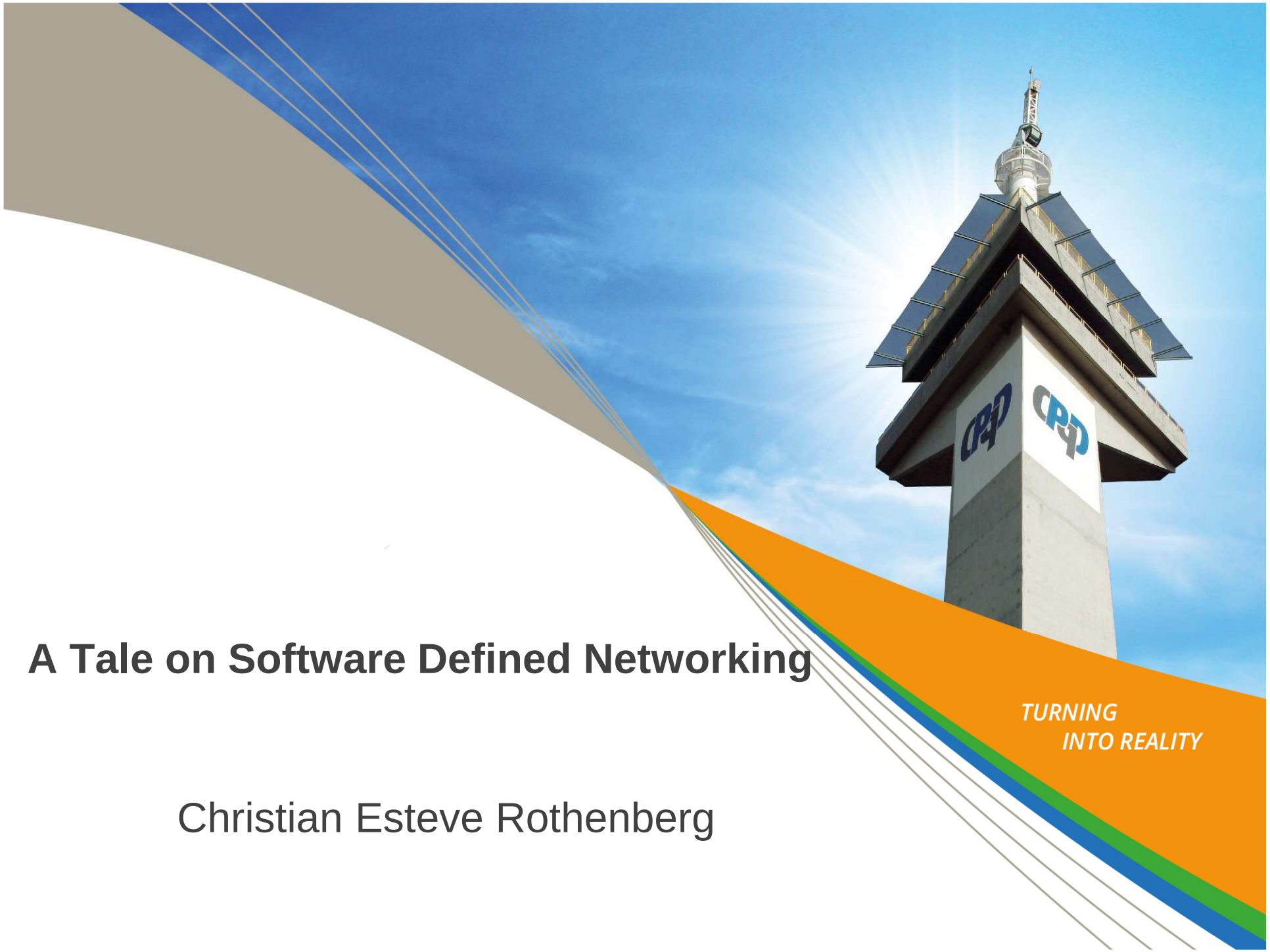


A Tale on Software Defined Networking

Christian Esteve Rothenberg

Disclaimer

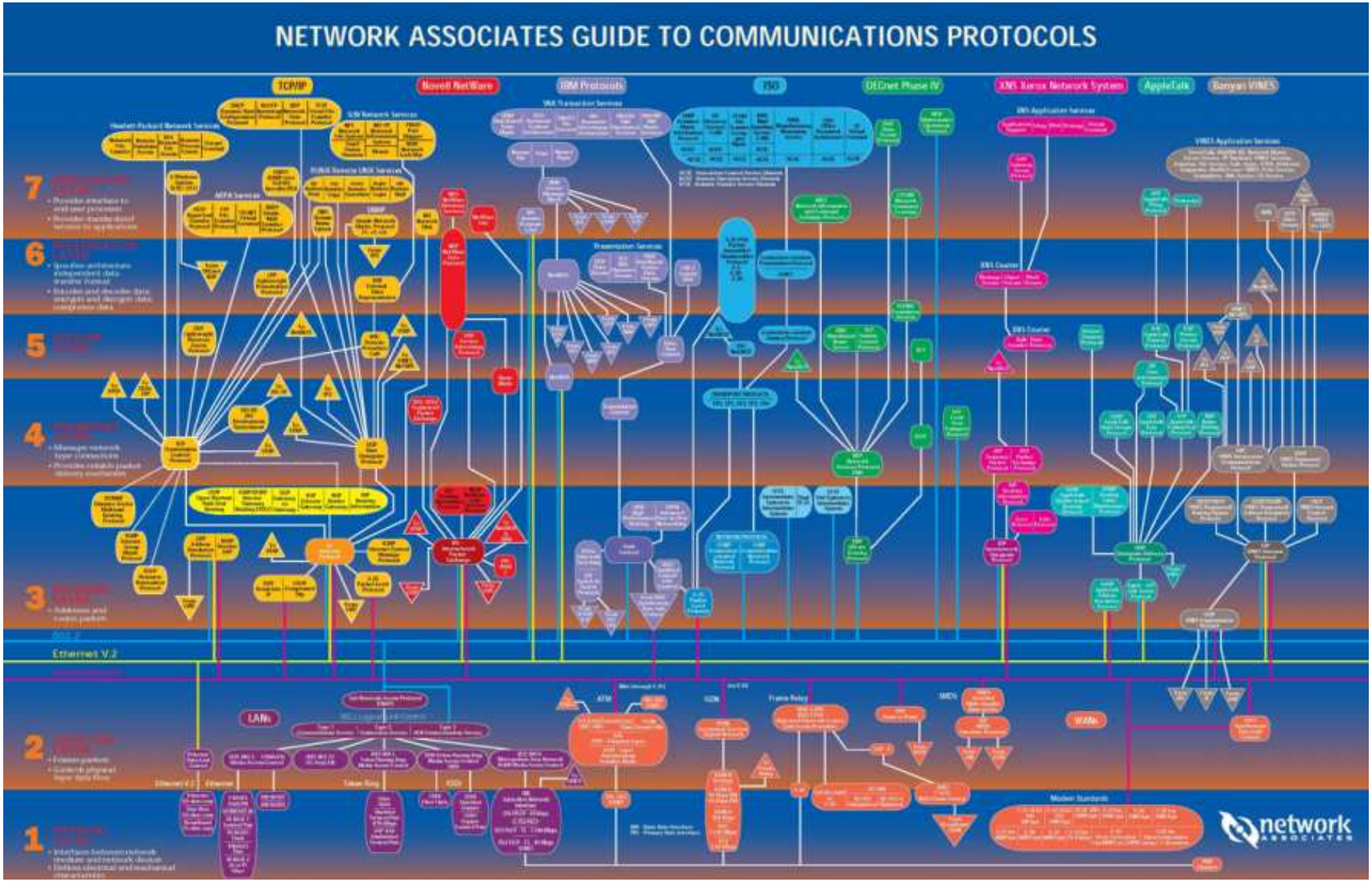
Personal view on SDN

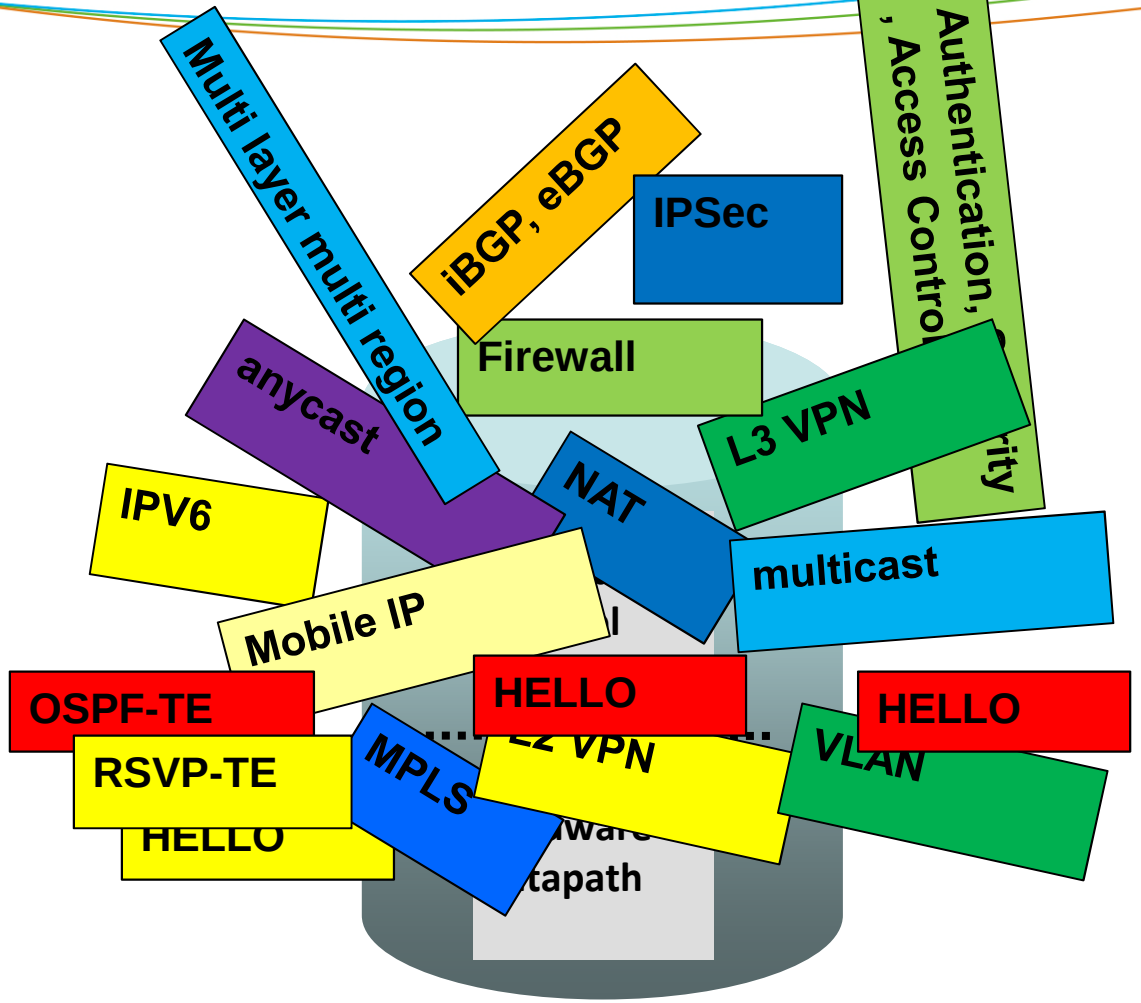
- Not by CPqD
- Not by ONF (or any affiliated company)

Actually, not that personal and original...

- See References and Credits

Where are we today?





Many complex functions baked into the infrastructure

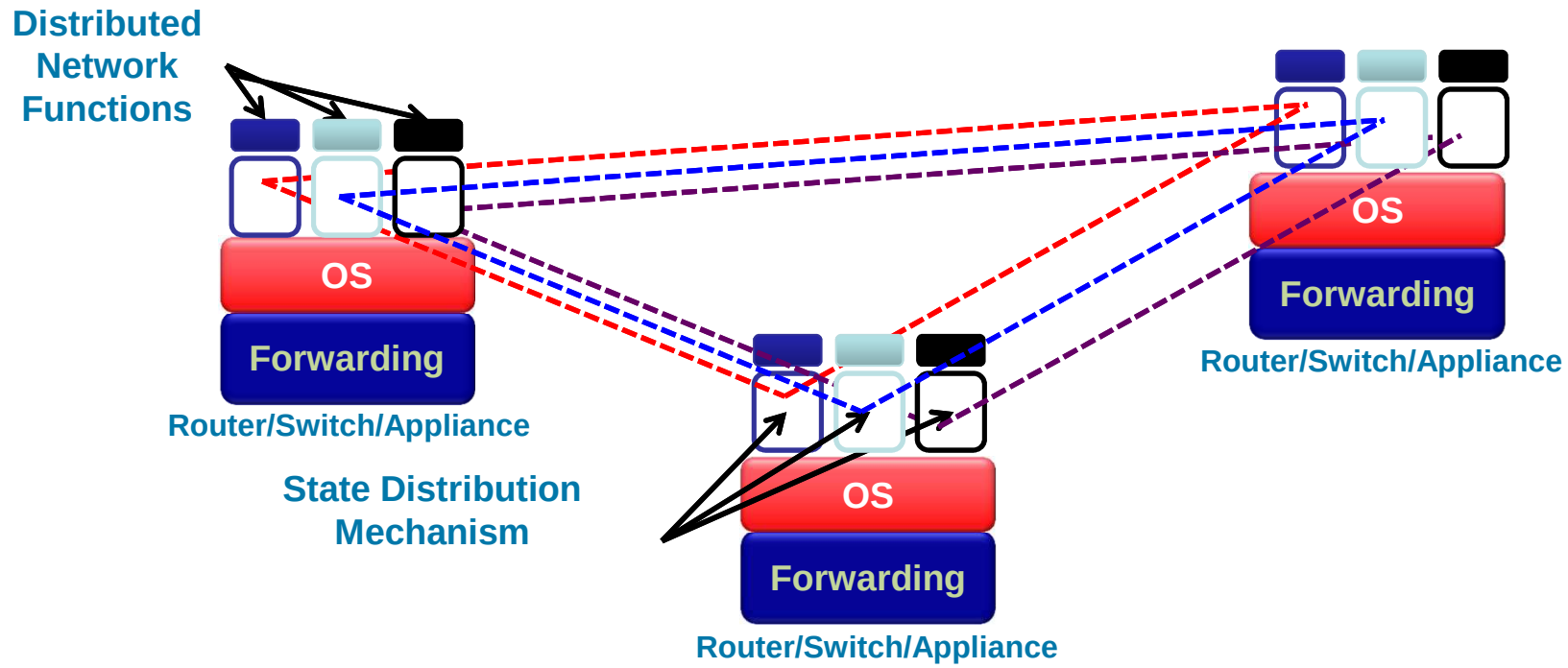
OSPF, BGP, multicast, differentiated services,
Traffic Engineering, NAT, firewalls, MPLS, redundant layers, …

An industry with a “mainframe-mentality”

Source: Stanford/Berkeley

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

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**

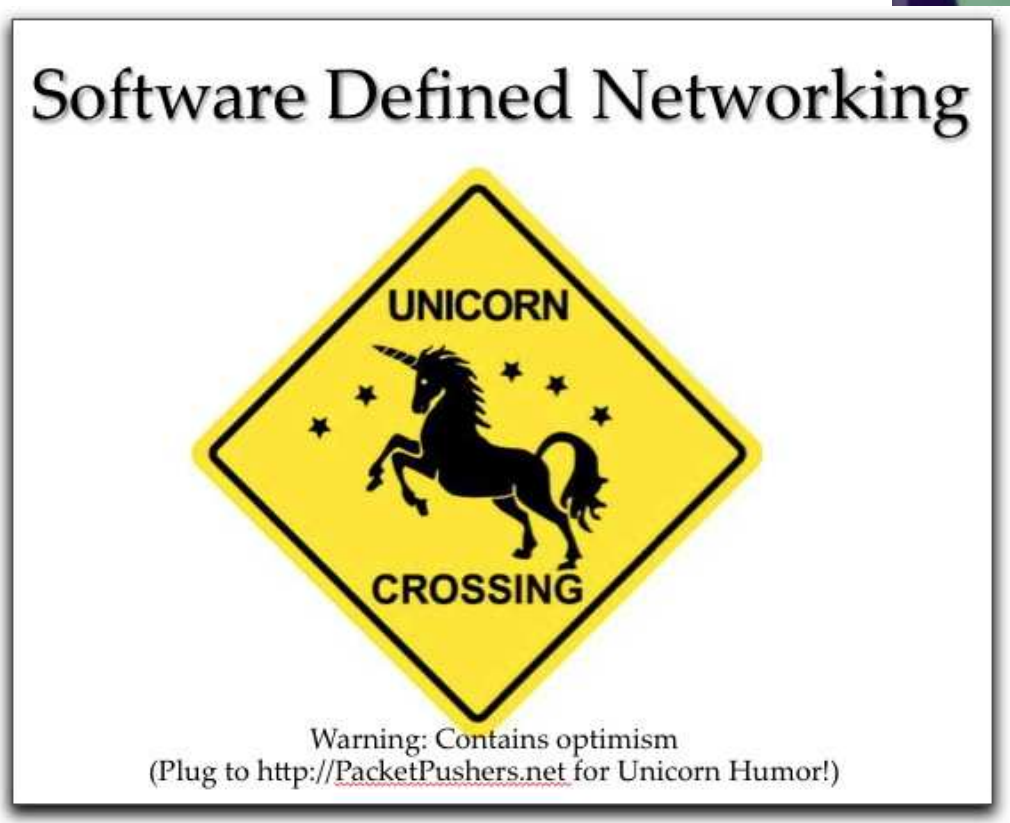
Physical topology drove the control design

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

--S. Shenker

Source: T. Koponen

SDN to the rescue!



Source: packetpushers.net

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 that will get my paper/grant accepted

-- Researcher Z


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

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

Some Interesting Use Cases

Use Case	Domains	Market Segments
DC Virtualization (Resource Slicing, Multi-tenant GW, Cloud Bursting)	Data Center (Edge - DC Edge)	Public Cloud High Performance
Data Center Interconnect	Data Center, Edge, WAN, Core	Public Cloud High Performance
Bandwidth Calendaring	WAN, Edge, Core	Public Cloud, High Performance, Service Providers, R& E
Multi-Layer Virtualization + Optimization	Core	SP, Public Cloud, High Performance, R&E
Content Request Routing (CDN)	Edge, Core, Data Center	SP
Virtual Patch Panel / Virtual Tap or Dynamic Traffic Steering	Campus & Branch, Datacenter, Core, Edge	SP, High Performance Public Cloud, R&E, Enterprise IT
Dynamic Threat Mitigation	Campus & Branch, Data Center, Edge	SP, Public Cloud, High Performance, R&E, Enterprise IT
Network Access Control	Campus & branch	Enterprise IT

Source: ONF

Solution- or Problem-focused Research?

solution-focused?

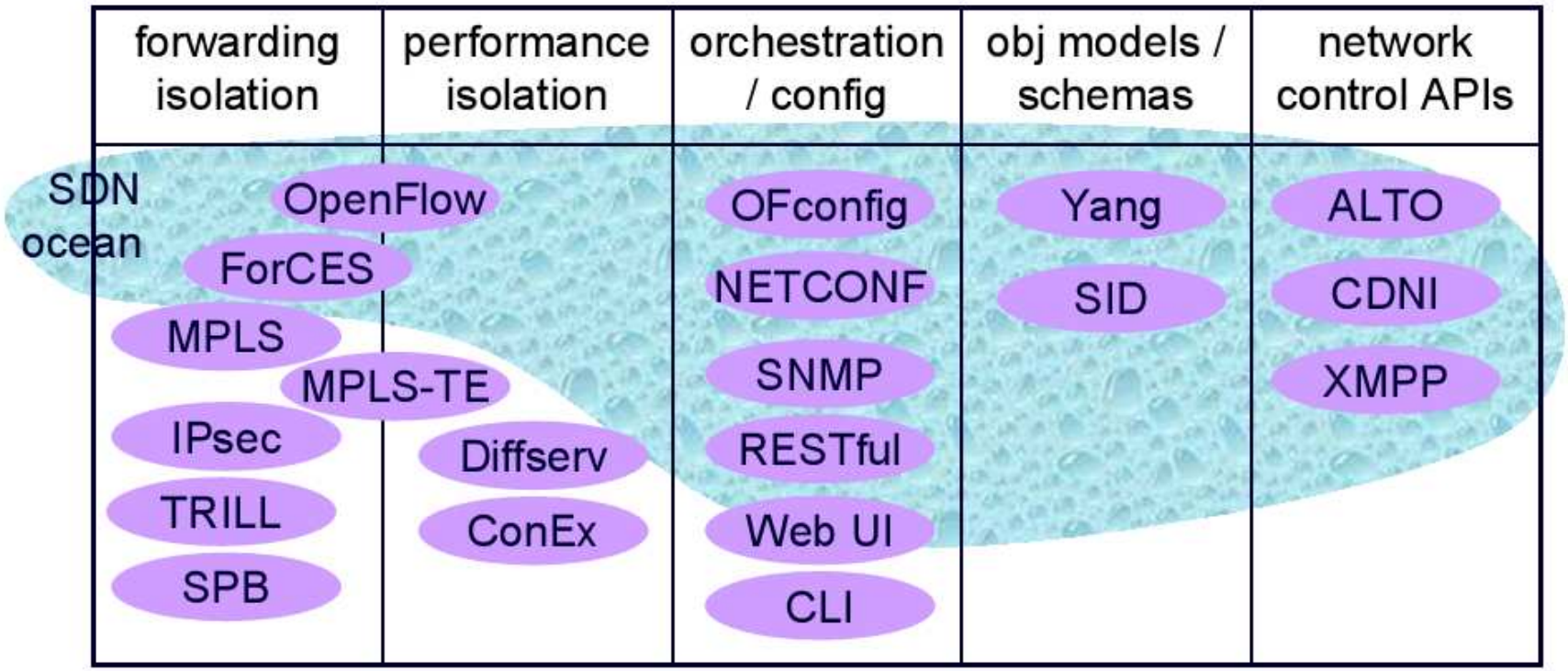
- assume SDN, then solve outstanding potential problems with it? e.g.
  - policy complexity
  - modelling scalability
  - inter-domain
  - validation of evolvability claims
  - ... see mailing list & other presentations (esp. Dave Ward's IAB plenary) for dozens more suggestions

problem-focused?

- pre-standards convergence on best approach to solve a problem?
  - how does centralised FIB distribution compare vs decentralised vs hybrid?
  - is it best for forwarding isolation to be independent of performance isolation for virtual networks?
  - is it best to use the same architecture for FIB distribution and for config?
  - how best to do multipath & traffic engineering

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

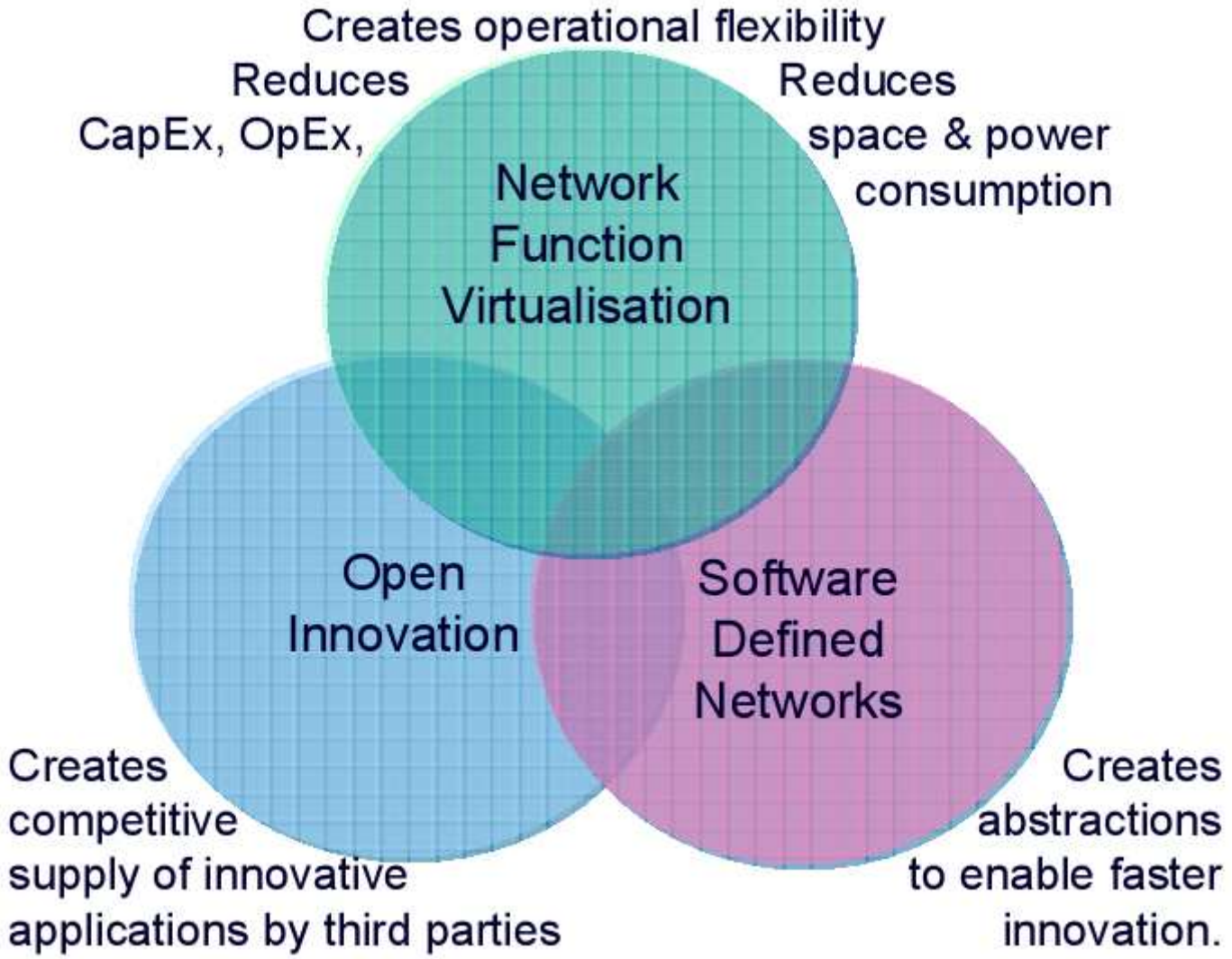
Mapping of technologies to problems



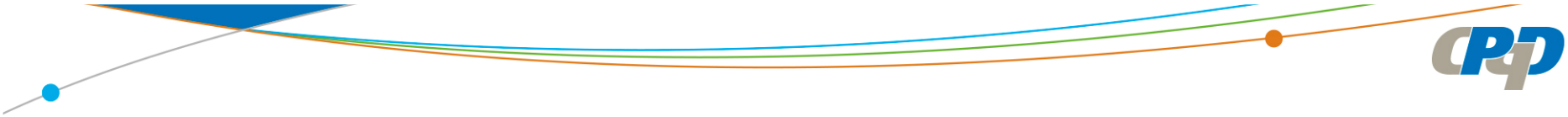
- we could focus on solving a particular problem (column)
- we could also ask
  - is an integrated solution (multi-column) good?
  - or "do one thing and do it well" so operators can pick & choose rather than lock-in to an über-solution?

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

Strategic Networking Paradigms for Network Operators

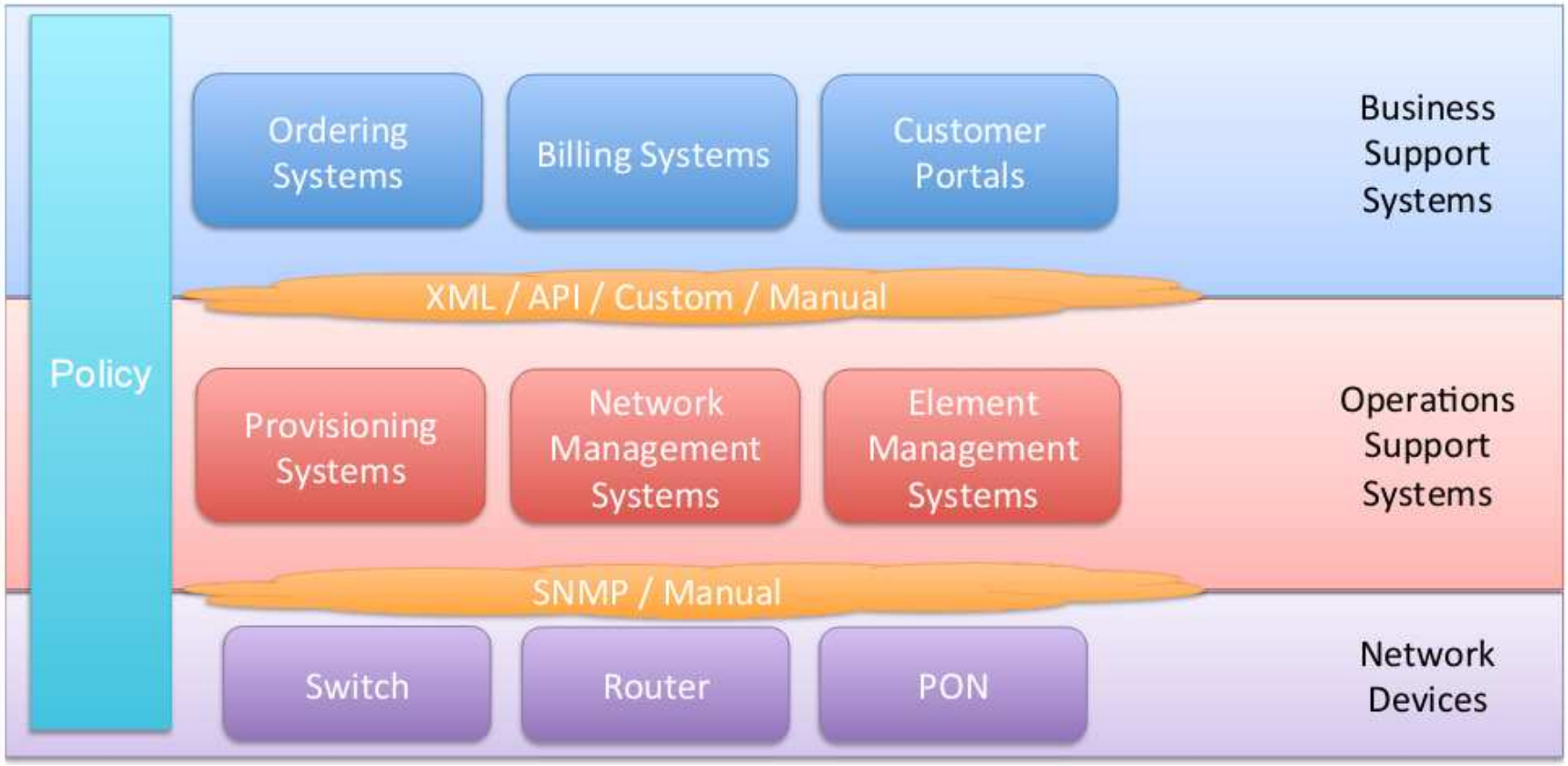


Source:NFV



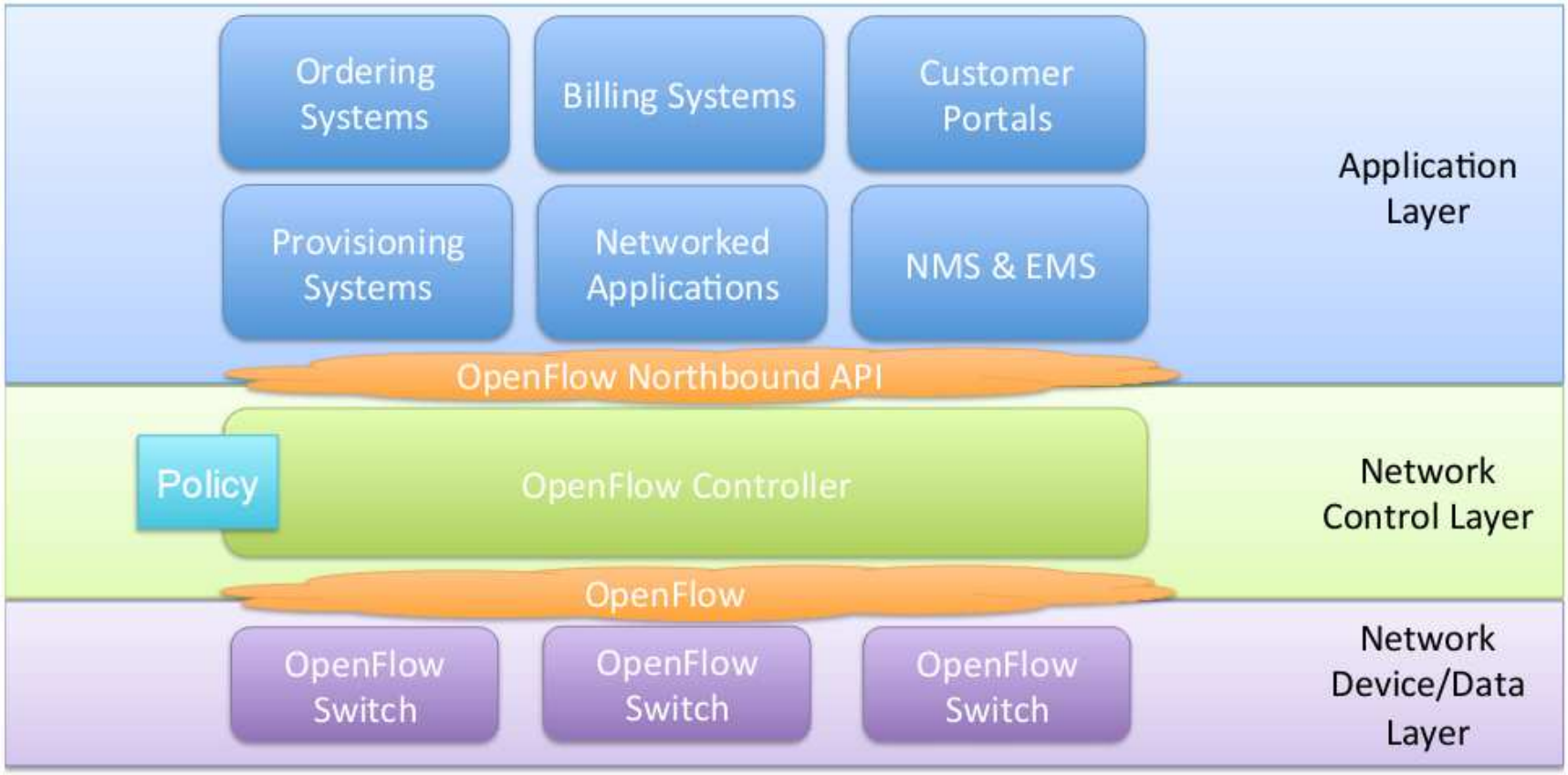
A VERY MUCH NEEDED DISCUSSION ON SDN MODEL(S)

Yesterday´s SDN



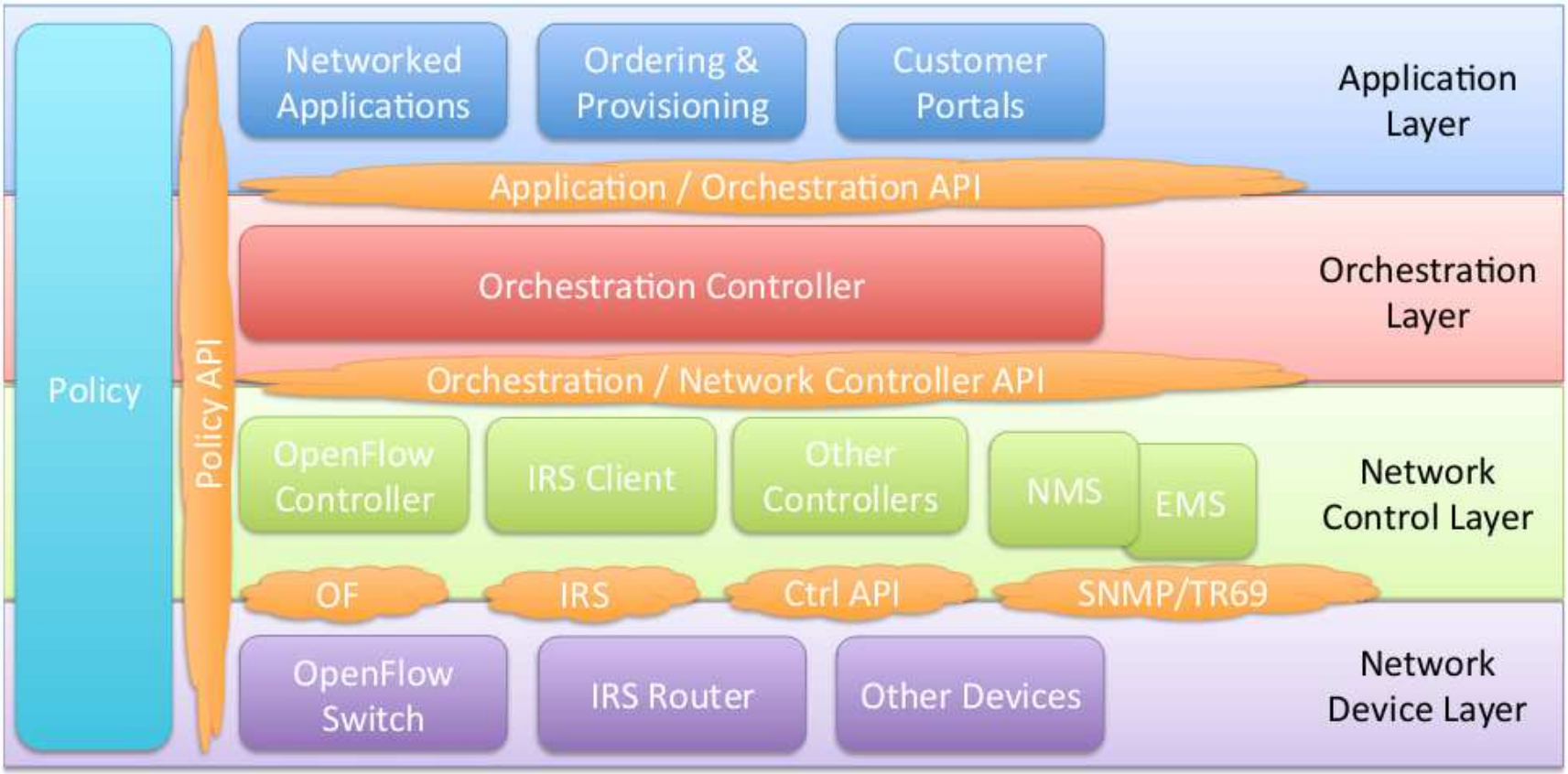
Source: Chris Grundemann

The OpenFlow model



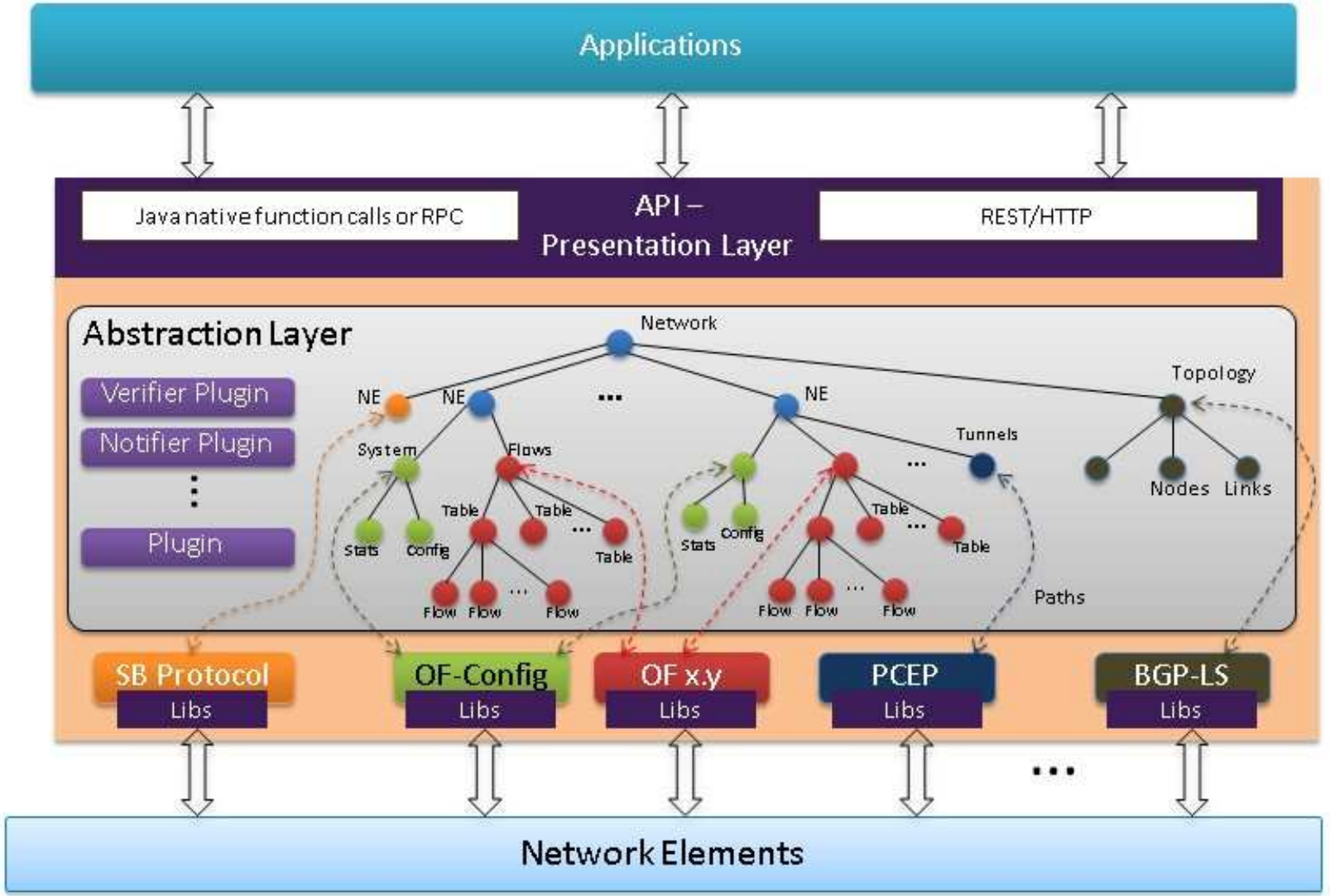
Source: Chris Grundemann

Emerging SDN Ecosystem



Source: Chris Grundemann

OpenDaylight Controller Platform



Source: opendaylight.org

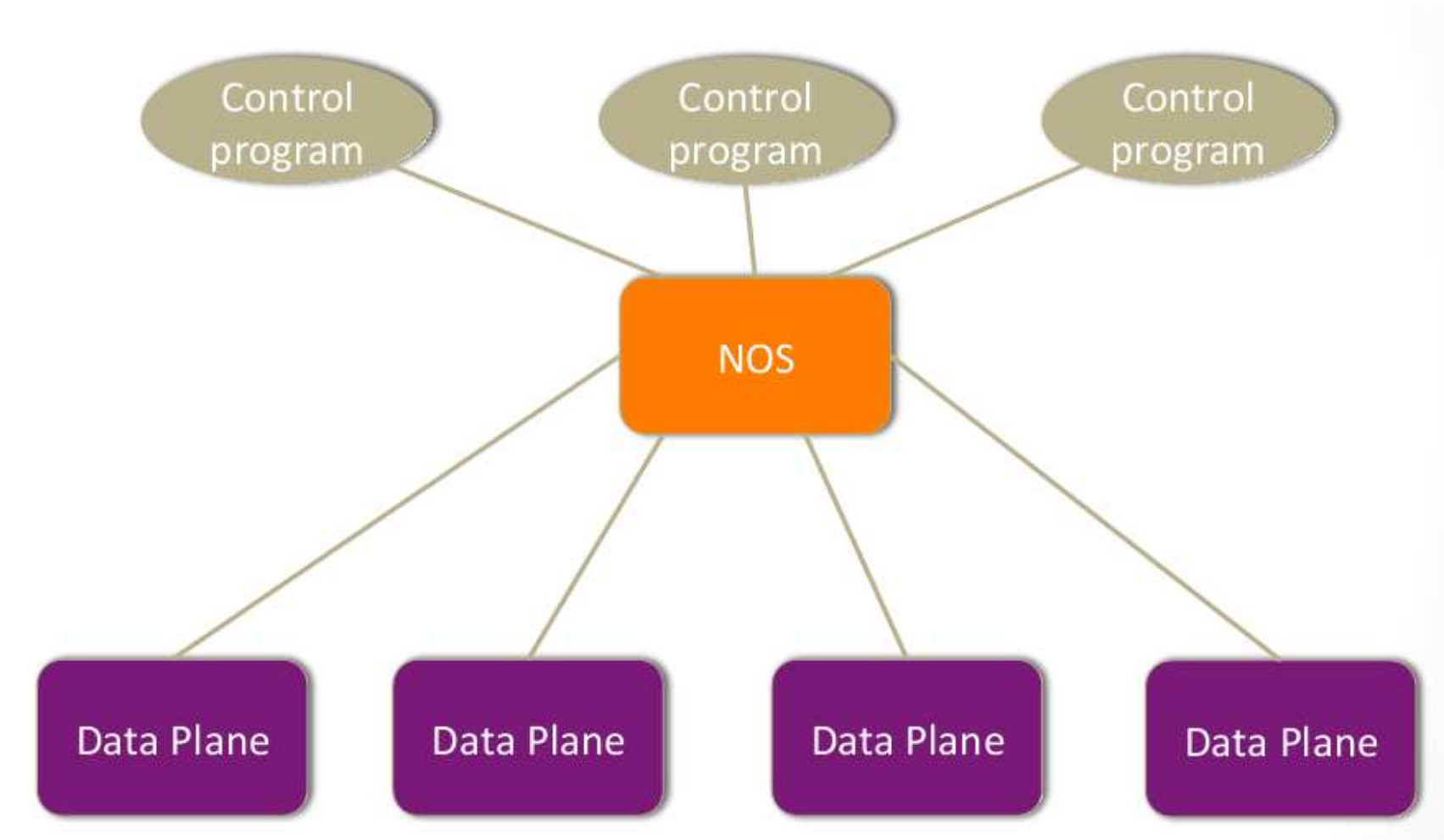
SDN Models

SDN can be considered in terms of three distinct models

- 1. A **Networking/Operating/System** that oversees the network data plane and hosts a number of “control programs” that define networking services
- 2. A **Broker** through which applications interact with and affect the network so that the apps are more effective, are more efficient and/or offer better user experience
- 3. A **Compiler** that translates a high-level language in which an operator defines what they want from the network and compiles it into low-level instructions for the data plane

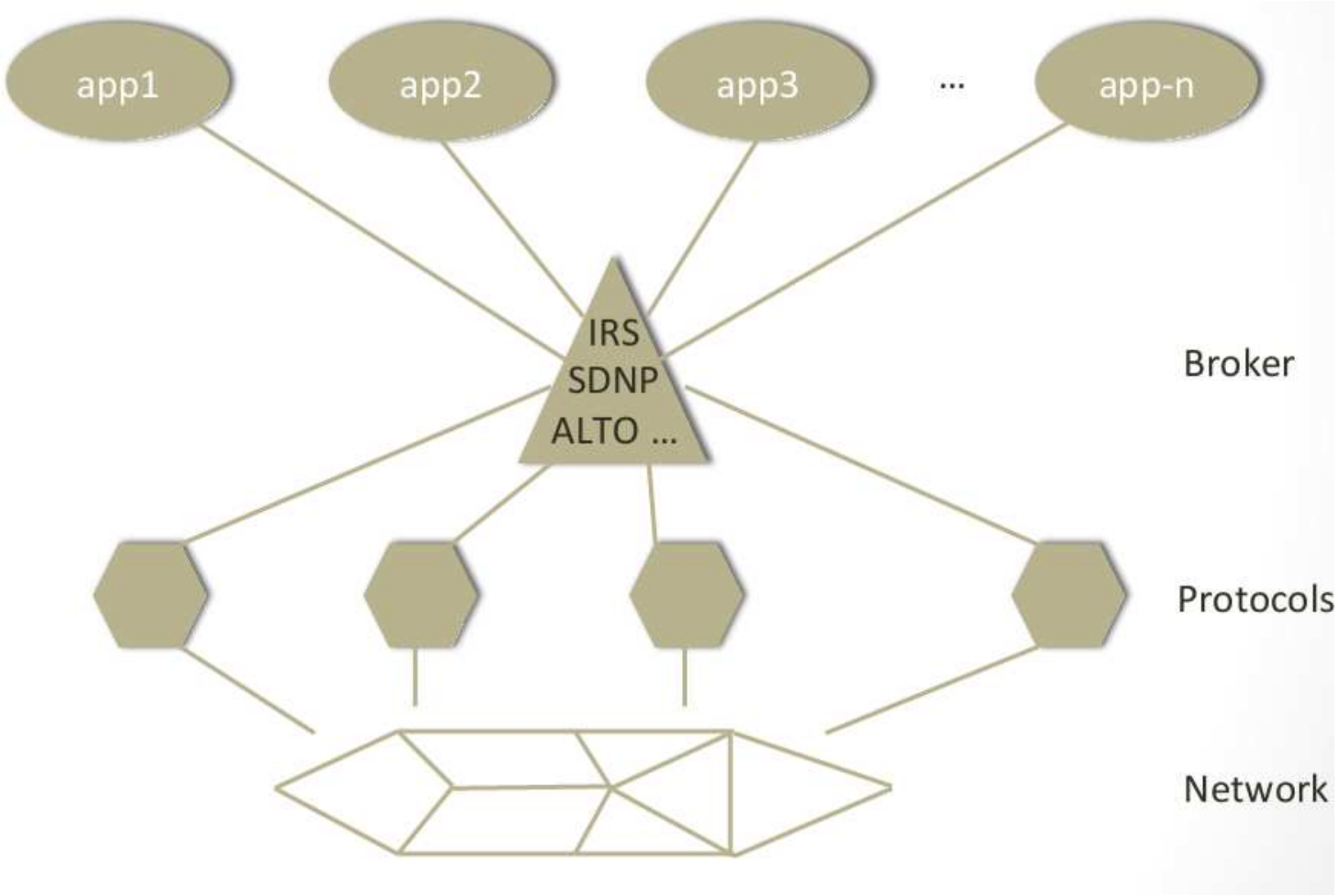
Source: K. Kompella, slides-85-sdnrg-2.pdf

Model : Networking Operating System



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

Model : Broker



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

OpenFlow Is Not the Only SDN Tool

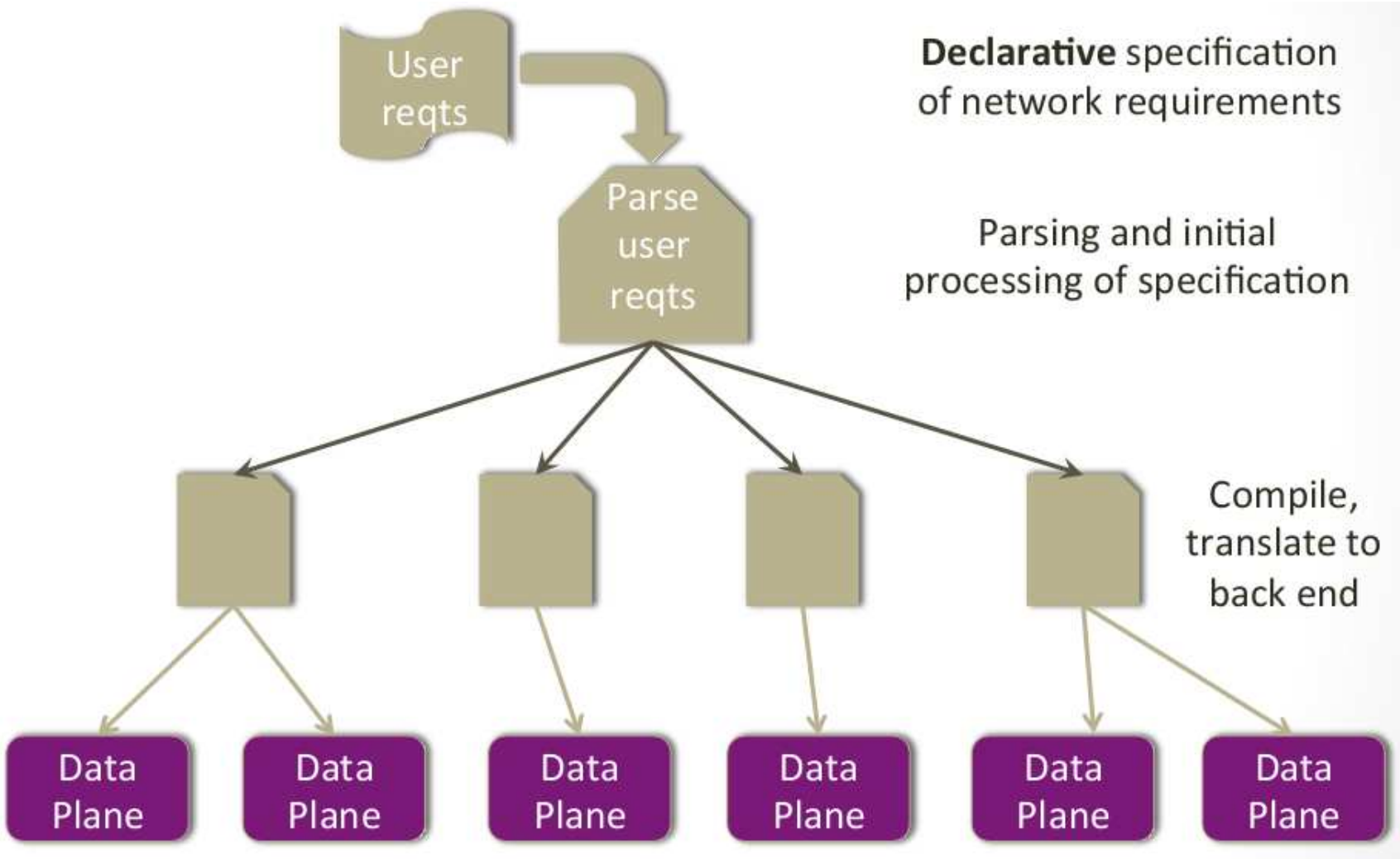
Tool/Standard	Functionality
OpenFlow (ONF)	FIB/TCAM manipulation
NETCONF (IETF)	Configuration management
OF-Config	OpenFlow switch configuration management (YANG schema)
Internet Routing System (IRS, IETF non-WG)	Routing table interaction/manipulation

Vendor APIs

- Cisco: Open Networking Environment (ONE), EEM (Tcl), Python scripts)
- Juniper: Junos XML API and SLAX (human-readable XSLT)
- Arista EOS: XMPP, Linux scripting (including Python and Perl)
- Dell Force10: Open Automation Framework (Perl, Python, NetBSD shell)
- F5: iRules (Tcl-based scripts)

Source: I. Pepelnjak

Model : Compiler

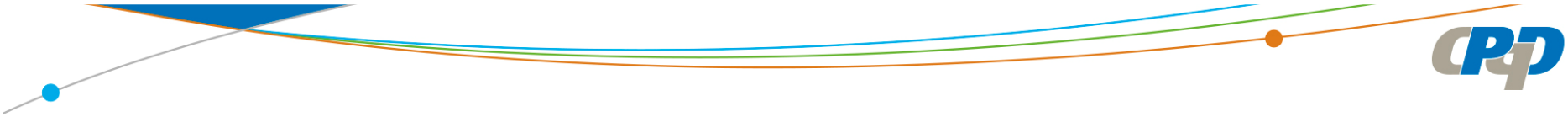


Source: K. Kompella, slides-85-sdnrg-2.pdf

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



SDN RESEARCH

Oportunidades / Desafios de Pesquisa

SW remoto

Aplicação/ Solução

High-level languages / policies, graph theory

[FML, Procera, NetCore]

Configuration Abstraction

Capture intent, not mechanism!

Controlador / SO de Rede

Sistemas Distribuidos (Bases de dados, P2P, HA)

Network Abstraction

Hardware / Forwarding Abstraction

Flexibility and Programmability!

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

Equipamento

SEmb/OpenFlow

HW

ASIC / NP /x86

Novas arquiteturas HW/SW, baixo custo, menor consumo
Algoritmos de flow matching, etc.

Simplificado/Otimizado para comutação de fluxos

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

Control Plane Research Questions

Distribution and Separation

- Optimality
- Resilience
- Coverage
- Visibility
  - Global versus local?
- Scale
  - horizontal,
  - data base,
  - offline path computation

Logically Centralized?

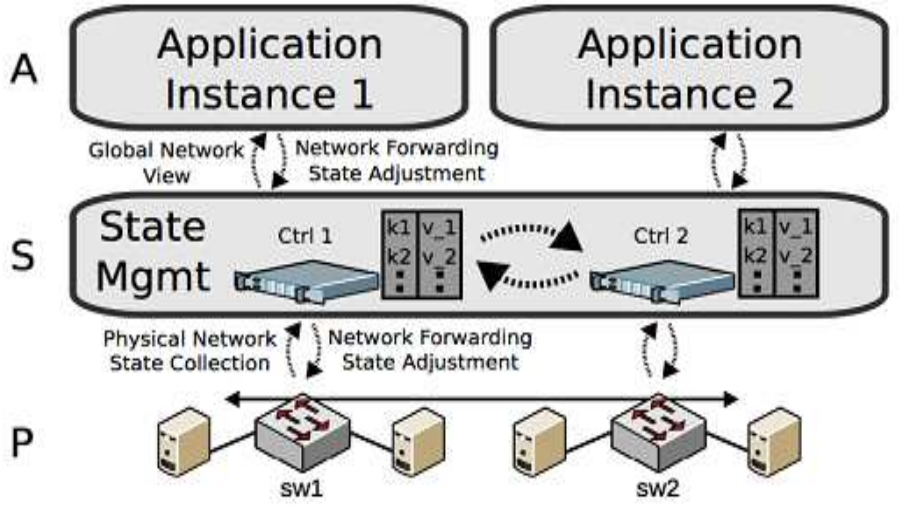


Figure 1: SDN state distribution and management conceptualized in layers: (A)pplication, (S)tate Management, (P)hysical Network

[Levit et al., HotSDN 12]

Key Observation: Logically centralized -> Distributed System -> tradeoffs between control plane convergence and state consistency m
And what about the loss of control plane/data plane fate sharing?

Source: D. Mayer

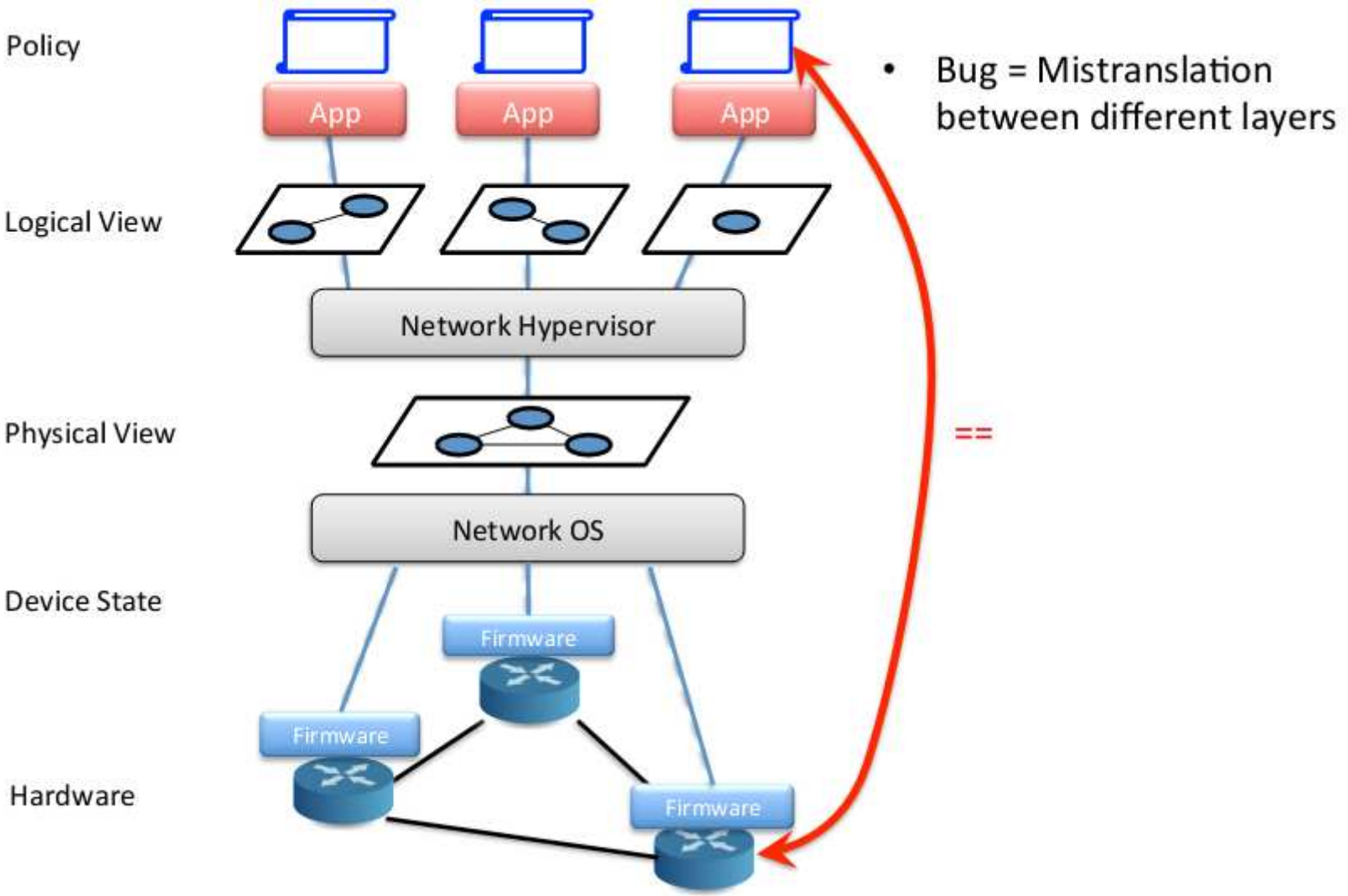
Active research topic: SDN Troubleshooting

SDN provides a unique opportunity for systematic troubleshooting.

- Decouples control plane from data plane
- State changes pushed from a logically centralizedlocation
- Easier to access/observe the state of the network
- SDN architecture provides clear abstraction for control plane functionality
- Richer troubleshooting techniques

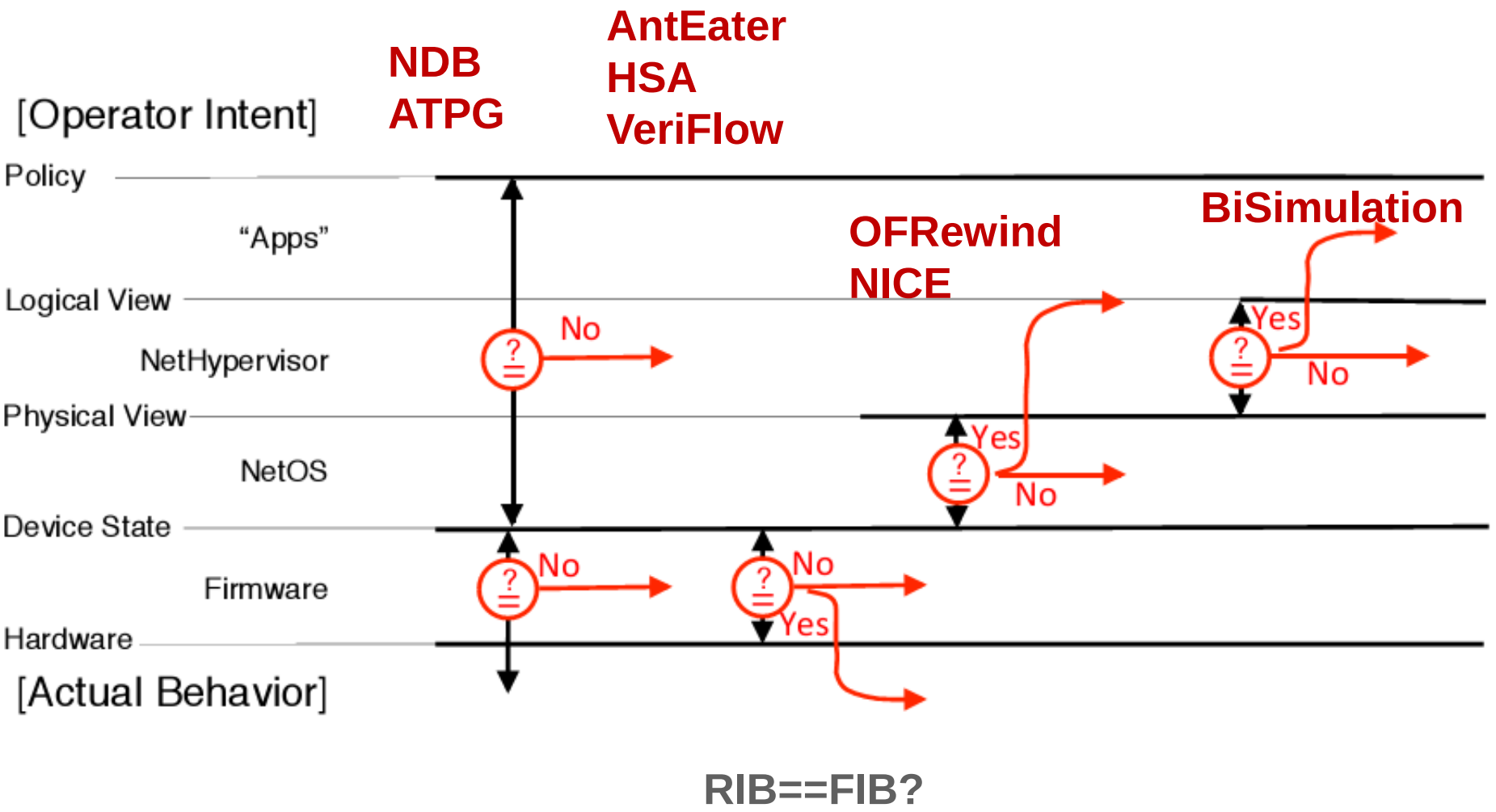
Source: P. Kazemian, slides-85-sdnrg-5.pdf

Bugs in SDN architectures



Source: P. Kazemian, slides-85-sdnrg-5.pdf

Ongoing research on SDN troubleshooting



Source: P. Kazemian, slides-85-sdnrg-5.pdf

SDN: a Fundamental Step Forward

- or just a new whip to beat vendors with?

What makes SND 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

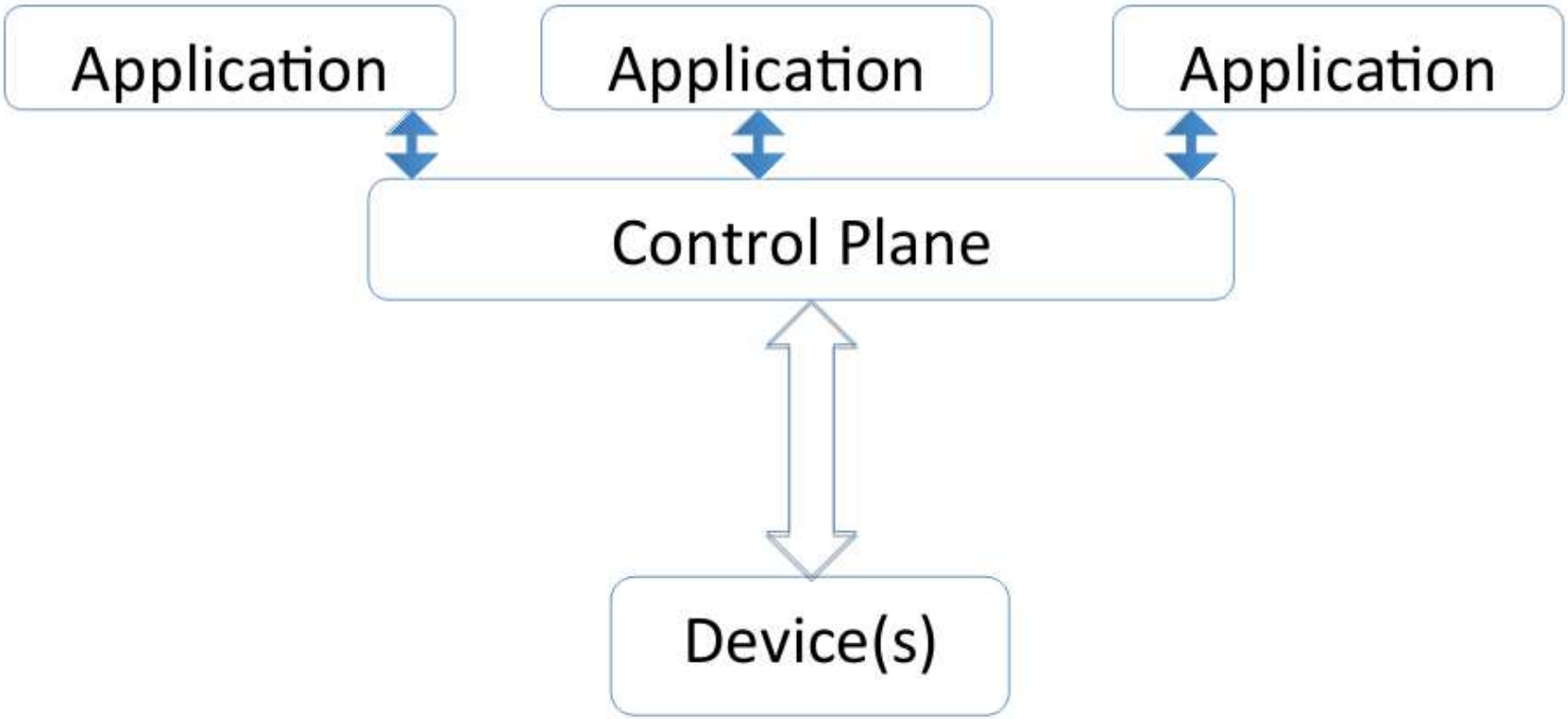
Source: K. Kompella, slides-85-sdnrg-2.pdf

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

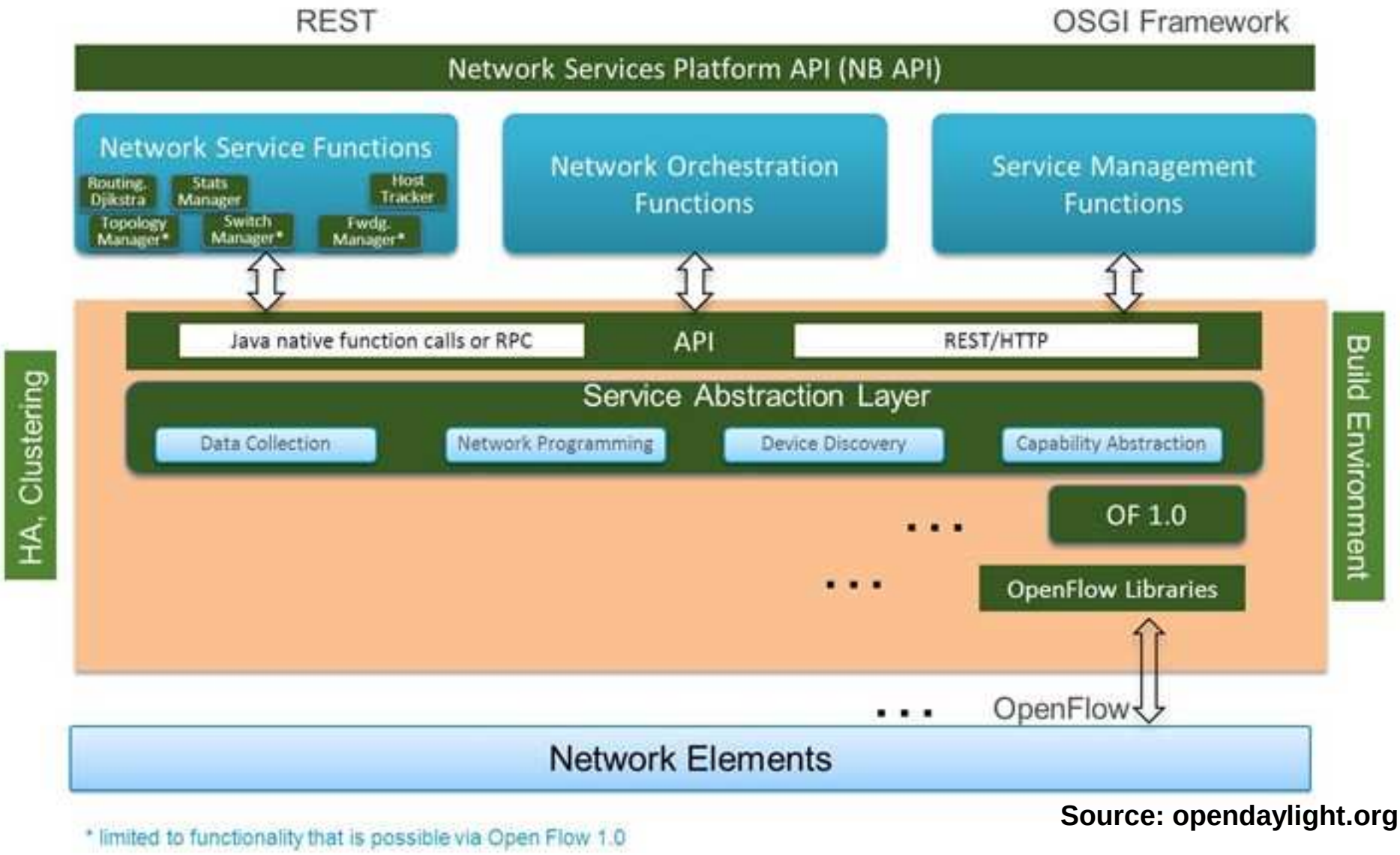
Source: K. Kompella, slides-85-sdnrg-2.pdf

Implementation of a SDN Control Plane



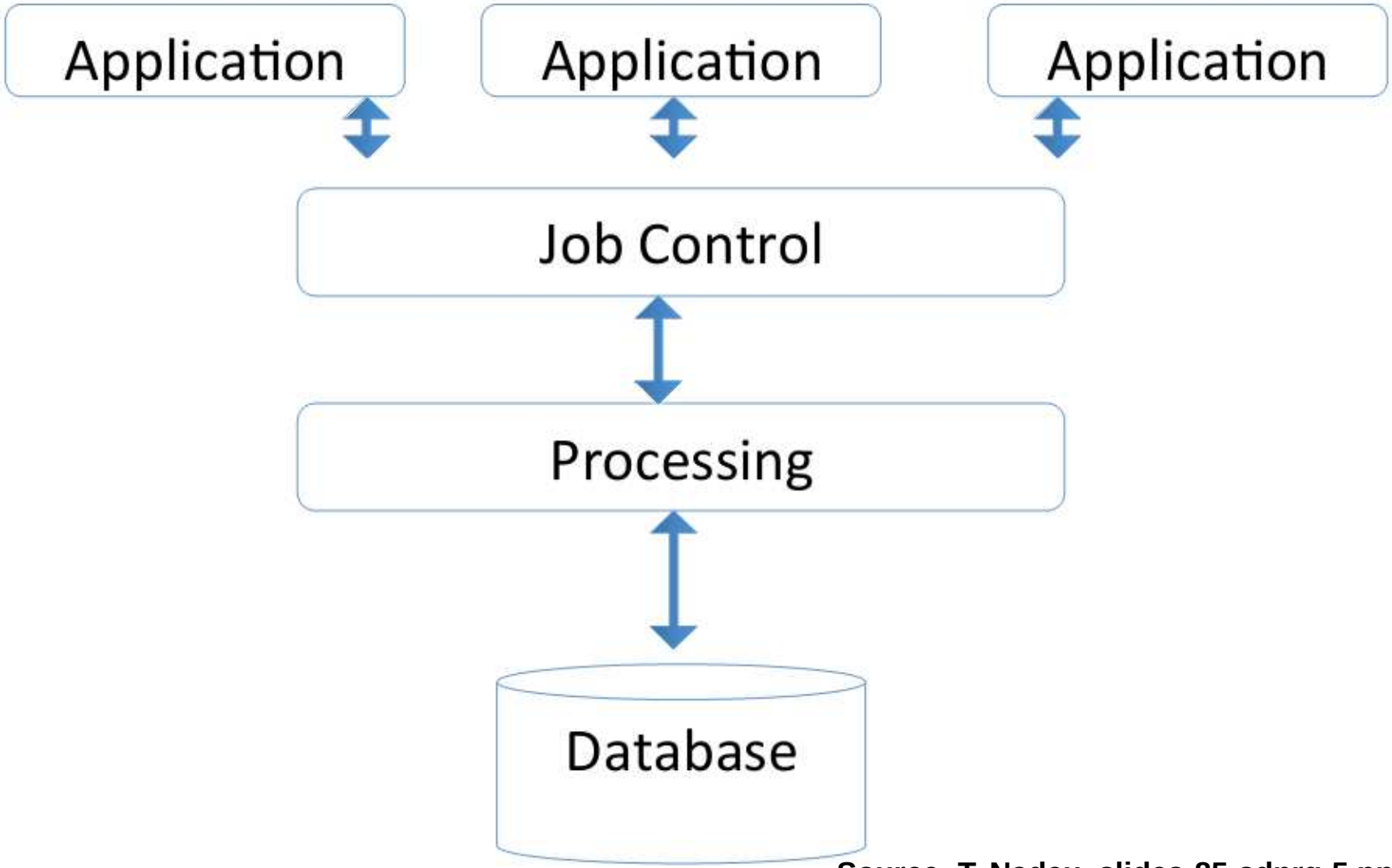
Source: T. Nadeu, slides-85-sdnrg-5.pptx

OpenDaylight Controller Platform



Source: opendaylight.org

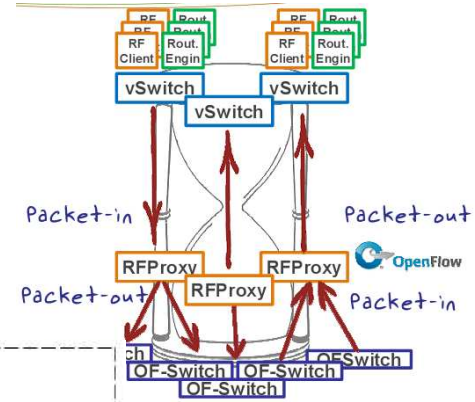
Anatomy of a Control Plane as a Modern Controller

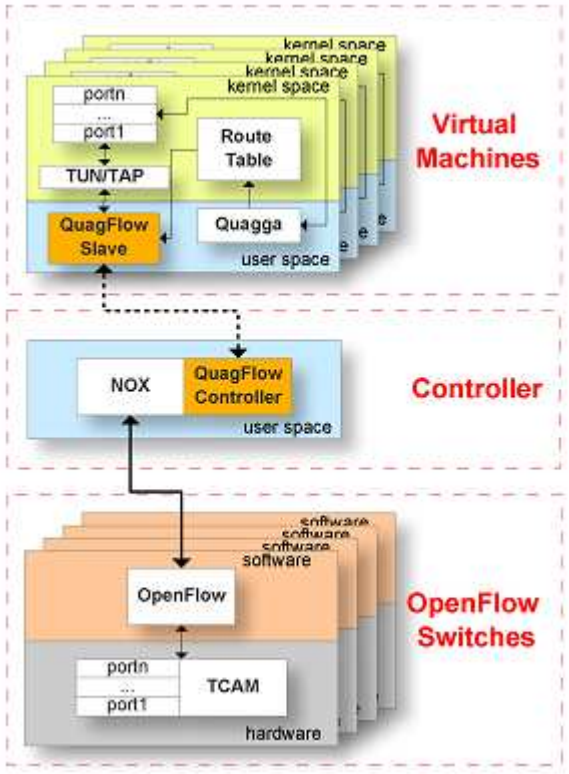


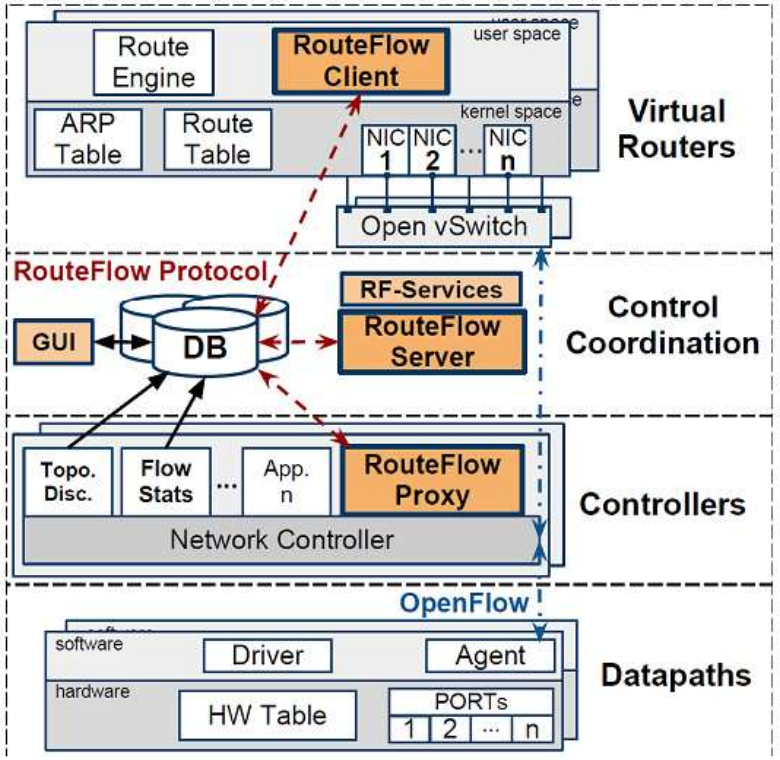
Source: T. Nadeu, slides-85-sdnrg-5.pptx

RouteFlow control plane architecture

- From QuagFlow to current implementation:

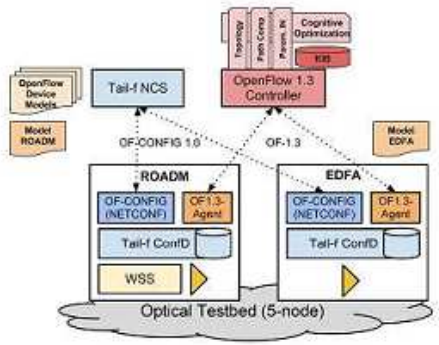






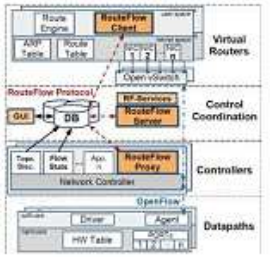
Ongoing R&D activities at CPqD

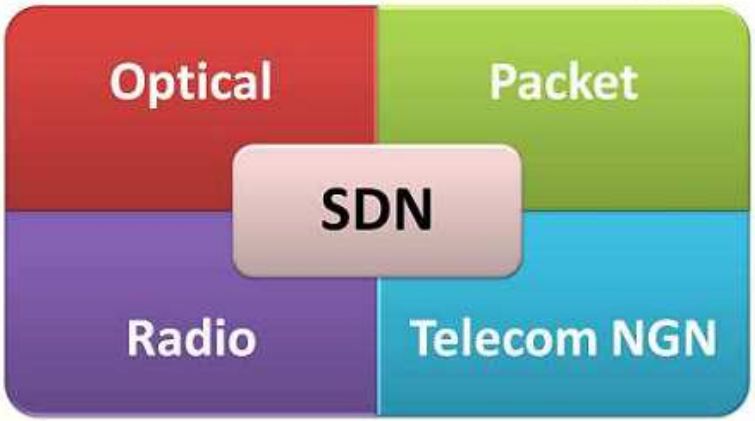
Software-Defined Optical Transport

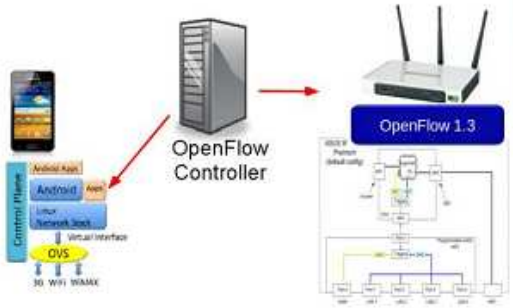


Software-Defined IP Routing

RouteFlow







openstack™

OpenFlow 1.x prototyping

Software-Defined Wireless Networking **Cloud** **Software-Defined Telecom Services**

Reflections on FIBRE research opportunities

- OpenFlow is just (but currently best) low-level tool

- FIBRE should be the *means* of SDN research not the *end*
- Think big on the SDN research opportunities!
- Problem- or Solution-focused research? Or solution-prompt?
- Spiral / DevOps approach (Design-> Experiment -> Refine)?

- FIBRE and dependency on the FlowVisor virtualization layer
- Plans to embrace OpenFlow 1.x?
- Rethink the virtualization strategy?

- Let’s work towards a network science discipline
- Not simple SDN-washed papers

Further reading

- HotSDN12, HotSDN13
- ONS 2013 Research Track
- IRTF SDNRG

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

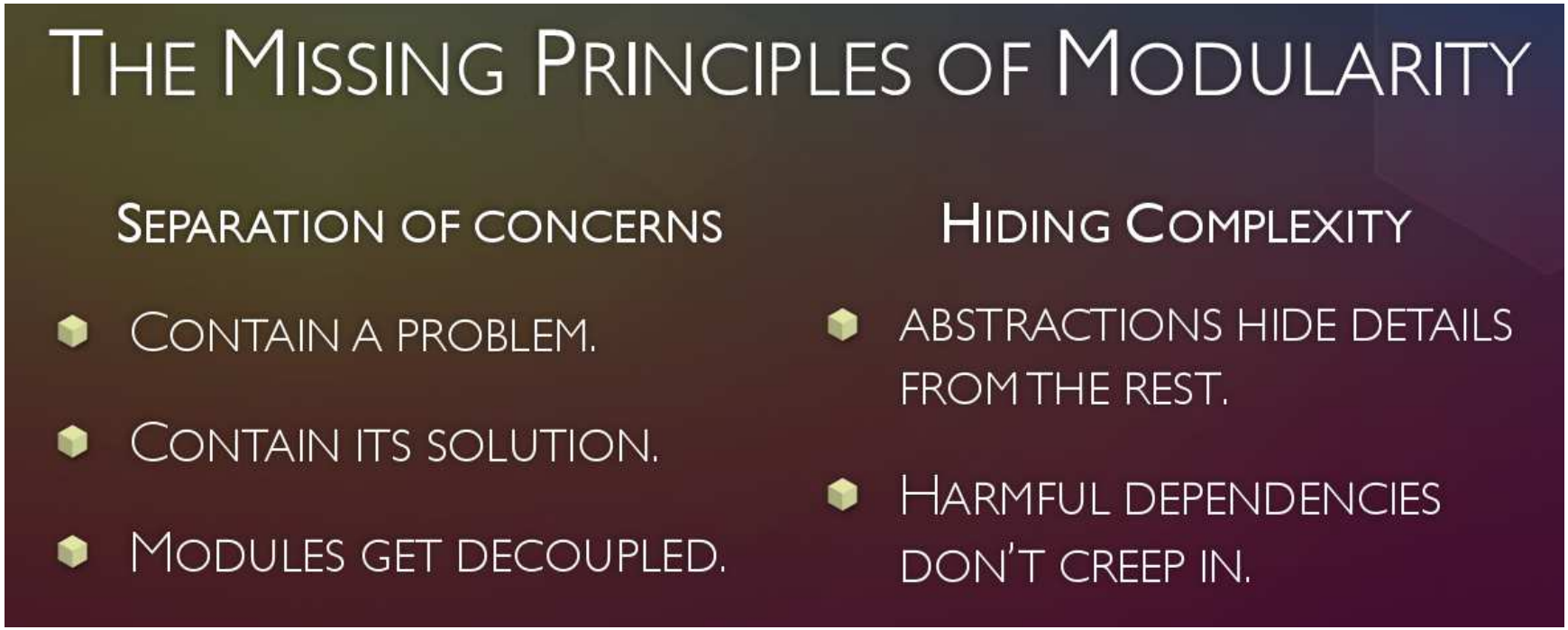
Thank You!

Visit our GitHub repos!

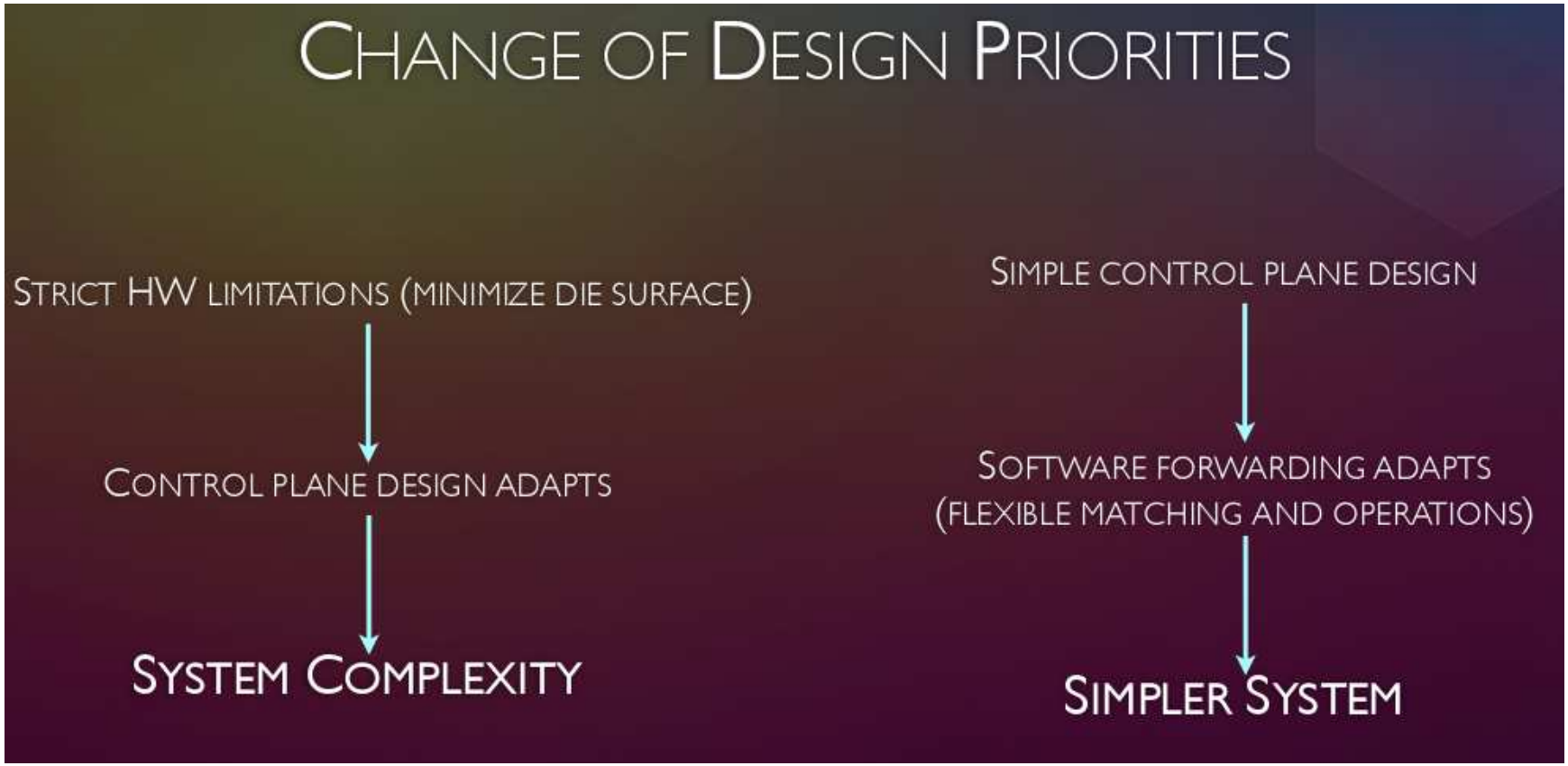
http://github.com/CPqD

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

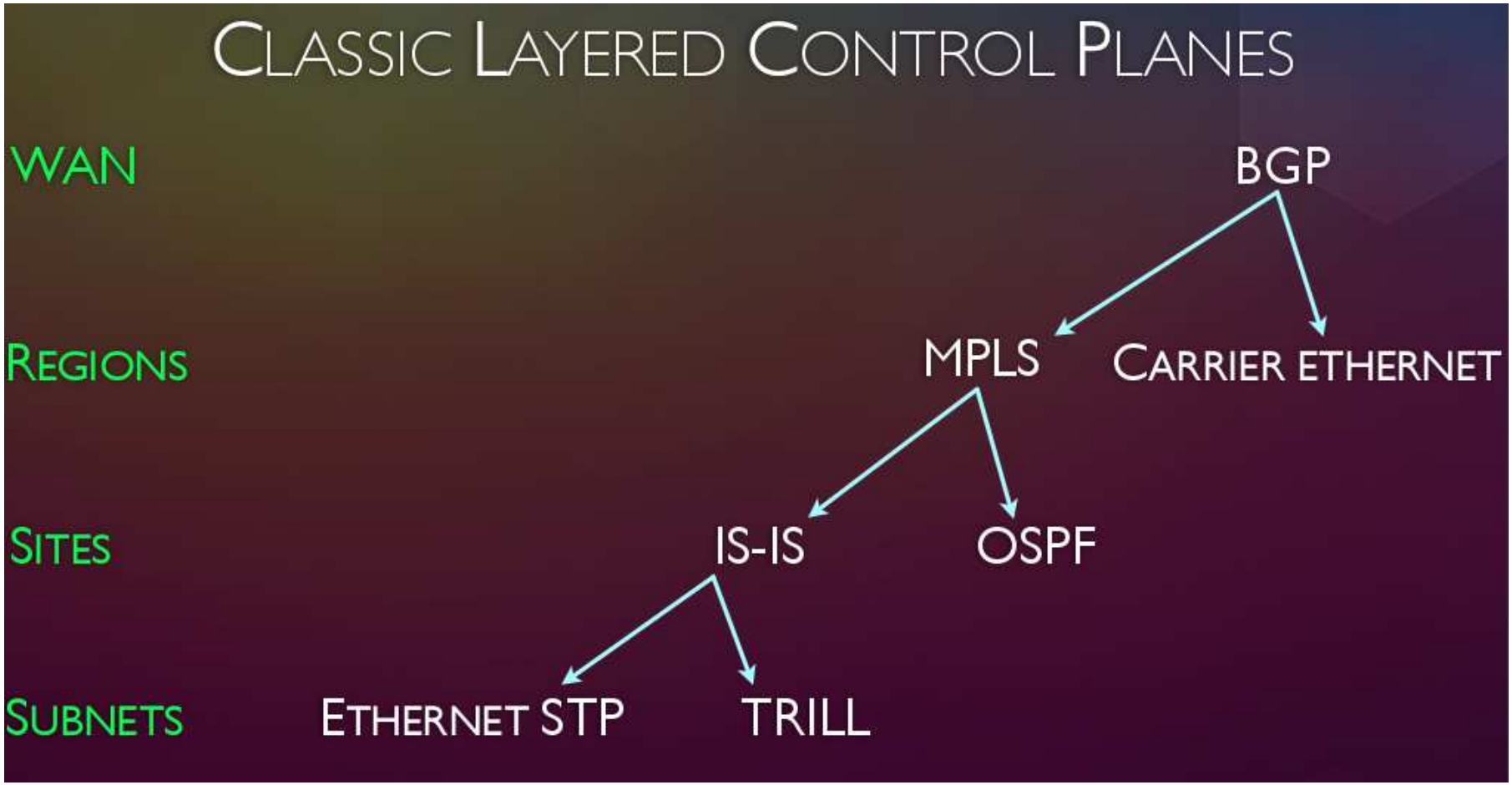
BACKUP



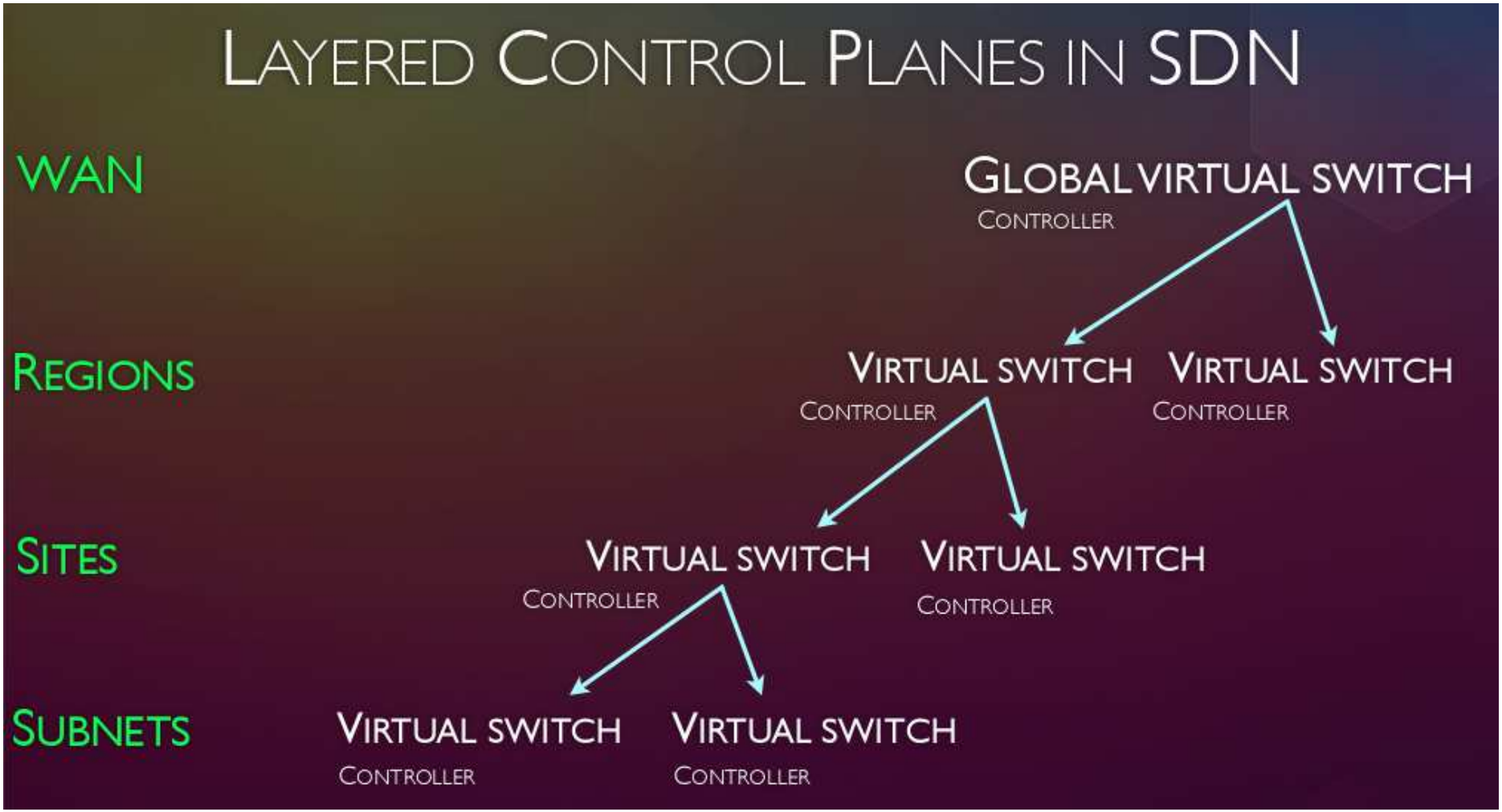
Source: T. Koponen



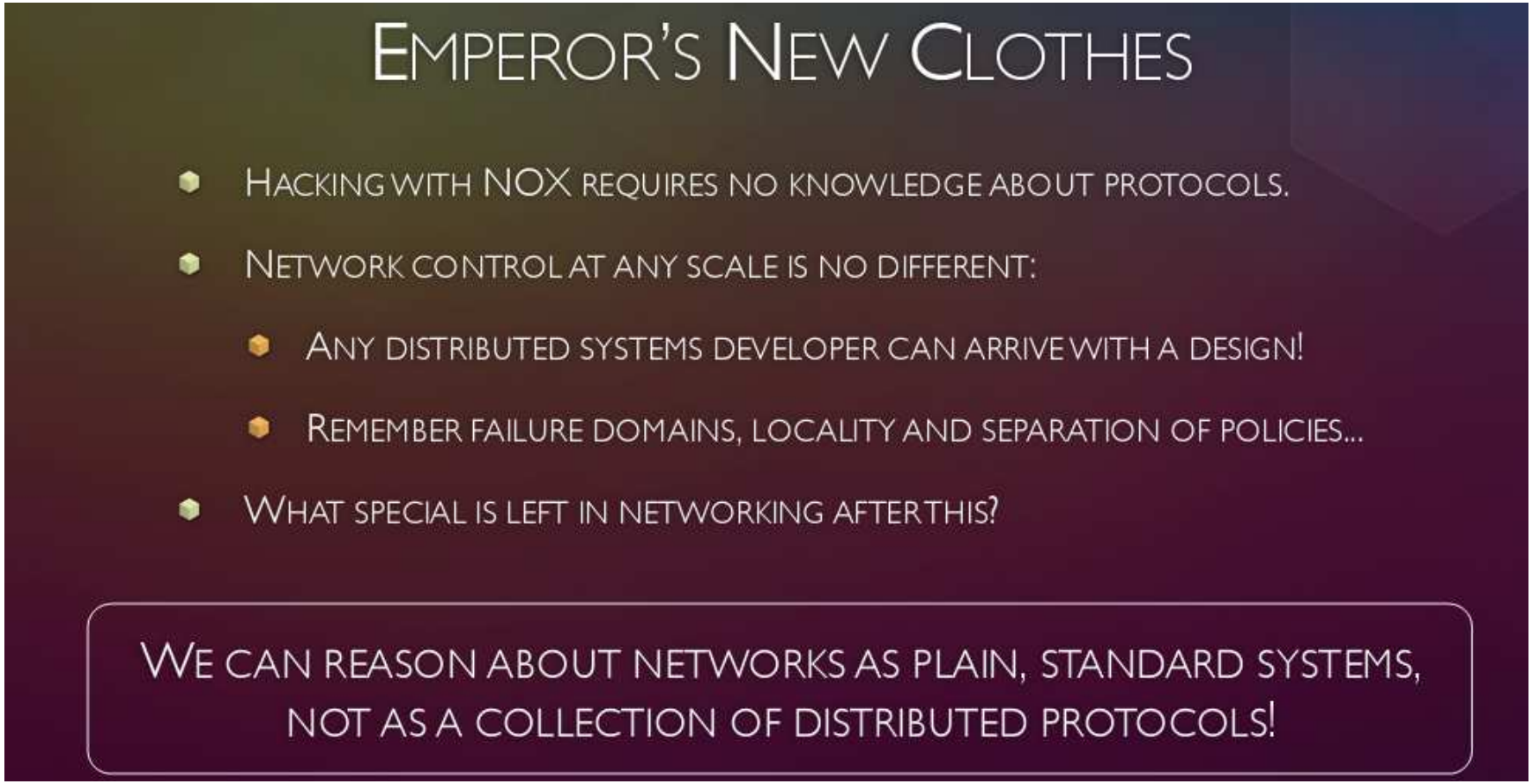
Source: T. Koponen



Source: T. Koponen

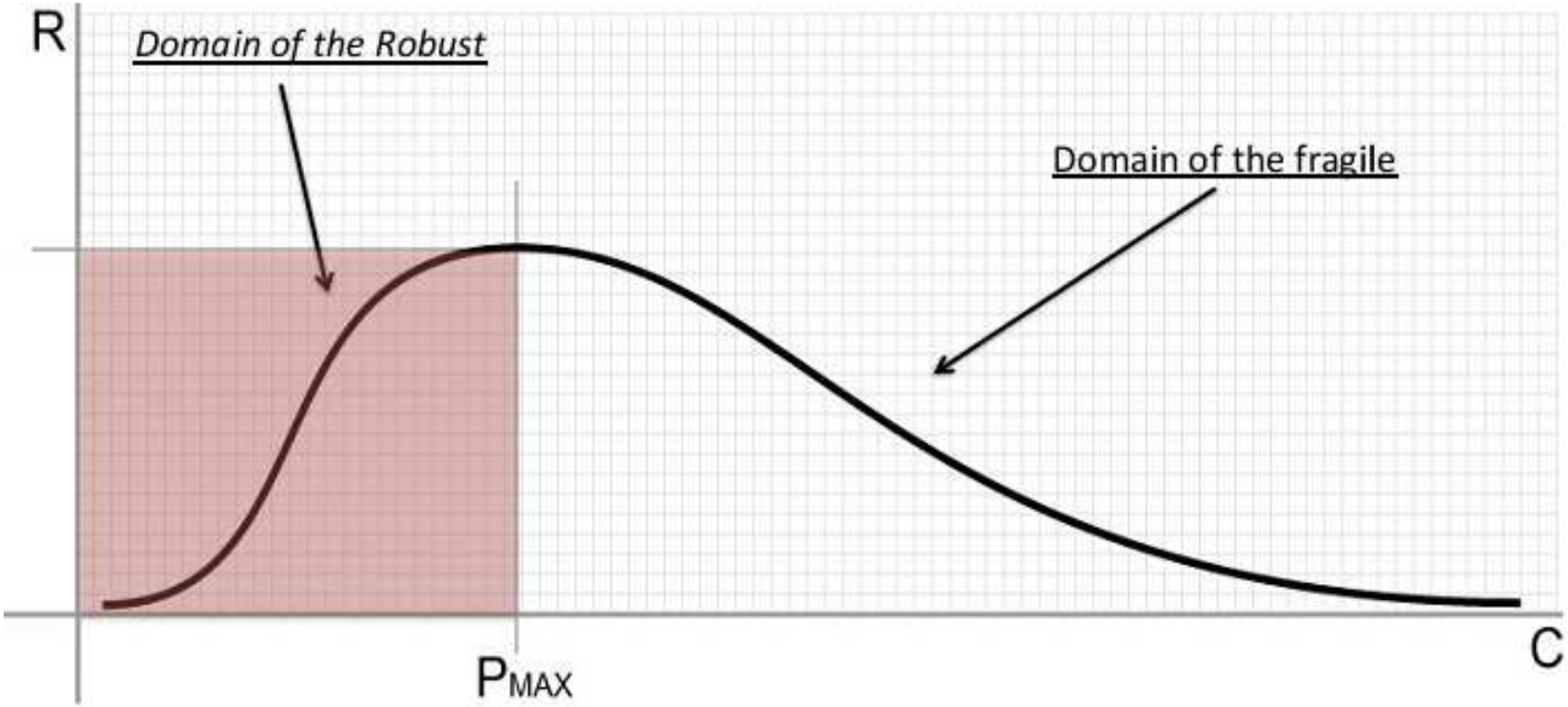


Source: T. Koponen



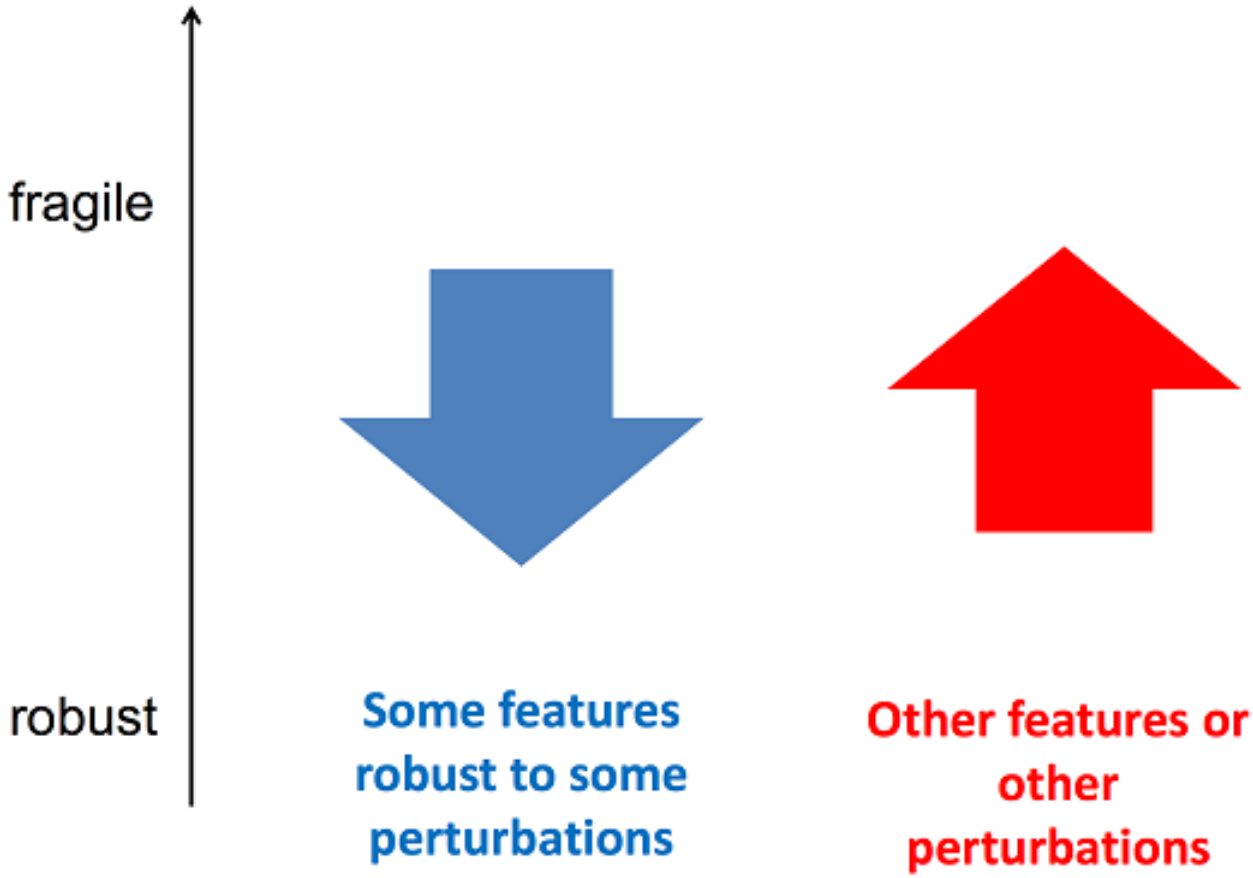
Source: T. Koponen

Robustness vs. Complexity Systems View



Can we characterize the Robust and the Fragile?

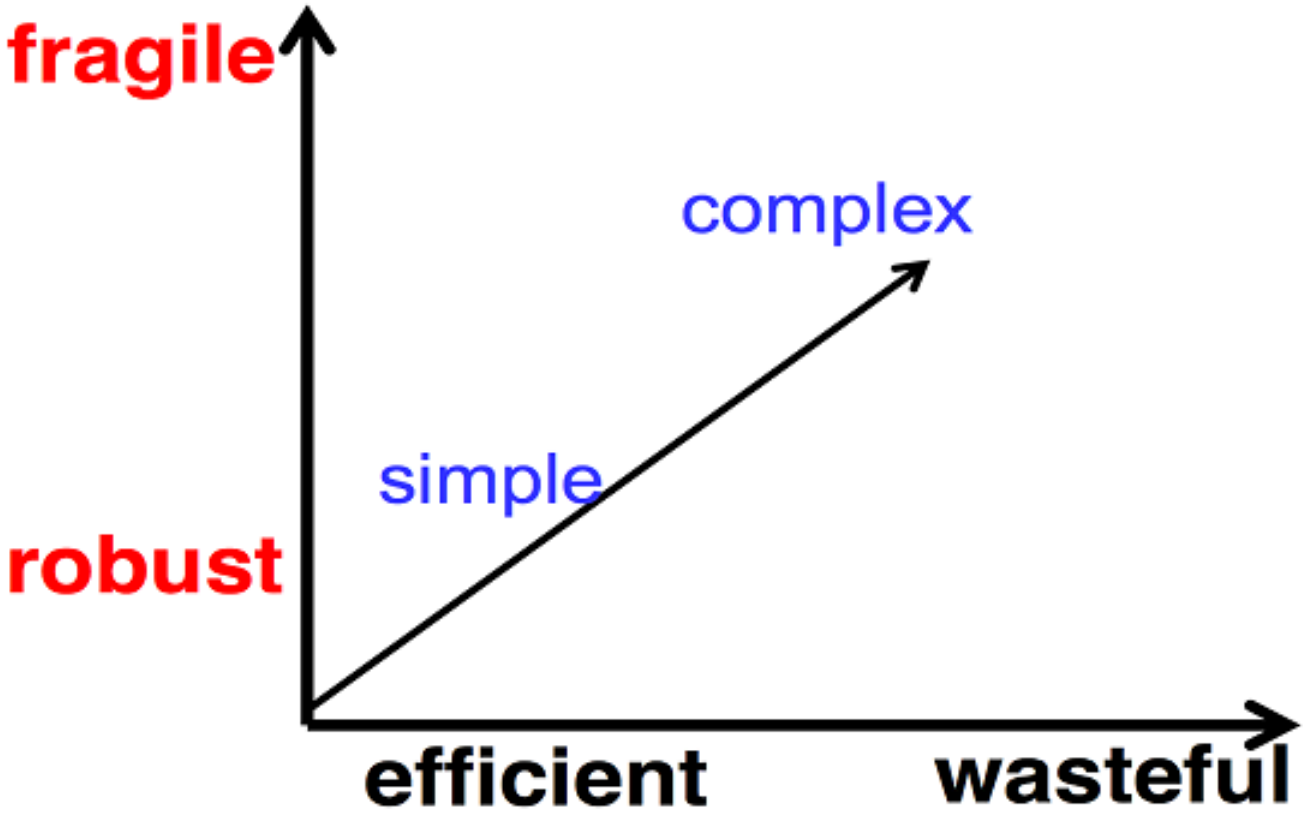
Robust-Yet-Fragile [Doyle]



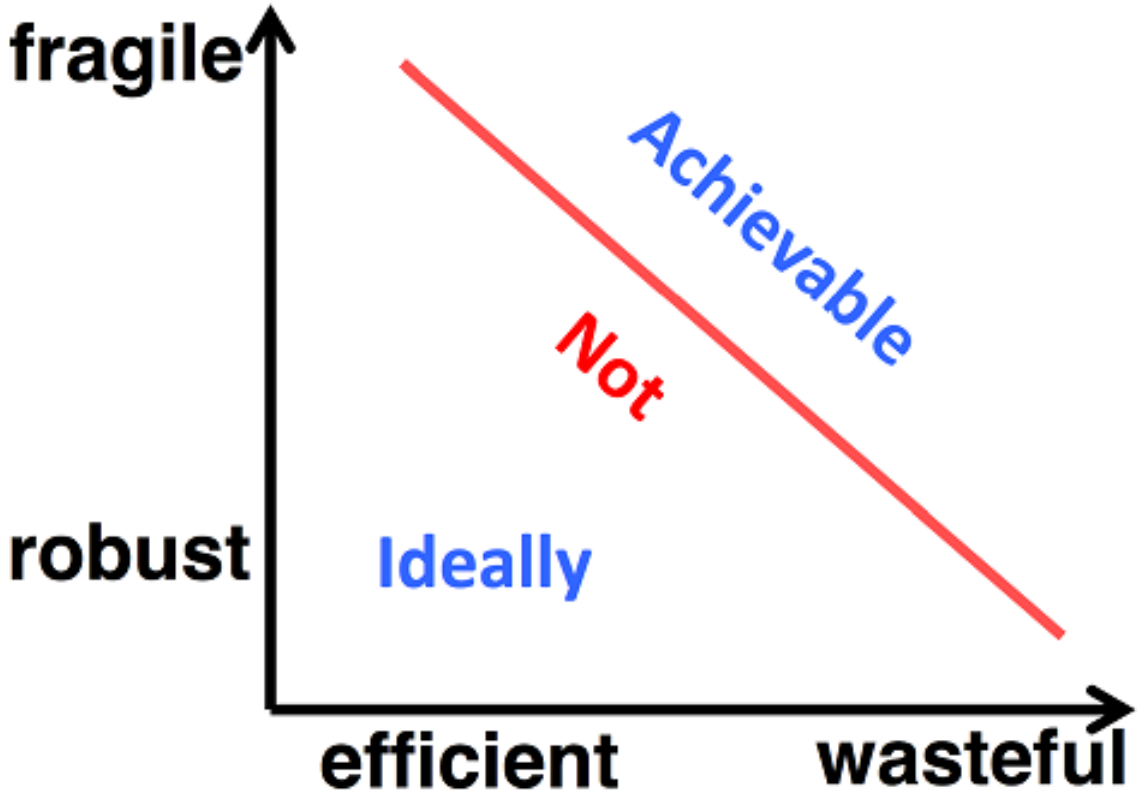
RYF Tradeoffs

Robust	*and*	Fragile
Modular	*and*	Distributed
Simple	~~*xor*~~	Complex
Plastic		Frozen
Evolvable		Frozen

RYF -- Another view



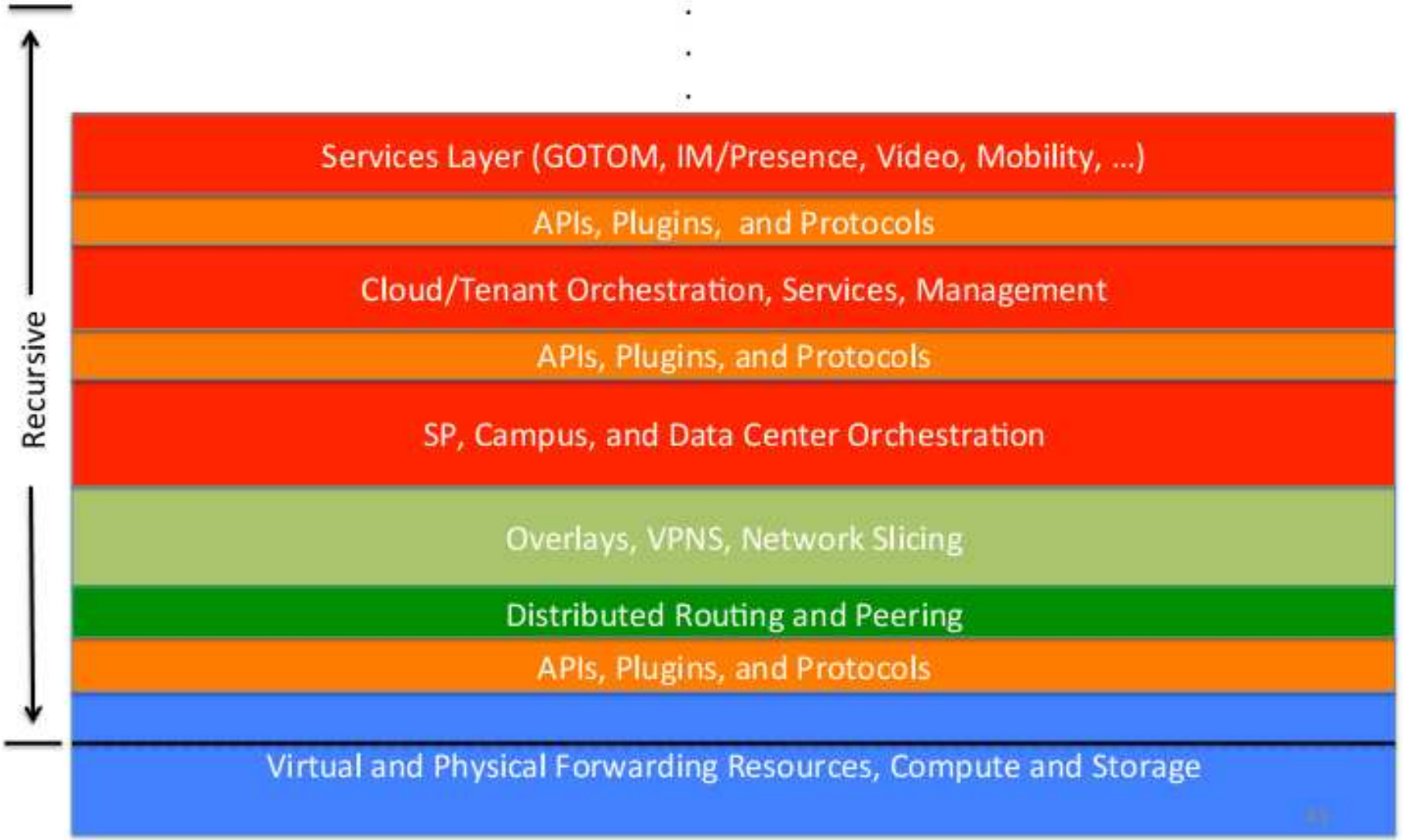
RYF - Limits


fragile
robust
efficient
wasteful
Achievable
Not
Ideally

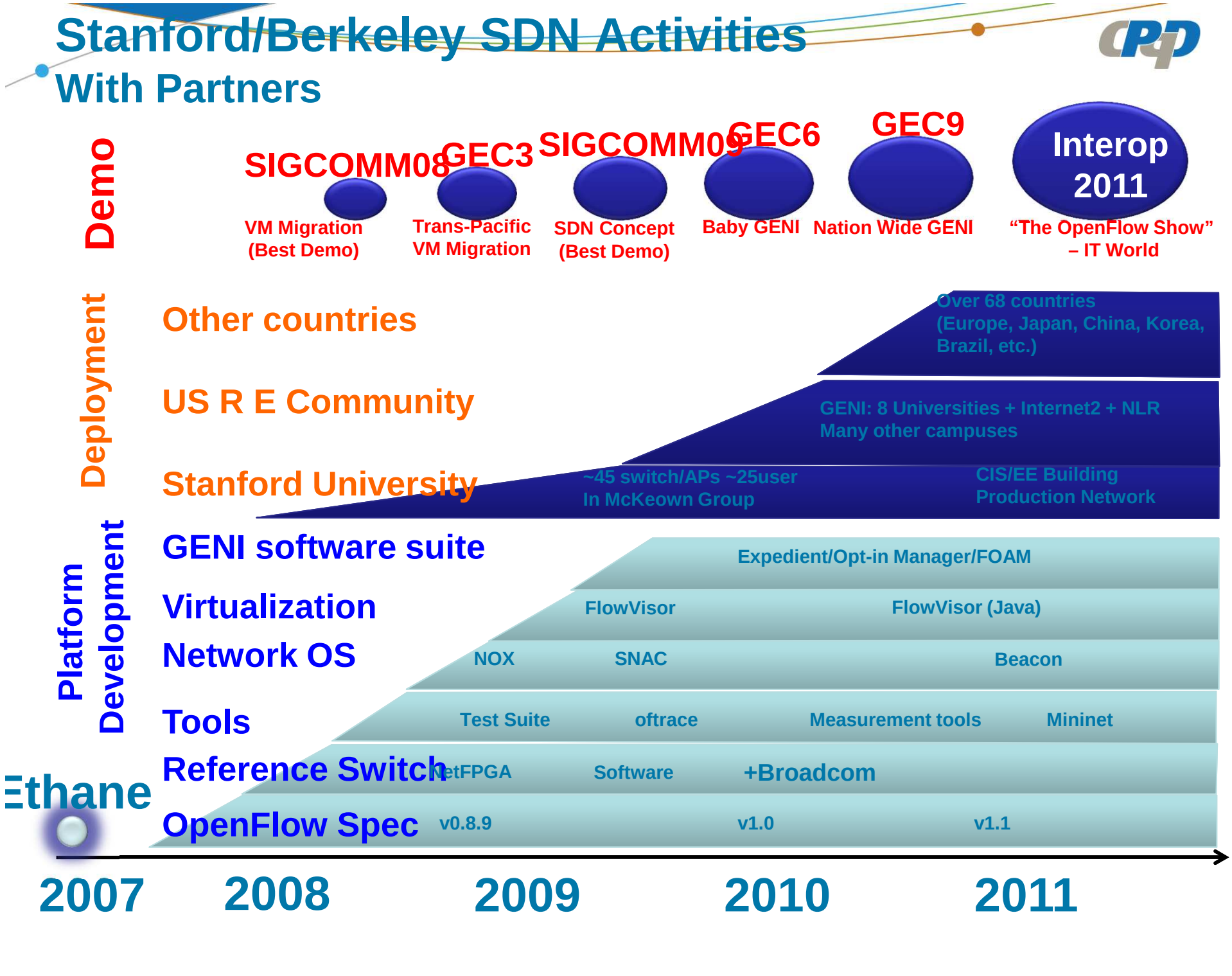
System Properties as Robustness

- **Reliability** is robustness to component failures
- **Efficiency** is robustness to resource scarcity
- **Scalability** is robustness to changes to the size and complexity of the system as a whole
- **Modularity** is robustness to structure component rearrangements
- **Evolvability** is robustness of lineages to changes on long time scales

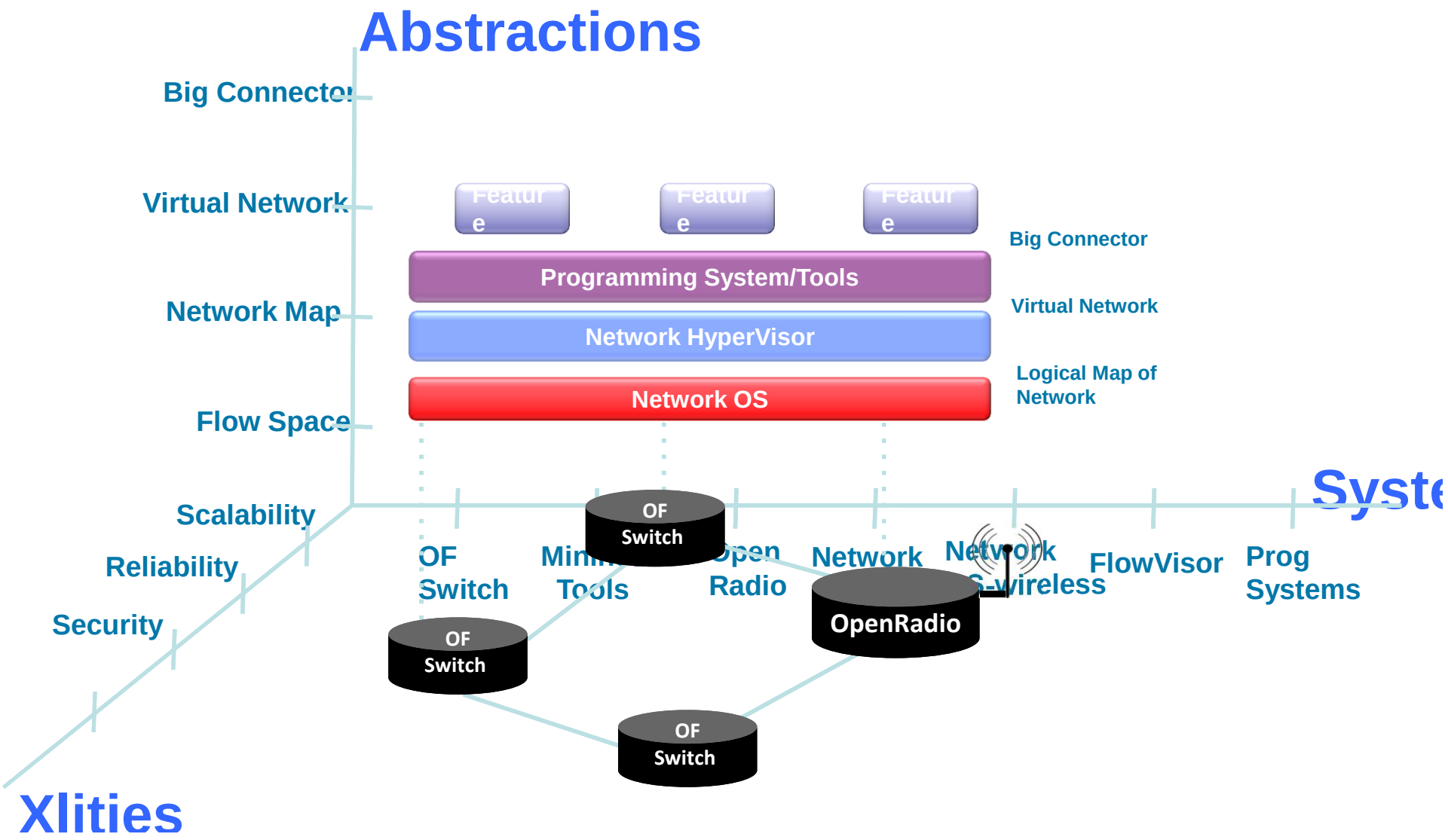
Stack View


Recursive
Services Layer (GOTOM, IM/Presence, Video, Mobility, ...)
APIs, Plugins, and Protocols
Cloud/Tenant Orchestration, Services, Management
APIs, Plugins, and Protocols
SP, Campus, and Data Center Orchestration
Overlays, VPNS, Network Slicing
Distributed Routing and Peering
APIs, Plugins, and Protocols
Virtual and Physical Forwarding Resources, Compute and Storage

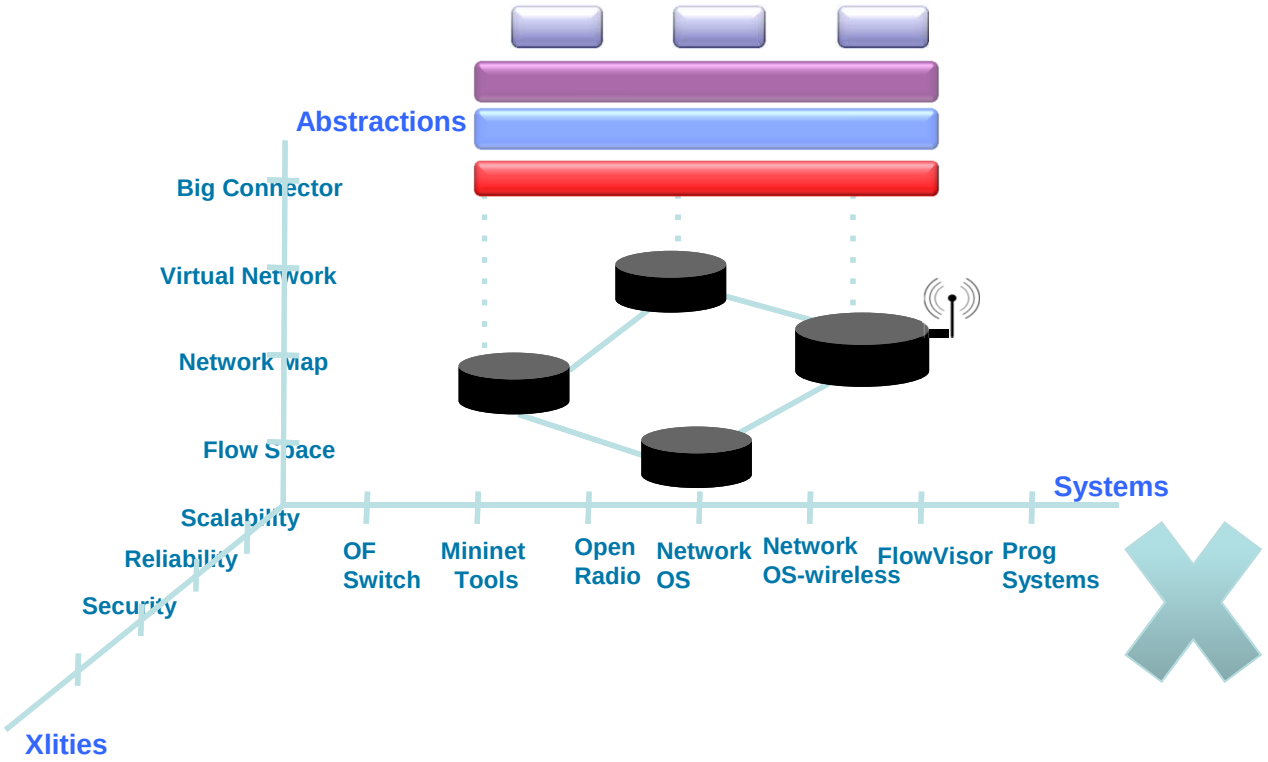
Stanford/Berkeley SDN Activities
With Partners



Scope of Activities



Scope of Activities Cont.



Domains of Use

- Enterprise Networks
- Datacenter Networks
- Service Provider Netwo
- Cellular Networks
- Home Networks

- Não existe uma disciplina de redes como existe na computação
- apenas uma cultura de protocolos de comunicações que são adicionados ou especializados a cada necessidade de nova funcionalidade ou correção, tornando as redes extremamente complexas.
- Isto explica porque até hoje não entendemos a dinâmica das redes, dos seus estados, do tráfego da Internet, porquê mal compreendemos como a Internet ainda funciona (ex: BGP ainda é um mistério).

- Cultura de remendos por meio de novos protocolos e novas funções é insustentável !!!

- Necessário definir/repensar a(s) disciplina(s) de redes

Desafios de Pesquisa

- Entendendo Redes e as redes
- Impactos dos requisitos atuais
- Impactos dos princípios de projeto atuais
- Impactos do modelo atual
- Impactos das arquiteturas atuais
- Impactos dos protocolos atuais
- Impactos das implementações atuais

Desafios de Pesquisa

- Repensando Redes:
  - Requisitos
    - Segurança, mobilidade, robustez, flexibilidade, liberdade de escolha, etc
  - Propriedades
    - Separação das preocupações, endereçamento lógico (nós, aplicações, objetos), recursividade, etc
  - Arquitetura
    - Orientação, separação dos planos de controle e encaminhamento, “end-to-end”, IPC, Overlays, Virtualização, etc
  - Abstrações
    - Comutação, hardware, da rede, topologia, políticas, configuração, grafos, compiladores, etc

Desafios de Ensino

- Pesquisadores atuais foram educados na disciplina corrente:
- Vícios dificultam outras formas de pensamento

- Professores do amanhã estão sendo educados na disciplina corrente:
- Vícios dificultam assimilação e outras formas de ensinamento

- Necessário encontrar uma forma que capte o interesse do aluno e facilite a absorção do conhecimento
- Teoria + Experimentação/Prototipagem (quando aplicável)

O desafio

- compreender essas dimensões,
- entender os porquês,
- entender que as redes não são mais entidades isoladas e sim parte de um sistema maior,
- em definir princípios e requisitos,
- em conceber um modelo e abstrações que caracterizem uma rede
- que tenham expressividade suficiente para nortear e suportar avanços científicos e desenvolvimentos tecnológicos do estado da arte,
- tal como tem ocorrido de forma explosiva na computação nas últimas décadas.

- Seguem como exemplo algumas questões explicitas que o desafio poderia abordar:
- Quais são os princípios que devem nortear o projeto e operação de uma rede?
- Mesmo um dos princípios amplamente aceitos, como é o do “fim-a-fim” (Saltzer, Reed, and Clark, 1981), não é propriamente um “principio” provável nem aplicável em todo contexto de rede, em especial sob os requisitos e objetivos das redes atuais, após mais de 3 décadas da sua formulação.

- Como conciliar os diferentes interesses conflitantes entre usuários e produtores de uma tecnologia de rede, ou seja, entre operadores ou usuários finais, e fabricantes de equipamento e desenvolvedores das soluções de software de controle?
- Um conflito semelhante acontece na operação de rede sob a óptica das múltiplas camadas. Por exemplo, considerando uma operadora de rede, as equipes da rede de “transporte” (comutação circuitos, SDH, WDM, OTN, etc.) e as equipes de “IP” (roteamento entre e intra-dominio) tem reduzida interação, existe pouca sinergia, o que resulta em operações ineficientes e custosas. Trate-se sim de um mal organizacional do empregador, mas que tem uma raiz na falta de princípios unificadores na disciplina de redes e que acaba se traduzindo numa especialização e fragmentação prejudicial para a indústria e ciência de redes como um todo.
- Analogamente, conforme discutido a seguir, existe uma

- Como reduzir o *gap* ou *separation of corncerns* ainda dominante entre computação e rede?
- A especialização dos cientistas e profissionais nessas duas áreas ou domínios de atuação tem ficado evidente com o surgimento da computação em nuvem e a necessidade de se definir novas arquiteturas de datacenter que virtualizem e orquestrem o controle dinâmico dos recursos de (1) computação, (2) de armazenamento e (3) de rede de forma holística e integrada.

- Como difundir na área de redes e como aplicar os conhecimentos e soluções de software gerados na ultima década para problemas de sistemas distribuídos e bases de dados para contribuir ao projeto e desenvolvimento dessas novas abstrações para o plano de controle de rede?
- De novo, uma abordagem multi-disciplinar é chamada para conseguir transformar o problema de redes em uma ciência / disciplina de redes como existe na computação

Impacto no Ensino

- A partir deste ponto, outro desafio passa a ser ensinar os alunos a pensar em redes sob esta perspectiva e não mais sob a dos protocolos.
- Faz-se necessário definir uma disciplina de redes correspondente que seja capaz de capturar o interesse dos alunos e de ser absorvida por eles.
- Esta disciplina deve ensinar princípios, abstrações e modelos de forma teórica e prática, formando alunos capazes de pensar e de produzir a inovação e de criar uma indústria pujante de redes de que o país tanto precisa.