On the role of open source in future networking technologies and research

Prof. Dr. Christian Esteve Rothenberg
chesteve@dca.fee.unicamp.br
FEEC/UNICAMP
18/March/2014

COPELABS BRAINstorming (C-BRAIN) #50:

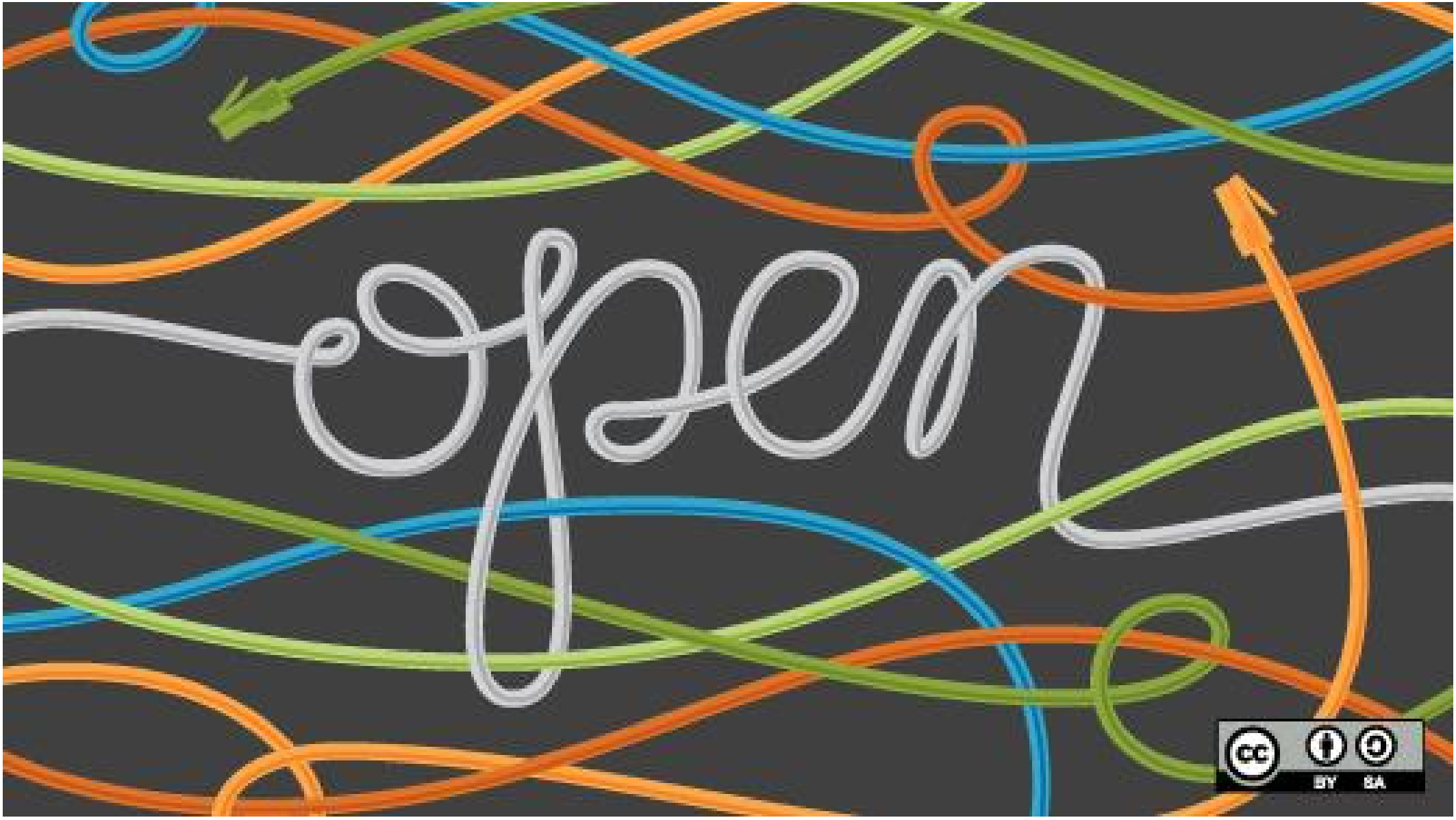


Image by opensource.com

Agenda

- Open Source met Networking
- When Open Source meets Network Control Planes
  - Software-Defined Networking and OpenFlow
  - The Open Frontier of Networking
- Detour on Layered Architectures and Complexity
- FLOSS Experiences from the RouteFlow project

Open Source Met Networking

So far, mainly software appliances with limited scope

Open Source & Networking

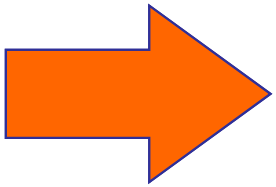
Functional level	FLOSS solutions	Overview	Legacy proprietary approach
Availability	Pacemaker, HAProxy, NGINX	Deployment and management of redundant, high-availability clusters	Network Load Balancing appliance
	Nagios, Cacti, Zabbix	Operation and performance monitoring software	Proprietary monitoring software
Security	IPCop, PFSense, Smoothwall, M0n0wall	Linux and BSD-based firewall systems	Firewall and IDS appliances
	OSSEC, Snort	Host and network-based Intrusion Detection Systems	
	SecurityOnion, OSSIM	Security Information Enterprise Management	
	PacketFence	Network Access Control (802.1x, captive portal, etc.)	NAC appliance
Management	cfEngine, Ansible, Puppet, Chef, mcollective, SaltStack	Configuration management and/or orchestration platforms	Proprietary, vendor-attached provisioning and management software
	SpaceWalk, Foreman	Server provisioning and configuration management	
Caching	Squid, Apache Traffic Server, Varnish Cache	Caching for performance at either client end or server end	Caching appliance
Storage	DRBD, GFS2, GlusterFS	Distributed and/or replicated filesystems	Storage appliance
	FreeNAS, OpenFiler	Open source storage management and sharing (iSCSI, NFS, etc.)	
Network Services (IPAM etc)	BIND, djbdns, TinyDNS, DNSmasq	Network services that provide directory services and address management services	---
Routing	Quagga, XORP, BIRD	Routing protocols implementation suites	Vendor lock between route engine and dedicated silicon router
Device OS / firmware	DD-WRT, OpenWRT, Gargoyle, PicOS	Open source firmware for wireless access points and Ethernet switches	Proprietary wireless firmware/protocols
Server virtualization	OpenVZ, LXC	OS kernel virtualization (container-based emulation)	Underutilized server hardware
	Xen, KVM	Server hardware full and/or para-virtualization	

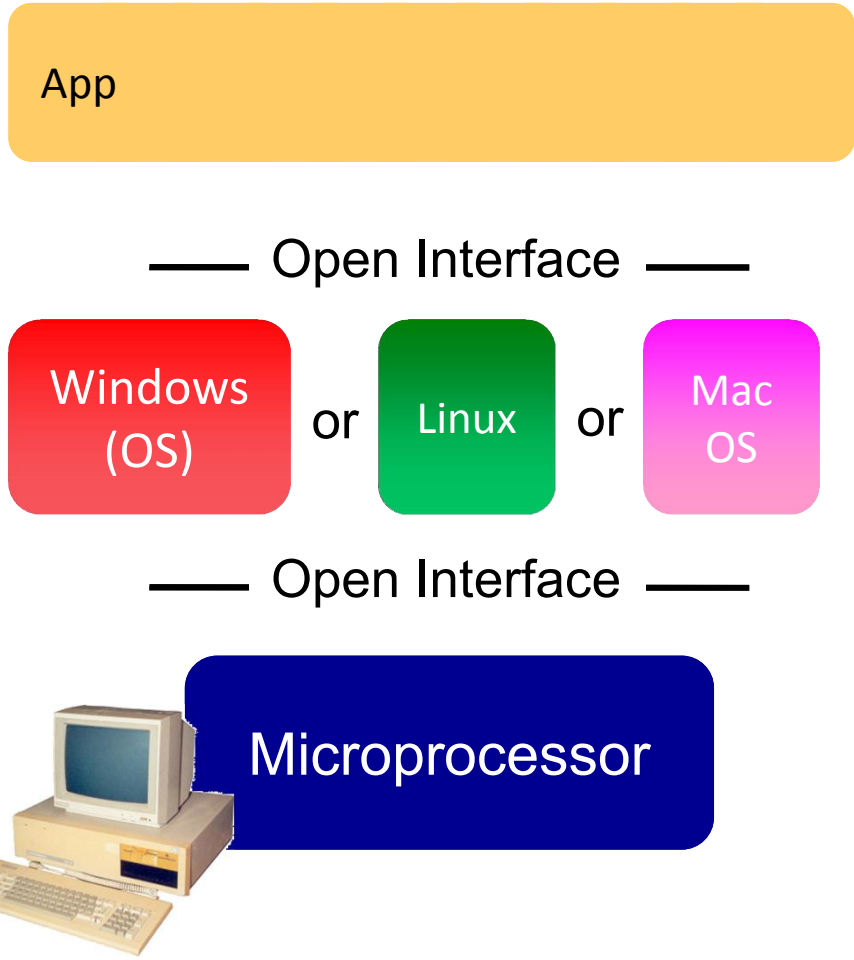
Open Source Meets Network Control

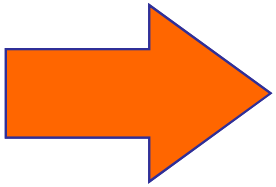
“SDN may be to networking what Linux was to computer OS”

The Computer Industry History

Vertically integrated
Closed, proprietary
Slow innovation
Small industry







Horizontal
Open interfaces
Rapid innovation
Huge industry

Source: N. McKeown

Software Defined Network (SDN)

App

Open Interface

Control Plane or Control Plane or Control Plane

Open Interface

Merchant Switching Chips

Vertically integrated
Closed, proprietary
Slow innovation

Horizontal
Open interfaces
Rapid innovation

Source: N. McKeown

Short Story: OpenFlow is an API

- Control how packets are forwarded (and manipulated)
- Implementable on COTS (Commercial-off-the-shelf) hardware
- Make deployed networks programmable
  - not just configurable
  - vendor-independent
- Makes innovation easier

- Goal (experimenter's perspective):
  - Validate experiments on deployed hardware with real traffic at line speed
- Goal (industry perspective):
  - Reduced equipment costs through commoditization and competition in the controller / application space
  - Customization and in-house (or 3rd party) development of new networking features (e.g. protocols).

Trend

"Mainframe"

Computer Industry

App | App | App

Windows (OS) | Linux | Mac OS

Virtualization layer

x86 (Computer)

Network Industry

App | App | App

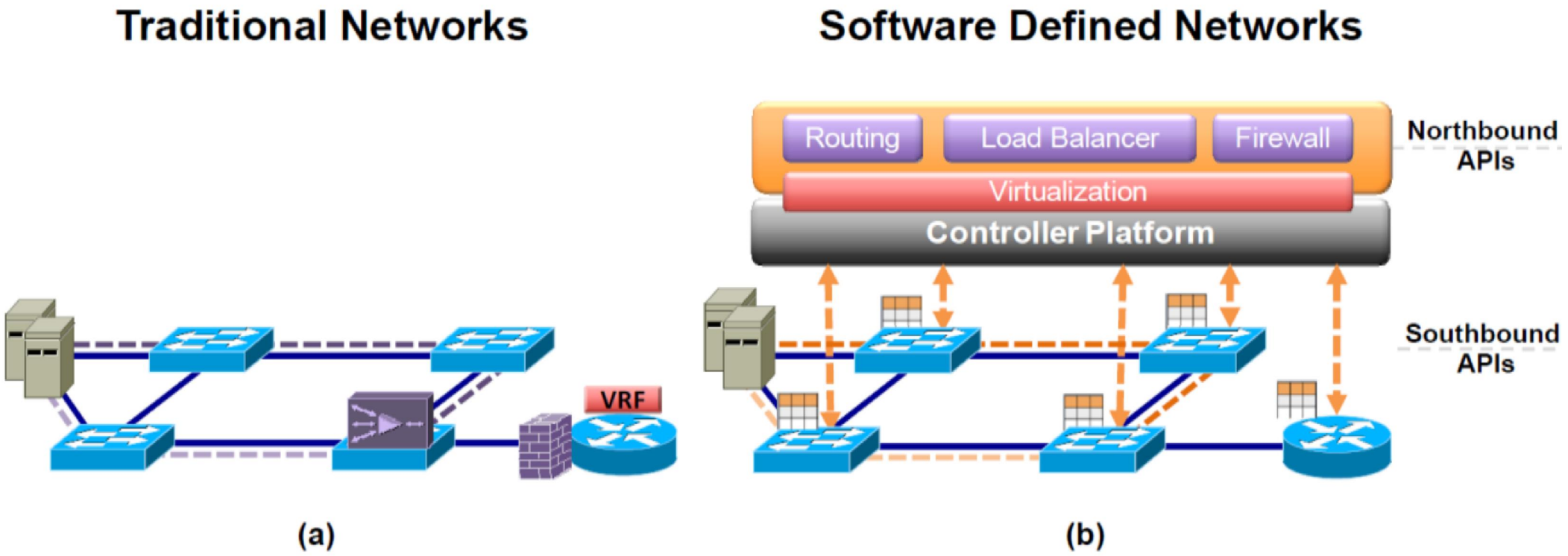
NOX (Network OS) | Network OS

Virtualization or "Slicing"

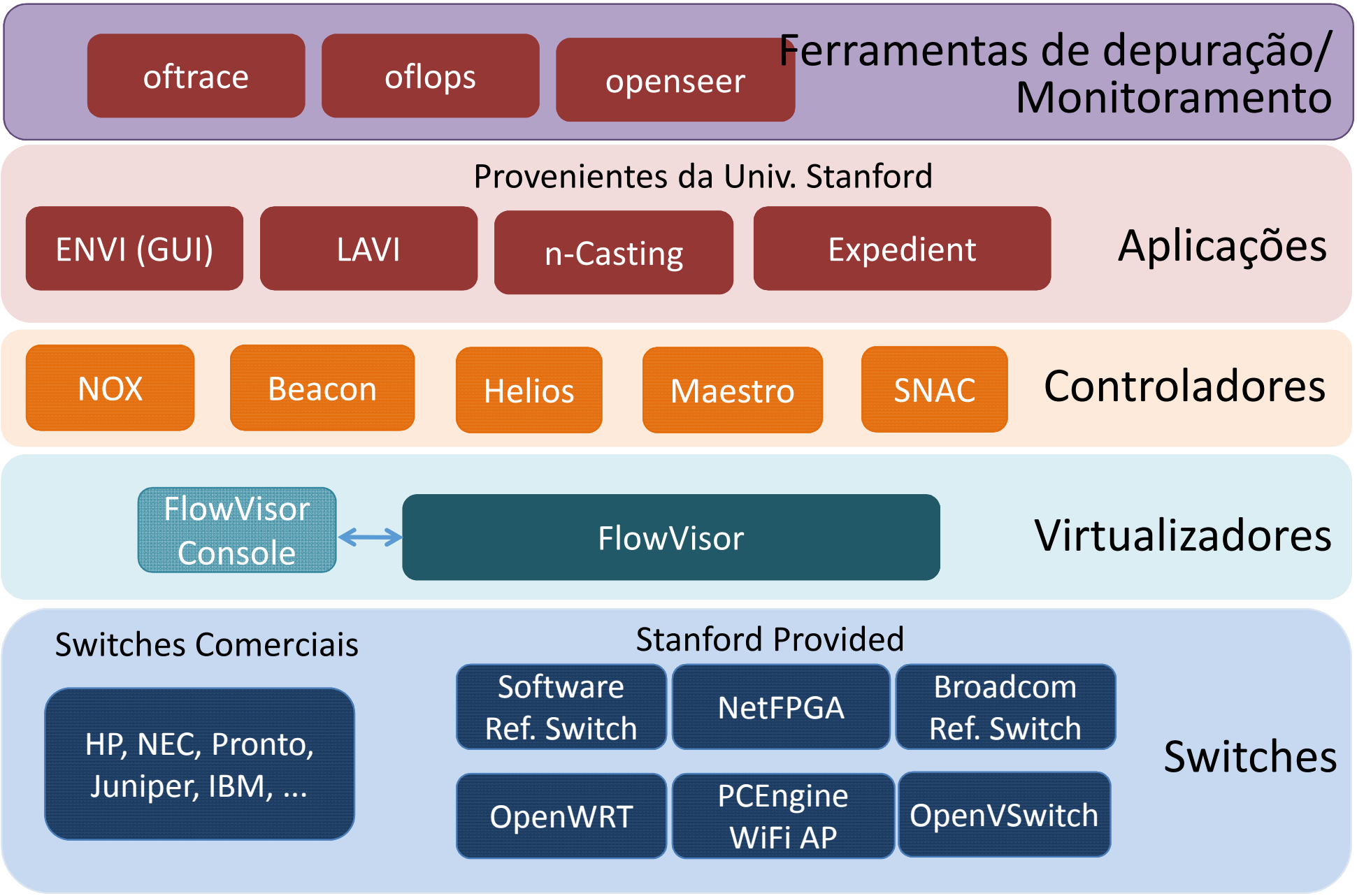
OpenFlow

Source: N. McKeown

Emerging SDN stack (FLOSS)



OpenFlow Ecosystem (around 2010)



Open Source & SDN (2013)

incomplete and already outdated

SDN stack layer / domain	Open Source Project Name
Benchmarking	Cbench (GPLv2, C/Perl/UNIX shell), OFLOPS (GPL, C)
Debugging / Testing / Simulation	ndb, OFRewind, STS (Apache, Python), OFDissector (BSD, C), liboftrace (BSD, C), OFTest (BSD, Python), Mininet (BSD, Python), fs-sdn (GPL, Python), ns-3 (GPL, C++)
Verification	Hassel (GPL, Python), NetPlumber (GPL, Python), NICE (BSD, Python), FlowChecker, OFTEN
Control & Management Apps	Topology discovery (GPL, many), HostTracker (GPL, many), Plug-n-Serve (BSD, C++), Aster*x (BSD, C++), FlowScale (Apache, Java), SNAC (Python/C++), Odin (Apache, Python), PANE (BSD, Haskell), FRESCO (GPL, Python/C++), OSCARS (BSD, Java), RouteFlow (Apache, Python/C/C++/Java)
Programming Abstractions / Compilers / Isolation	FatTire, Flog, FML, Frenetic (GPL, OCaml), HFT, NetCore, Nettle, Procera, Pyretic (BSD, Python), Maple (Pyhon), FlowN, LibNetVirt (GPL, C/Python), OpenStack Neutron (Apache,Python)
Controller Platforms	NOX (GPL, C++); POX (GPL, Python); Maestro (LGPL, Java); Beacon (GPL, Java); Floodlight (Apache, Java); Ryu (Apache, Python); Trema (GPL, C/Ruby); FlowER (MIT, Erlang); NodeFlow (GPL, javascript); Mul (GPL, C); RipL-POX (GPL, Python); OpenDaylight (Eclipse, Java)
Data Plane Virtualization	FlowVisor (BSD, Java), PortVirt (BSD, C), Expedient (Apache, Python)
OpenFlow Protocol Libraries / Driver Implementations	OFLib-Node (BSD, JavaScript), OpenFlowJ (BSD, Java), OpenFaucet (Apache, Python), Pylib-OpenFlow (BSD, Python/C++)
Data Plane Implementations	NetFPGA (BSD, C/Verilog), Open vSwitch (Apache, C), Reference design (BSD, C), ofsoftswitch13 (BSD, C), OpenWRT/Pantou (GPL, C), Switch Light (Eclipse, C), Indigo Virtual Switch (Eclipse, C), LINC-Switch (Apache, Erlang)

Open Source & Research Projects

- Help researchers to share openly important technological advances & disseminate results
- Commercialization avenue to exploit the project potential
- Means to sell “know how” and “knowledge”
- Statement for future Standardization
- Open source does NOT mean free.
  - Patents might still apply, “Not for commercial use” licenses, etc.

EU FP7 project evaluators care about proposing a convincing open source strategy (and executing it!)

Adapted from: Miguel Ponce, Open Source & Research in EU FP7

EU projects & open source

PROSE
Promoting Open Source in European Projects

:: Empowering FLOSS in European Projects
http://www.ict-prose.eu

- Projects using FLOSS (just a sample)

http://irati.eu

IRATI
INVESTIGATING RINA

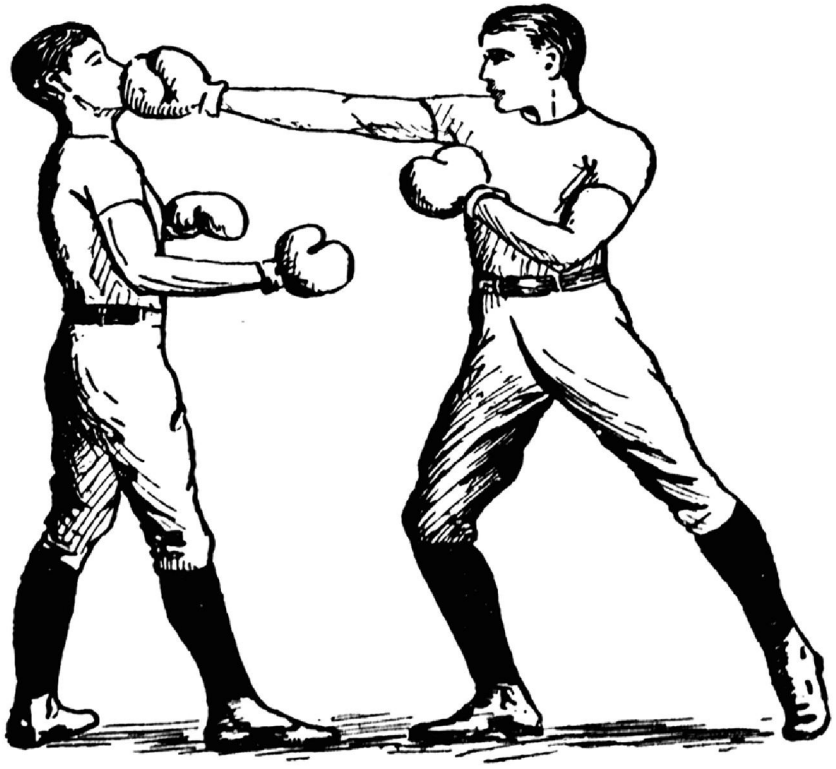
http://www.netinf.org

NetInf
Network of Information

http://www.opentinos.org

OpenTinos
A Network Protocol Experimentation Framework

The Frontier of Networking



Existing

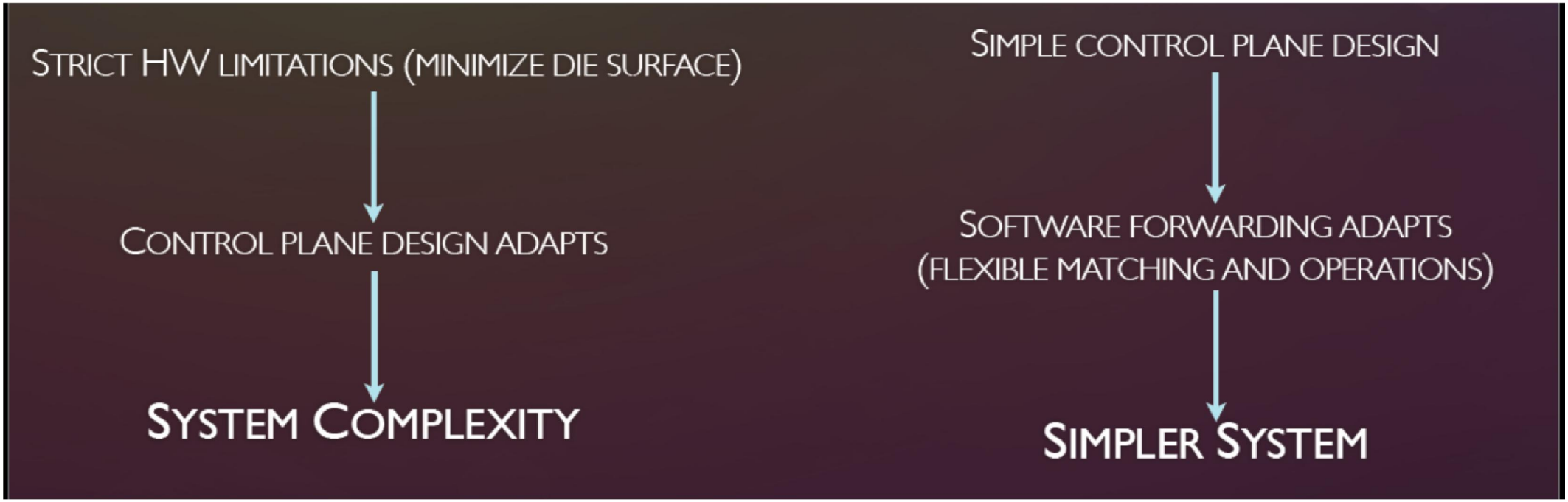
- CLIs
- Closed Source
- Vendor Lead
- Classic Network Appliances

New

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

Adapted from: Kyle Mestery, Next Generation Network Developer Skills

Change of Design Patterns (/Priorities)

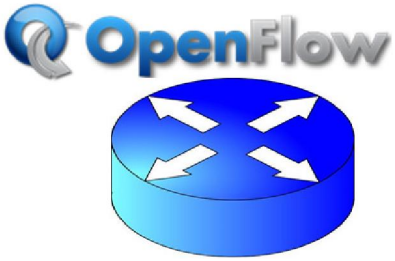


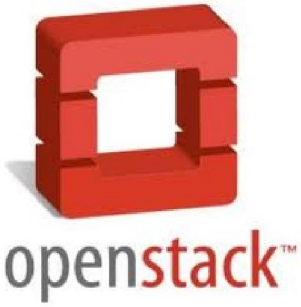
"I DON'T DESIGN PROTOCOLS. I WRITE C++"

-- TEEMU KOPONEN (NICIRA, VMWARE)

In invited lecture "STRUCTURE AND DESIGN OF SDN"

Components of the New Frontier





"We'd love to see OpenDaylight do for networking what Linux has done for the computing industry."

Adapted from: Kyle Mestery

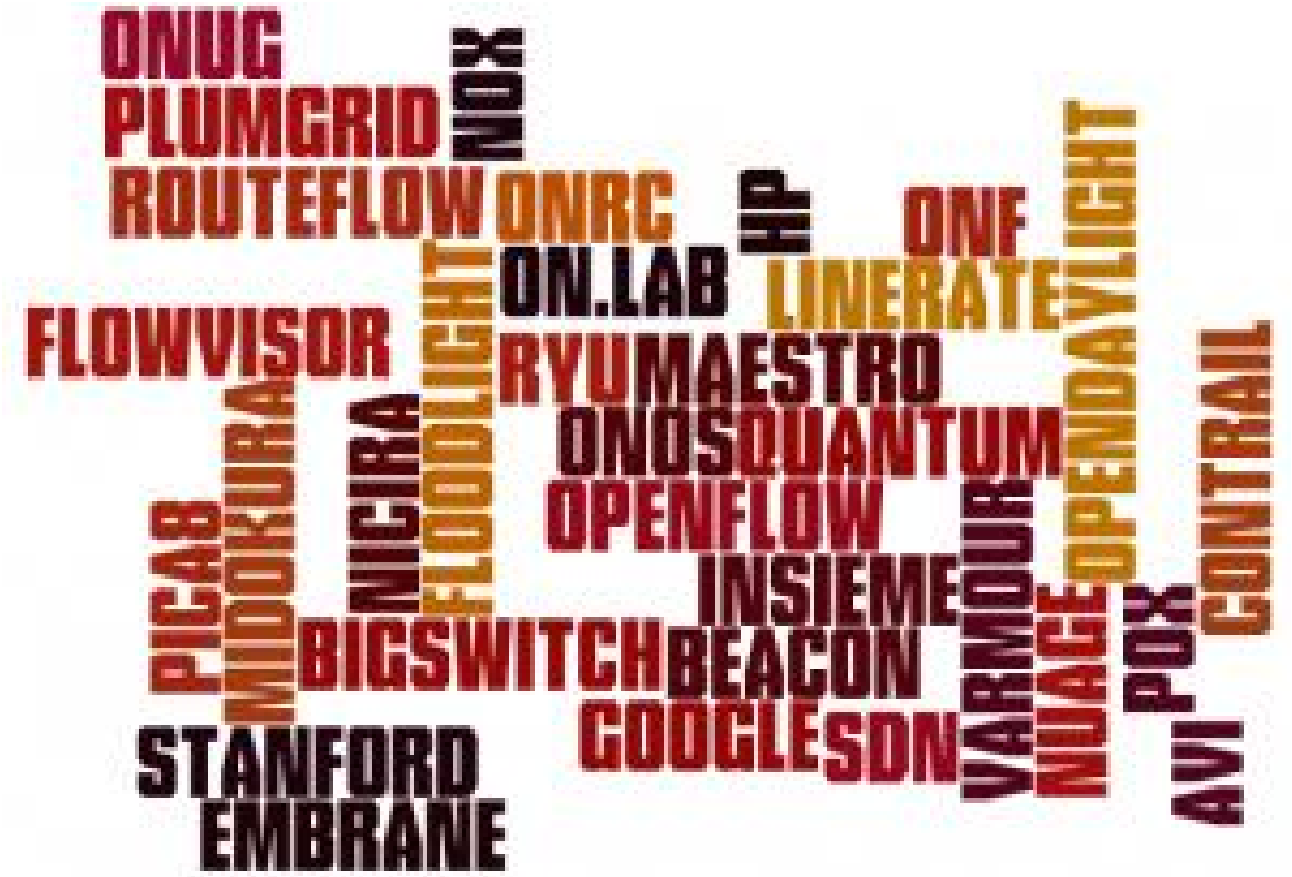
Towards open software ecosystems

Current trends: Open Source /*Everything*

- http://www.opendaylight.org
- http://www.openstack.org
- http://opencompute.org

The future is all about Software Ecosystems

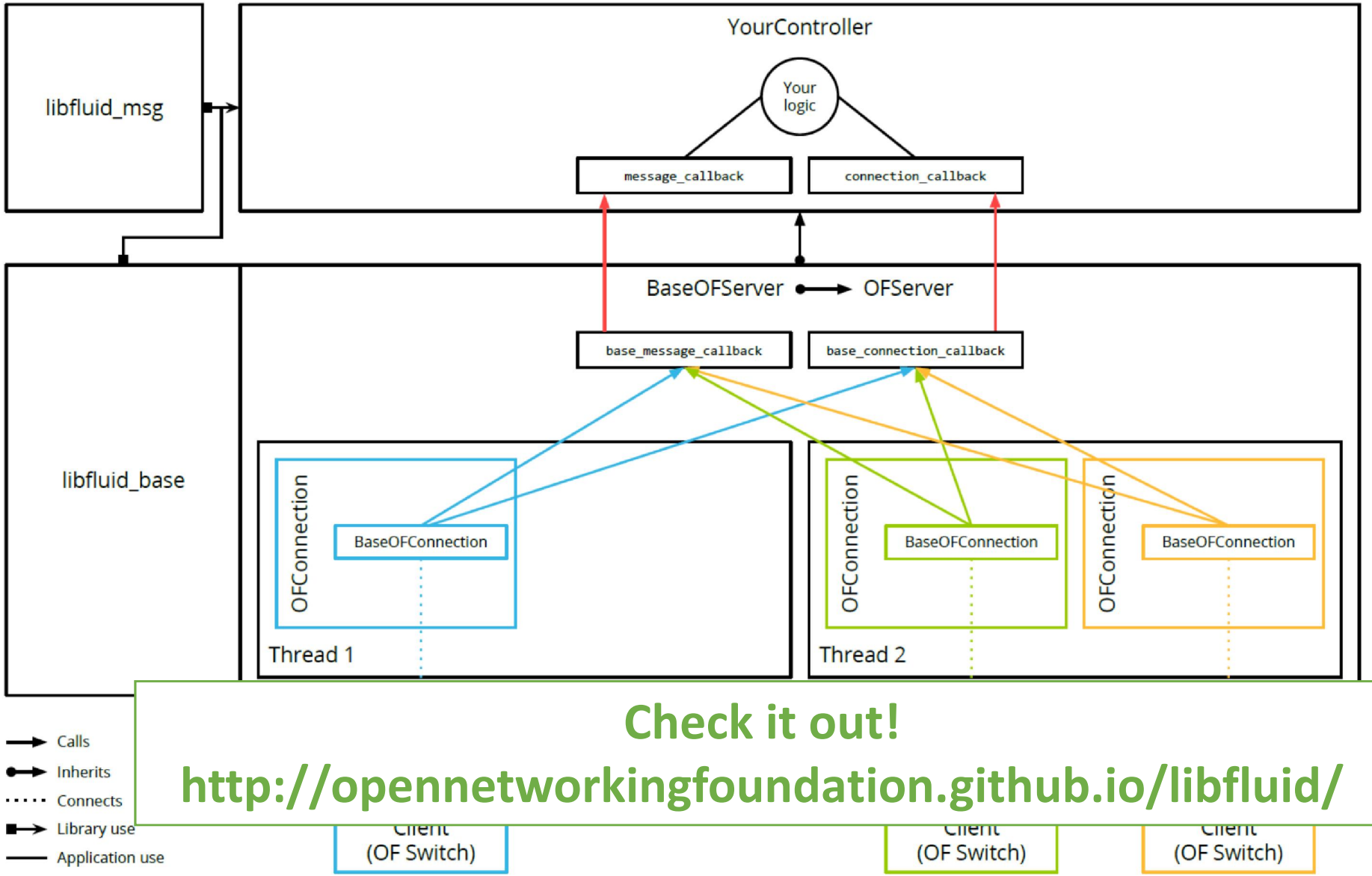
- Open Interfaces: Protocols, APIs, Code, Tool Chains
- Open Control Platforms at every level (layer?)
- “Best of Breed” markets



sdn central

SDN Hub

libfluid ONF Driver Implementation



Check it out!

http://opennetworkingfoundation.github.io/libfluid/

*Joint work with E. Fernandes (CPqD), A. Vidal (CPqD), M. Salvador (Lenovo), F. Verdi (Ufscar)

All of This Leads Us To …

Software Defined Networking

DevOps

DevOps Defined Networking

Slide courtesy: Kyle Mestery

People Like SDN Because

SDN promises to make networks:

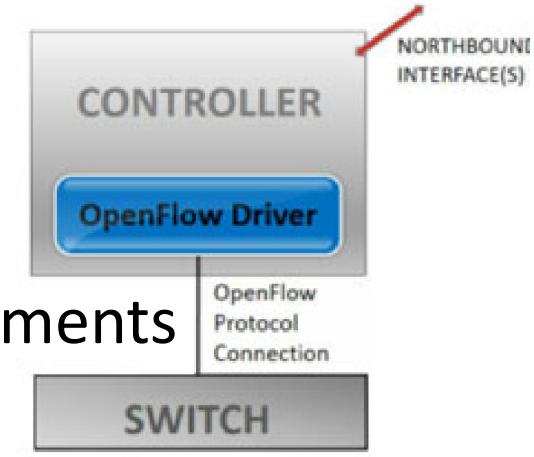
1. Dynamically provisioned
2. Provide scalable capacity
3. Provide abstracted HW complexity
4. Bandwidth on demand

Open Source Software promises to foster SDN stacks

From “Rough Consensus and Running Code” to real “Code before Standardize” dynamics

Recent Events involving Open Source

- The Battle for the Hypervisor Switch[1]
  - Open vSwitch vs Nexus1000v vs Hyper-V Virtual Switch
- The Battle for the Cloud: Stack Wars
  - Stack wars: OpenStack v. CloudStack v. Eucalyptus
- The Battle for the SDN control platform
  - OpenDaylight vs Floodlight vs etc. etc.
- The Battle for the SDN Northbound APIs
  - ONF vs OpenDaylight vs OpenStack vs etc.
- ONF OpenFlow Driver Competition
  - An open source driver to accelerate developments



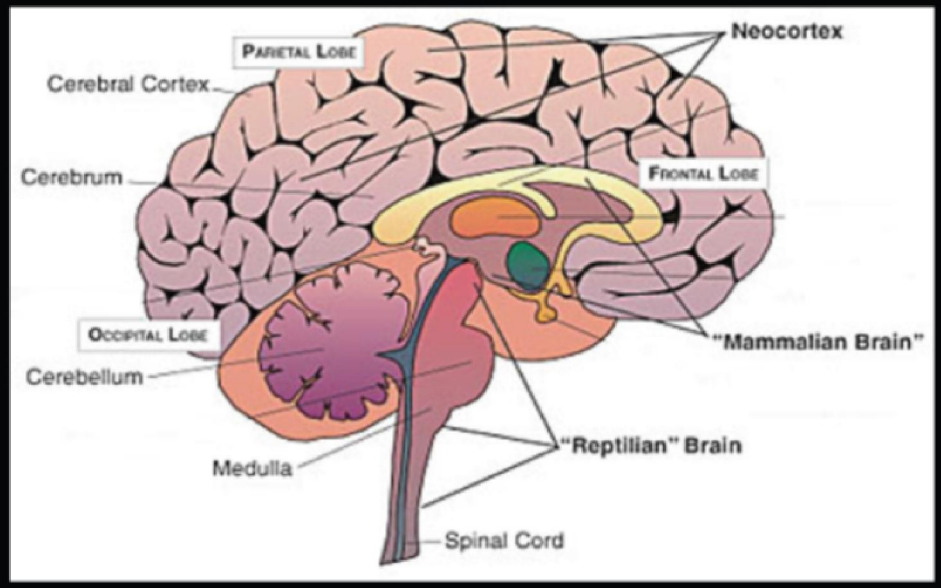
[1]http://www.networkworld.com/community/blog/battle-hypervisor-switch-and-future-networking

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

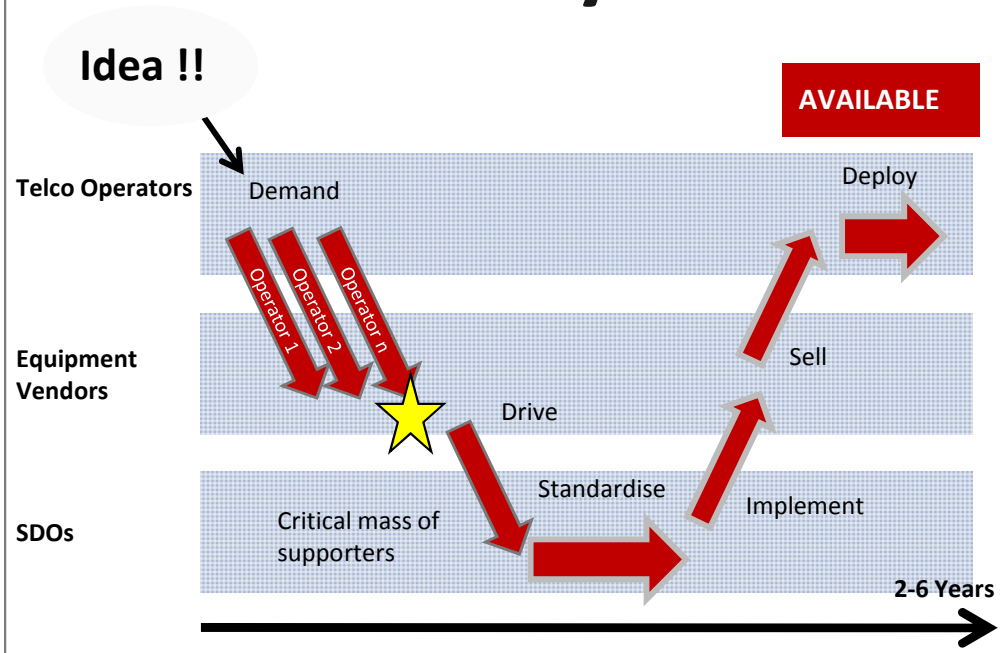
Once you have HW
its all about code[1]…

Slide courtesy: D. Meyer

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

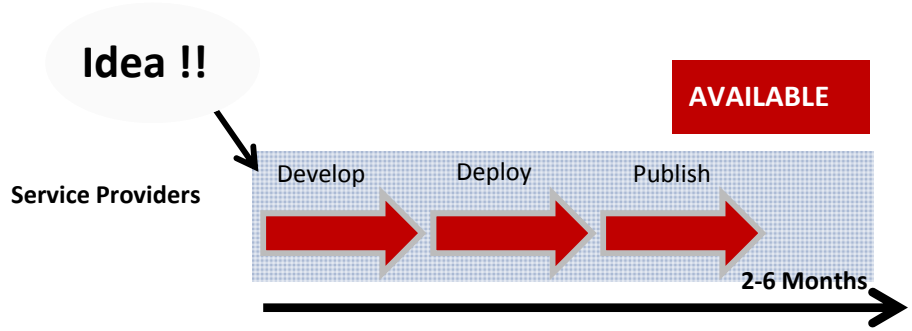
Sisyphus on Different Hills

Telco Cycle



2-6 years

Service Providers Cycle



2-6 months

Source: Telefonica I+D / ETSI NFV

NFV – Network Functions Virtualisation

Traditional Network model

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

Virtualised Network Model

- Net functionalities are **SW-based over well-known HW**
- **Multiple roles** over **same HW**

Telco industry is optimised to work with the Telco cycle, and this cannot be changed overnight

EVOLUTIONARY APPROACH

- New specific needs for different nodes
- Reuse of equipment still in amortization
- Leverage on new planned elements in architecture

BY RELEVANT USE CASES

- Virtualised PoP
- Traffic analysis
- SW-defined datacenter
- Pervasive security

Effect of **scale** (better statistical multiplexing)

Reduction of geographical **dispersion** of HW

CAPEX

Reuse same HW for different role when no longer applicable

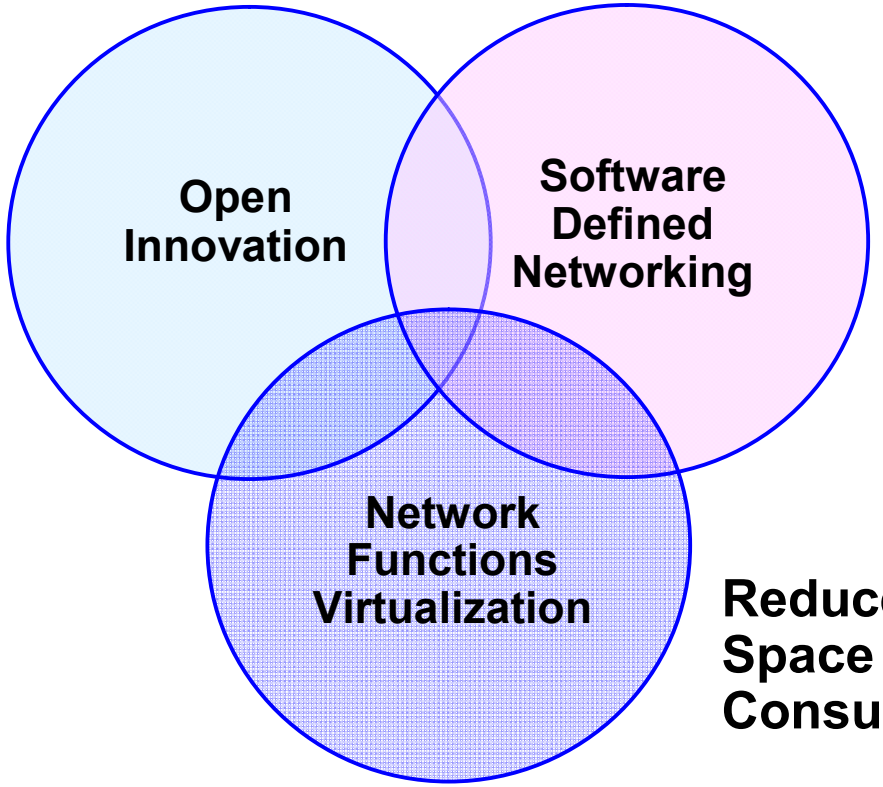
OPEX

Source: Telefonica I+D / ETSI 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

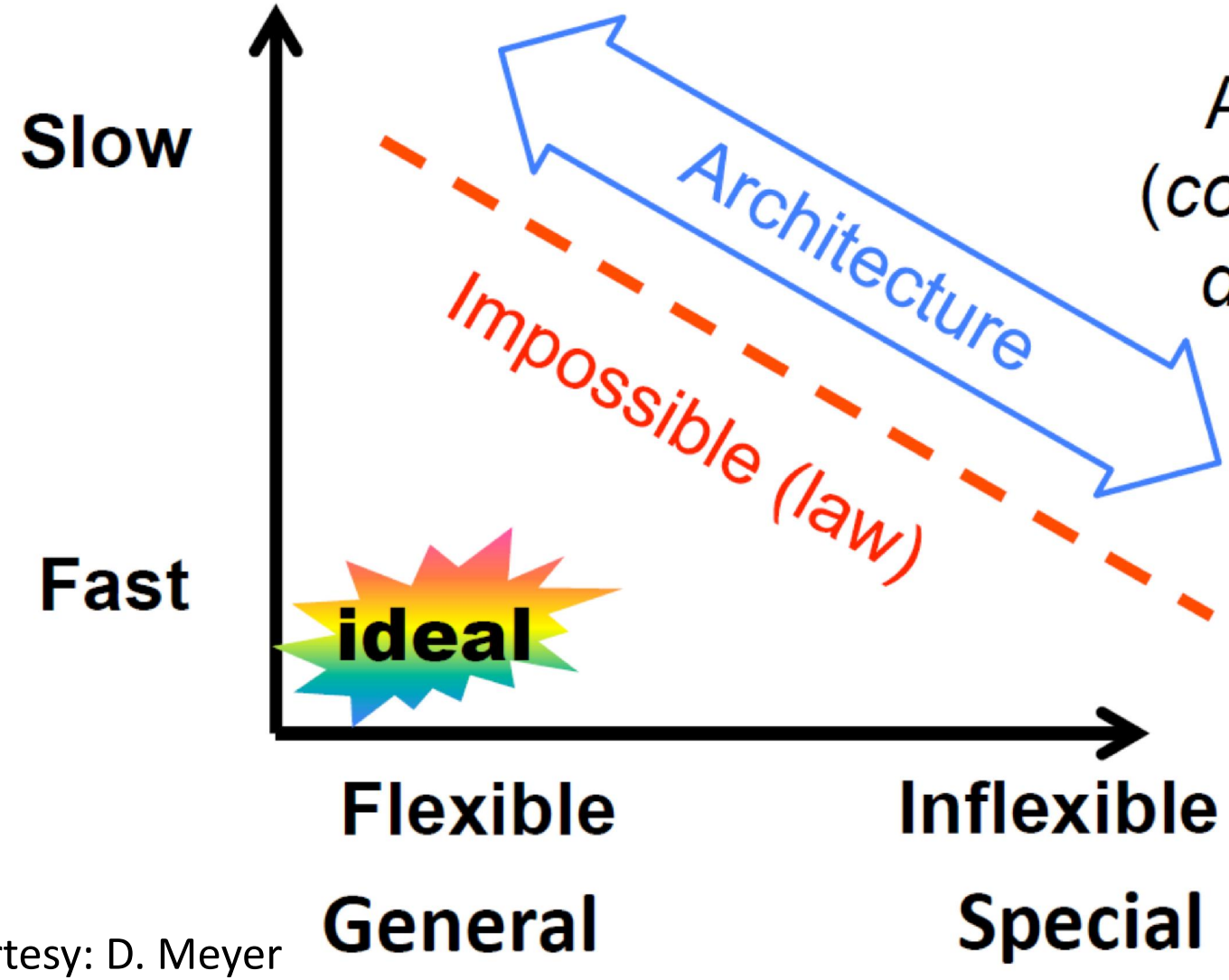
Source: ETSI NFV

Detour on Layered Architectures and Complexity

And what does this have to do with SDN and Open Source

Universal Laws and Architectures (Turing)

Layering, Formal Systems, Hard Tradeoffs

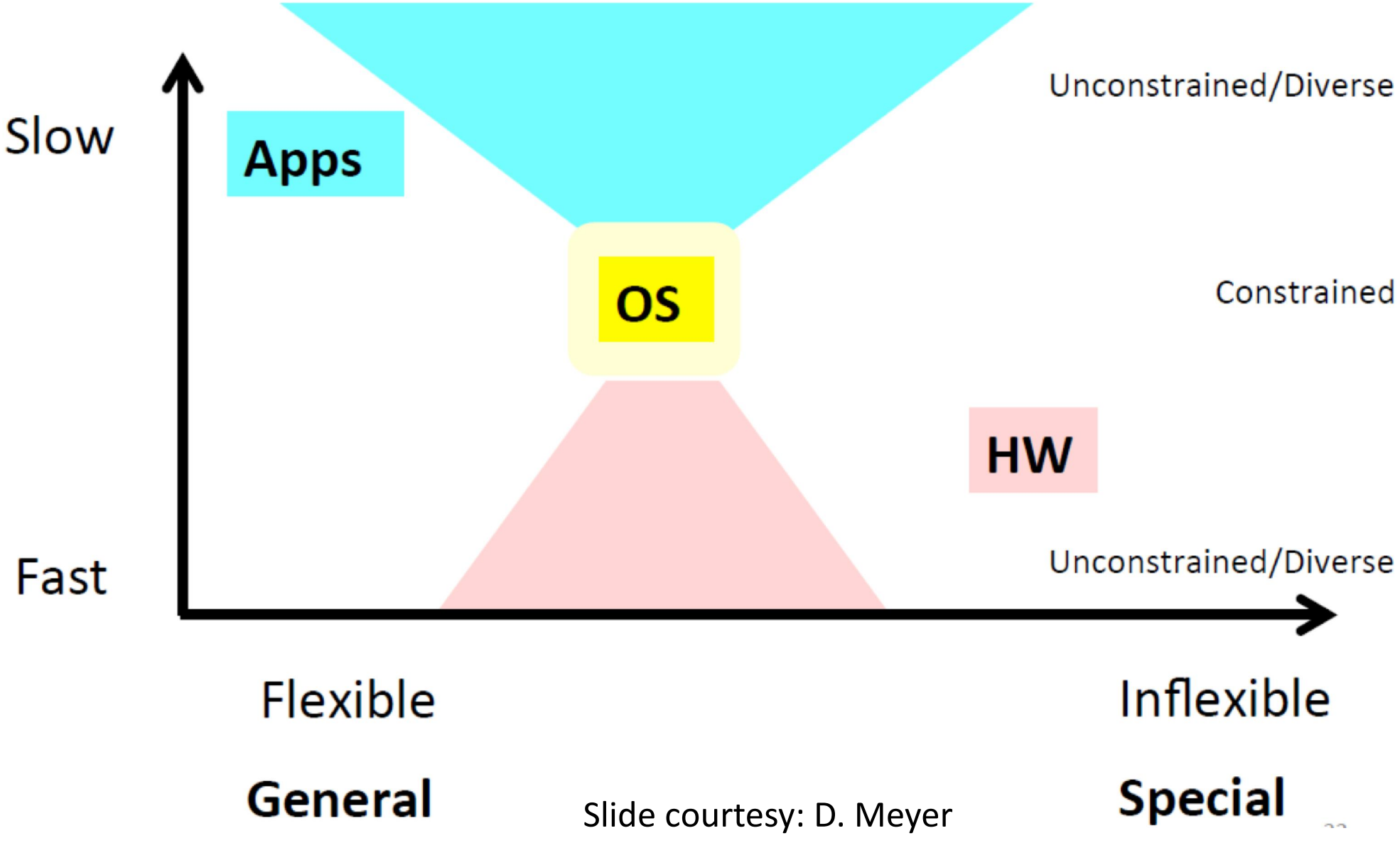


Architecture
(*constraints that deconstrain*)

Slide courtesy: D. Meyer

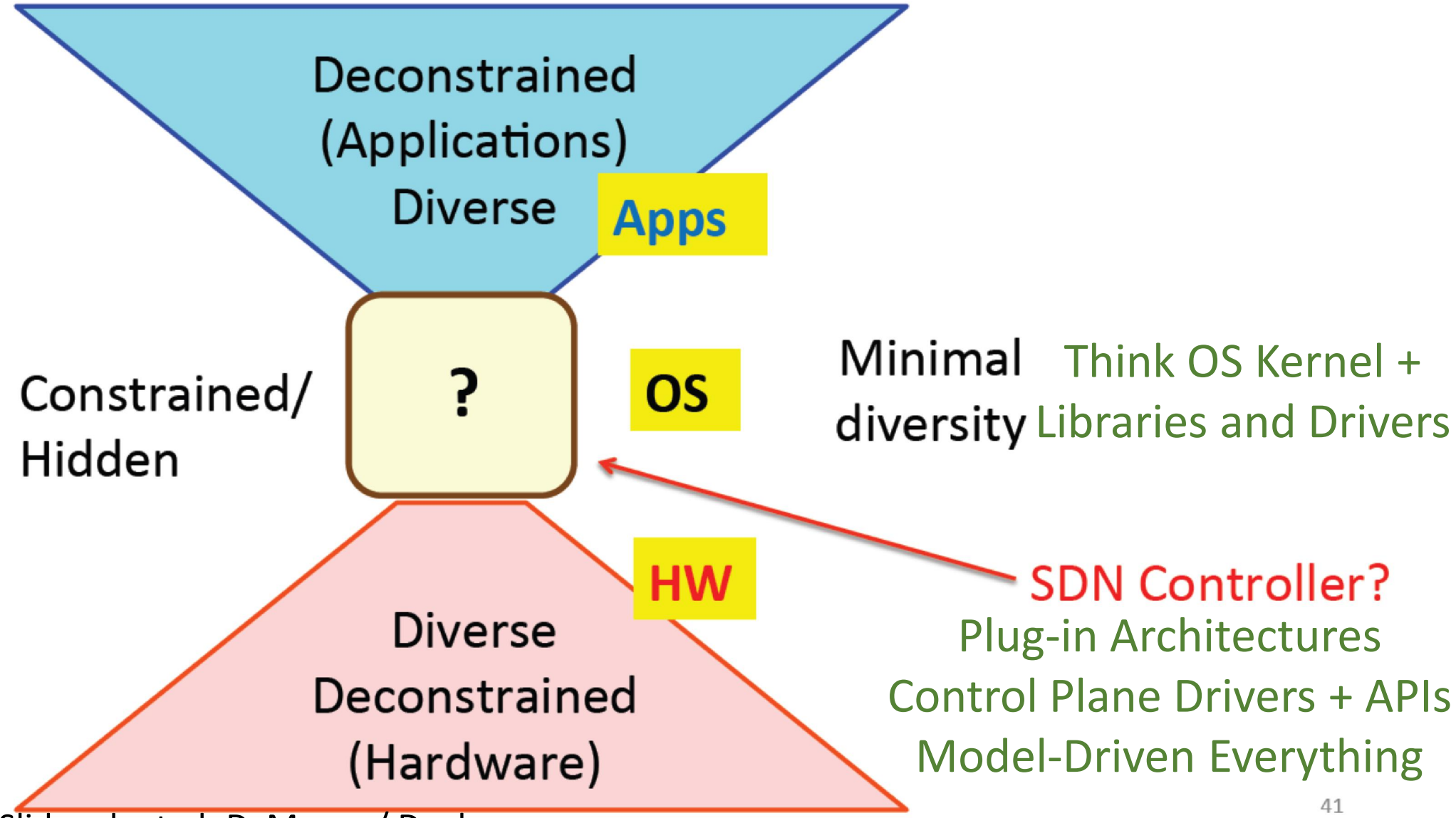
Original slide courtesy John Doyle

Overlaying Tradeoffs



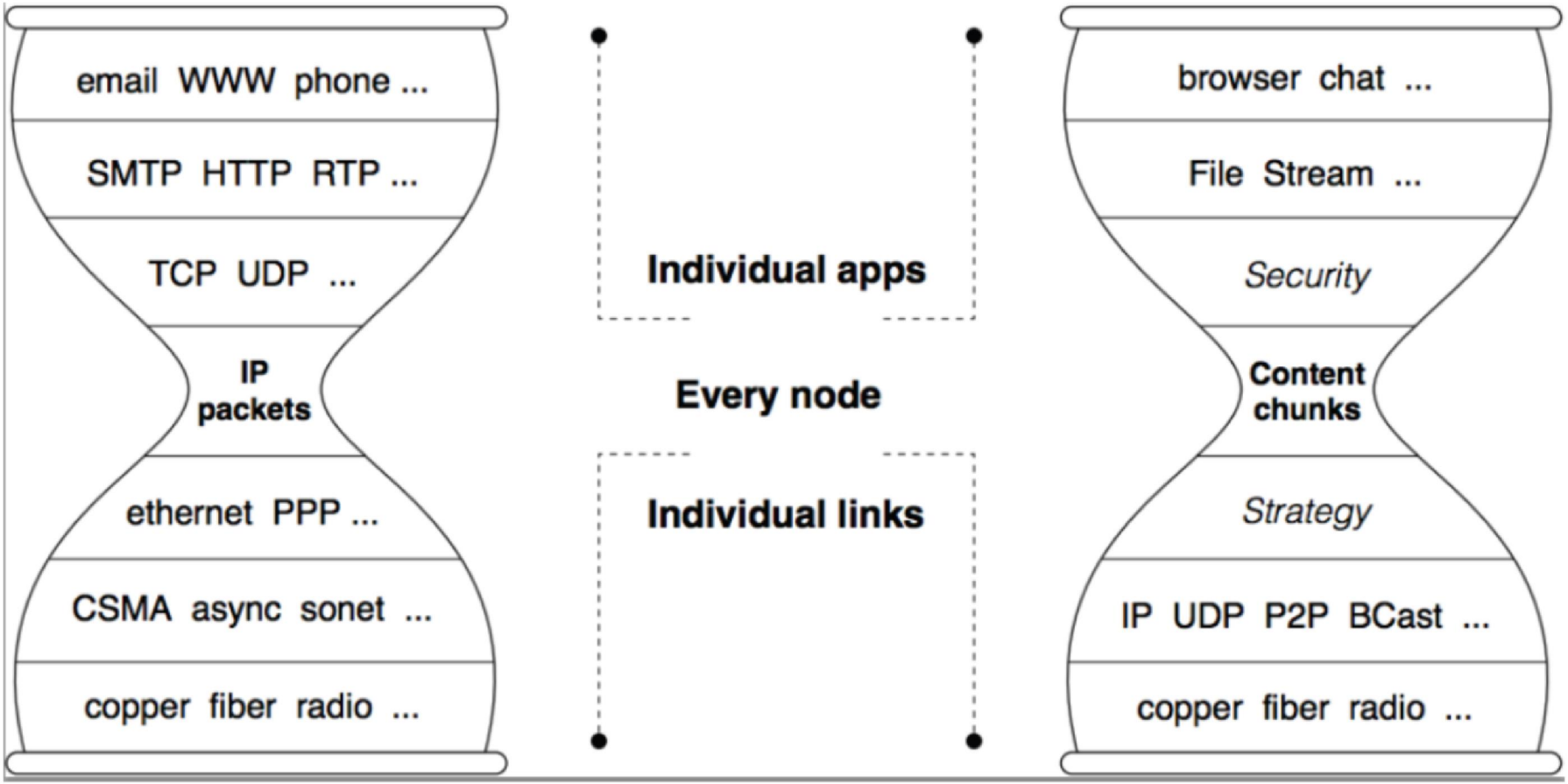
Slide courtesy: D. Meyer

Layered architectures make Robustness and Evolvability *compatible*



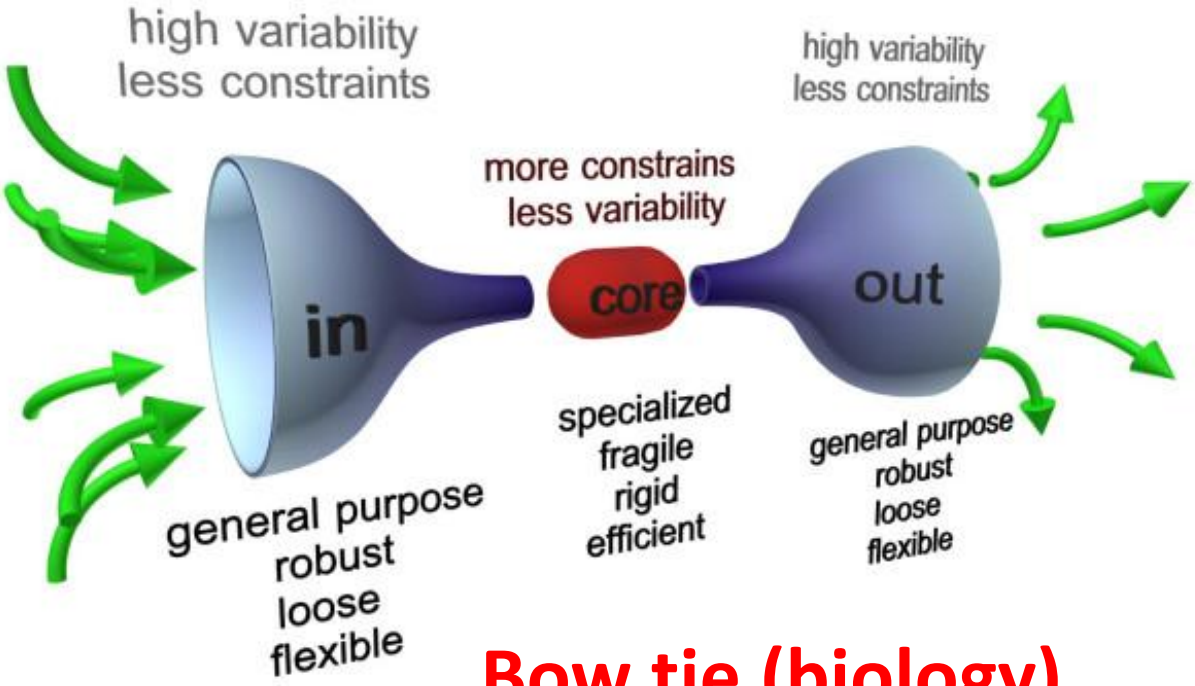
Slide adapted: D. Meyer / Doyle

Information-Centric Networking
NDN Hourglass

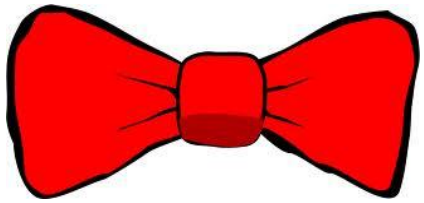


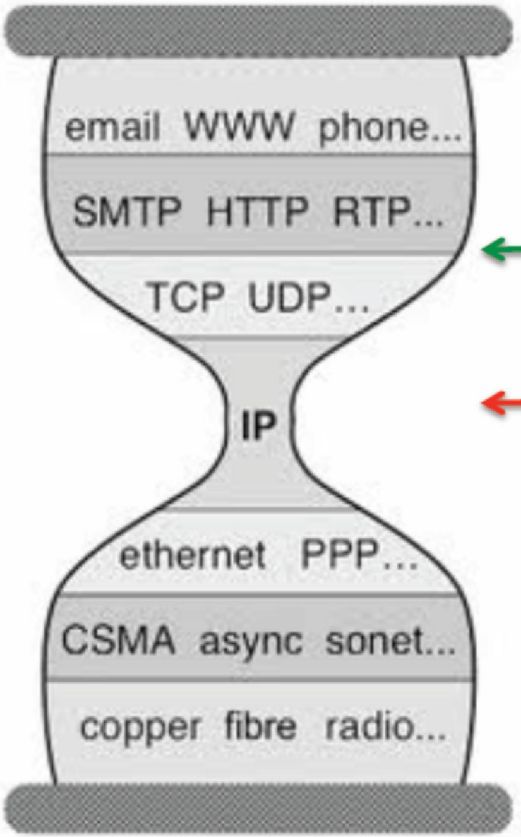
The NSF-funded Named Data Networking project has a strong open source ethos
Read more: http://named-data.net/project/ndn-copyright-and-patents

Bowties /Hourglasses?



Bow tie (biology)





OL/SDN

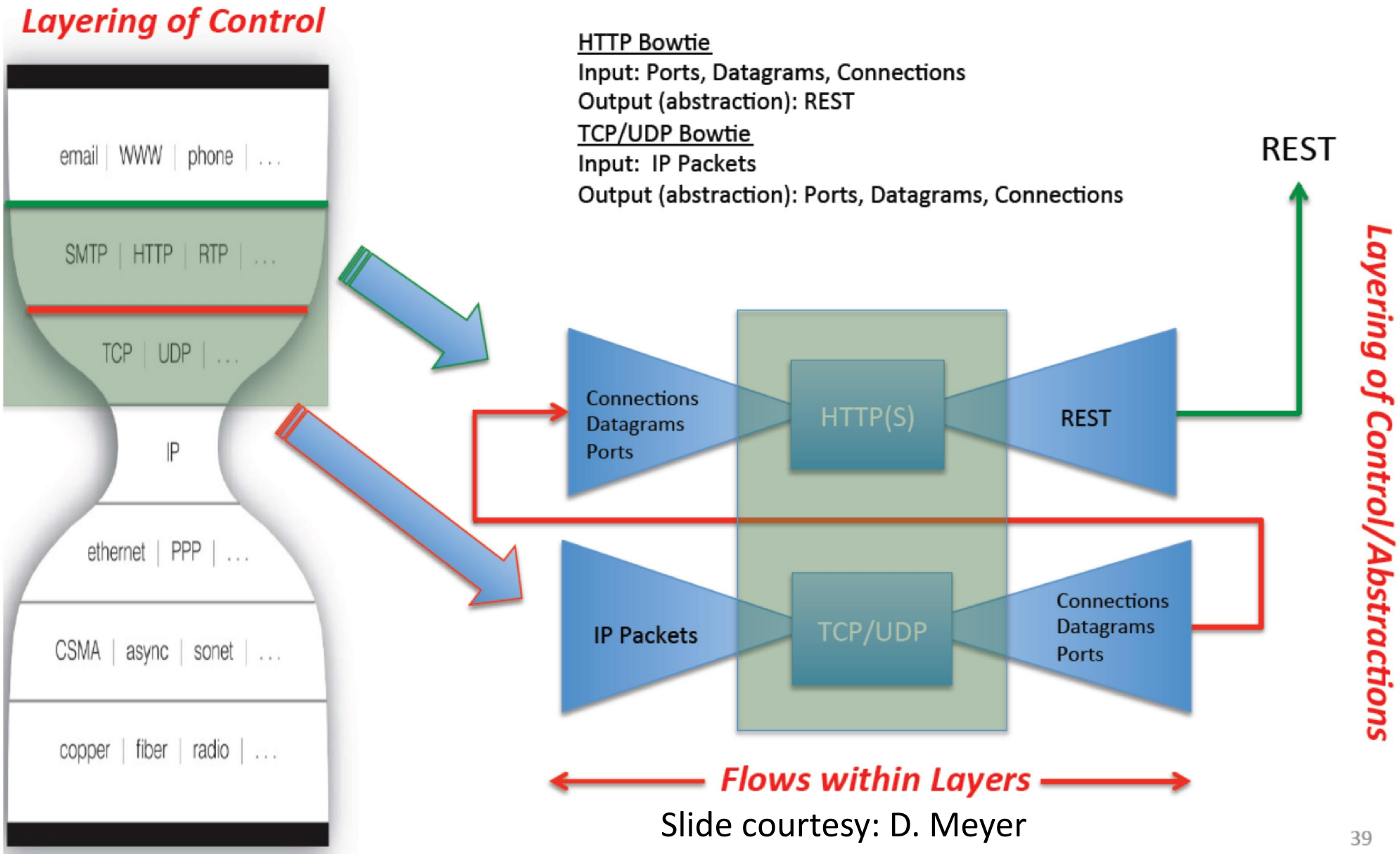
CP/SDN

OF/SDN?

Open Source is a wildcard

Adapted Slide courtesy: D. Meyer

The Nested Bowtie/Hourglass Architecture of the Internet



Slide courtesy: D. Meyer

Your network´s control plane is 100% proprietary? Hey, this is 2014 not 1994

The RouteFlow Project

RouteFlow: Introduction

RouteFlow

FUNTTEL

Ministério das Comunicações

GOVERNO FEDERAL BRASIL PAÍS RICO É PAÍS SEM POBREZA

FINEP AGÊNCIA BRASILEIRA DA INOVAÇÃO

Background

Glue of IP routing stacks with OpenFlow

Controller-centric hybrid IP networks

Migration path to SDN

Architecture

Modular (3 components)

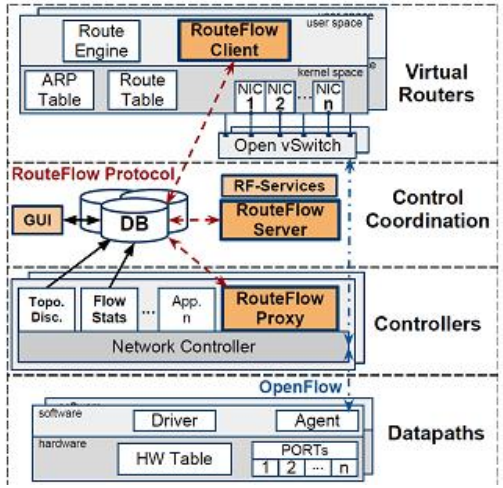
Hierarchical, distributed

Multi-controller support

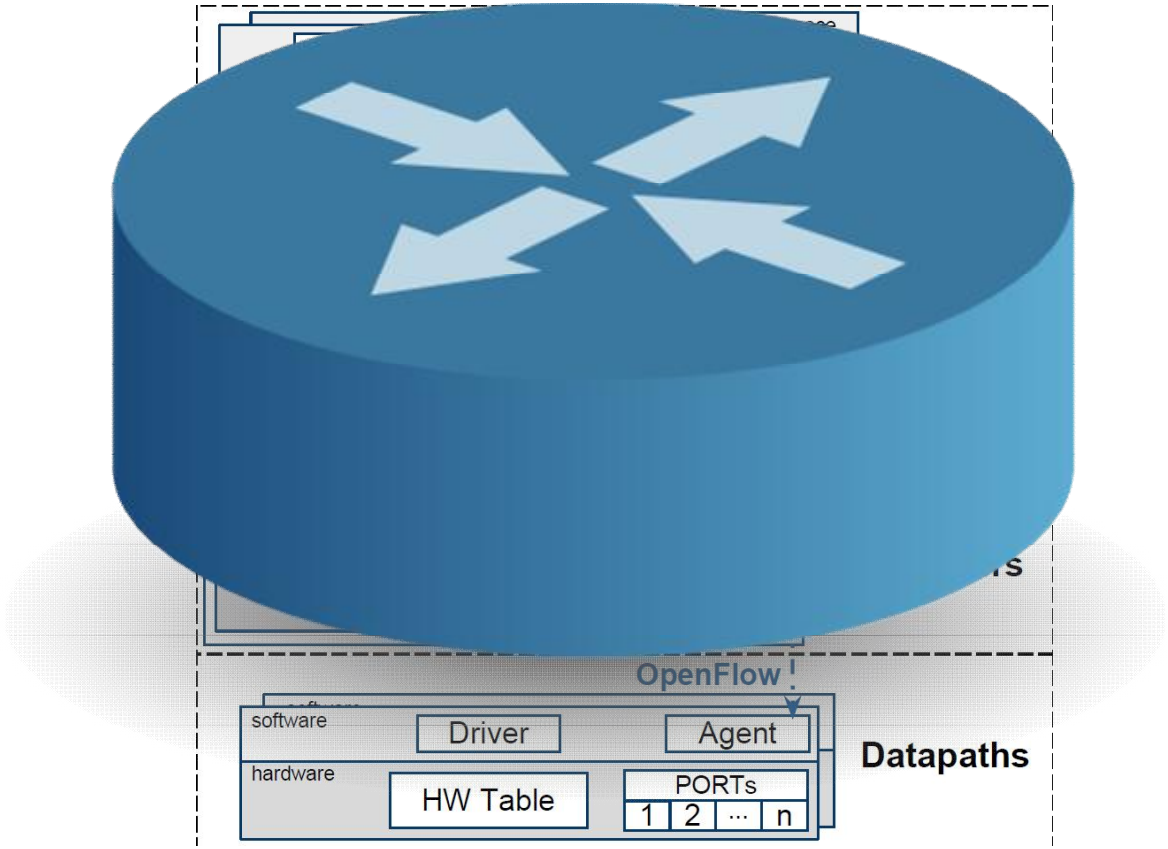
(POX, NOX, Floodlight, Ryu)

Any Linux-based routing stack

(Quagga, XORP, BIRD)



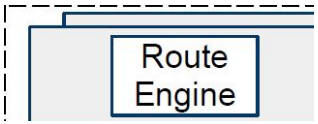
RouteFlow: Basics



Control Plane

Data Plane

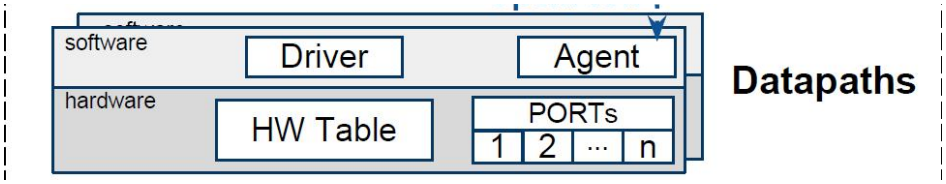
RouteFlow: High-level Architecture



Linux

Control Plane

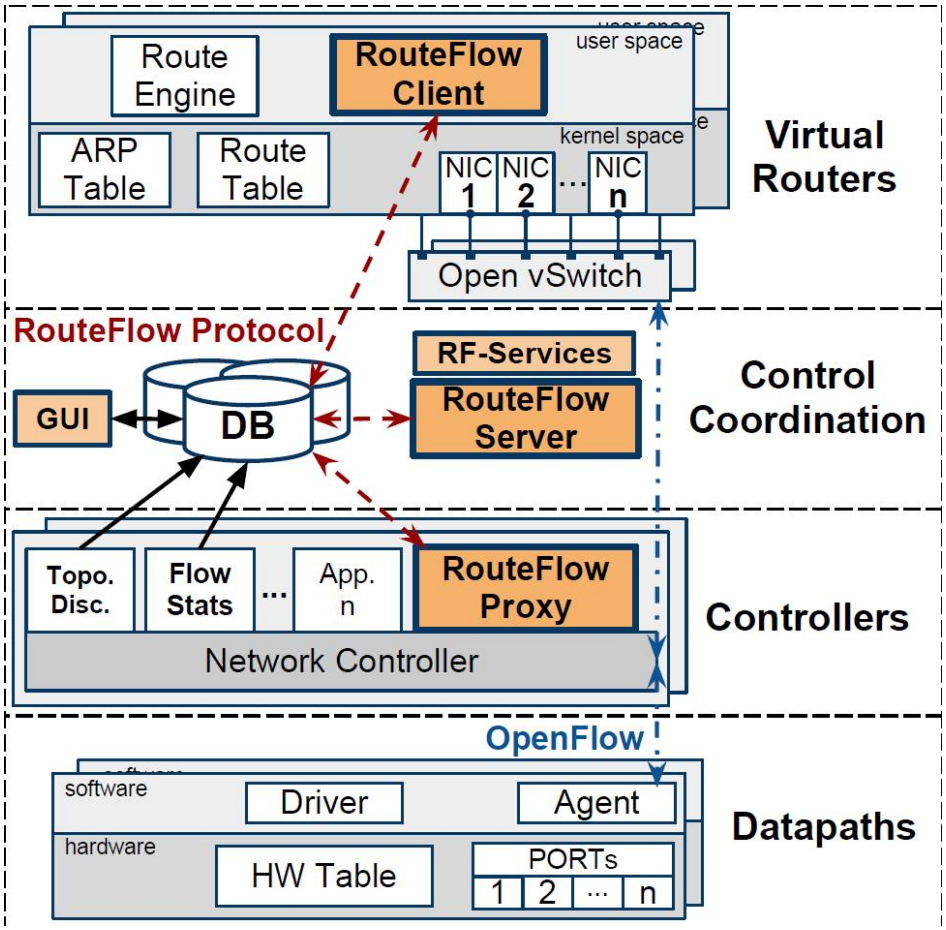
Glue

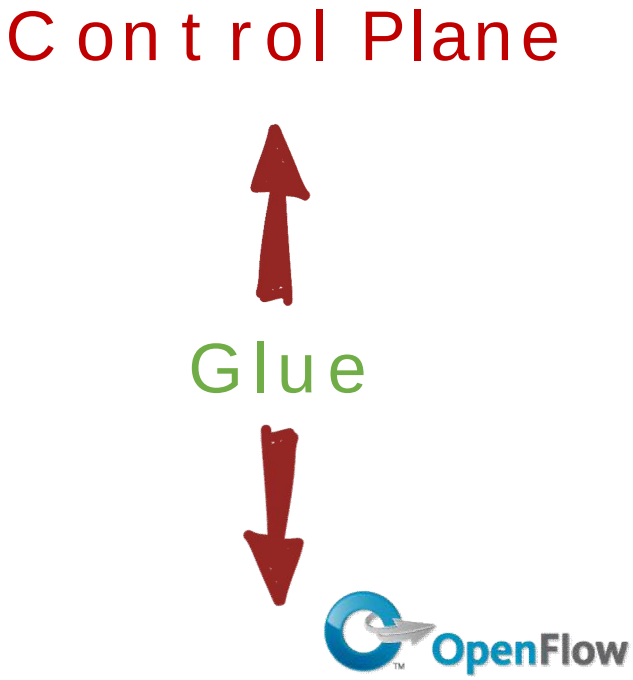


Datapaths

Data Plane

RouteFlow: High-level Architecture





Open Source SW | RouteFlow SDN/OpenFlow architecture

Quagga, XORP, BIRD

Linux

Open vSwitch, softswitch13

LXC, QEMU, XEN, libvirt

MongoDB, redis

jQuery, JIT

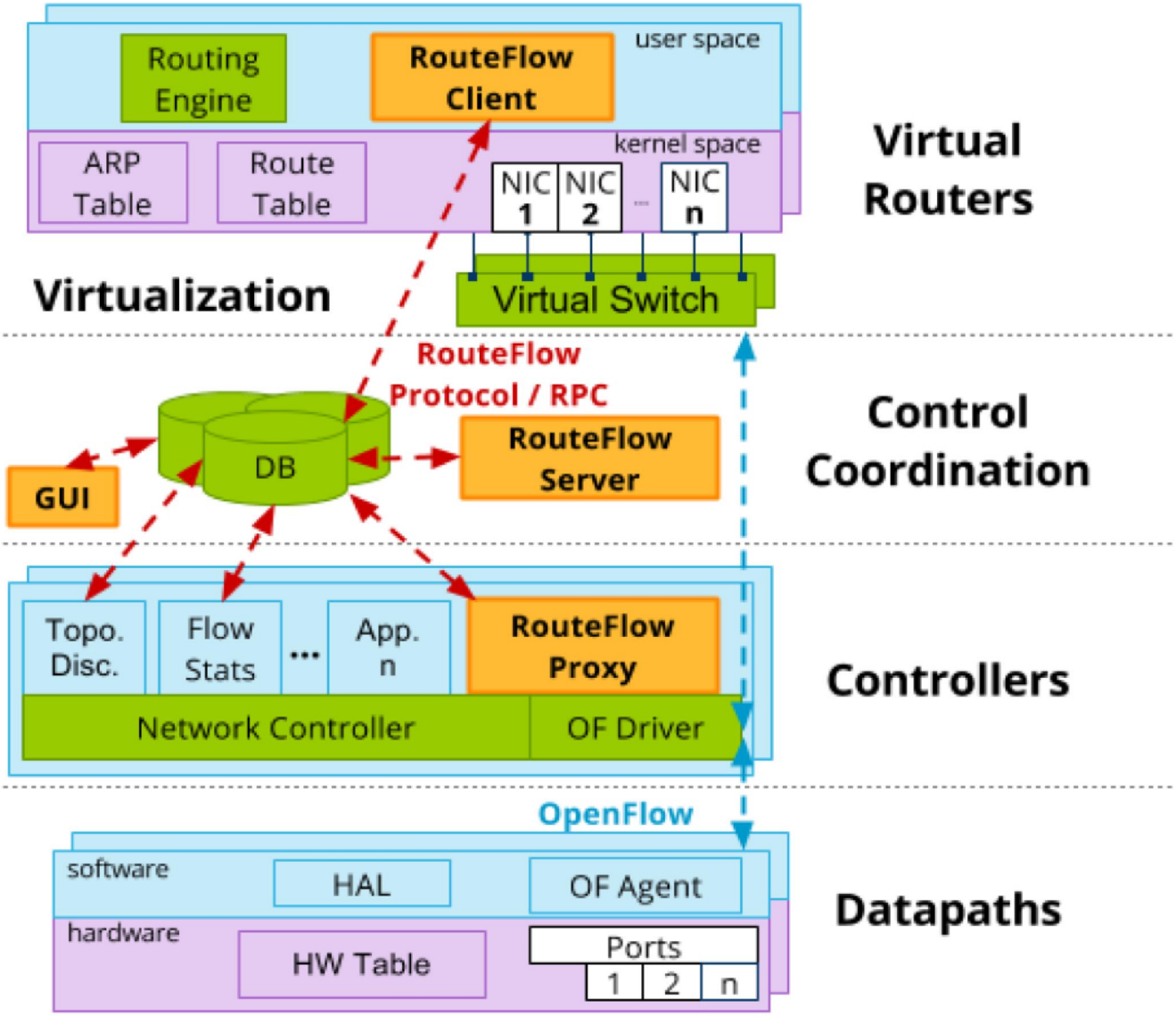
NOX, POX, Floodlight, Ryu

ONF driver

FlowVisor

Mininet, OVS,

softswitch13, Pica8 OS



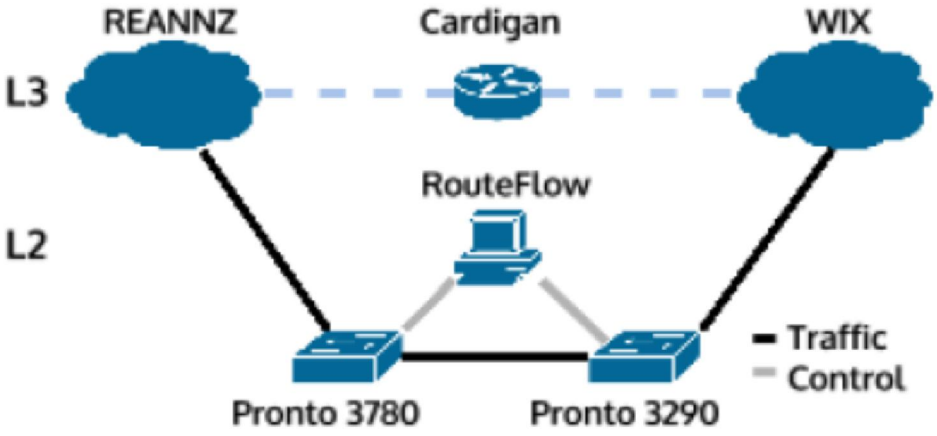
RouteFlow Project History

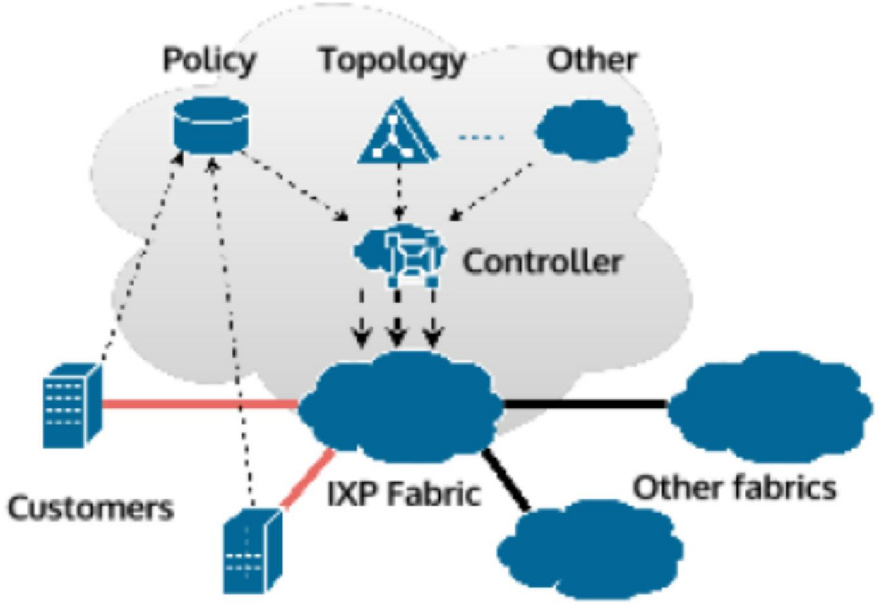
Jan / 2010 — **Aug / 2010** — **Dez / 2010** — **May / 2011** — **Oct / 2011** — **Nov / 2011** — **Apr / 2012** — **Aug / 2012**

- Start Msc. Thesis work by Marcelo N.
- First Prototype
- First Short-Paper @ WPEIF
- QuagFlow Poster @ SIGCOMM
- Evaluation on NetFPGA testbed
- Open-Source Release
- Demos @ ONS11
- Tutorial & Demo @ OFELIA/CHANGE SS
- Indiana University - Pronto OF switches + BGP peering with Juniper MX
- Demo @ SuperComputing 11
- Demos @ ONS12
- HotSDN Paper
- Running on FIBRE / OFELIA testbed
- Collaboraion with NTT

RouteFlow 2013

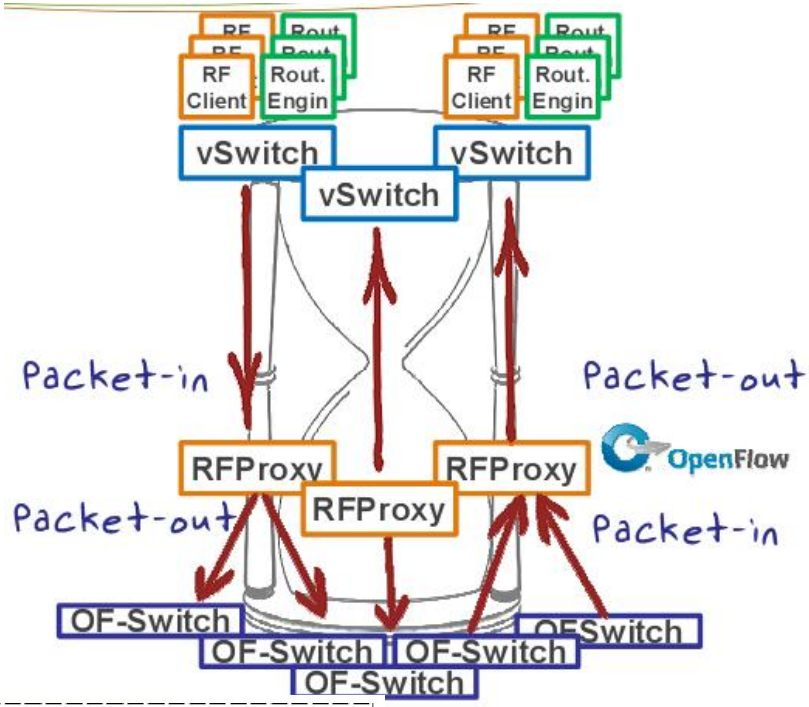
- …
- Inter-VLAN routing at UNESP, Brazil.
- Google Summer of Code (GSOC 2013)
- Significant contributions from the CARDIGAN (in live IXP for > 9 months) and VANDERWECKEN forks:



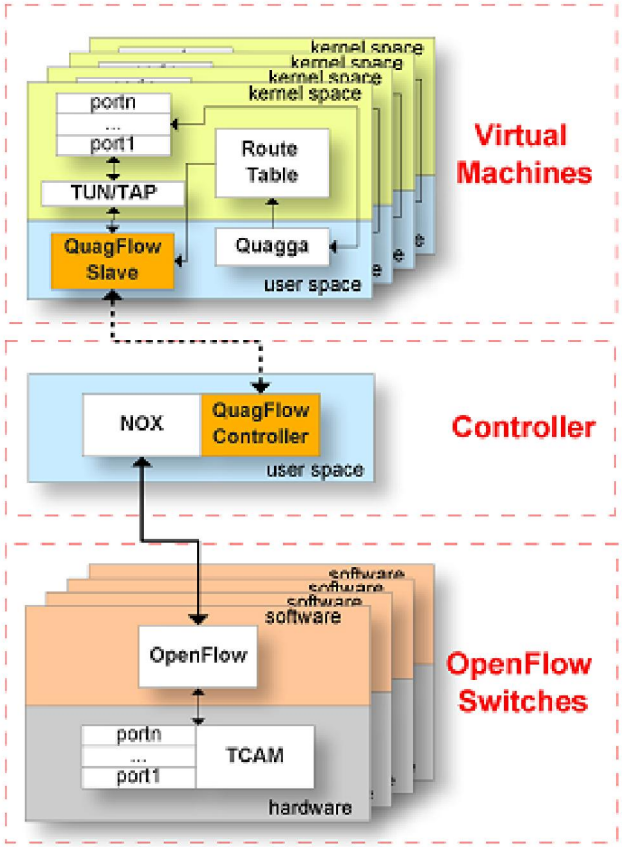


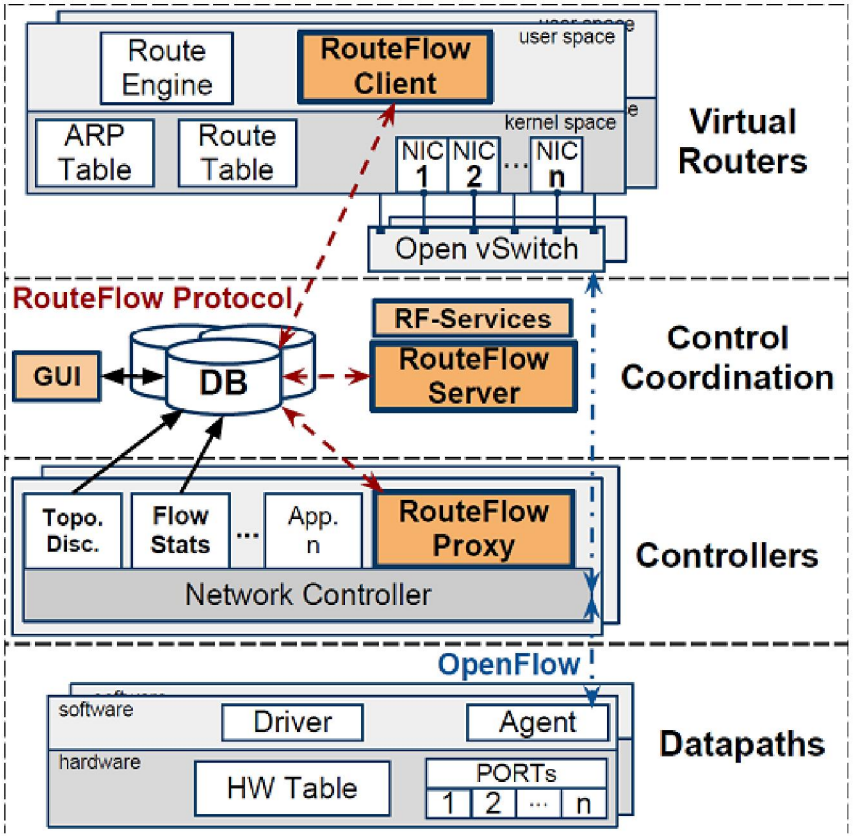
RouteFlow architecture

- Evolution since QuagFlow PoC



Dev(Ops)

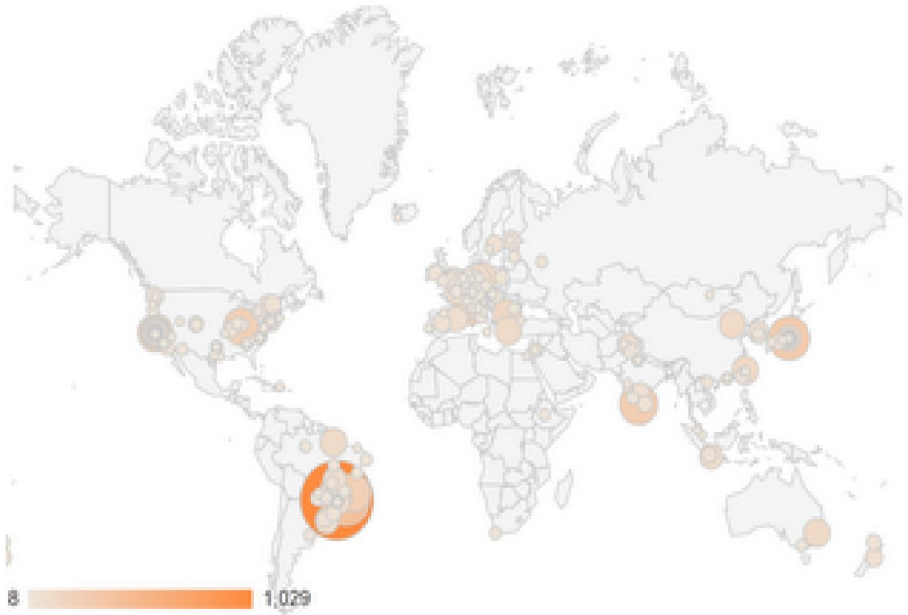


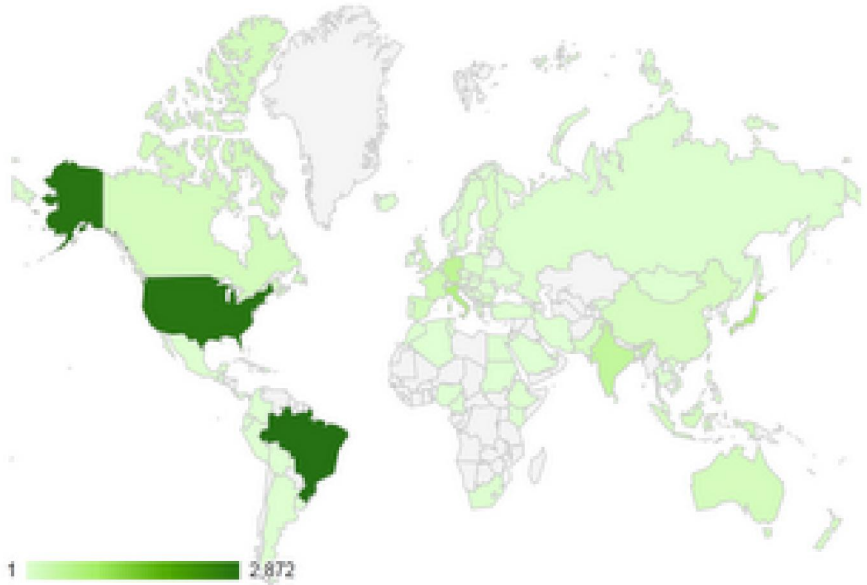


Open Innovation

Visitors: 40,000+ (20,000+ unique)

From 3,000+ cities from 130+ countries!





http://go.cpqd.com.br/routeflow/

RouteFlow

900

days since
project start

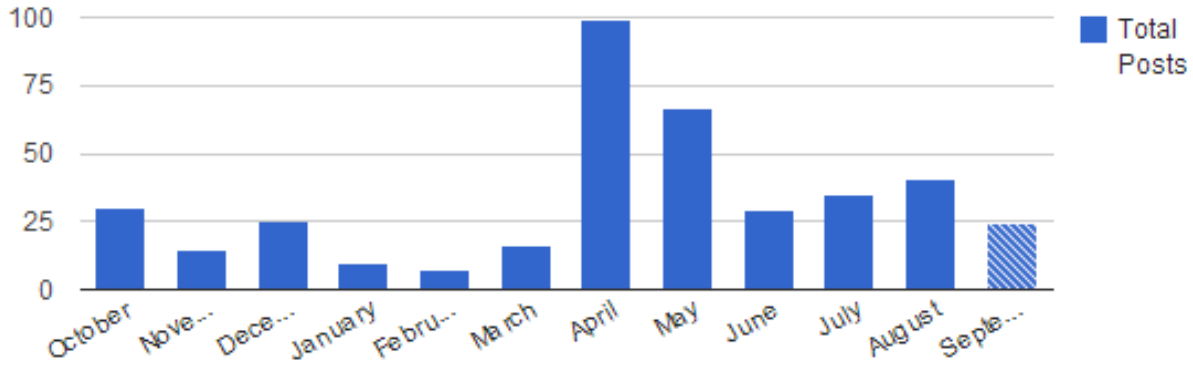
1000s downloads!

github SOCIAL CODING

Apache

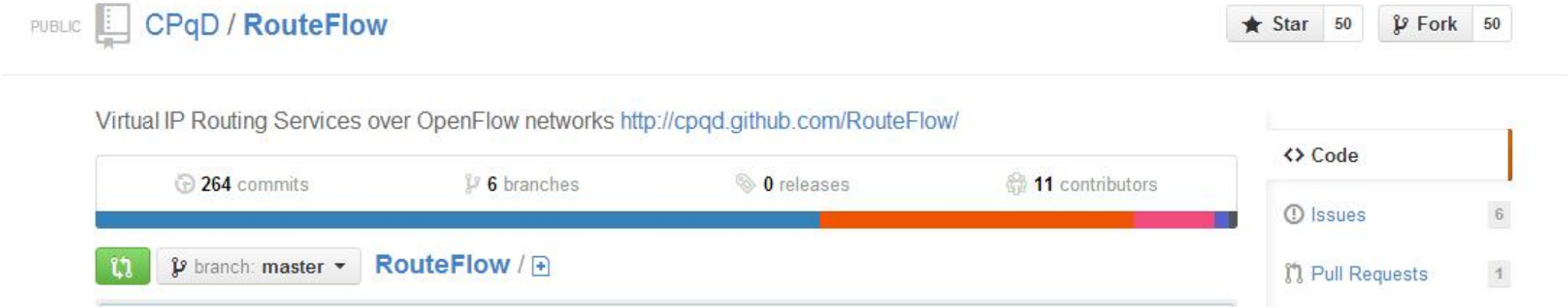
Community development

- Mailing List:
  - 200 Members
  - 160 Topics

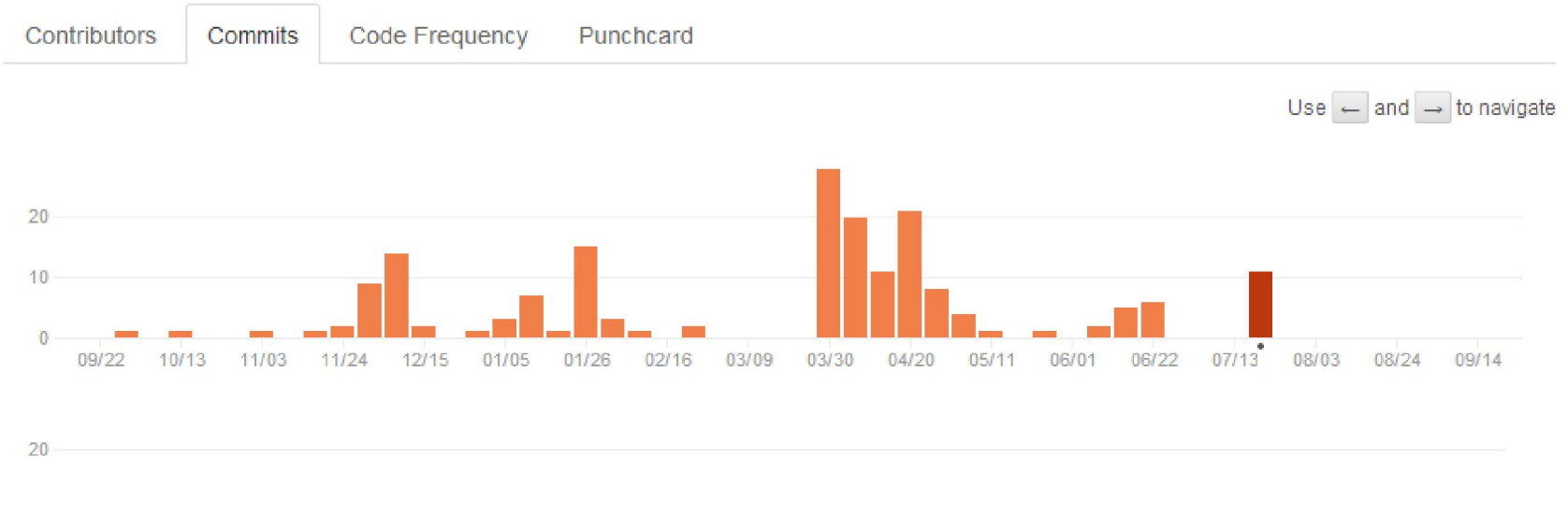


- Code contributions:

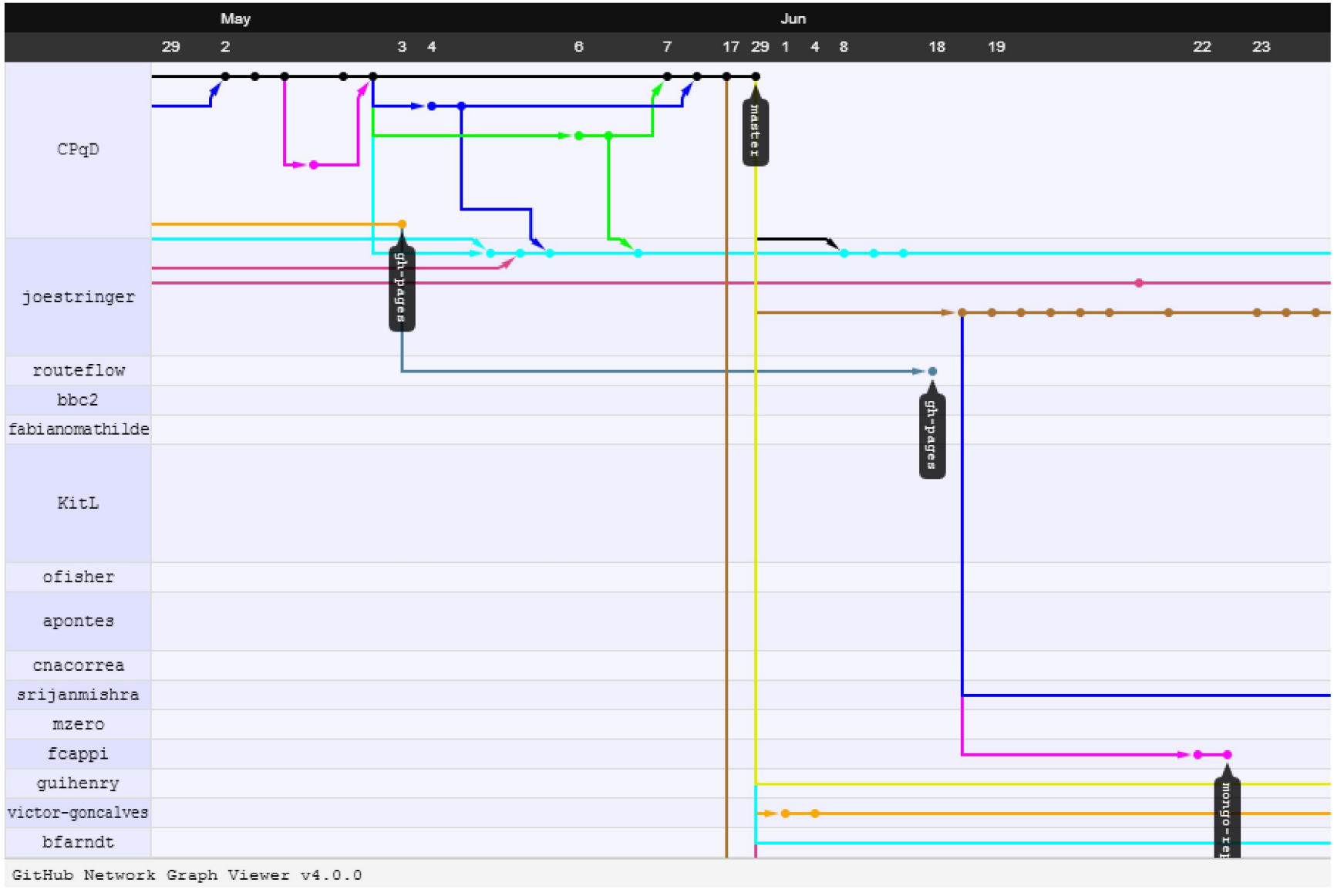
  (5K – 10K LOC, many testing hours, bug reports, etc)



Github Activity



Github Activity



Multiplying results with Open Innovation

- Web-based UI & Internet 2 HW pilot [C. Small, Indiana] ✓
- Aggregated BGP Routing Service [C. Corrêa, Unirio] ✓
- SNMP plugin [J. Stringer, Google] ✓
- Vanderwecken & Cardigan forks (IPv6, MPLS) [REANZ, Google] ✓

RouteFlow

CPqD UNICAMP UNIRIO ufscar NTT Communications InCNTRE Google Open Source Routing

Evolving the IP routing landscape with OpenFlow/SDN

Concluding Remarks

- Think “open” in your research activities
- Be “open” in your research results
  - Carry an effective open source strategy

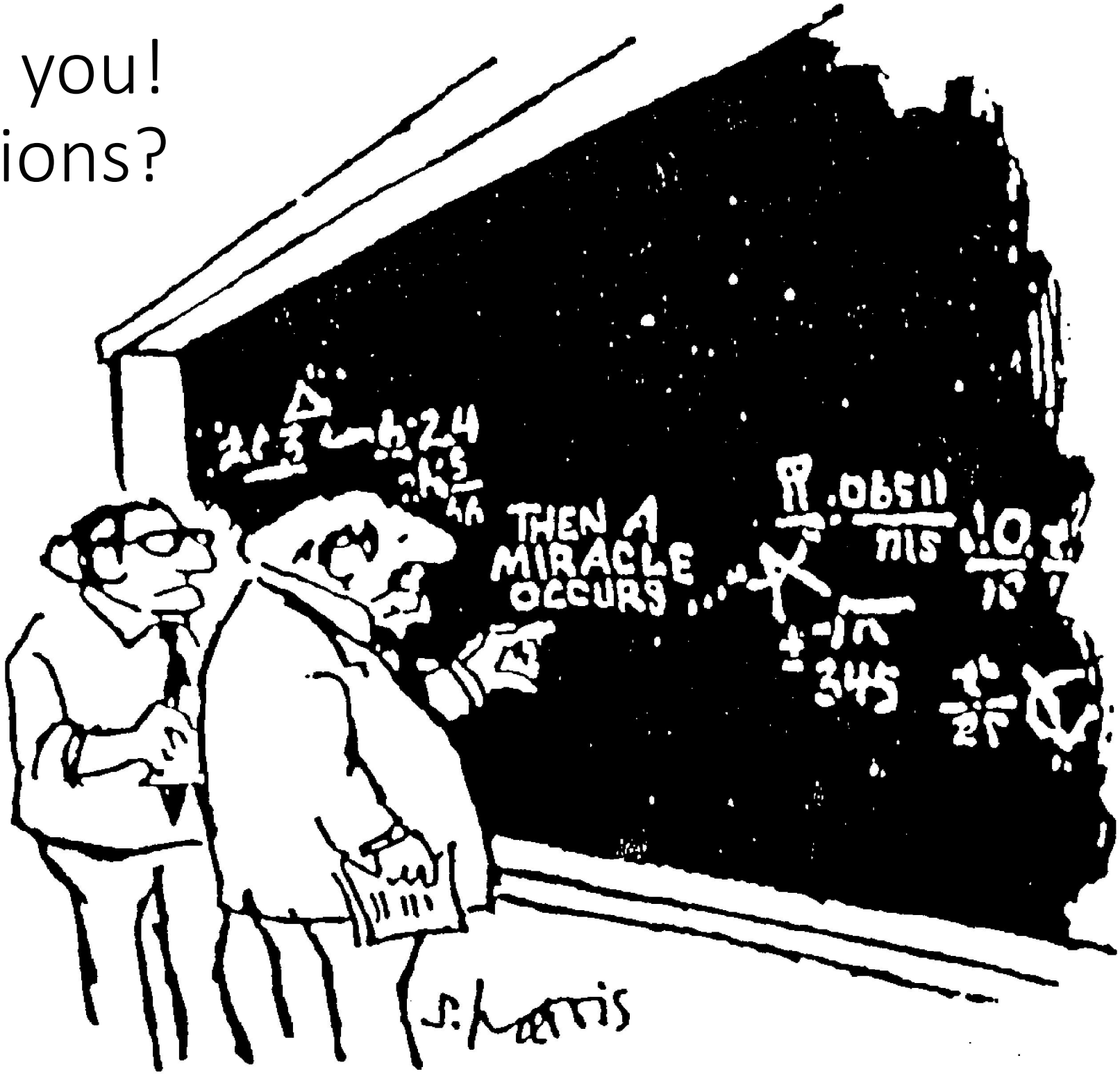
The future is all about Software Ecosystems

- Open Interfaces: Protocols, APIs, Code, Tool Chains
- Open Control Platforms at every level (layer?)
- “Best of Breed” markets

References, Further Readings & Credits

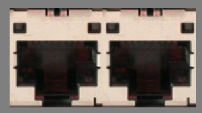
- Alderson, David L., and John C. Doyle. "Contrasting views of complexity and their implications for network-centric infrastructures." *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on* 40.4 (2010): 839-852.
- David Meyer, Macro Trends, Architecture, and the Hidden Nature 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

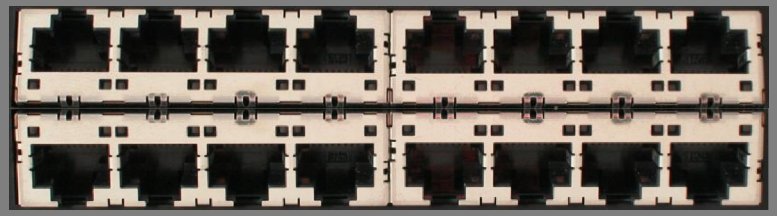
Thank you!
Questions?

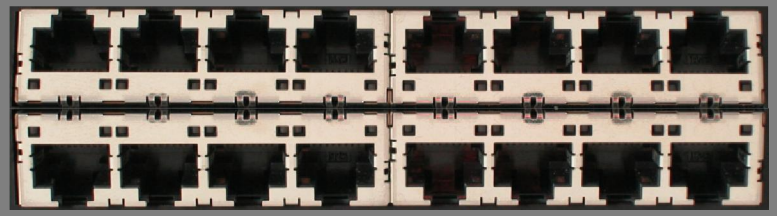


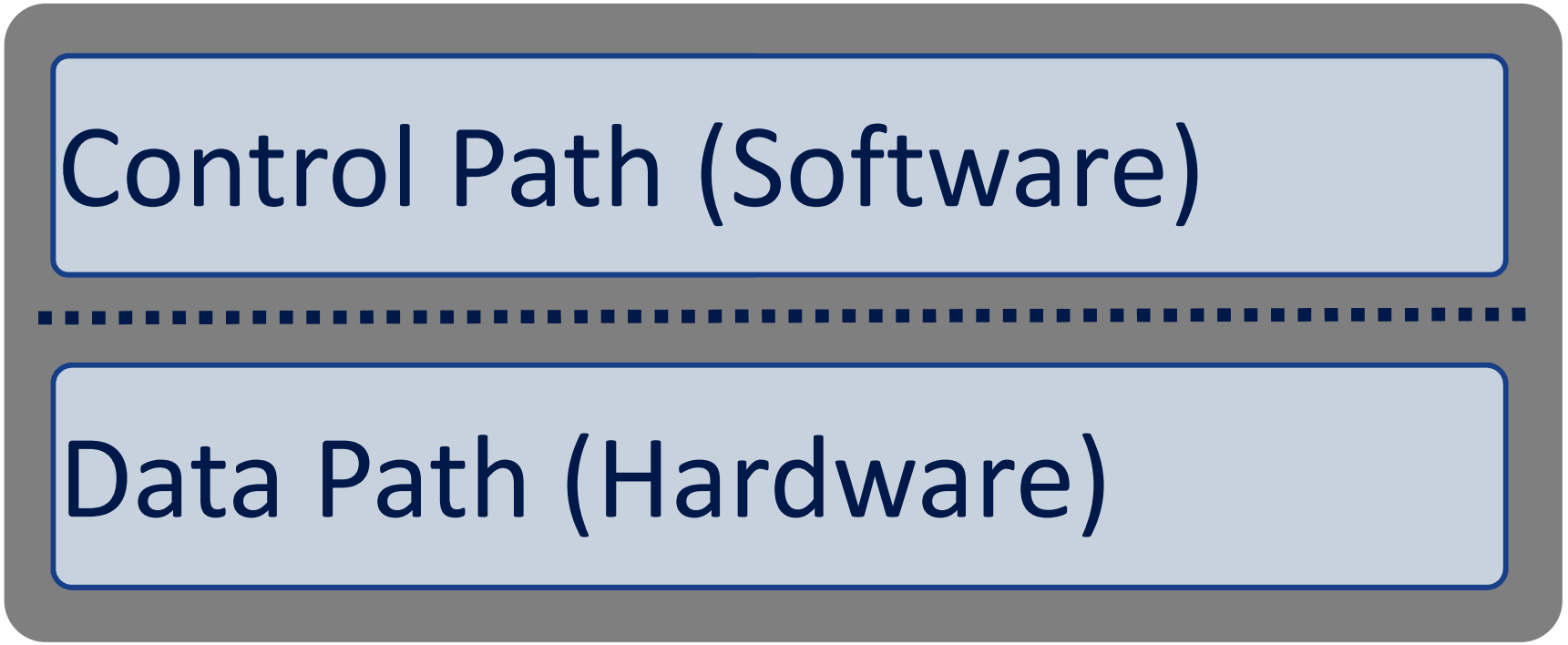
How does OpenFlow work?

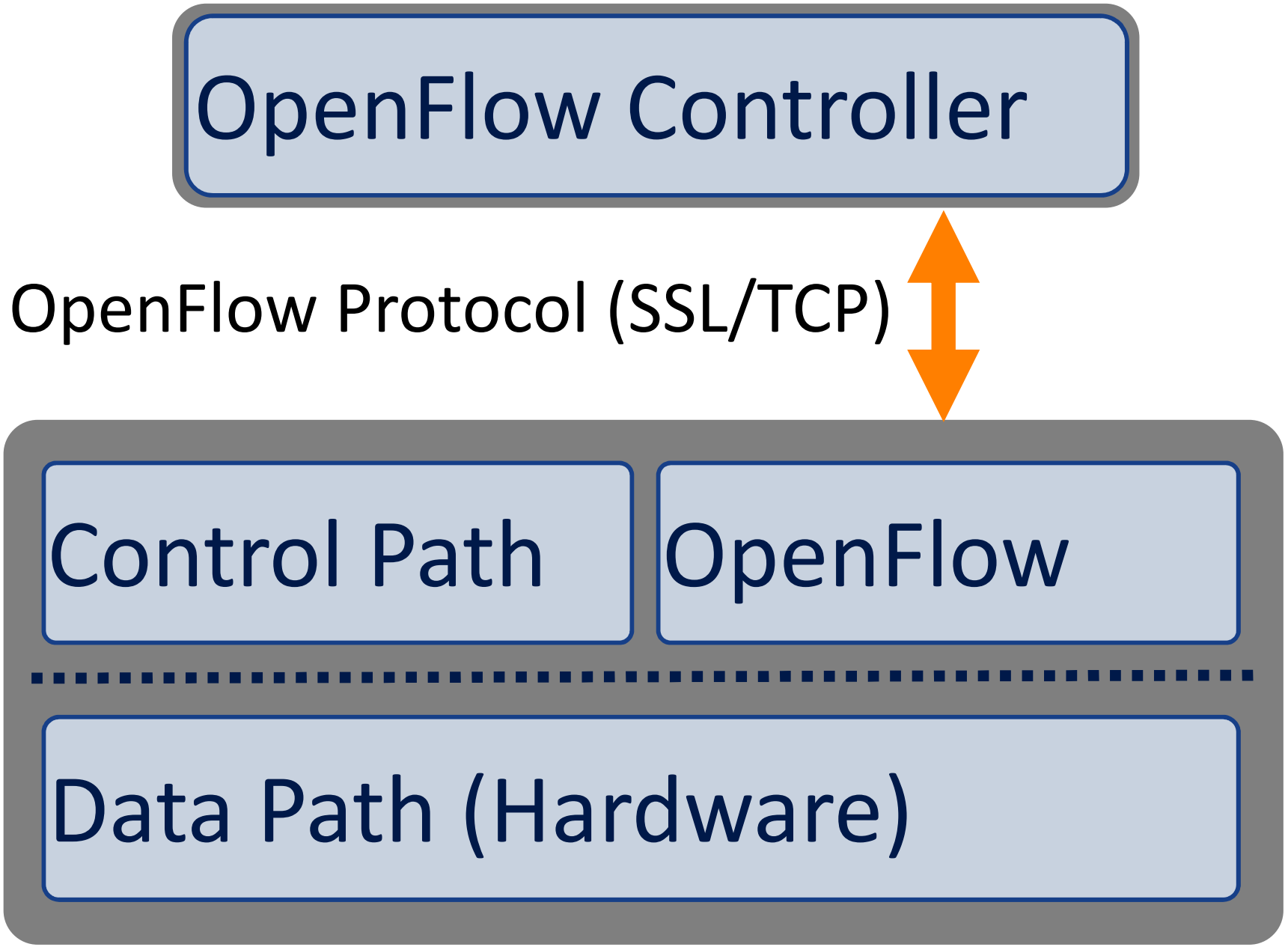
Ethernet Switch



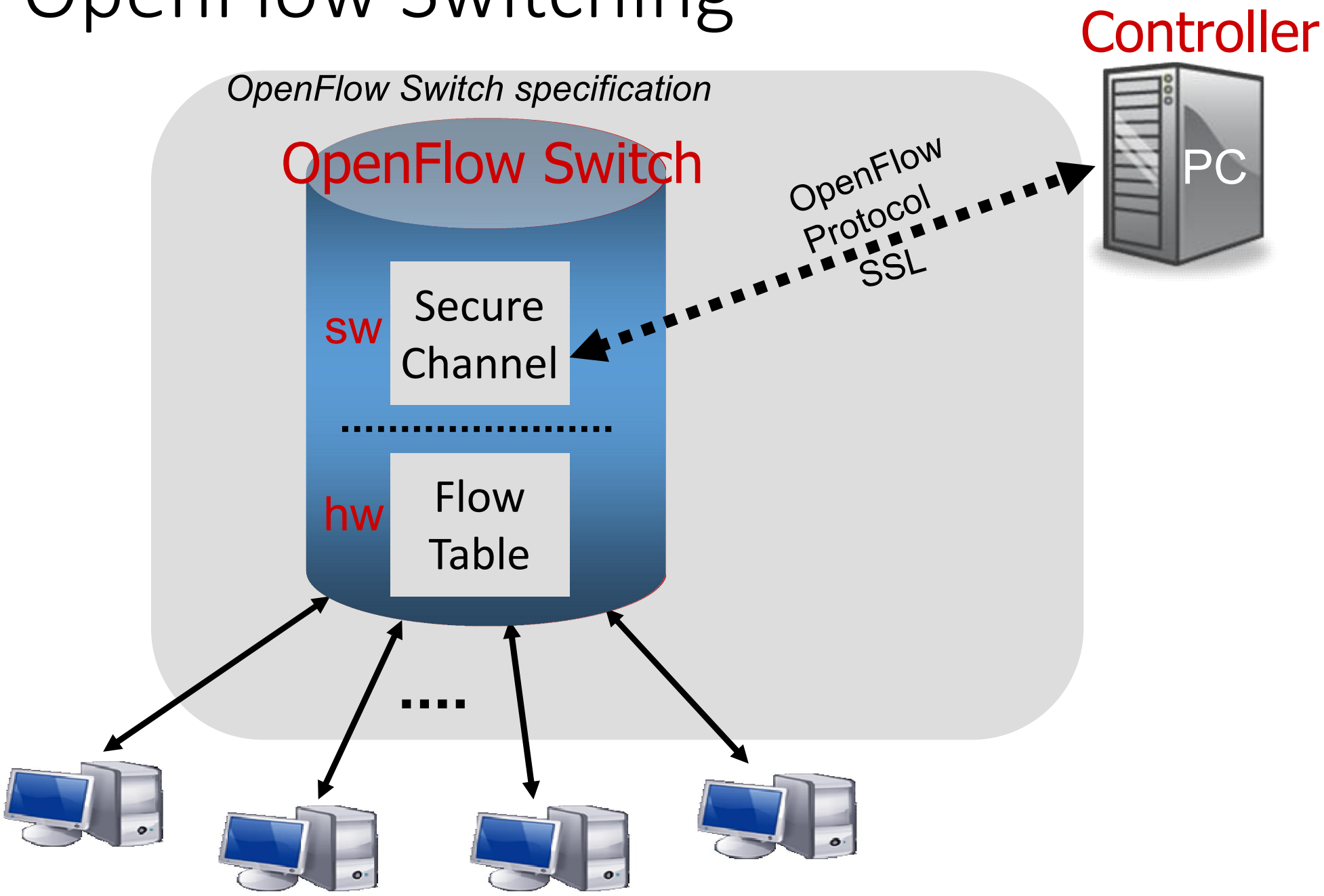





Control Path (Software)
Data Path (Hardware)

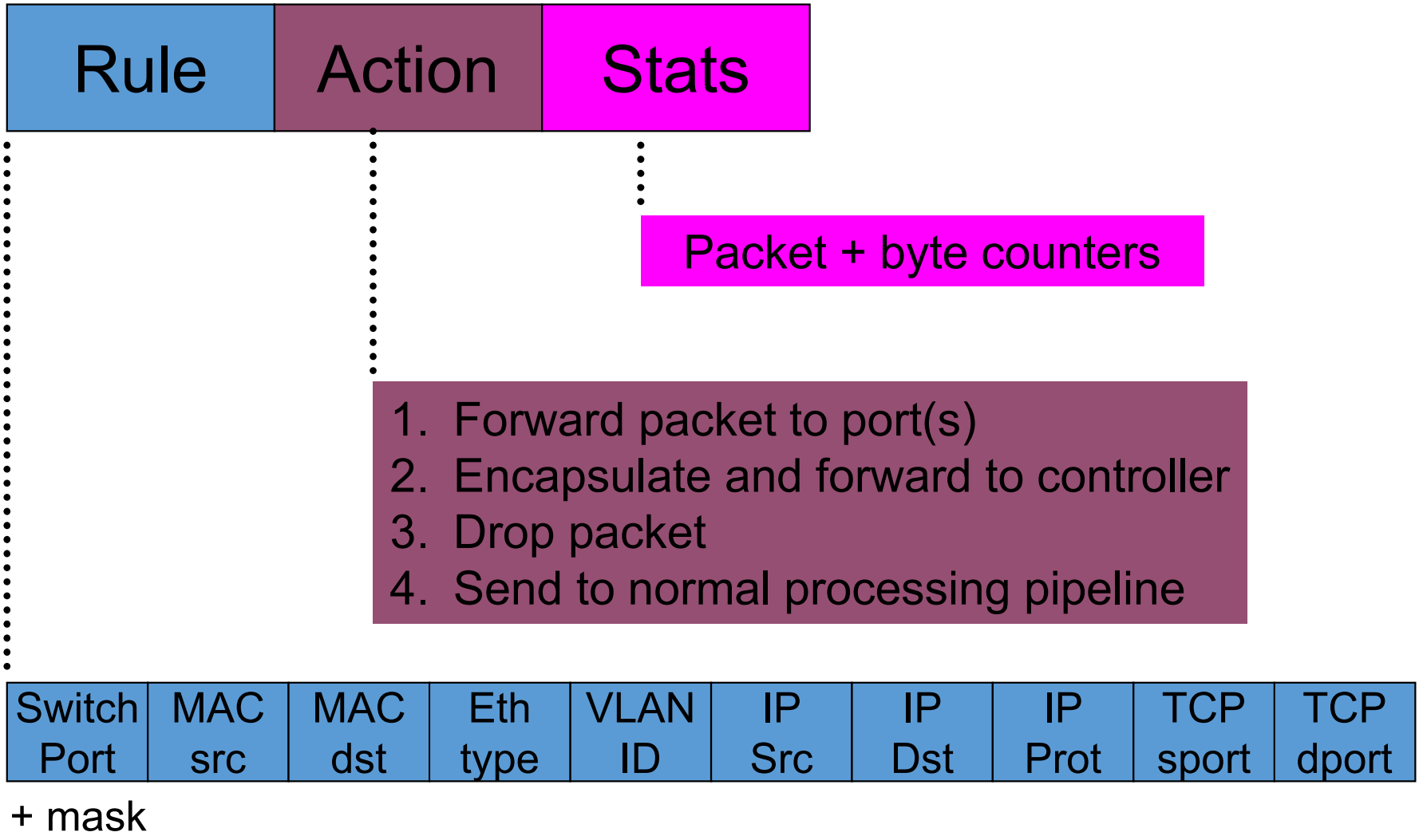

OpenFlow Controller
OpenFlow Protocol (SSL/TCP)
Control Path
OpenFlow
Data Path (Hardware)

OpenFlow Switching



Flow Table Entry

"Type 0" OpenFlow Switch



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

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

Firewall

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