UE PRIP

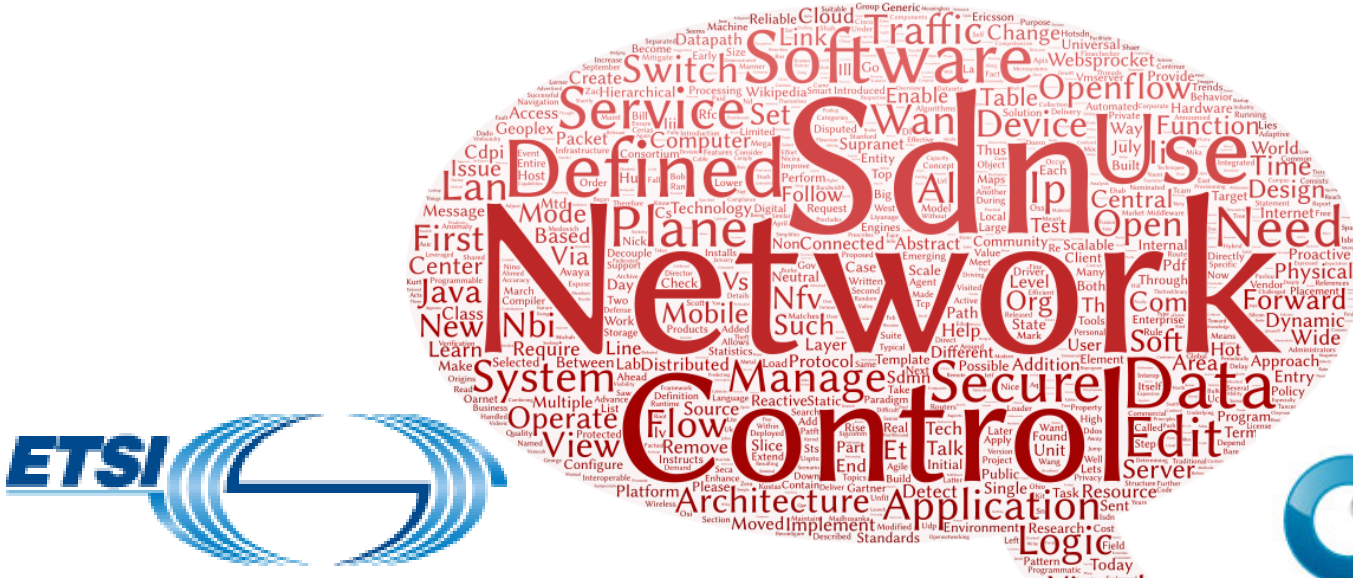
Principes des réseaux informatiques par la pratique

Introduction to new networking paradigms: SDN and NFV

Isabel Amigo

2022

The topics covered today...



Lecture's Objectives

1. Introduce future networking paradigms and concepts
2. At the end of the lecture you should be able to respond to the following questions regarding these concepts:
   - Why? What? How? Who?
   - Are SDN and folks disruptive technologies?

Structure

SDN drivers and overview

SDN

OpenFlow

A word on NETCONF/YANG

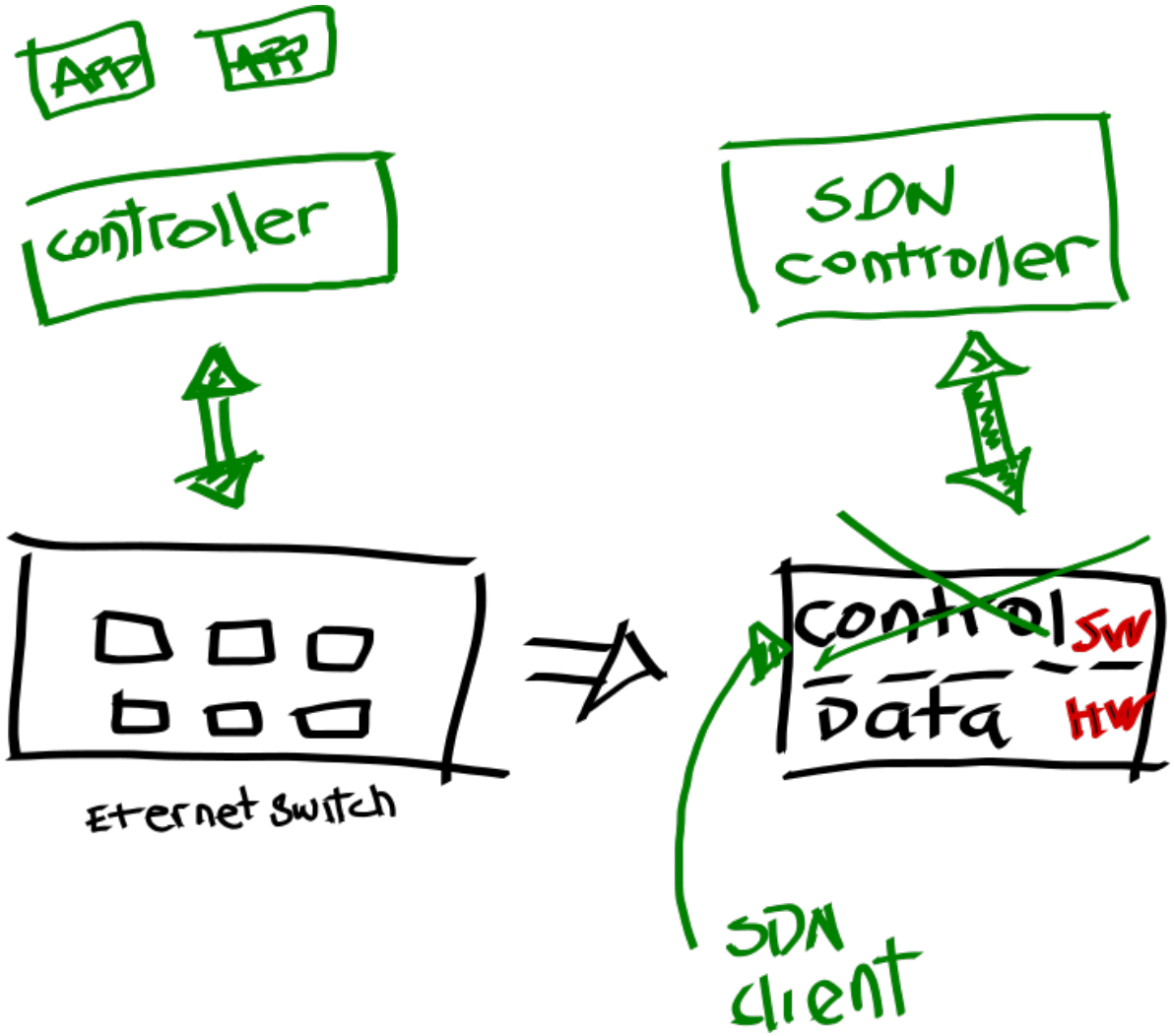
NFV

Current State of SDN

Conclusion

Traditional Node

SDN Node


App
App
controller
SDN
controller
control
data
SW
HW
Ethernet switch
SDN
client

Recall

Management Plane

- Configuration
- Monitoring

Control Plane

- Establishing the state in routers
- Determines how and where packets are forwarded
- Slow time-scale (per control event)

Data Plane

- Processing and forwarding packets
- Based on state in routers and endpoints
- Per-packet timescale (fast!)

SDN DRIVERS

Why do we need new paradigms?

Why have networks remain almost the same for years?

- Difficult to innovate
- Expensive equipments
- Closed systems

Driver 1: Datacenters (DC) and Virtualization

- The trend is to virtualize, why?
- Virtualization acceleration on DCs, with the underutilization of servers
- Multi-tenant DCs $\Rightarrow$ VMs
- Need of isolated networks and to reconfigure if VM migrates $\rightarrow$ VLANs, VXLANs

Can't we have something more agile to programmatically construct and dynamically change the logical network infrastructure?

Driver 2: The need of abstractions

Abstraction $\Rightarrow$ Interfaces $\Rightarrow$
Modularity

Imagine...

- A computer for which you can only develop programs using vendor-dependent commands?
- Having to configure every element of the system (memory, disk etc.)

We need abstractions in networks, decouple the problem, make it modular

Current networking abstractions

Layers! ☺
but they only deal with the data plane ☹

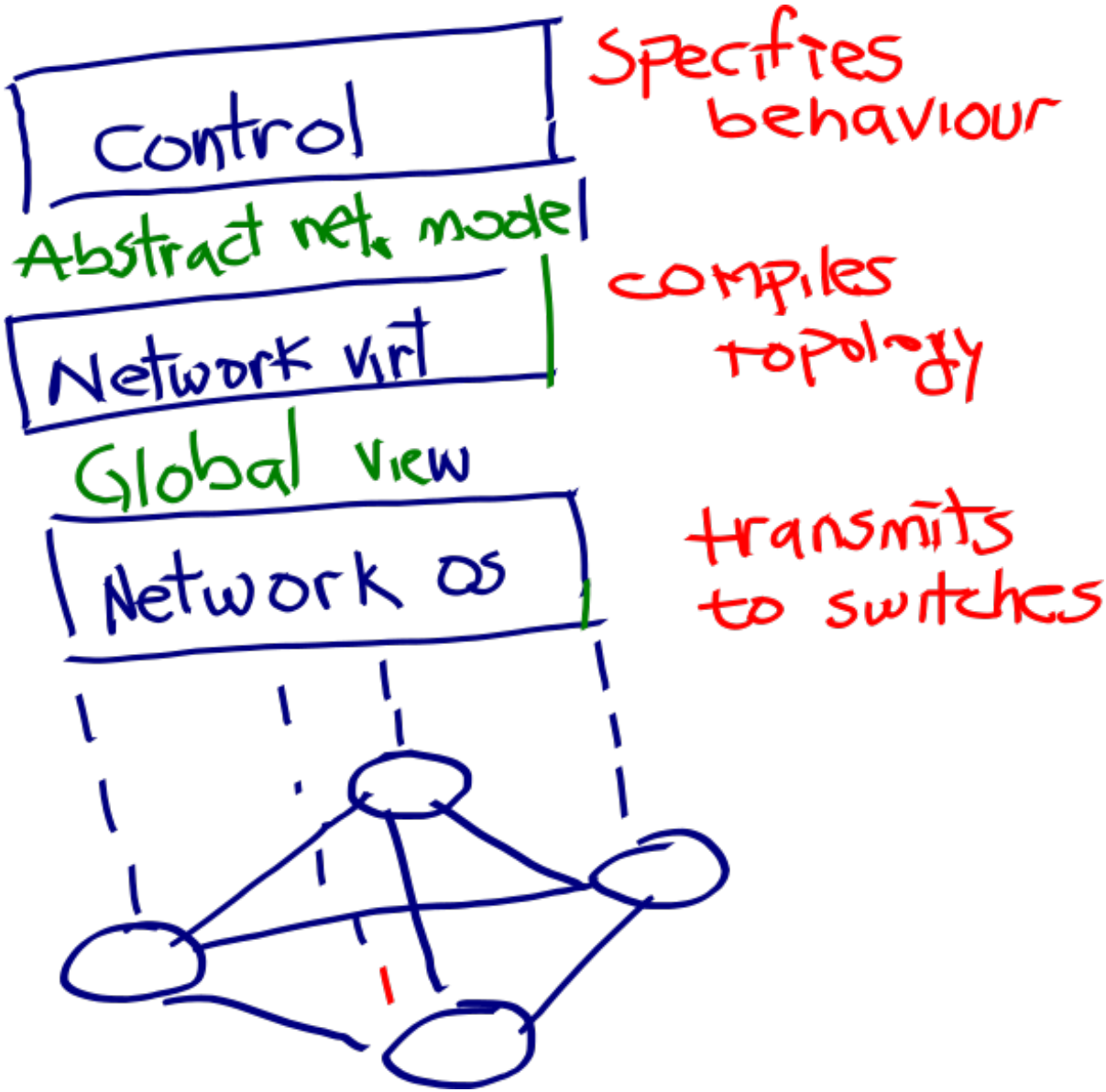
What abstractions do networks have related to control plane?

None!

Abstractions in the genesis of SDN

1. Global network view
   - Instead of distributed protocols, configuration is Function(view)
2. Simplified network view
   - Control specifies goals, doesn't configure physical devices
3. Forwarding
   - Communication of control program with forwarding circuits (HW)

Initial SDN Architecture proposal



Driver 3: An existent tendency

Tendency to separate control and data planes already existed:

- MPLS edge routers computing the path
- Control and Data plane in routers already separated... but within the same chassis
- Always been like that in telephony networks

SDN Genesis

- ...
- 2007 Ethane (Martin Casado *et al.* [2])
  - Part of Clean Slate program: *What would the Internet look like in 15 years if we restarted from a clean slate?*
  - Main contributions: Centralized controller and OpenFlow
- 2011 ONF is created, to standardize OpenFlow
- May 2011 Marvell and Larch Networks announced the availability of an OpenFlow-enabled switch
- ...

ONF: Open Network Foundation [5]

- *Open Networking Foundation (ONF) is a user-driven organization dedicated to the promotion and adoption of Software-Defined Networking (SDN) through open standards development.*
- Founding members Deutsche Telekom, Facebook, Google, Microsoft, Verizon, and Yahoo!

ONF members

Mainly vendors and Telcos.

Membership Levels

ONF is funded primarily through membership dues. Any operator or supply chain organization that shares the goals of the ONF is invited to join.

PARTNERS

Partner members share ONF's vision and mission and seek to closely align with our strategy and execution. Partners lead and influence the entire ONF ecosystem, leverage our portfolio of open source platforms and solutions, and actively work with ONF and project teams to transform their businesses and the industry at large.

MEMBERS

Member organizations often focus on 'consuming' a specific ONF project. Membership helps support ONF's work and engineering team, and in turn ONF is resourced to help educate, support and promote members to help ensure their success.

x9 among which: Intel, Google, AT&T,Deutsche Telekom

x110 among which Cisco, Dell, Ericsson, Nokia, Microsoft

+ several collaborators

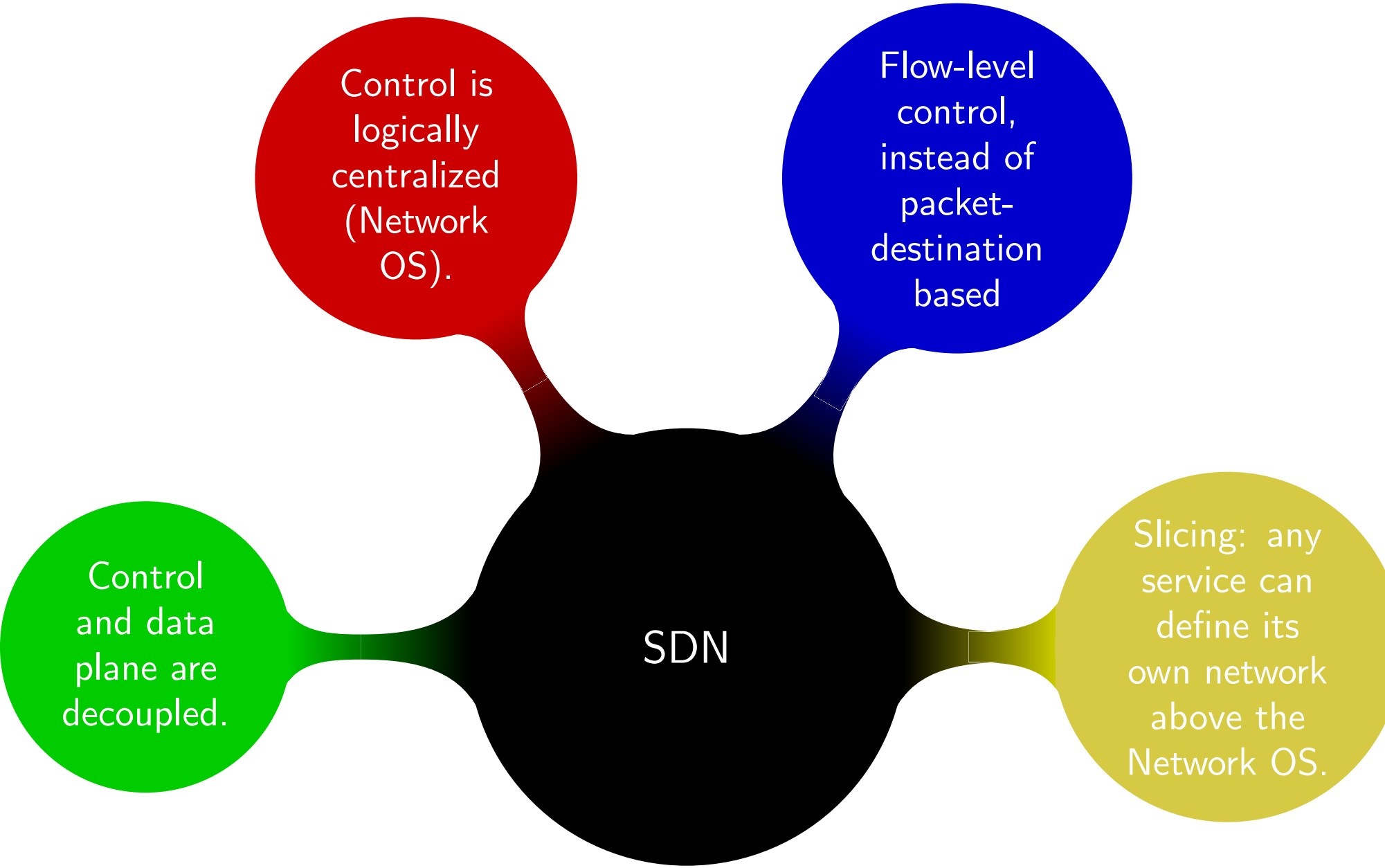
As for March 2022

So what is SDN?

SDN principles

- Separate control and data plane
- Control plane executable in commodity HW
- Programmable data plane

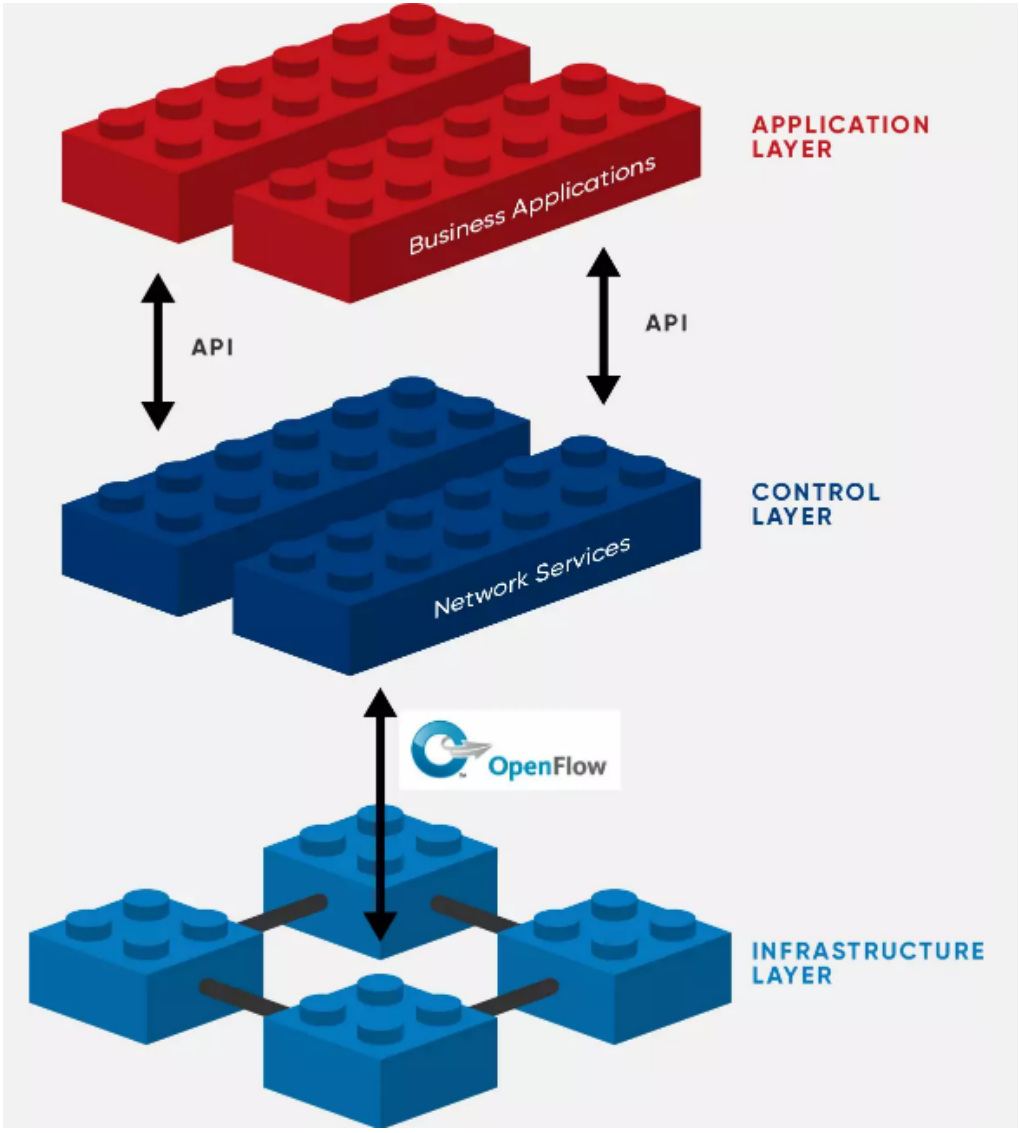
The four key components of SDN



SDN Wish list

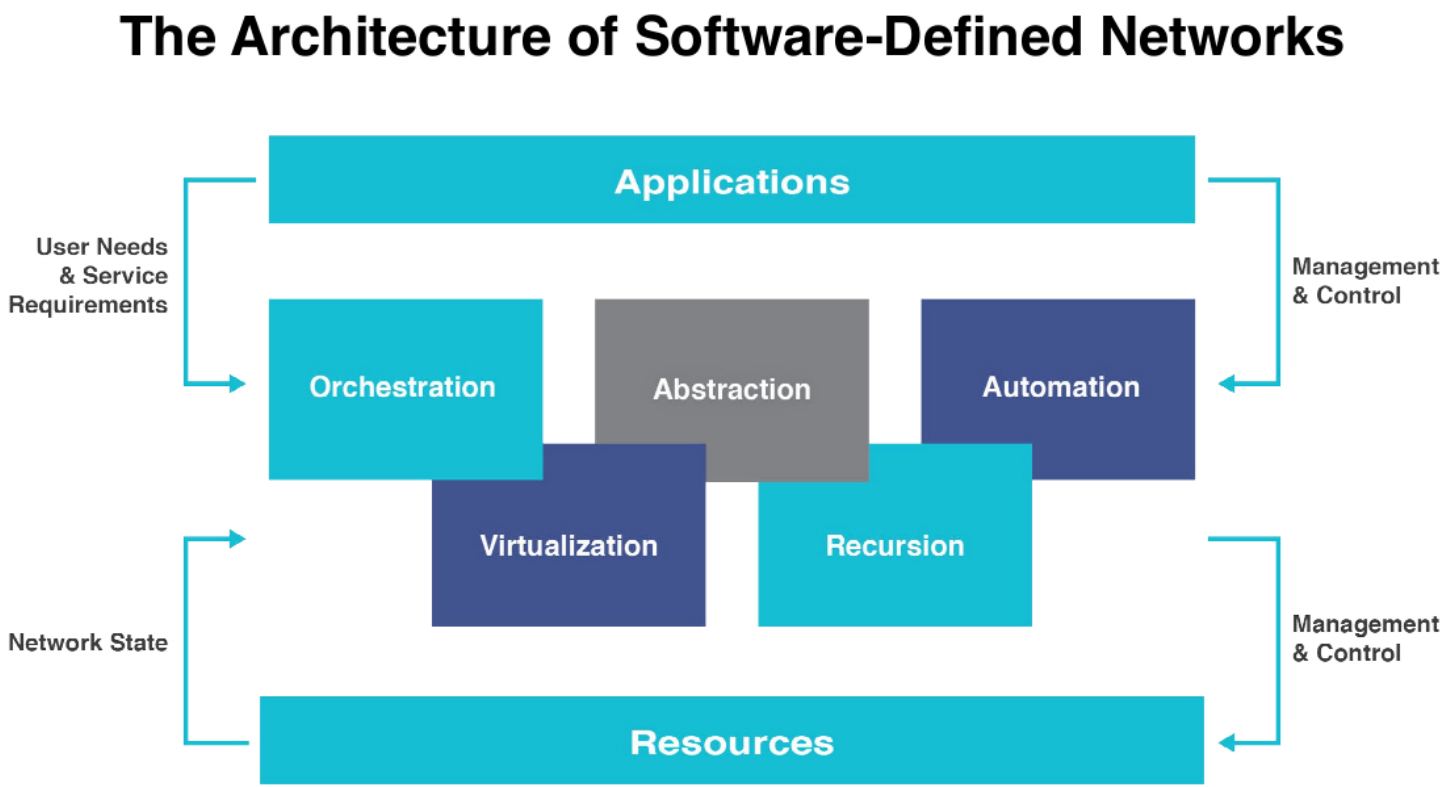
- Facilitate innovation
- Allow experiments and research without the need of expensive equipments (reduce CapEx)
- Flexibility
- Fast upgrades
- Increase speed to market
- Reduce OpEx

ONF Proposed Architecture



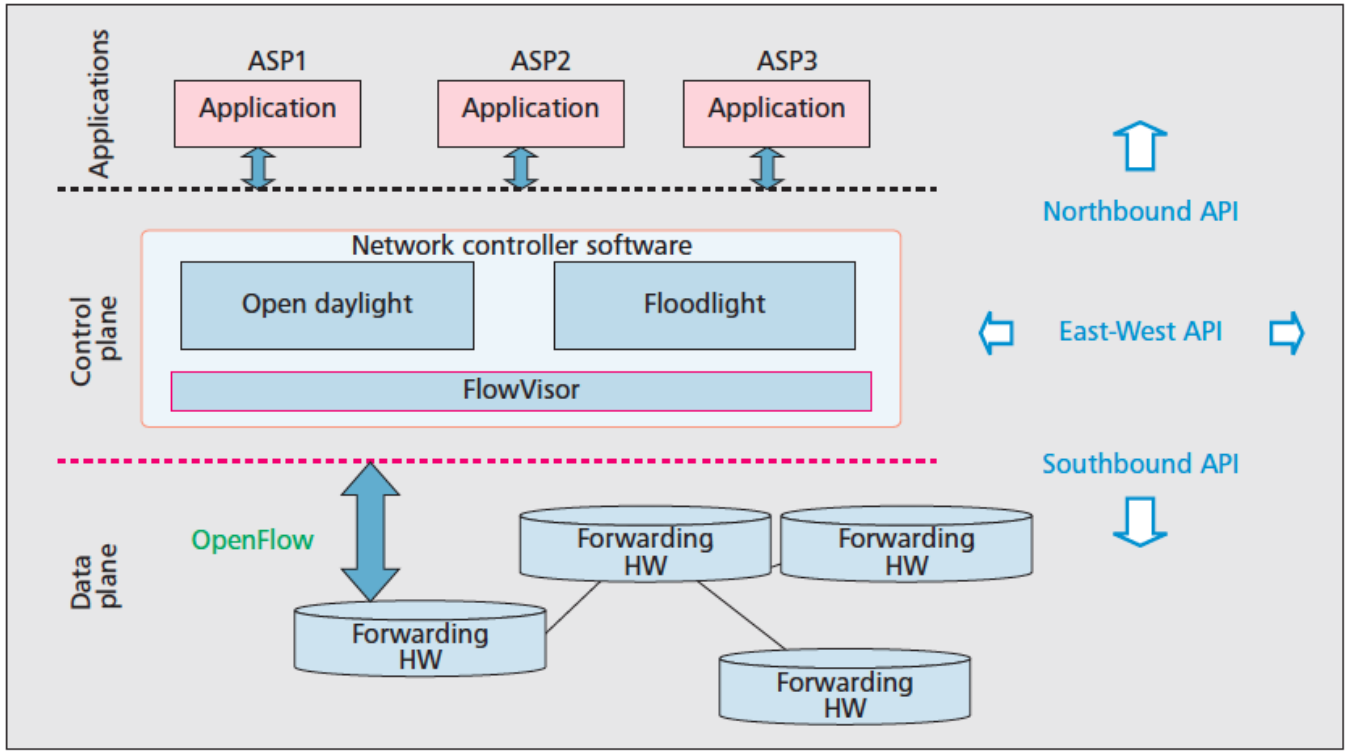
Source: `https://opennetworking.org/sdn-definition/`

ONF Proposed Architecture



Source: SDN Architecture - A Primer, White Paper, ONF, 09/2016

Commonly used reference architecture



Software-Defined Networking, What is it? Several attempts of definitions

Def. 1 (Wikipedia)

SDN is an approach to computer networking that allows network administrators to programmatically initialize, control, change, and manage network behavior dynamically via open interfaces and abstraction of lower-level functionality.

Software-Defined Networking, What is it? Several attempts of definitions

Def 2 (ONF)

What is SDN? The physical separation of the network control plane from the forwarding plane, and where a control plane controls several devices.

Software-Defined Networking, What is it? Several attempts of definitions

The OpenDayLight presentation

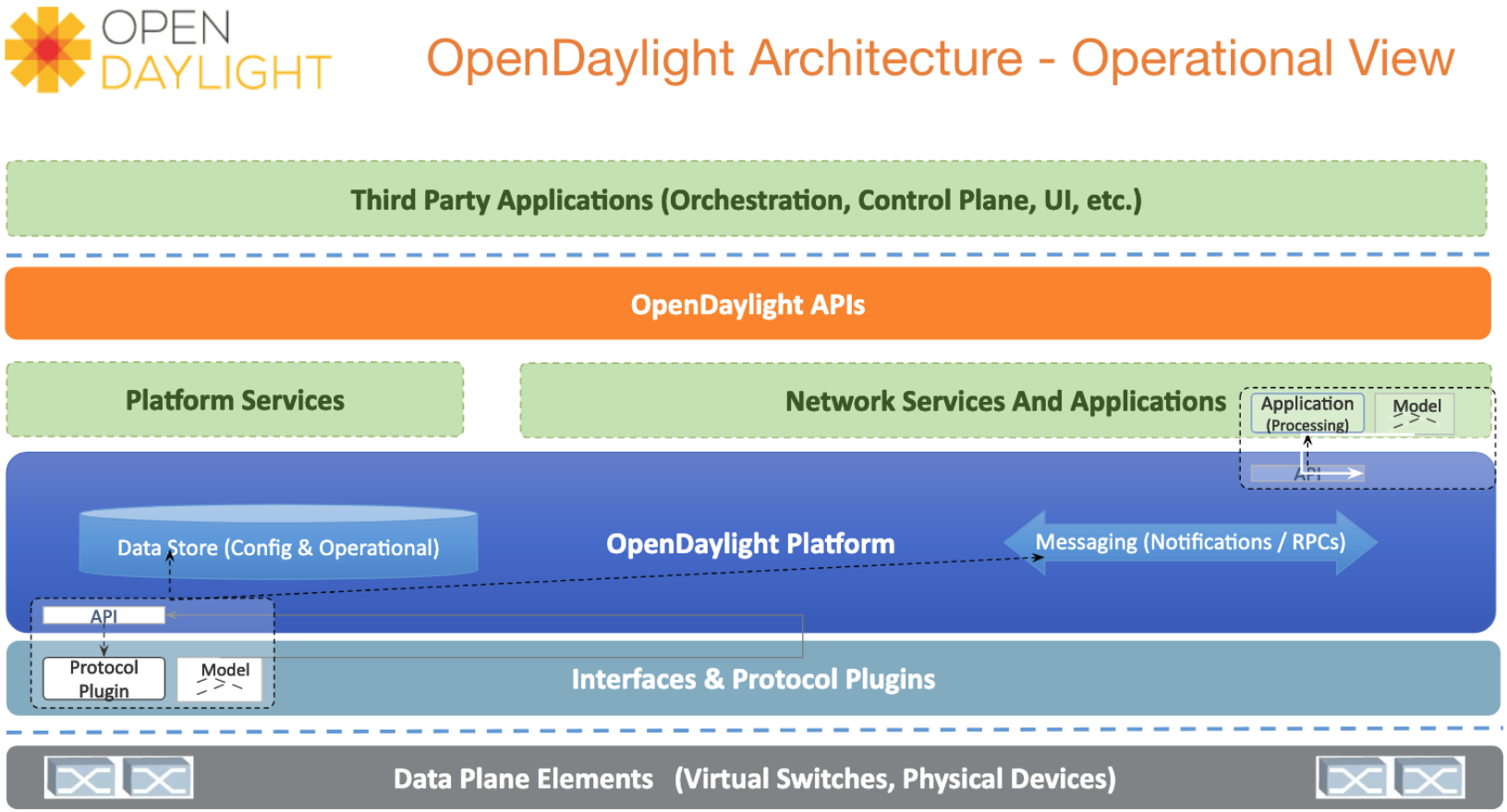
The modern software-defined networking (SDN) movement grew out of a simple question: why shouldn't networking devices be programmable just as other computing platforms are?
The benefits of such an approach were obvious: no more arcane protocols to learn. No more waiting and hoping for networking vendors to develop specialized features you need. And if you could develop your own features, you could then optimize your device selection for price and performance independently of feature-richness.

Standardization Bodies for SDN-related protocols

- ONF: Open Network Foundation
- IETF
- NFV/ETSI
- ...

OpenDayLight

In OpenDayLight words: the most widely deployed open source SDN controller platform(in 6 years, 10 releases, 1000+ authors/submitters, 100K+ commits, powers networks of 1B+ global subscribers).



What is OpenFlow? Is it an essential building block of SDN?

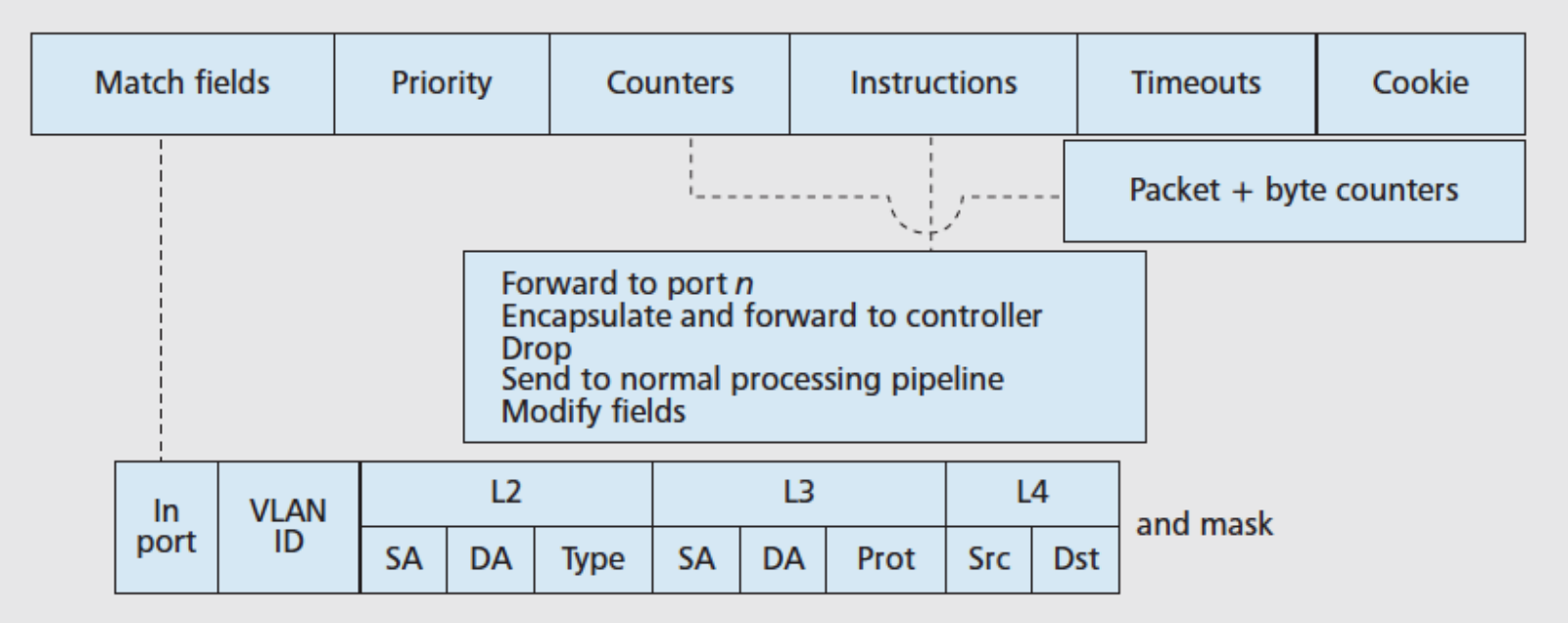
OpenFlow

- *First standard communications interface defined between the control and forwarding layers of an SDN architecture.*
- Standardised by ONF

OpenFlow?

- Instructions available for programming network devices
- Allows communication between control and data plane
- Forwarding: determined by flow tables
- Flow tables are filled-in by the controller

OpenFlow Forwarding- One entry on one table



Some simplified examples

Switching

Switch Port	MAC src	MAC dst	Eth type	VLAN ID	IP Src	IP Dst	IP Prot	TCP sport	TCP dport	Action
*	*	00:1f:..	*	*	*	*	*	*	*	port6

Flow Switching

Switch Port	MAC src	MAC dst	Eth type	VLAN ID	IP Src	IP Dst	IP Prot	TCP sport	TCP dport	Action
port3	00:20..	00:1f..	0800	vlan1	1.2.3.4	5.6.7.8	4	17264	80	port6

Firewall

Switch Port	MAC src	MAC dst	Eth type	VLAN ID	IP Src	IP Dst	IP Prot	TCP sport	TCP dport	Action
*	*	*	*	*	*	*	*	*	22	drop

Some simplified examples

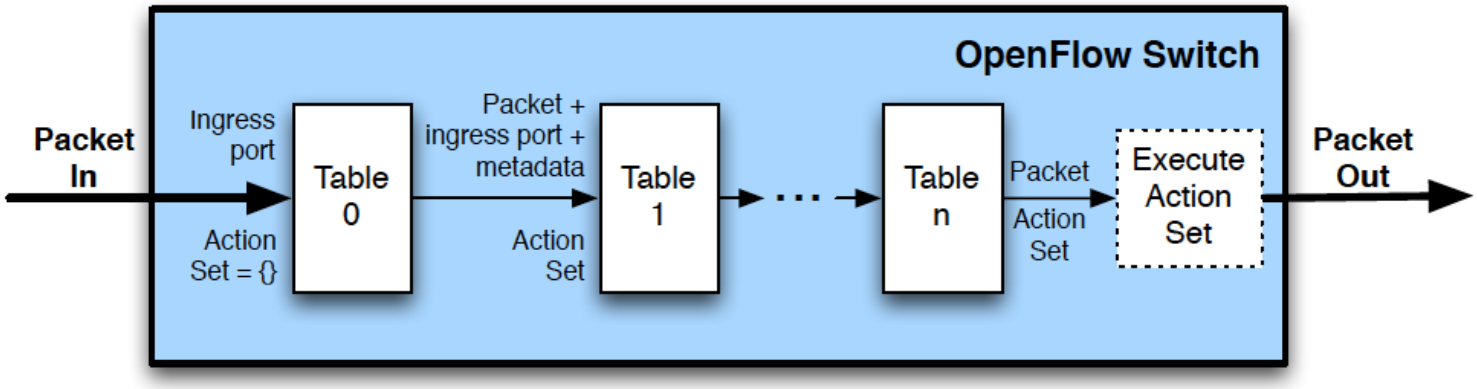
Routing

Switch Port	MAC src	MAC dst	Eth type	VLAN ID	IP Src	IP Dst	IP Prot	TCP sport	TCP dport	Action
*	*	*	*	*	*	5.6.7.8	*	*	*	port6

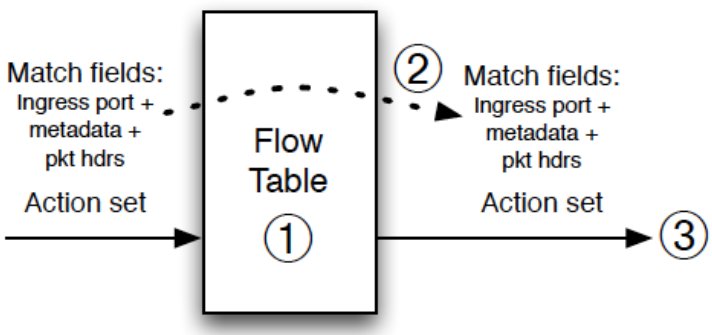
VLAN Switching

Switch Port	MAC src	MAC dst	Eth type	VLAN ID	IP Src	IP Dst	IP Prot	TCP sport	TCP dport	Action
*	*	00:1f..	*	vlan1	*	*	*	*	*	port6, port7, port9

OpenFlow Forwarding-Pipeline



(a) Packets are matched against multiple tables in the pipeline



① Find highest-priority matching flow entry

② Apply instructions:
- i. Modify packet & update match fields (apply actions instruction)
- ii. Update action set (clear actions and/or write actions instructions)
- iii. Update metadata

③ Send match data and action set to next table

(b) Per-table packet processing

OpenFlow Actions

Output	identifies the output port (i.e.: interface) for the packet
Set-Queue	identifies a queue within the output port
Group	actions defined for that group must be applied
Push-Tag/Pop-Tag	modifies VLAN id or MPLS tags
Set-Field	modifies values in header (e.g. DSCP, dst IP,..)
Change-TTL	modifies TTL value (IPv4/MPLS) or hop limit (IPv6)
Drop	if no action provided, then the action is drop

Why multiple tables?

Example:

- table 0 identifies the output interface, according to dst. address
- table 1 identifies the scheduling policy according to the value of DSCP

Defining sets of actions across multiple tables provides a more modular configuration

OpenFlow Protocol - Messages

- Messages Header; version (8 bytes), Type(8 bytes) Length (16 bytes)
- 30 type of messages, among which:
  - Hello
  - Packet-In
  - Packet-out
  - Set-config
- Messages can belong to one of three categories:
  - Controller-to switch
  - Asynchronous
  - Symmetric

So OpenFlow...

- Communication protocol controller $\leftrightarrow$ OpenFlow switch
- Runs over TCP
- Security: TLS (non mandatory, but encouraged!)
- Switching based on flows and tables
- Current specification version 1.5.1 (May 2015), 1.6 (September 2016)
- Probably being to be replaced with a more flexible alternative

OpenFlow is not the only possible southbound interface

Other southbound interfaces:

- NETCONF
- XMPP
- Open vSwitch Data Base Management Protocol (OVSDB)
- Cisco's onePK

What is NETCONF/YANG? Is it SDN-oriented?

NETCONF/YANG

NETCONF: (Network Configuration Protocol)

- provides mechanisms to install, manipulate, and delete the configuration of network devices
- standardised by IETF
- can be seen as another southbound SDN interface
- usually used along with YANG

YANG (Yet Another Next Generation)

- a data **modelling language**
- provides a **standardized way** to model the operational and configuration data of a network device

NETCONF/YANG provides a standardized way to programmatically update and modify the configuration of a network device

NETCONF vs OpenFlow?

OpenFlow was the first SDN southbound standard, but no longer the only one/preferred one.

Are there in contradiction? Not really...

- NETCONF is a protocol that allows to modify networking device's configuration
- OpenFlow is a protocol that allows to modify its forwarding table

Is NETCONF/YANG SDN-friendly?

- NETCONF does not really separate control from data plane, why?
- but solves some of existing problems we've already seen, which ones?

What is NFV?

Is it the same as SDN? Are they concurrents?

NFV: Network Function Virtualization

Def. (wikipedia)

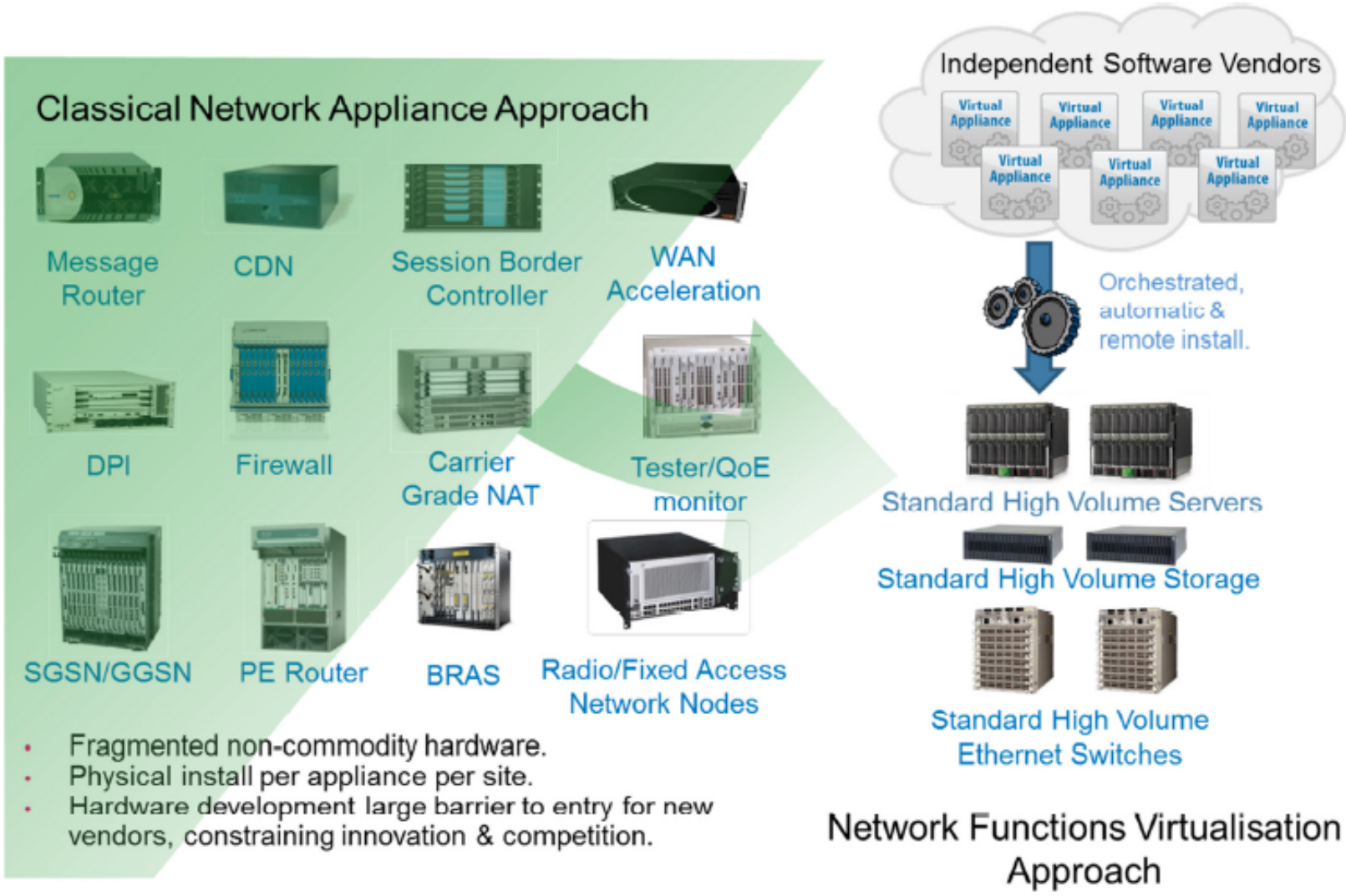
is a network architecture concept that uses the technologies of IT virtualization to virtualize entire classes of network node functions into building blocks that may connect, or chain together, to create communication services.

NFV

Def. (ETSI [3])

NFV transforms network architectures through the implementation of network functions in software that can run on a range of industry standard server hardware...

NFV: from specialized closed systems to network functions implemented in white boxes



In NFV, Network functions are:

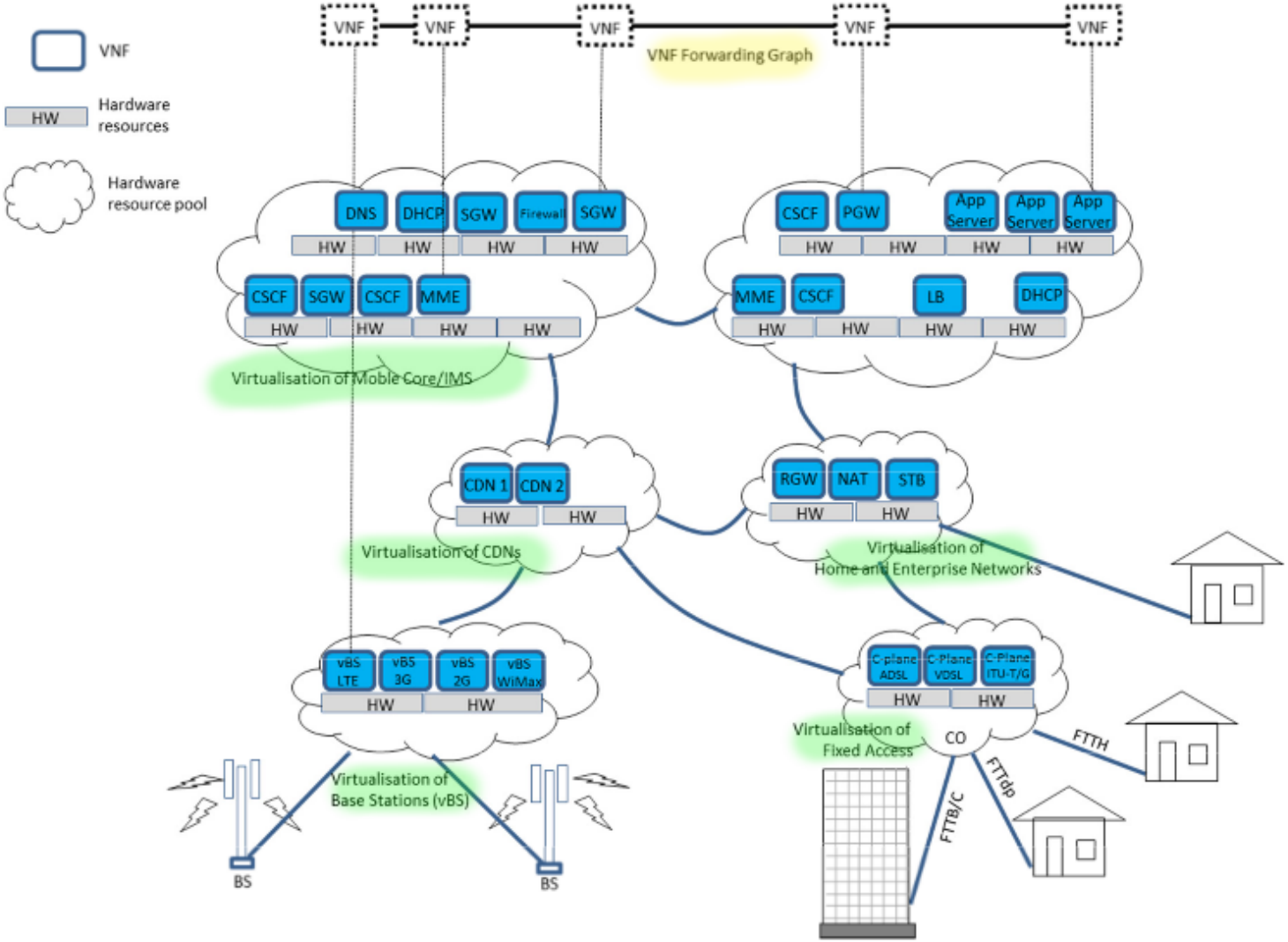
- Written in software
- Deployed within data centers or in “white boxes” programmed as network nodes
- Objectives: agility, easy modification, service oriented networking

we have thus virtual network functions (VNFs)

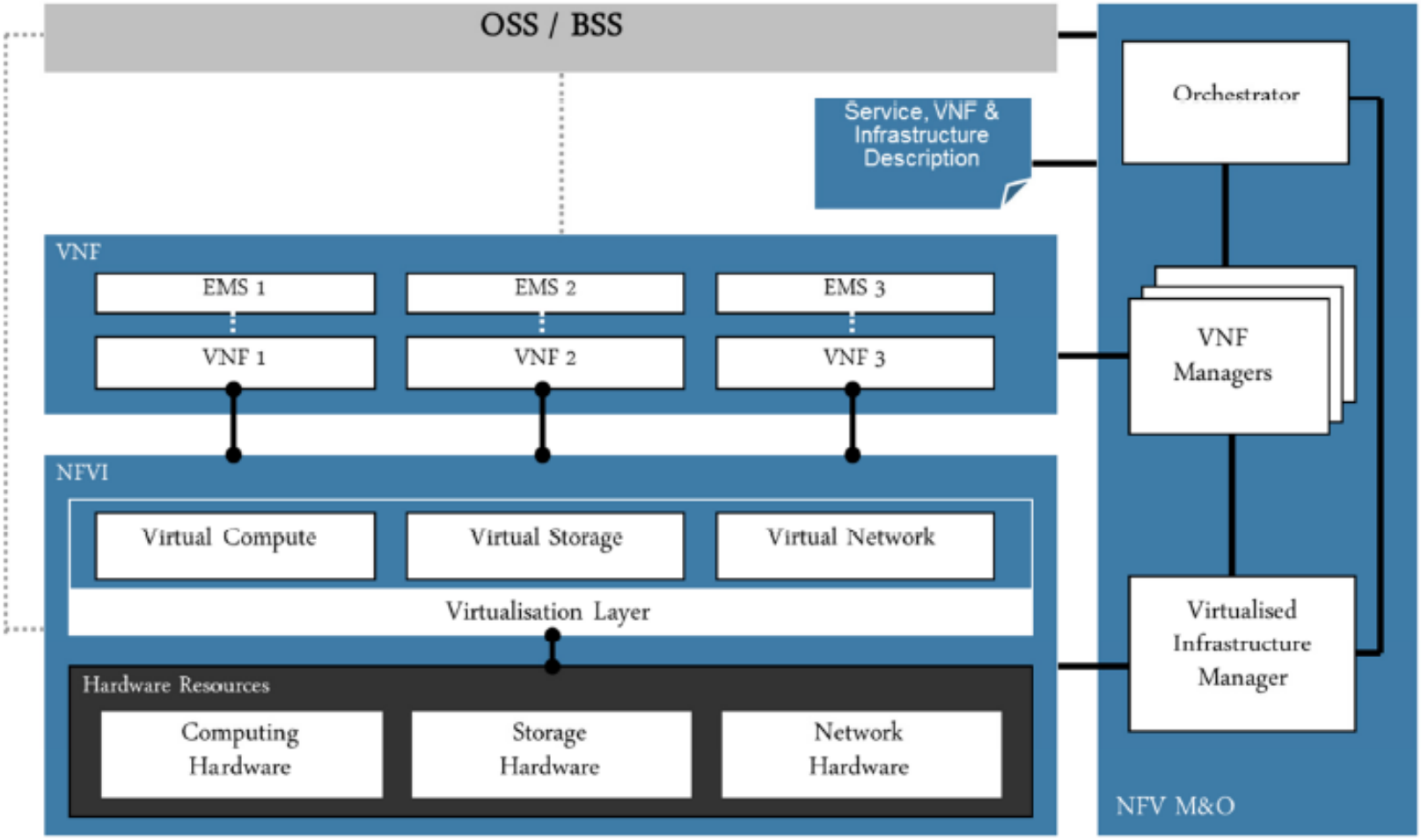
VNFs

- Analogy with cloud computing
- Virtualization of Network Functions
- Use cases: Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS), Network as a Service (NaaS)

VNFs examples



NFV architecture proposed by ETSI



Source: `https://portal.etsi.org/NFV/NFV_White_Paper2.pdf`

NFV and SDN can be independent but are complementary

- NFV alone: VNFs on VMs running on commodity servers interconnected by standard networking
- SDN alone: Network Functions on dedicated hardware interconnected through an SDN controller
- NFV + SDN: VNFs on VMs running on commodity servers interconnected through an SDN controller

SDN Open questions/issues

- Scalability?
- Redundancy?
- Controller placement?
- Controller-controller interface? not standardized yet

Are there real products and deployments?

SDN Controllers

- There are several software controllers
- Some of them: OpenDayLight, ONOS, floodlight ...
- Others more academic POX, Ryu, ...
- Vendors such as Cisco have also their controller solutions

OpenFlow Switches

- There are commercial switches supporting OpenFlow
- Vendors including Cisco, Juniper, Big Switch Networks, Brocade, Arista, Extreme Networks, IBM, Dell, NoviFlow, HP, NEC, among others
- But vendors also sell their own SDN solutions based on their own abstractions and interfaces

Some ONF active projects

- CORD (Central Office Re-architected as a Datacenter)
- ONOS (Open Network Operating System) is an SDN controller
- Mininet
- SD-RAN SDN for 5G Radio Access Network

Google's SDN deployment: A well-know success story [4]

- Private WAN connecting Google's data centers across the planet
- SDN architecture using OpenFlow switches
- Before: long convergence time for setting up MPLS tunnels
- After: centralized traffic engineering service drives links to near 100% utilization
- Splitting application flows among multiple paths to balance capacity against application priority/demands

SDN Adoption

SDN has seen wide adoption:

- across data centers (64%),
- WANs (58%),
- and access networks (40%).

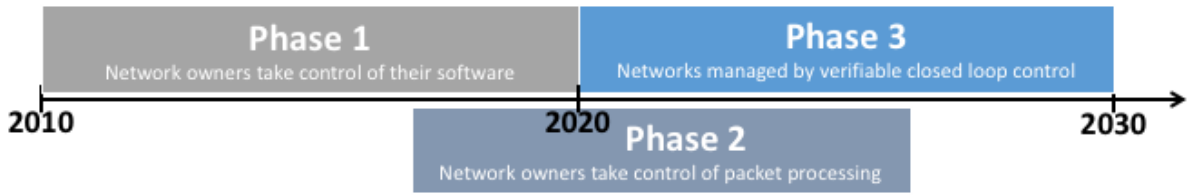
As for the "2020 Global Networking Trends report" [1].

So SDN and folks: Disruptive or Accommodating Technologies?

Conclusion

- New networking paradigms, key words: Programmability, agility, virtualisation, abstraction
- Great interest of the industry (some adoptions + members of standardization groups)
- Earliest adoptions in the Datacenters
- Two killing use-cases
  - SD-WAN
  - 5G network slicing
- Further adoption?
- will SDN controller become a new vendor lock-in product?

Future of SDN



Projecting into the future, with Phase 3 of SDN focusing on verifiable, top-down control of network behavior [6].

Conclusion

Whats next in networking?

- Intent-based networking
- Self-driven networks
- 'From Automated to Autonomous Networks' (as by BikasKoley, Google)
- The Rise of Network as a Service (NaaS) (as by '2022 Global Networking Trends Report', Cisco)

References I

[1] Cisco 2020 Global Networking Trends.
Technical report, 2020.

[2] Martin Casado, Michael J. Freedman, Justin Pettit, Jianying Luo, Nick McKeown, and Scott Shenker.
Ethane: Taking Control of the Enterprise.
SIGCOMM Comput. Commun. Rev., 37(4):1–12, August 2007.

[3] ETSI; European Telecommunications Standards Institute ISG NFV.
http://www.etsi.org/technologies-clusters/technologies/nfv.

[4] Sushant Jain, Alok Kumar, Subhasree Mandal, Joon Ong, Leon Poutievski, Arjun Singh, Subbaiah Venkata, Jim Wanderer, Junlan Zhou, Min Zhu, Jon Zolla, Urs Hölzle, Stephen Stuart, and Amin Vahdat.
B4: Experience with a Globally-deployed Software Defined Wan.

References II

SIGCOMM Comput. Commun. Rev., 43(4):3–14, August 2013.

[5] ONF: Open Networking Foundation.
https://www.opennetworking.org.

[6] Larry Peterson, Carmelo Cascone, Brian O'Connor, Thomas Vachuska, and Bruce Davie.
Software-Defined Networks: A Systems Approach.

Acronyms

SDN	Software-defined networking
NFV	Network Function Virtualization
VNF	Virtual Network Function
ONF	Open Networking Foundation
NOS	Network Operating System
ODL	OpenDaylight
XMPP	Extensible Messaging and Presence Protocol
NETCONF	Network Configuration Protocol
VXLAN	Virtual eXtensible LAN
CDN	Content Delivery Network