FÓRUNS PERMANENTES

CIÊNCIA, TECNOLOGIA E INOVAÇÃO

Arquivamento de documentos em nuvem (cloud)

30 de outubro de 2014 - Centro de Convenções da Unicamp

Software Defined Networking & Network Functions Virtualization (Redes controladas por software)

Prof. Christian Esteve Rothenberg
FEEC/UNICAMP

chesteve@dca.fee.unicamp.br

INTRIG INFORMATION & NETWORKING TECHNOLOGIES RESEARCH & INNOVATION GROUP

UNICAMP

Department of Computer Engineering and Industrial Automation
School of Electrical and Computer Engineering
State University of Campinas

SDN in 2010

Anonymous uploader

SDN in 2011 – 2012

SDN in 2013 - 2014

Academia

Vendor A

Start-up 1

Start-up 2

Vendor B

...

Vendor C

Start-up n

SDN in 2014+ ?

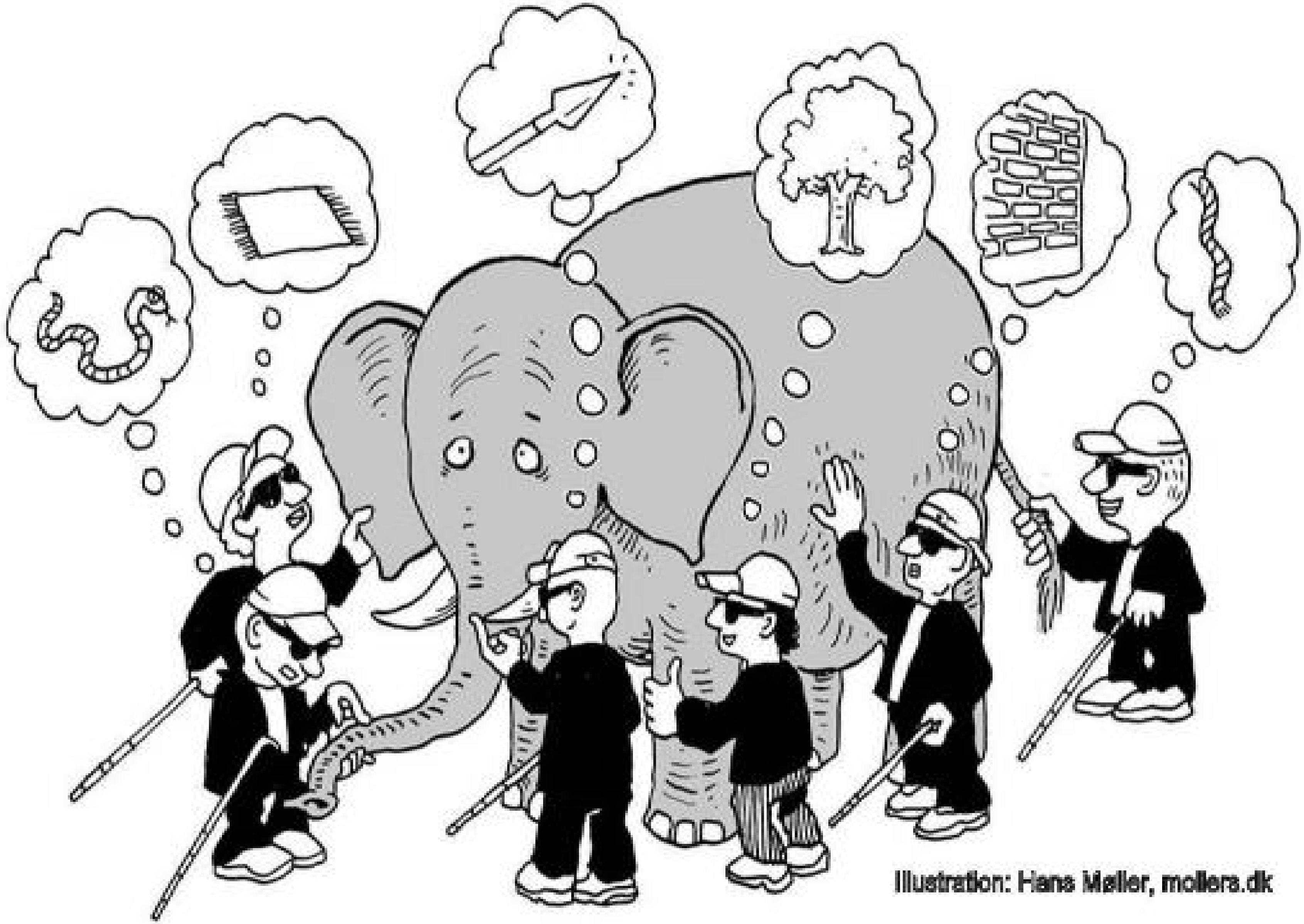
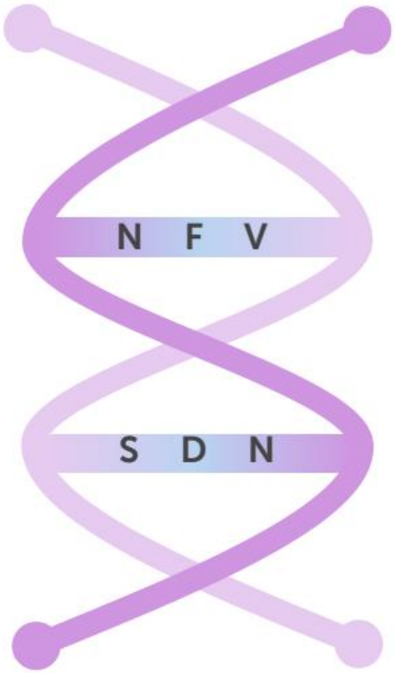


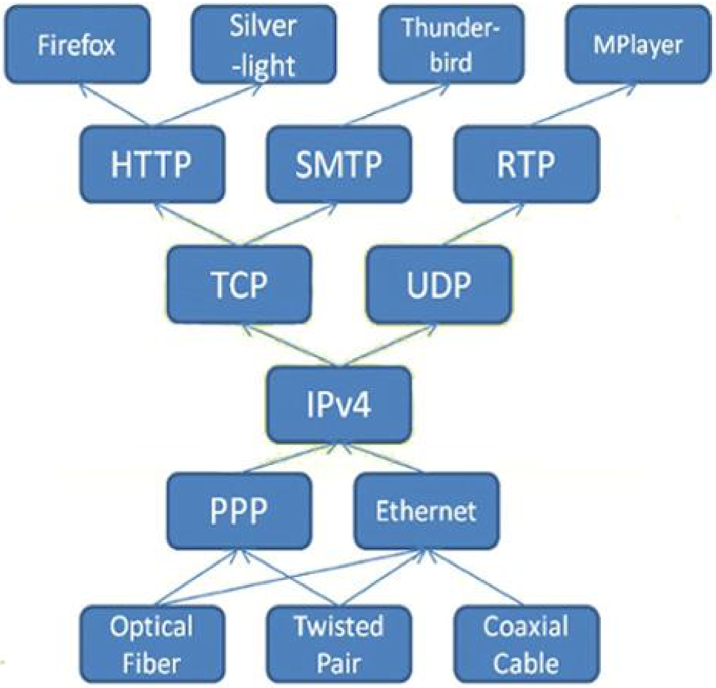
Illustration: Hans Møller, mollers.dk

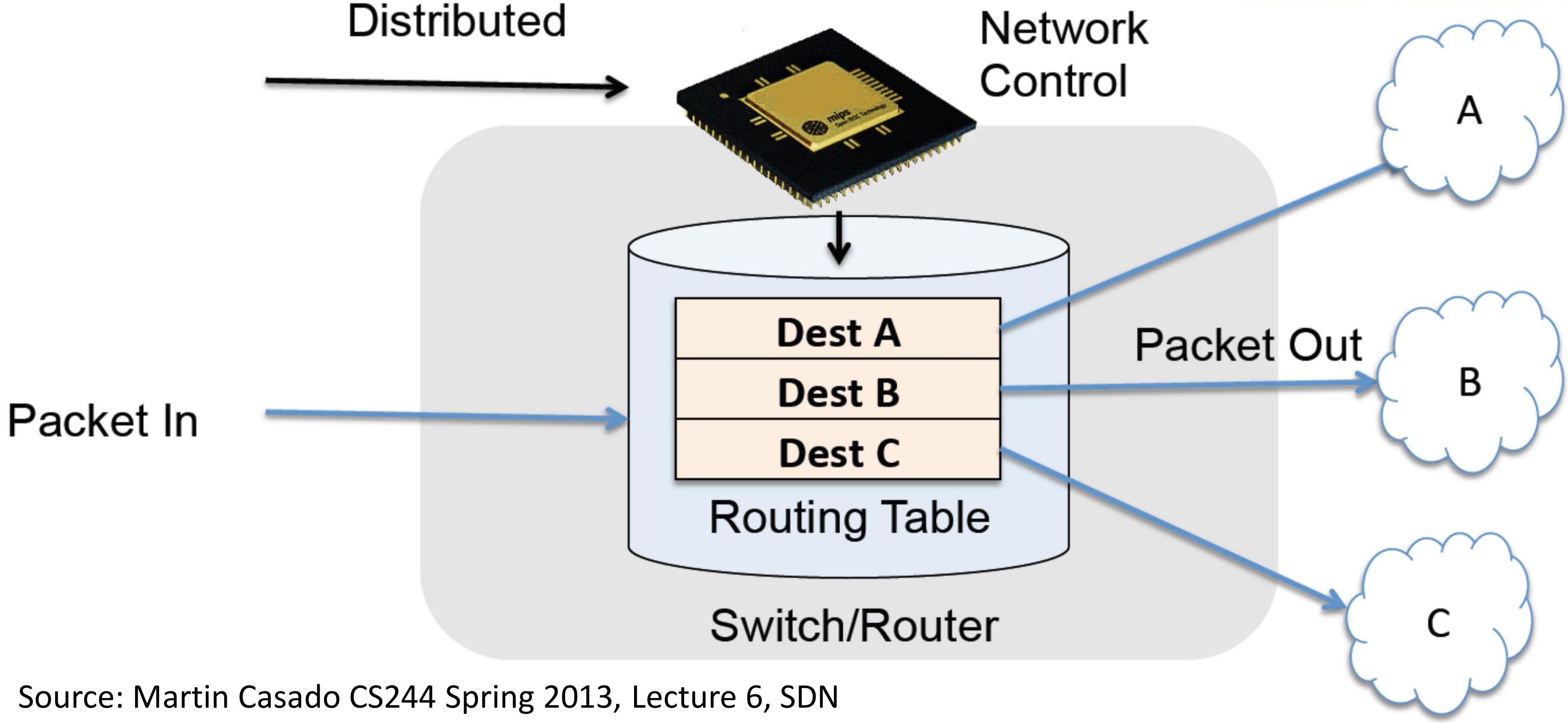
Agenda



- SDN (Software-Defined Networking)
  - Why? What? How?
- NFV (Network Functions Virtualization)
  - Why? What? How?
- SDS (Software-Defined Storage)
- Interlude: Layering of Abstractions & Open Source
- Conclusion

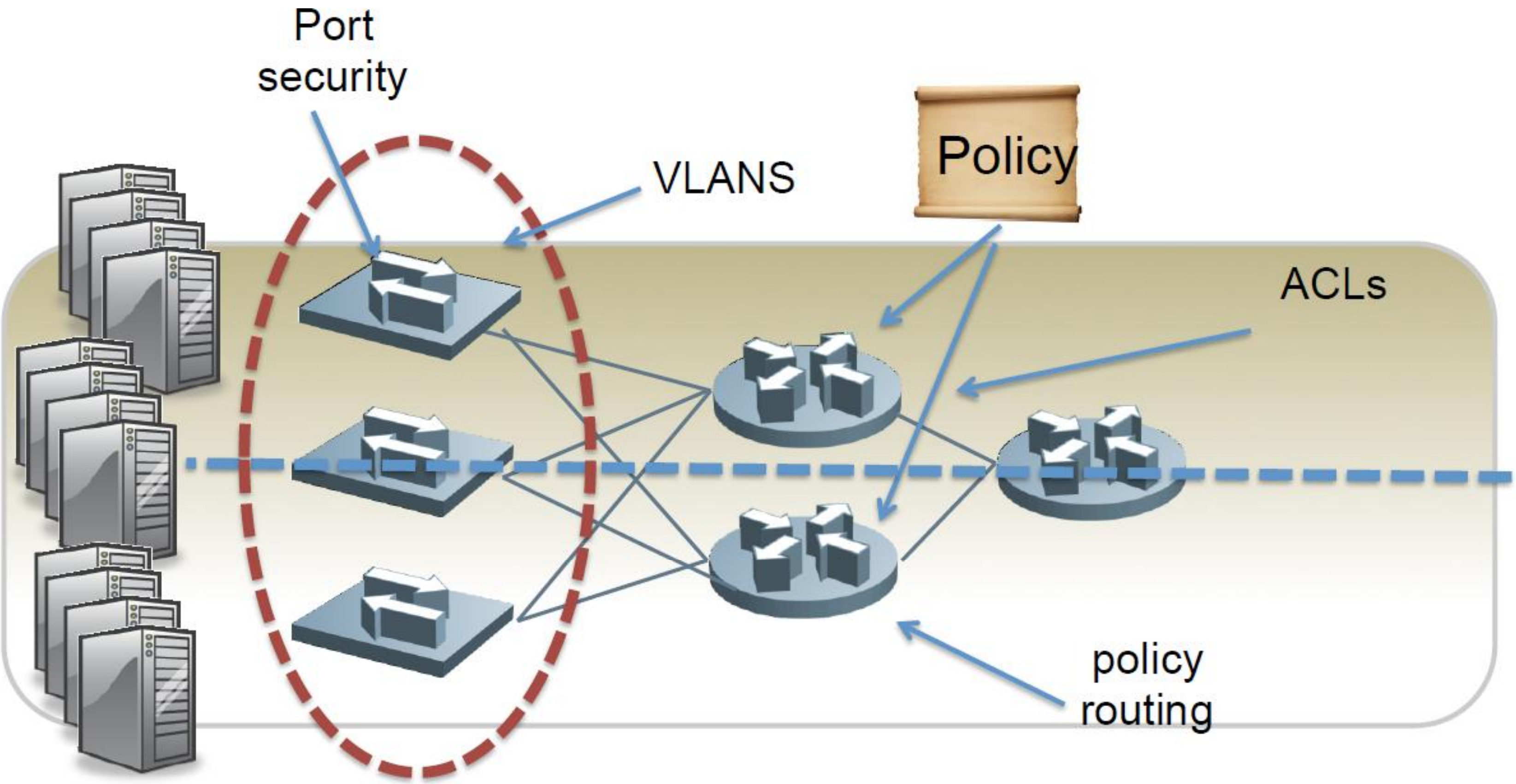
Networking as Learned in School (text books)





Source: Martin Casado CS244 Spring 2013, Lecture 6, SDN

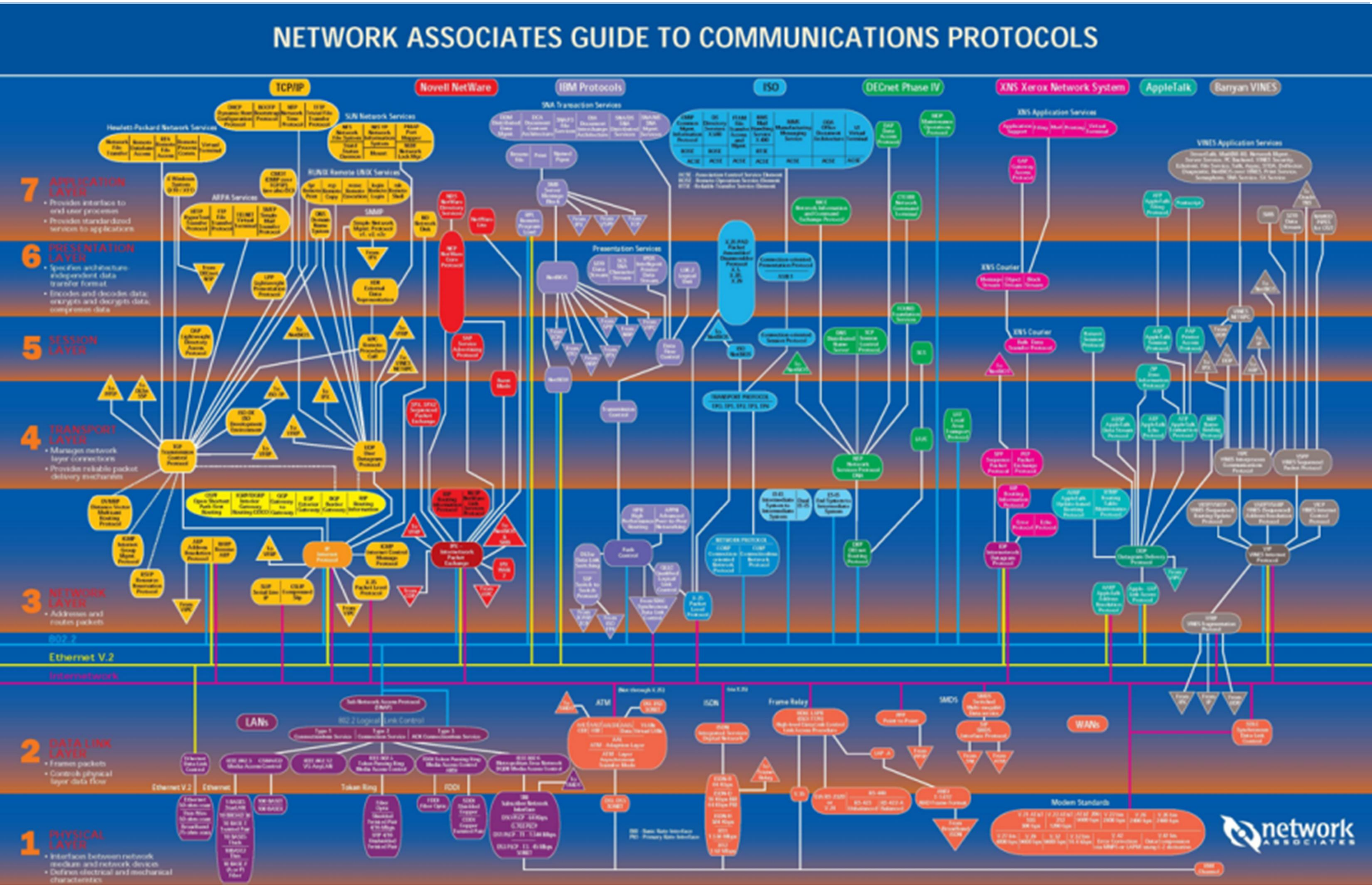
Networks in Practice



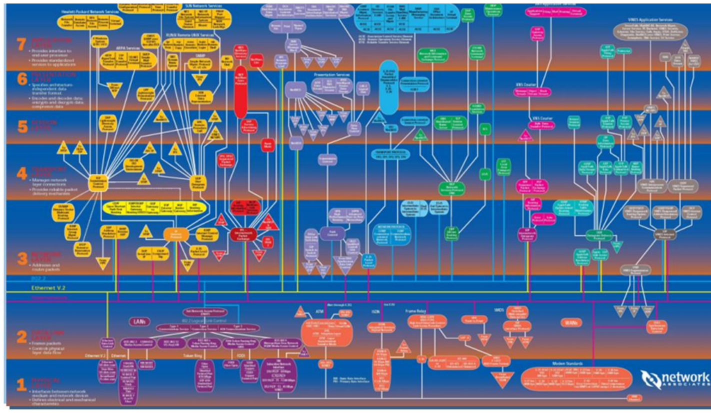
- Limited redundancy
- Constrained topology
- Poor response to dynamic events
- Scaling limited by operational complexity

Source: Martin Casado

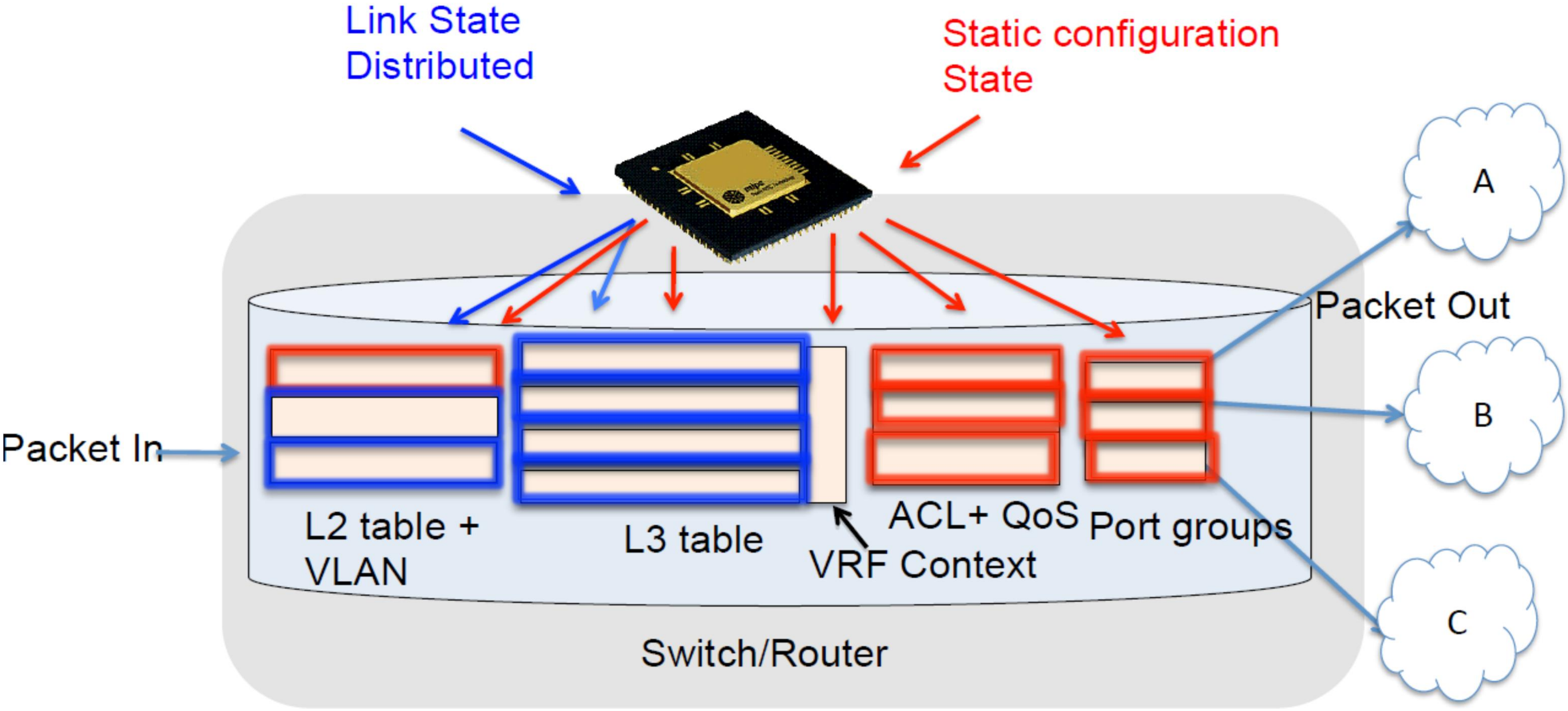
Where are we today in networking?


NETWORK ASSOCIATES GUIDE TO COMMUNICATIONS PROTOCOLS
TCP/IP
Novell NetWare
IBM Protocols
ISO
DECnet Phase IV
XNS Xerox Network System
AppleTalk
Banyan VINES
Hewlett-Packard Network Services
SUN Network Services
ARPA Services
SNA Transaction Services
XNS Application Services
VINES Application Services
Presentation Services
XNS Courier
7 APPLICATION LAYER
• Provides interface to end user processes
• Provides standardized services to applications
6 PRESENTATION LAYER
• Specifies architecture-independent data transfer format
• Encodes and decodes data; encrypts and decrypts data; compresses data
5 SESSION LAYER
4 TRANSPORT LAYER
• Manages network layer connections
• Provides reliable packet delivery mechanisms
3 NETWORK LAYER
• Addresses and routes packets
802.2
Ethernet V.2
Internetwork
2 DATA LINK LAYER
• Frames packets
• Controls physical layer data flow
1 PHYSICAL LAYER
• Interfaces between network medium and network devices
• Defines electrical and mechanical characteristics
LANs
802.2 Logical Link Control
Ethernet V.2
Ethernet
Token Ring
FDDI
ATM
ISDN
Frame Relay
SMDS
WANs
Modem Standards
network ASSOCIATES
]

Networking in Practice



“in theory, theory and practice are the same; in practice they are not...”



Link State Distributed

Static configuration State

Packet In

L2 table + VLAN

L3 table

VRF Context

ACL+ QoS

Port groups

Packet Out

A

B

C

Switch/Router

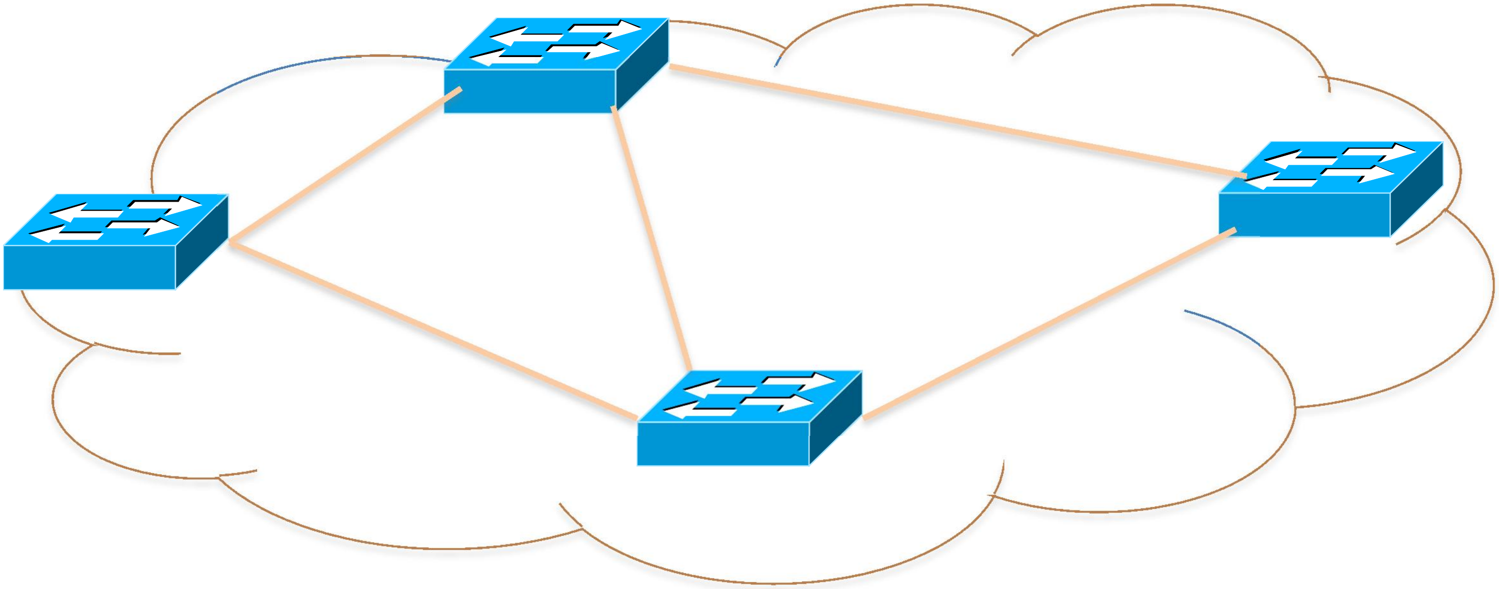
Source: Martin Casado CS244 Spring 2013, Lecture 6, SDN

OpenFlow/SDN to the rescue!

Source: packetpushers.net

Rethinking the "Division of Labor"
Traditional Computer Networks

Data plane: **Packet streaming**

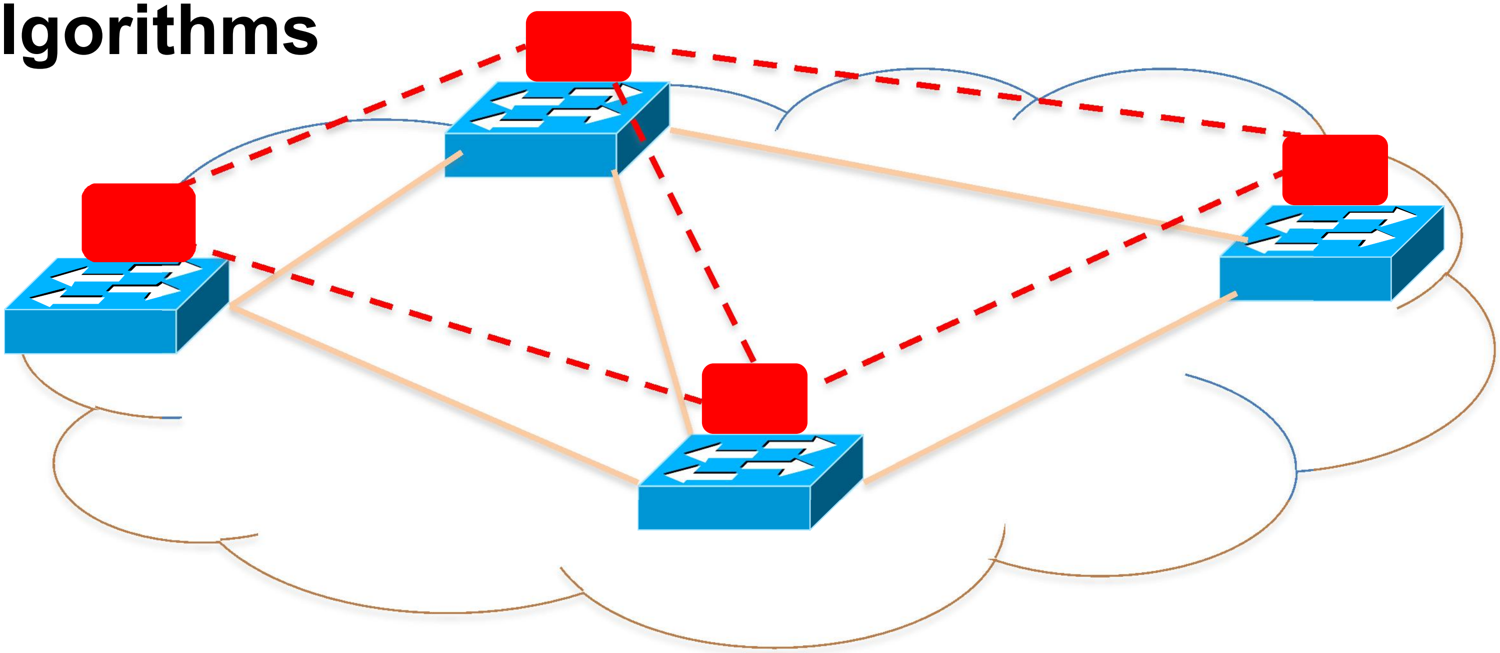


Forward, filter, buffer, mark, rate-limit, and measure packets

Source: Adapted from J. Rexford

Rethinking the "Division of Labor"
Traditional Computer Networks

Control plane:
Distributed algorithms



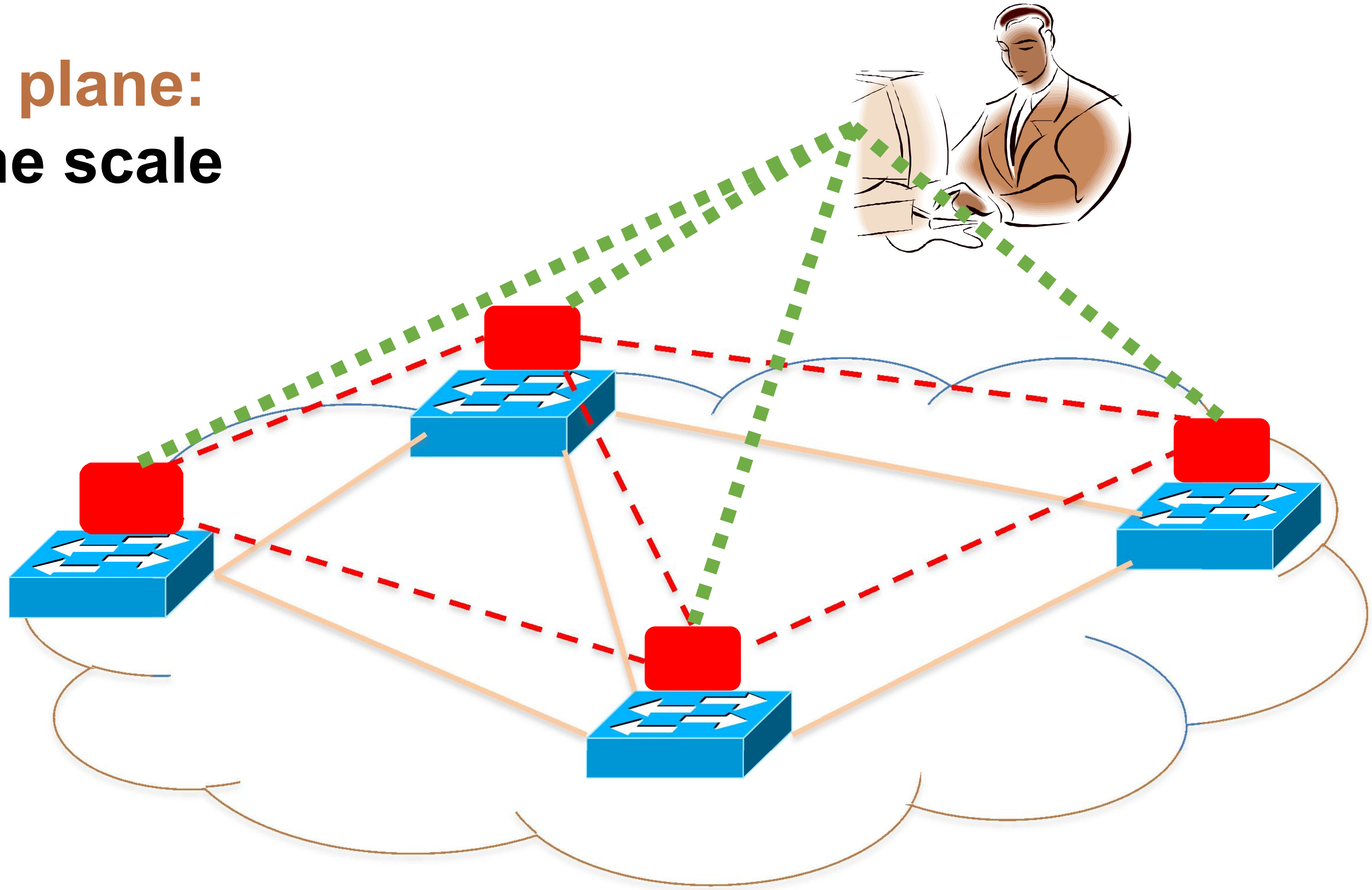
Track topology changes, compute routes, install forwarding rules

Source: Adapted from J. Rexford

Rethinking the "Division of Labor"
Traditional Computer Networks

Management plane:

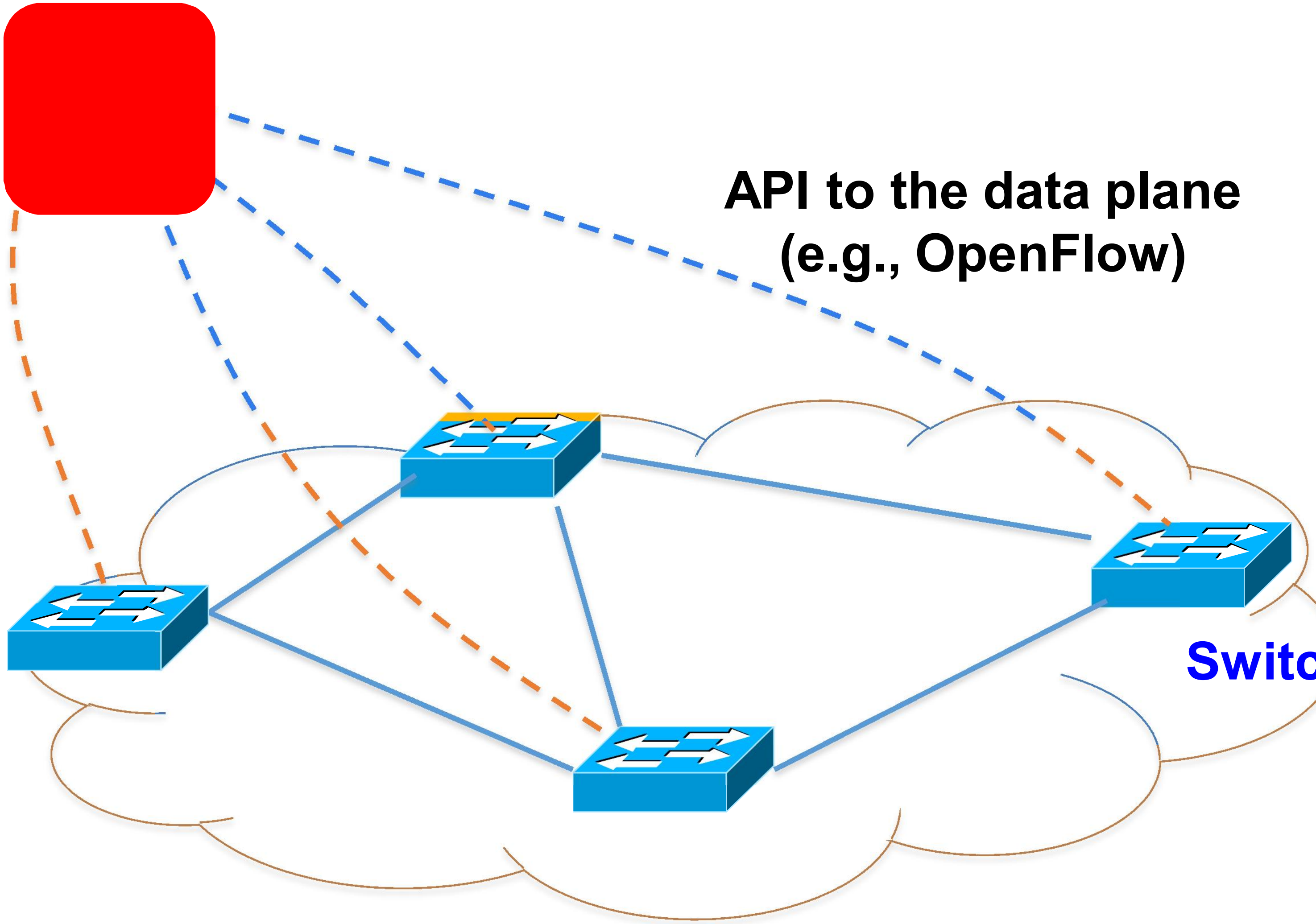
Human time scale



Collect measurements and configure the equipment

Source: Adapted from J. Rexford

Software Defined Networking (SDN)



Logically-centralized control

Smart, slow

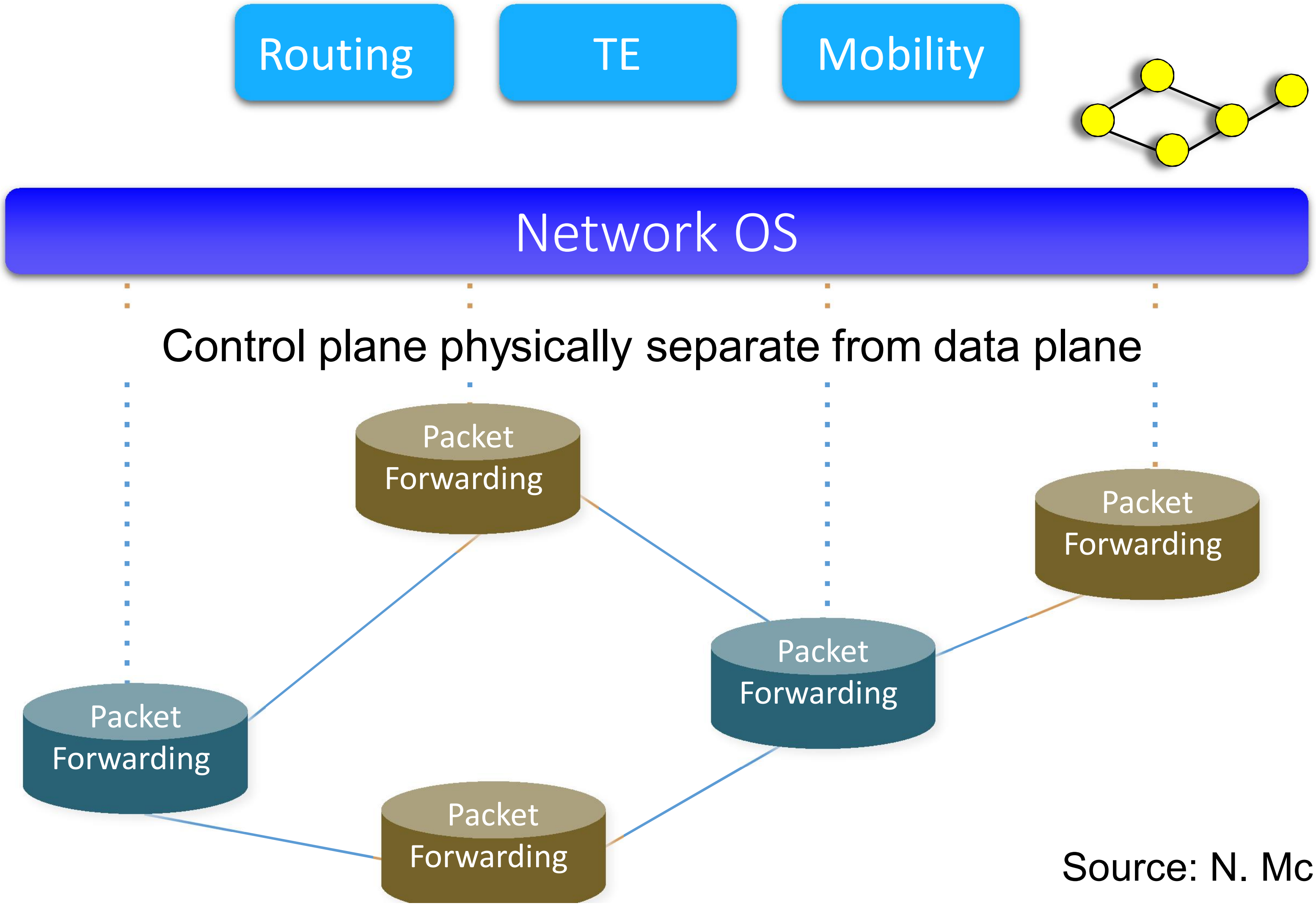
API to the data plane (e.g., OpenFlow)

Dumb, fast

Switches

SDN: Fundamental Elements

Single control plane controls several forwarding devices



Routing

TE

Mobility

Network OS

Control plane physically separate from data plane

Packet Forwarding

Packet Forwarding

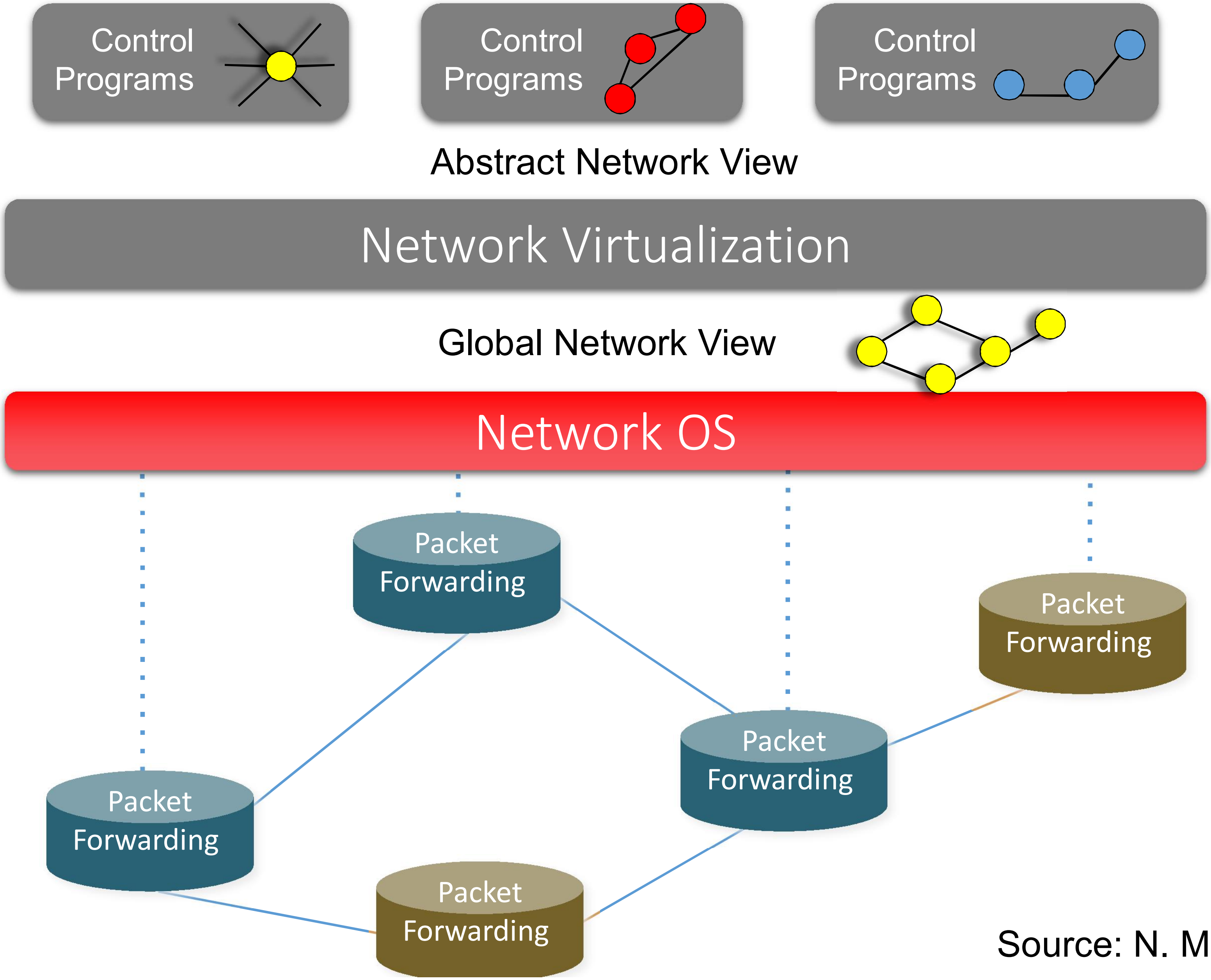
Packet Forwarding

Packet Forwarding

Packet Forwarding

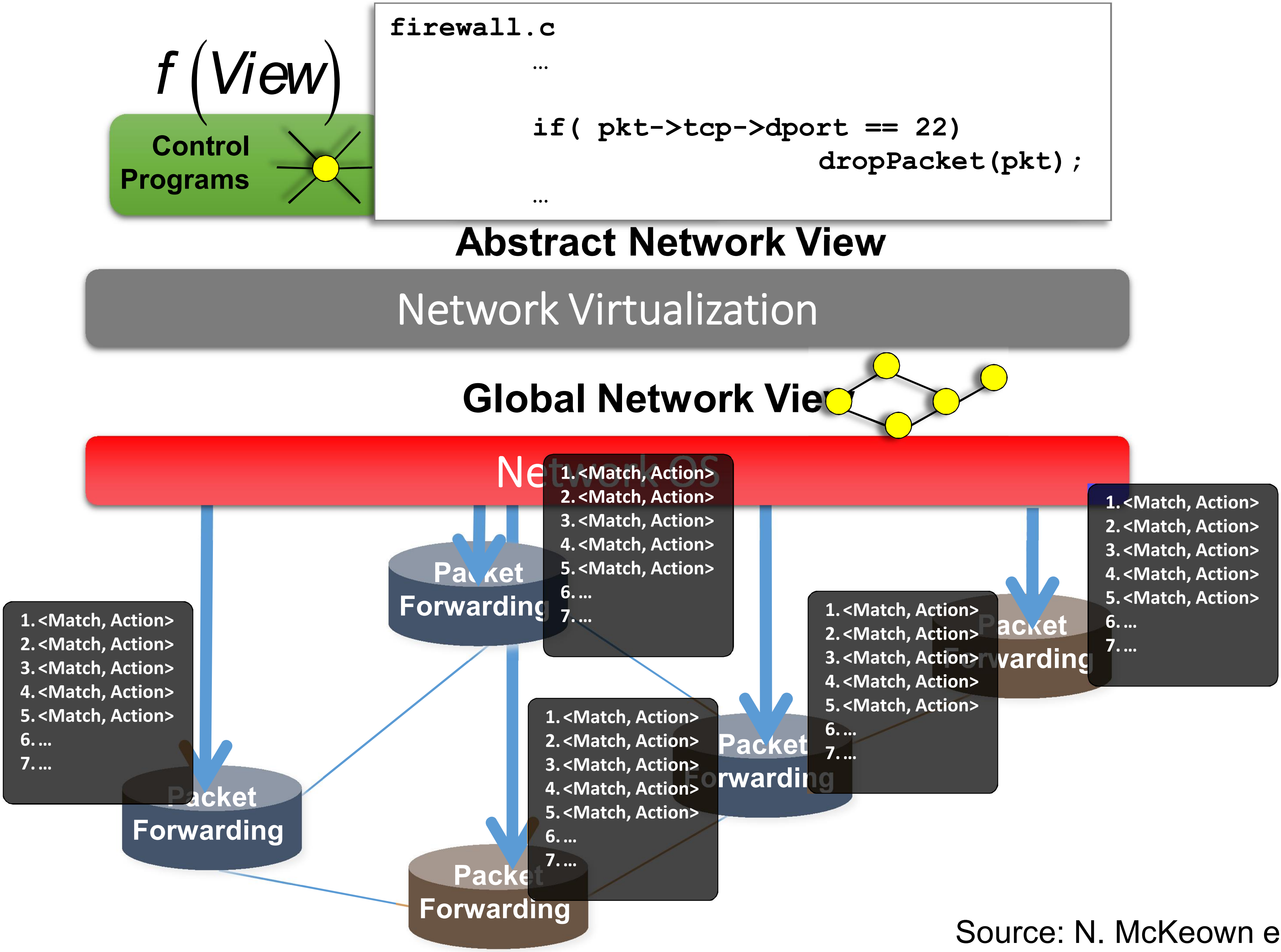
Source: N. McKeown et al.

SDN: Virtualization



Source: N. McKeown et al.

Software Defined Network (SDN)



```
firewall.c
        …

        if( pkt->tcp->dport == 22)
                        dropPacket(pkt);
        …
```

1. <Match, Action>
2. <Match, Action>
3. <Match, Action>
4. <Match, Action>
5. <Match, Action>
6. …
7. …

1. <Match, Action>
2. <Match, Action>
3. <Match, Action>
4. <Match, Action>
5. <Match, Action>
6. …
7. …

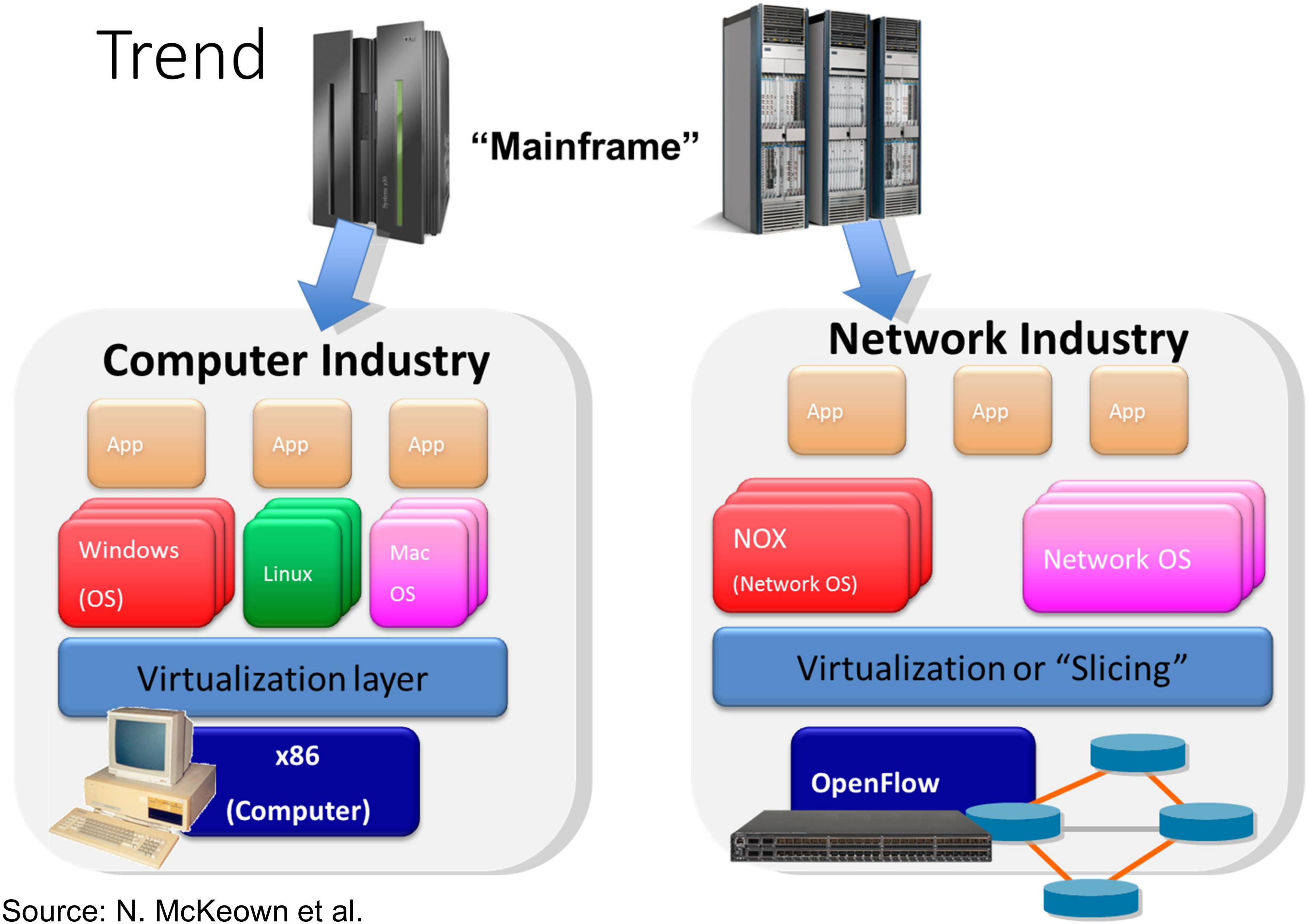
1. <Match, Action>
2. <Match, Action>
3. <Match, Action>
4. <Match, Action>
5. <Match, Action>
6. …
7. …

1. <Match, Action>
2. <Match, Action>
3. <Match, Action>
4. <Match, Action>
5. <Match, Action>
6. …
7. …

1. <Match, Action>
2. <Match, Action>
3. <Match, Action>
4. <Match, Action>
5. <Match, Action>
6. …
7. …

Source: N. McKeown et al.

Trend



“Mainframe”

Computer Industry

App

App

App

Windows (OS)

Linux

Mac OS

Virtualization layer

x86 (Computer)

Network Industry

App

App

App

NOX (Network OS)

Network OS

Virtualization or “Slicing”

OpenFlow

Source: N. McKeown et al.

Vision: Bare Metal SDN

WHAT'S INSIDE A SWITCH?

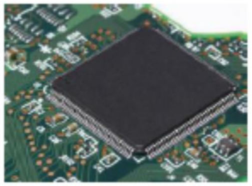
Application

Network OS

Hardware Driver

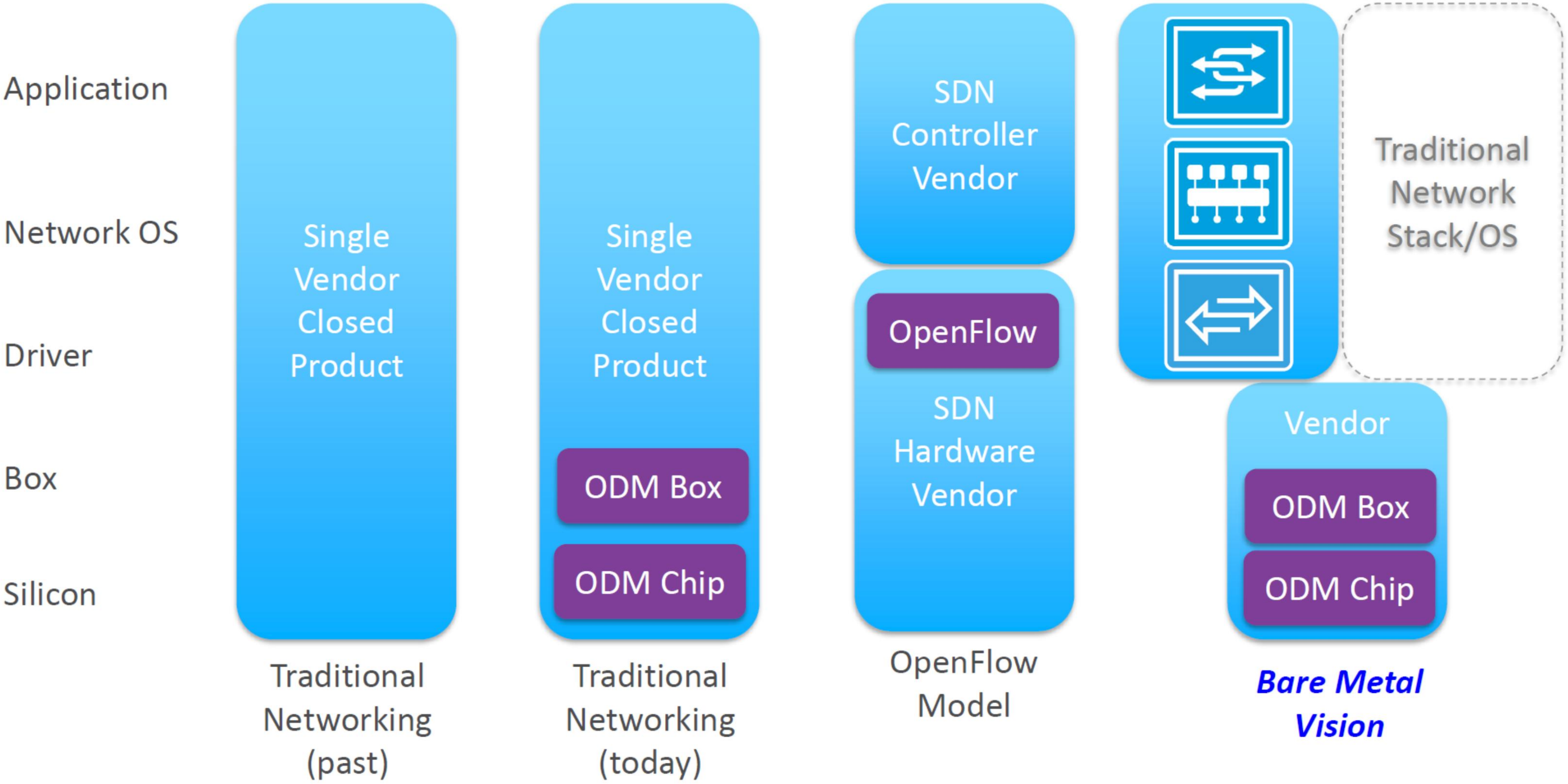
Box

Silicon



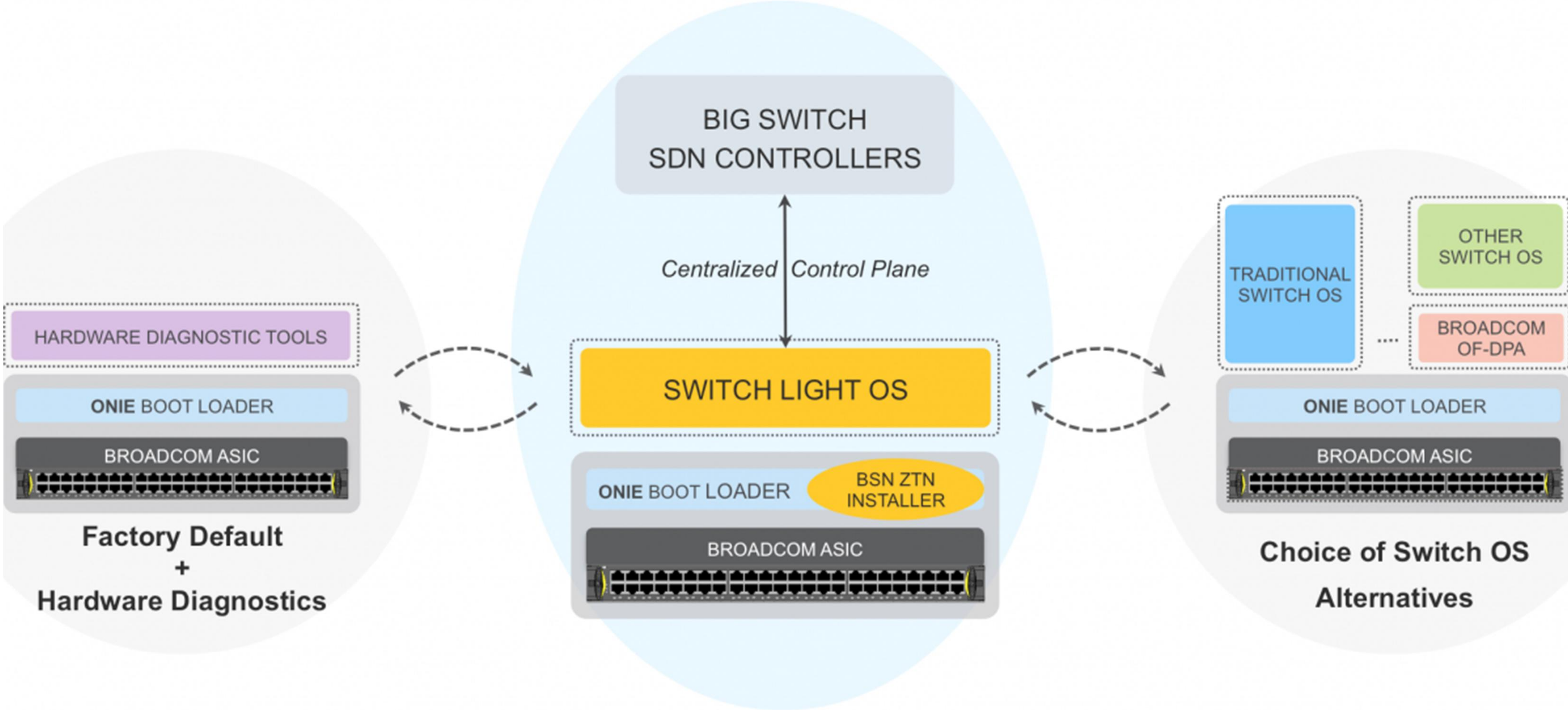
Source: Rob Sherwood, Big Switch Networks

Vision: Bare Metal SDN



Source: Rob Sherwood, Big Switch Networks

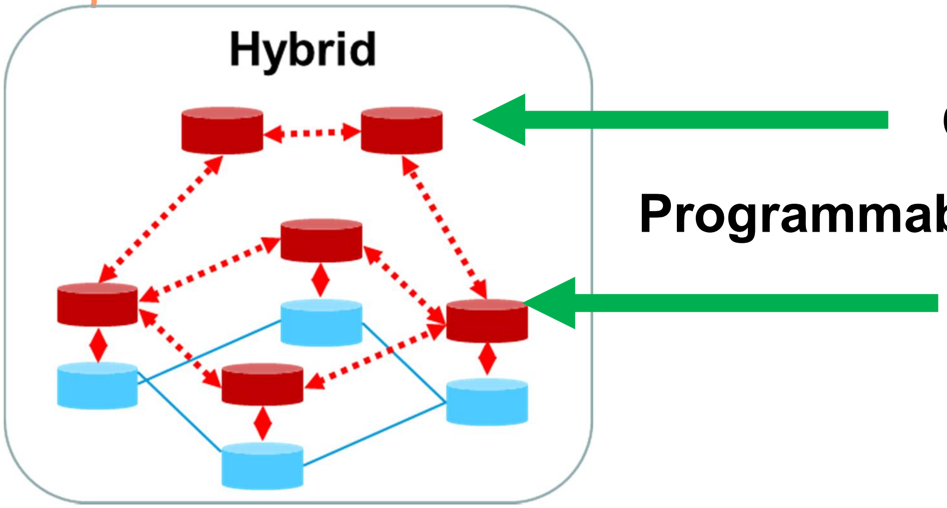
Bare Metal Switches: Choice of Switch OS



Big Switch SDN Fabric Deployment Mode

Source: Rob Sherwood, Big Switch Networks

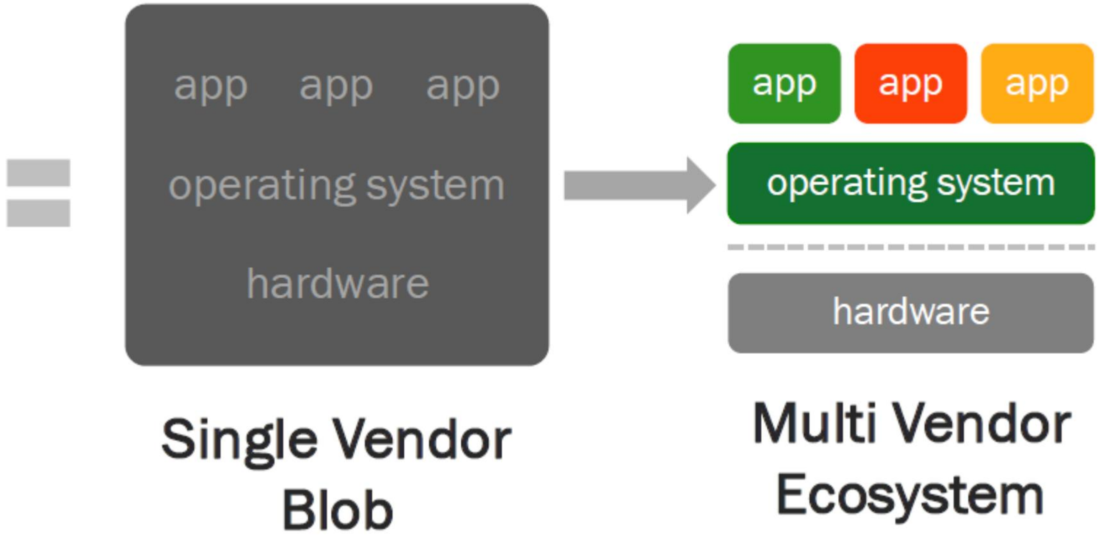
Bare Metal Switching and Programming
Blows Up the SDN Blob!

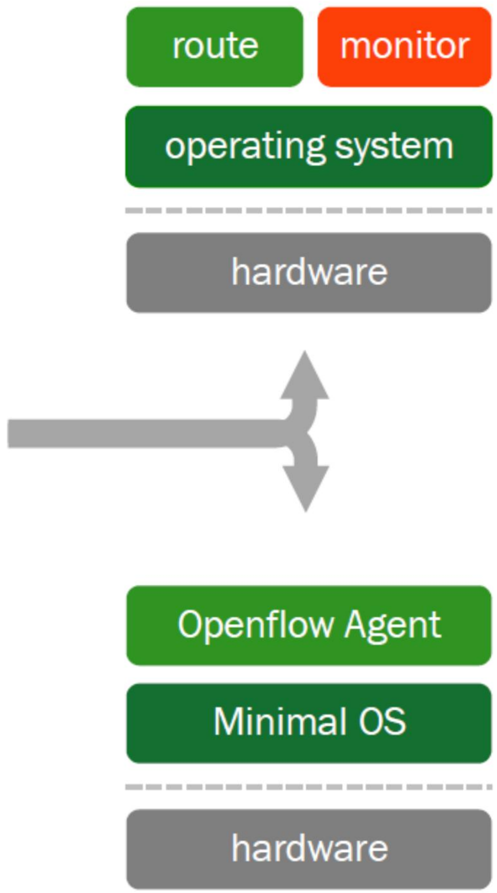


Out-of-the-box

Programmable Control SW

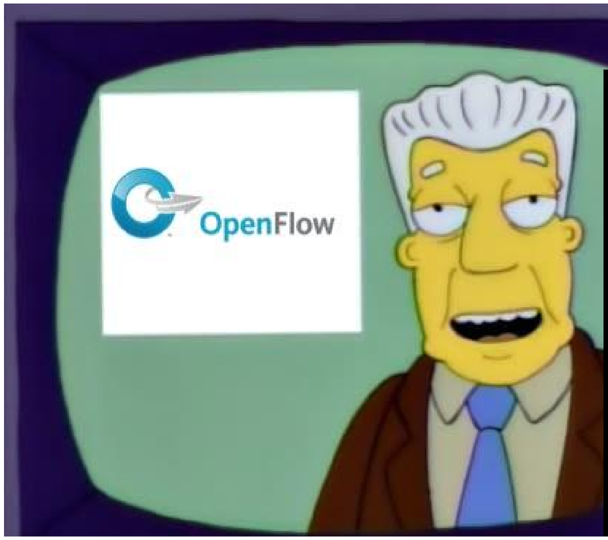
In-the-box

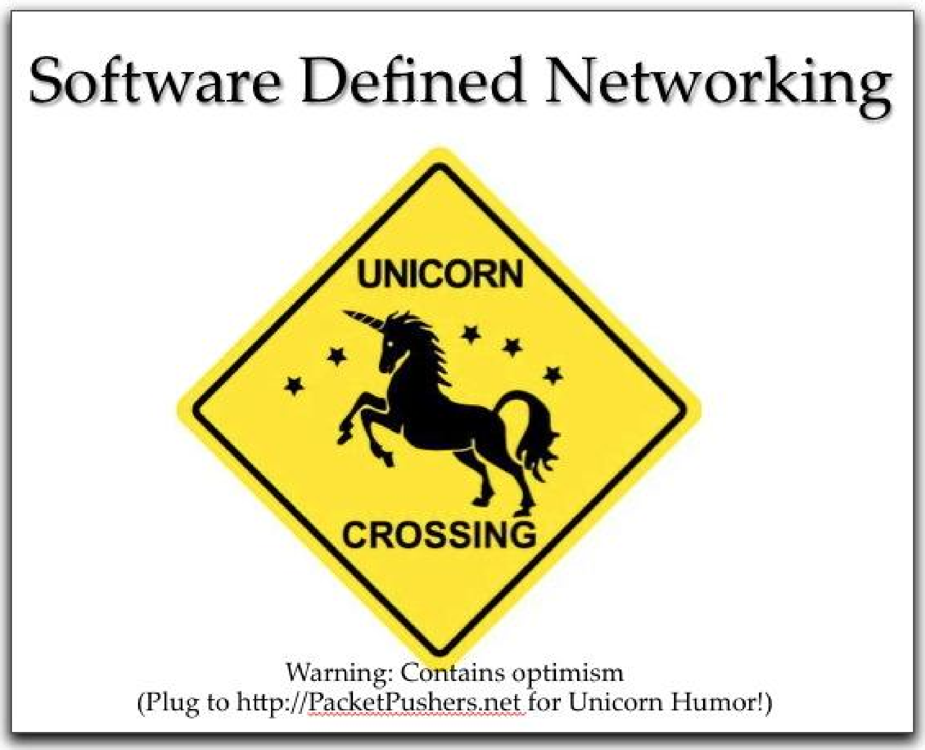




Source: Adapted from Shrijeet Mukherjee, Cumulus Networks

SDN to the rescue!





Software Defined Networking

UNICORN
CROSSING

Warning: Contains optimism
(Plug to http://PacketPushers.net for Unicorn Humor!)

What is SDN?

OpenFlow is SDN, but SDN is not OpenFlow

-- Networking community

(Does not say much about SDN)

Don’t let humans do machines’ work

-- Networking Professional

Let’s call whatever we can ship today SDN

-- Vendor X

SDN is the magic buzzword that will bring us VC funding

-- Startup Y

SDN is the magic that will get my paper/grant accepted

-- Researcher Z

SDN definitions

- With the original definition, SDN represented a network architecture where the forwarding state is solely managed by a control plane and is decoupled from the data plane.
- The industry, however, has moved on from the *original academic purist view* of SDN to referring *to anything disruptive or fundamentally new* as part of SDN.

At least two definitions for SDN:

1. **academic**
   (purist view : strict decoupling of the data and control plane)
2. **industry**
   (many-fold business-driven views)

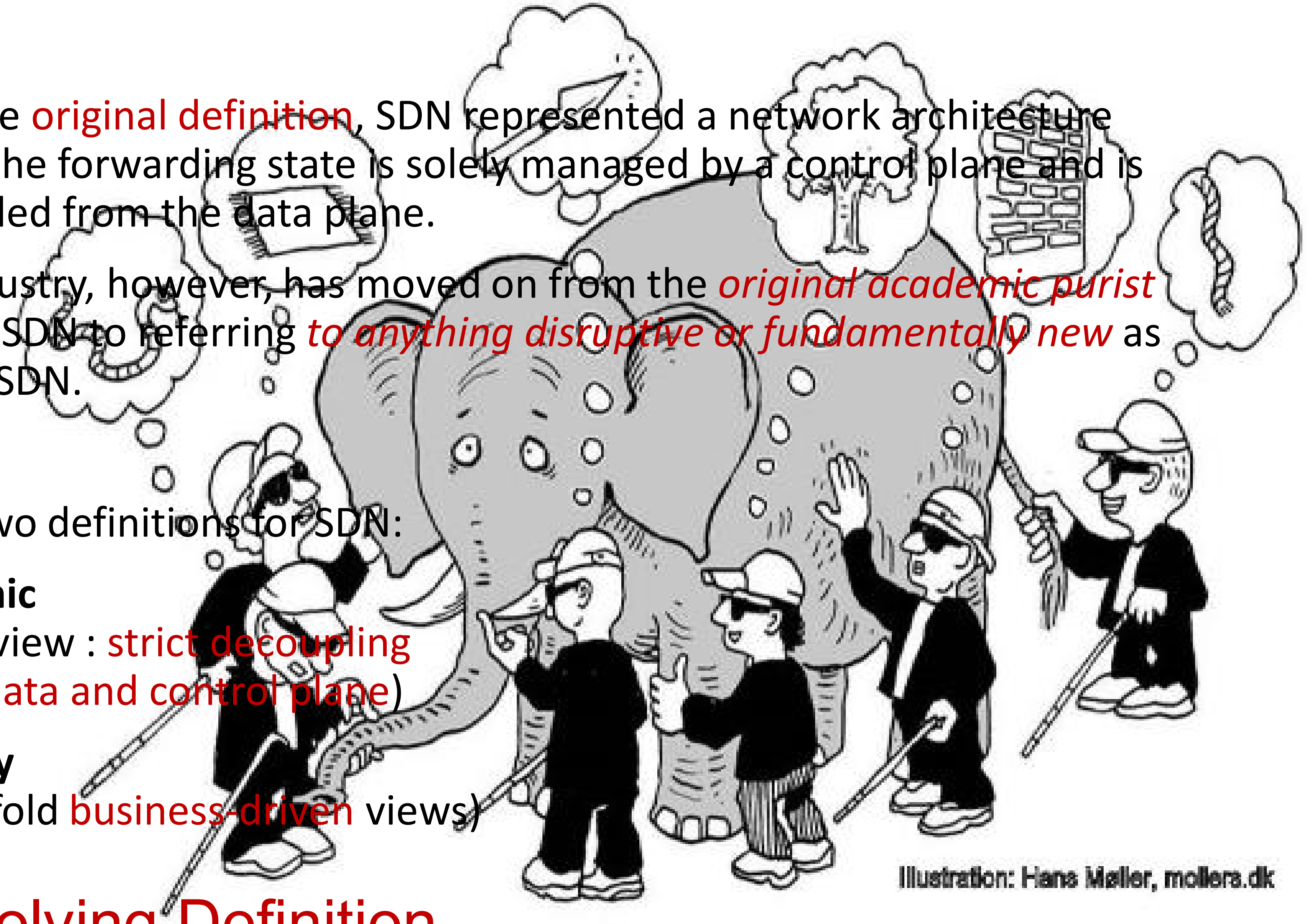


Illustration: Hans Møller, mollers.dk

SDN – Evolving Definition

What is SDN?

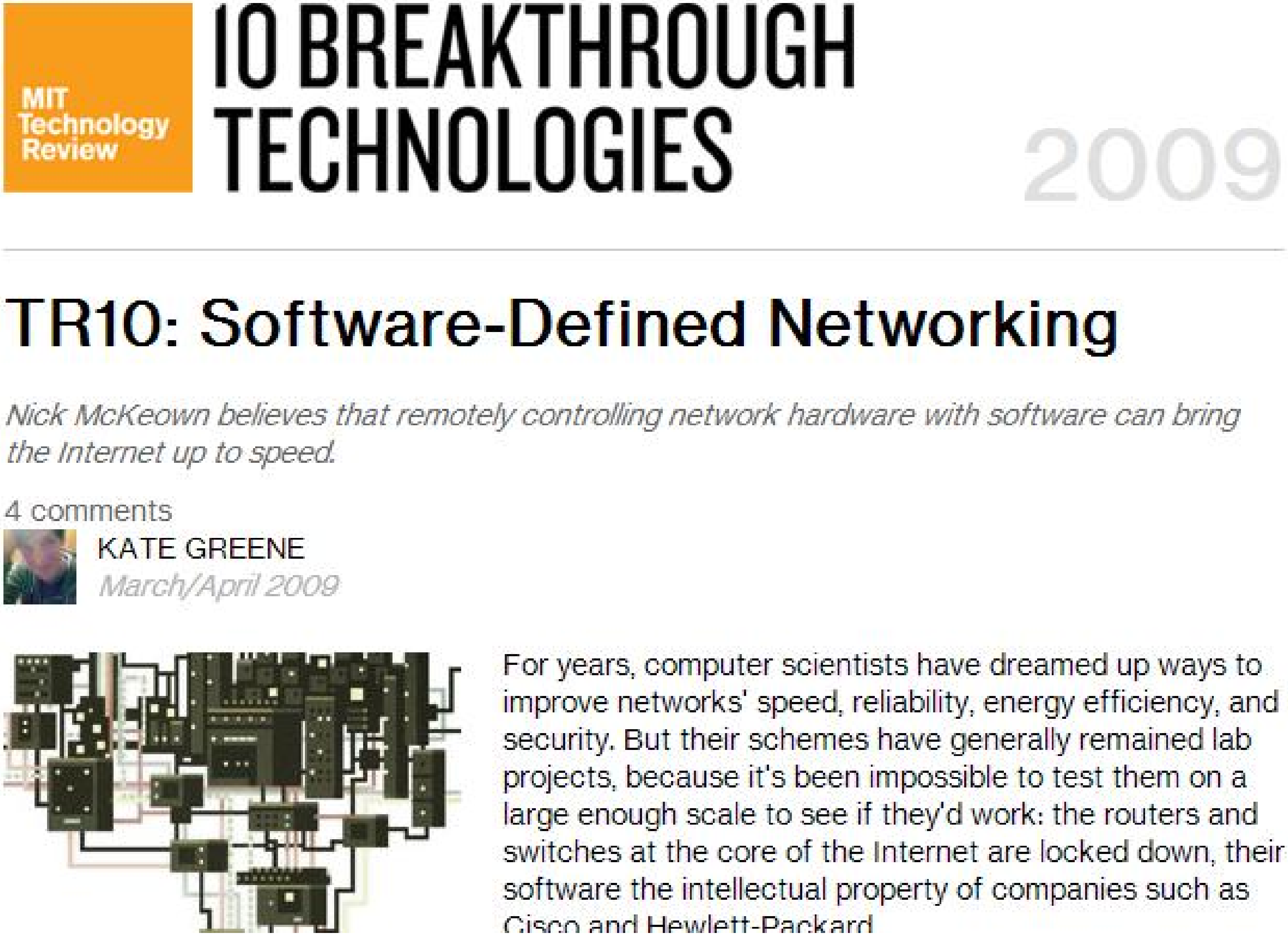
In the SDN architecture, the control and data planes are decoupled, network intelligence and state are logically centralized, and the underlying network infrastructure is abstracted from the applications.

-- Open Networking Foundation white paper

Software Defined Networking (SDN) refactors the relationship between network devices and the software that controls them. Open interfaces to network switches enable more flexible and predictable network control, and they make it easier to extend network function.

-- HotSDN CFP

The “origins” of the SDN term



MIT Technology Review

10 BREAKTHROUGH TECHNOLOGIES 2009

TR10: Software-Defined Networking

Nick McKeown believes that remotely controlling network hardware with software can bring the Internet up to speed.

4 comments

KATE GREENE

March/April 2009

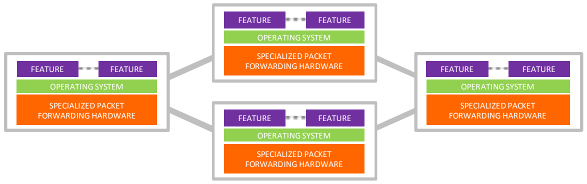
For years, computer scientists have dreamed up ways to improve networks' speed, reliability, energy efficiency, and security. But their schemes have generally remained lab projects, because it's been impossible to test them on a large enough scale to see if they'd work: the routers and switches at the core of the Internet are locked down, their software the intellectual property of companies such as Cisco and Hewlett-Packard

Software Defined Networking

Network equipment as Black boxes

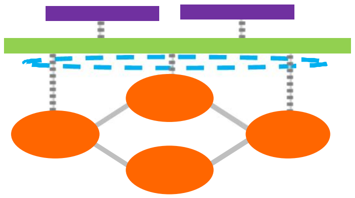
SDN

Open interfaces (OpenFlow) for instructing the boxes what to do

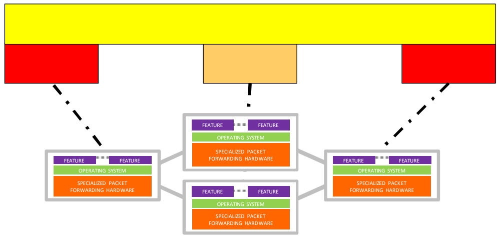


Boxes with autonomous behaviour

SDN

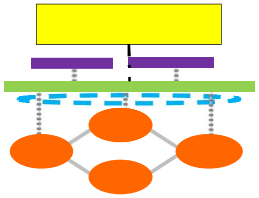


Decisions are taken out of the box



Adapting OSS to manage black boxes

SDN



Simpler OSS to manage the SDN controller

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

SDS – Software-Defined Storage

Definição

- Existem muitas definições para SDS
- Muitas empresas estão aproveitando a atenção gerada para chamar seus produtos de SDS.
- A maioria são apenas soluções de virtualização de Storage antigas.

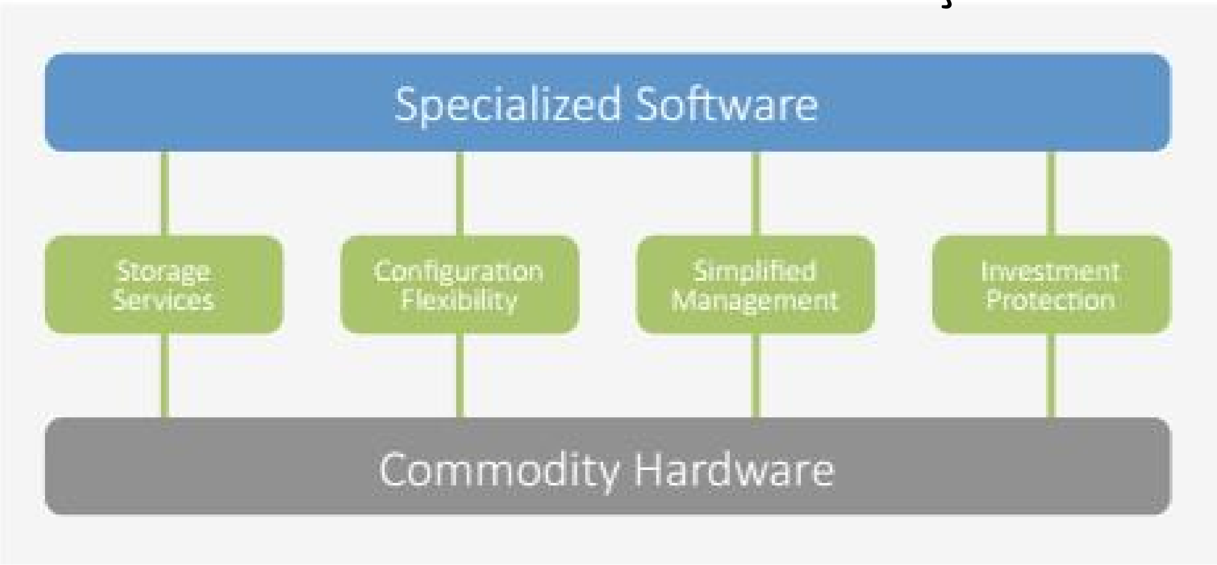
Uma definição abrangente

- "Software-Defined Storage abstrai a camada de controle dos equipamentos de storage do hardware para uma camada, separada, de software. Essa camada, acima do plano de dados, controla centralizadamente todos os recursos de storage e é responsável pelo provisionamento, otimização e monitoramento dos recursos e dados "

Características | Vantagens

1. Storage Orientada à políticas (IOPS, latency, reliability, Fault tolerance, Provisioning, QoS)
2. Arquitetura Scale-out
3. “Storage as a Seamless pool of resource”
4. Programabilidade
5. HW commodity

- Data centers sem vendor lock-in
- Gerenciamento centralizado e simplificado
- Implementação de políticas de performance para diferentes tipos de dados e aplicações
- Melhoria na qualidade dos serviços
- Storage Tiering automático
- Diminuição de custos



SDS@Lenovo Innovation Center

Alguns Tópicos de Interesse de Pesquisa

- Gerenciamento
- Monitoramento
- Arquitetura
  - Tolerancia à falhas
  - Escalabilidade
  - Alta disponibilidade
  - Eficiência Energética

- Interesse em pesquisa de serviços para SDS
  - Deduplicação
  - Técnicas eficientes de replicação e proteção dos dados
  - Compression
  - Snapshots

Scale and Virtualization in the Timeline

Early twentieth century

- Manual Switching
- Very intensive in human resources
- Era **dominated by hardware**

Mid-twentieth century

- Electromechanical Switching
- Less intensive in human resources
- Era **dominated by complex hardware**

Virtualization technologies enables overcoming physical constraints and generating multiplexing gains…

- Digital Switching
- Much less intensive in human resources
- Era **dominated by complex and specific hardware**. **Software appears and is important**
- Services defined by telco

Second half of the twentieth century

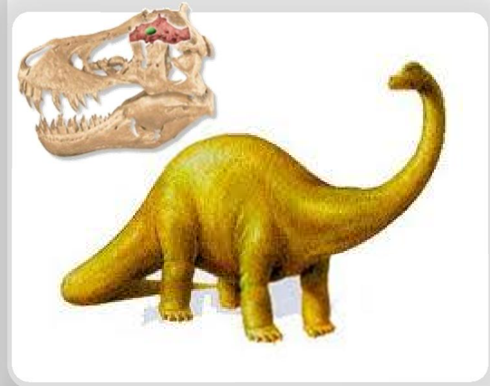
- Internet connectivity opens the door to the development of OTT services (without operator)
- **Software becomes a differentiation asset**

Early twenty-first century

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

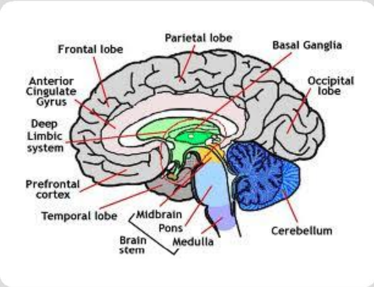
Enter the Software-Defined Era

Traditional telcos



- Very **intensive in hardware**
- Software not at the core

Internet players



- Very **intensive in software**
- Hardware is a necessary base

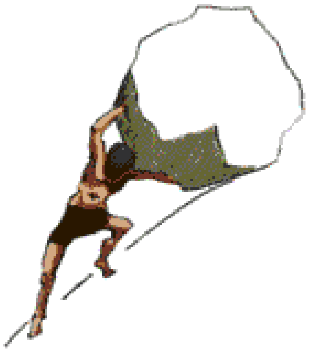
HARDWARE **SOFTWARE**

AT&T, Telefonica, Telebras

Google, Facebook

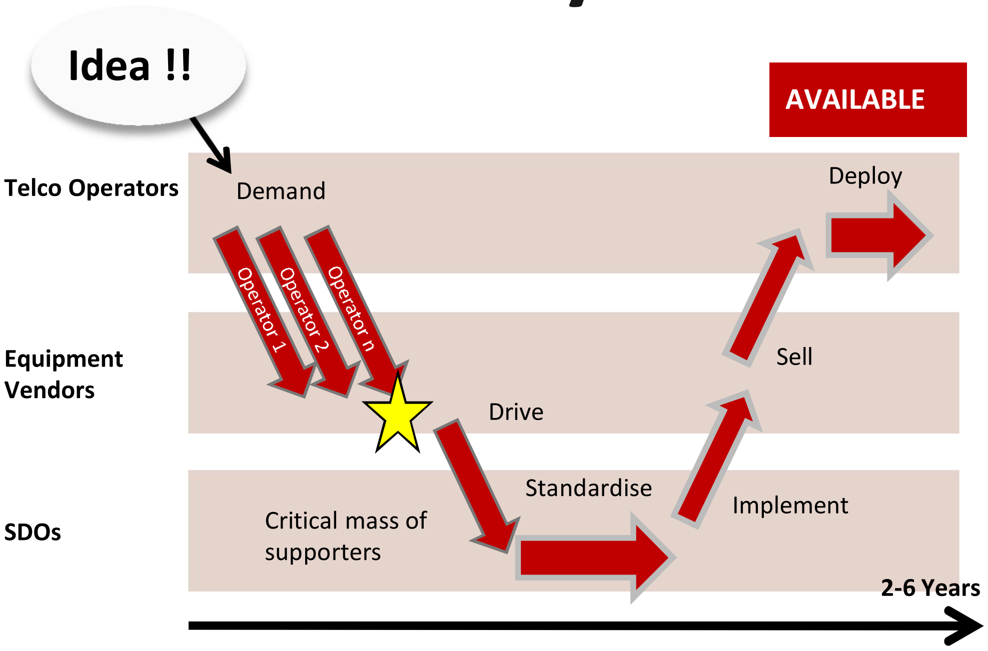
Adapt to survive: Evolution focus shifting from hardware to software

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



Sisyphus on Different Hills

Telco Cycle



Idea !!

AVAILABLE

Telco Operators

Demand

Deploy

Operator 1

Operator 2

Operator n

Equipment Vendors

Drive

Sell

SDOs

Critical mass of supporters

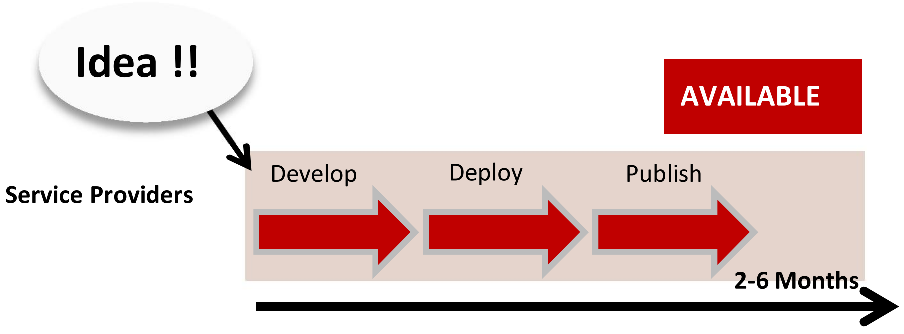
Standardise

Implement

2-6 Years

2-6 years

Service Providers Cycle



Idea !!

AVAILABLE

Service Providers

Develop

Deploy

Publish

2-6 Months

2-6 months

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

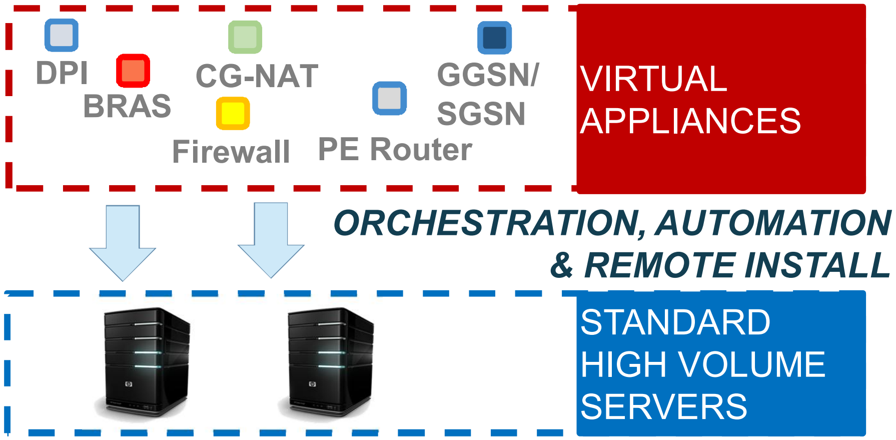
The NFV Concept

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


DPI
BRAS
CG-NAT
Firewall
PE Router
GGSN/ SGSN
VIRTUAL APPLIANCES
ORCHESTRATION, AUTOMATION & REMOTE INSTALL
STANDARD HIGH VOLUME SERVERS

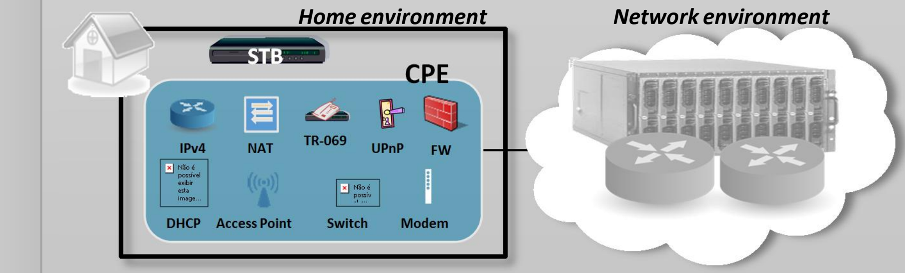
- Network Functions are **SW-based over well-known HW**
- **Multiple roles** over **same HW**

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

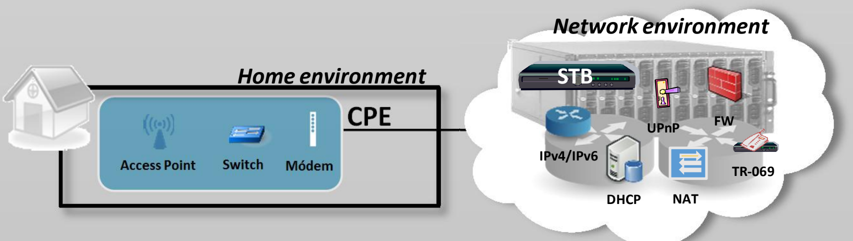
Some Drivers

Virtual CPE

Complex home environment



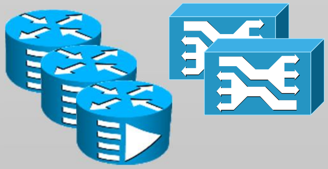
Home simplification



- Simplification or even supression (STB)
- No need for home router replacement as it is updated by configuration
- Fast deployment for new services
- Inexpensive IPv6 migration maintaining legacy home routers

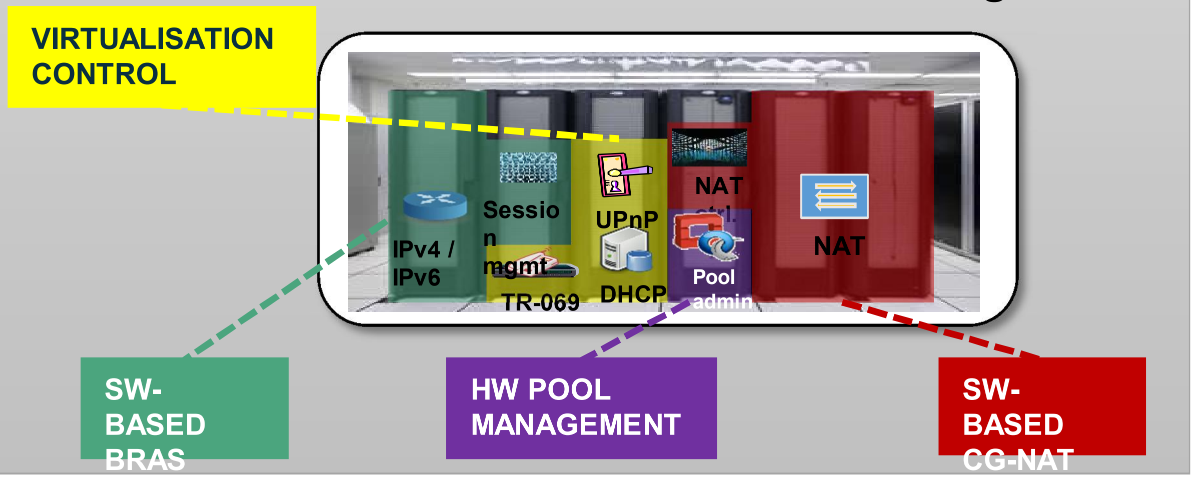
Virtual IP Edge

Multiple IP Edges



- An IP Edge for each service (voice, video content, Internet)
- Scattered and not well integrated control functions (e.g. DPI, BRAS, PCRF)

A unified software IP Edge



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

Interlude

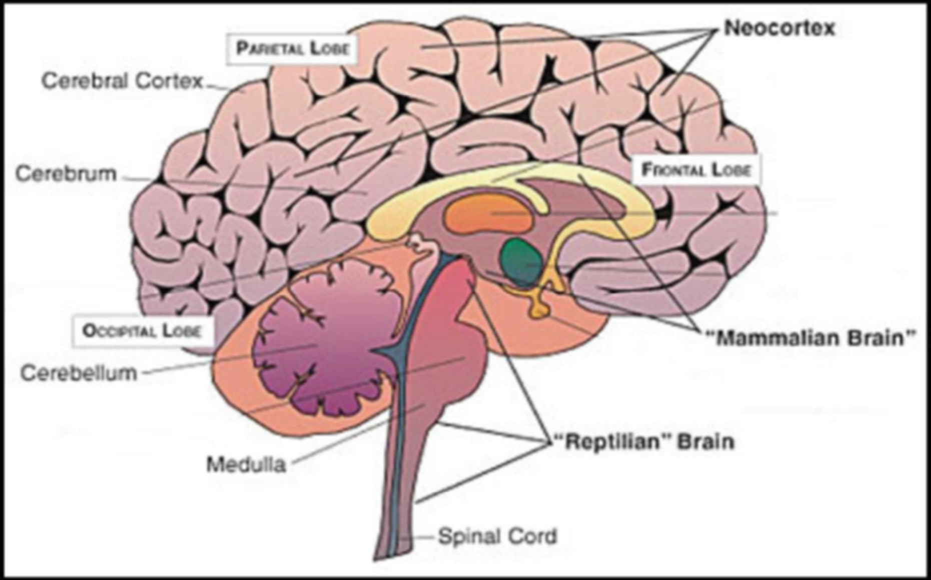
On SDN architectural evolution, HW/SW and biology

Trend: The Evolution of Intelligence

Precambrian (Reptilian) Brain to Neocortex → Hardware to Software

HARDWARE

SOFTWARE



- Key Architectural Features of Scalable/Evolvable Systems
  - RYF-Complexity (behavior)
  - Layered Architecture
  - Bowties and Hourglasses
  - Horizontal Transfer (H*T)
  - Protocol Based Architectures

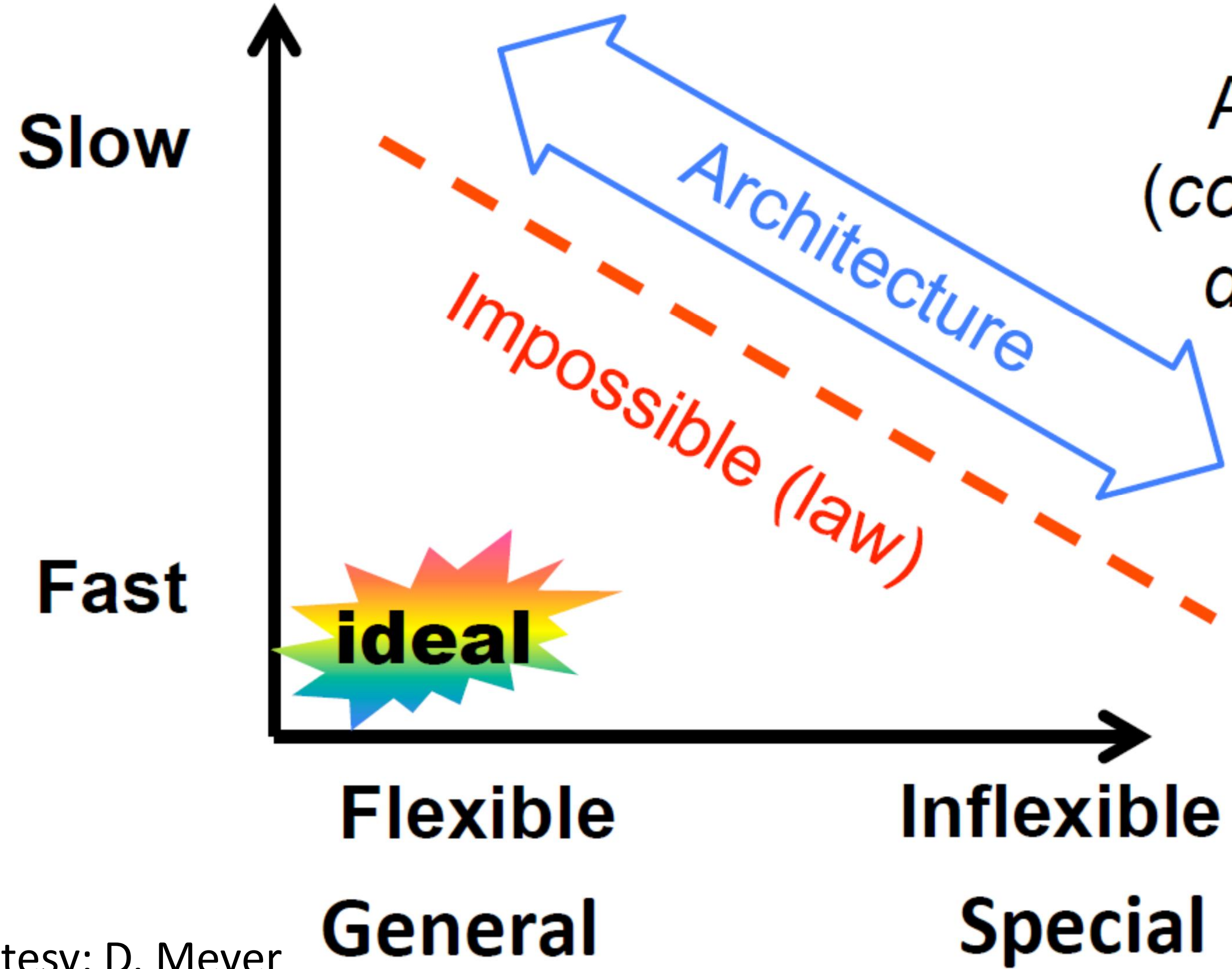
Once you have HW
its all about code[1]…

Slide courtesy: D. Meyer

[1] http://www.joelonsoftware.com/articles/Biculturalism.ht

Universal Laws and Architectures (Turing)

Layering, Formal Systems, Hard Tradeoffs



Slow

Fast

ideal

Impossible (law)

Architecture

Flexible

General

Inflexible

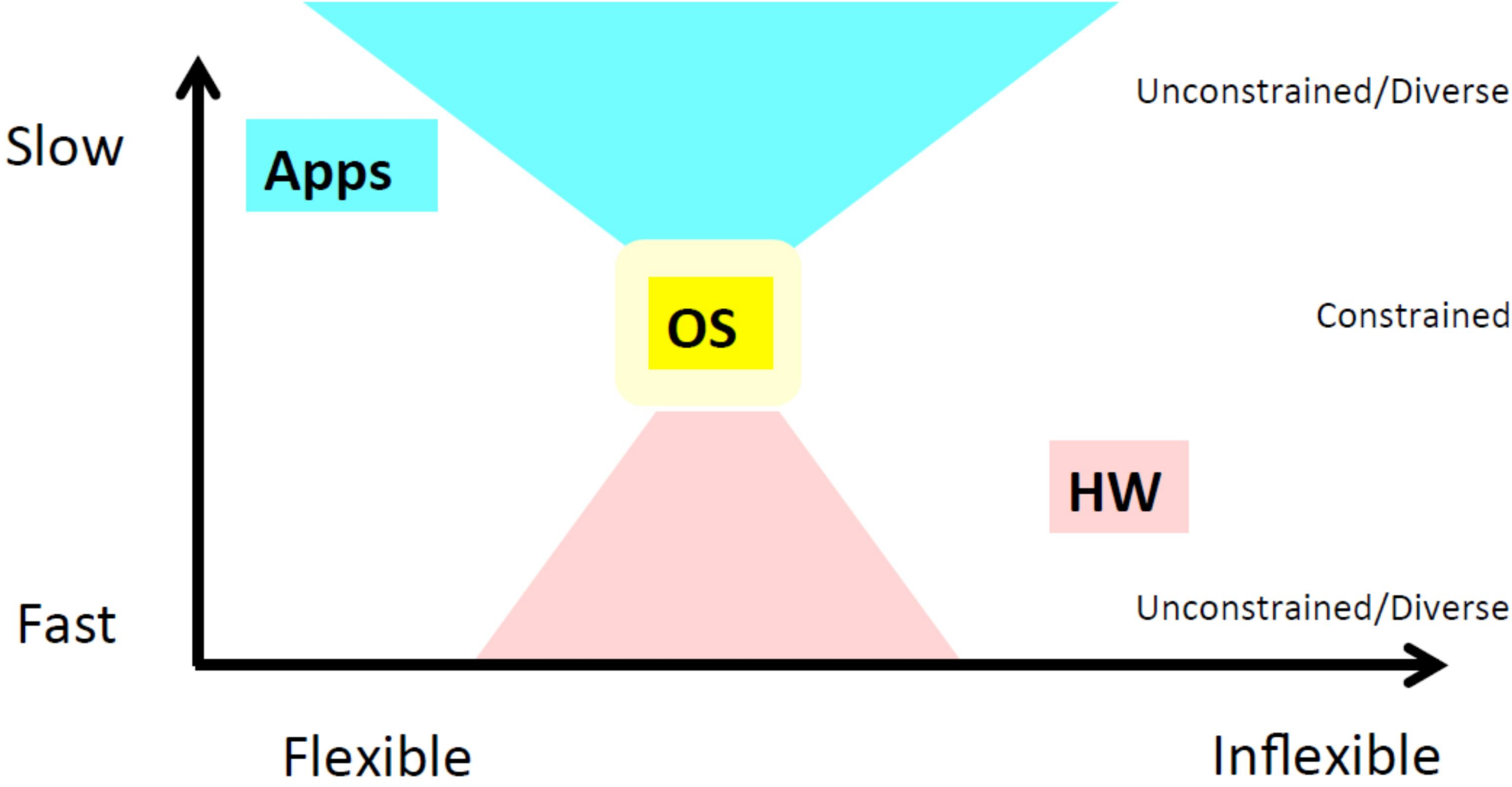
Special

Architecture (*constraints that deconstrain*)

Slide courtesy: D. Meyer

Original slide courtesy John Doyle

Overlaying Tradeoffs



Unconstrained/Diverse

Slow

Apps

OS

Constrained

HW

Fast

Unconstrained/Diverse

Flexible

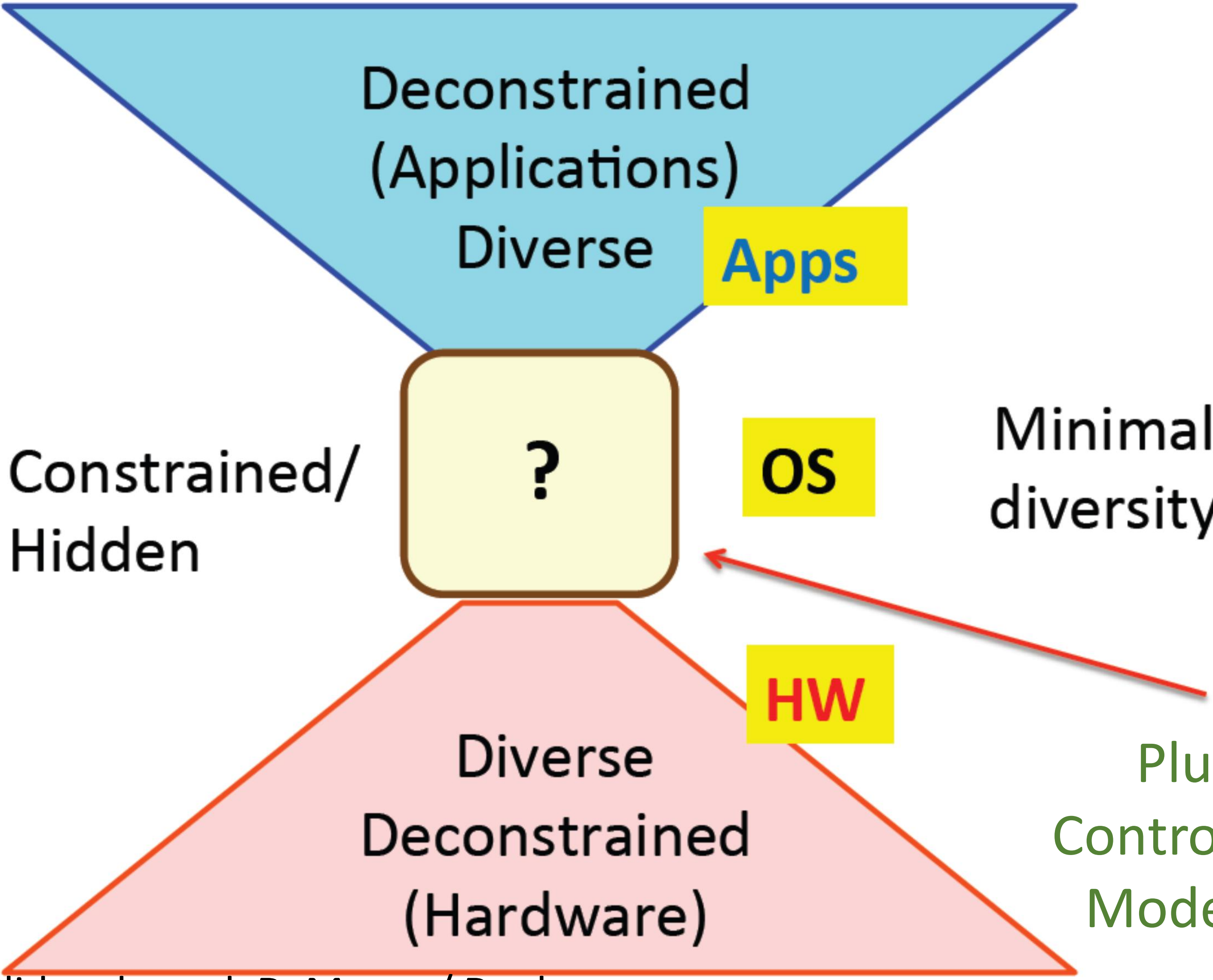
Inflexible

General

Slide courtesy: D. Meyer

Special

Layered architectures make Robustness and Evolvability *compatible*



Deconstrained (Applications) Diverse

Apps

Constrained/ Hidden

?

OS

Minimal diversity

Think OS Kernel + Libraries and Drivers

HW

Diverse Deconstrained (Hardware)

SDN Controller?

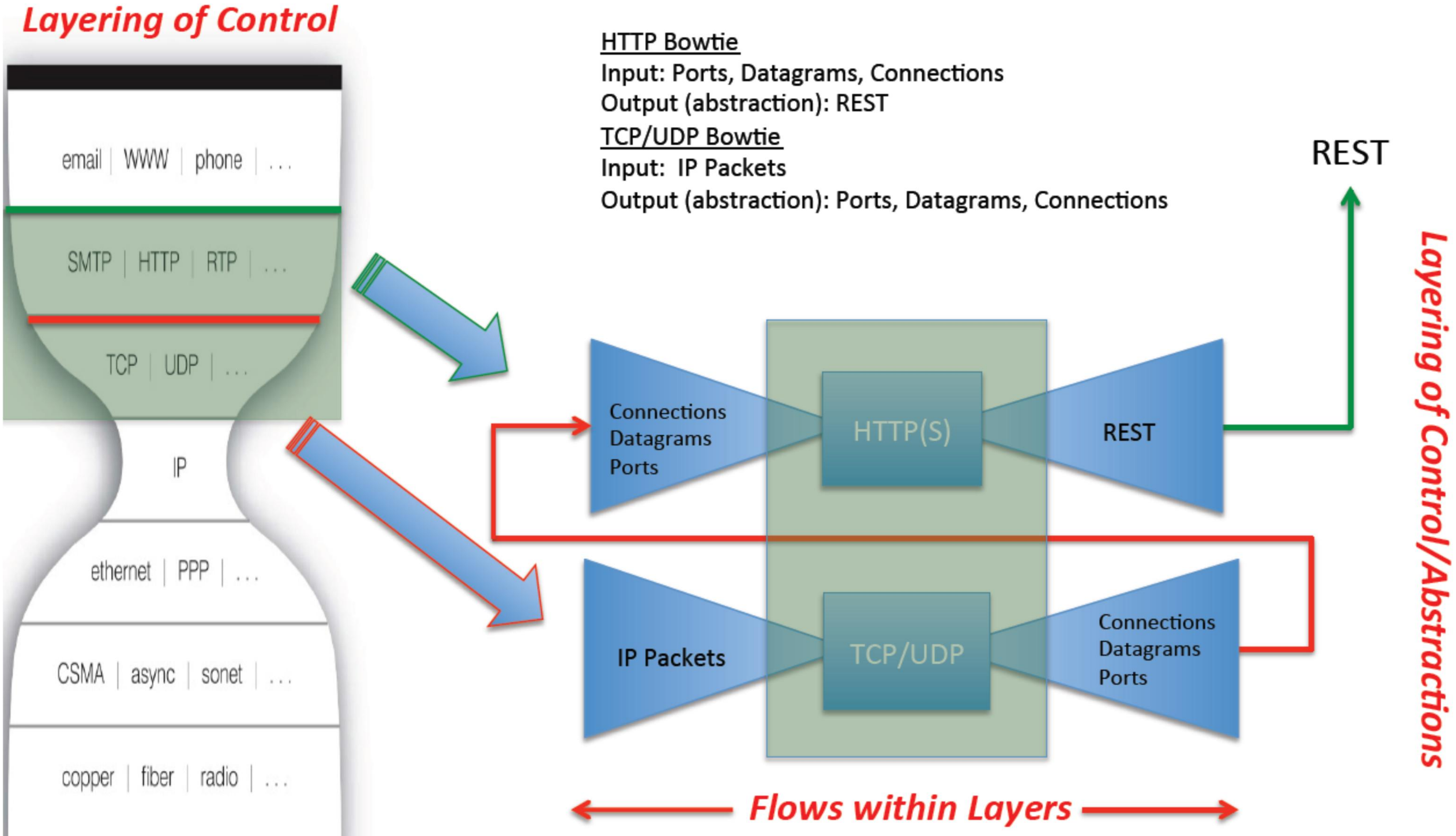
Plug-in Architectures

Control Plane Drivers + APIs

Model-Driven Everything

Slide adapted: D. Meyer / Doyle

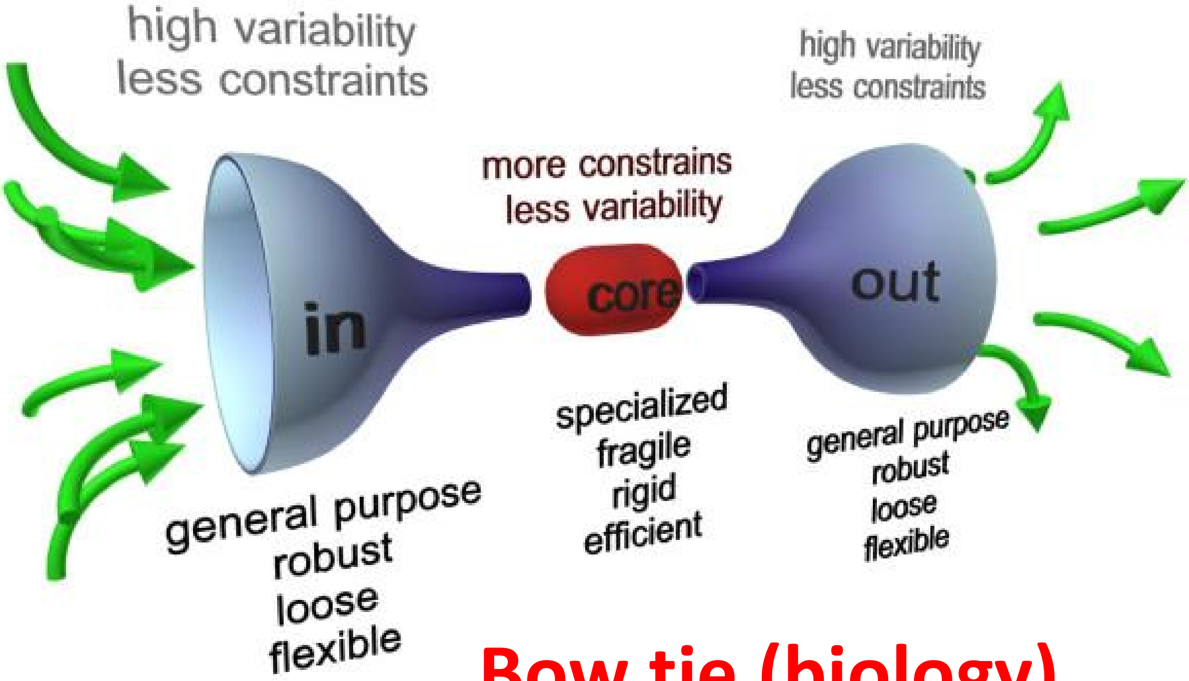
The Nested Bowtie/Hourglass Architecture of the Internet



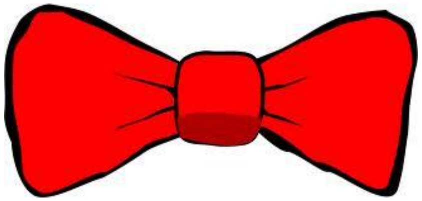
<u>HTTP Bowtie</u>
Input: Ports, Datagrams, Connections
Output (abstraction): REST
<u>TCP/UDP Bowtie</u>
Input: IP Packets
Output (abstraction): Ports, Datagrams, Connections

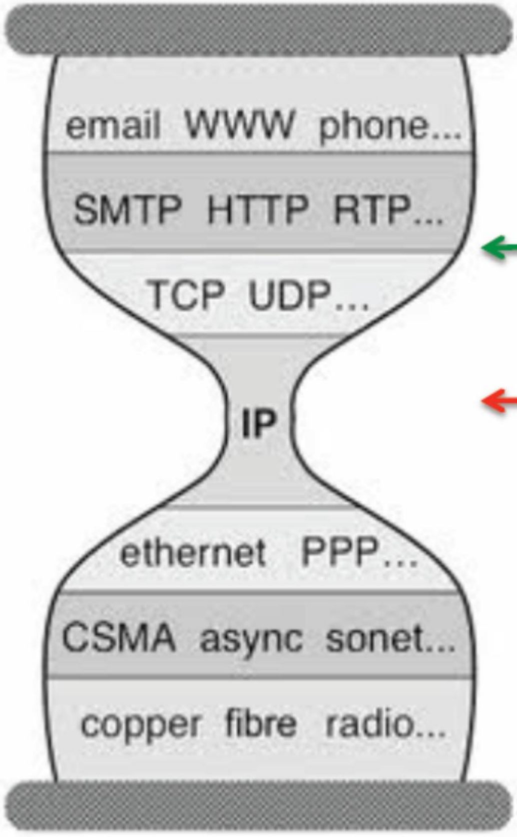
Slide courtesy: D. Meyer

Bowties /Hourglasses?



Bow tie (biology)





OL/SDN

CP/SDN

OF/SDN?

Open Source is a wildcard

Adapted Slide courtesy: D. Meyer

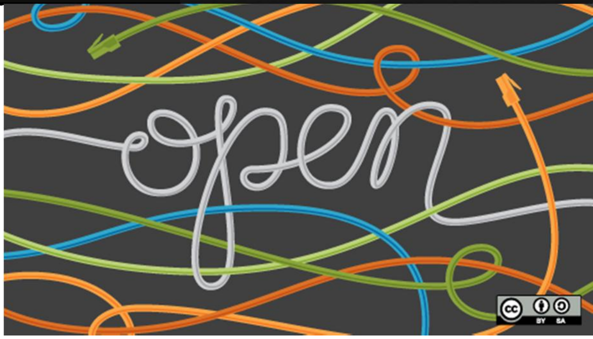
Open Source & David vs. Goliaths

- In order for the little guy to compete,
  - As much a business challenge as a technology challenge
- Big companies have huge distribution channels
- How do you get the reach of a big company without the resources?

Open Source as a distribution model

- Barrier to entry is low
- Early adopters have more patience
- Community based development
- Features come quicker than going it along
- Easier to build partnerships and alliances

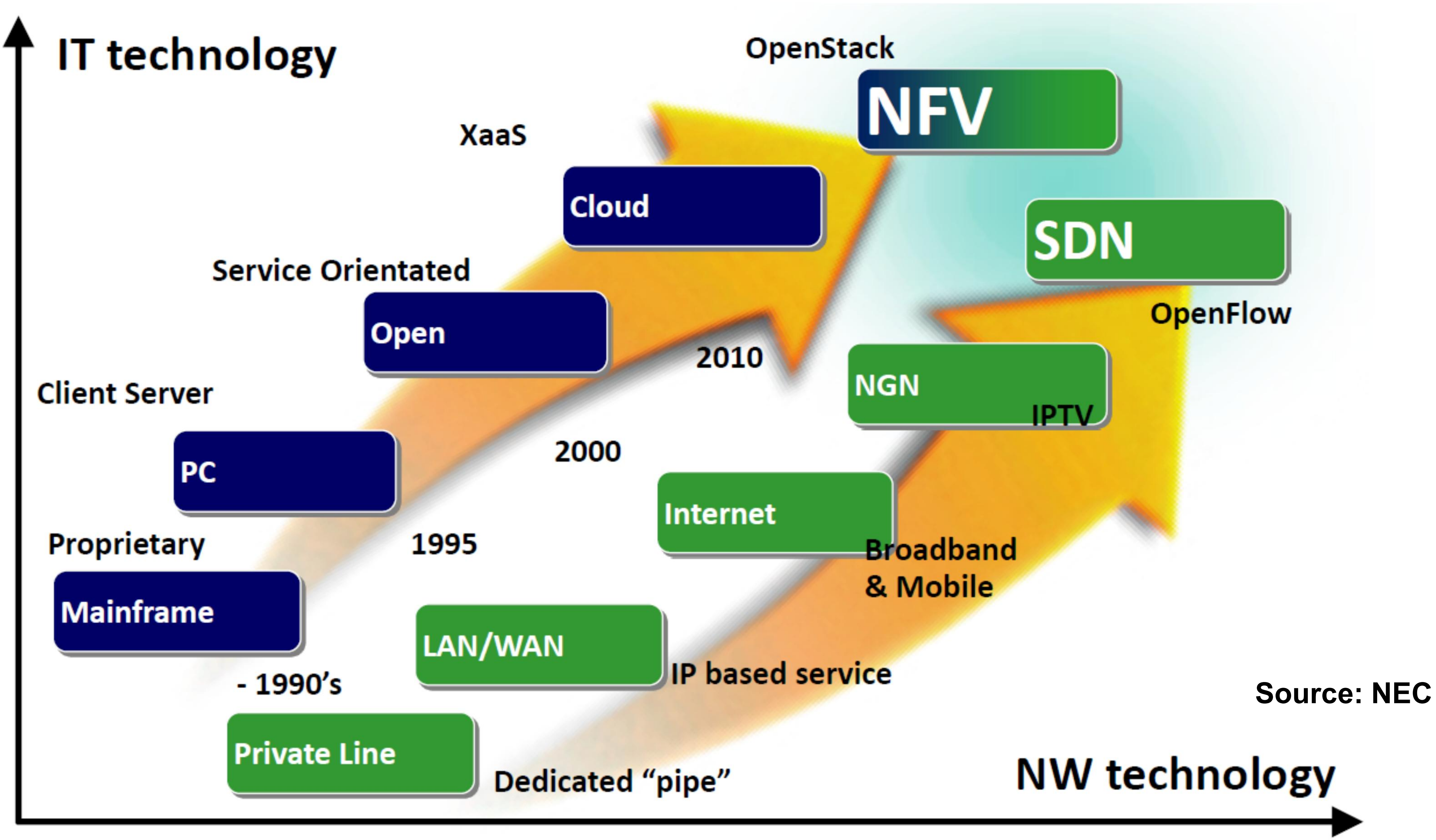
Why open SD(x) will prevail



- No one wants proprietary lock in
- Open is first class citizen in the infrastructure software world
  - OpenStack
  - Hadoop
  - OpenDayLight
- Large companies now have policies on how to adopt open source
- Strength of partners

Concluding remarks

IT & Networking Growing Together



Source: NEC

So, why we need/want NFV(/SDN)?

1. Virtualization: Use network resource without worrying about where it is physically located, how much it is, how it is organized, etc.

2. Orchestration: Manage thousands of devices

3. Programmable: Should be able to change behavior on the fly.

4. Dynamic Scaling: Should be able to change size, quantity, as a F(load)

5. Automation: Let machines / software do humans' work

6. Visibility: Monitor resources, connectivity

7. Performance: Optimize network device utilization

8. Multi-tenancy: Slice the network for different customers (as-a-Service)

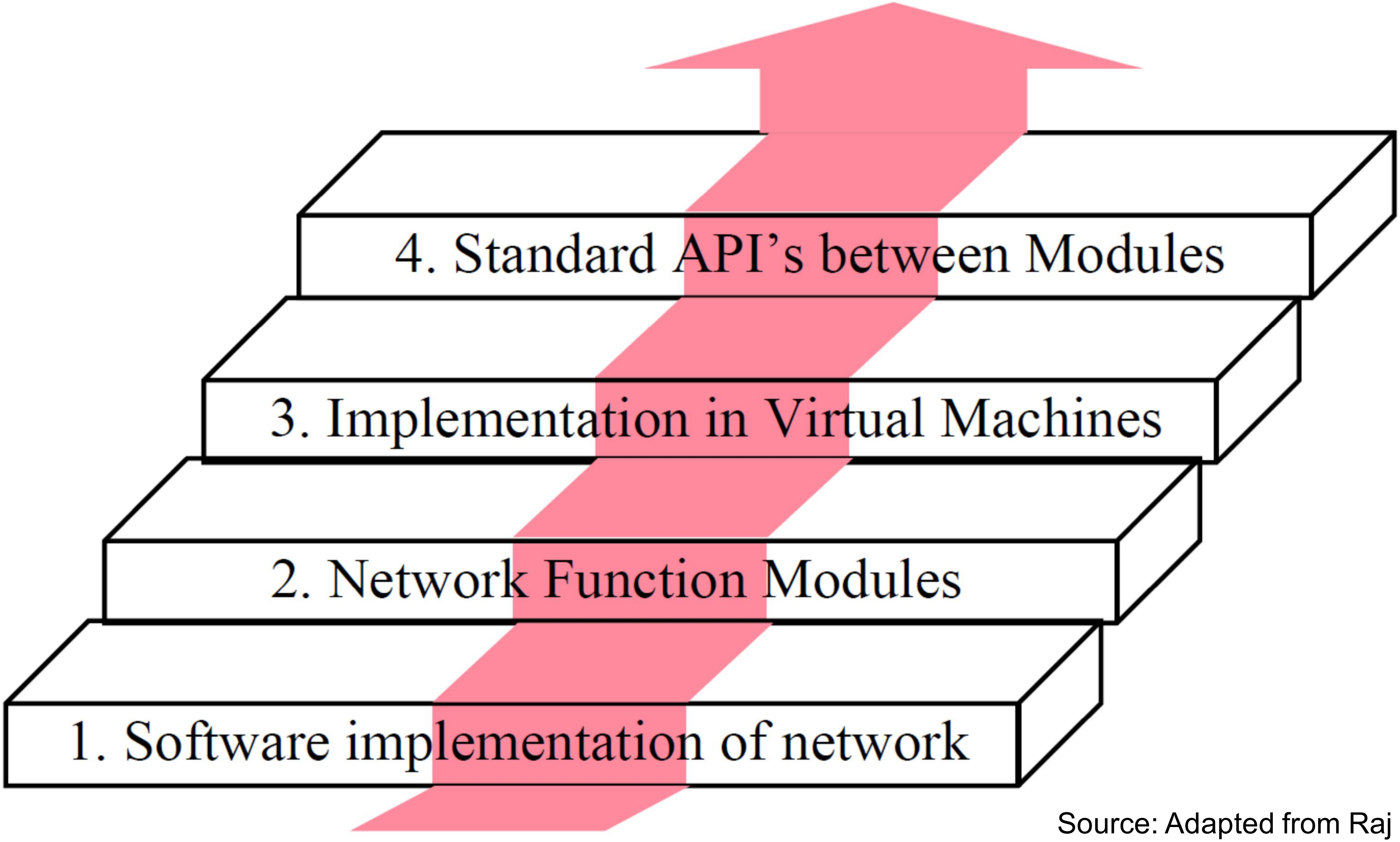
9. Service Integration: Let network management play nice with OSS/BSS

10. Openness: Full choice of modular plug-ins

Note: These are exactly the same reasons why we need/want SDN.

Source: Adapted from Raj Jain

Wrapping up: Innovations of NFV



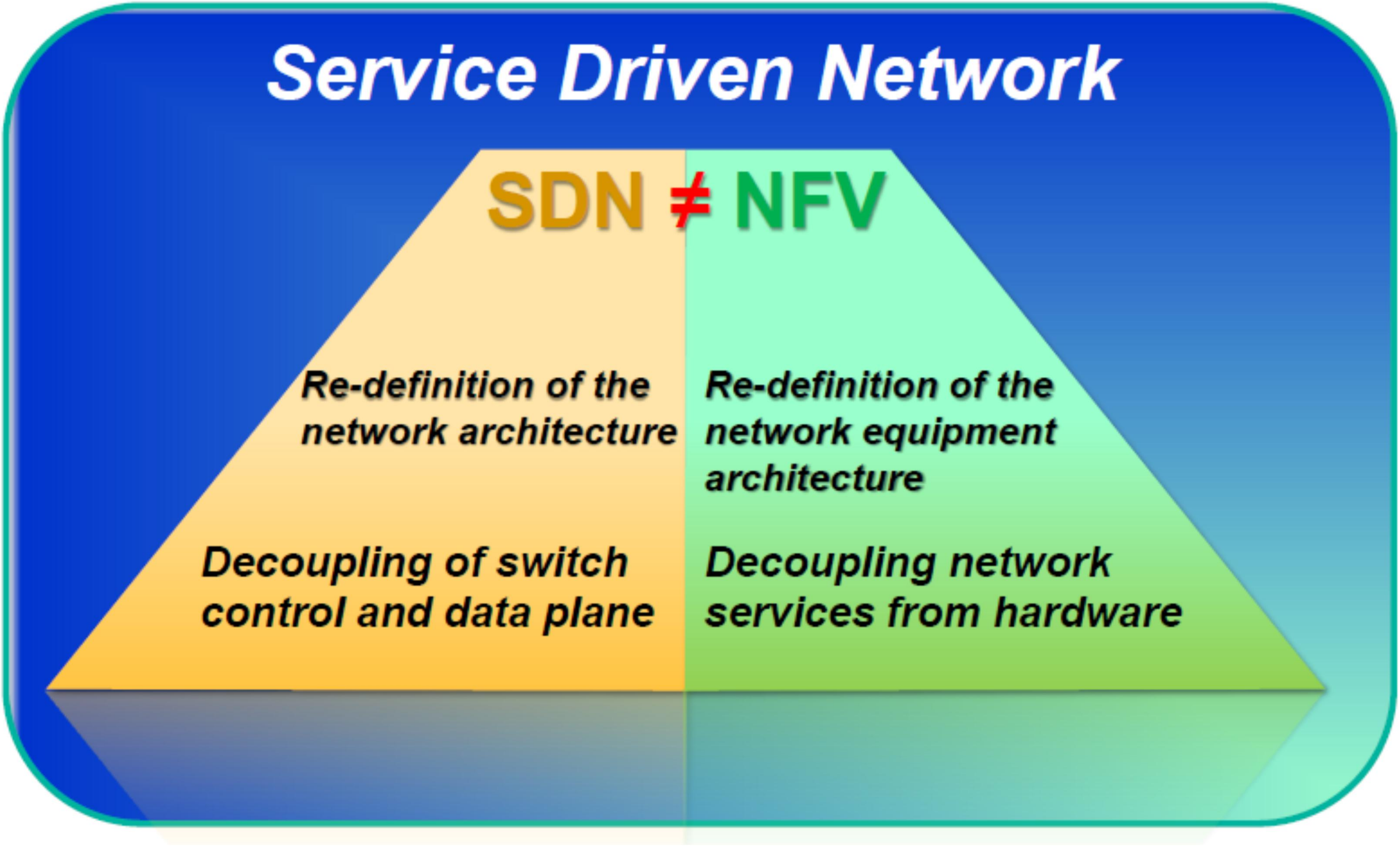
4. Standard API's between Modules

3. Implementation in Virtual Machines

2. Network Function Modules

1. Software implementation of network

Source: Adapted from Raj Jain



Service Driven Network

SDN ≠ NFV

Re-definition of the network architecture

Re-definition of the network equipment architecture

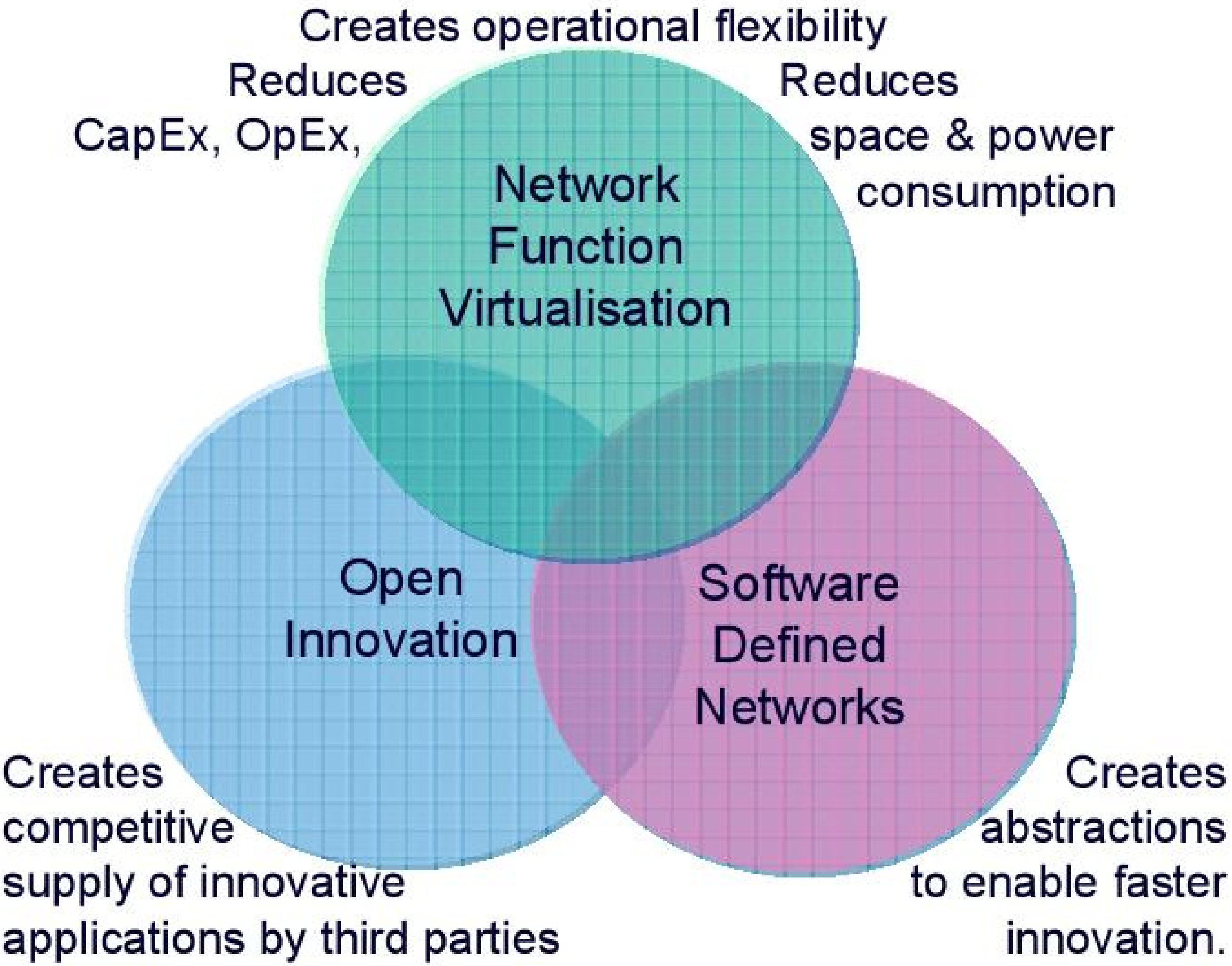
Decoupling of switch control and data plane

Decoupling network services from hardware

Source: NEC

SDN & NFV

Strategic Networking Paradigms for Network Operators



Creates operational flexibility

Reduces CapEx, OpEx,

Reduces space & power consumption

Network Function Virtualisation

Open Innovation

Software Defined Networks

Creates competitive supply of innovative applications by third parties

Creates abstractions to enable faster innovation.

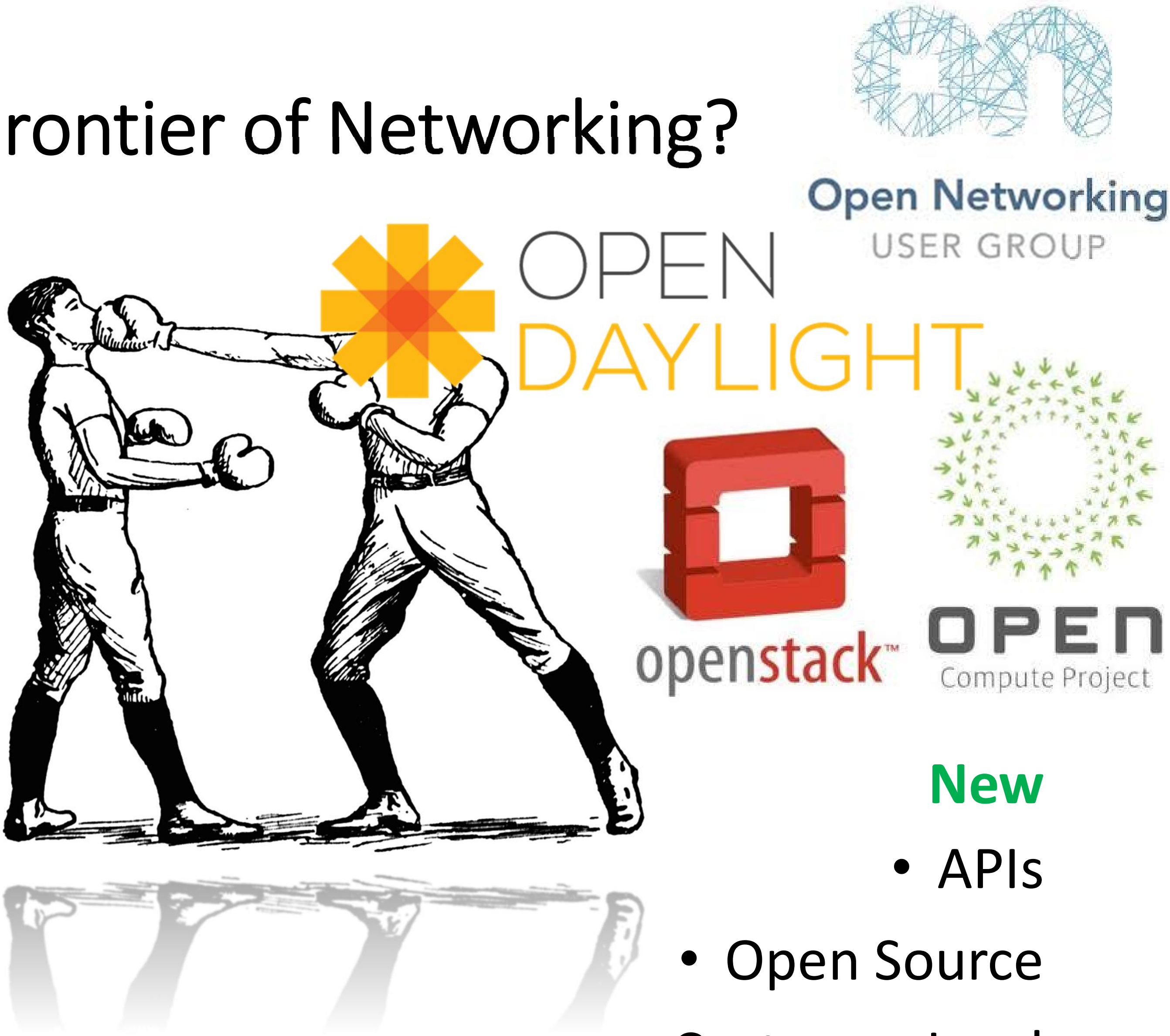
Source:NFV

The increasing role of software

- So, my network device should be cheaper over time. But where is the money going?
  - If the hardware needed to forward Terabits is actually commodity (from merchant silicon to merchant optics?)
- That means what you’re actually buying is software.
  - Software is hard.
  - Routing protocols, CLIs, network management platforms, and feature after feature after feature after feature…
- Software is what you’re actually buying.
  - The hardware is just a delivery vehicle, so you don’t feel so bad for spending millions on invisible electrons.
  - But the software is what you actually care about.

Source (adapted from):
https://www.nanog.org/sites/default/files/wednesday.general.temkin.panel.pdf

SDN & NFV: The Frontier of Networking?



Existing

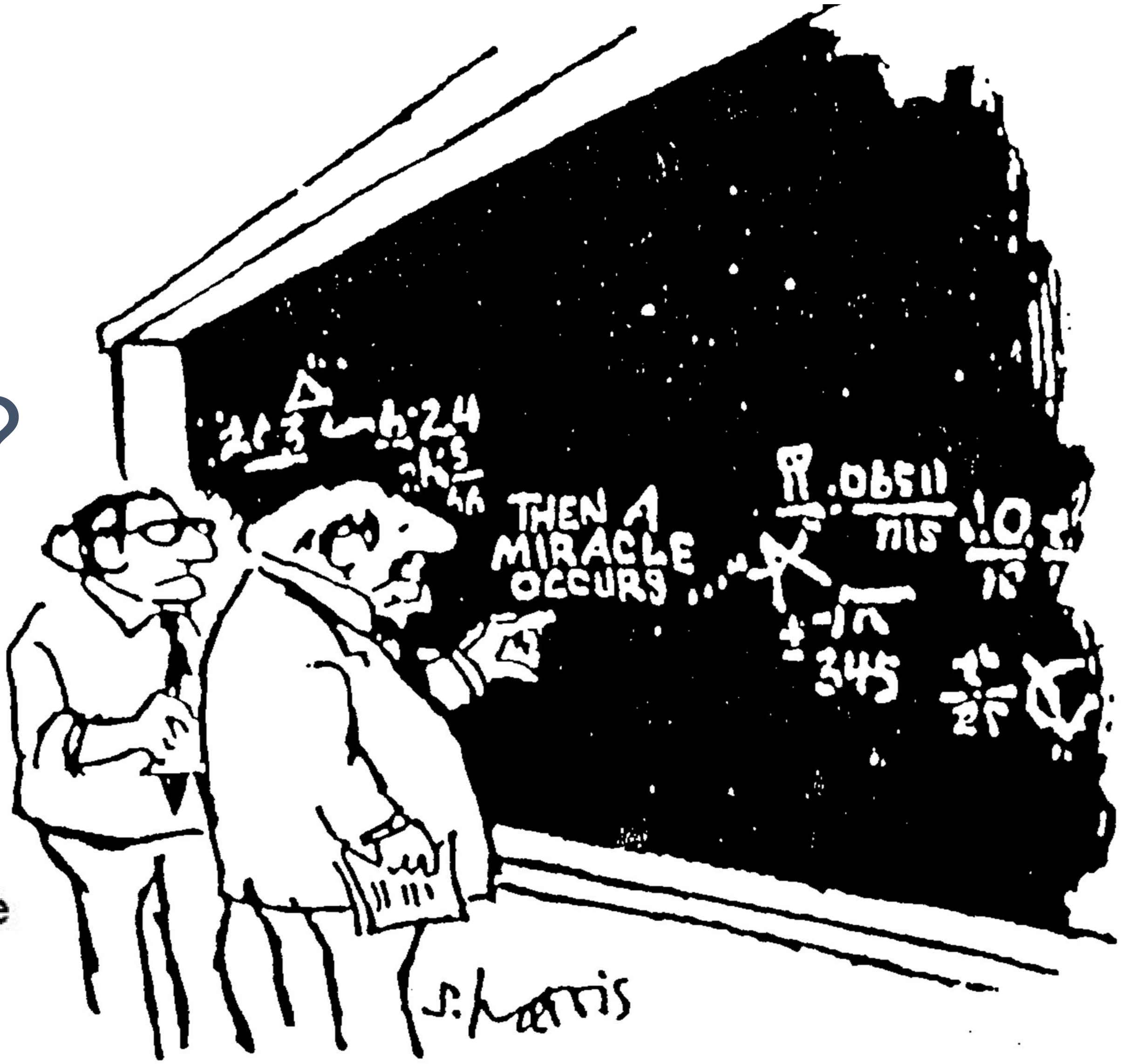
- CLIs
- Closed Source
- Vendor Lead
- Classic Network Appliances

New

- APIs
- Open Source
- Customer Lead
- Network Function Virtualization (NFV)

Adapted from: Kyle Mestery, Next Generation Network Developer Skills

Obrigado!
(mais)
Perguntas?



"I think you should be more explicit here in step two"

Further reading:
"**Software-Defined Networking: A Comprehensive Survey**".
To appear in Proceedings of the IEEE, 2015. Available online: http://arxiv.org/abs/1406.0440

Contributions welcome:
https://github.com/SDN-Survey/latex/wiki

Credits

- Arpit Joshipura, Dell. In SDN World Congress Keynote-SDN and NFV: Twins or Distant Cousins?
- Ben Cherian. Software Defined Storage vs Software Defined Networking: a comparison. Slides on Open SDx & David vs. Goliaths
- David vs Goliath. Figure. http://www.tiopan.com/~jonahs/ot-event.html
- See footnotes

References, Further Readings & Credits

- Alderson, David L., and John C. Doyle. "Contrasting views of complexity and their implications for network-centric infrastructures." *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on* 40.4 (2010): 839-852.
- David Meyer, Macro Trends, Architecture, and the Hidden Nature of Complexity (and what does this have to do with SDN?) , Work in Progress. http://www.1-4-5.net/~dmm/talks/macro_trends_complexity_and_sdn.pdf
- Crowcroft, Jon, et al. "Is SDN the de-constraining constraint of the future internet?." *ACM SIGCOMM Computer Communication Review* 43.5 (2013): 13-18.
- Kyle Mestery, Next Generation Network Developer Skills, http://www.slideshare.net/mestery/next-gennetworkengineerskills
- Miguel Ponce de Leon, Open Source & Research in EU FP7, http://www.slideshare.net/miguelpdl/2010-10-19-open-source-research-in-fp7-future-networks
- Stanford Networking Seminar: Teemu Koponen, VM Ware - Structure And Design Of Software-Defined Networks, http://netseminar.stanford.edu
- RouteFlow, http://cpqd.github.io/RouteFlow

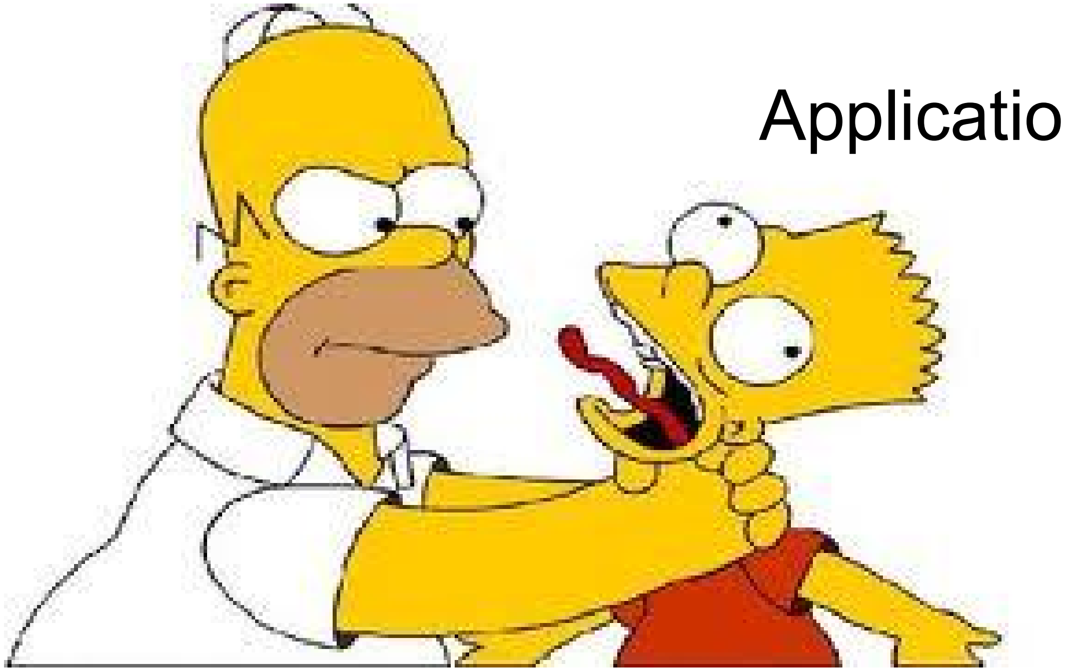
SDN & Single Throat to Choke

Who provides solution support in a decoupled SDN???

Switch Vendor?

Controller Provider?

Application Developer?

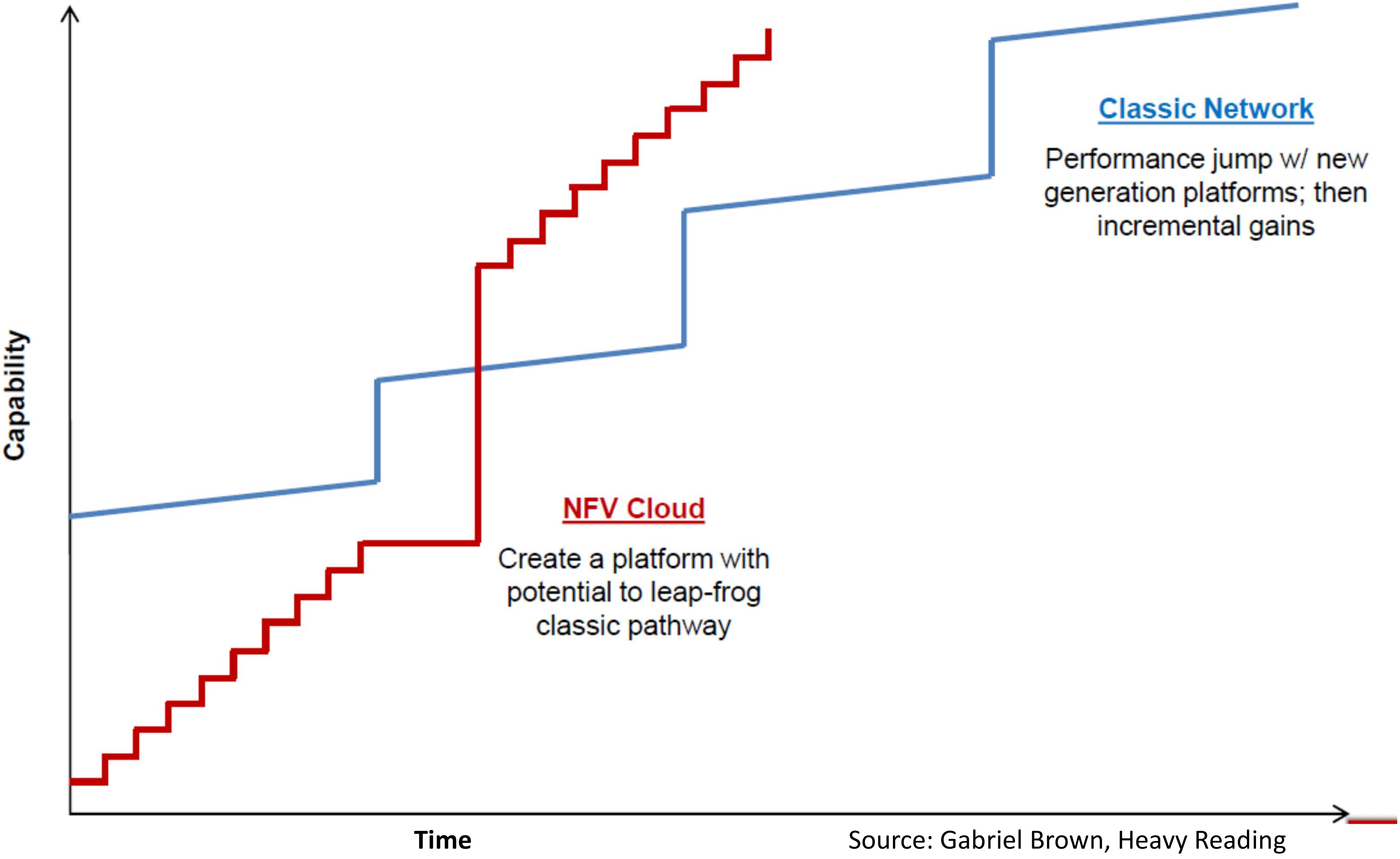


Cenário atual

- Não existe solução que traga todas as características de SDS
  - Porém há varios produtos com uma ou várias delas.

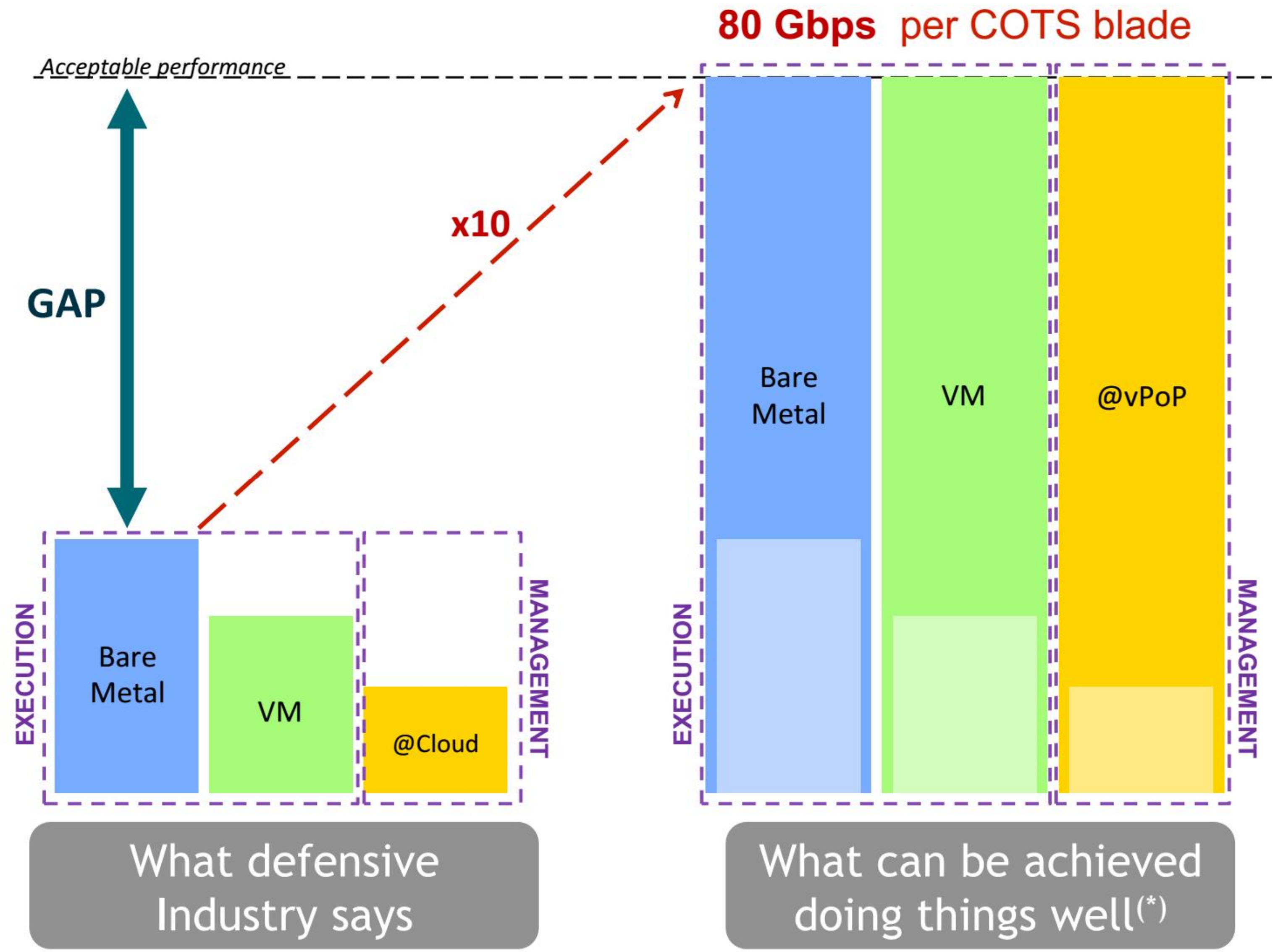
Produto	Característica
VMWare VSAN	Políticas, Storage Pool, Scale-out
StarWind	Data protection, reporting
VSANSymphony	Storage Pool, data protection
EMC’s ViPR	Storage Pool, Scale-out, HW commodity

Is NFV Technology Good Enough?



Source: Gabriel Brown, Heavy Reading

High and Predictable Performance is Achievable



(*) ETSI NFV Work Item "NFV Performance & Portability Best Practises": DGS/NFV-PER001 Current version: v0.0.7 (stable draft – 15/10/2013)

Some Use Case Examples

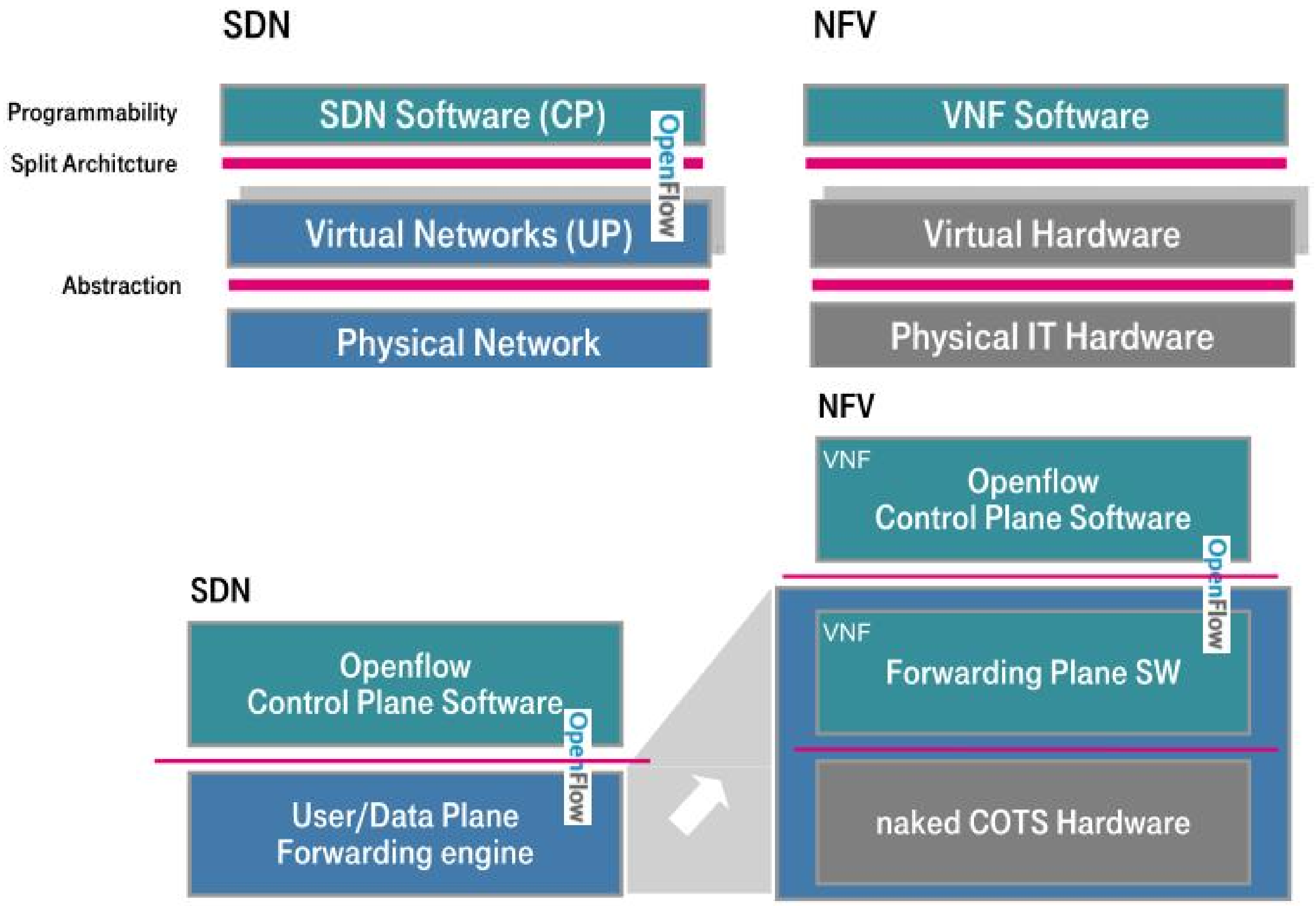
…not in any particular order

- Switching elements: BNG, CG-NAT, routers.
- Mobile network nodes: HLR/HSS, MME, SGSN, GGSN/PDN-GW.
- Home networks: Functions contained in home routers and set top boxes to create virtualised home environments.
- Tunnelling gateway elements: IPSec/SSL VPN gateways.
- Traffic analysis: DPI, QoE measurement.
- Service Assurance: SLA monitoring, Test and Diagnostics.
- NGN signalling: SBCs, IMS.
- Converged and network-wide functions: AAA servers, policy control and charging platforms.
- Application-level optimisation: CDNs, Cache Servers, Load Balancers, Application Accelerators.
- Security functions: Firewalls, virus scanners, intrusion detection systems, spam protection.

Source: NFV

SDN and NFV

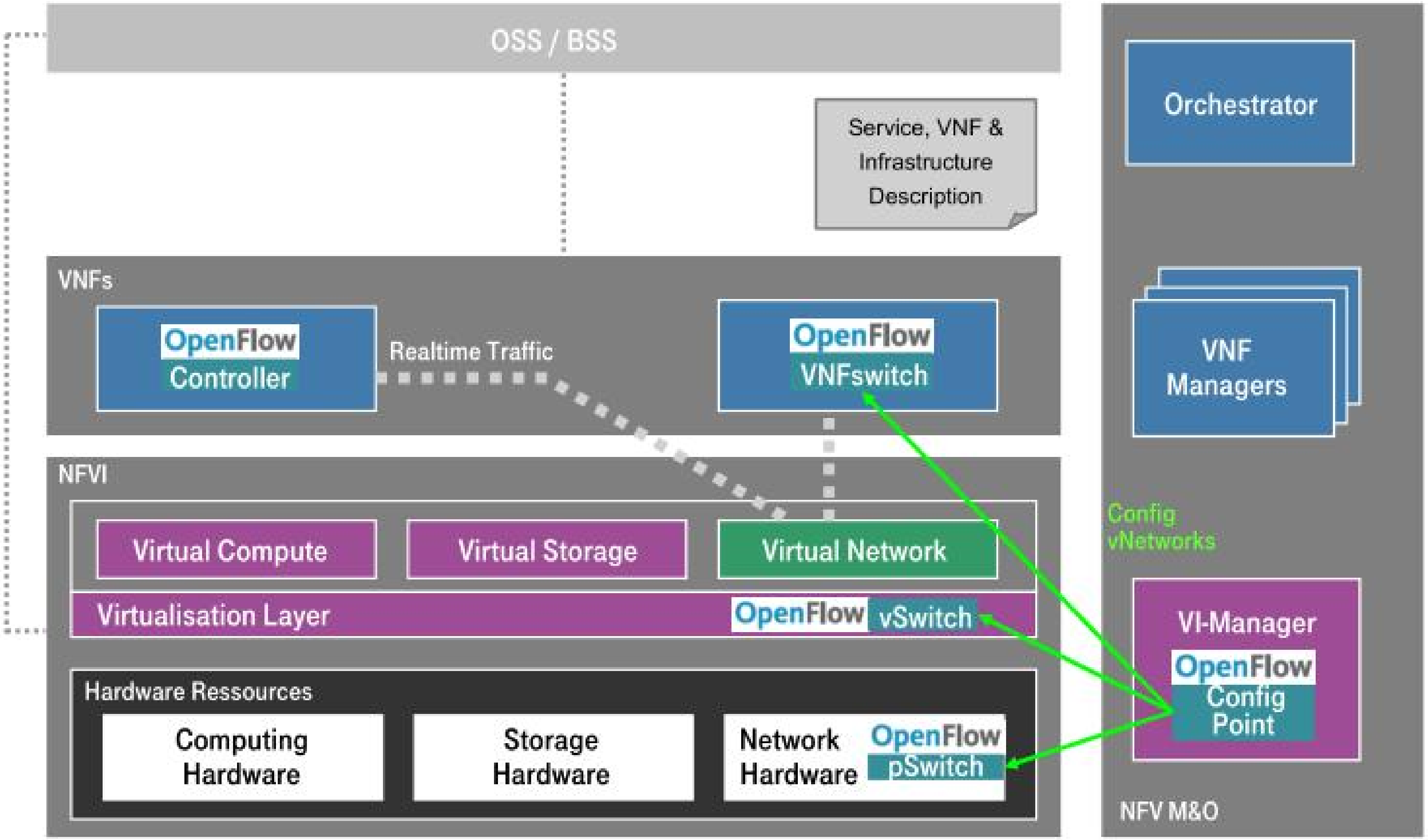
- SDN and NFV do NOT depend on each other



Source: Uwe Michel, T-Systems

SDN and NFV

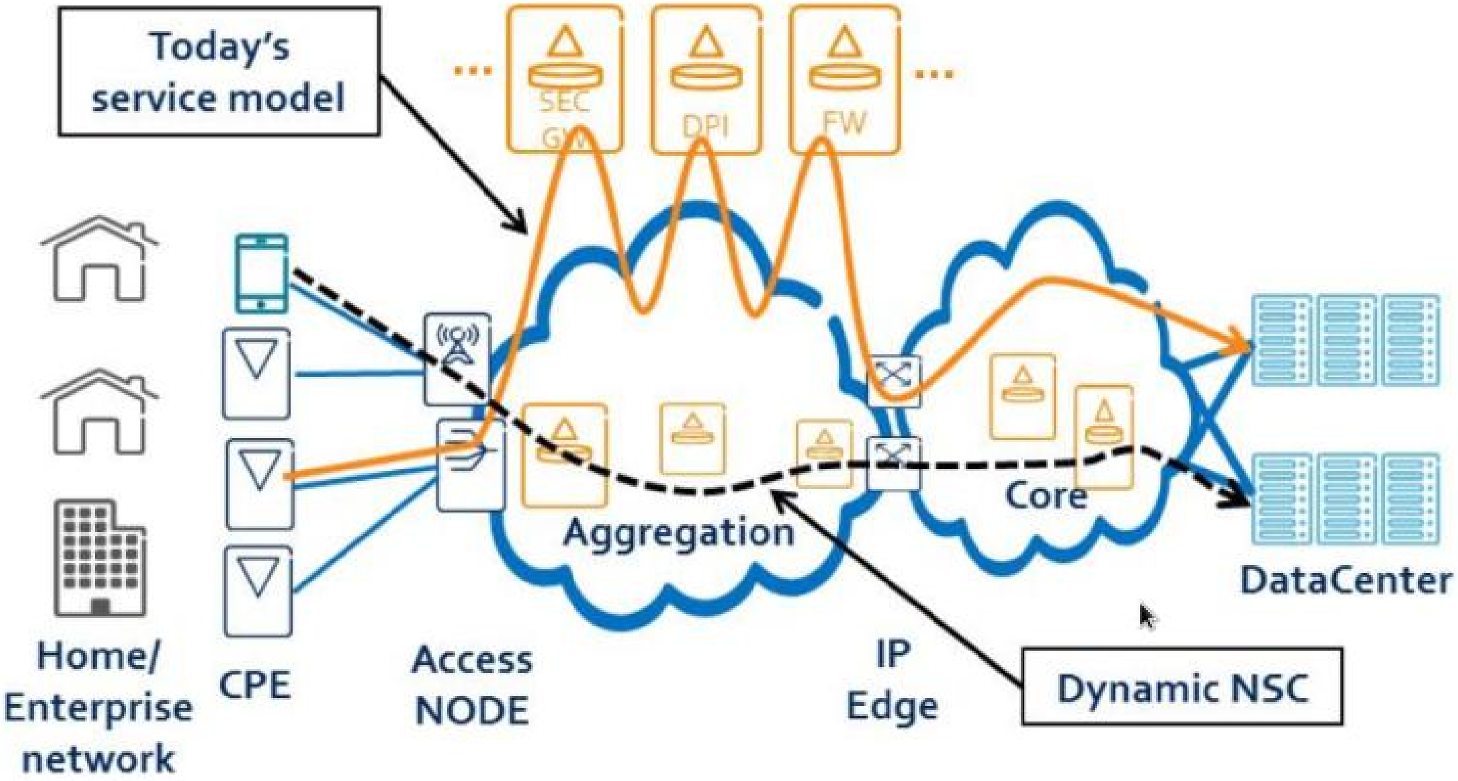
- SDN poses to NFV:
  - Central point of contact / Orchestrate VNFs (NSC)



Source:
Uwe Michel
T-Systems

NFV Forwarding Graphs

- Network Service Chaining
  - Networks paths: old stratified vs. dynamic new



Source: Ericsson, EU UNIFY