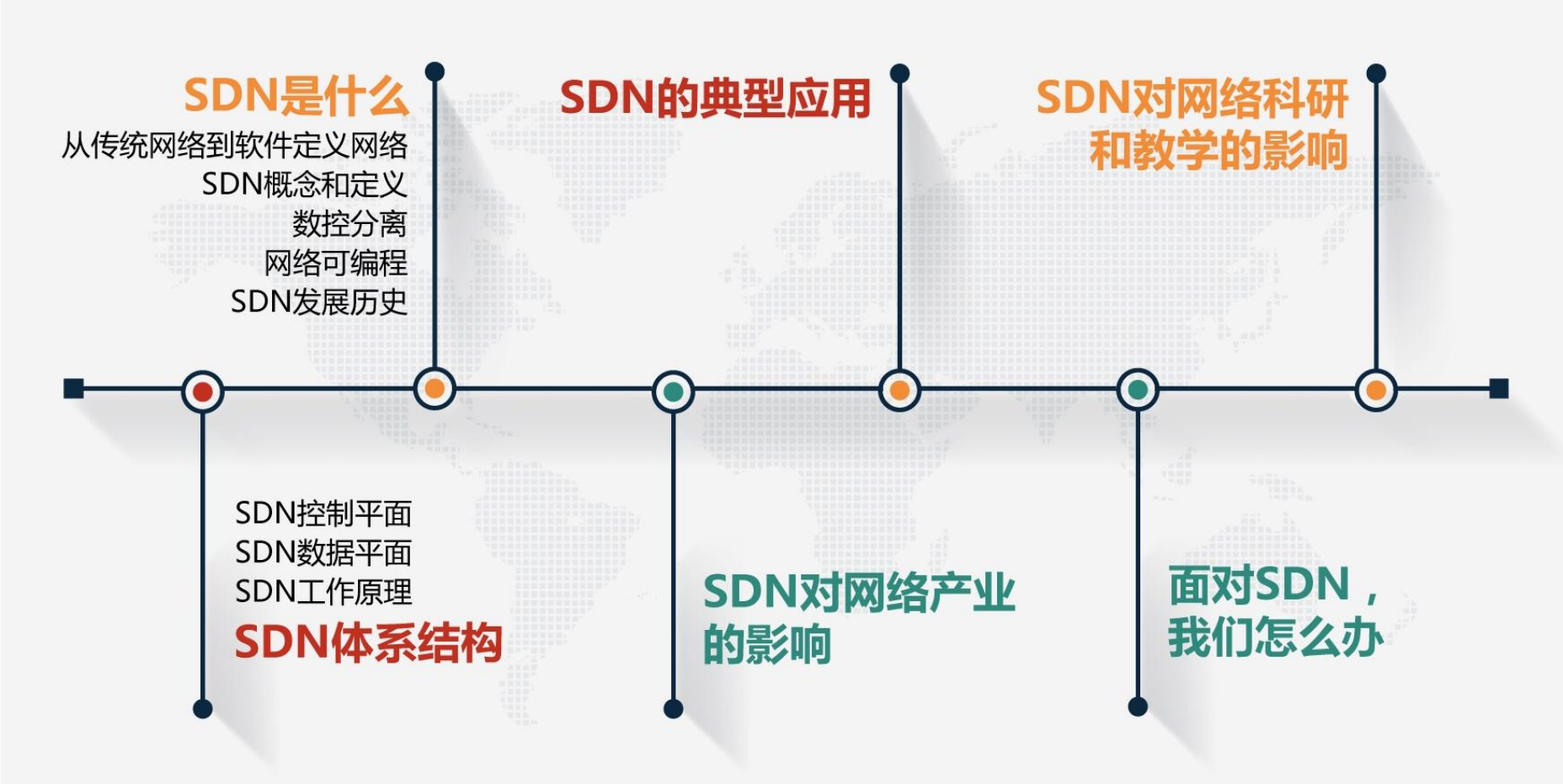
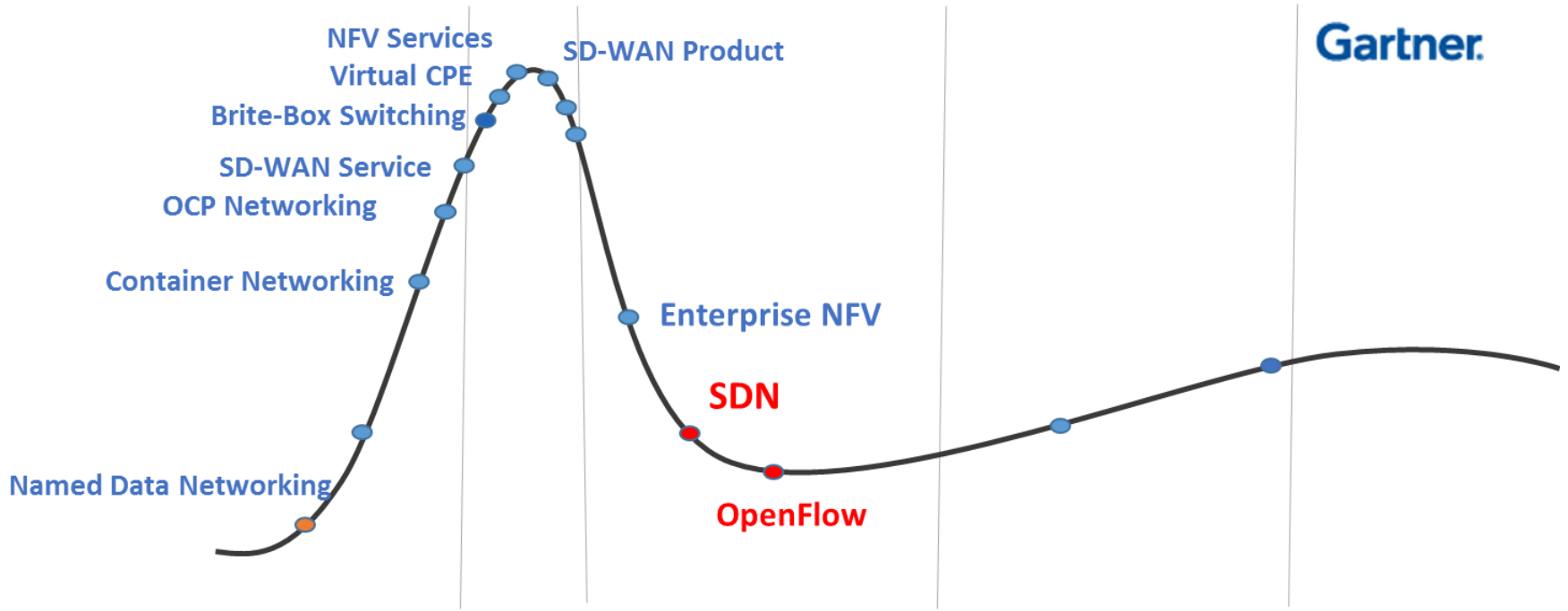
SDN前世今生

两小时，让你真正读懂SDN

未来网络学院
Future Network Institute

我们学习哪些内容？


SDN是什么
从传统网络到软件定义网络
SDN概念和定义
数控分离
网络可编程
SDN发展历史
SDN的典型应用
SDN对网络科研
和教学的影响
SDN控制平面
SDN数据平面
SDN工作原理
SDN体系结构
SDN对网络产业
的影响
面对SDN，
我们怎么办


Gartner
NFV Services
Virtual CPE
SD-WAN Product
Brite-Box Switching
SD-WAN Service
OCP Networking
Container Networking
Enterprise NFV
SDN
Named Data Networking
OpenFlow

第一部分 - SDN是什么？

SDN之前的网络架构

交换机，路由器，防火墙，VPN，NAT，OSPF，BGP，Traffic Engineering ...

网络功能　网络功能

厂商操作系统

数百万行封闭的软件源码　　6,000 RFCs

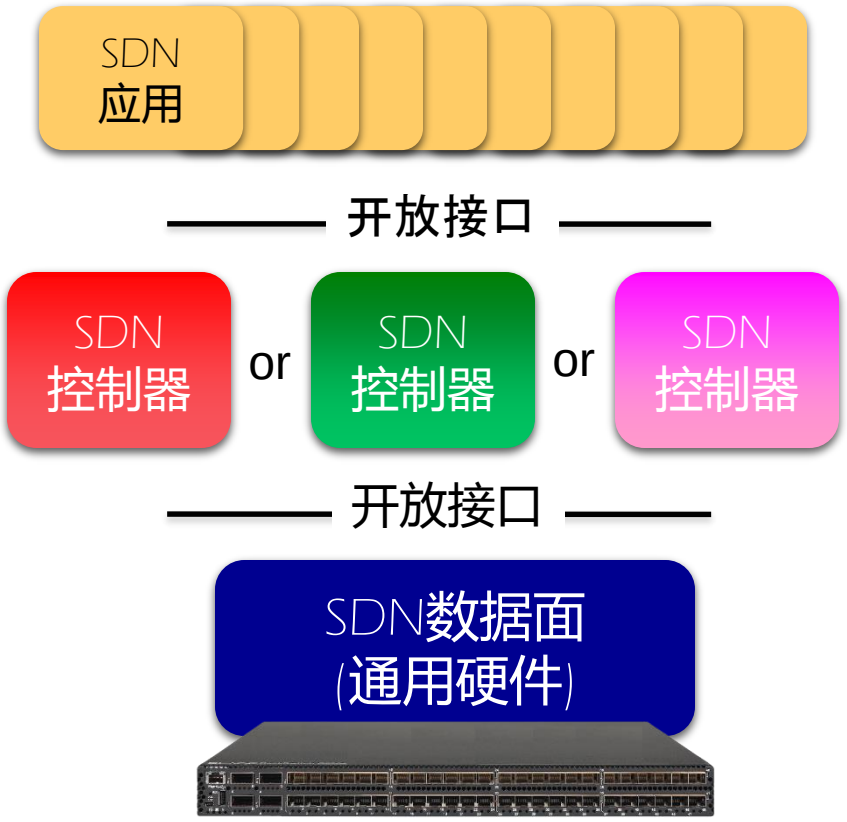
定制化硬件

不可移植　　厂商专有API

- 垂直集成的封闭系统：厂商负责制
- 网络功能的简单堆砌：一个问题，一个协议
- 行业创新基本停滞：设备厂商独家话语权

Source: Nick Mckeown

从传统网络到SDN



封闭的架构
早期的IBM大型机
问题 - 协议 - 实现

开放的架构
Intel + linux + APP
问题 - 应用

Source: Nick Mckeown

两个权威组织的SDN定义

SDN architecture **decouples** the network control and forwarding functions enabling the network control to **become directly programmable** and the underlying infrastructure to be **abstracted** for applications and network services. The **OpenFlow®** protocol is a foundational element for building SDN solutions.

ON.LAB

SDN's key attributes include: **separation of the data and control planes**; **a uniform, vendor-agnostic interface**(called OpenFlow) between control and data planes; **a logically centralized control plane** that offers a consistent, network-wide view to programmers or operators.

大众认可的定义

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

- SDN is meant to address the fact that the static architecture of traditional networks doesn't support the **dynamic**, scalable computing and storage needs of more modern computing environments such as data centers.

- This is done by **decoupling** or disassociating the system that makes decisions about where traffic is sent (the SDN controller, or control plane) from the underlying systems that forward traffic to the selected destination (the data plane).

一段话描述SDN

SDN是一种新的网络体系结构，给传统网络带来最大的改变是网络可编程和开放性。网络用户追逐SDN的关键是想获得更多的网络可编程能力，获得更多的网络定制开发能力和自主权。SDN的开放分层架构加速了网络产业的参与度，越来越多的网络用户、网络软件公司和初创公司都加入到网络产业中来，这种开放竞争进一步加速整个产业的创新。

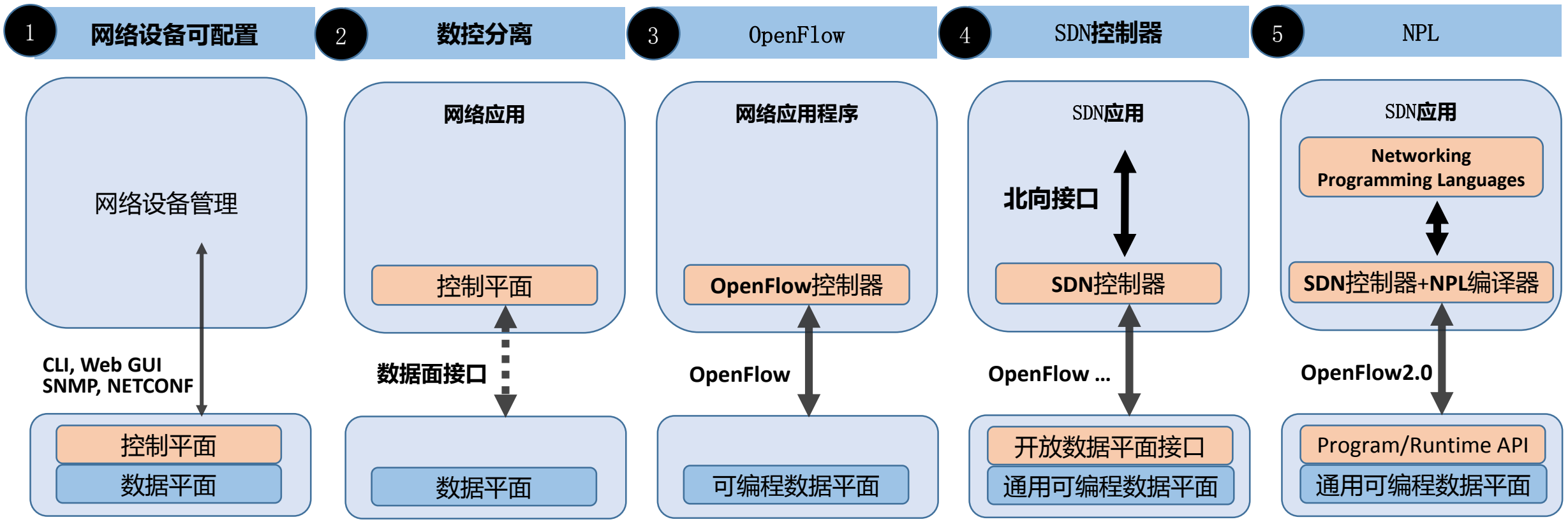
SDN的解读

网络开放可编程

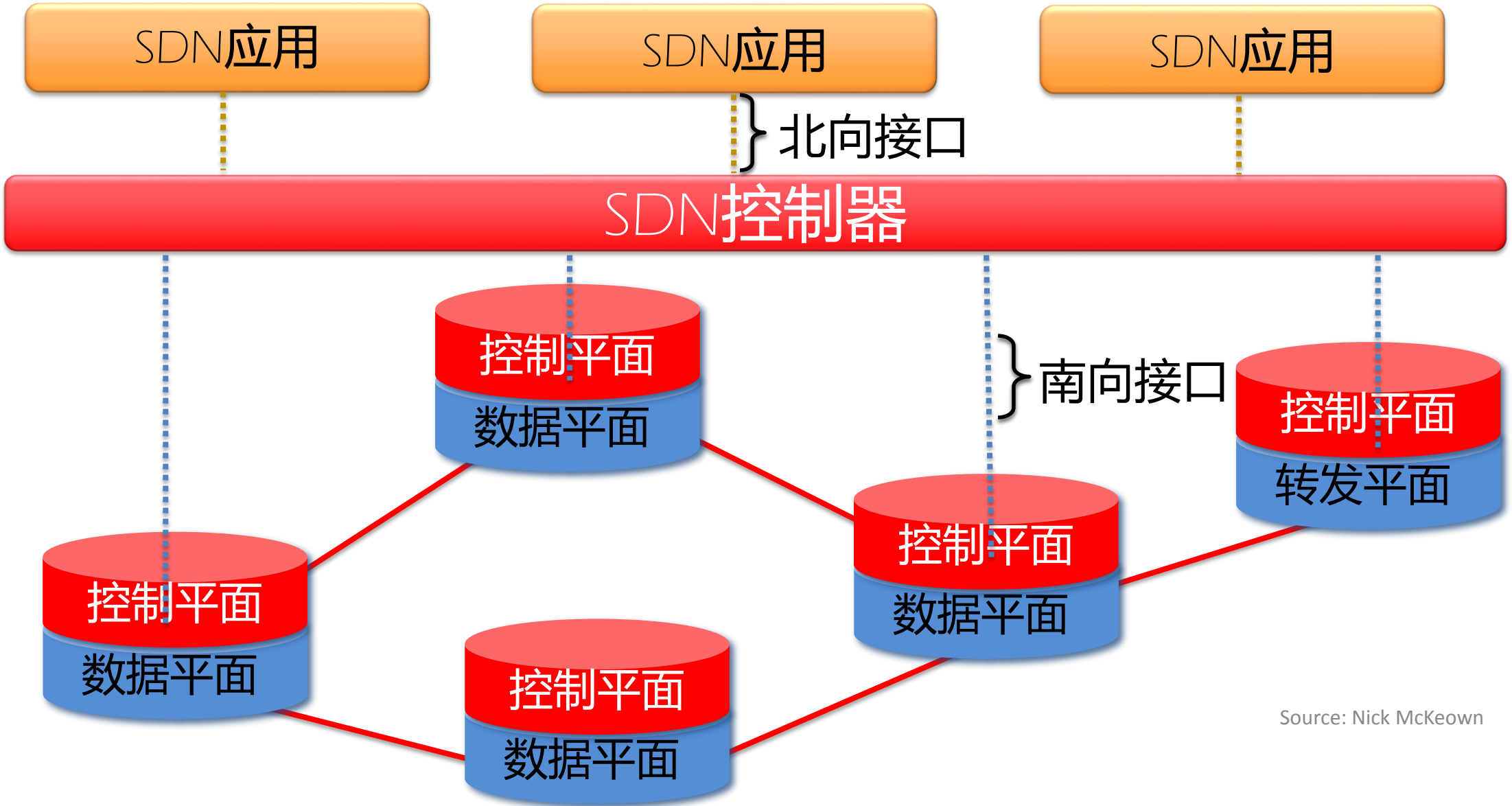
数控分离

逻辑上集中控制

网络开放可编程

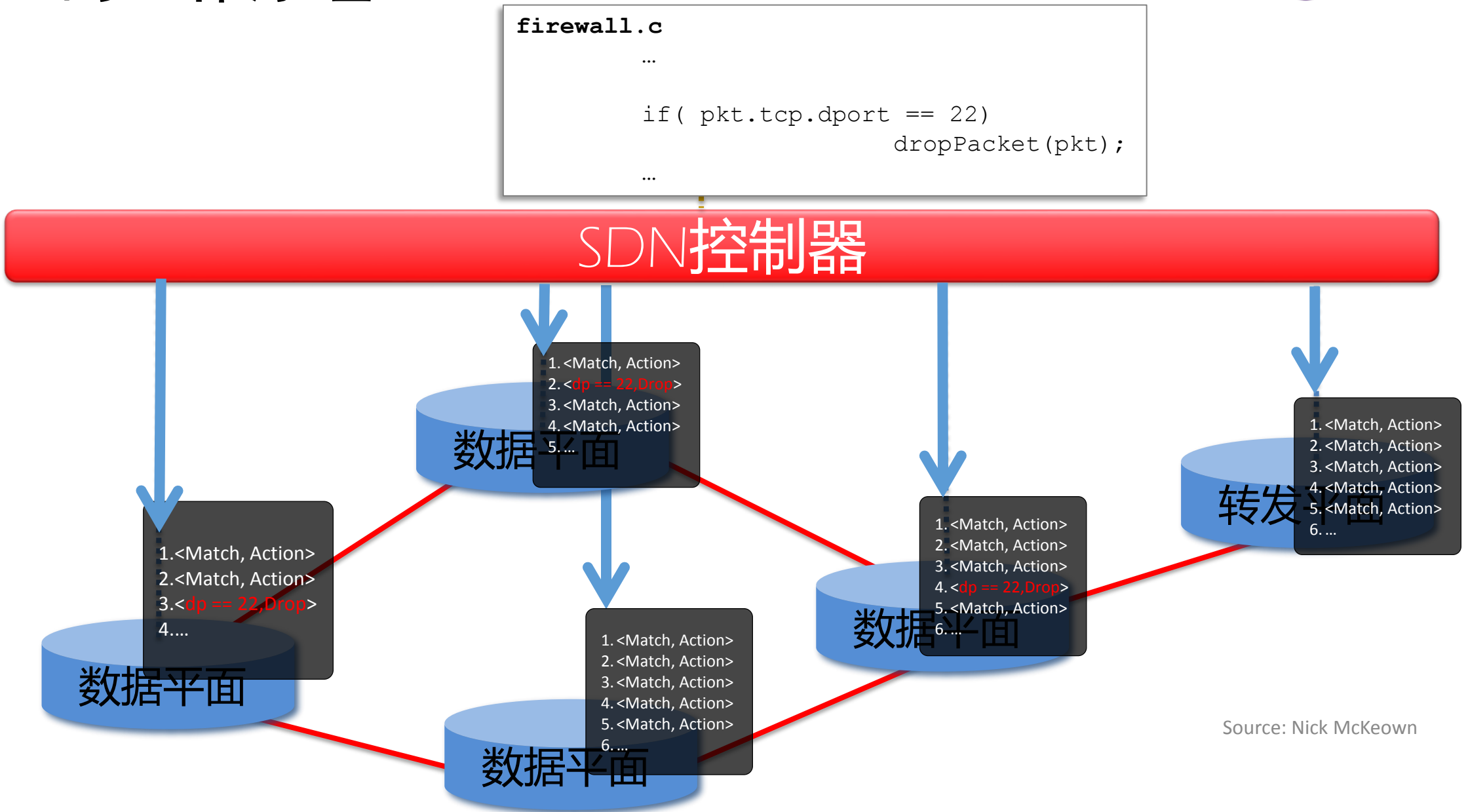


SDN的分层体系结构



Source: Nick McKeown

SDN的工作原理



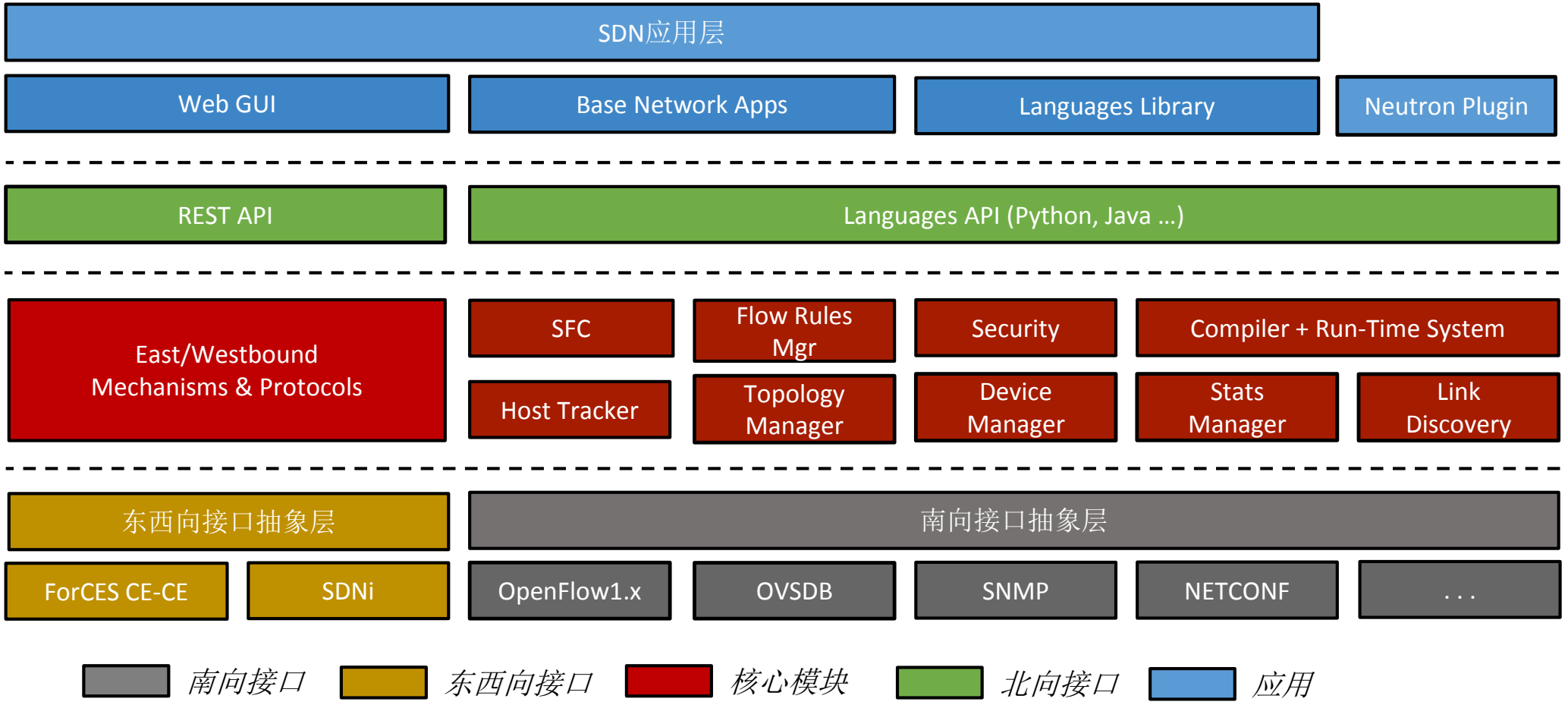
Source: Nick McKeown

SDN数据平面

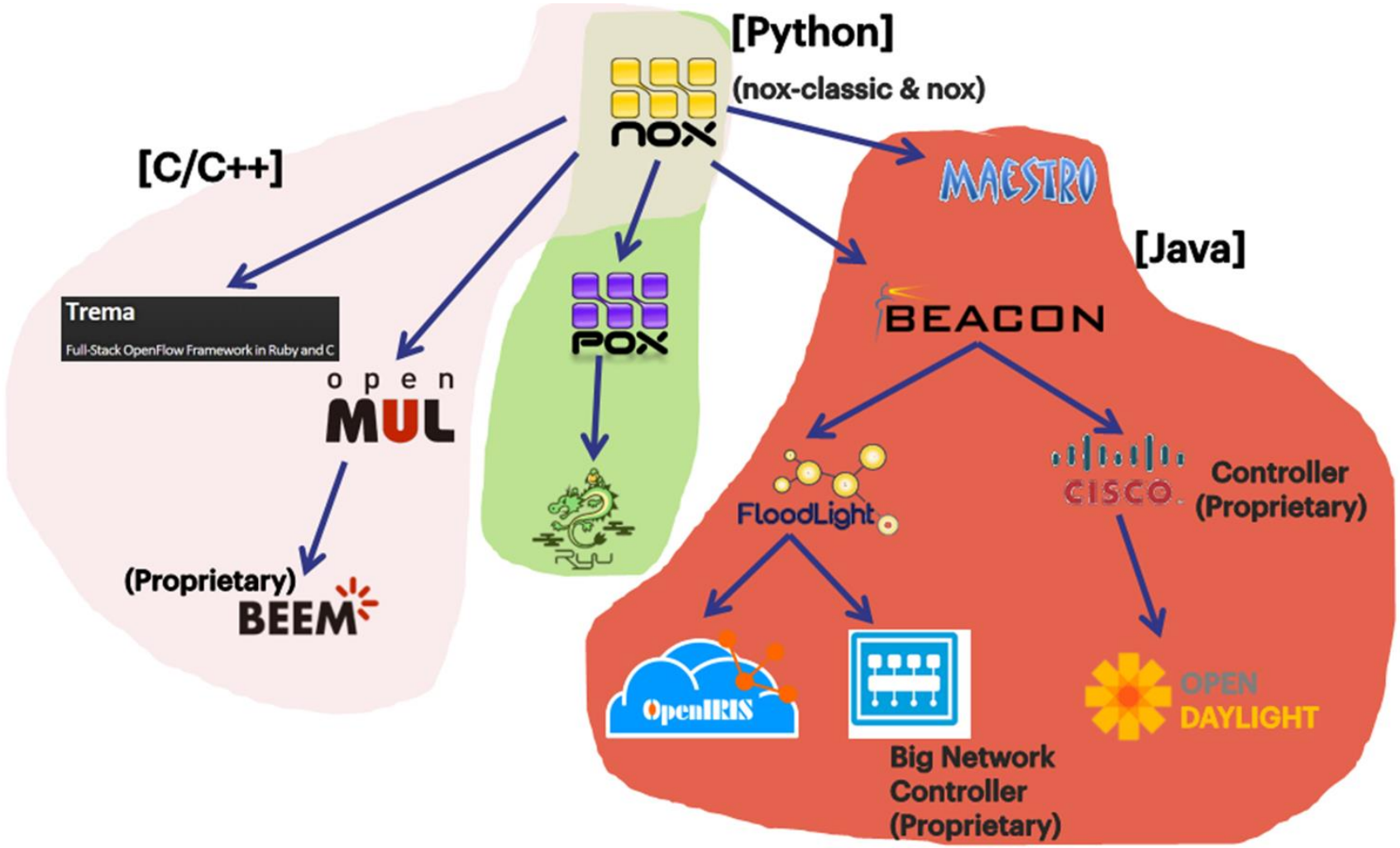
通用可编程网络数据平面

Description	MAC SRC	MAC DST	IP SRC	IP DST	TCP DPORT	Action	Counter
L2 Switching	*	10:20	*	*	*	output	250
L3 Routing	*	*	*	5.6.7.8	*	output	300
Micro-flow	*	*	*	*	25	drop	892
Macro-flow	*	*	*	192.*	*	local	120
Firewall	*	*	*	*	*	Controller	11

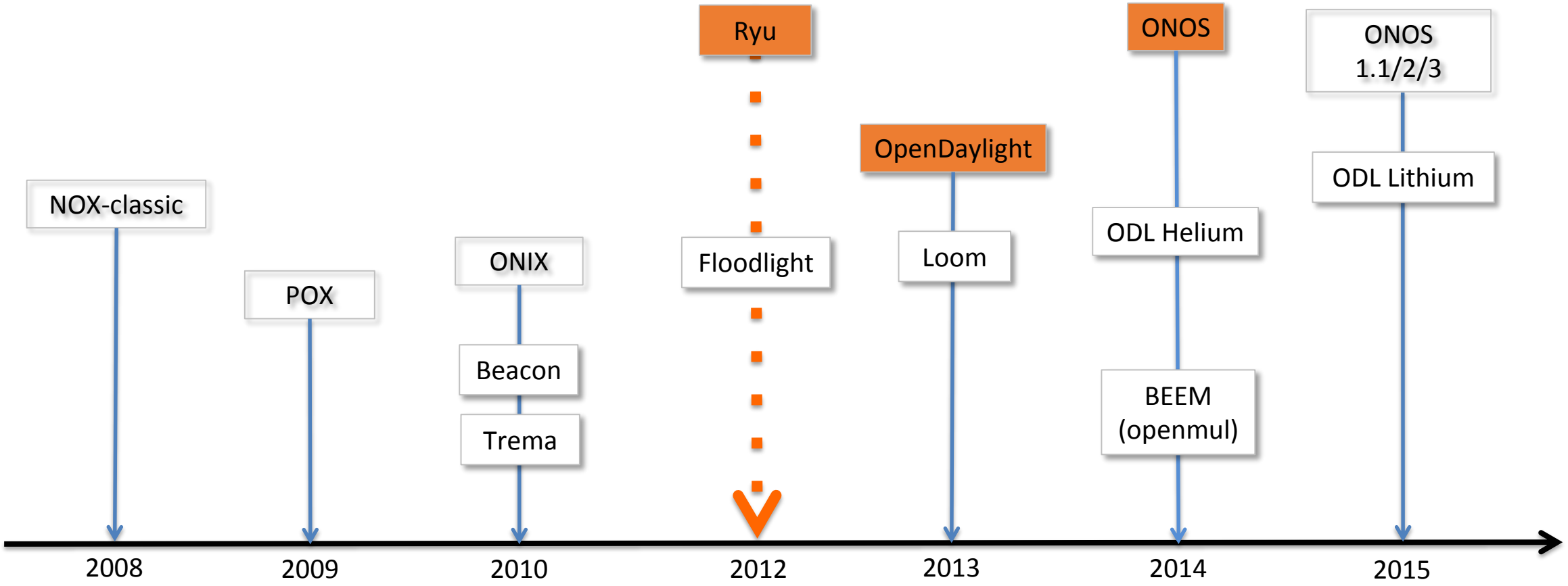
SDN控制平面


SDN应用层
Web GUI
Base Network Apps
Languages Library
Neutron Plugin
REST API
Languages API (Python, Java ...)
East/Westbound Mechanisms & Protocols
SFC
Flow Rules Mgr
Security
Compiler + Run-Time System
Host Tracker
Topology Manager
Device Manager
Stats Manager
Link Discovery
东西向接口抽象层
南向接口抽象层
ForCES CE-CE
SDNi
OpenFlow1.x
OVSDB
SNMP
NETCONF
. . .
南向接口
东西向接口
核心模块
北向接口
应用

SDN控制平面发展历史



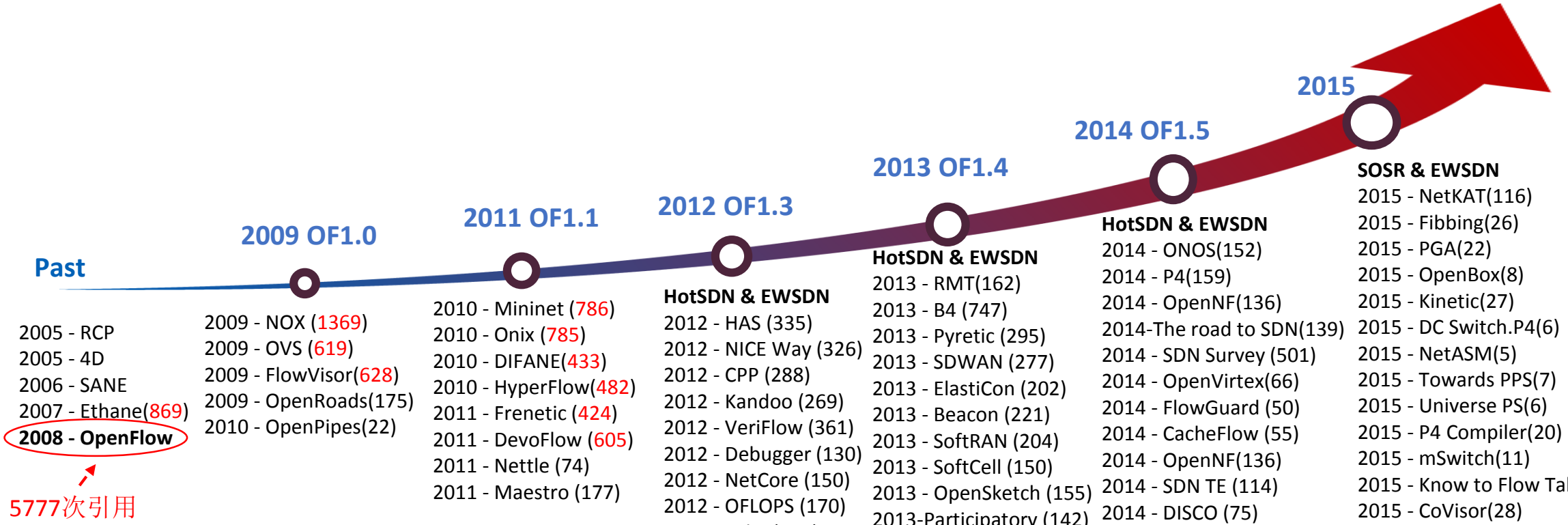
SDN控制平面发展历史



开源SDN控制器比较

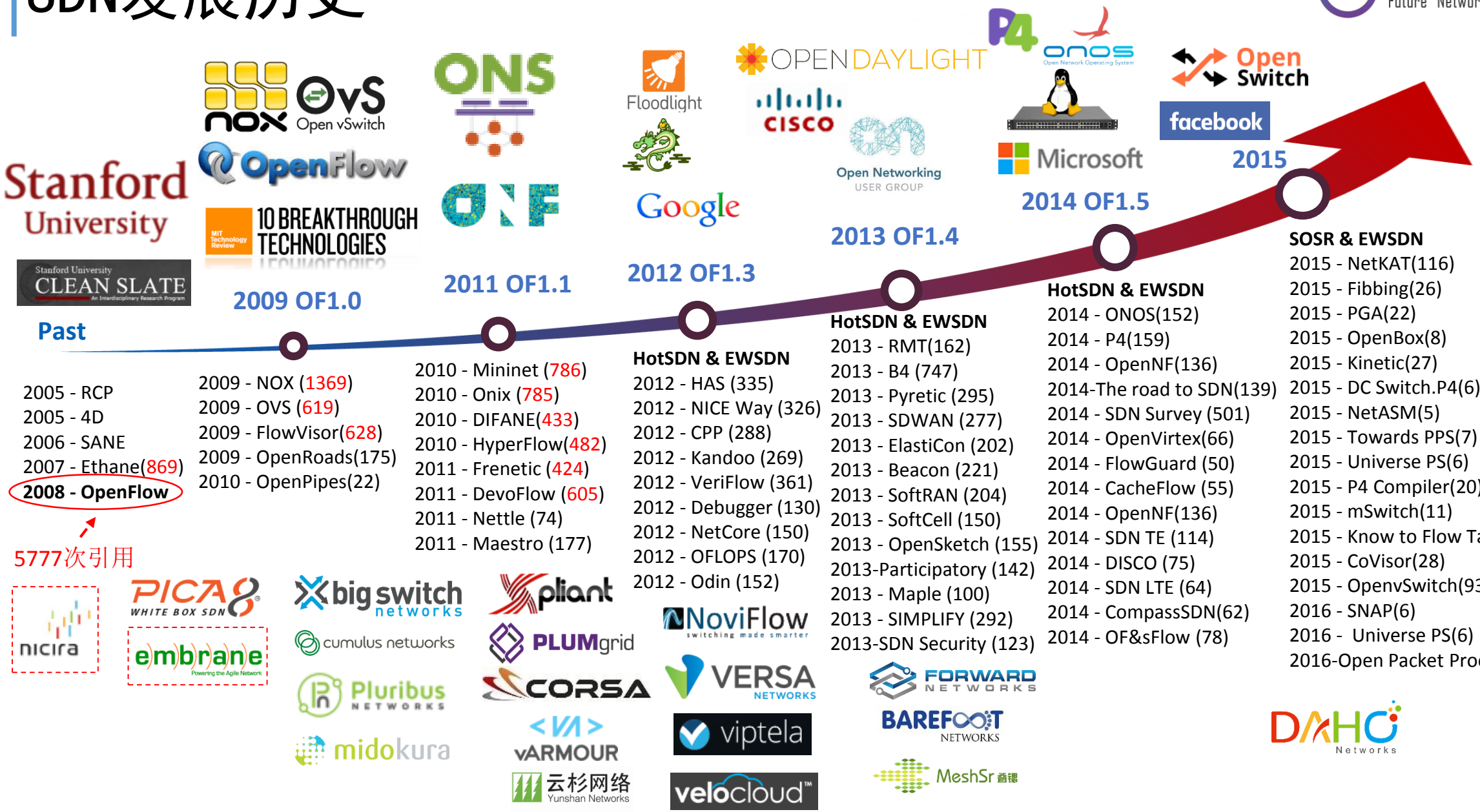
	NOX (2008)	**Ryu** (2012)	**Floodlight** (2012.1)	**ONOS** (2014.12)	**ODL** (2013.12)
系统架构	Centralized multi-threaded	Centralized multi-threaded	Centralized	Distributed	Distributed
北向接口	C++ API	REST API	RESTful API/Java API	RESTful API	**REST/RESTCONF/Java**
编程语言	C++/Python	Python	Java	Java	Java
管理	--	CLI	Web UI	Web GUI/CLI	Web GUI/CLI
南向接口	OpenFlow	Full	OpenFlow	Full	Full
OpenStack支持	No	Yes	Yes	--	Yes
OpenFlow支持	v1.0	**v1. {0,2,3,4,5 } OF++**	V1. {0,3 }	v1.0	v1. {0,3 }
一致性	No	No	No	Strong	Weak
容错能力	No	No	No	Yes	No
学习曲线	Medium	**Easy**	Medium	Medium	Hard
License	GPLv3	Apache 2.0	Apache	Apache 2.0	EPL v1.0
社区活跃度	28 / 6	2460 / 57	2511 / 53	4453 / 55	**6384 / 60**

SDN发展历史



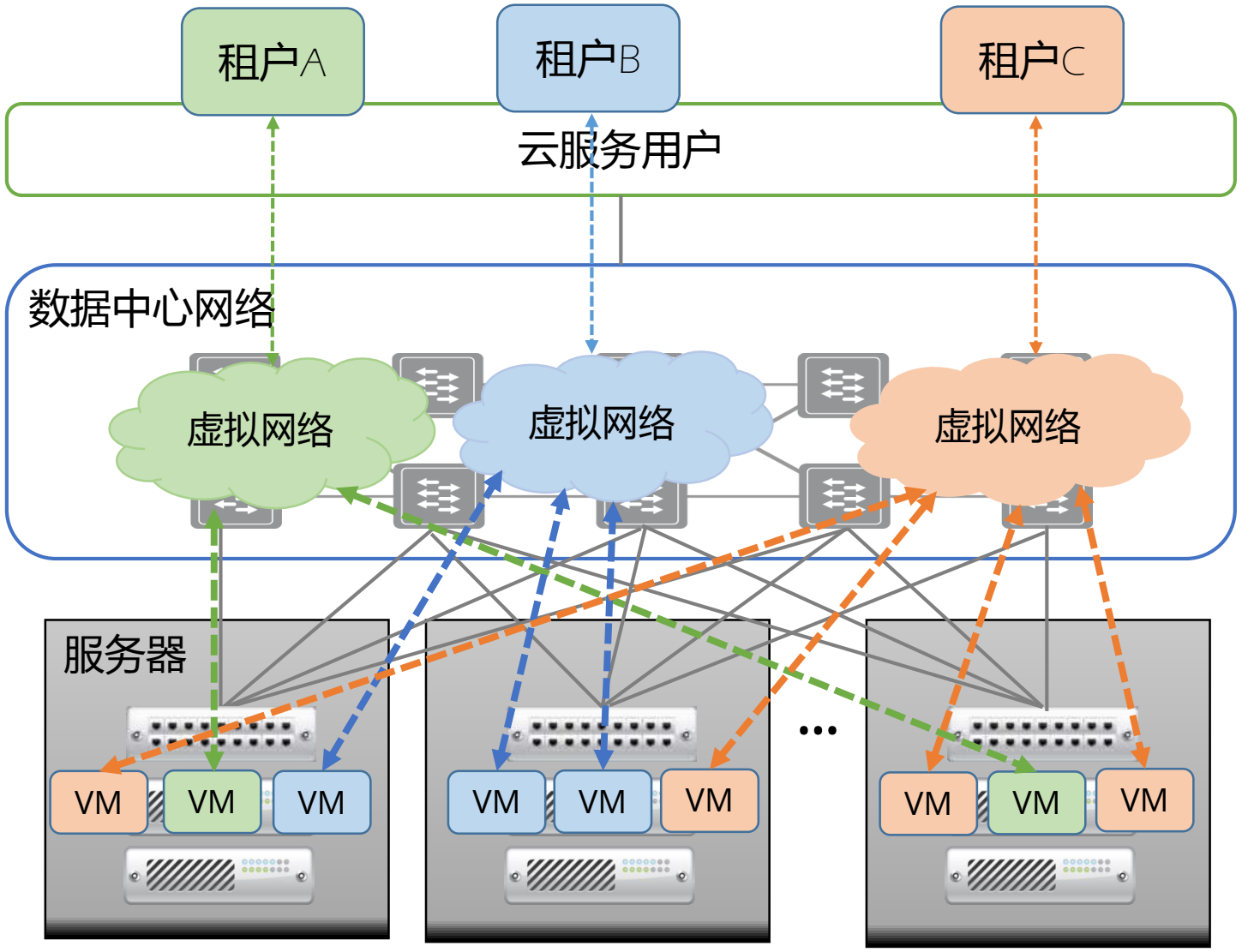
SDN发展历史

未来网络学院 Future Network Institute

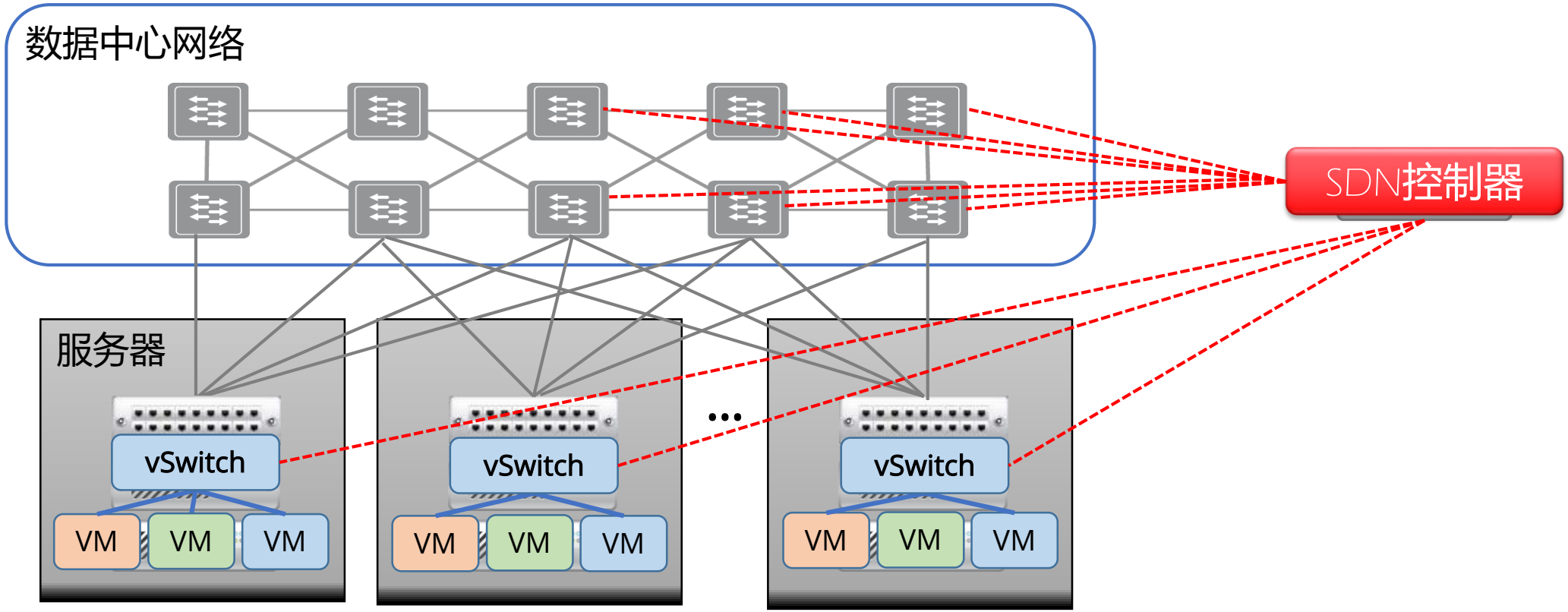


第二部分 - SDN典型应用

多租户网络需求

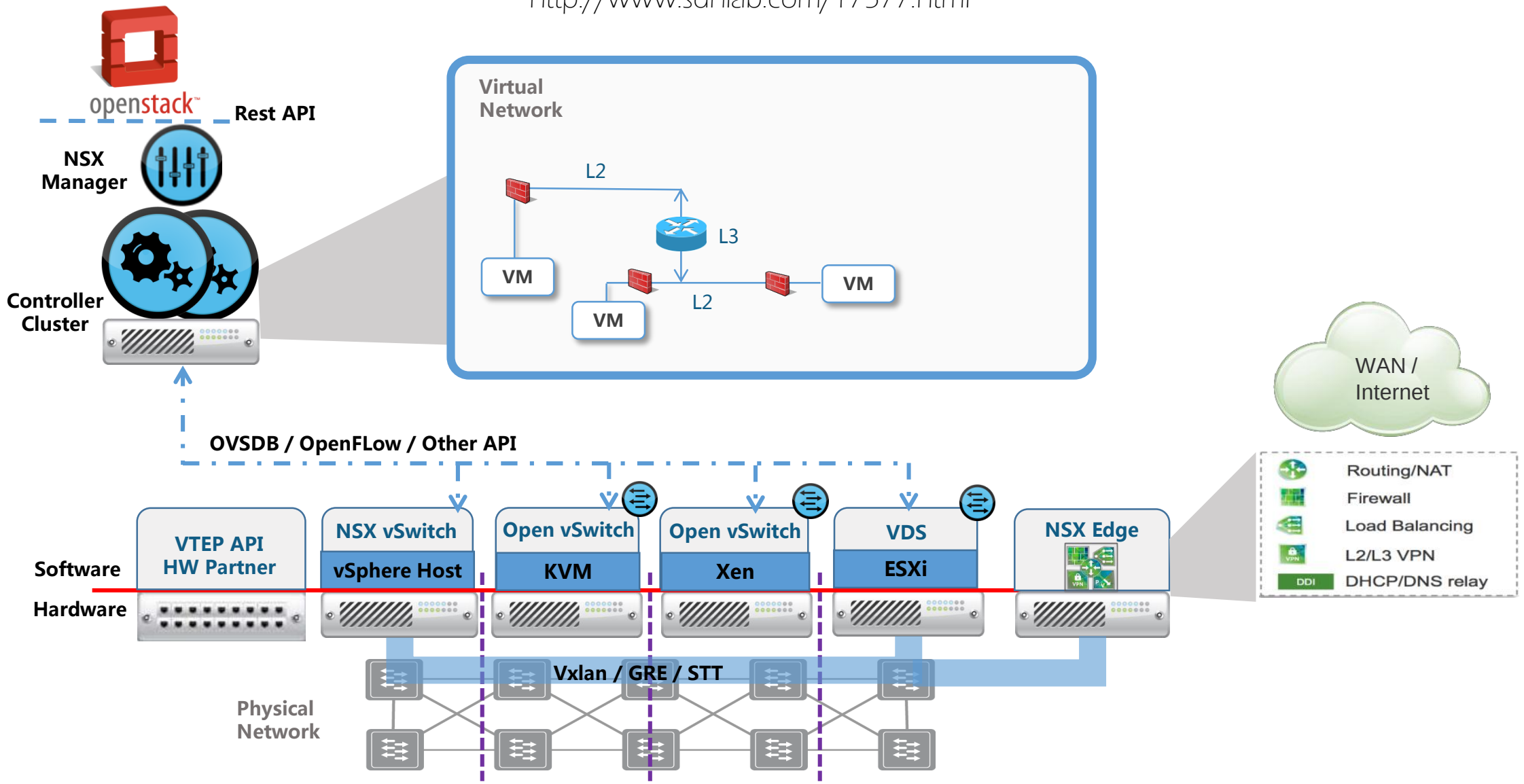


SDN在数据中心网络的应用

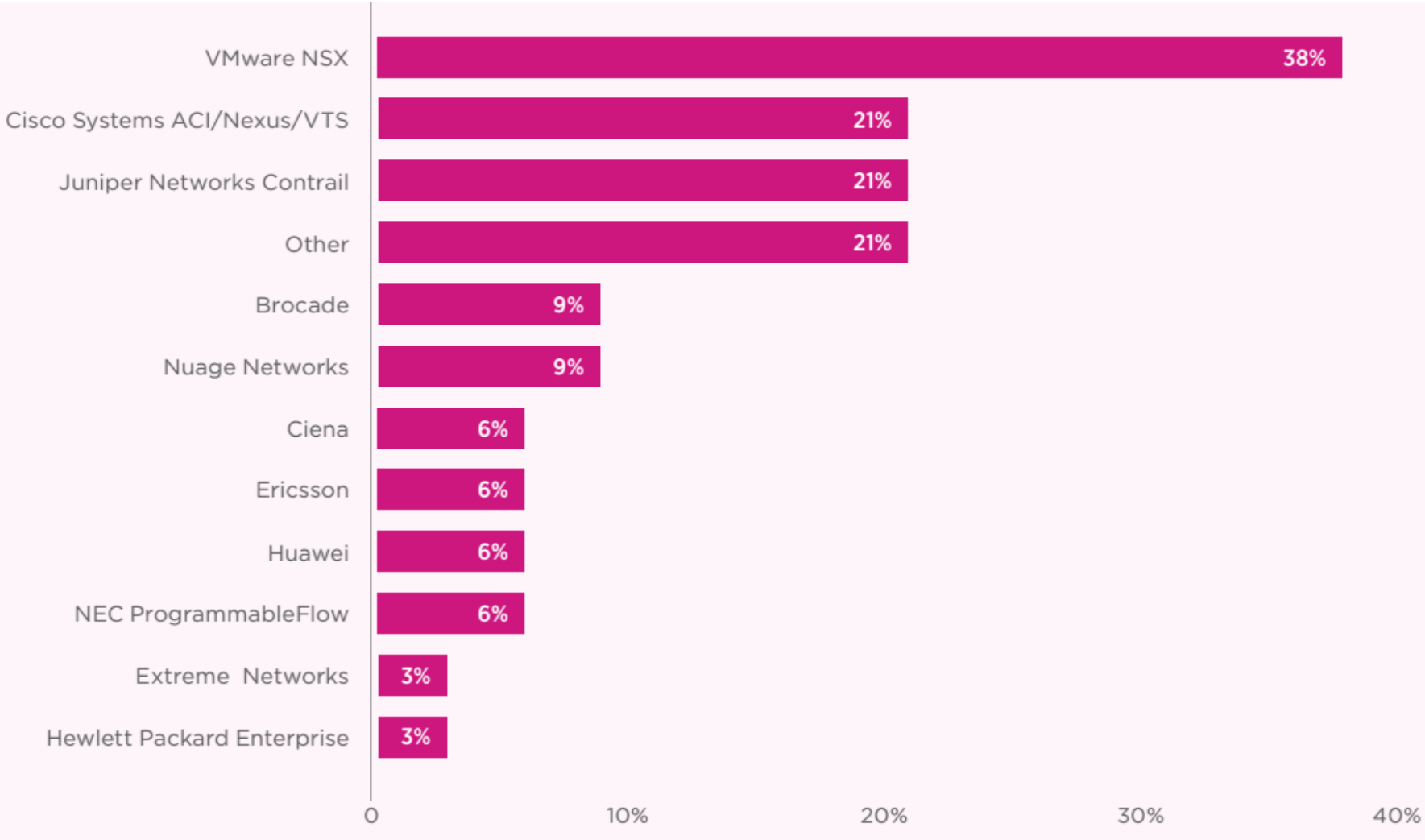


VMware NSX方案

SDN实战团分享（二十九）：VMware NSX技术分享:
http://www.sdnlab.com/17577.html



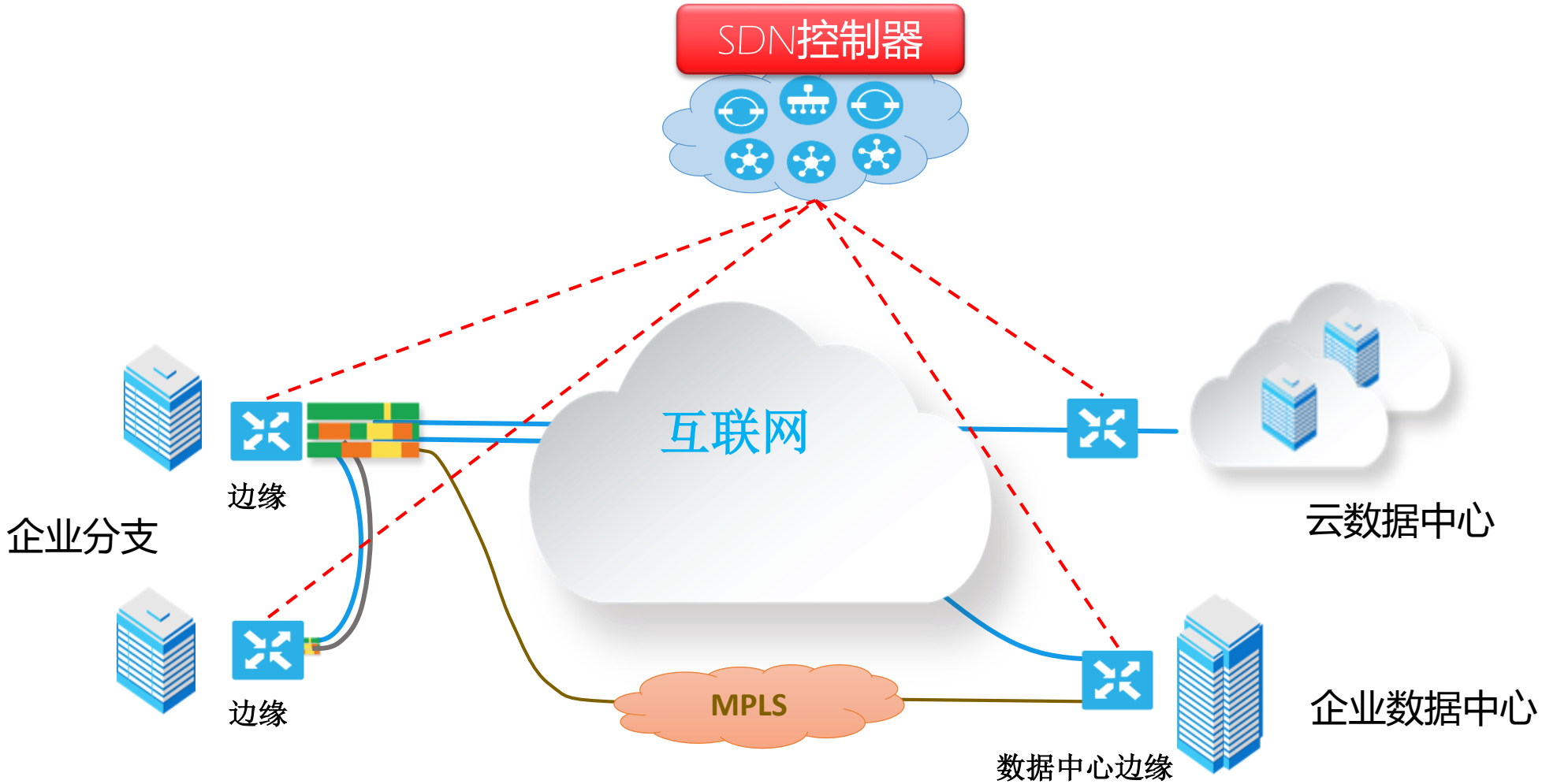
方案比较



方案比较

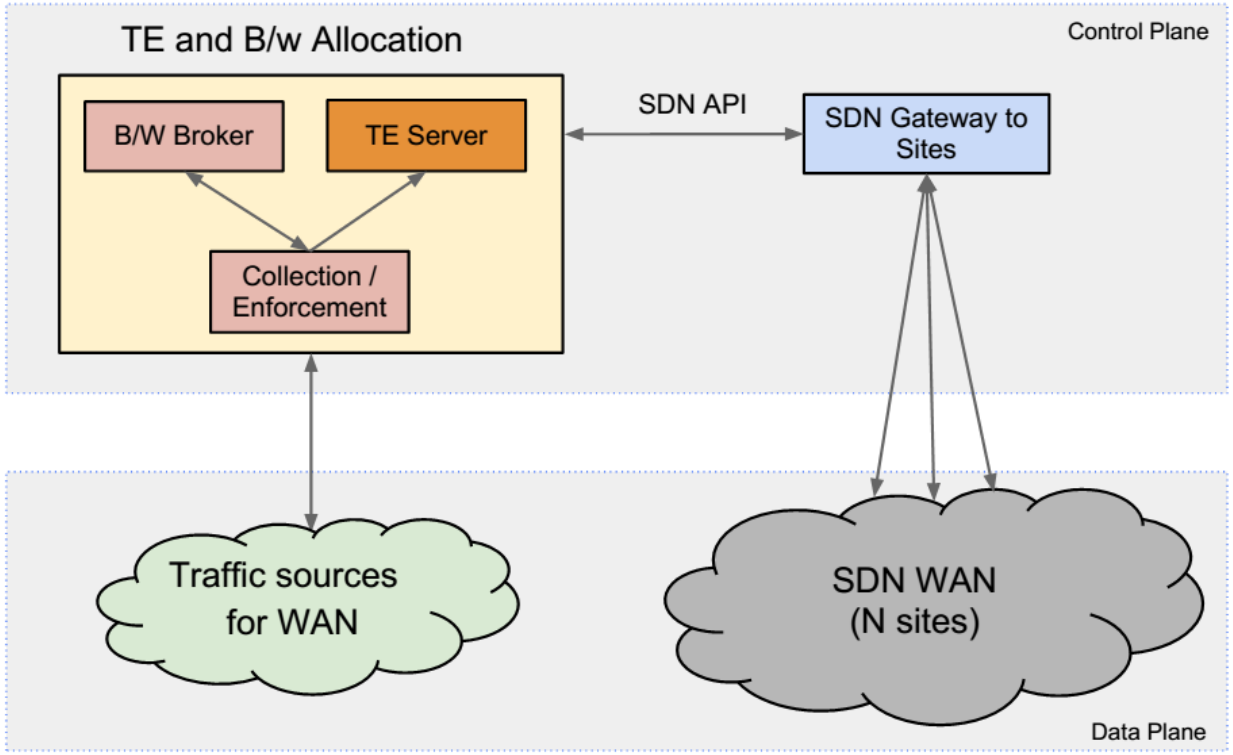
	VMware NSX	Cisco ACI
SDN控制器	NSX控制器集群	APIC
隧道协议	VXLAN, GRE, STT	VXLAN, NVGRE
南向接口	OpenFlow, OVSDB Vendor-specific RESTful API	OpFlex
软硬件集成	specific L2 Gateway integrations with leading ToR	through Cisco APIC Nexus 1000V
网络编程	NSX API，支持 L2-L7虚拟化网络编程	支持Python API， 可通过REST API来配置Cisco APIC控制器
优势	网络硬件无关	物理硬件集成了策略驱动

软件定义广域网-SDWAN

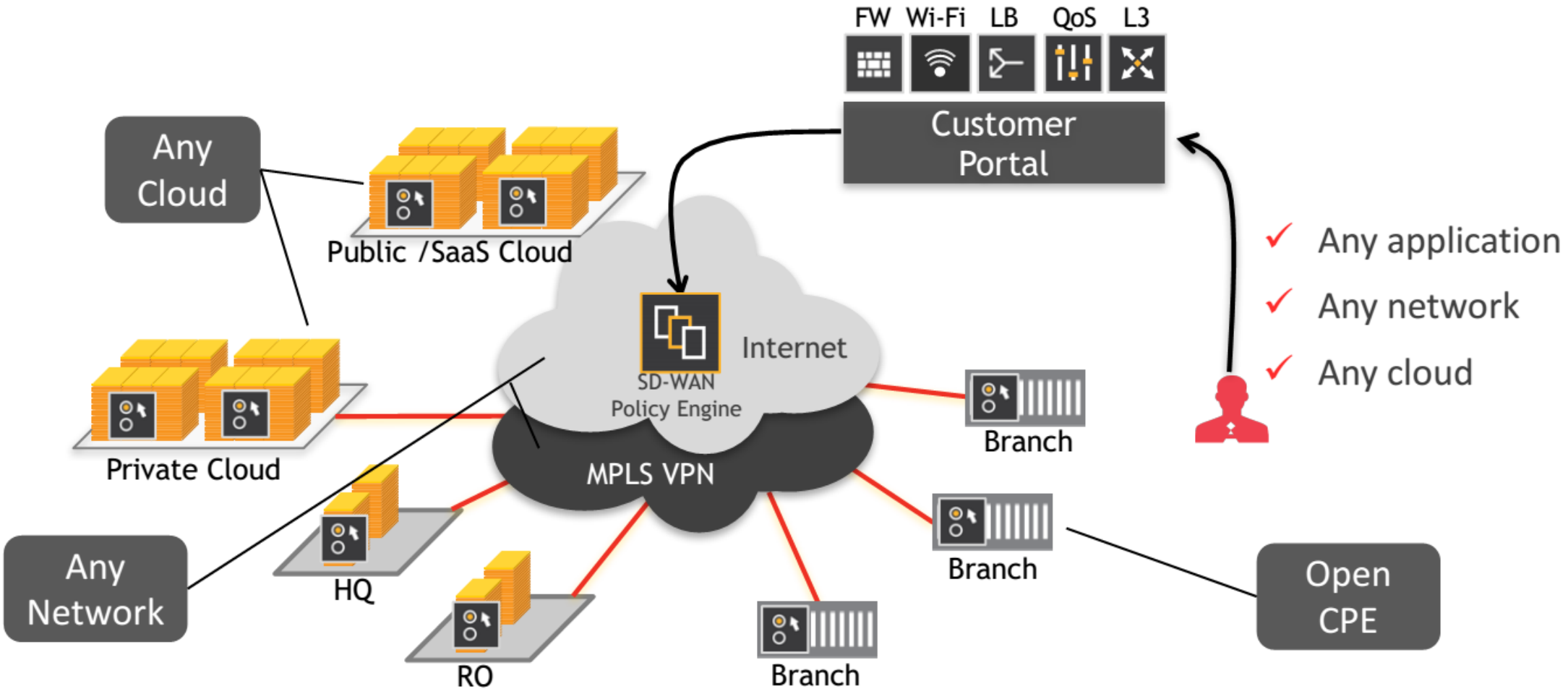


Google B4案例

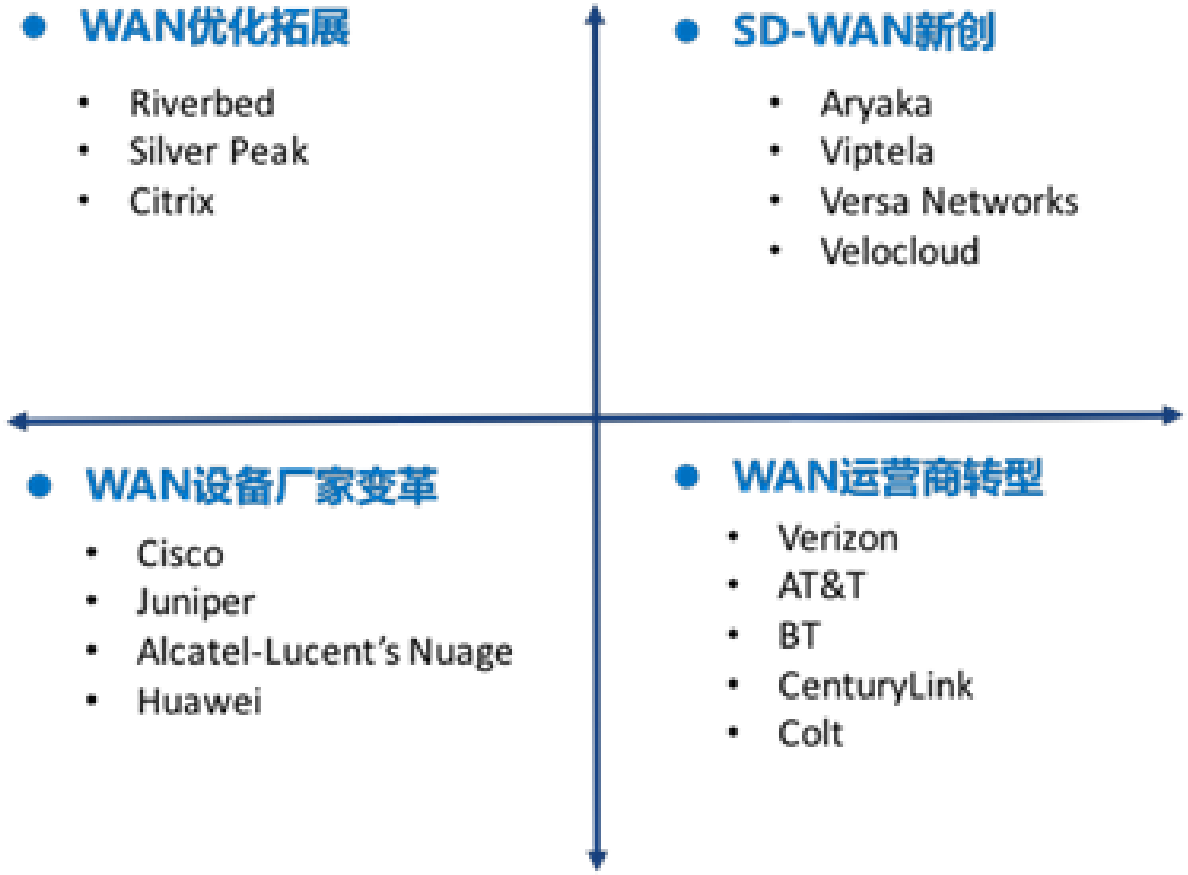
- 早期规模商用案例
- 两个意料之外
- 采用OpenFlow
- 部署和完善经历了四个阶段
- 链路利用率从30%~40%提升到90%



Nuage Network方案



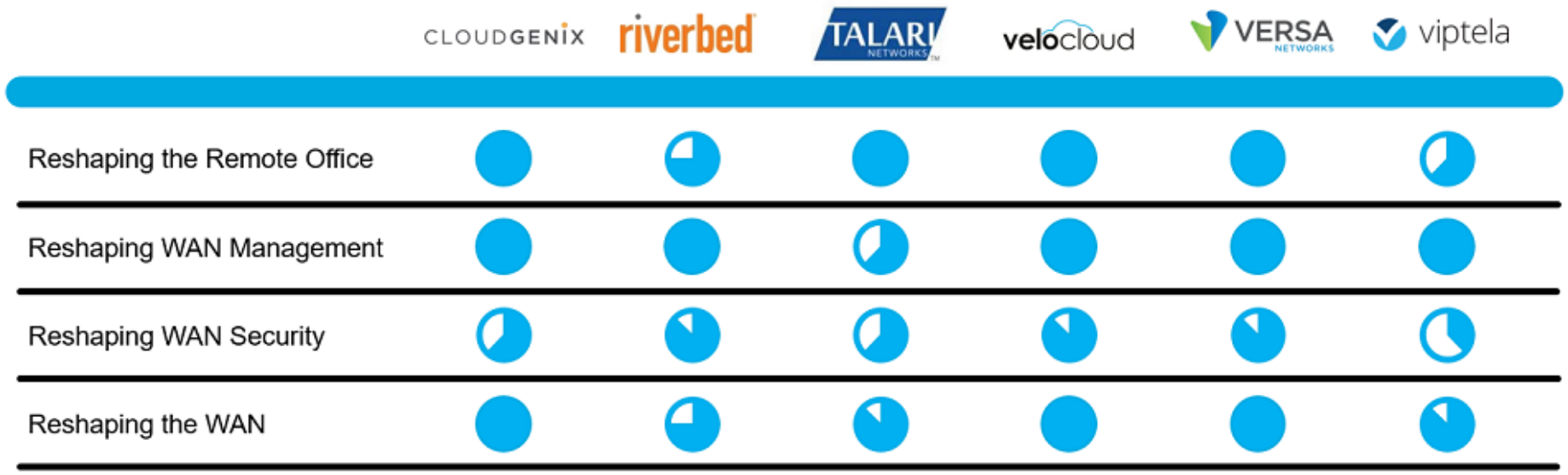
方案比较


WAN优化拓展
Riverbed
Silver Peak
Citrix
SD-WAN新创
Aryaka
Viptela
Versa Networks
Velocloud
WAN设备厂家变革
Cisco
Juniper
Alcatel-Lucent's Nuage
Huawei
WAN运营商转型
Verizon
AT&T
BT
CenturyLink
Colt

方案比较

运营商	Startup合作伙伴	提供服务
AT&T	VeloCloud	将VeloCloud的网关设备集成到AT&T的MPLS网络
	Riverbed（早期合作）	virtual WAN accelerator
Verizon	Viptela	将Viptela的管理平台纳入Verizon的WAN平台
	Versa Networks	Virtual customer premises equipment (vCPE) Security as-a-service
	Riverbed（早期合作）	virtual WAN accelerator
BT	Nuage	Virtualized Network Services (VNS)
	Cisco	Cisco IWAN
CentryLink	Versa Networks	可管理的SD-WAN 服务，软件定义安全服务
DT	VeloCloud	将VeloCloud的网关设备集成到MPLS网络环境

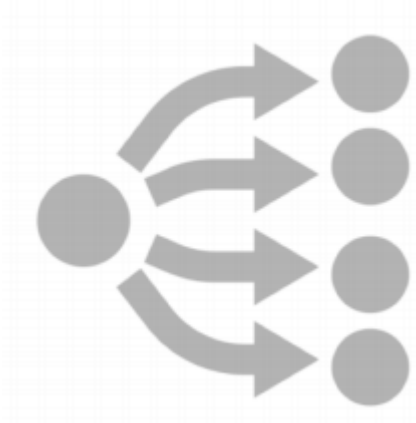
方案比较


CLOUDGENIX
riverbed
TALARI NETWORKS
velocloud
VERSA NETWORKS
viptela
Reshaping the Remote Office
Reshaping WAN Management
Reshaping WAN Security
Reshaping the WAN

下一个杀手级应用场景？

下一个杀手级应用场景？

SDN-LAN

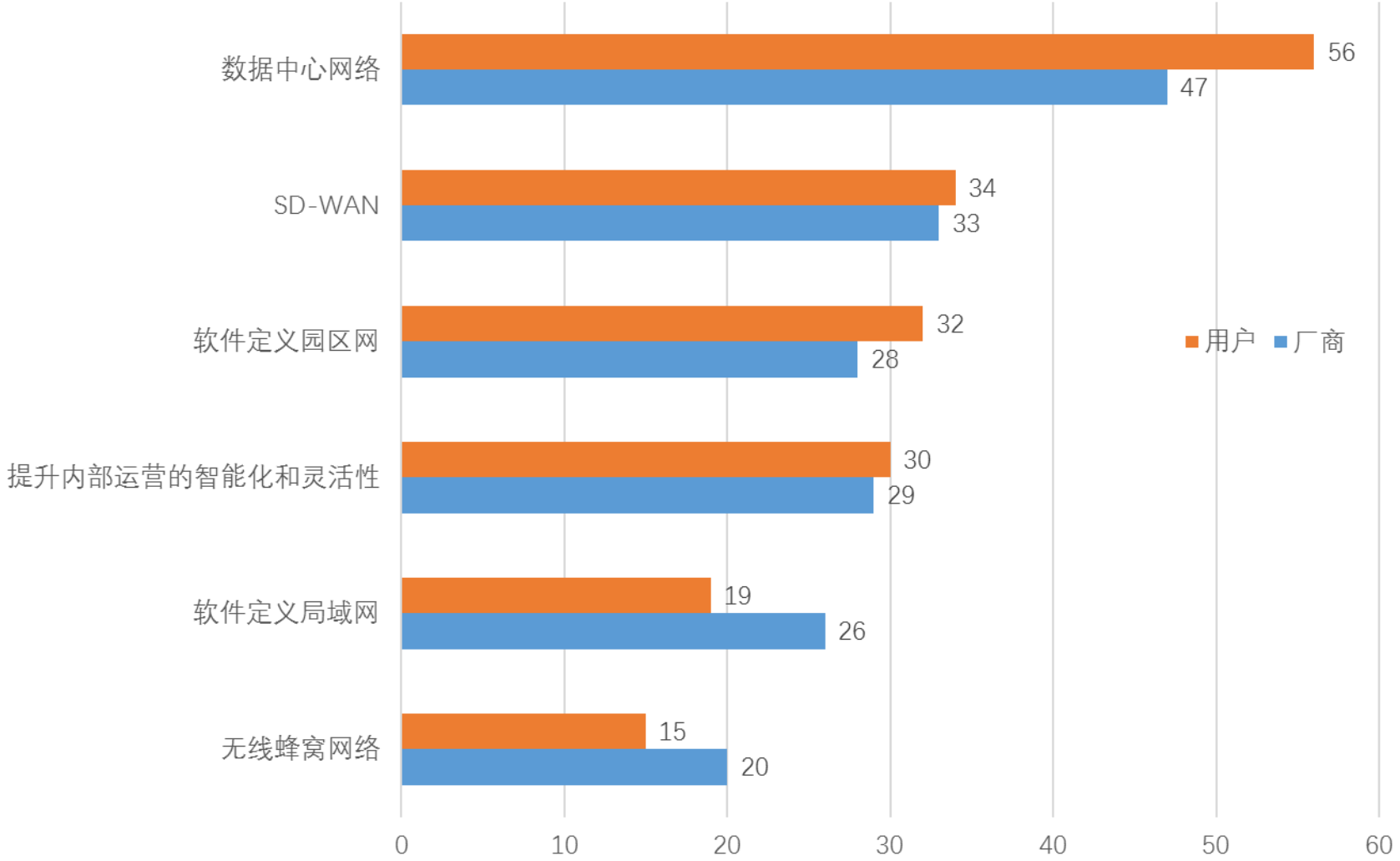


SDN-WAN

SDN-DCN

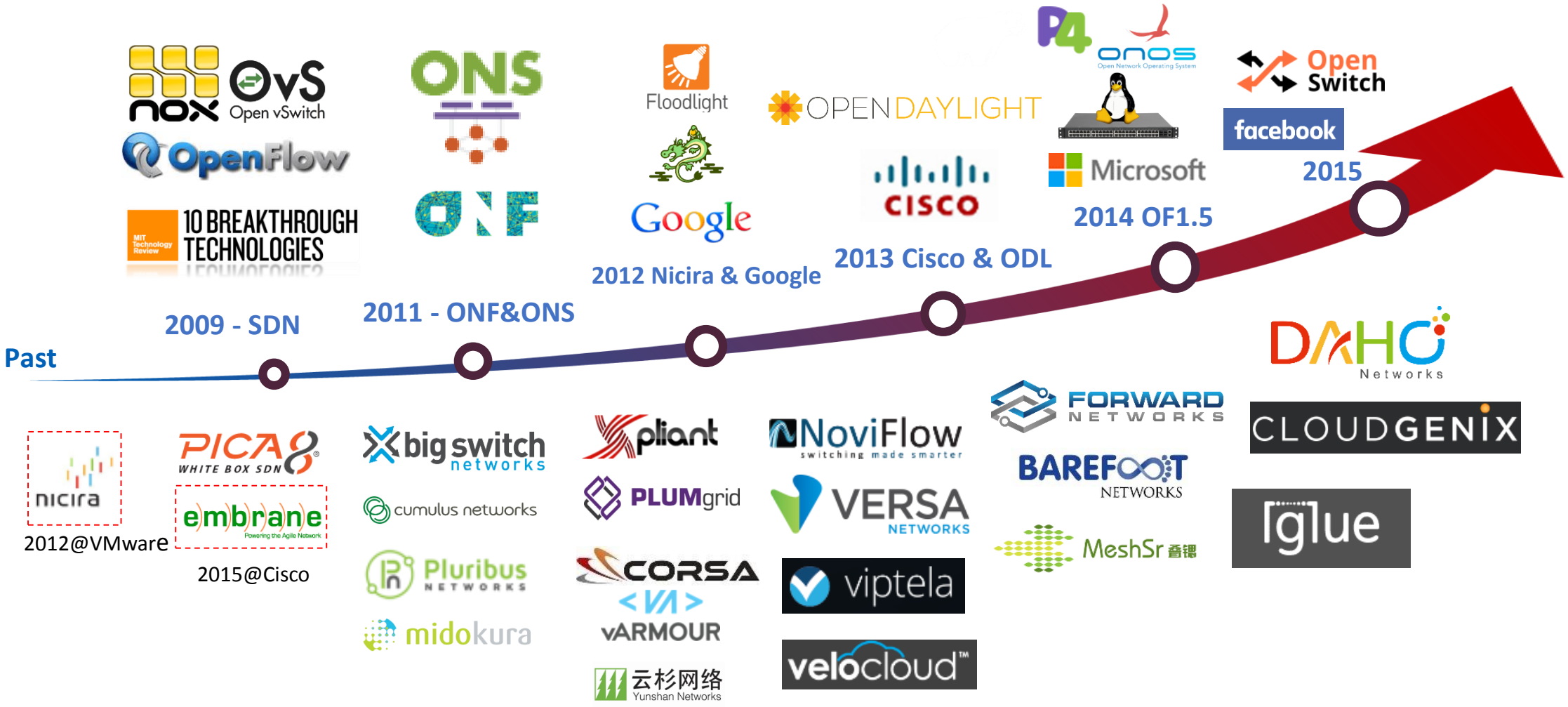
下一个杀手级应用场景？

未来网络学院
Future Network Institute



第三部分 - SDN对网络行业的影响

SDN在产业界的发展历史



最著名的Nicira Networks

未来网络学院 Future Network Institute

- Founded in June 2007 – Joint research Stanford / UC Berkeley
- Co-developed with Google to achieve massive scale and robustness
- Backed by Andreessen Horowitz, LVP and NEA
- Currently in beta with production deployments in Q1 2011

Executive Team

Steve Mullaney – CEO
Palo Alto Networks, Blue Coat, Force10, Cisco, Bay Networks, SynOptics

Martin Casado – Co-Founder, CTO
Stanford University, Lawrence Livermore National Lab

Frank D'Agostino – VP, Worldwide Technical Operations
Cisco, Anixter, SoCal Edison, Kaiser Hospitals

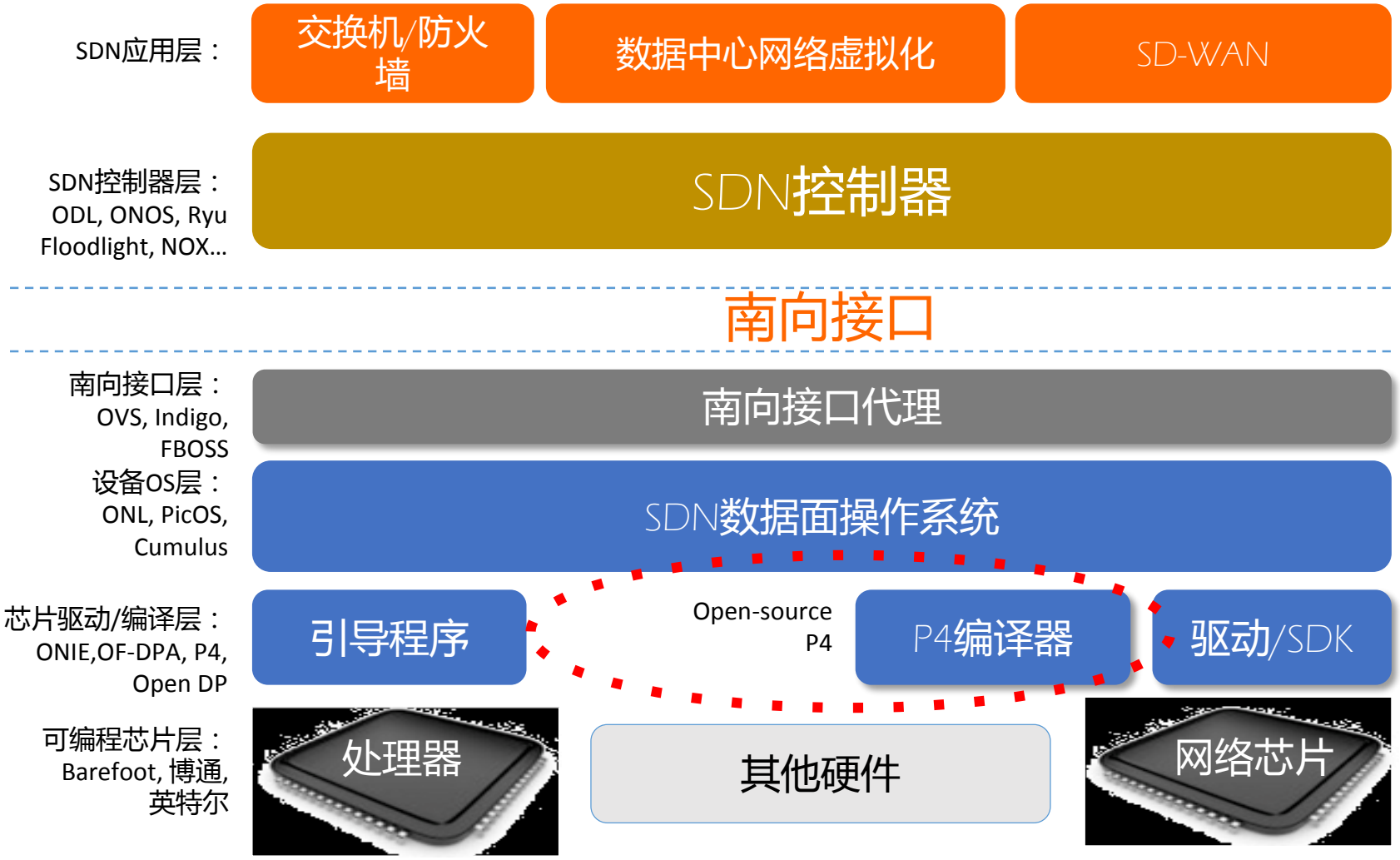
Rob Enns – VP, Engineering
Juniper, FORE Systems, BNR

Paul Fazzone – VP, Product Management
Cisco, Chase Manhattan Bank

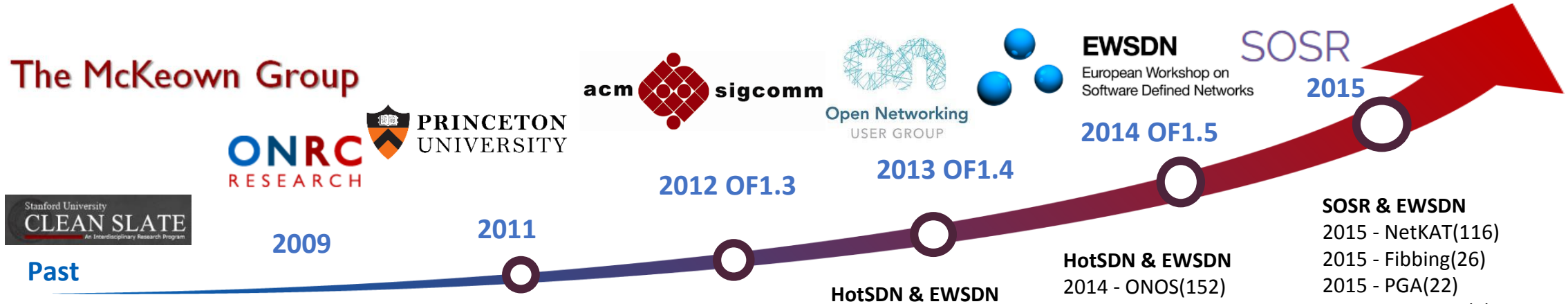
John Jendricks – VP, Business Development & Operations
Force10, Juniper, Cisco, StrataCom, Wellfleet, Adobe

	日期	金额	投资方
Seed			
Series A	2009.6/2010.1	$575k/$3.89M	Mark McDonnell, Peter Thorp
Series B	2010.9	$11.35M	Ben Horowitz, SV Angel
Series C	2011.2	$26M	Andreessen Horowitz, Lightspeed Venture Partners, Peter Thorp
Acquired	2012.7	$1.26B	VMware

新的网络产业竞争格局


SDN应用层：
交换机/防火墙
数据中心网络虚拟化
SD-WAN
SDN控制器层：
ODL, ONOS, Ryu
Floodlight, NOX...
SDN控制器
南向接口
南向接口层：
OVS, Indigo,
FBOSS
南向接口代理
设备OS层：
ONL, PicOS,
Cumulus
SDN数据面操作系统
芯片驱动/编译层：
ONIE,OF-DPA, P4,
Open DP
引导程序
Open-source
P4
P4编译器
驱动/SDK
可编程芯片层：
Barefoot, 博通,
英特尔
处理器
其他硬件
网络芯片

SDN对网络科研的影响



2005 - RCP
2005 - 4D
2006 - SANE
2007 - Ethane(869)
2008 - OpenFlow

5777次引用

2009 - NOX (1369)
2009 - OVS (619)
2009 - FlowVisor(628)
2009 - OpenRoads(175)
2010 - OpenPipes(22)

2010 - Mininet (786)
2010 - Onix (785)
2010 - DIFANE(433)
2010 - HyperFlow(482)
2011 - Frenetic (424)
2011 - DevoFlow (605)
2011 - Nettle (74)
2011 - Maestro (177)

HotSDN & EWSDN
2012 - HAS (335)
2012 - NICE Way (326)
2012 - CPP (288)
2012 - Kandoo (269)
2012 - VeriFlow (361)
2012 - Debugger (130)
2012 - NetCore (150)
2012 - OFLOPS (170)
2012 - Odin (152)

HotSDN & EWSDN
2013 - RMT(162)
2013 - B4 (747)
2013 - Pyretic (295)
2013 - SDWAN (277)
2013 - ElastiCon (202)
2013 - Beacon (221)
2013 - SoftRAN (204)
2013 - SoftCell (150)
2013 - OpenSketch (155)
2013-Participatory (142)
2013 - Maple (100)
2013 - SIMPLIFY (292)
2013-SDN Security (123)

HotSDN & EWSDN
2014 - ONOS(152)
2014 - P4(159)
2014 - OpenNF(136)
2014-The road to SDN(139)
2014 - SDN Survey (501)
2014 - OpenVirtex(66)
2014 - FlowGuard (50)
2014 - CacheFlow (55)
2014 - OpenNF(136)
2014 - SDN TE (114)
2014 - DISCO (75)
2014 - SDN LTE (64)
2014 - CompassSDN(62)
2014 - OF&sFlow (78)

SOSR & EWSDN
2015 - NetKAT(116)
2015 - Fibbing(26)
2015 - PGA(22)
2015 - OpenBox(8)
2015 - Kinetic(27)
2015 - DC Switch.P4(6)
2015 - NetASM(5)
2015 - Towards PPS(7)
2015 - Universe PS(6)
2015 - P4 Compiler(20)
2015 - mSwitch(11)
2015 - Know to Flow Table(38)
2015 - CoVisor(28)
2015 - OpenvSwitch(93)
2016 - SNAP(6)
2016 - Universe PS(6)
2016-Open Packet Processor(56)

SDN科研一览

未来网络学院
Future Network Institute

Source: Diego Kreutz, SDN: A Comprehensive Survey v3-2014.10

Software-Defined Networking: A Comprehensive Survey

Diego Kreutz, Fernando M. V. Ramos, Paulo Verissimo, Christian Esteve Rothenberg, Siamak Azodolmolky, Steve Uhlig

(Submitted on 2 Jun 2014 (v1), last revised 8 Oct 2014 (this version, v3))

Software-Defined Networking (SDN) is an emerging paradigm that promises to change this state of affairs, by breaking vertical integration, separating the network's control logic from the underlying routers and switches, promoting (logical) centralization of network control, and introducing the ability to program the network. The separation of concerns introduced between the definition of network policies, their implementation in switching hardware, and the forwarding of traffic, is key to the desired flexibility: by breaking the network control problem into tractable pieces, SDN makes it easier to create and introduce new abstractions in networking, simplifying network management and facilitating network evolution. In this paper we present a comprehensive survey on SDN. We start by introducing the motivation for SDN, explain its main concepts and how it differs from traditional networking, its roots, and the standardization activities regarding this novel paradigm. Next, we present the key building blocks of an SDN infrastructure using a bottom-up, layered approach. We provide an in-depth analysis of the hardware infrastructure, southbound and northbound APIs, network virtualization layers, network operating systems (SDN controllers), network programming languages, and network applications. We also look at cross-layer problems such as debugging and troubleshooting. In an effort to anticipate the future evolution of this new paradigm, we discuss the main ongoing research efforts and challenges of SDN. In particular, we address the design of switches and control platforms -- with a focus on aspects such as resiliency, scalability, performance, security and dependability -- as well as new opportunities for carrier transport networks and cloud providers. Last but not least, we analyze the position of SDN as a key enabler of a software-defined environment.

Comments: Version 2.01: 61 pages, 11 figures, 17 tables, 579 references
Subjects: **Networking and Internet Architecture (cs.NI)**
Cite as: arXiv:1406.0440 **[cs.NI]**
(or arXiv:1406.0440v3 **[cs.NI]** for this version)

Submission history

From: Diego Kreutz [view email]
[v1] Mon, 2 Jun 2014 16:42:24 GMT (1306kb,D)
[v2] Mon, 6 Oct 2014 23:30:20 GMT (1436kb,D)
[v3] Wed, 8 Oct 2014 18:43:04 GMT (1437kb,D)

579引用文献
61页/3个版本

SDN对网络教学的影响

未来网络学院
Future Network Institute

软件定义网络课程

- **Princeton University (Jennifer Rexford)-普林斯顿大学**
- **Columbia University (Li Erran Li)-哥伦比亚大学**
- **Coursera (Nick Feamster)-普林斯顿大学**
- **ETHz (Bernhard Plattner)-苏黎世联邦理工大学**

高级网络课程（SDN）

- COS461: Computer Network, Princeton University Spring 2016
- CS244: Advanced Topics in Networking, Spring 2016-Stanford
- CS740: Advanced Computer Networks, WISC (Aditya Akella),
- CMU (Vyas Sekar), USC (Minlan Yu), Duke (Theophilus Benson), University of Kentucky (Zongming Fei), Tel Aviv University (Mooly),
- Brown (Rodrigo Fonseca), WUSTL (Raj Jain), Cornell (Nate Foster), University of Colorado Boulder (Eric Keller), UC (Scott), UMICH(Z.Morley Mao), University of Illinois(Klara Nahrstedt),

Module 1: History and Evolution of Software Defined Networking

Time: Approximately 5 hours | **Difficulty Level:** Easy

Learning Objectives

- Review the timeline of software defined networks from the 1980s to present time.
- Raise awareness about the genesis of various ideas and principles, and why we have these principles in the first place.
- Recognize architectural themes in computer networking where the SDN principles came from. (Why? Because the specific technologies may evolve, but the principles are in some sense timeless or at least have been recognized as "best practice").

Key Terms

- **Software Defined Networking (SDN):** A new networking paradigm whereby the behavior of a network is controlled by a single high-level software program. The general term for network architectures whereby the control plane (software that controls network behavior) and the data plane (the devices that forward traffic) are separate from one another.
- **Control plane:** The functions in the network that control the behavior of the network (e.g., network paths, forwarding behavior). Typically, the control plane is instantiated as a single, high-level software controller.
- **Data plane:** The functions in the network that are responsible for forwarding (or not forwarding) traffic. Typically, the data plane is instantiated as forwarding tables in routers, switches, firewalls, and middleboxes.
- **Active networks:** A collection of network architecture projects in the 1990s that shared many of the same goals as software-defined networking.
- **Network virtualization:** The notion of instantiating many distinct logical networks on top of a single, shared physical network infrastructure.

Checklist to complete Module 1

Watch the following lectures:

- Module 1.1: Central Control [6 minutes]
- Module 1.2: Programmable Networks [12 minutes]
- Module 1.3: Network Virtualization [12 minutes]
- Module 1.4: Control Plane Evolution [9:19]
- Module 1.5: The Road to SDN [7:05]
- Tutorial: Vagrant/Mininet Course VM Setup [1:48]

Complete the following Activities:

- Read the article "The Road to SDN: An Intellectual History of Programmable Networks"
- Quiz 1: History and Evolution of SDN
- Programming Assignment 1: Setup Virtual Box/Mininet Environment for SDN - *This is an activity that sets you up for future programming assignments; you do not have to submit anything. The assignment also provides everything you need to get set up. We will attempt to provide a setup video lecture shortly to help you follow along.*

Guest Interviews

- Interview with Martin Casado, Founder of Nicira Networks (recently acquired by VMWare). Topics discussed:
  - Evolution of Software Defined Networking
  - Relationship of network virtualization to SDN
  - Myths and misunderstandings of Software Defined Networking
  - The future of SDN: challenges and opportunities
- Interview with Nick McKeown

Support Resources

- Virtual Box (virtual machine environment)
- David Medberry's Hints on Virtual Box Host-Only Networking Setup
- Jacob Cox's Hints on Vagrant VM Setup on Windows 7

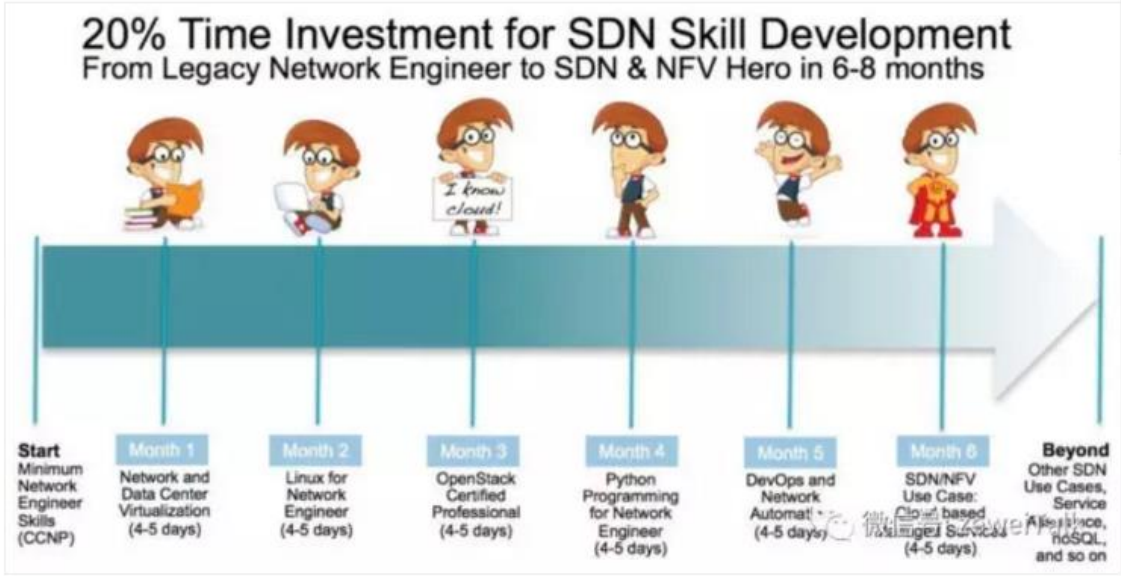
- Feamster, Nick et al. "The Road to SDN: An Intellectual History of Programmable Networks.ACM Queue. December 2013.
- Caesar, Matthew, et al. "Design and implementation of a routing control platform." Proceedings of the 2nd conference on Symposium on Networked Systems Design & Implementation-Volume 2. USENIX Association, 2005.

面对SDN我们怎么办？

给大家的建议

面对SDN，我们怎么办

2017-04-25 泽卫 SDN学习与研究

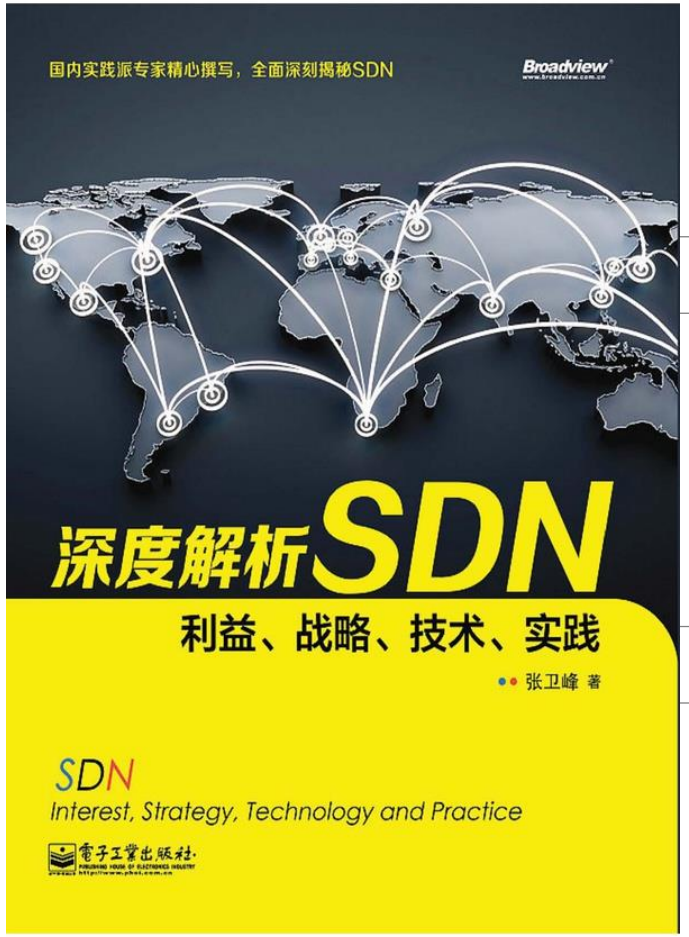


回顾SDN的发展历史，可以发现：SDN作为一种新的网络体系结构，对网络学术圈和工业界都产生了深远的影响。SDN并不是昙花一现的网络技术新概念，而是一场网络领域的技术变革。

我们将如何应对这场变革：SDN初学者如何入门和进阶？网工如何在熟悉领域纵深发展？产品经理如何在SDN架构下设计下一代网络产品？创业者如何寻找下一个SDN杀手级应用？投资者如何找到SDN领域的独角兽？**这些问题依然值得讨论。**

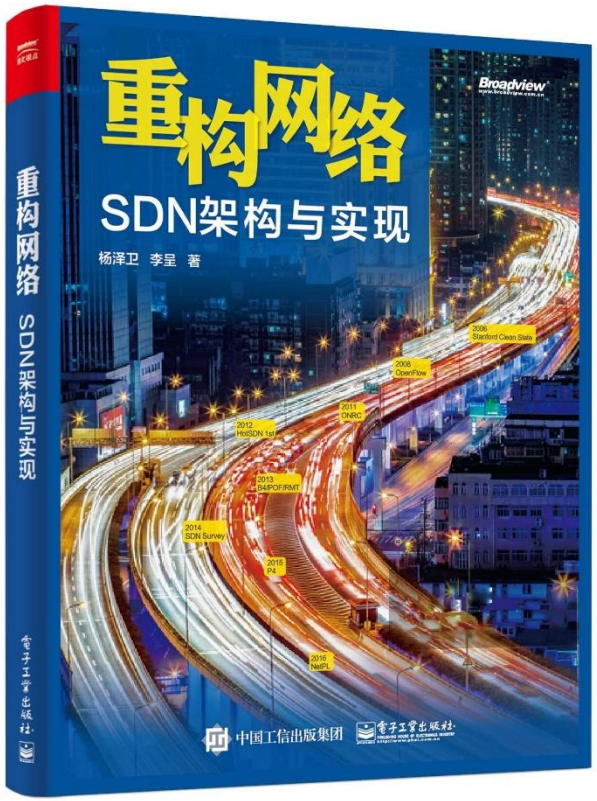
如何入门？

未来网络学院
Future Network Institute



切记碎片化学习

如何进阶？



Software Defined Networking

关于此课程： In this course, you will learn about software defined networking and how it is changing the way communications networks are managed, maintained, and secured.

制作方： 普林斯顿大学

PRINCETON UNIVERSITY

教学方： Dr. Nick Feamster, Professor
Department of Computer Science

语言	English
如何通过	通过所有计分作业以完成课程。

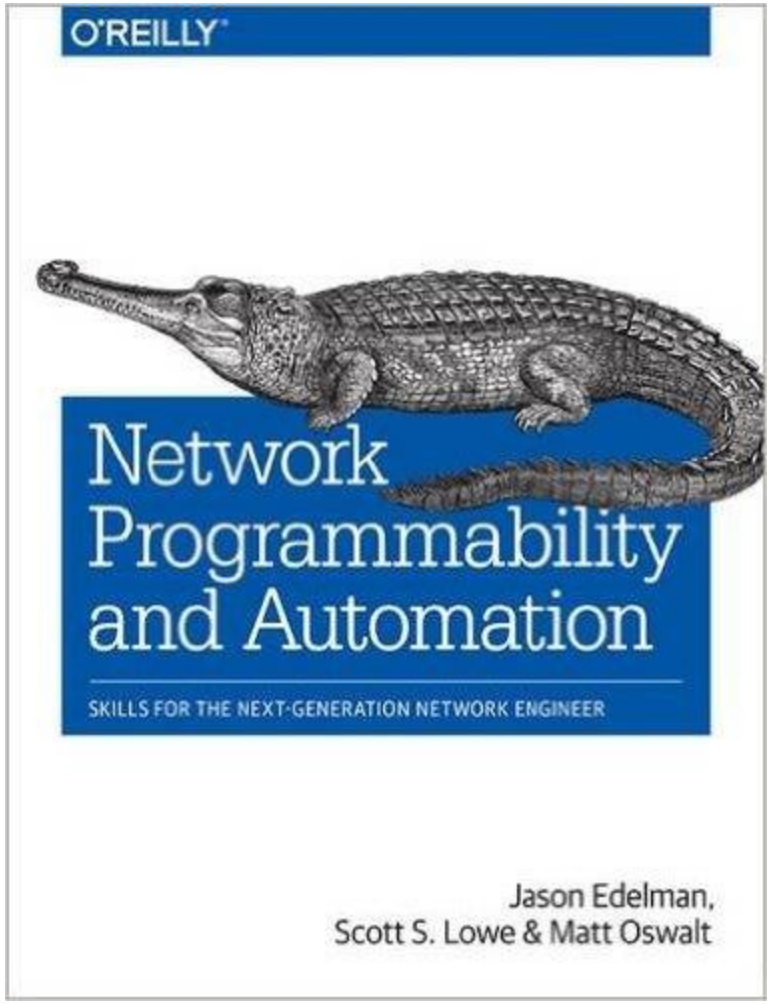
动手搭建SDN网络

未来网络学院
Future Network Institute

从网络工程师到网络开发者

未来网络学院
Future Network Institute

Project	Primary Language Written In
Open vSwitch	C
OpenDaylight	Java
Floodlight	Java
Ryu	Python
OpenStack	Python
CloudStack	Java
OpenContrail	C++



从网络工程师到网络开发者

未来网络学院
Future Network Institute

FORWARD NETWORKS

Building the best ideas in Computer Science to networking

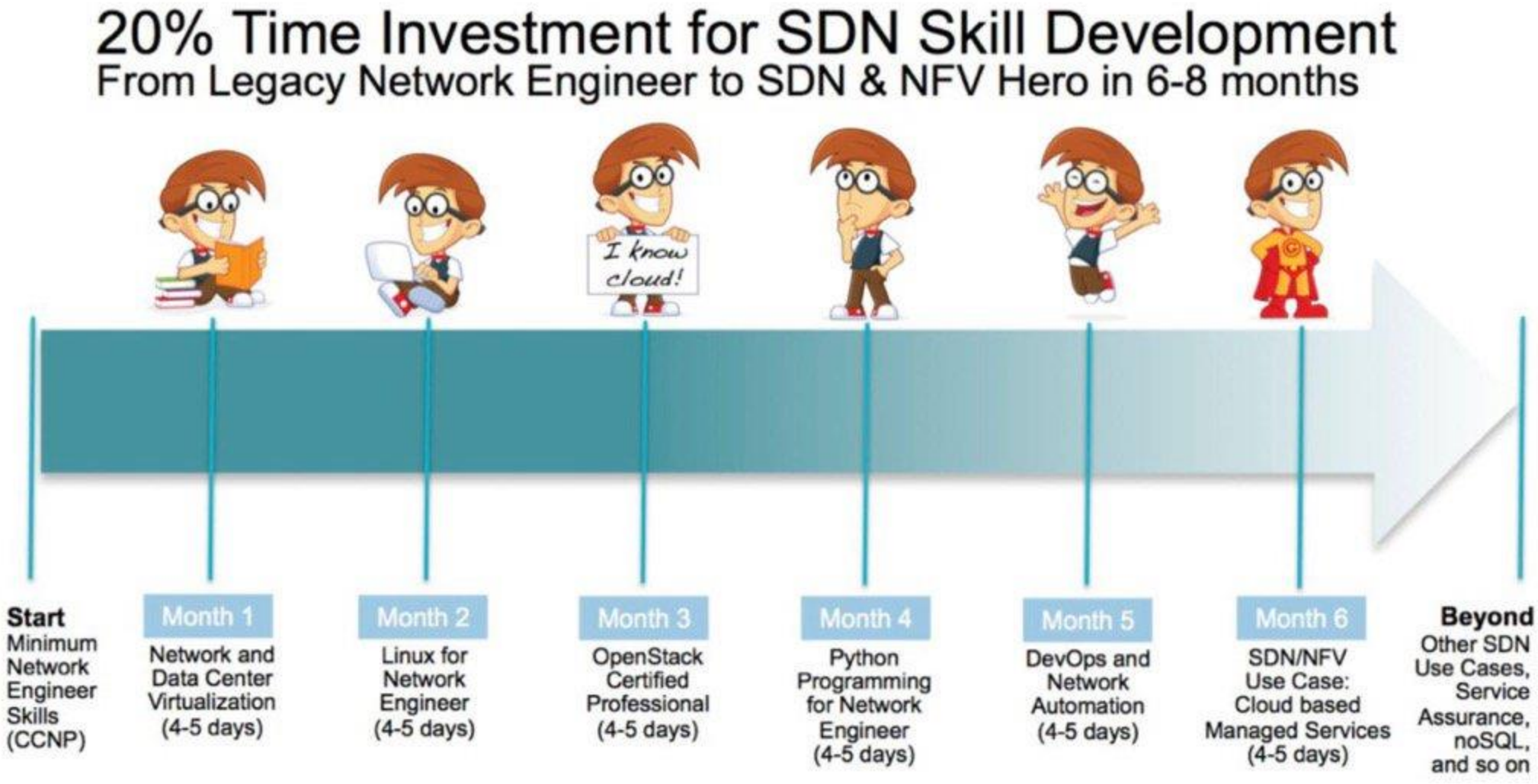
David Erickson (CEO)

Brandon Heller (CTO)

Peyman Kazemian

Nikhil Handigol

SDN学习之路



讨论

网络的未来属于SDN