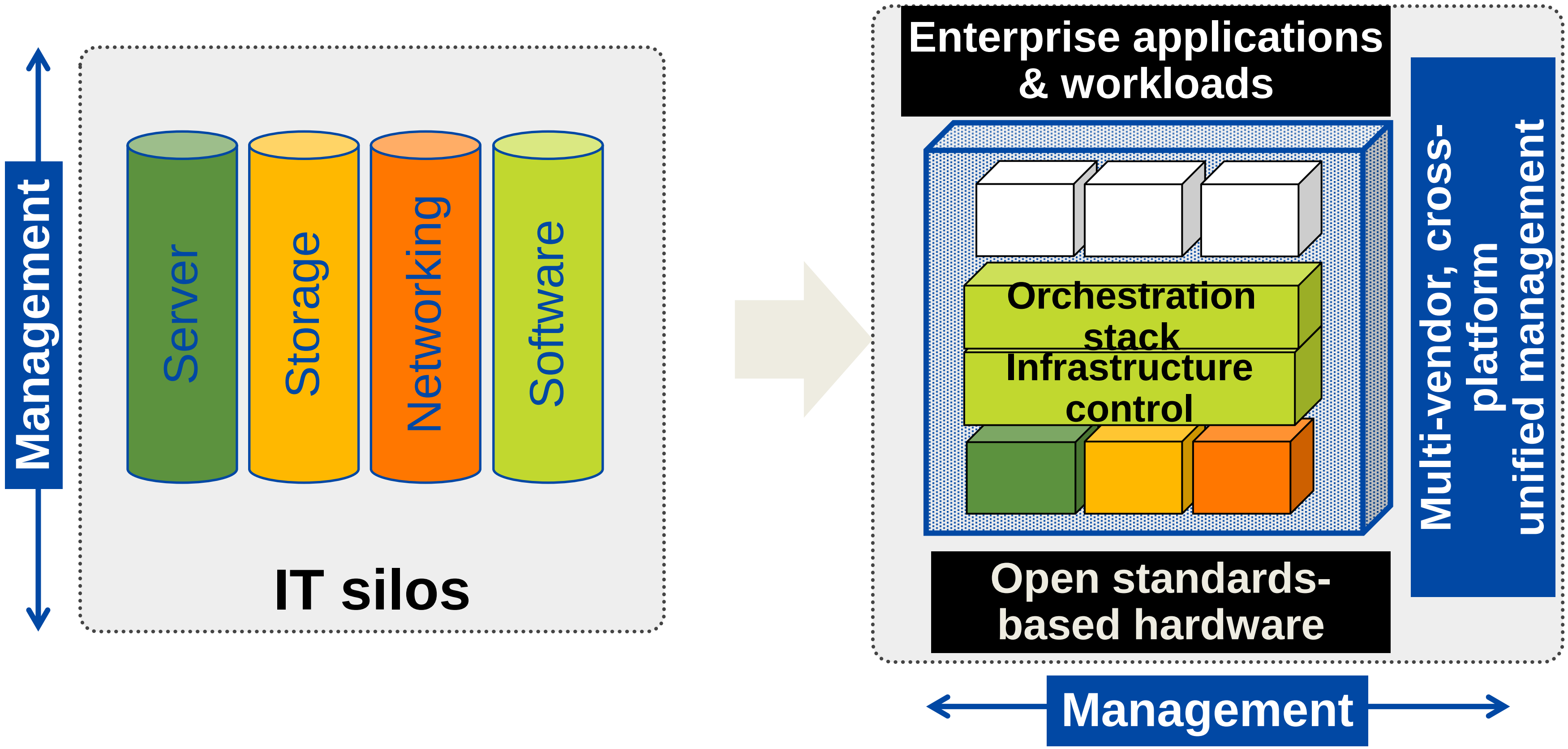
- NFV and CORD

- Software Defined DataCenters (SDDC)
  - Modern data centers
  - Datacenter network topology
  - Network slicing
  - Multi-tenant Datacenter

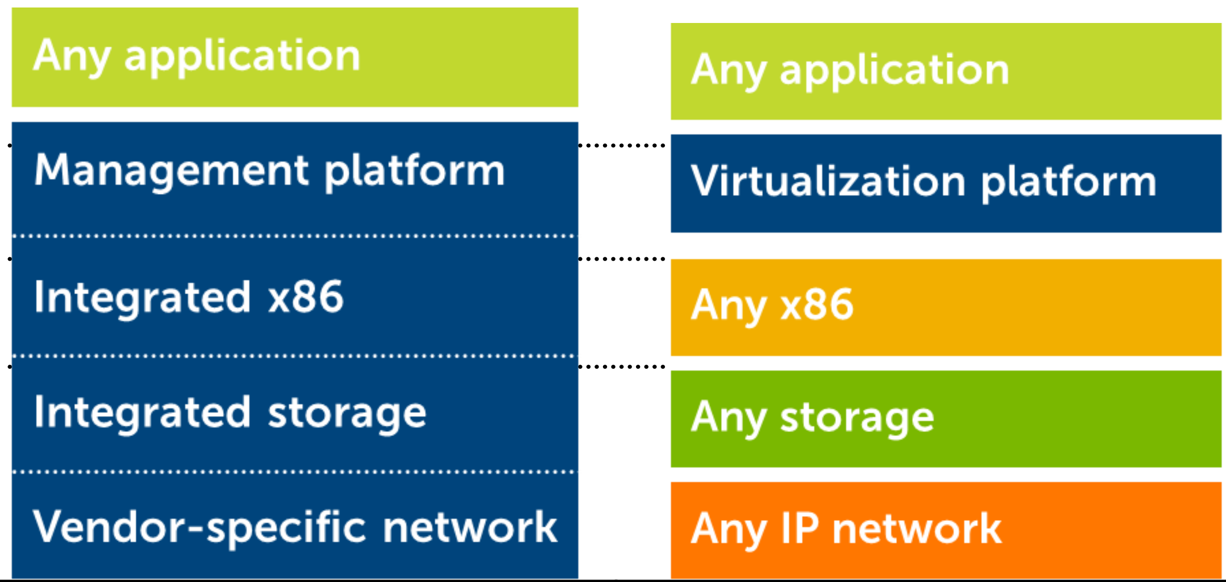
Traditional and new IT

A common compute-centric, software-driven platform for traditional and new

Shift the focus from infrastructure to service

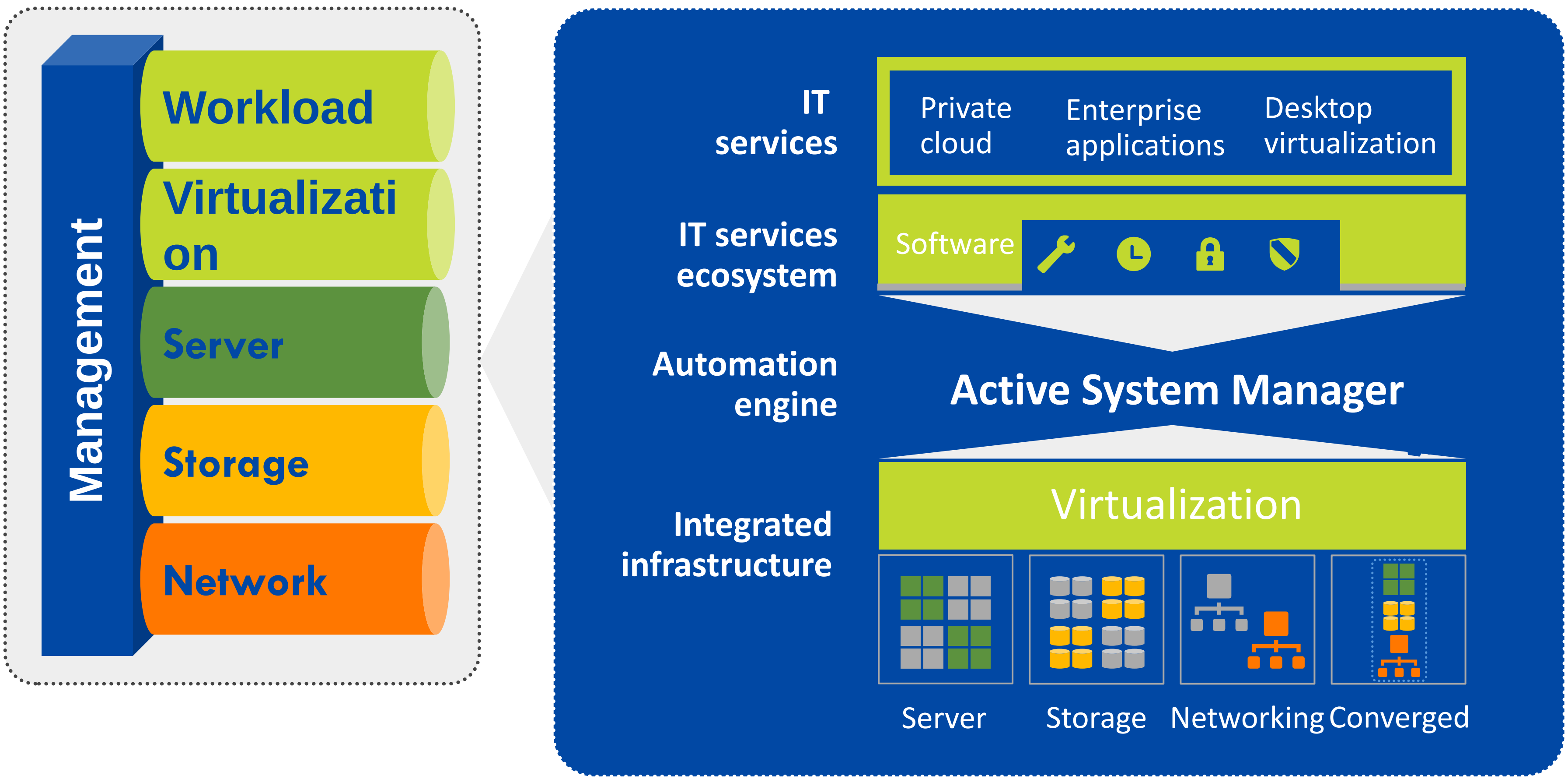


SDDC delivers needed agility and efficiency

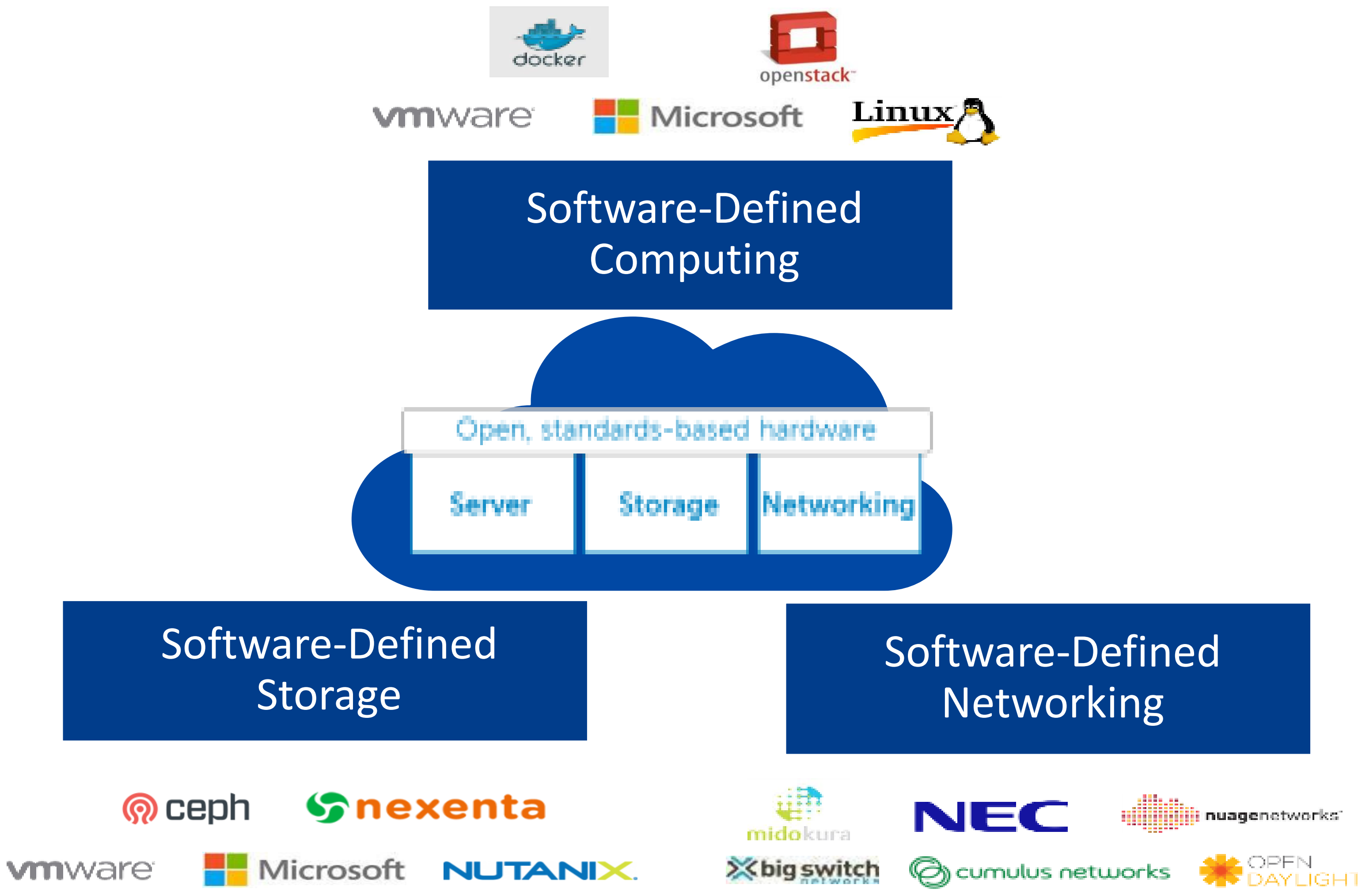


Benefit	Hardware-defined (HDDC)	Software-defined (SDDC)
Innovation	**Slow** Long hardware/ASIC cycles	**Fast** Rapid software innovation
Flexibility	**No** Lock-in	**Yes** Choice of infrastructure
Ease of insertion/ deployment	**Low** Requires forklift upgrade	**High** Non-disruptive

Enabling the unified datacenter



Enabling the Future Ready Enterprise

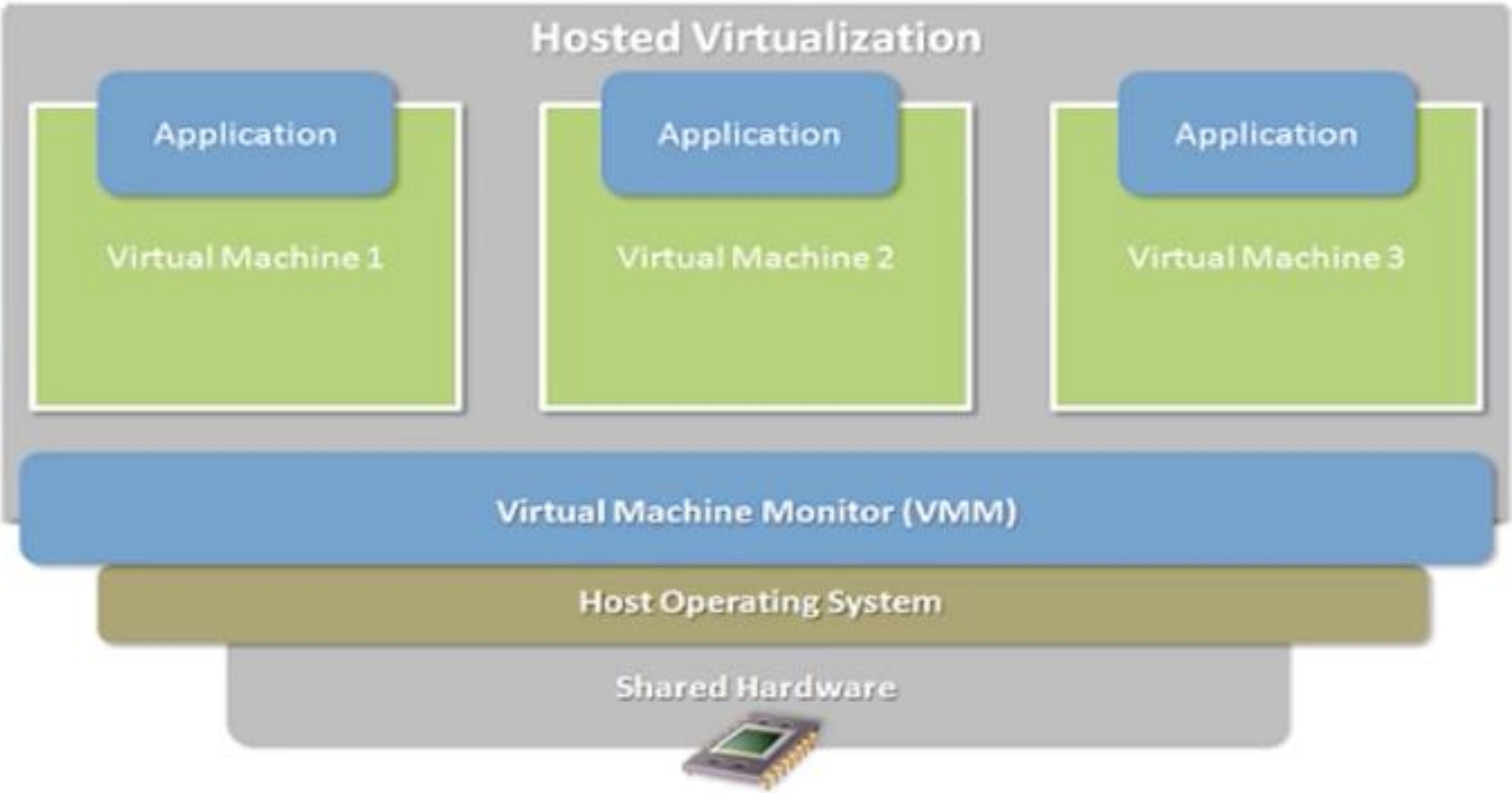


Cloud computing and data centres

- Elastic resources
  - Pay-as-you-go
  - Infrastructure on Demand
- Multi-tenancy
  - Multiple independent users
  - Amortize the cost of the share infrastructure

- New services models
  - SaaS, PaaS, IaaS

Enabling technology: Virtualisation

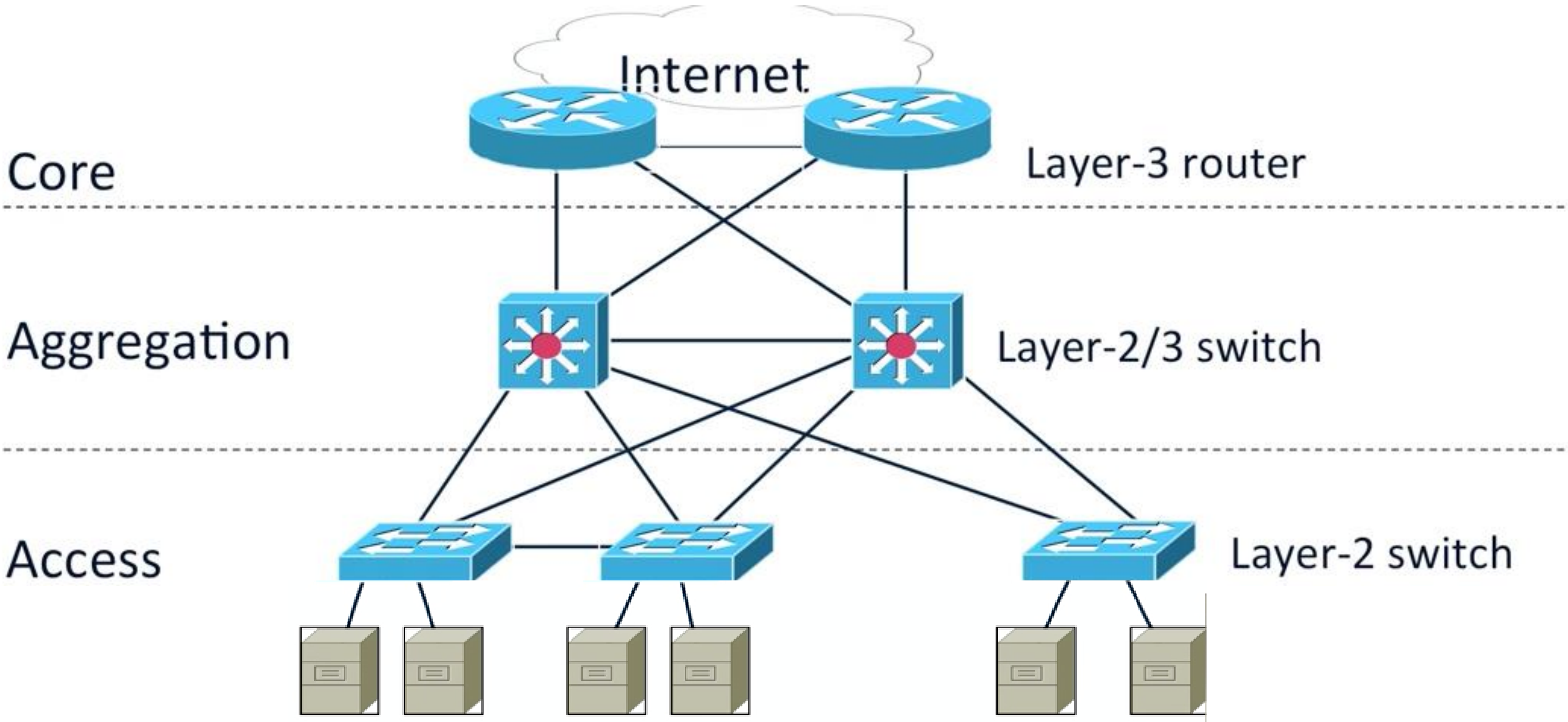


– VM can migrate from one computer to another

Design Requirement for Data Centres

- Easy migration of Virtual machines
- Minimal switch configuration
- Efficient communication along forwarding paths
- No forwarding loops
- Fast, effective failure detection

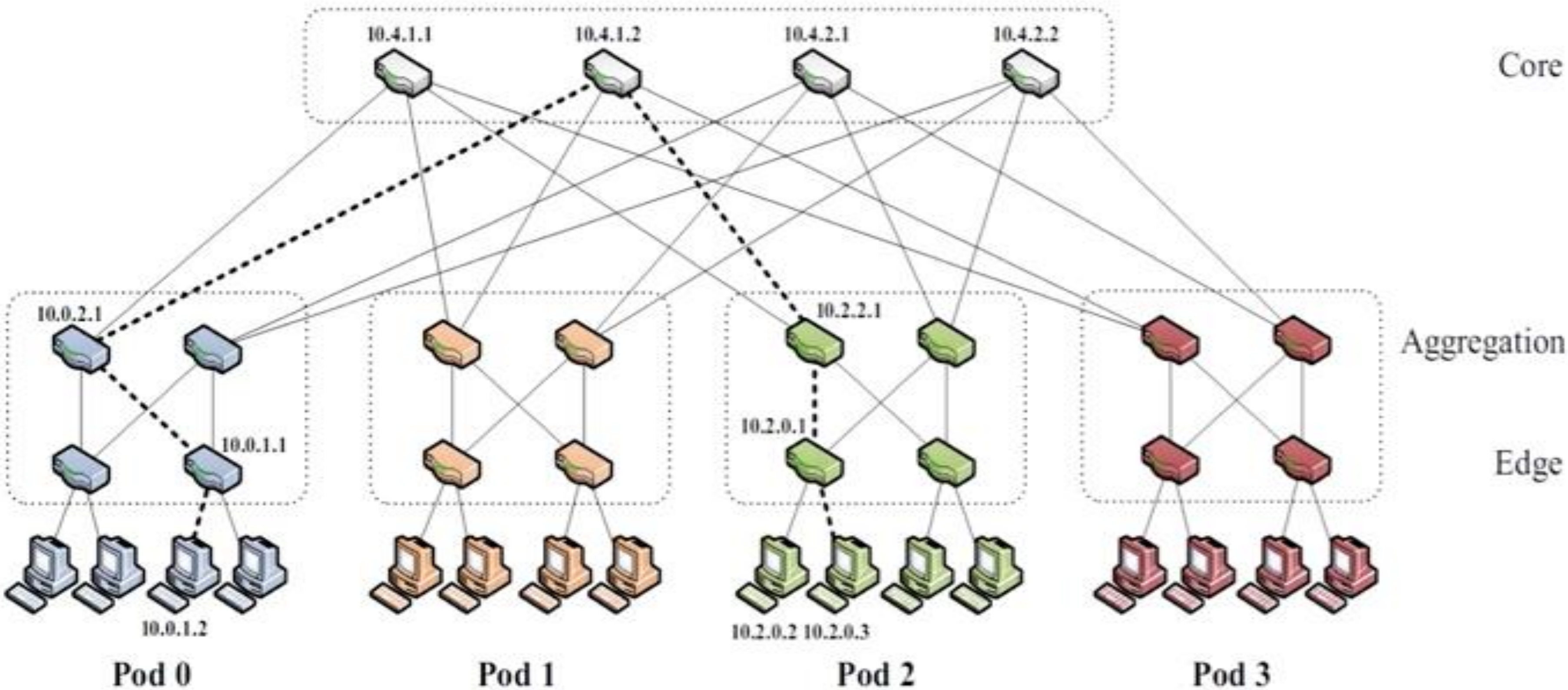
Common Data Centre Topology



- **Problems**
  - Single point of failure
  - Over subscription of links higher up in the topology

Fat-Tree Topology

- Multi-rooted tree topology
- Capacity increases towards the root(s) of the tree
- Inherent fault tolerance



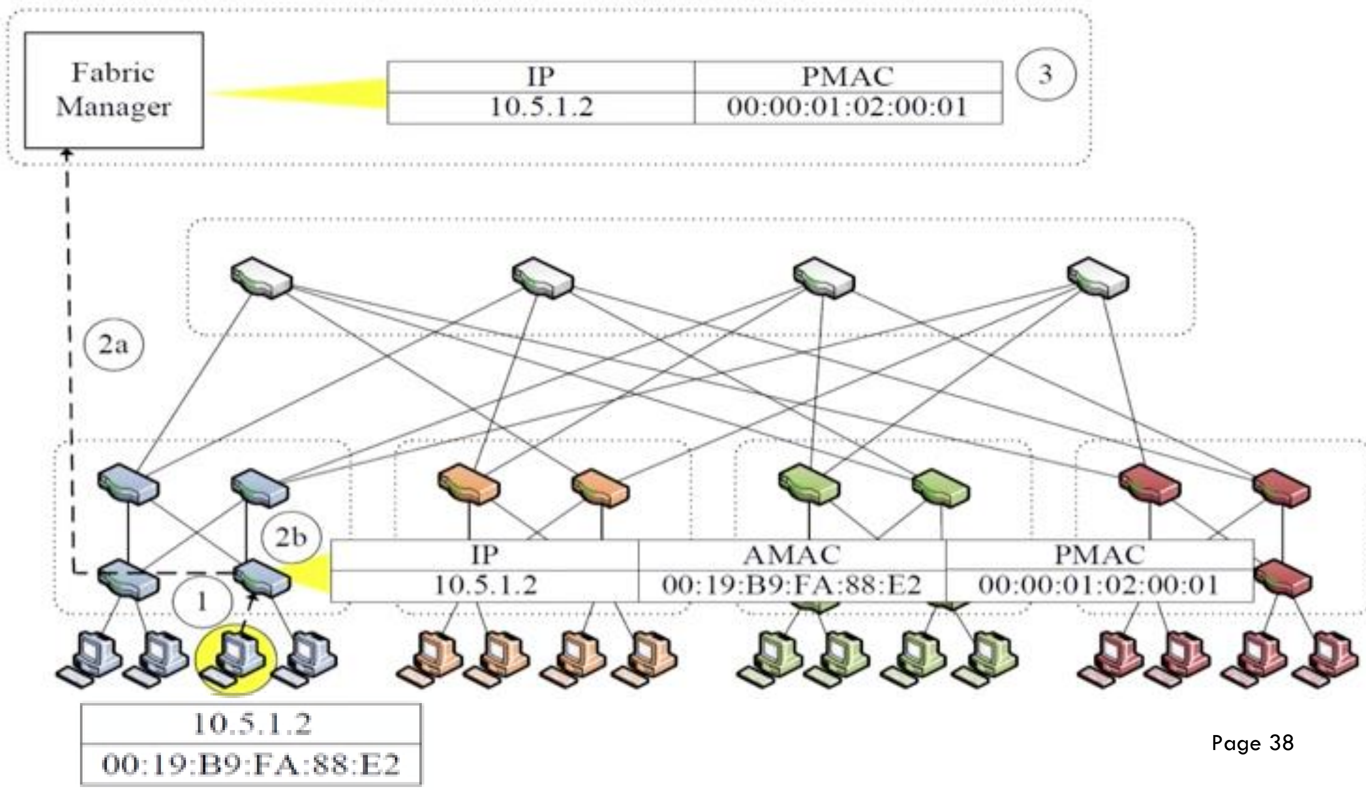
Satisfying the design Requirements

- Need for a large, layer two topology
  - Plug-and-play, minimal configuration

- Many scaling problems to solve
  - State required for layer-2 forwarding
  - Avoiding flooding (e.g., ARP request)
    - ARP: Address Resolution Protocol
  - Fast updates to addressing upon VM migration

PortLand Fabric Manager

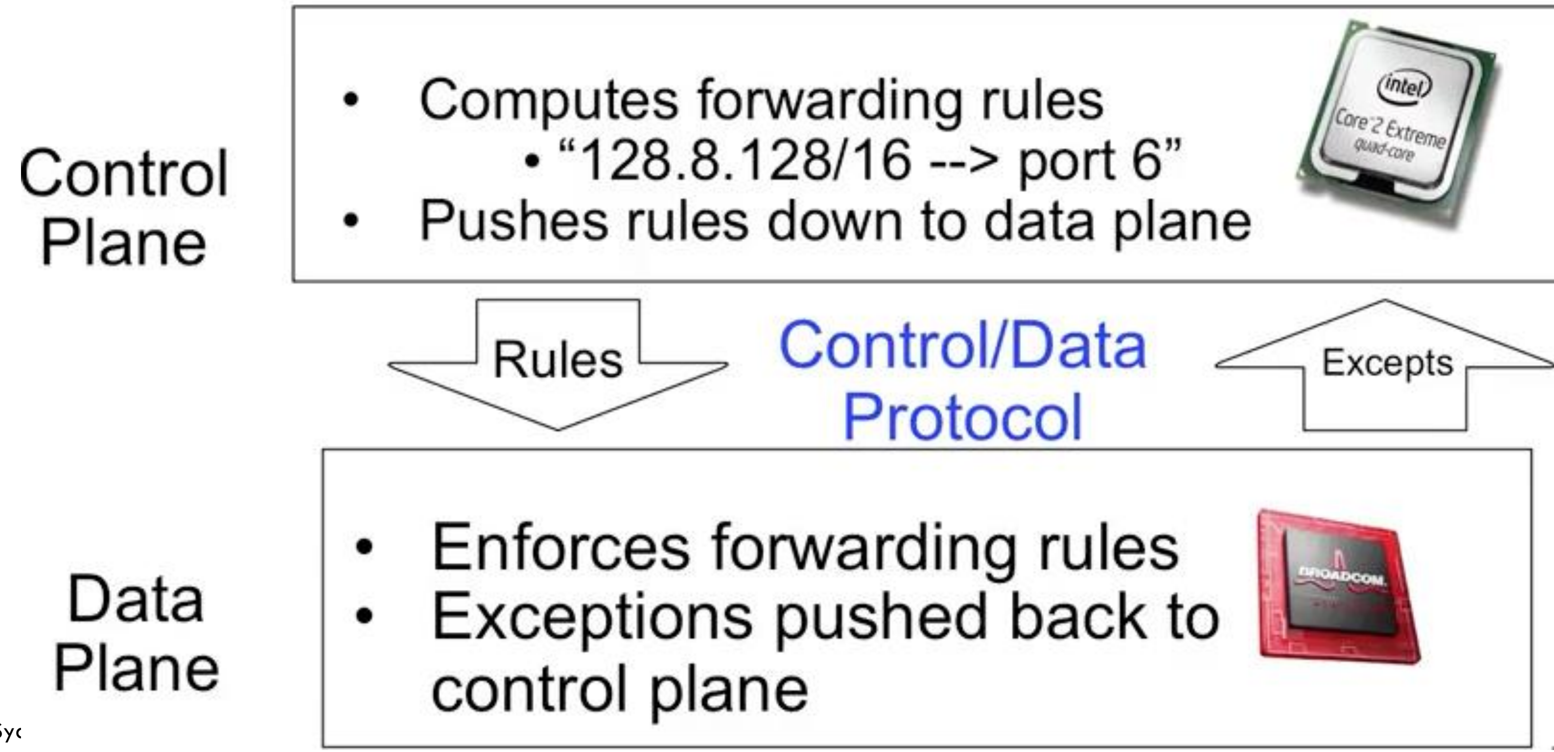
- An early SDN controller for data centers
- Logically centralized fabric manager
- Positional Pseudo MAC addresses
- Address resolution: Proxy ARP
- Forwarding based on pseudo MAC
- Efficient forwarding



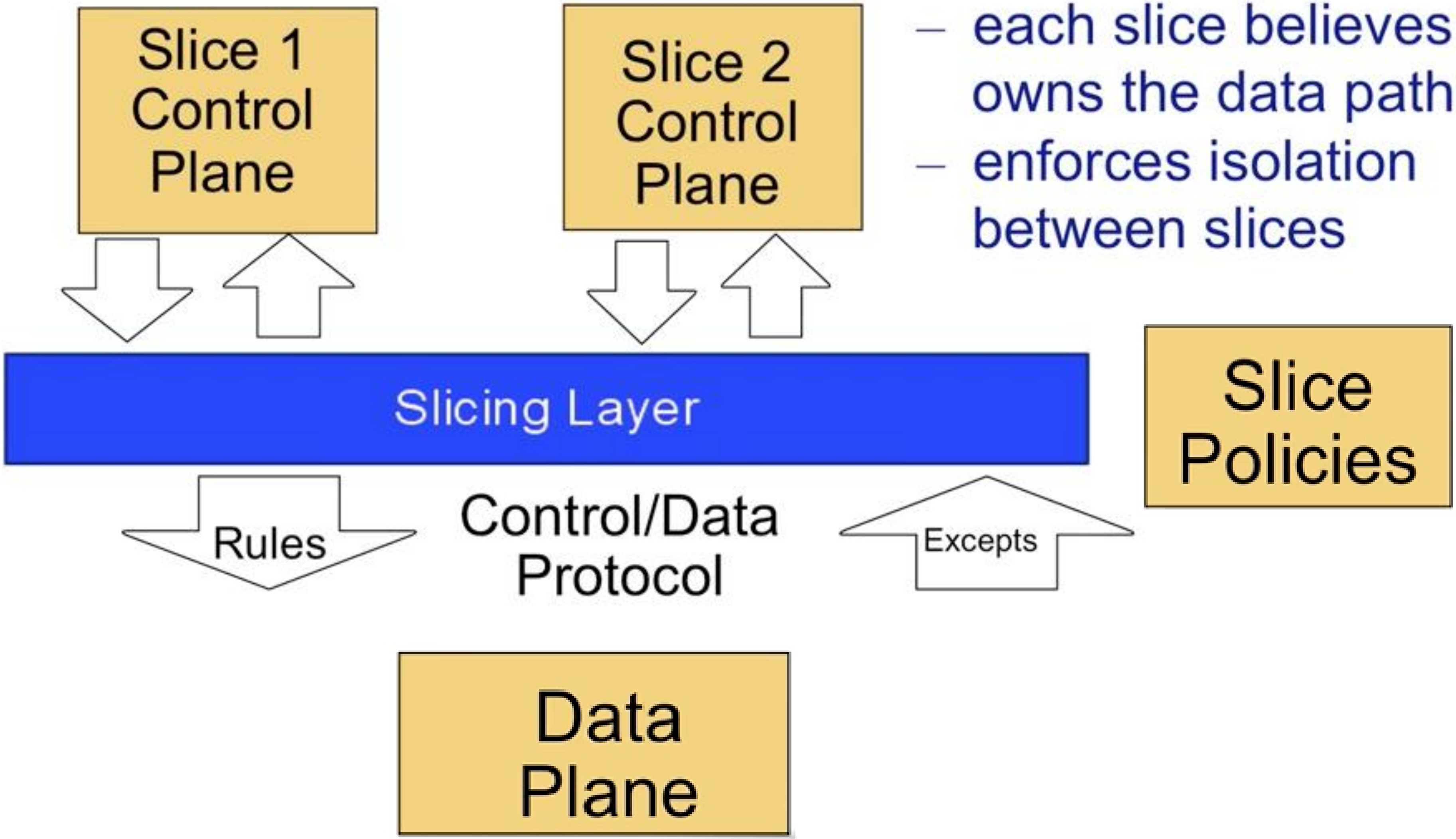
Network Slicing

– One of the enabling technologies for SDN in Data Center

– Traditional Network Device Control



Add a slicing Layer



- each slice believes it owns the data path
- enforces isolation between slices

Features of Network Slicing

- Divide the production network into logical slices
  - Each slice controls its own packet forwarding
  - Users pick which slice controls their traffic
  - Existing production services run in own slice

- Enforce strong isolation between slices
  - Actions in one slice do not affect another
- Each slice can mirror a production network
  - Production
  - Testing
  - Research

Why slice the network

- Multiple administrative groups
  - Different departments on a campus
- Multiple customers
  - Tenants in a shared data center
  - Researchers on a shared infrastructure
- Experiments vs. operational network
  - Support research without breaking real services
- Expanding a network's footprint
  - Lease components in another carrier's network
- Multiple services or applications in one domain

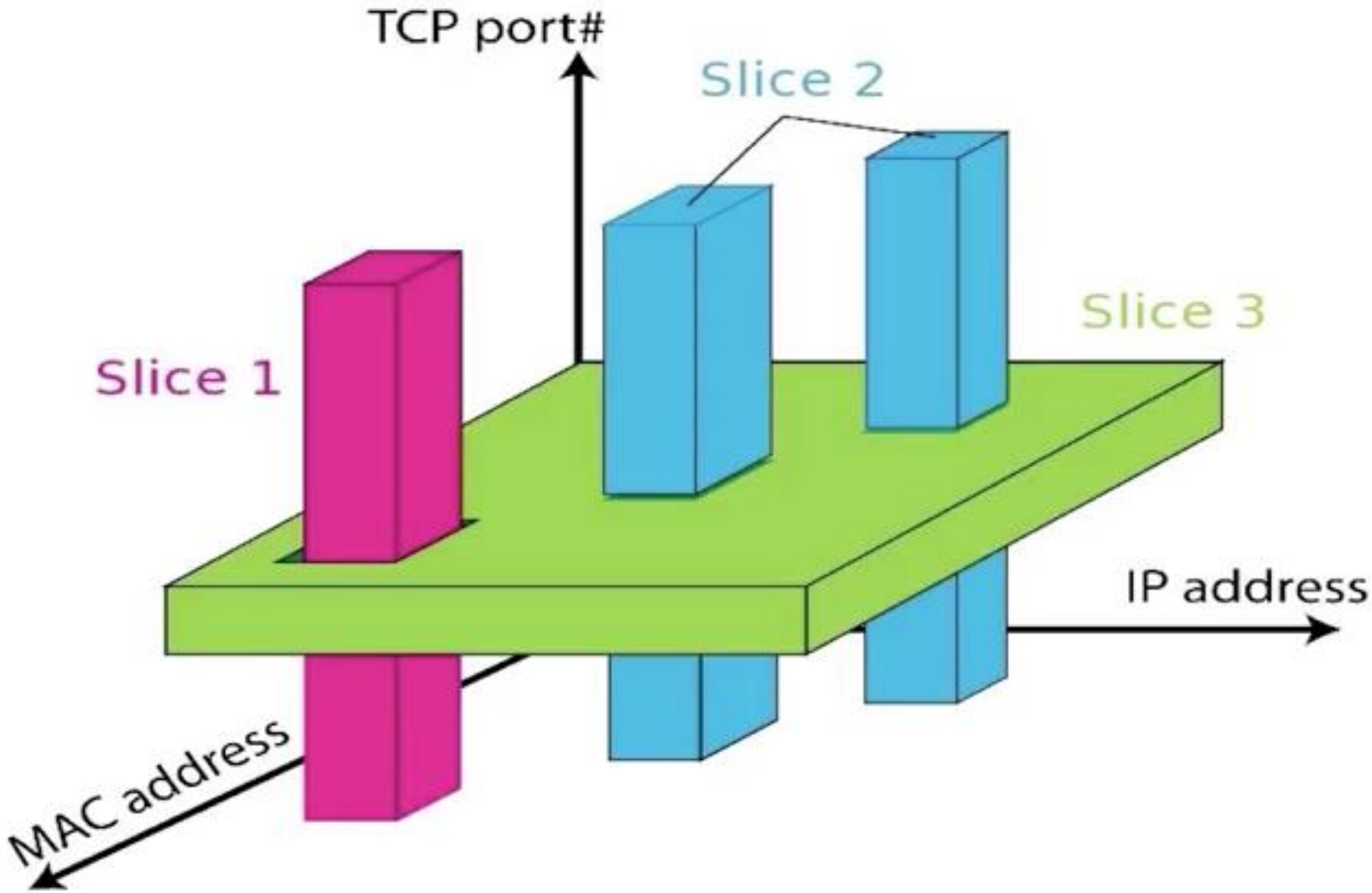
How to facilitate network slicing

- Data plane unmodified
  - No performance penalty

- Control Policy: Specify resource limits for each slice
  - Link bandwidth
  - Maximum number of forwarding rules
  - Topology
  - Fraction of switch/router CPU

Flow Space

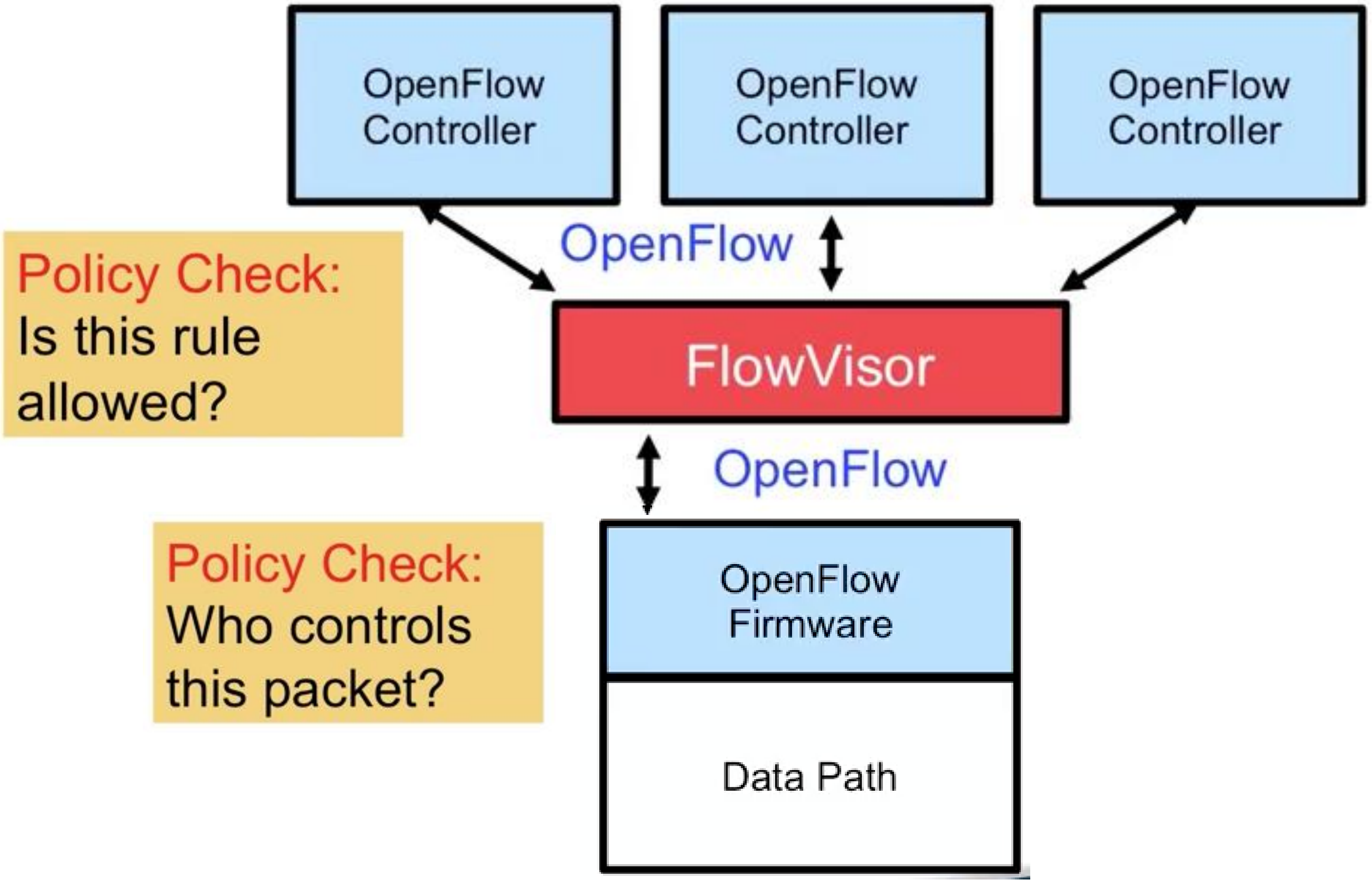
– Which packets does the slice control?



FlowVisor

- FlowVisor is an OpenFlow controller that acts as a transparent proxy between OpenFlow switches and multiple OpenFlow Controllers.
- Slicing
  - It enables any combination of switch ports (layer 1), source/destination Ethernet address or type (layer 2), src/dst IP address or type (layer 3), and src/dst TCP/UDP port or ICMP code/type (layer 4).
- Enforces isolation between each slice

FlowVisor on OpenFlow



Different ways to slice the network

- By switch port – Lower layer
  - Basically the same functionality as VLANs
  - Data center networks

- By application – higher layer (e.g., TCP port)
  - Would require some more complicated access control lists
  - Dynamism possibly a bit more difficult without SDN
  - Home networks

Multi-Tenant Datacenters

- Single physical datacenter shared by many “tenant” users
  - Customers
  - Applications/services
  - Developers

- Challenges
  - Workloads require different topologies, services
  - Address space overlaps with physical network

Multi-Tenant Datacenters architecture

- Each host in the datacenter has multiple VMs
  - Each host has a hypervisor with an internal switch
  - Switch forwards to local VM or another Hypervisor

- Need a Network Hypervisor to build right network abstractions for tenants
  - Control abstraction: Tenants define a set of logical network data plane elements that they can control.
  - Packet abstraction: packets sent by endpoints should see the same service as in a “native” network (same as physical network).

Implementation

- Implementing the abstraction
  - Network hypervisor sets up tunnels between host hypervisors
  - Physical network simply sees IP packets
  - Centralized SDN Controller configures the hosts’ virtual switches
  - Logical data path implemented entirely on the sending host
- Implementing the logical data path
  - Tunnel endpoints are virtual switches running on host hypervisors
    - Implemented with Open vSwitch
  - Controller (normally a cluster) can modify flow table entries and set up tunnels

Controller Structure

- Hypervisor and physical gateways provide the controller with location and topology information
- Service providers configure the controller
- Forwarding state pushed to Open vSwitches via OpenFlow

Challenges

- Making the software switching at end hosts fast
  - Exact-match flows in kernel
  - User-space program matches on full flow table, installs exact match in the kernel
  - Future packets for the same flow are matched in-kernel
- Scaling controller computation
  - Two-layer distributed controller
    - Logical controllers: Compute flows and tunnels for logical datapaths
    - Physical controllers: Communicate with hypervisors, gateways and service nodes
  - Logical controller avoids dealing with the full mesh of tunnels

Role of SDN and NV in Data Center

- Network Virtualisation != SDN
  - Predates SDN, and doesn't require SDN
- Easier to virtualize an SDN switch
  - Run separate controller per virtual network
  - Partition the space of all flows
  - Use open interface to the hardware
- Network virtualisation can also use software switches
  - NV can be deemed as a Killer App of SDN

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