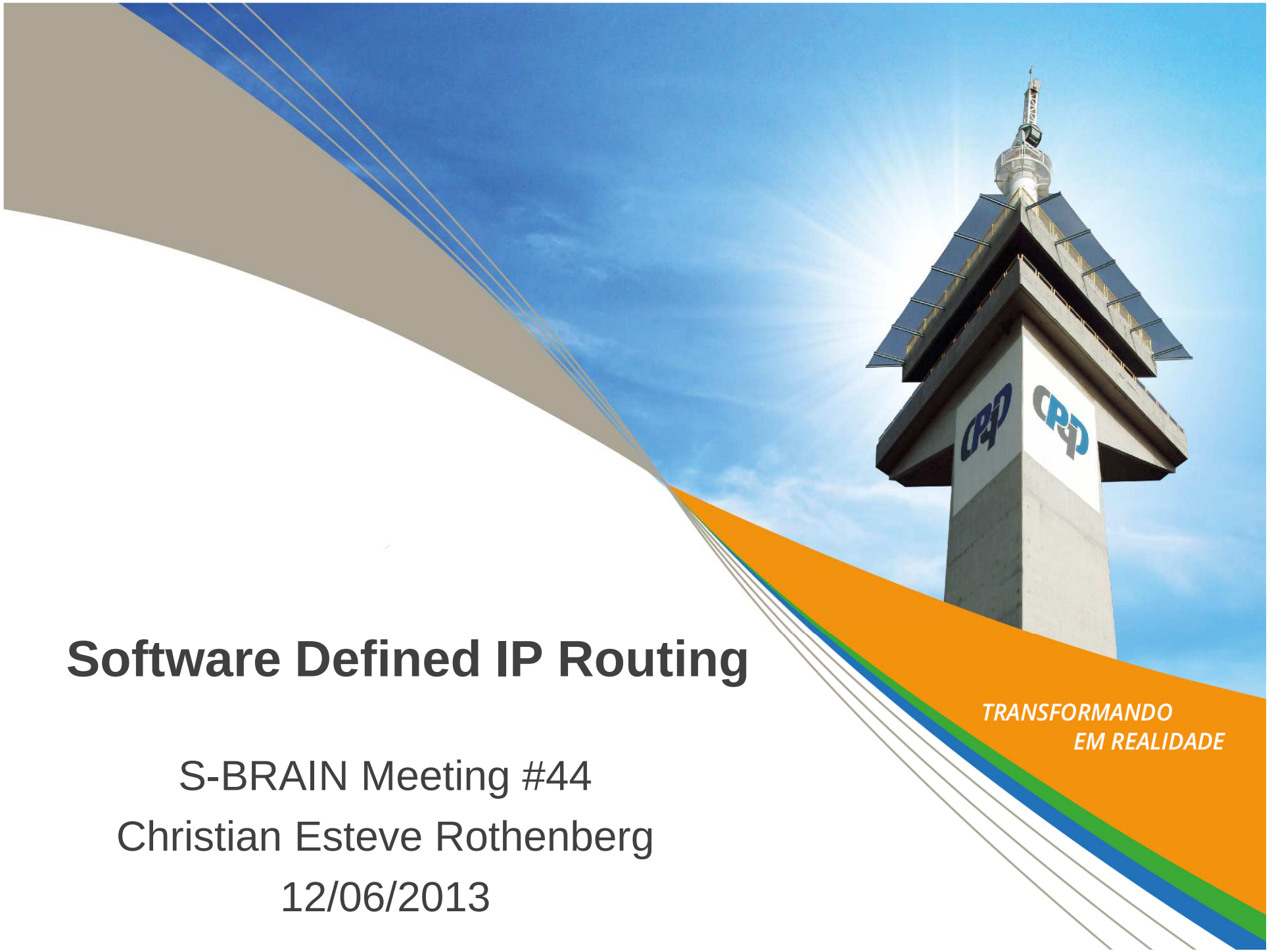


Software Defined IP Routing

S-BRAIN Meeting #44

Christian Esteve Rothenberg

12/06/2013

Agenda

- A view on Software Defined Networking
- (OpenFlow basics)
- IP Routing & OpenFlow/SDN
- RouteFlow
  - Architectural discussions
  - Research perspectives
  - Use Cases and Pilots
- Final remarks

Disclaimer

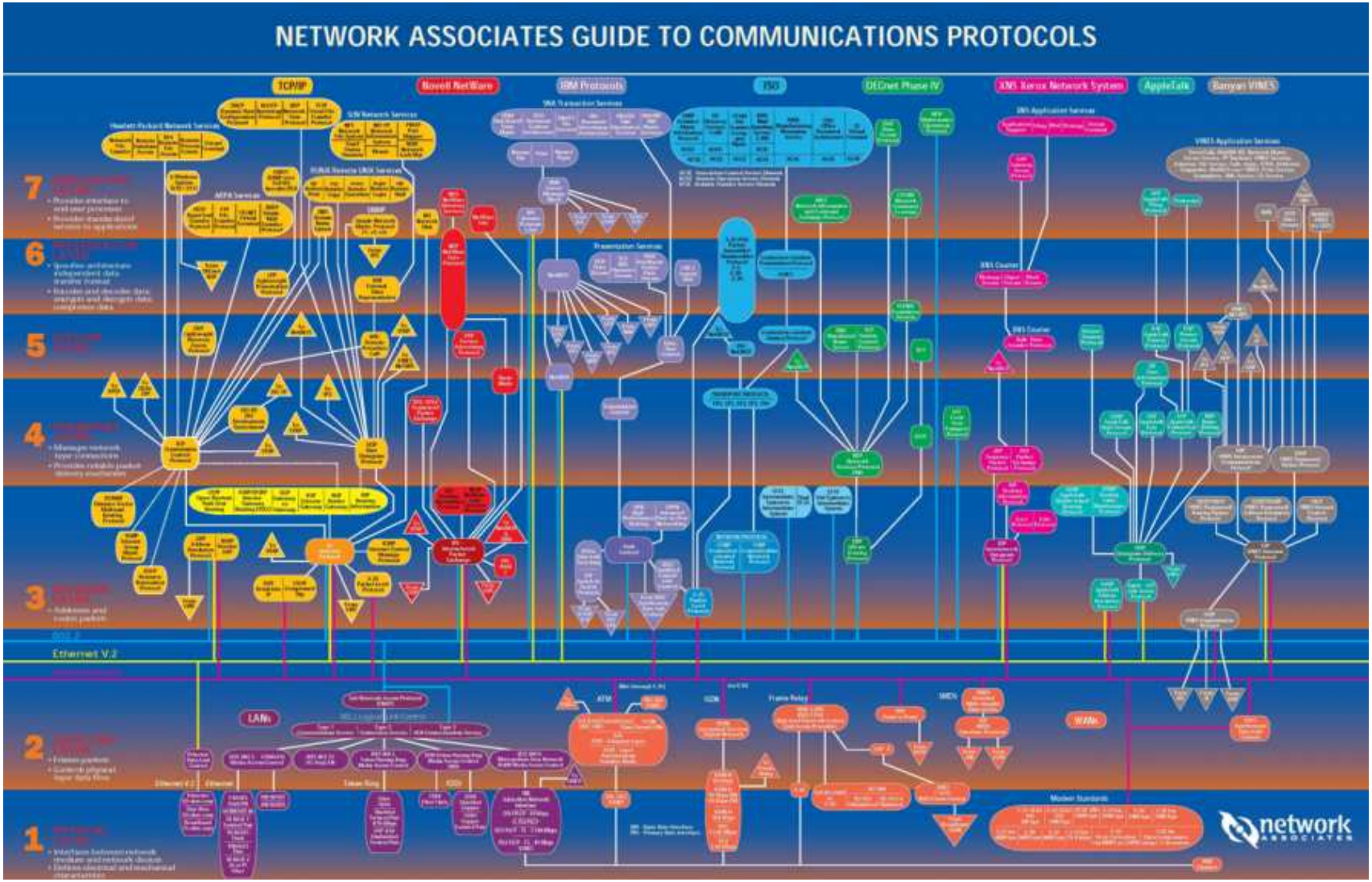
Personal view on SDN

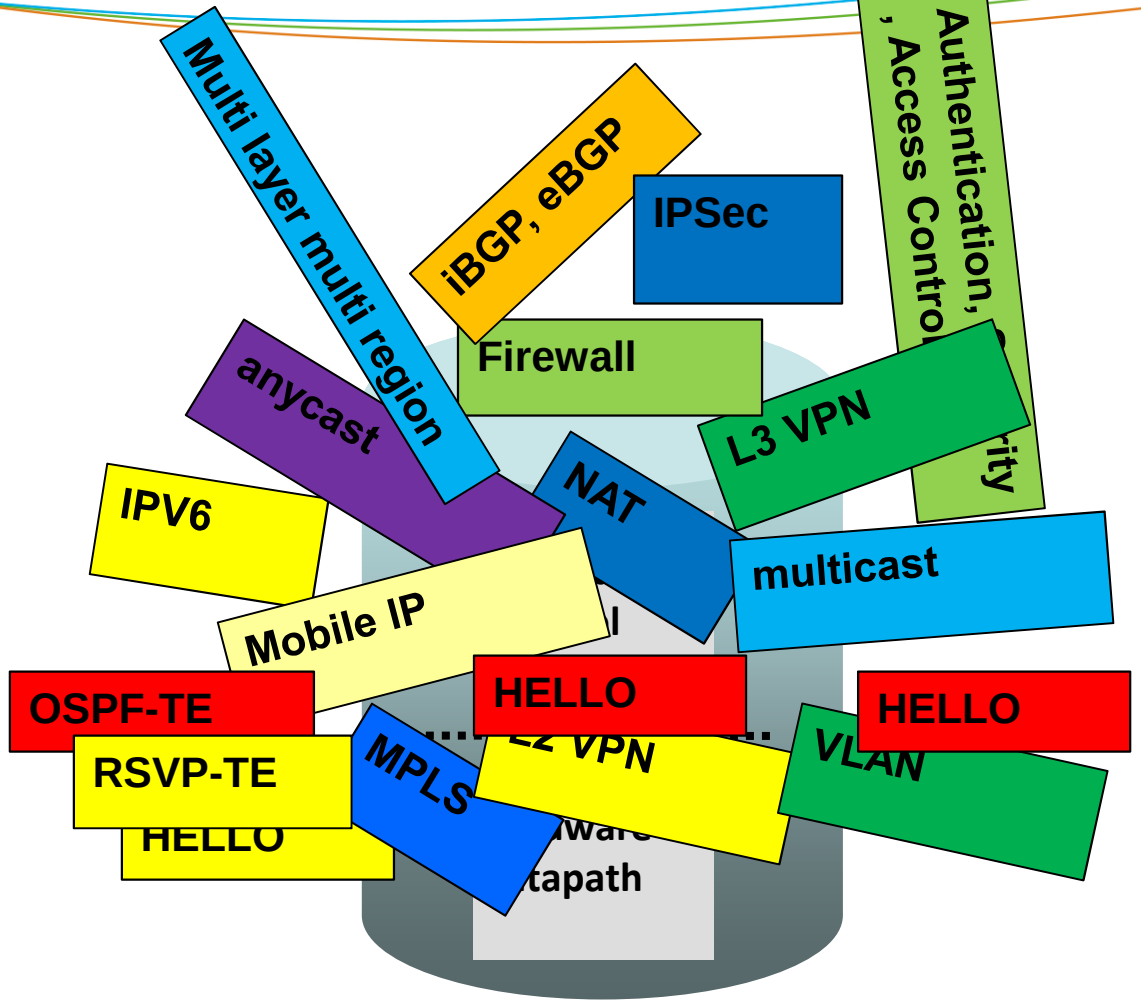
- Not by CPqD
- Not by ONF (or any affiliated company)

Actually, not that original...

- See References and Credits

Where are we today?





Many complex functions baked into the infrastructure

OSPF, BGP, multicast, differentiated services,
Traffic Engineering, NAT, firewalls, MPLS, redundant layers, …

An industry with a “mainframe-mentality”

Source: Stanford/Berkeley

How did we get here?

A guaranteed recipe for disaster:

1. **Invent a new data plane mechanism**
2. **Hack a new control plane for it**
3. **Jump back to 1**

The ability to master complexity is not the same as the ability to extract simplicity

– S. Shenker

Source: T. Koponen

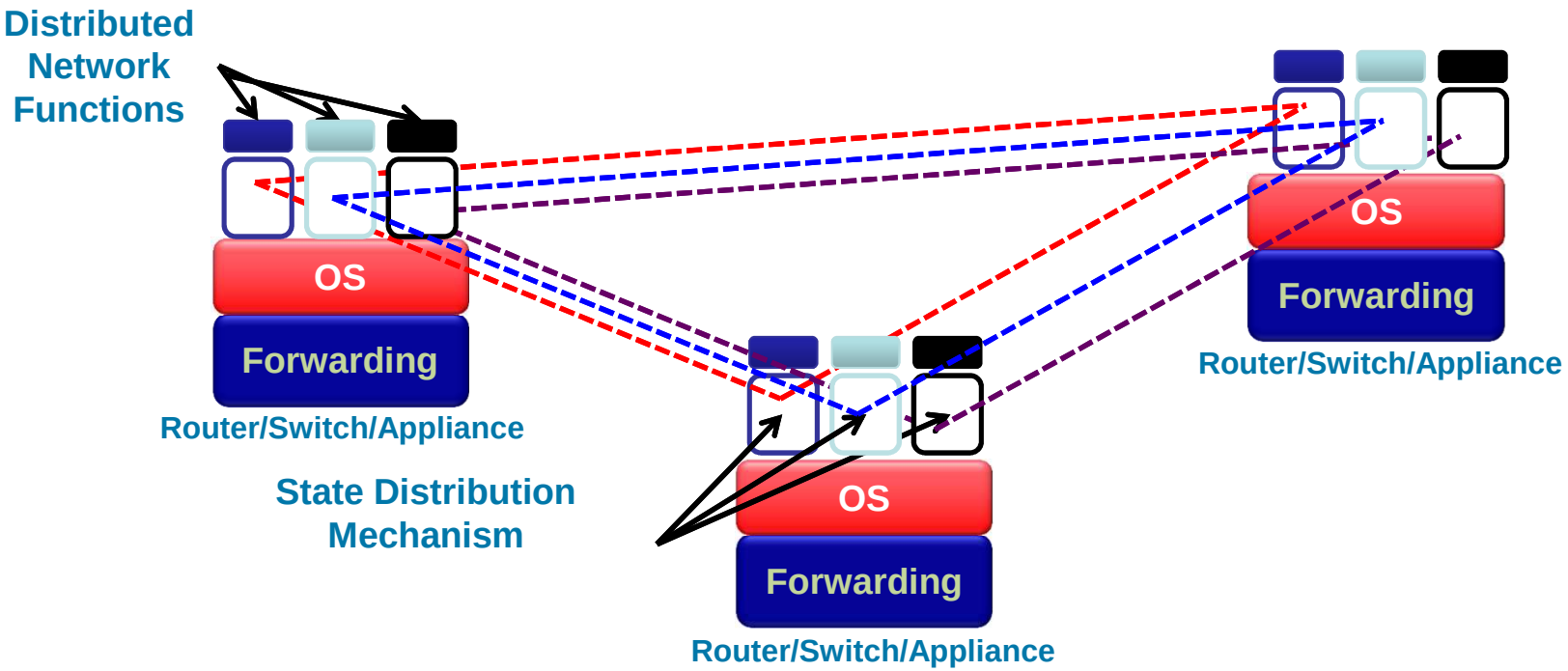
Problem: No Abstractions for Control Plane

Addition of a new function to the network

- Highly complex distributed system problem

Networks too difficult to program and to reason about

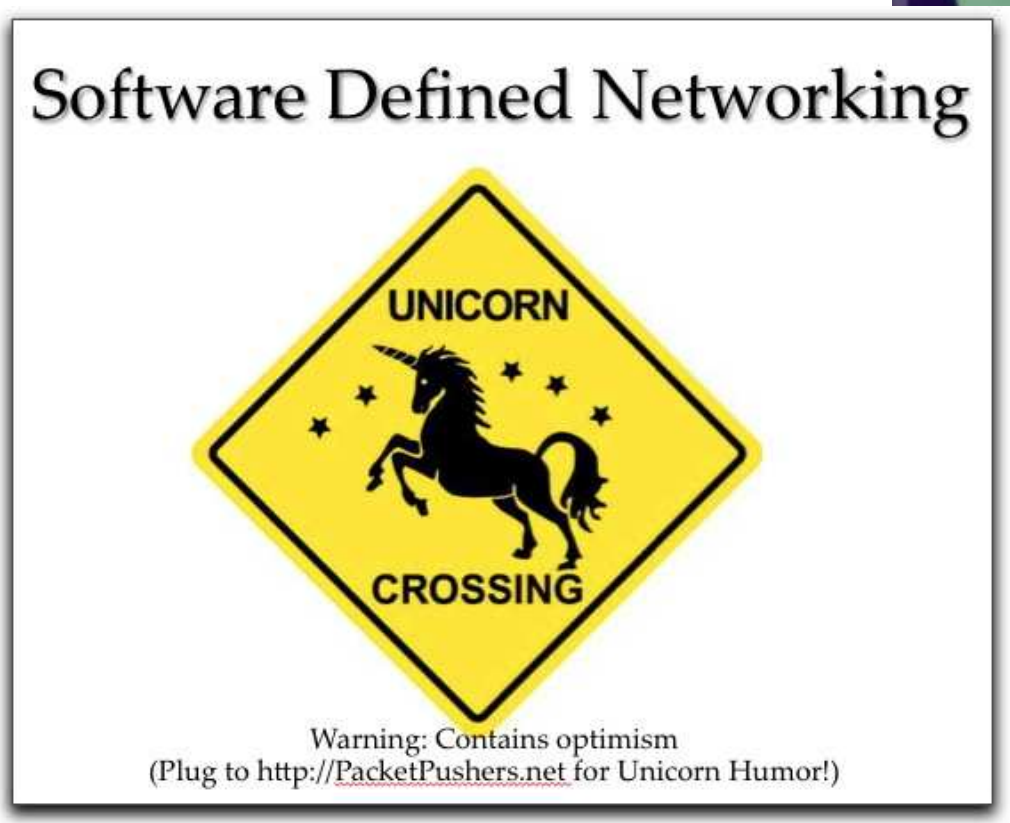
- no good abstractions and interfaces



Not good for even network vendors!

Source: Stanford/Berkeley

SDN to the rescue!



Source: packetpushers.net

April/2009 : The term SDN was coined

http://www2.technologyreview.com/article/412194/tr10-software-defined-networking/

MIT Technology Review

10 BREAKTHROUGH TECHNOLOGIES 2009

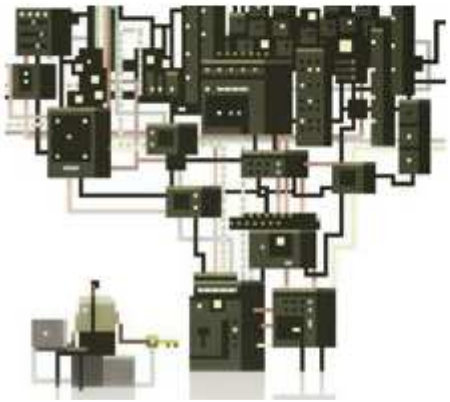
TR10: Software-Defined Networking

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.

Frustrated by this inability to fiddle with Internet routing in the real world, Stanford computer scientist Nick McKeow-

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

"*OpenFlow is SDN, but SDN is not OpenFlow*"

- Does not say much about SDN

"Let`s call whatever we can ship today SDN"

– Vendor X

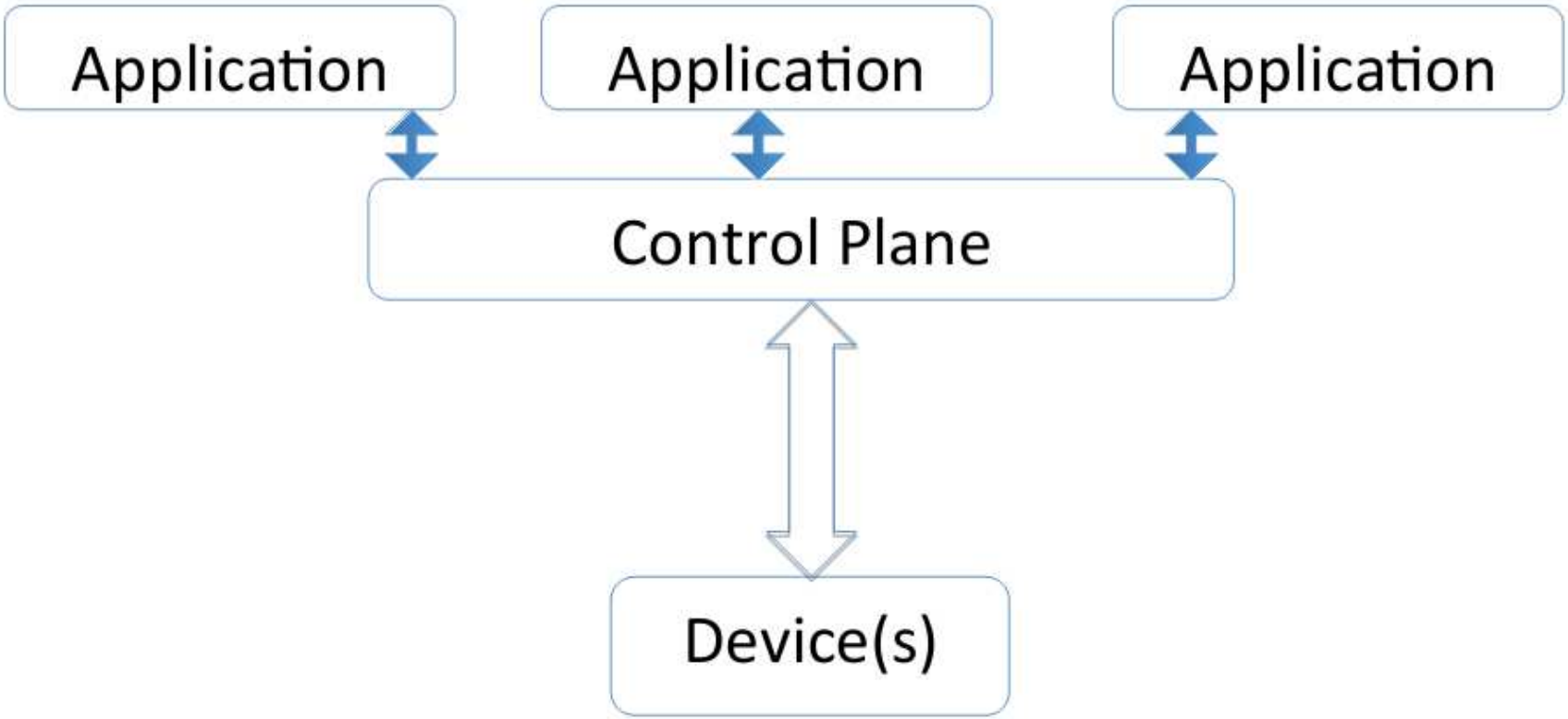
"SDN is the magic buzzword that will bring us VC funding"

– Startup Y

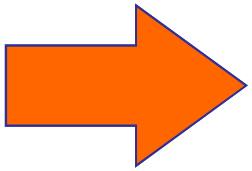
"SDN is the magic concept that will get my paper/grant accepted"

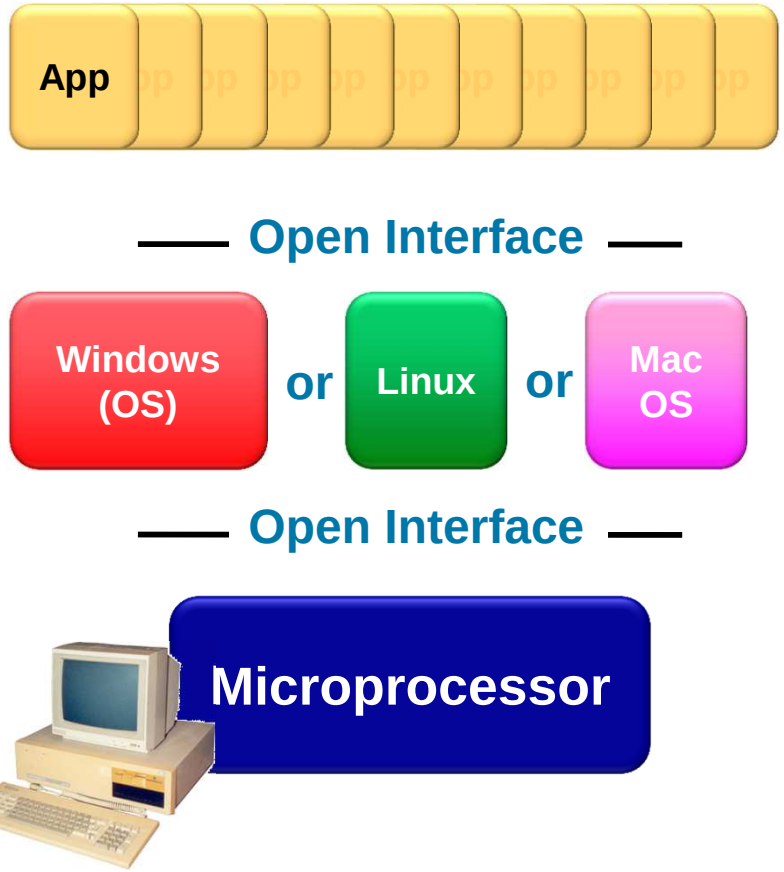
– Researcher Z

Generic SDN Architecture

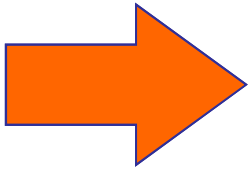


Vertically integrated
Closed, proprietary
Slow innovation
Small industry



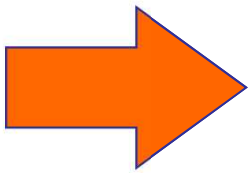


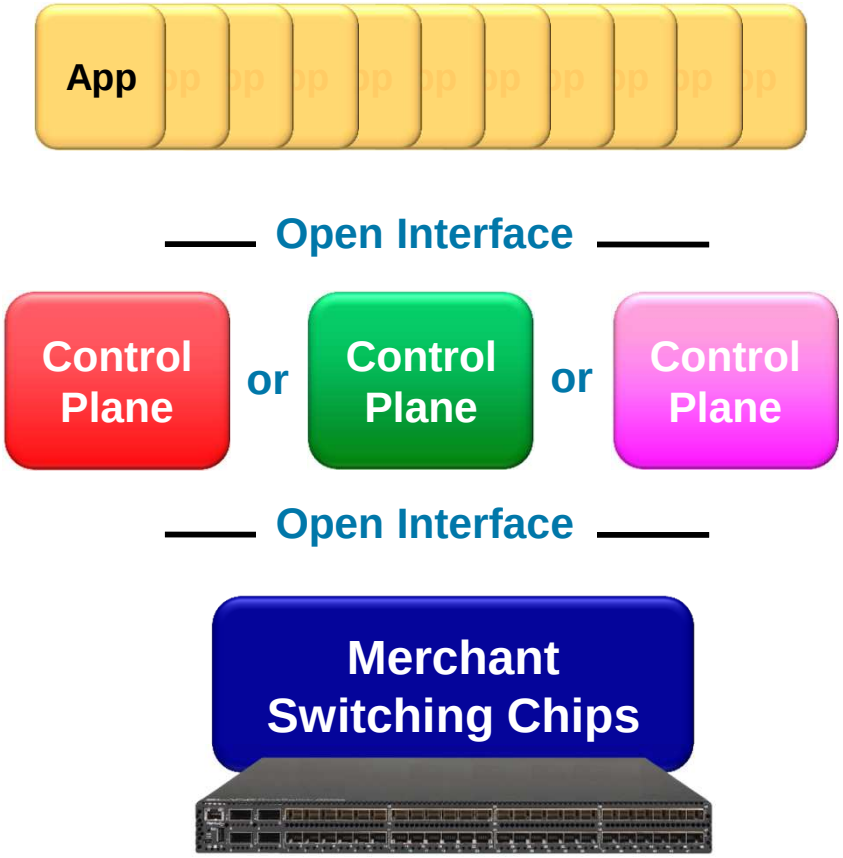
Horizontal
Open interfaces
Rapid innovation
Huge industry



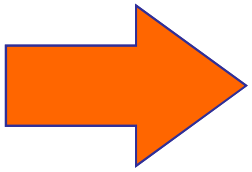
Source: McKeown

Vertically integrated
Closed, proprietary
Slow innovation





Horizontal
Open interfaces
Rapid innovation



Source:
McKeown


SDN in 2010
Anonymous uploader

SDN in 2011 – 2012

SDN in 2013

Academia

Start-up 1
Start-up 2
...
Start-up n

Vendor A

Vendor B

Vendor C

Origins

Martin Casado

Ethane

1. Programmatic control of Enterprise networks
2. Global policy, directly enforced
3. Global vantage point
4. OpenFlow

Research Community:
How to deploy new ideas?

1. NSF/GENI
2. OpenFlow/SDN on 10 campuses
3. Research demonstrations
4. Now on 100+ campuses
5. US, Europe, Asia, Brazil

Industry Trend:
Networks being built this way

1. Data Center Networks
2. WANs
3. Enterprise and WiFi
4. Vendors startups emerging

Source: D. Pitt (ONF)

OpenFlow standards

Evolution path:

- OF 1.0 (03/2010): Most widely used version, MAC, IPv4, single table (from Stanford)
- OF 1.1 (02/2011): MPLS tags/tunnels, multiple tables, counters (from Stanford)
- OF 1.2 (12/2011): IPv6, extensible expression
- OF-Config 1.0 (01/2012): Basic configuration: queues, ports, controller assign
- OF 1.3.0 (04/2012): Tunnels, meters, PBB support, more IPv6
- OF-Config 1.1 (04/2012): Topology discovery, error handling
- OF-Test 1.0 (2H2012): Interoperability conformance test processes, suites, labs
- OF 1.3.2 (est. May 2013), 19 errata, final review
- OF 1.4 (est. June 2013), 9 changes + 13 extensions

Goals:

- Widespread adoption, experimentation w/OF 1.3.x
- Accommodate current merchant silicon
- Move beyond limitations of current merchant silicon

Source: ONF

Technical activities

Chartered Working Groups

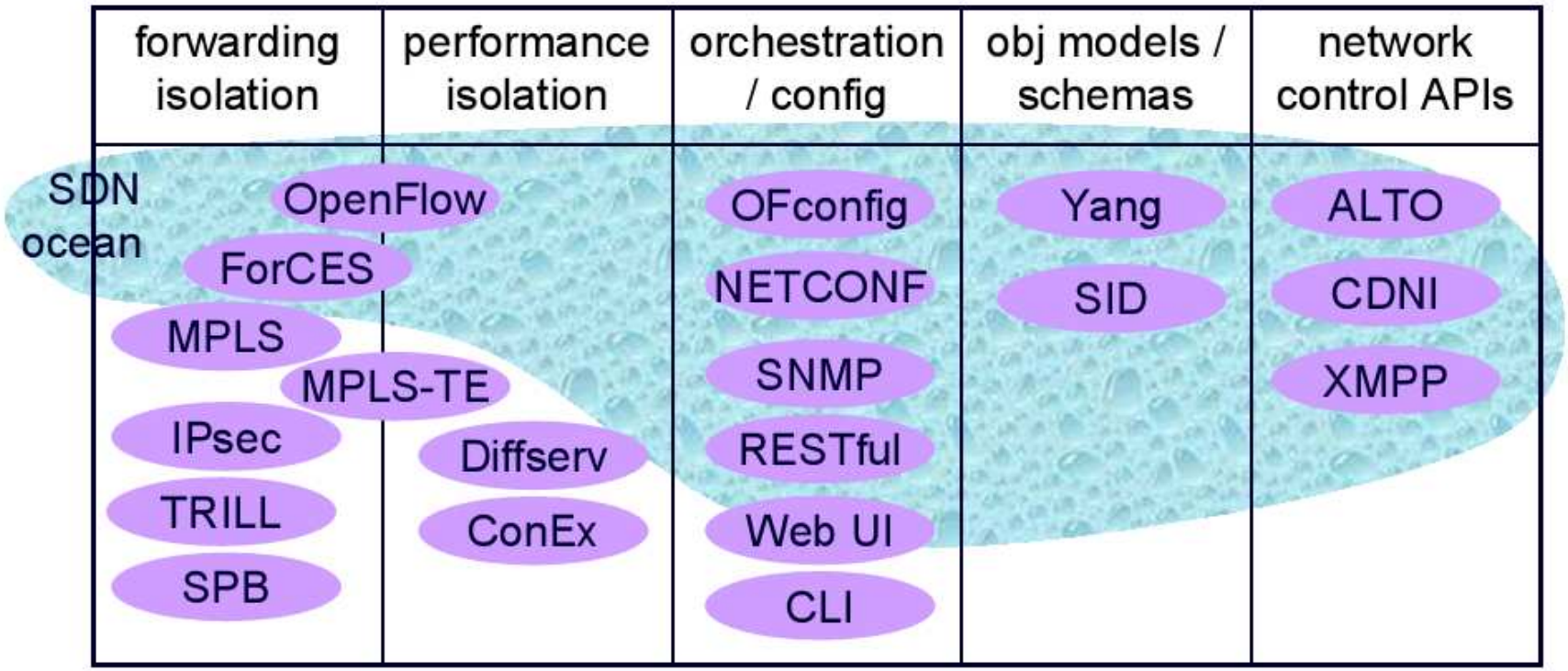
- Extensibility (chair: Jean Tourrilhes, HP): OpenFlow protocol
- Config-mgmt (chair: Deepak Bansal, Microsoft): basic switch configuration
- Testing-Interop (chair: Michael Haugh, Ixia): conformance, interop., benchmarking
- Hybrid (chair: Jan Medved, Cisco): mixed OpenFlow/legacy switches networks → Migration WG (chair: Justin Dustzadeh, Huawei)

Discussion Groups

- OpenFlow-Future: forwarding-plane models
- NorthboundAPI: how the network relates to the applications
- NewTransport: OpenFlow for optical, circuits, wireless
- Market Education (chair: Isabelle Guis, Big Switch): marketing, customer value

Source: ONF

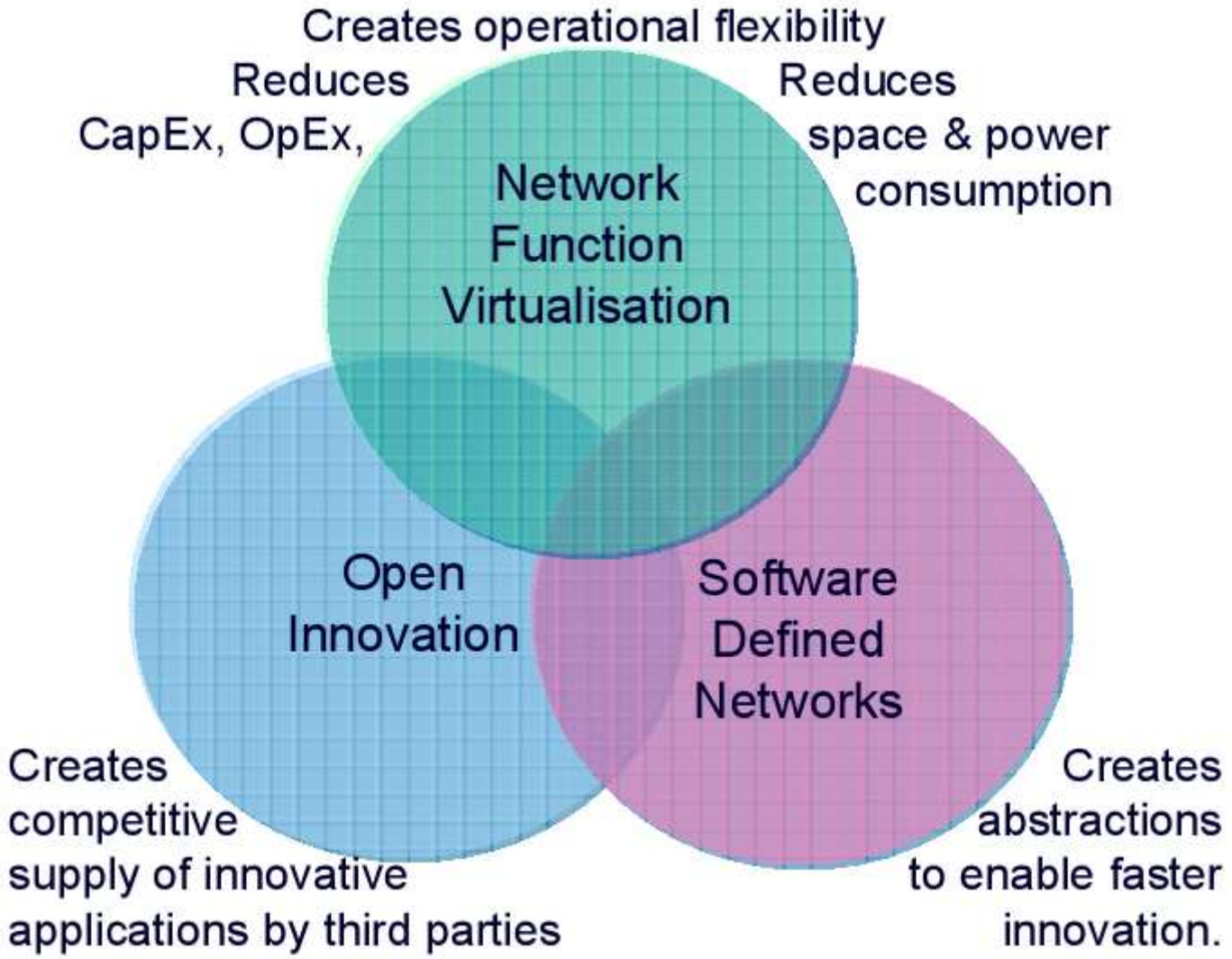
Mapping of technologies to problems



- we could focus on solving a particular problem (column)
- we could also ask
  - is an integrated solution (multi-column) good?
  - or "do one thing and do it well" so operators can pick & choose rather than lock-in to an über-solution?

Source: B. Briscoe, slides-84-sdnrg-0.pdf

Strategic Networking Paradigms for Network Operators



Source:NFV

Towards a SDN taxonomy

	Data plane (Elements used for traffic handling)	Controller solutions (Decoupled control plane)	Management (Extensible mgmt SW API)
L2-L4 routing	• **SDN-D-PS**witch: Simplified physical data plane elements without a control plane (e.g., Pica8 Pronto) • **SDN-D-VS**witch: Simplified virtual data plane elements without a control plane (e.g., OpenVSwitch) • **SDN-D-F**abric: Data plane elements, with inbuilt control plane, that collaborate to form a unified fabric (e.g., Pluribus server-switch)	• **SDN-C**-**O**pen**F**low: Control plane using the OpenFlow API (e.g., BigSwitch Floodlight) • **SDN-C-OV**erlay: Control plane managing network overlays (e.g., Nicira NVP, PLUMgrid)	• **SDN-N-M**anagement: Value-added network management software (e.g., Cyan Blue Planet, OpenStack Quantum, Cariden NS-OS)
L4-L7 services	• **SDN-S-D**ataplane: Data plane elements to process sessions (e.g., Linerate Proxy) • **SDN-S**-**F**abric: Scale-out enforcement of L4-L7 services where dataplane and control plane are co-located (e.g., Cisco vPath)	• **SDN-S**-**C**ontrol: Decoupled control plane for enforcing policy (e.g., vArmour)	• **SDN-S-O**rchestrator: Platform for elastic L4-L7 services (e.g., Embrane Heleos)

Source: S. Srini Seetharaman, SDNCentral

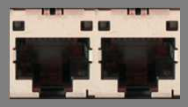
What is OpenFlow?

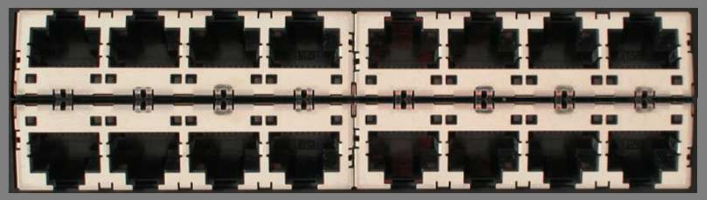
Short Story: OpenFlow is an API

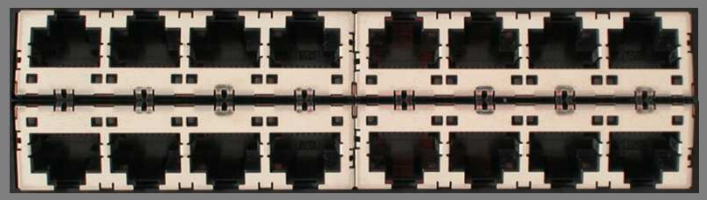
- Control how packets are forwarded (and manipulated)
- Implementable on COTS hardware
- Make deployed networks programmable
  - not just configurable
  - Vendor-independent
- Makes innovation easier
- Goal (experimenter's perspective):
  - Validate your experiments on deployed hardware with real traffic at full 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).

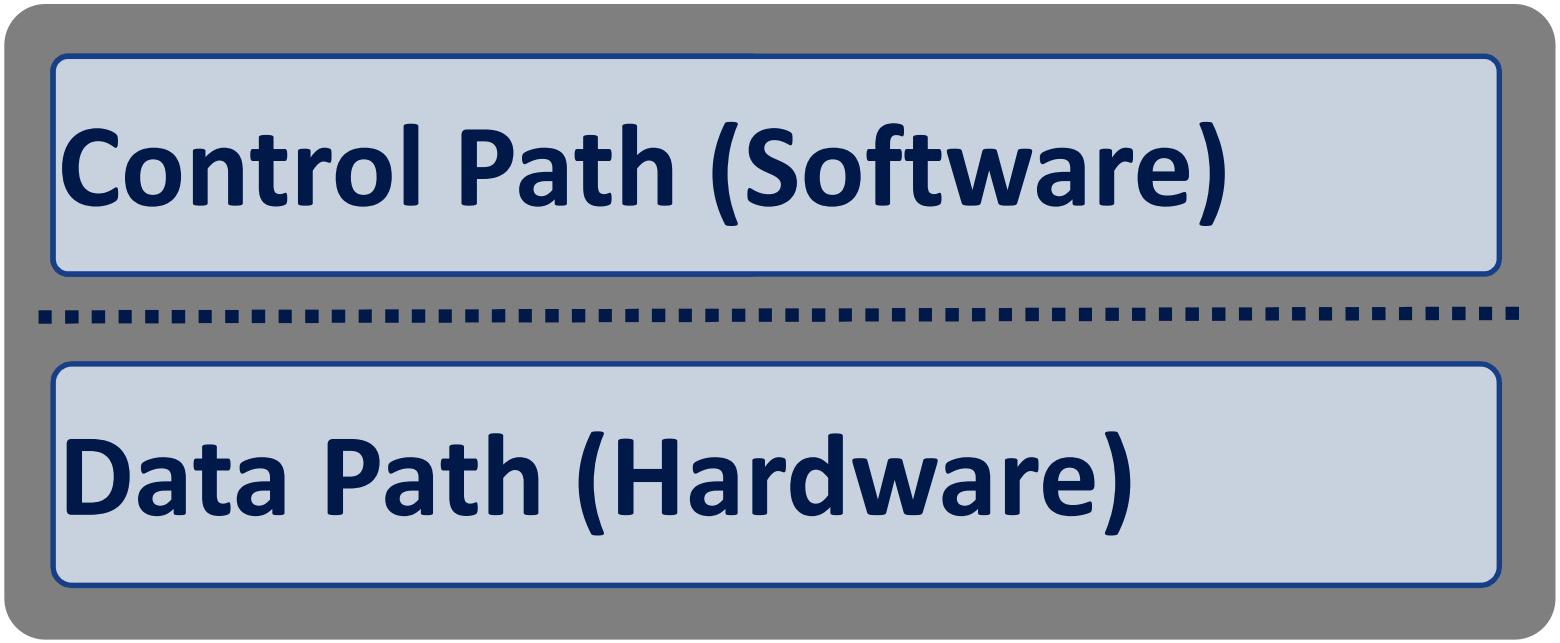
How does OpenFlow work?

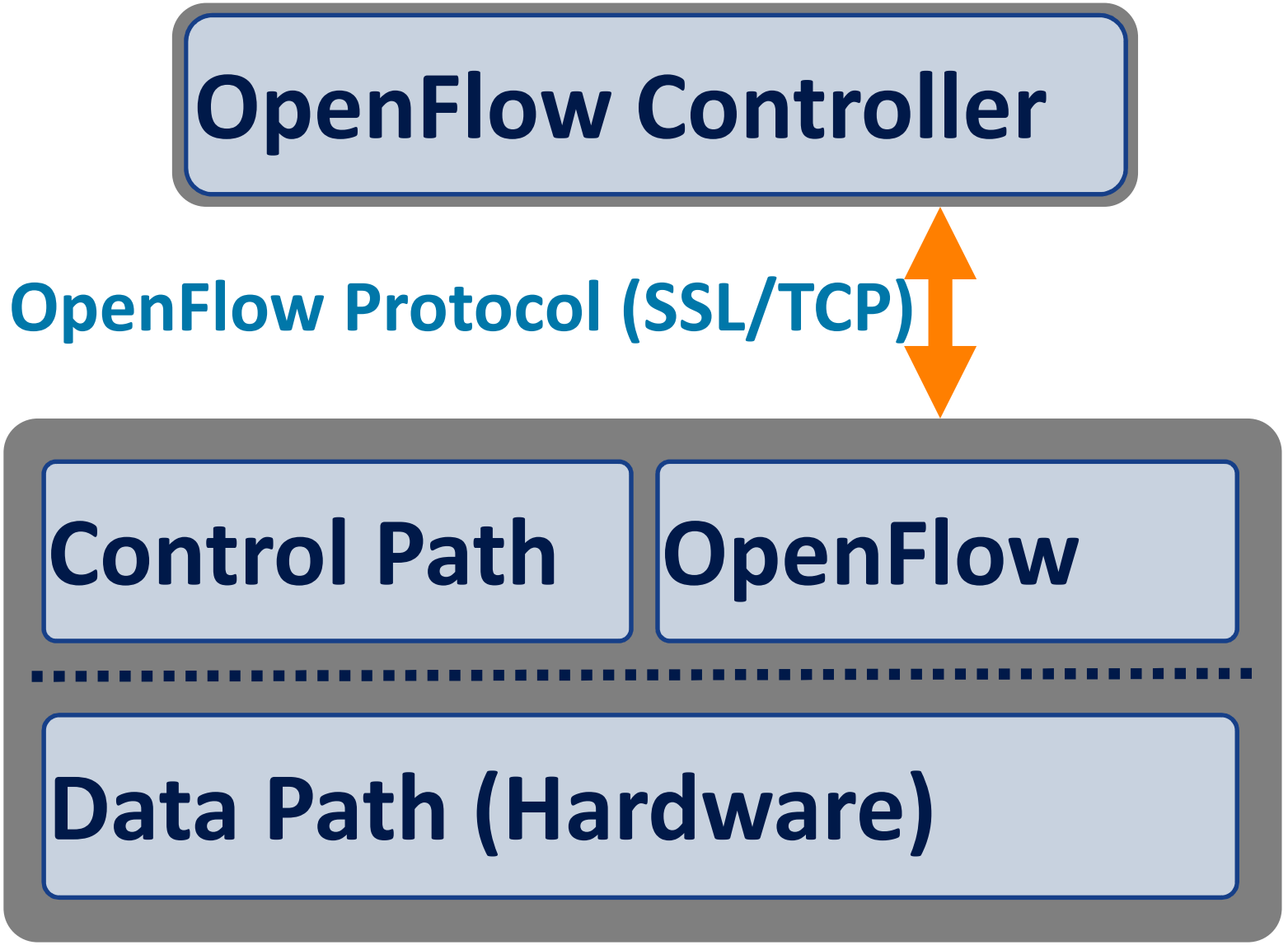
Ethernet Switch



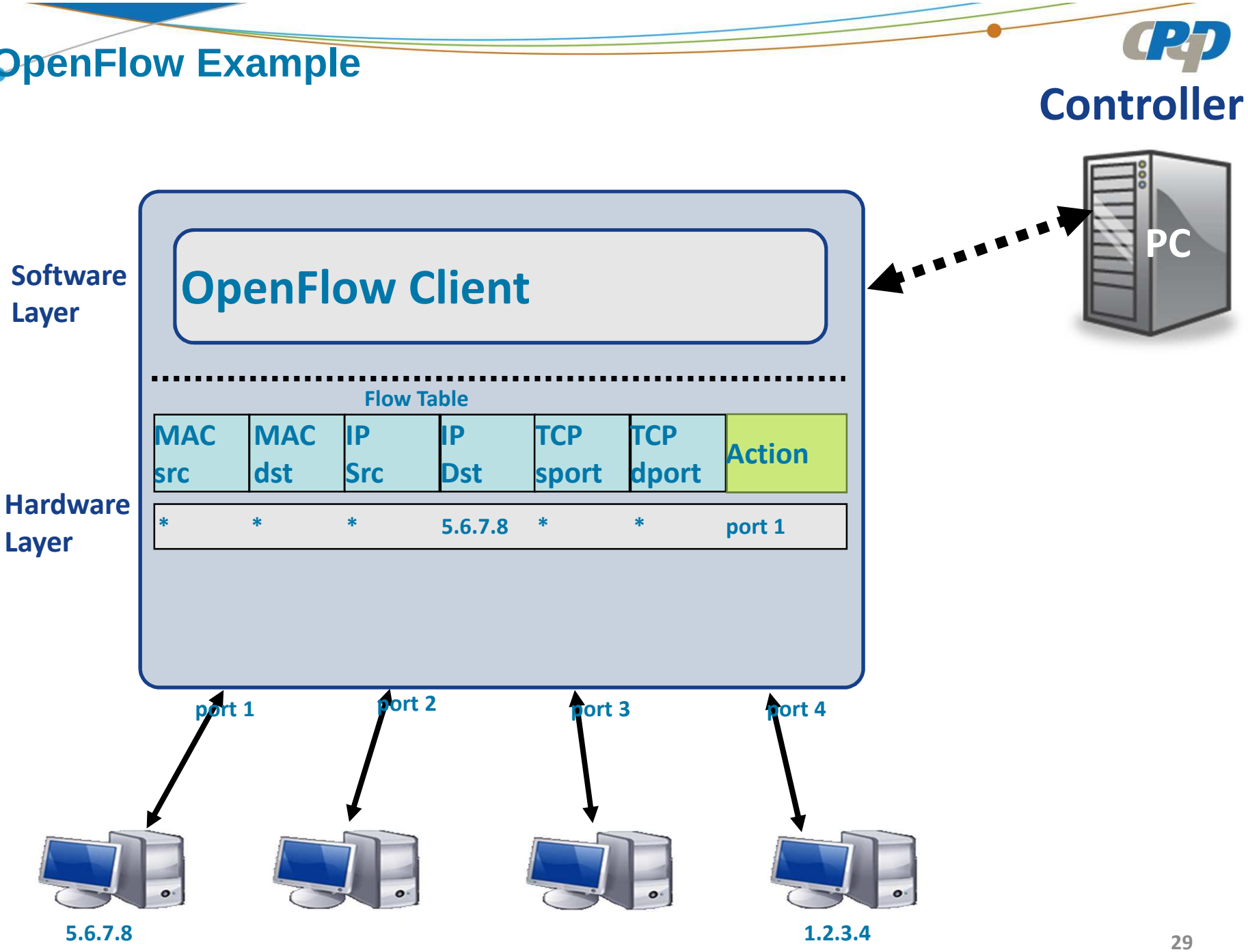





Control Path (Software)
Data Path (Hardware)

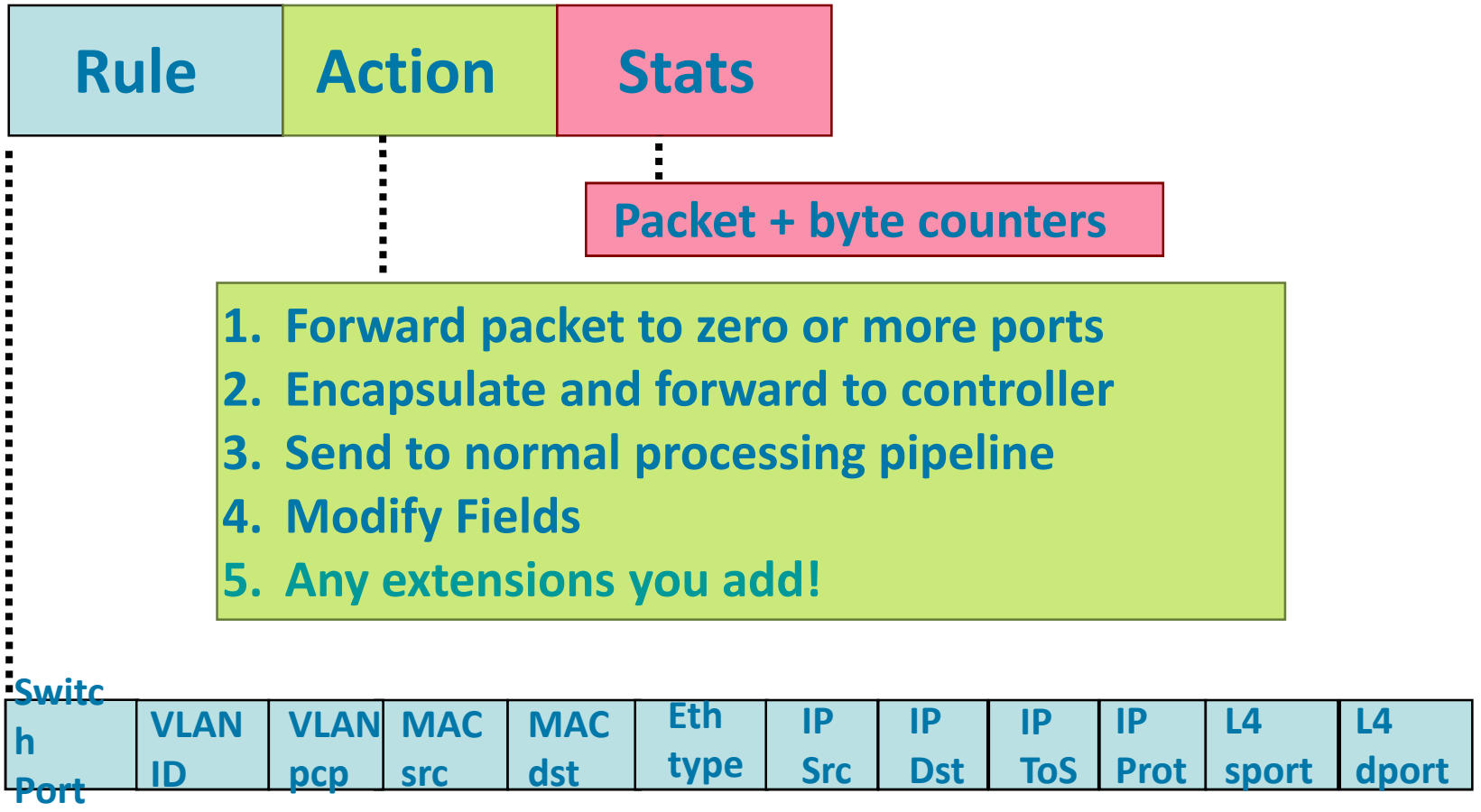

OpenFlow Controller
OpenFlow Protocol (SSL/TCP)
Control Path
OpenFlow
Data Path (Hardware)

OpenFlow Example



OpenFlow Basics

Flow Table Entries



+ mask what fields to match

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

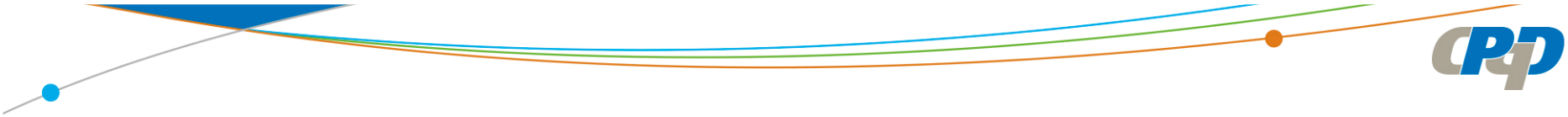
Examples

Routing

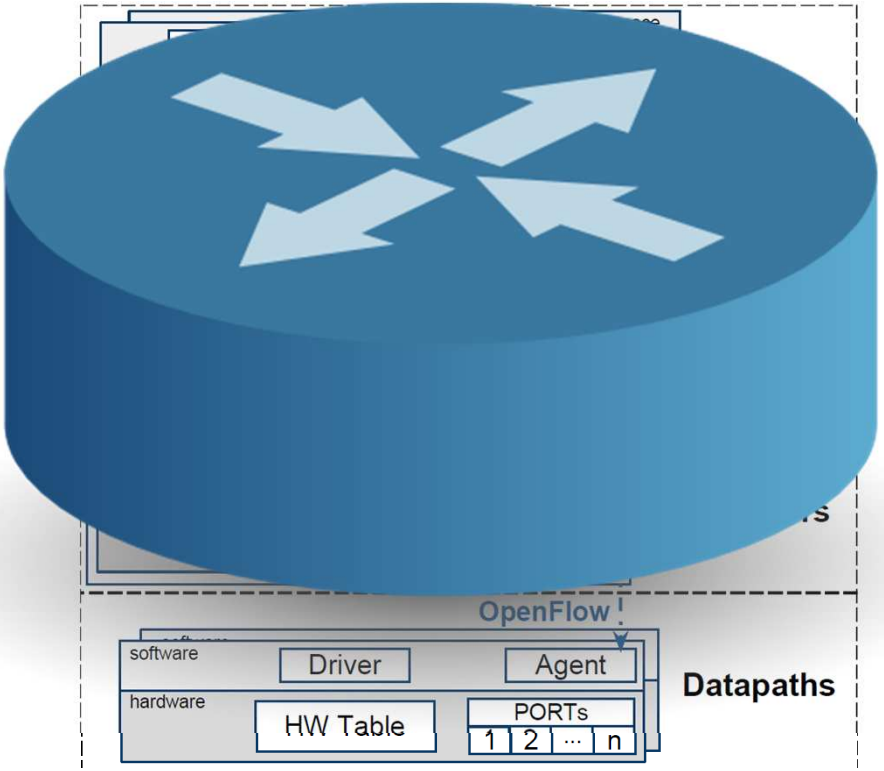
Switch Port	MAC src	MAC dst	Eth type	VLAN ID	IP Src	IP Dst	IP Prot	TCP sport	TCP dport	Action
*	*	*	*	*	*	5.6.7.8	*	*	*	port6

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



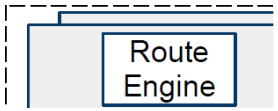
OPENFLOW & IP ROUTING



Control Plane

Data Plane

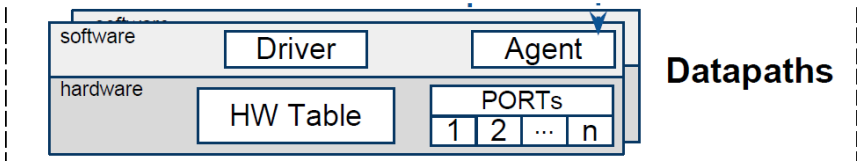
Open-Source Routing Stacks



Linux

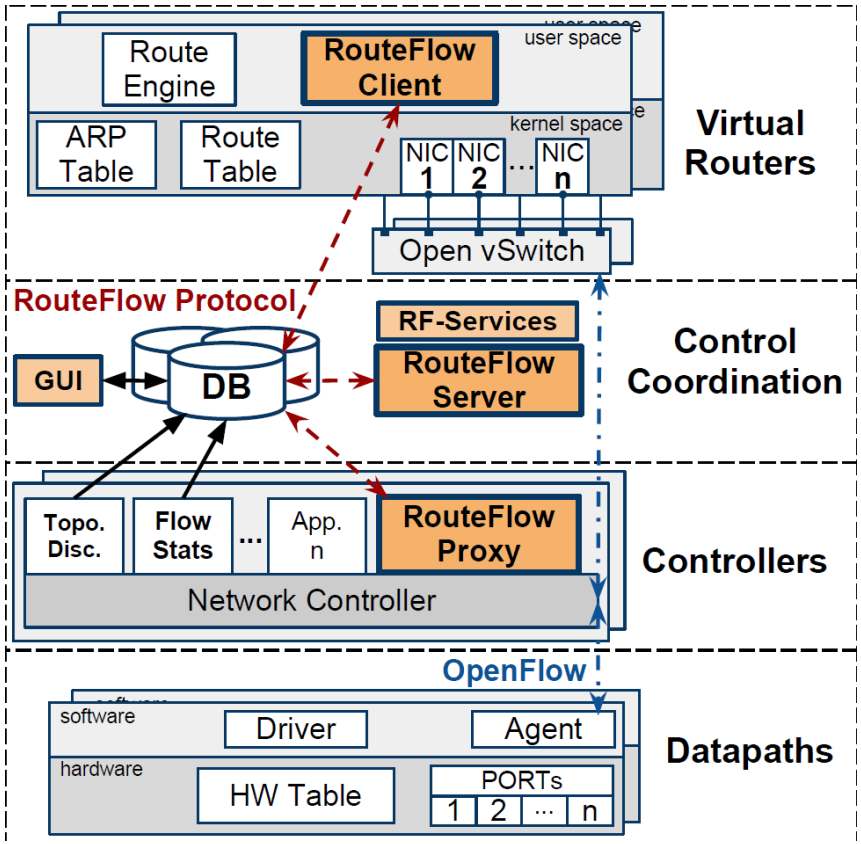
Control Plane

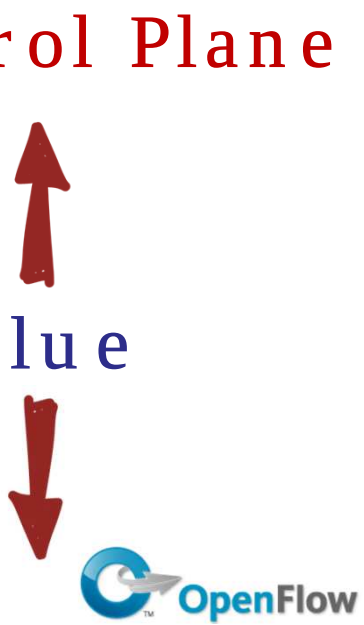
Glue



Datapaths

Data Plane



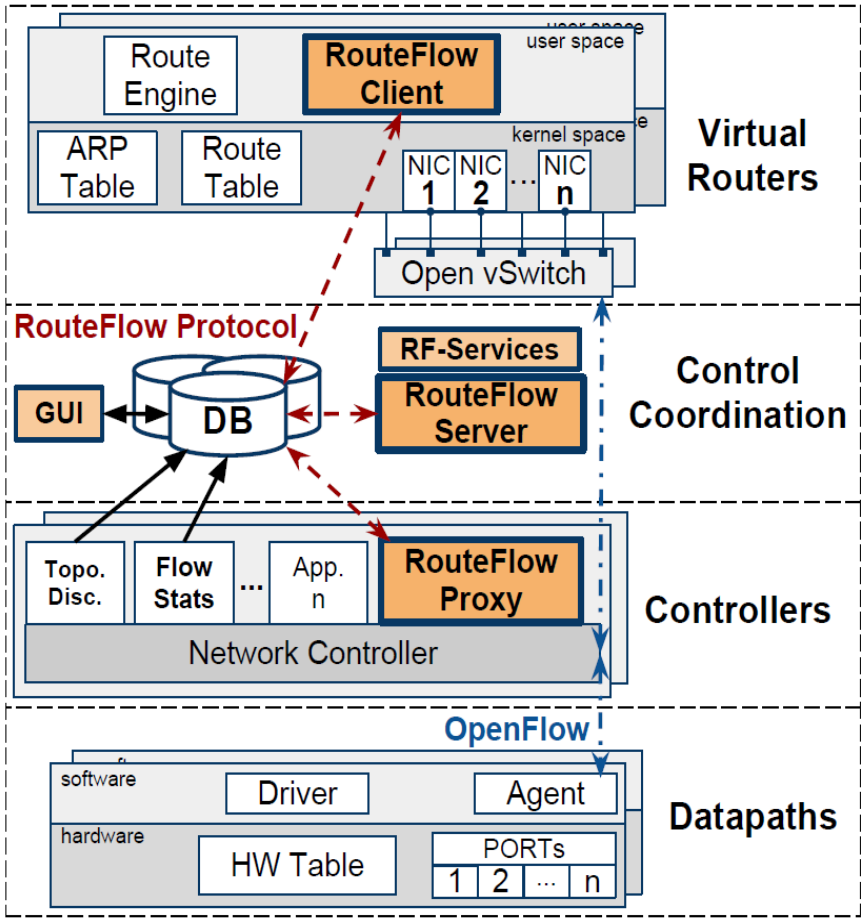


OpenFlow protocol used as:

1) A mechanism to install the FIB

2) A mechanism to steer control plane traffic

Design



- Modular architecture
  - RF-Proxy
  - RF-Server
  - RF-Client
- Database layer
  - JSON-based IPC
  - Resillient core state
  - Programmer-friendly
- Multi-Controller support
  - NOX, POX, Ryu, Floodlight

Software Defined IP Routing

BGP OSPF ISIS LDP

RouteFlow

Open interface

Controller Controller Controller

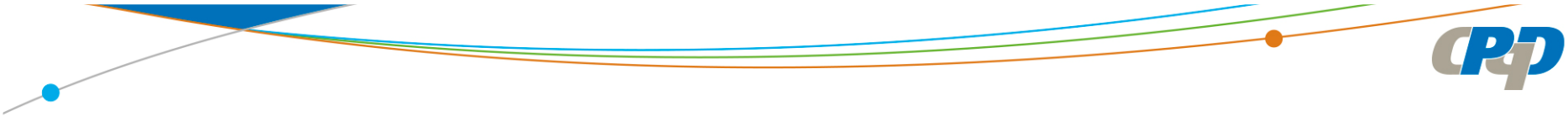
Open interface

OpenFlow Switches

High cost
Specialized config.
Closed source
Slow innovation

Low cost (commodity)
Multi-vendor modularity
Open source
Rapid innovation

Source: McKeown



Architectural discussions

ROUTEFLOW

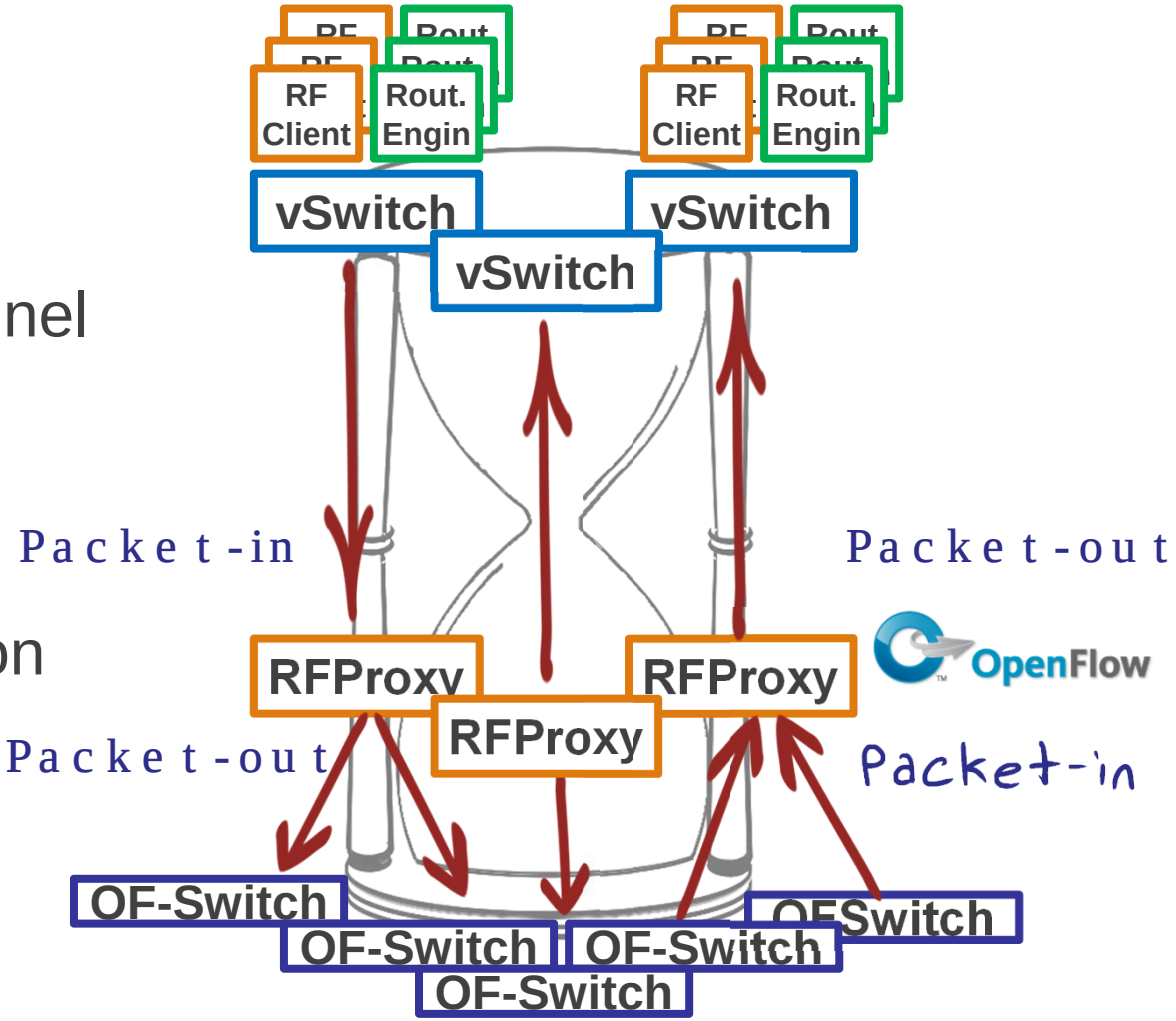
Architectural Discussion

Control-Data Channel

- OpenFlow-based

Physical Distribution

- Scalability
- Resiliency



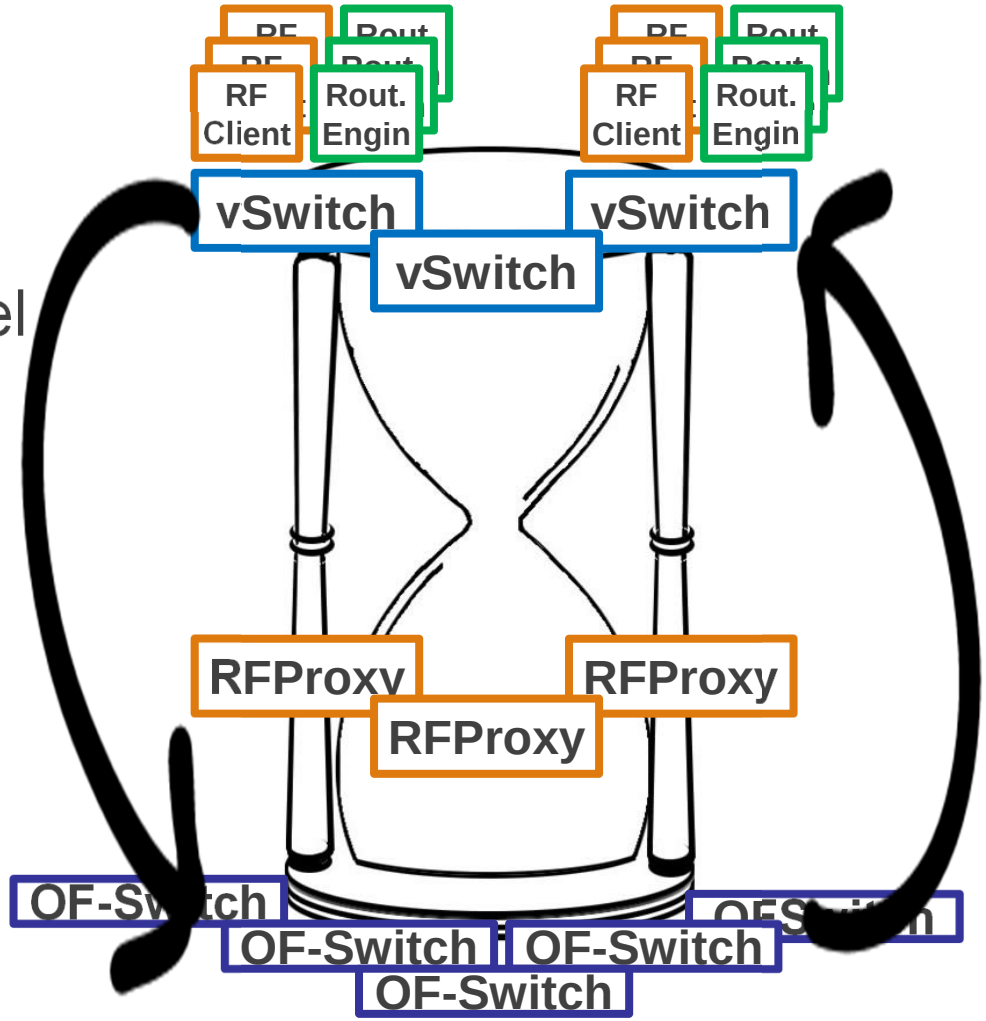
Architectural Discussions

Control-Data Channel

- OpenFlow-based
- OpenFlow-defined

Physical Distribution

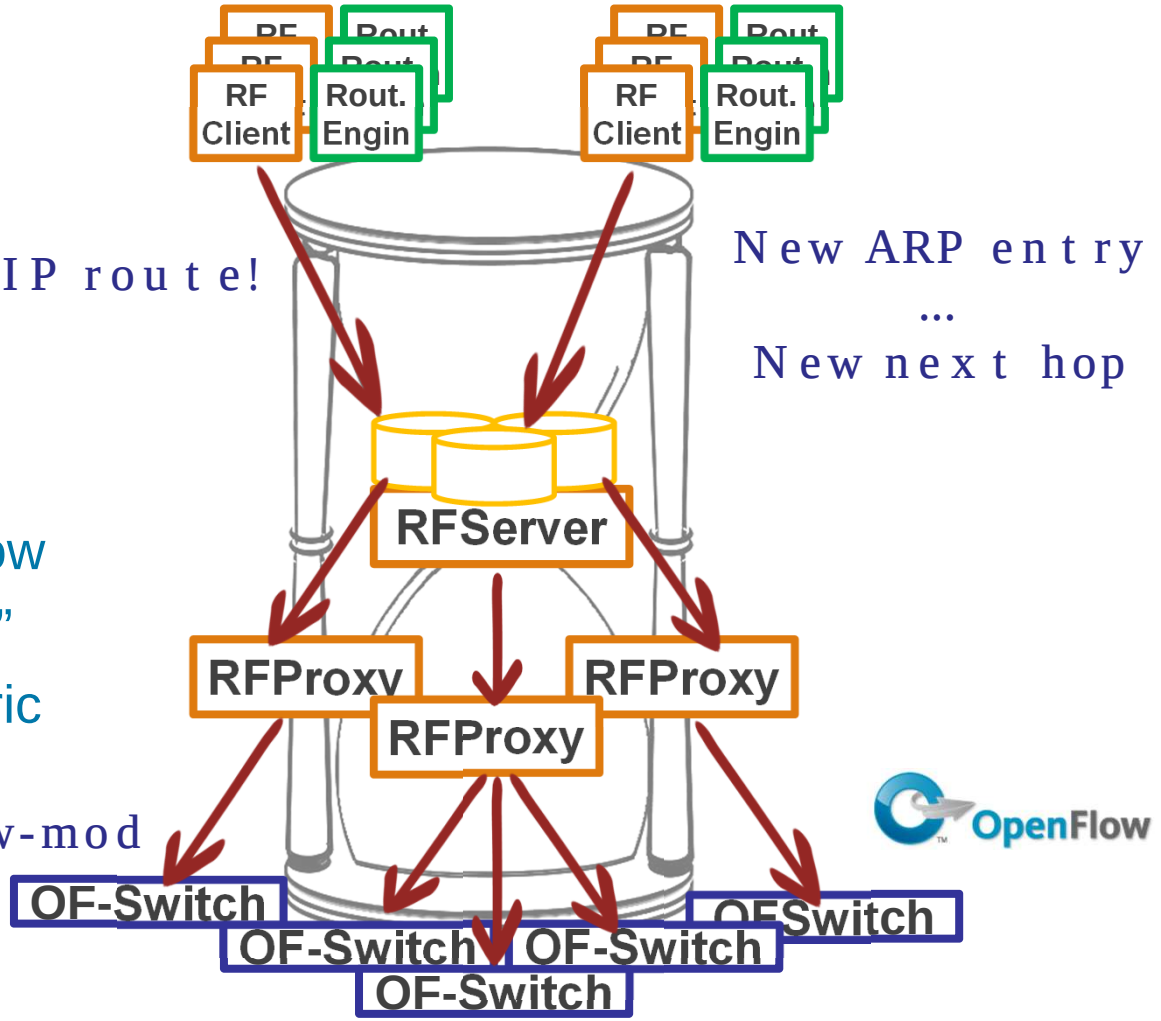
- Scalability
- Resiliency



Architectural Discussion

Centralized Logic

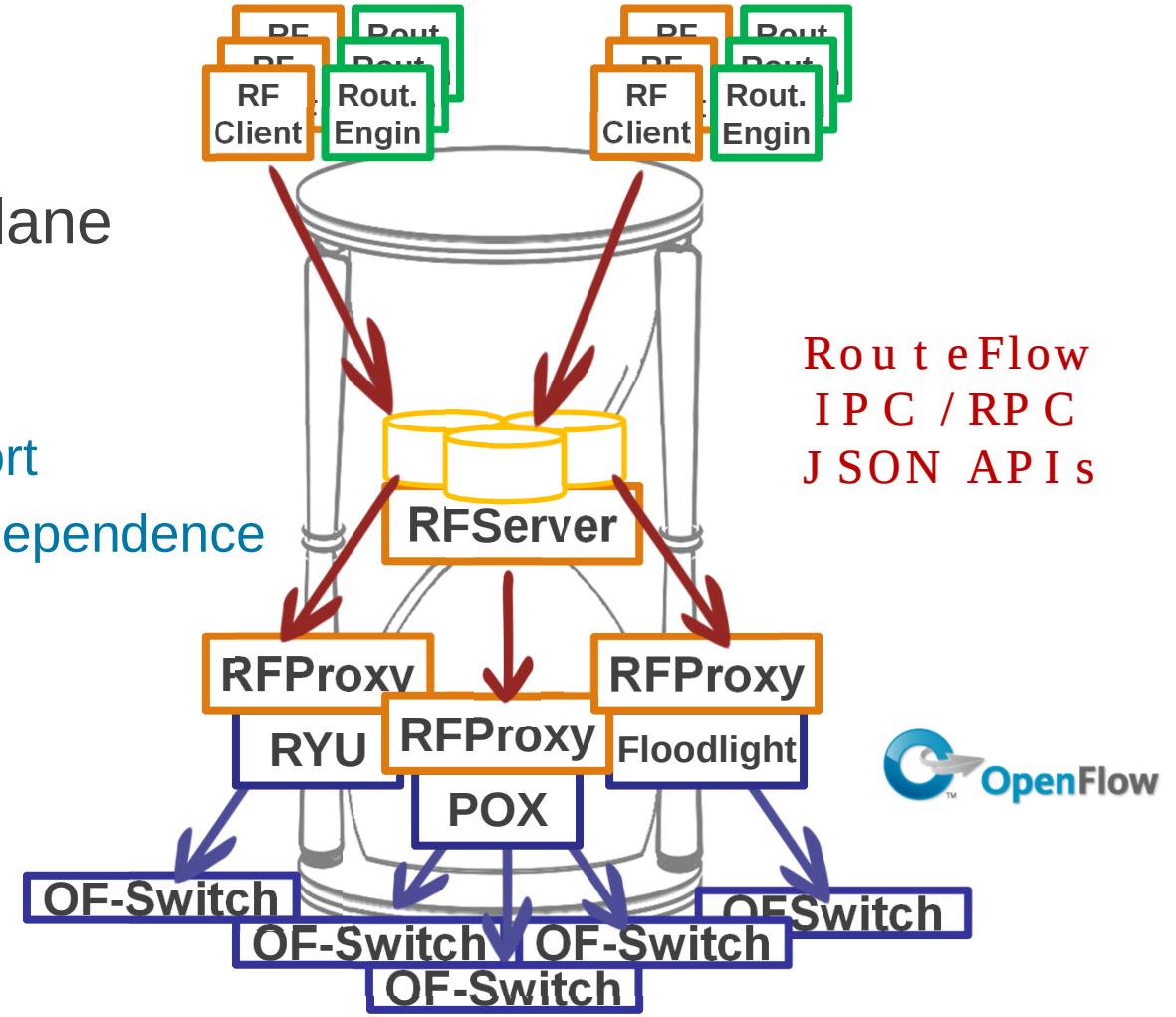
- CP/DP Mapping
- RIB-to-FIB-to-OpenFlow
- IP forwarding “policies”
- Intra-domain SDN fabric



Architectural Discussion

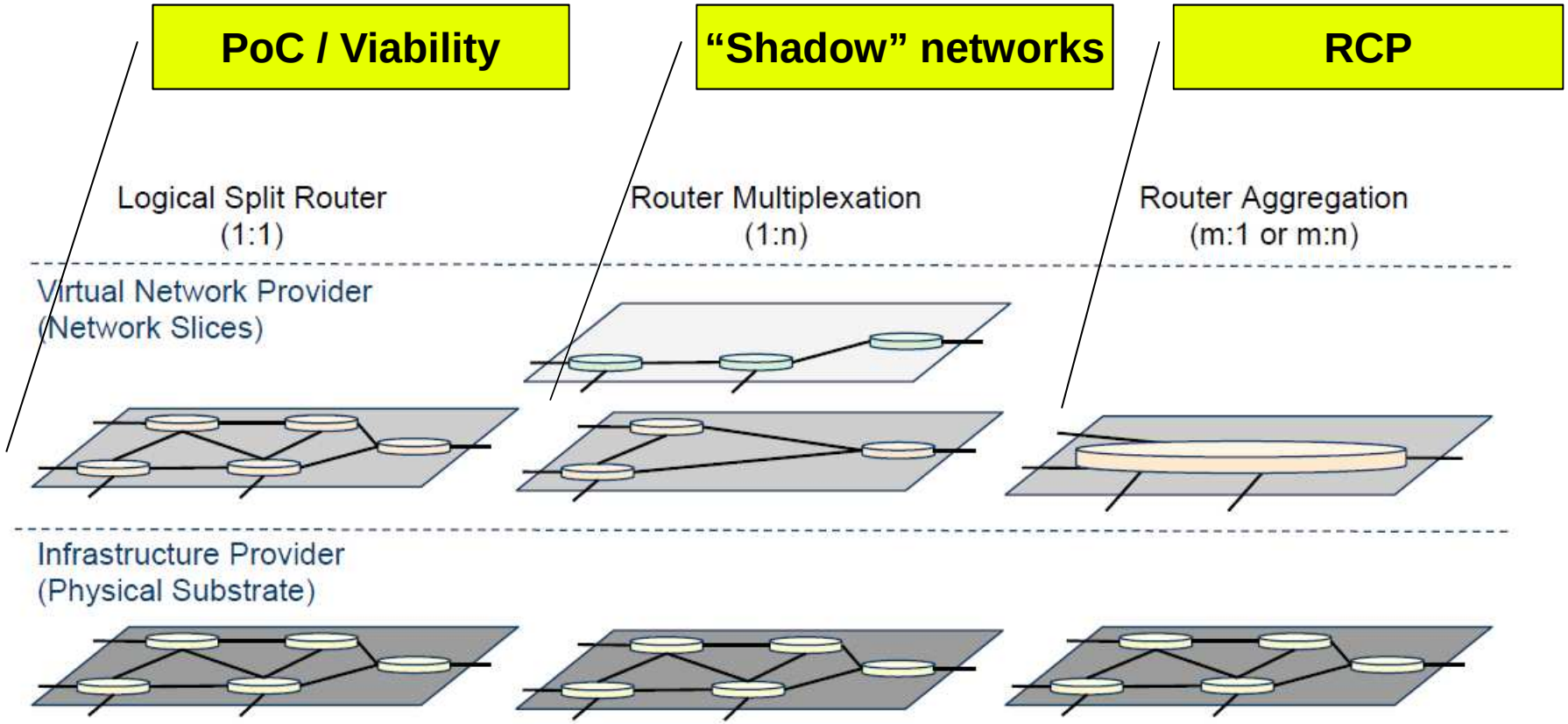
RouteFlow control plane

- Hierarchical
- Distributed
- Multi-Controller support
- OpenFlow-version independence



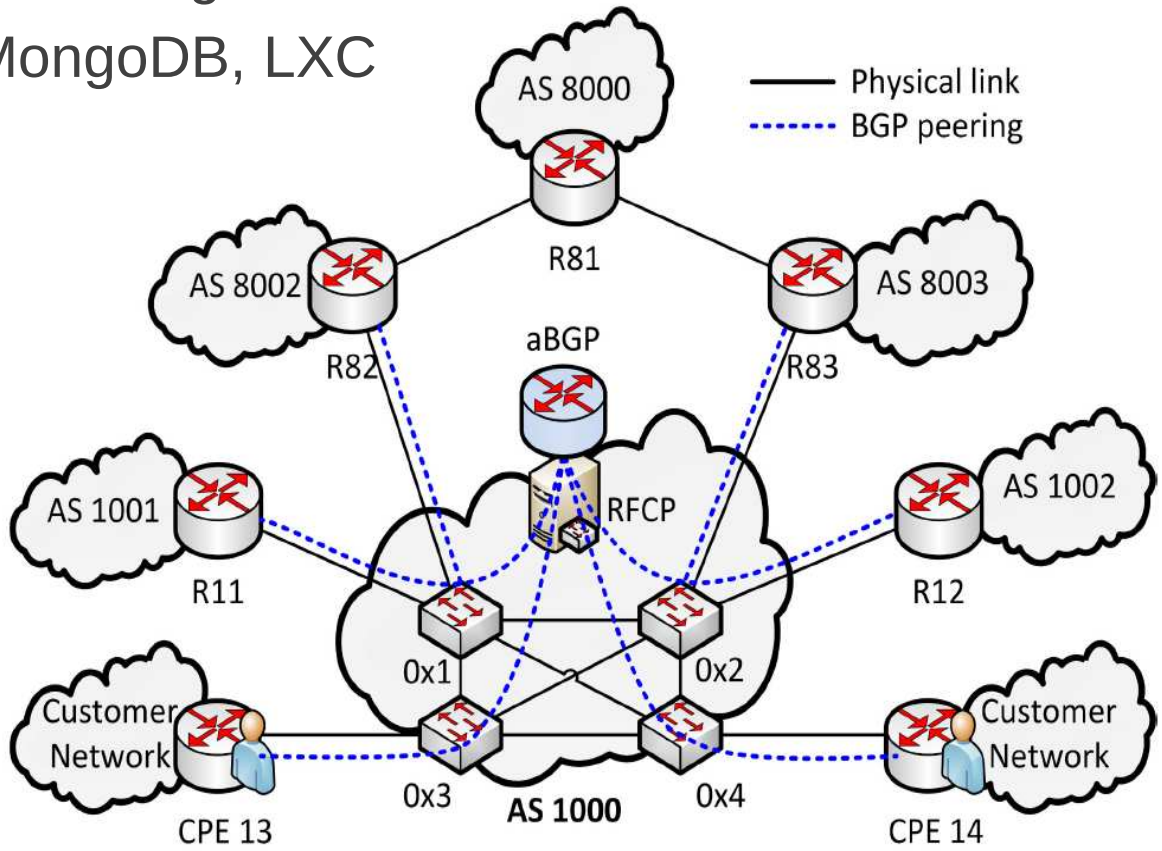
Modes of operation

- From logical routers (akin VRFs) to single node abstractions over flexible virtual networks.
- New design choices on the distribution of the control nodes.

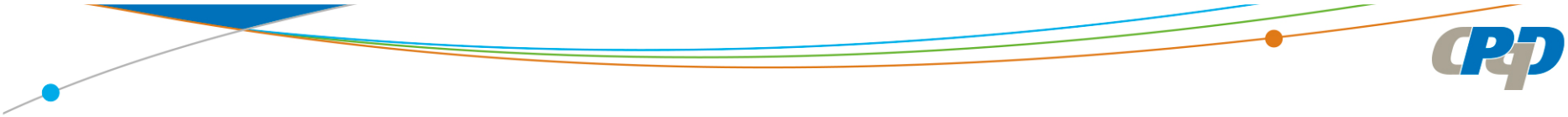


Aggregated BGP routing service

- Single node abstraction of a domain-wide eBGP router
  - Think modern multi-chasis routing architectures with external route processors and OpenFlow switches acting as line cards
- Aggregation logic defined in the RF-Server
- NOX, MongoDB, LXC



[HotSDN´12 Paper]



Research space

ROUTEFLOW

Research in scope and contribution

- Early work on Routing Control Platforms (RCP)
  [Ramjee 2006, Feamster 2004, Van der Merwe 2006, Wang 2009]
  - In operation at AT&T, considered a differentiator for "dynamic connectivity management".
- Research Question:
  - Re-examine the concept of RCP with the visibility
    (i.e., network-wide, multi-layer, flow and topology maps, full RIBs)
    and direct control capabilities
    (i.e., actual FIB installation, rich matching and instruction set)
    of the SDN abstraction set and the specifics of the OpenFlow choice

- RouteFlow glues virtualized IP routing stacks with OpenFlow
- RouteFlow acts as a new indirection layer for
  - routing protocol messages (e.g. BGP session terminates in servers)
  - RIB-(to-FIB)-to-OpenFlow transformations

Research Areas

SW (remote)

Application/ Solutions

High-level languages / policies, graph theory

[FML, Procera, NetCore]

Configuration Abstraction

Capture intent, not mechanism!

Controller / Network OS

Disitributed Systems, DB, P2P, High Availability

Network Abstraction

Hardware / Forwarding Abstraction

Flexibility and Programmability!

[e.g., POF (Protocol Oblivious Forwarding]

Devices

OpenFlow/APIs

HW/SW

ASIC / NP /x86

New HW/SW architectures (low-cost, energy-efficient,)
Flow matching algorithms

Simplified/Optimized for “flow” forwarding

Routing-centric use cases under research

- Engineered path selection
  - Think Google WAN, performance-based routing, etc.
- Optimal best path reflection
  - Per ingress/customer [draft-ietf-idr-bgp-optimal-route-reflection-01]
- Path protection with prefix independent convergence
  - Hierarchical FIBs w/ OF 1.X Tables + LFA route-precomputation
- Security
  - Data plane blackholes and middlebox injections,
  - Secure Inter-domain routing ideas (crypto intense S*-BGP, etc..)
- Simplifying customer multi-homing
  - Easy to set and control cost/performance/policy-based routing
- IPv6 migration
  - Flow matching for service termination in v4-v6 migration solutions

[RouteFlow HotSDN´12 paper]

Challenges

- Centralized BGP
  - Shown to scale well in modern CPU architectures
  - Centralized does not mean not disitrbuted (but removal from edge)
- Small OpenFlow table sizes
  - Transient limitation?
  - Expose existing FIB data structures as an IP lookup OF table?
  - Smart RIB&FIB reduction (e.g., simple [draft-ietf-grow-simple-va-04]
  - HW/SW flow offloading
- Limited OpenFlow processing in datapath
  - Transient / Un-optimized implementations
- High availability
  - Previous ideas from disitributed RCPs
  - Database-centric designs
  - Development in-progress of “BGP SHIM” for transparent eBGP redundancy

Some recent interesting research topics

SDX: A Software Defined Internet Exchange

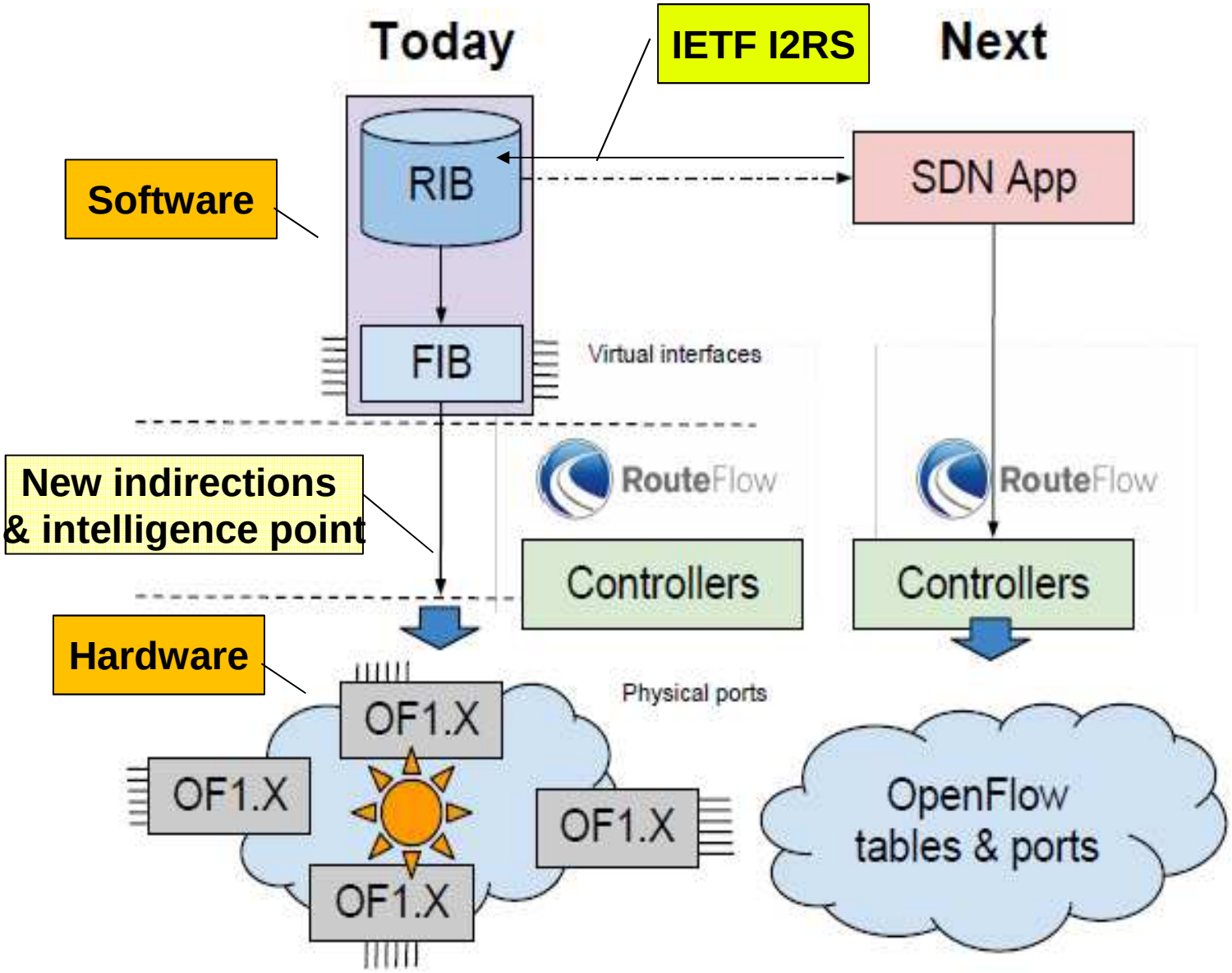
- Nick Feamster, Jennifer Rexford, Scott Shenker, Dave Levin, Russ Clark, Ron Hutchins, Josh Bailey
- http://www.opennetsummit.org/pdf/2013/research_track/poster_papers/final/ons2013-paper44.pdf
- http://www.ietf.org/proceedings/86/slides/slides-86-sdnrg-6

Leveraging SDN Layering to Systematically Troubleshoot Networks

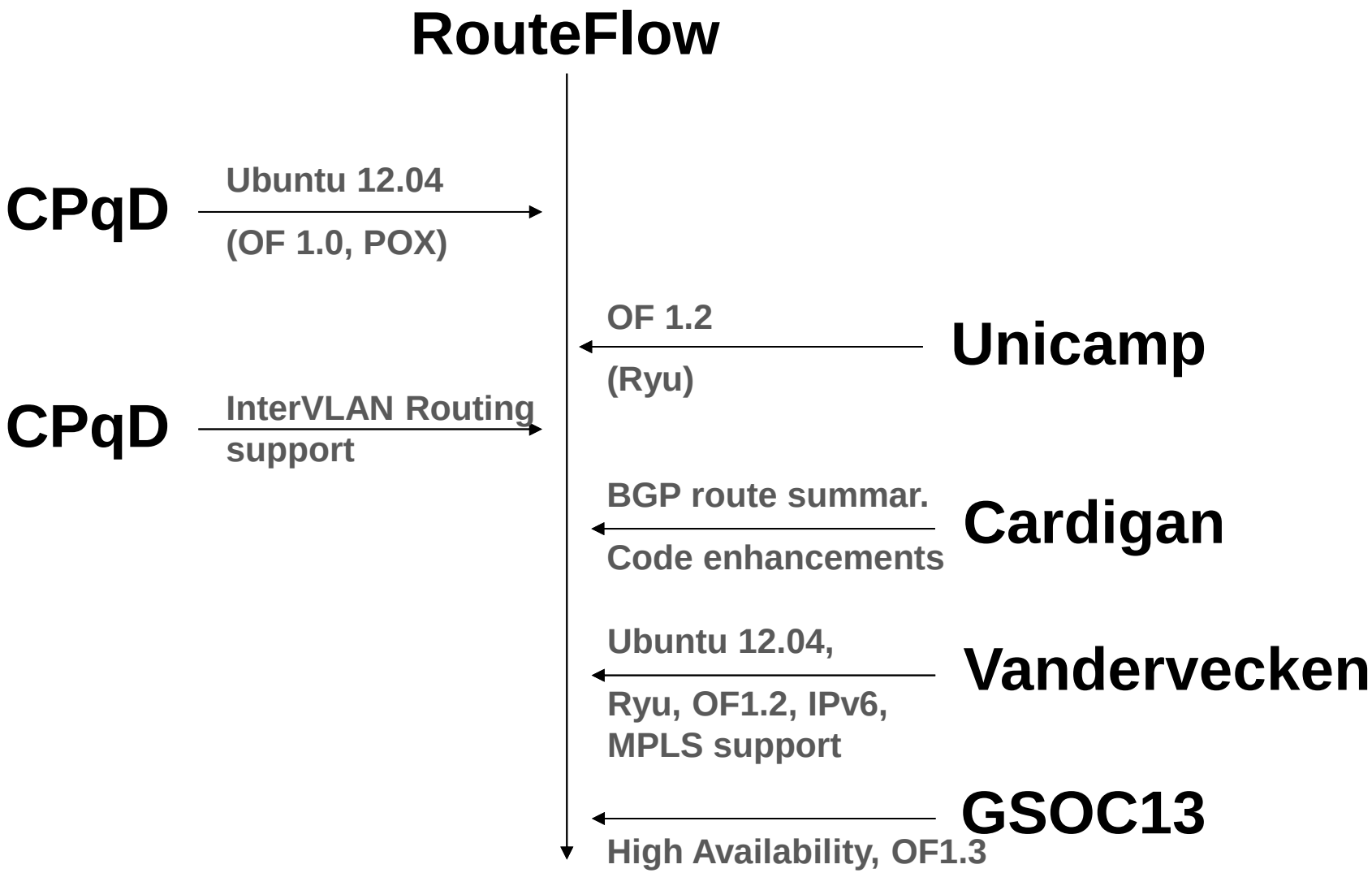
- Brandon Heller, Colin Scott, Nick McKeown, Scott Shenker, Andreas Wundsam, Hongyi Zeng, Sam Whitlock, Vimalkumar Jeyakumar, Nikhil Handigol, Murphy McCauley, Kyriakos Zarifis and Peyman Kazemian.
- HotSDN13

Advancements in latest OpenFlow protocol versions

- High availability, QoS, IPv6, multi-path, load-balancing, fast re-route, etc.


Today
IETF I2RS
Next
Software
RIB
SDN App
FIB
Virtual interfaces
RouteFlow
RouteFlow
New indirections
& intelligence point
Controllers
Controllers
Hardware
Physical ports
OF1.X
OF1.X
OF1.X
OF1.X
OpenFlow
tables & ports

Latest updates



Google Summer of Code 2013

- Home
- My Profile
- My Dashboard
- About
- Events & Timeline
- Connect With Us
- Search
- FAQ
- Logout

RouteFlow

Edit | Start Connection | Transfer slots to pool

DESCRIPTION

RouteFlow is an open source project that aims to provide virtualized IP routing services over OpenFlow enabled hardware. Licensed under Apache 2.0, worldwide pioneer RouteFlow has 1000+ downloads and is being used by institutions such as Google Research, NTT MCL, Indiana University, Internet2 NDDI and many others across the globe.

Ideas page » Homepage »

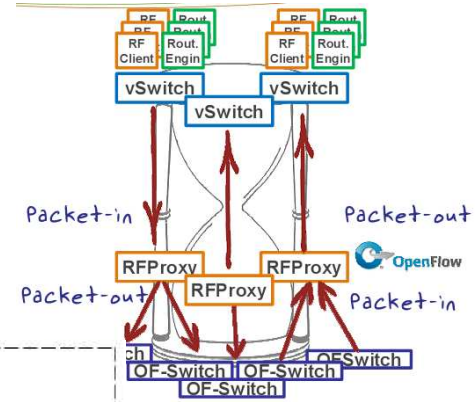
TAGS

OpenFlow, Software-Defined Networks, IP Routing, Quagga, Linux, Python, C++, Java

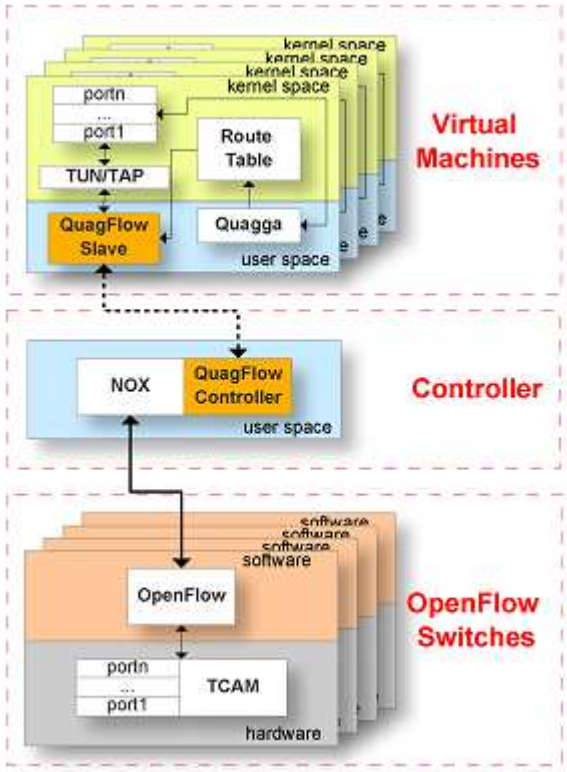
CONTACT

RouteFlow control plane architecture

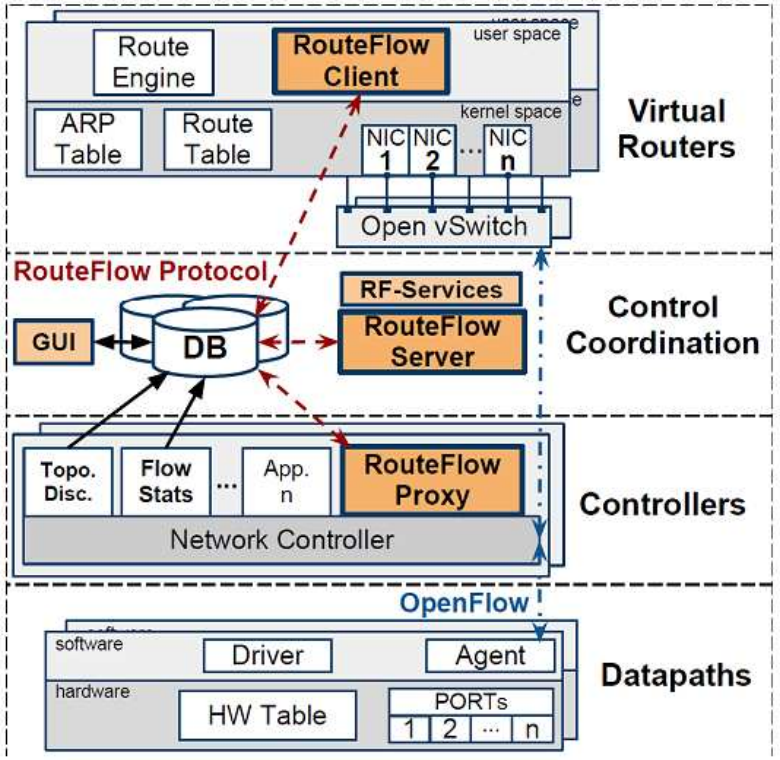
From QuagFlow to current implementation:



2 010 *Dev(Ops)* 2 013



[SIGCOMM´10 Poster]



[HotSDN´12 Paper]

Gaining practical experience through use cases, pilots, demos, etc.

ROUTEFLOW

Field Trial at the University of Indiana

Network setup

1 physical OpenFlow switch
- Pronto 3290

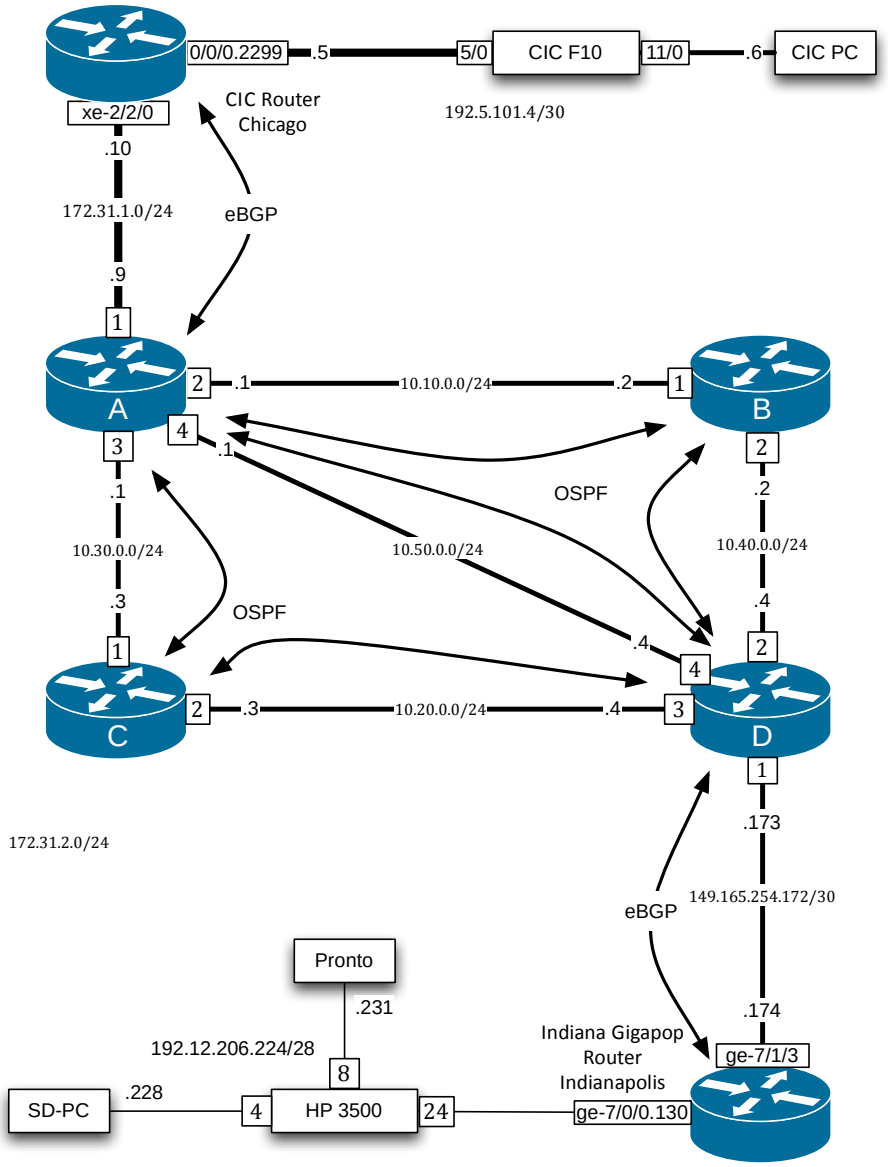
4 Virtual routers out of the physical OpenFlow switch

10 Gig and 1 Gig connections

2 BGP connections to external networks
- Juniper routers in Chicago and Indianapolis

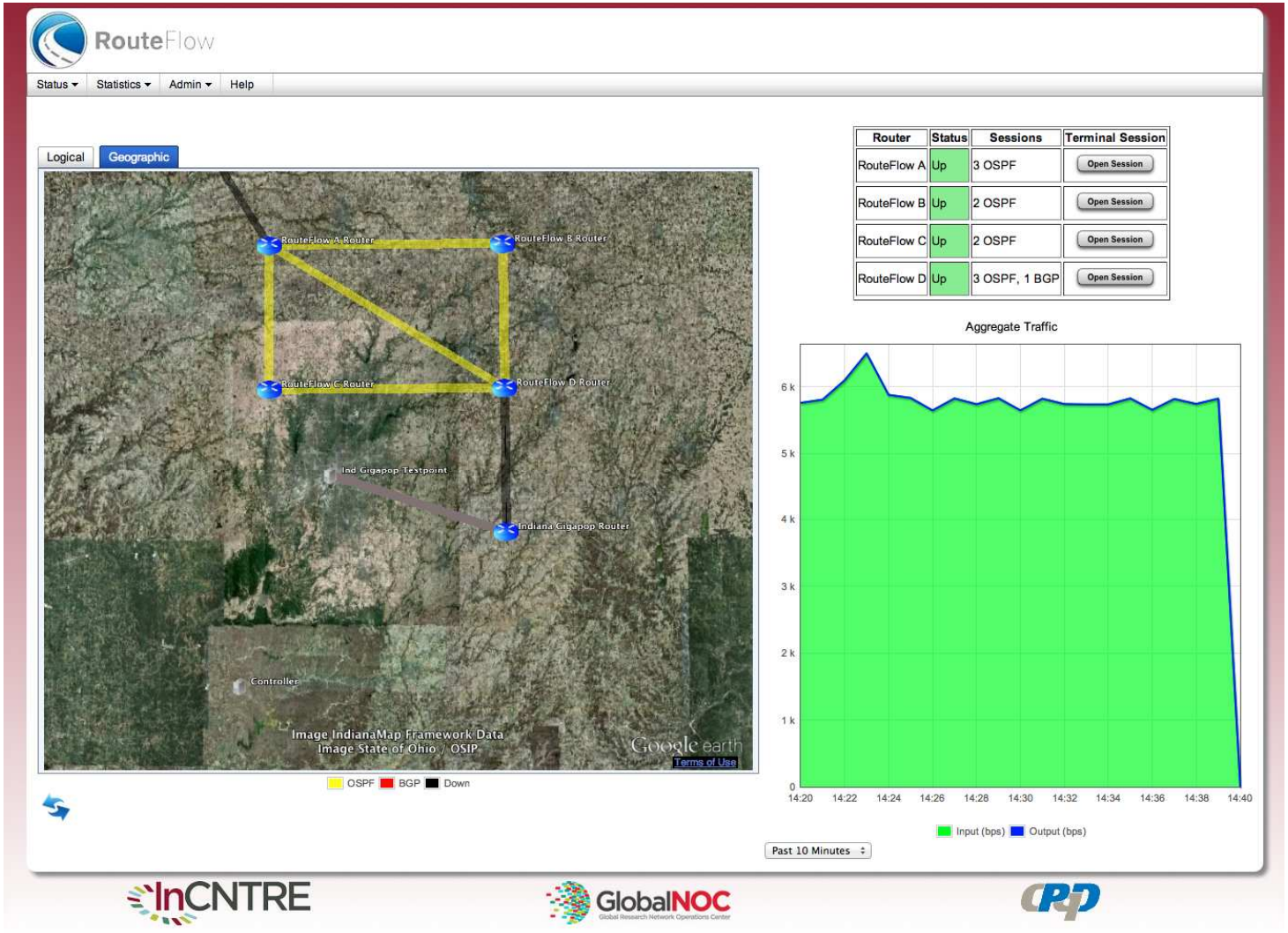
Remote Controller

New User Interface

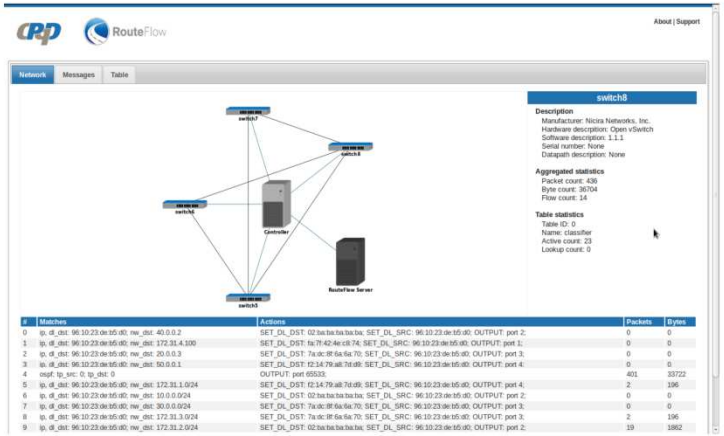


Field Trial at the University of Indiana

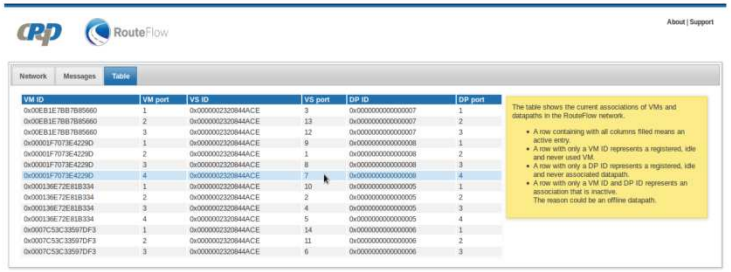
User Interface



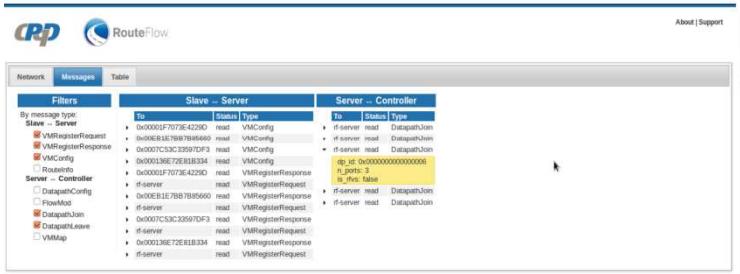
Advancing the RouteFlow GUI



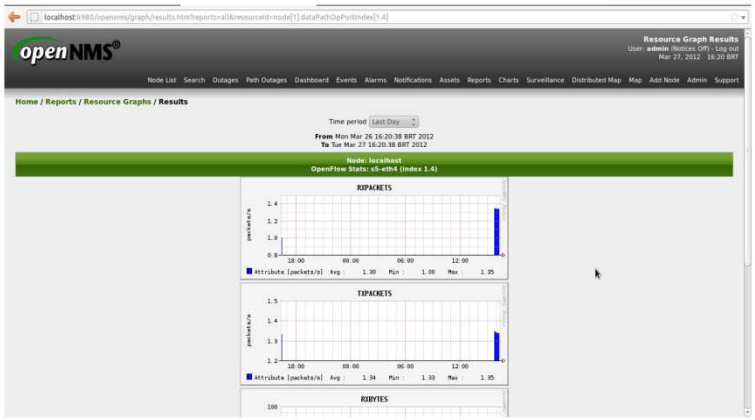
Topology and Statistics



Resource Status and Mapping



RouteFlow Protocol



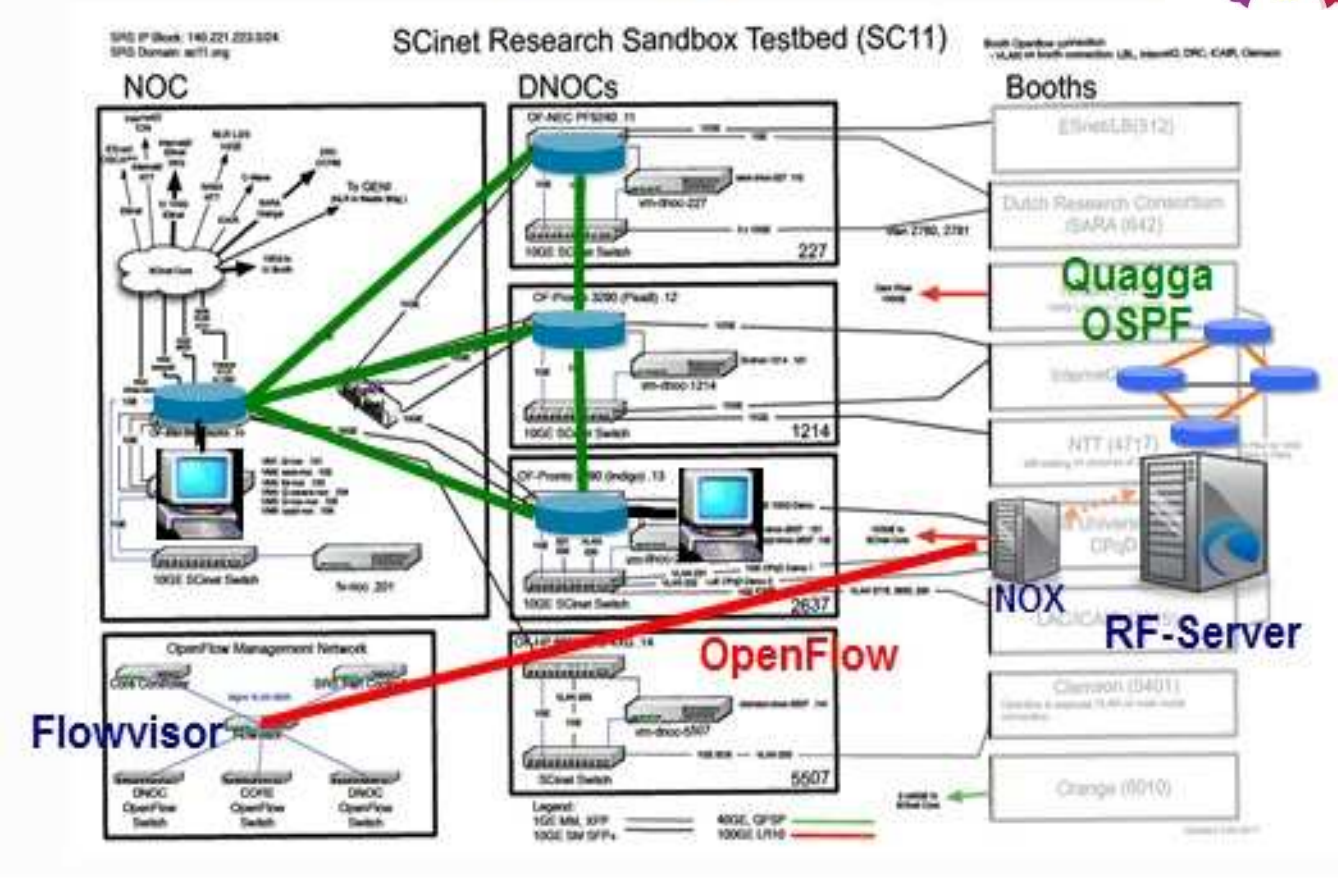
OpenNMS SNMP

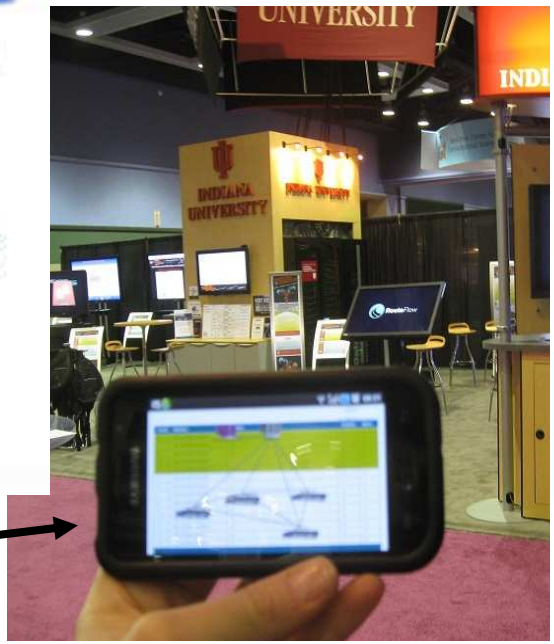
Demonstration at Supercomputing 11

RouteFlow

SRS Demo @ SC11

SC11 Seattle, WA Connecting communities through HPC

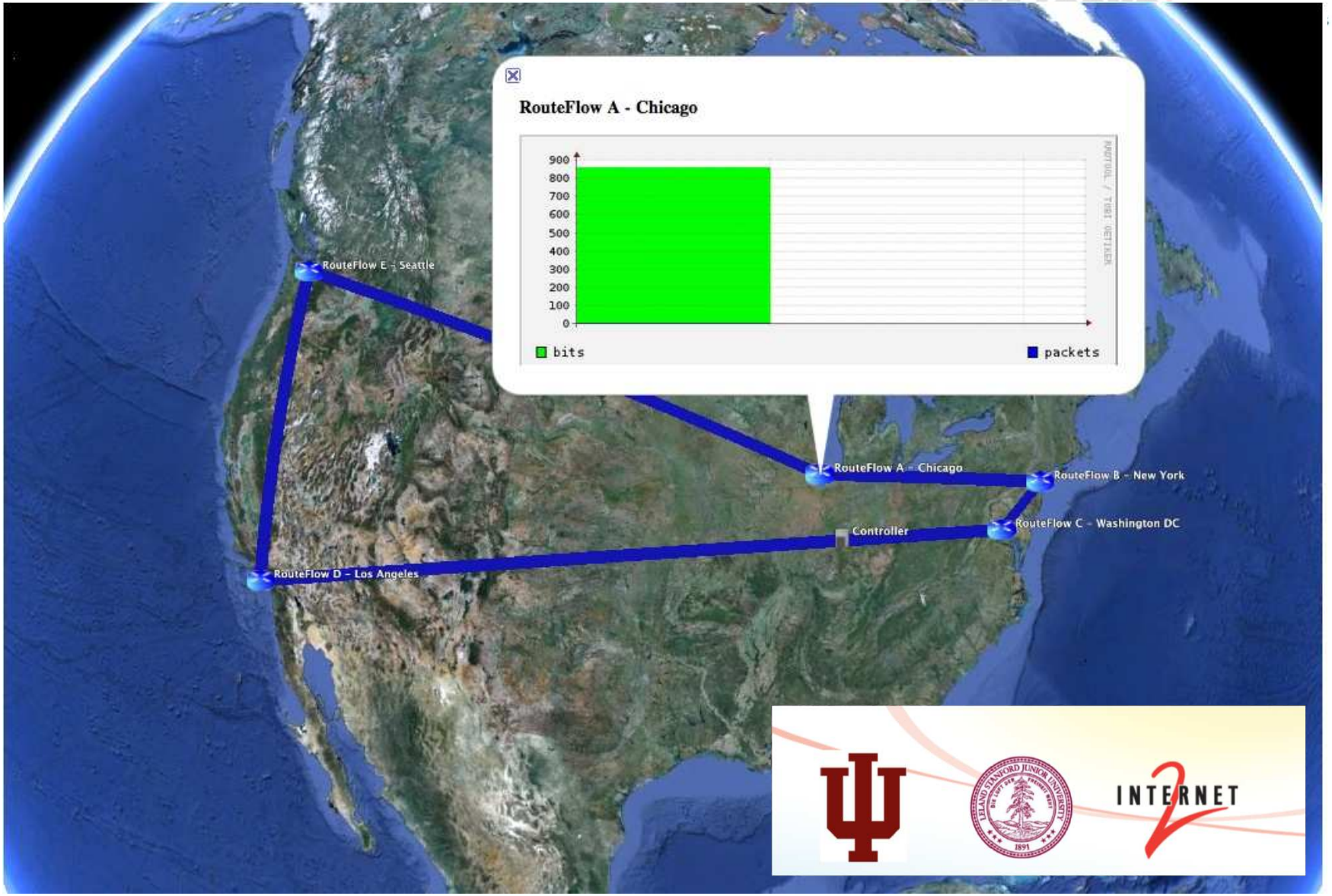




Routing configuration at your fingertips

RouteFlow NDDI Deployment

[Joint Techs]



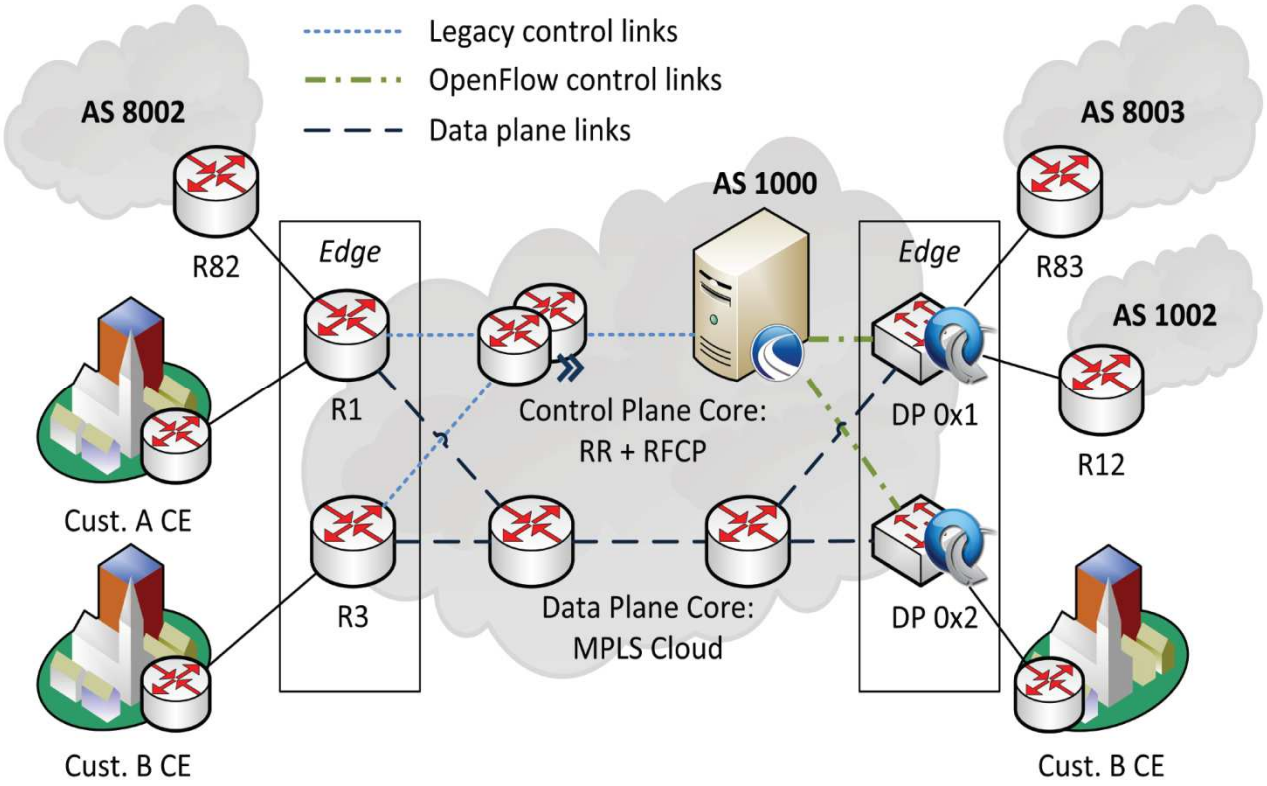
European SDN Congress

Controller-Centric Hybrid Networking

A migration path to roll out OpenFlow technology

Not a revolution, but an evolution of current iBGP RRs to essentially eBGP Route Controllers

- "BGP-free edge": A cost-effective simplified edge for SW-driven innovations



III Open Networking Summit (Santa Clara)

DEMO 1 — IP-Routed Network on Demand

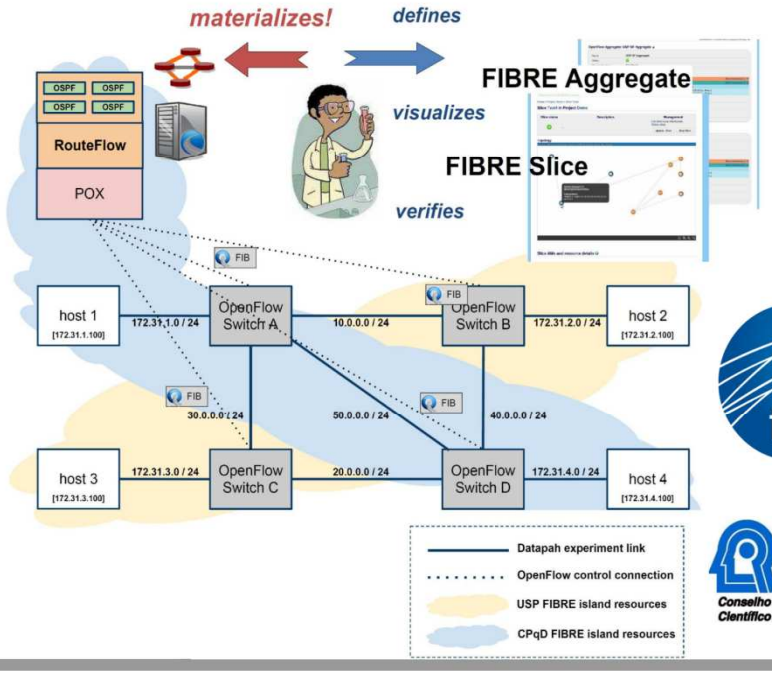
Highlights

Federation of FIBRE islands
- 4 OpenFlow switches (NetFPGAs) and 2 XEN Agents

Runs RouteFlow experiment on requested slice
- IP routing (OSPF) on sliced topology

Islands connected through VLANs in existing infrastructure (GIGA network)

Federation through FIBRE CMF provides unified cross-island experimental setup



Partners

Universidade de São Paulo

fibre — FUTURE INTERNET TESTBEDS EXPERIMENTATION BETWEEN BRAZIL AND EUROPE

CNPq — Conselho Nacional de Desenvolvimento Científico e Tecnológico

COOPERATION

III Open Networking Summit (Santa Clara)

DEMO 2 — InterVLAN Routing

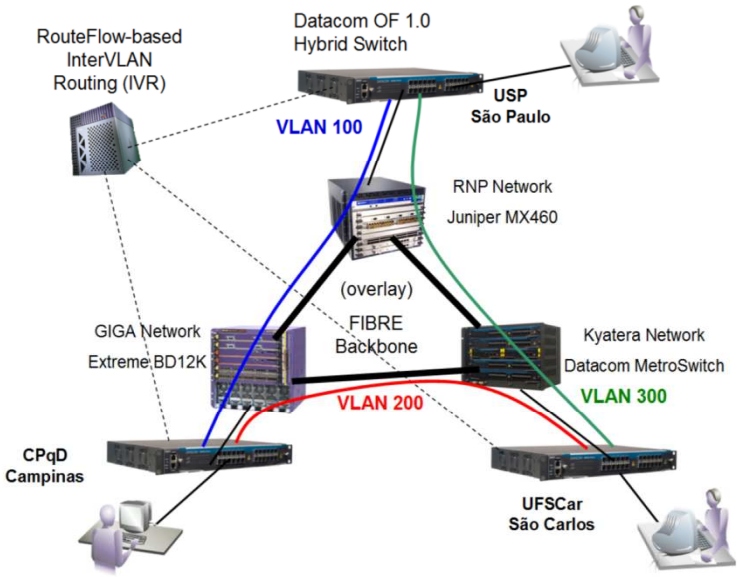
Highlights

Goal: Interconnect 32 campi

RFServer defines InterVLAN routing logic
- Router-on-a-stick paradigm

Seamless VLAN configuration

OpenFlow rules match on destination IP and perform VLAN rewrite actions



Partners

III Open Networking Summit (Santa Clara)

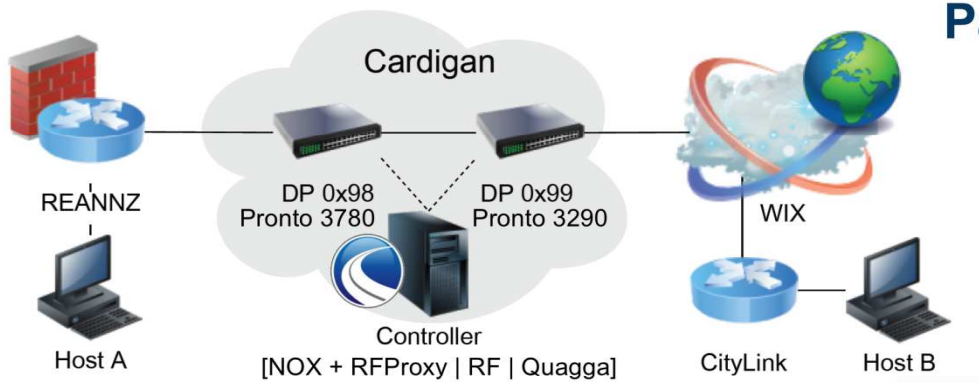
DEMO 3 — Distributed IX Router

Highlights

Deploying a distributed routing fabric
Production traffic in live IXP

Reduced operational complexity

Easier to understand
Aids modification and diagnosis



Partners

Snapshot: 1134 flow entries in each switch

8 flows matching control plane traffic
(e.g., ARP, ICMP, BGP, etc.)

1 flow entry to drop traffic by default

98 flows describing BGP speakers

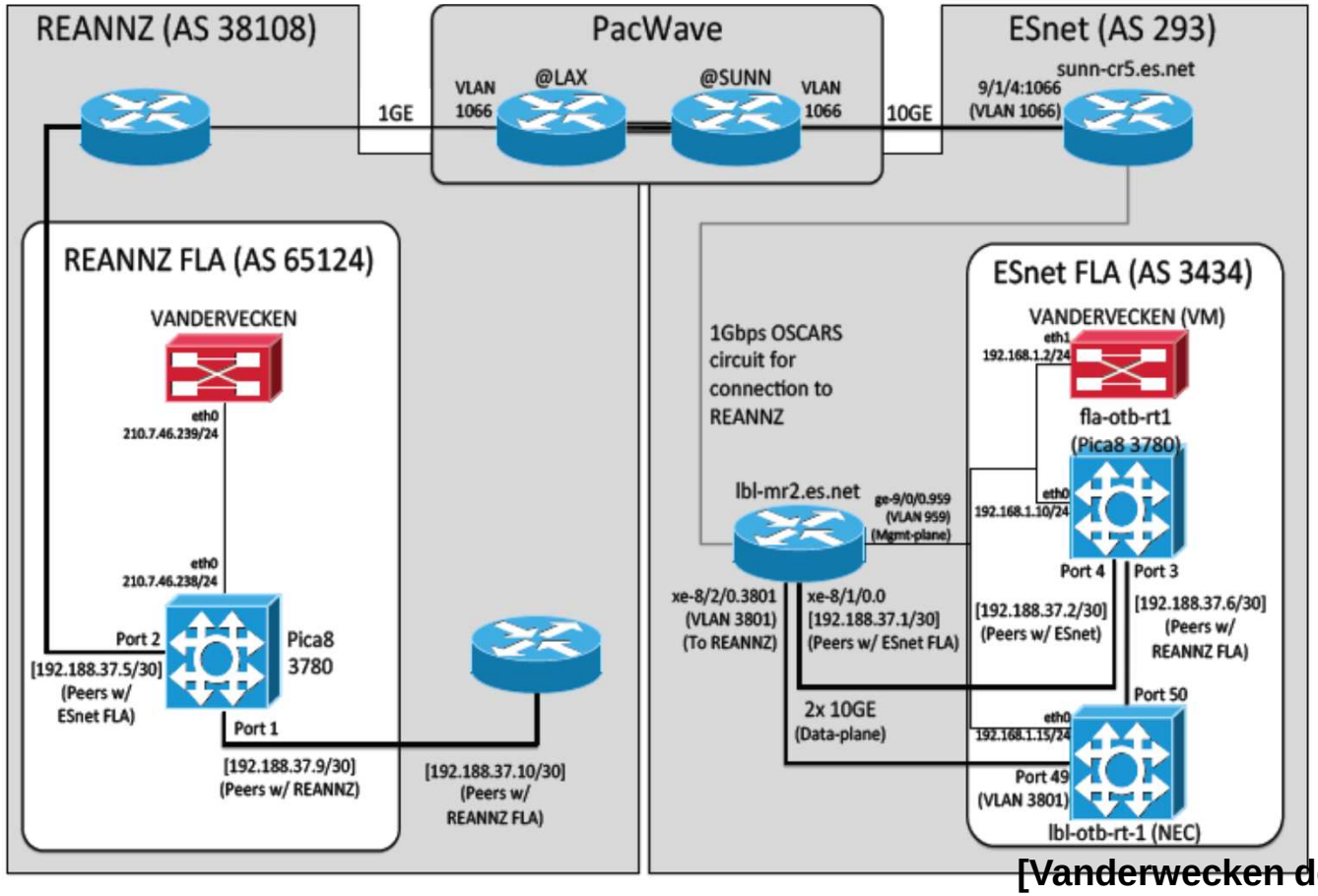
1028 flows representing L3 routes

[RouteFlow/Cardigan HotSDN´13 poster]

International BPG peering using RouteFlow

- Innovative FIB compression enables using commodity OpenFlow switches for peering
- Leverages community open-source packages. RouteFlow and Quagga

Next: Internet2, USLHCnet, UvA and RNP



[Vanderwecken demo @ ONS´13]

… building a community

Visits: 35,000+ (17,000+ Unique)

From over 2600 cities of 130+ countries all over the globe!

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

RouteFlow

772

days since Project Launch

GIGA
REDE EXPERIMENTAL DE ALTA VELOCIDADE

FUNTTEL FINEP

1000s downloads!

github SOCIAL CODING

Apache

CONCLUDING REMARKS

SDN asks (at least) three major questions

Where the control plane resides "Distributed vs Centralized" ?

How does the Control Plane talk to the Data Plane ?

How are Control and Data Planes programmed ?

Source: T. Nadeu, slides-85-sdnrg-5.pptx

SDN asks (at least) three major questions

Where the control plane resides
"Distributed vs Centralized" ?

- **What state belongs in distributed protocols?**
- **What state must stay local to switches?**
- **What state should be centralized?**

- **What are the effects of each on:**
  - state synchronization overhead
  - total control plane overhead
  - system stability and resiliency
  - efficiency in resource use
  - control loop tightness

Source: E. Crabbe, slides-85-sdnrg-7.pdf

SDN: a Fundamental Step Forward

- or just a new whip to beat vendors with?

What makes SDN attractive?

The idea that a network is more than the sum of its parts

- I.e., take a network-wide view rather than a box-centric view

The idea that creating network services can be a *science* rather than a set of *hacks on hacks on hacks*

- Especially hacks that vary by box, by vendor and by OS version

The idea that there should be a *discipline* and *methodology* to service *correctness*

- Rather than testing (and more testing), declaring victory, only to fail in the real world because of some unanticipated interaction

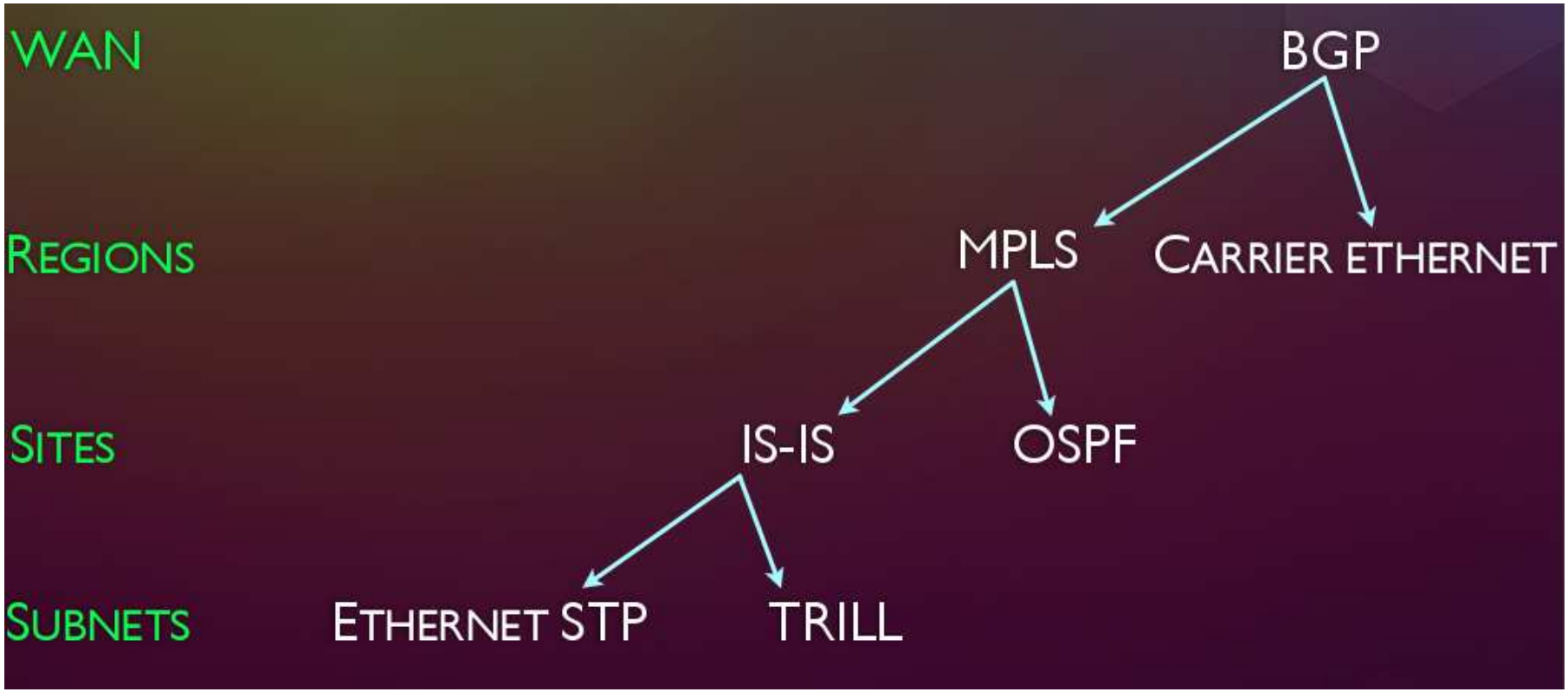
Source: K. Kompella, slides-85-sdnrg-2.pdf

SDN is a real step

- 1. IF SDN gives us an *abstraction* of the network
- 2. IF, through this abstraction, we have a means of *reasoning* about the network and network services
- 3. IF SDN offers a means of *verifying* correct operation of the network or of a service
- 4. IF SDN offers a means of *predicting* service interaction
- 5. Finally, IF SDN offers a means of setting (conceptual) asserts by which we can get *early warning* that something is wrong

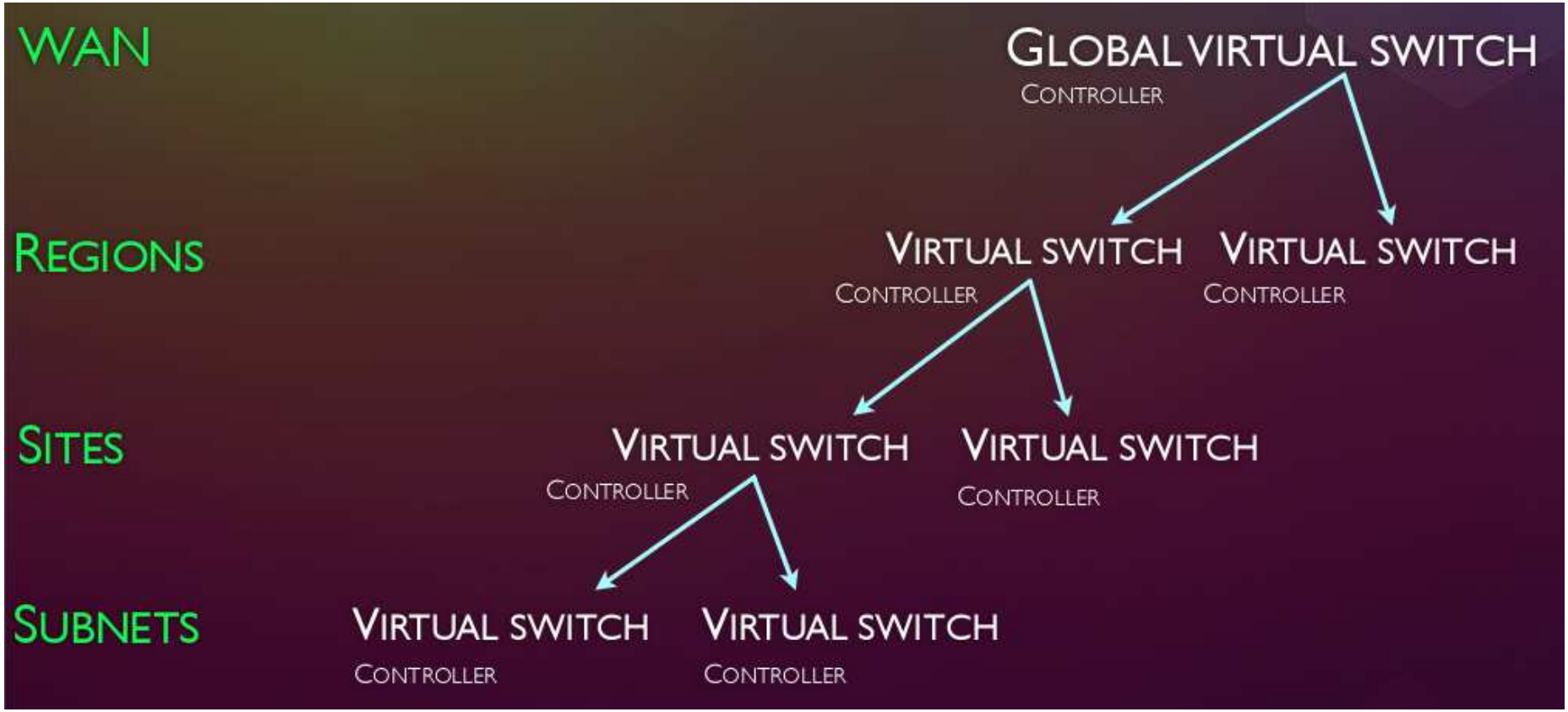
Source: K. Kompella, slides-85-sdnrg-2.pdf

Classic Layered Control Planes



Source: T. Koponen

Layered Control Planes in SDN



Further reading:
"Extending SDN to Large-Scale Networks"
by Aurojit Panda (UC Berkeley), Scott Shenker and Murphy McCauley (UC Berkeley and ICSI), and Teemu Koponen and Martin Casado (VMware)

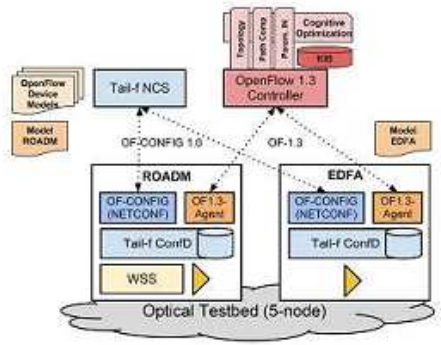
Source: T. Koponen

Change of Design Priorities

Source: T. Koponen

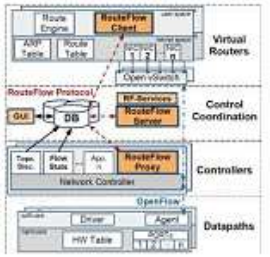
Ongoing SDN activities at CPqD

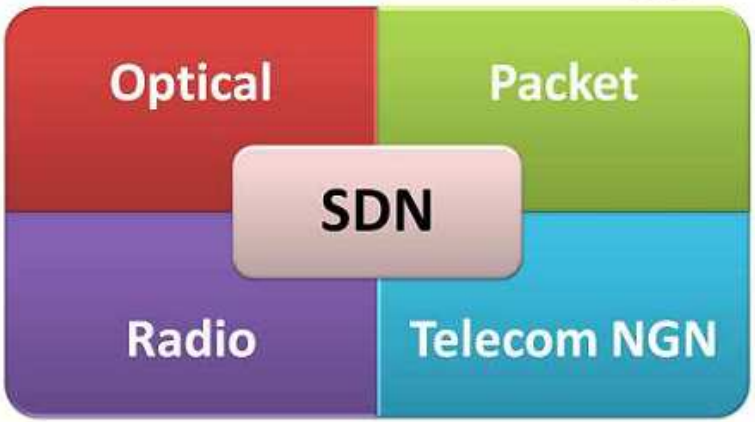
Software-Defined Optical Transport

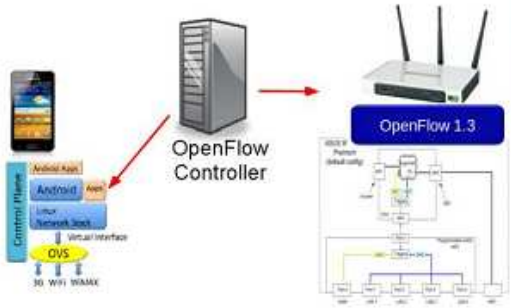


Software-Defined IP Routing

RouteFlow







openstack™

OpenFlow 1.x prototyping

Software-Defined Wireless Networking

Cloud & Software-Defined Telecom Services

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)

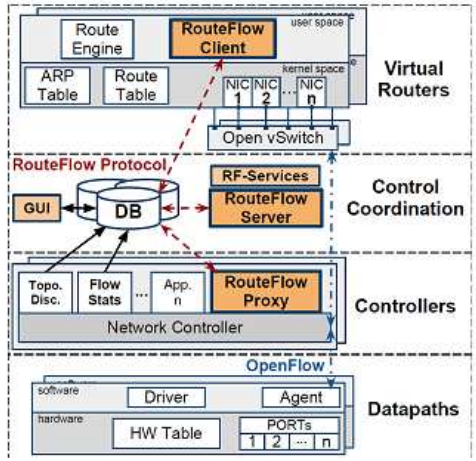
Hierarchical, distributed

Multi-controller support

(POX, NOX, Floodlight, Ryu)

Any Linux-based routing stack

(Quagga, XORP, BIRD)



Obrigado!

*www.**cpqd**.com.br*

References

IRTF Software Defined Networking Research Group (SDNRG)

http://trac.tools.ietf.org/group/irtf/trac/wiki/sdnrg

Bob Briscoe, IRTF SDNRG meeting,
http://www.ietf.org/proceedings/84/slides/slides-84-sdnrg-0.pdf

Srini Seetharaman, A Taxonomy For SDN Solutions,

http://www.sdncentral.com/products-technologies/what-is-not-sdn/2013/05/

Chris Grundemann, Litghtining talk at NANOG 56

Ivan Pepelnjak (ip@ioshints.info),

https://ripe65.ripe.net/presentations/19-OpenFlow_and_SDN_(RIPE).pdf

D. Mayer, Recent talks

http://www.1-4-5.net/~dmm/vita.html

T. Koponen, Structure and Design of Software-Defined Networks

http://netseminar.stanford.edu/03_14_13.html

Peyman Kazemian, Troubleshooting SDNs

Thomas Nadeau, What are the hard (and interesting) open research problems in SDN?

http://www.ietf.org/proceedings/85/slides/slides-85-sdnrg-5.pptx