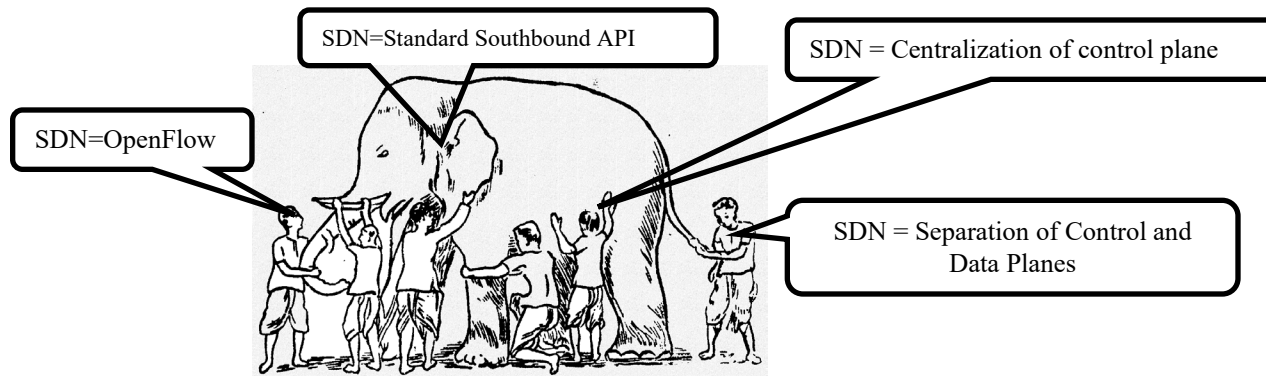


# Introduction to Software Defined Networking (SDN)



Raj Jain  
Washington University in Saint Louis  
Saint Louis, MO 63130  
Jain@cse.wustl.edu

These slides and audio/video recordings of this class lecture are at:

<http://www.cse.wustl.edu/~jain/cse570-19/>

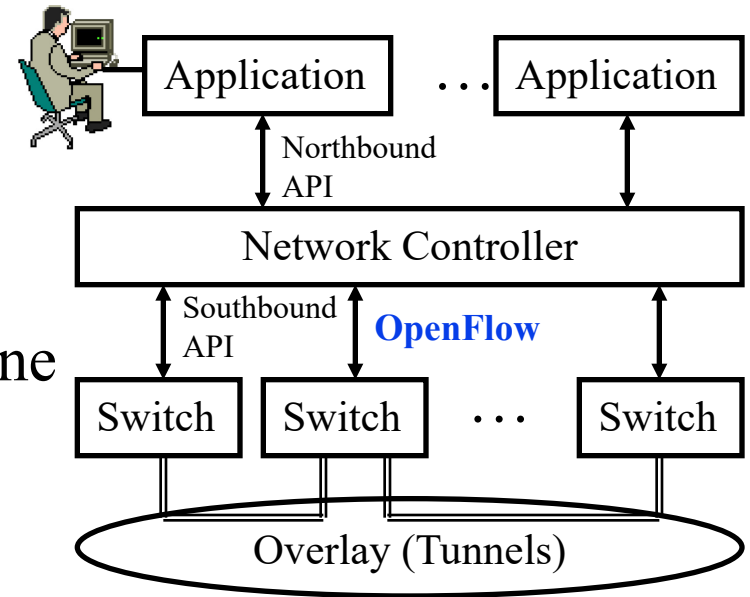


1. What is SDN?
2. SDN Controllers
3. Alternative APIs: XMPP, PCE, ForCES, ALTO
4. RESTful APIs and OSGi Framework

Note: This is the second module of three modules on OpenFlow, SDN and NFV in this course.

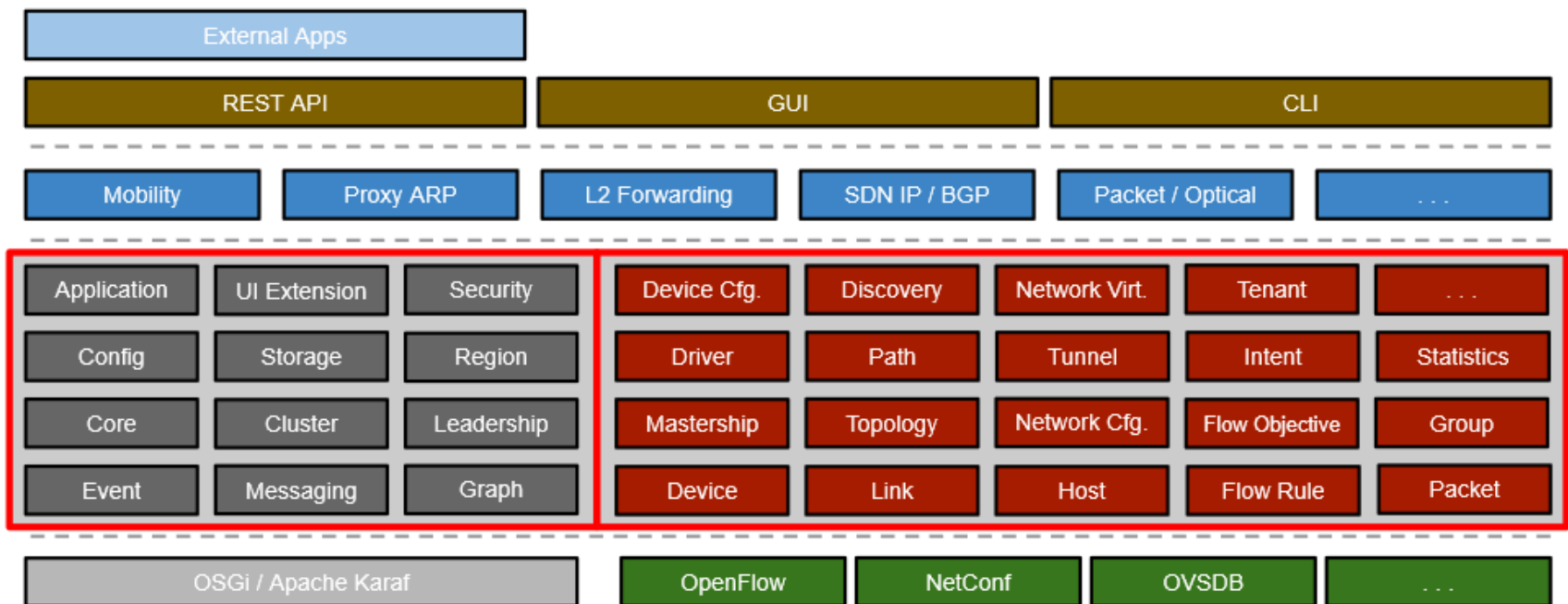
# Origin of SDN

- ❑ SDN originated from OpenFlow
- ❑ Centralized Controller
  - ⇒ Easy to program
  - ⇒ Change routing policies on the fly
  - ⇒ Software Defined Network (SDN)
- ❑ Initially, SDN=
  - Separation of Control and Data Plane
  - Centralization of Control
  - OpenFlow to talk to the data plane
- ❑ Now the definition has changed significantly.



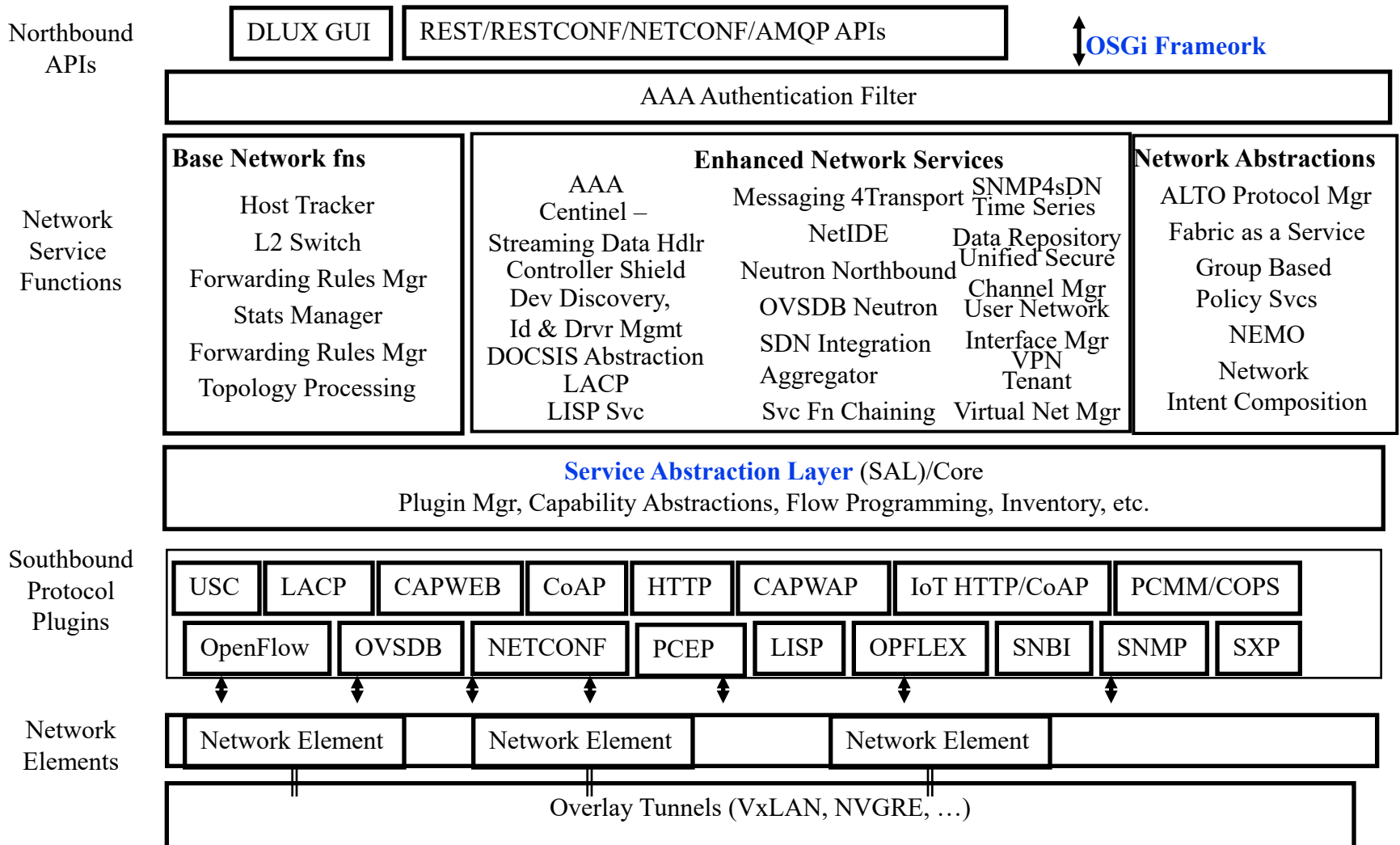
# ONOS

- ❑ Open Network Operating System:  
Distributed OpenFlow OS for a large WAN
- ❑ Initially OpenFlow-only. Now multi-protocol southbound.



Ref: ONOS Architecture, [https://docs.google.com/presentation/d/1Y4S82YZysqKAZW4kKm-6IIO04h\\_nnrQyfrsCdt-I/edit?usp=sharing](https://docs.google.com/presentation/d/1Y4S82YZysqKAZW4kKm-6IIO04h_nnrQyfrsCdt-I/edit?usp=sharing)  
Washington University in St. Louis <http://www.cse.wustl.edu/~jain/cse570-19/> ©2019 Raj Jain

# OpenDaylight: Multi-Protocol SDN



# OpenDaylight SDN Controller Platform (OSCP)

- ❑ Multi-company collaboration under Linux foundation
- ❑ Many projects including OpenDaylight Controller
- ❑ Supports multiple southbound protocols via plug-ins including OpenFlow
- ❑ Dynamically linked in to a Service Abstraction Layer (SAL) Abstraction  $\Rightarrow$  SAL figures out how to fulfill the service requested by higher layers irrespective of the southbound protocol
- ❑ Modular design using **OSGI framework**
- ❑ A rich set of North-bound APIs via **RESTful** services for loosely coupled applications and OSGI services for co-located applications using the same address space

Ref: OpenDaylight Getting Started Guide, <https://docs.opendaylight.org/en/stable-sodium/getting-started-guide/index.html>  
[https://wiki.opendaylight.org/view/Main\\_Page](https://wiki.opendaylight.org/view/Main_Page)

# RESTful APIs

- ❑ Software architecture style developed by W3C.
- ❑ Introduced by Roy Fielding in his PhD thesis.
- ❑ WWW uses this style. Very popular in other applications.
- ❑ Goals: Scalability, Generality, Independence, and allow intermediate components
- ❑ Client-Server Model: Clients and servers can be developed independently.
- ❑ Server is stateless
- ❑ Responses can be cached for the specified time
- ❑ Intermediate Servers (Proxies) can respond. End point is not critical.

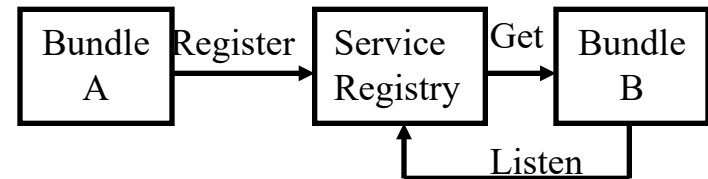
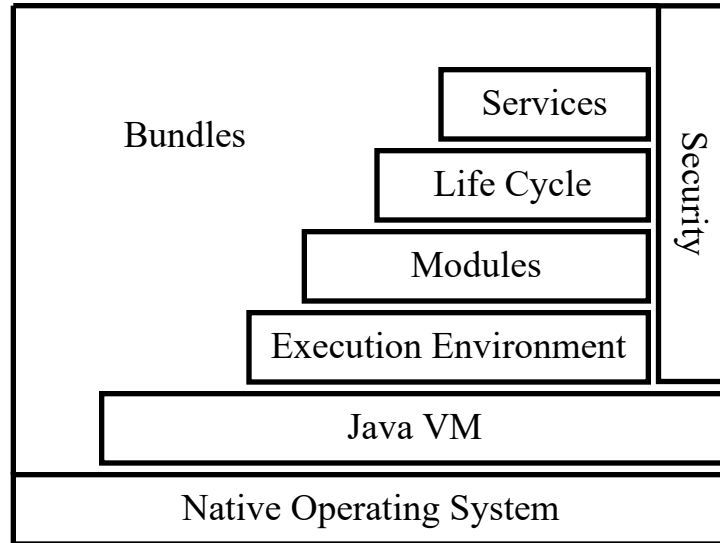
# REST (Cont)

- ❑ Create, Read, Update, Delete (CRUD) Operations
- ❑ Uniform Interface: GET (Read), POST (Insert), PUT (write), DELETE
- ❑ Resources identified by global identifiers, e.g., URI in Web.
- ❑ Get `http://<fqdn-or-ip-address>/rest/v1/model/<data-type>/<optional-id>?<optional-query-params>`  
E.g., GET `http://odcp.org/rest/v1/model/controller-node`
- ❑ Data Types: Controller node, Firewall rule, Topology configuration, Switch, Port, link, flow entry, VLAN, ...
- ❑ Data types can include commercial entities, such as, Big Virtual Switch from Big Switch Networks, vCenter from VMware, ...
- ❑ If optional-id and query parameters are omitted, the returned text includes all of the items of the given data type.



# OSGi Framework

- ❑ Initially, Open Services Gateway initiative
- ❑ A set of specifications for dynamic application composition using reusable Java components called bundles
- ❑ Bundles publish their services with OSGi services registry and can find/use services of other bundles



Ref: <http://www.osgi.org/Technology/WhatIsOSGi>

Washington University in St. Louis

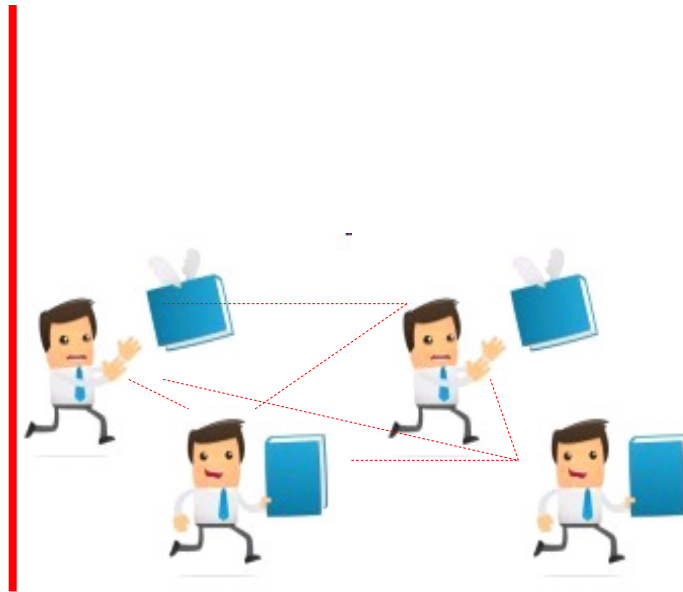
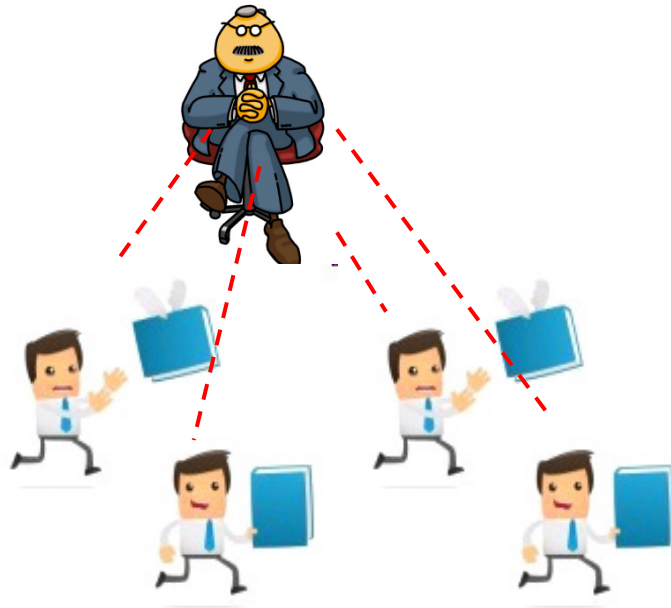
<http://www.cse.wustl.edu/~jain/cse570-19/>

©2019 Raj Jain

# OSGi (Cont)

- ❑ Bundles can be installed, started, stopped, updated or uninstalled using a lifecycle API
- ❑ Modules defines how a bundle can import/export code
- ❑ Security layer handles security
- ❑ Execution environment defines what methods and classes are available in a specific platform
- ❑ A bundle can get a service or it can listen for a service to appear or disappear.
- ❑ Each service has properties that allow others to select among multiple bundles offering the same service
- ❑ Services are dynamic. A bundle can decide to withdraw its service. Other bundles should stop using it  
⇒ Bundles can be installed and uninstalled on the fly.

# Centralized vs. Distributed



- ❑ Fast Response to changes
- ❑ Fast Consistency
- ❑ Less overhead  $\Rightarrow$  Scalable
- ❑ Single Point of Failure

- ❑ Time to converge
- ❑ Slow consistency
- ❑ Not scalable
- ❑ Fault Tolerant

# Four Confusions About SDN

## 1. Policies vs. Control:

Control = All bits and messages not sent by the user  
In IP, control includes all header bits and all routing messages.

## 2. Separation of Control Plane:

Elements have only data plane and have no brains

## 3. SDN vs. OpenFlow:

OpenFlow is the father of SDN but not SDN.

## 4. Need OpenFlow:

- OpenFlow is micro-management.
- It is not scalable.
- For large infrastructure, need scalable solutions.

# Separation vs. Centralization

Separation of  
Control Plane

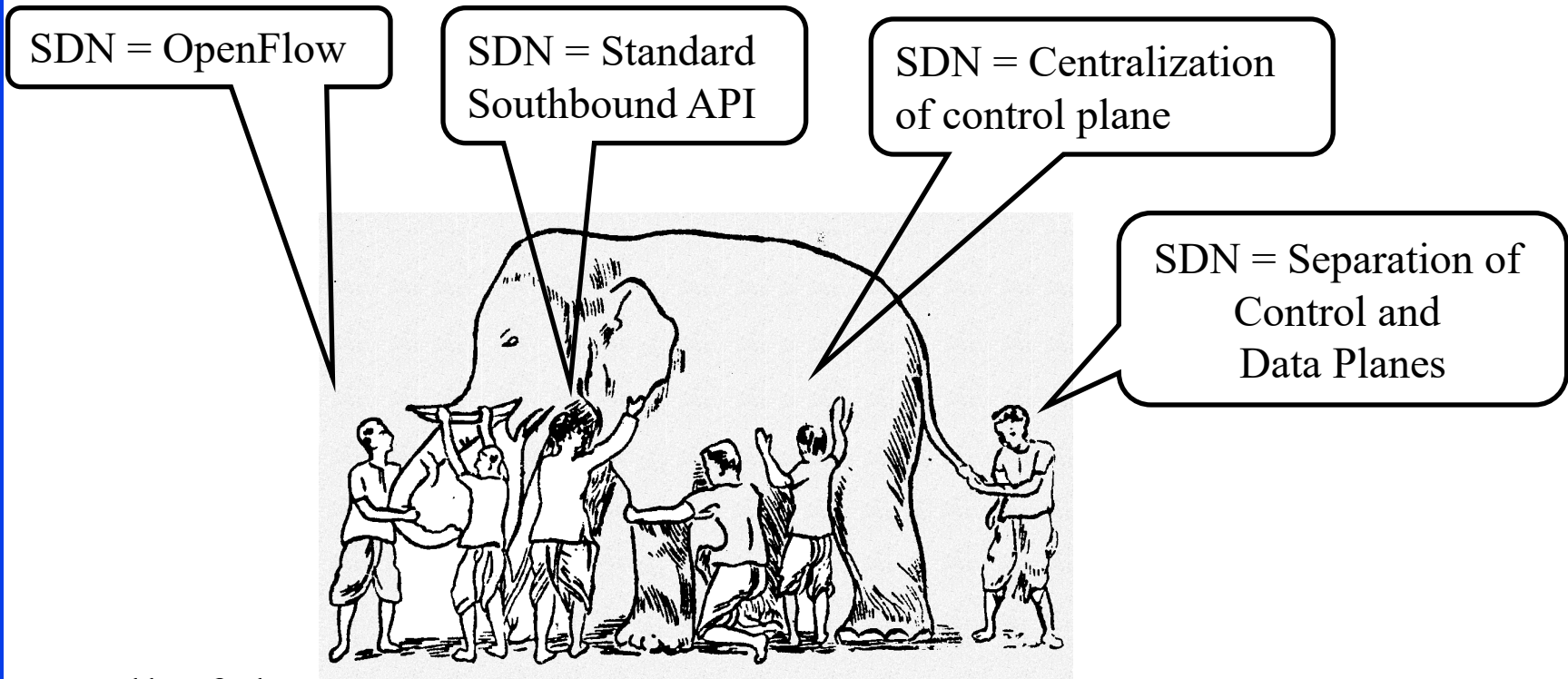


Centralization of  
Policies



Micromanagement is not scalable

# What SDN is Not?

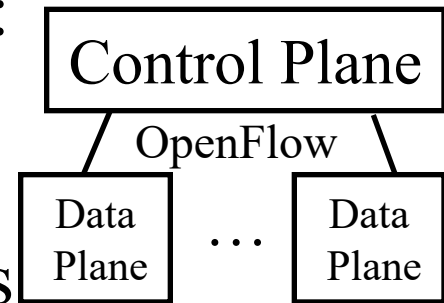


- ❑ All of these are mechanisms.
- ❑ SDN is *not* about a mechanism.
- ❑ It is a framework  $\Rightarrow$  Many solutions

# Trend 1: SDN to Disaggregation

❑ SDN was invented in 2009. Then: SDN:

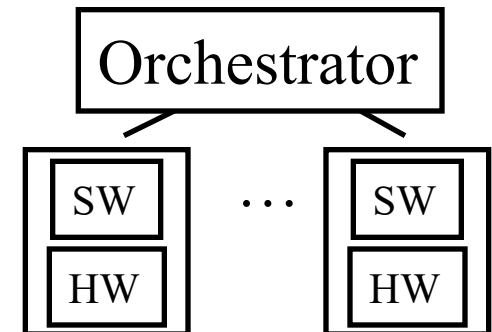
- Separation of control and data planes
- Centralization of Control
- Standard Protocol between the planes



❑ **203** Papers on OpenFlow on IEEEExplore in 2018!

❑ Now: Software Defined = **Disaggregation** of HW/SW

- Commodity hardware
- Software on commodity HW
- Legacy protocols survive



Ref: D. M Batista, G. Blair, F. Kon, R. Boutaba, D. Hutchison, R. Jain, R. Ramjee, C. Rothenberg, "Perspectives on software-defined networks: interviews with five leading scientists from the networking community" Journal of Internet Services and Applications 2015, 6:22, <http://www.cse.wustl.edu/~jain/papers/jisa15.htm>

J. Skorupa and D. Cisco, "State of SDN: If You Think SDN Is the Answer, You're Asking the Wrong Question,"

Gartner Report G00325601, 24 August 2017, 9 pp.

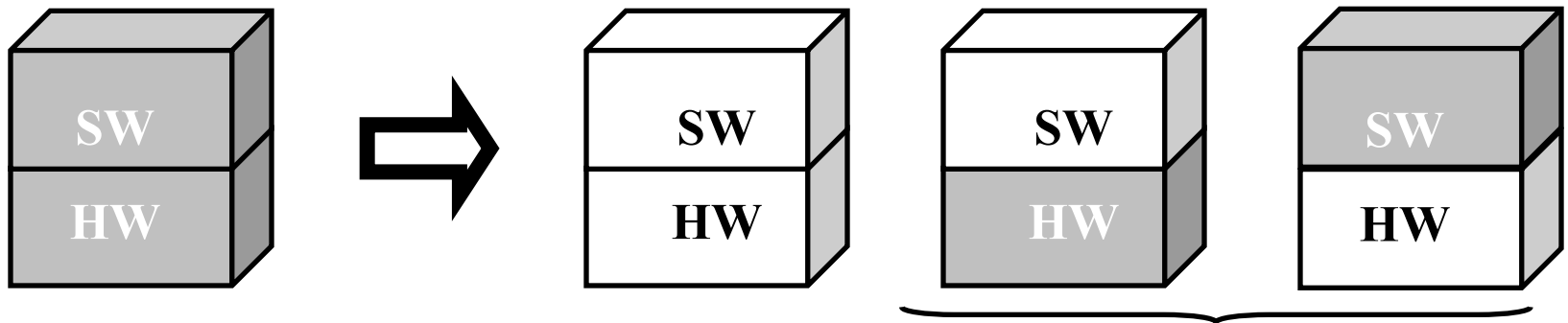
Washington University in St. Louis

<http://www.cse.wustl.edu/~jain/cse570-19/>

©2019 Raj Jain

# Disaggregation: Black Box to White Box

- ❑ Differentiation via software  $\Rightarrow$  White box networking
- ❑ **Black Box**: Proprietary HW with Proprietary SW
- ❑ **White Box**: Open Source Hardware and Software
- ❑ Software on a different hardware  
 $\Rightarrow$  hardware can change  
Different software on a hardware  
 $\Rightarrow$  Software can change
- ❑ **Bright Box**: Branded White box =  
Branded SW on open HW or Open SW on Branded HW

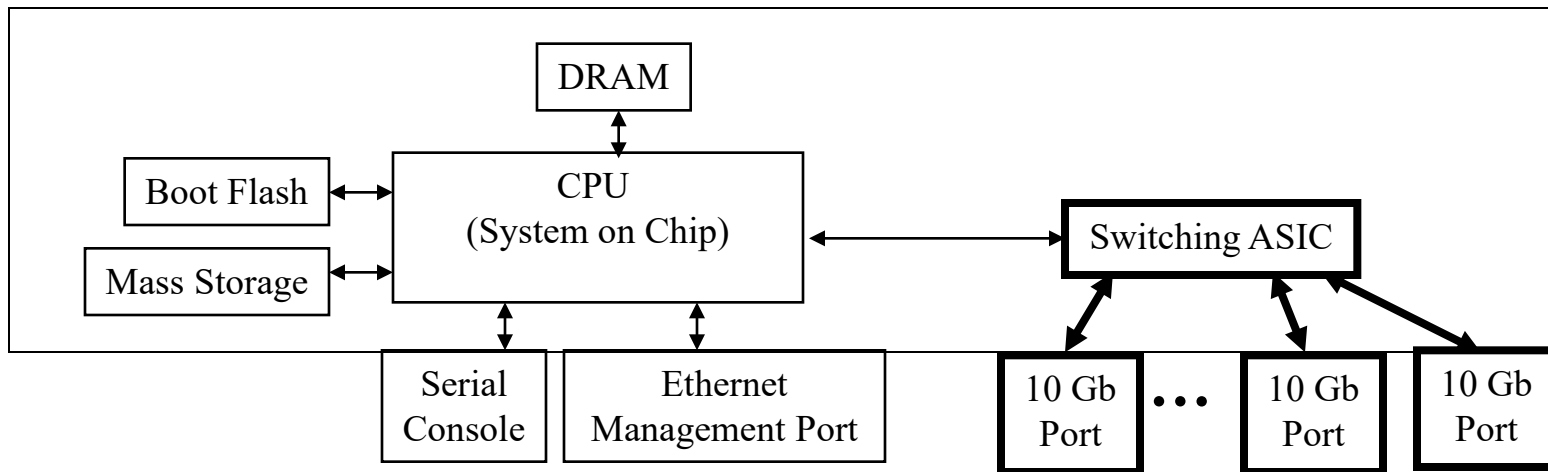


Ref: A. Lerner, "Branded Switching + White-Box Switching = Brite-Box Switching," Nov 14, 2014,  
<https://blogs.gartner.com/andrew-lerner/2014/11/19/britefuture/>



# Bare Metal Switches

- ❑ Hardware that can be used to load different network operating systems
- ❑ Open Network Linux is supported by hardware from: Accton/Edge-Core, Quanta, Dell, Mellanox, Netberg, Inventec, Celestica, HPE, DNI, Ingrasys, and Alpha Networks



# Open Source Forwarding Agents

- ❑ **Quagga**: A popular open source routing software suite including OSPF, RIP, BGP, ...
- ❑ **FRRouting**: a fork of Quagga. Linux routing protocol suite including BGP, IS-IS, LDP, OSPF, PIM, and RIP (Free Range Routing?)
- ❑ **BIRD**: Internet Routing Daemon developed as a school project at Charles University, Prague. Supports IPv4, IPv6, BGP, RIP, OSPF, ...
- ❑ Facebook Open Switching System (**FBOSS**): S/w stack for controlling and managing network switches with several user-space applications
- ❑ Azure Software for Open Networking in the Cloud (**SONiC**)
- ❑ Google **gNOS**

Ref: <https://www.nongnu.org/quagga/>, <https://www.opensourcerouting.org/>, <http://bird.network.cz/>,  
<https://github.com/facebook/fboss>, <http://azure.github.io/SONiC/>

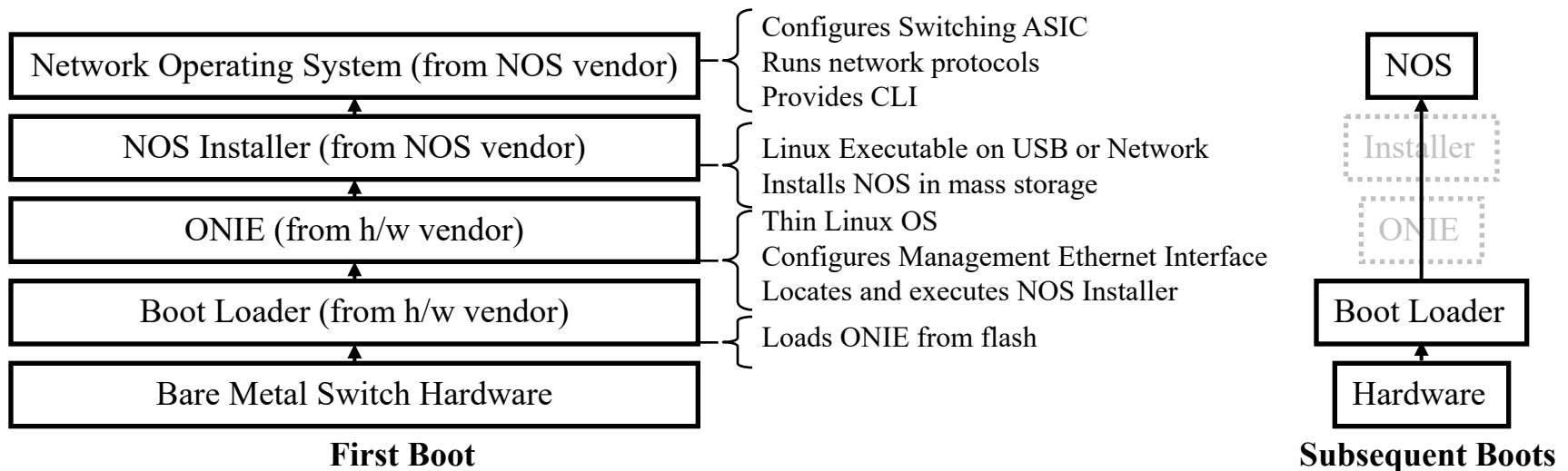
# Open Network Linux

- ❑ Linux distribution for “*open hardware*” bare metal switches
- ❑ Part of Open Compute Project
- ❑ Supports multiple *switch fabric APIs*:
  - **OF-DPA**: OpenFlow Data Plane Abstraction (API) for Broadcom chips
  - **OpenNSL**: Open Network Switch Layer for Broadcom switches
  - **SAI**: Switch Abstraction Interface (vendor independent API to control forwarding elements)
- ❑ Compatible with many open-source *forwarding agents* or routing protocol suites

Ref: <http://opennetlinux.org/> , <https://github.com/Broadcom-Switch/OpenNSL>, <https://github.com/Broadcom-Switch/of-dpa>, <https://github.com/opencomputeproject/SAI>

# Open Network Install Environment (ONIE)

- ❑ Part of **Open Compute Project (OCP)** open source initiative
- ❑ Allows many different “**Network Operating Systems (NOS)**” on bare metal network switches
- ❑ Like a firmware that locates the NOS boot image and loads it
- ❑ ONIE sets the environment on the first boot and is not required subsequently



Ref: ONIE Overview, <https://opencomputeproject.github.io/onie/overview/index.html#onie-overview>

Washington University in St. Louis

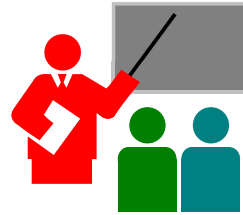
<http://www.cse.wustl.edu/~jain/cse570-19/>

©2019 Raj Jain

# Mininet

- ❑ Widely used open source network emulation environment.
- ❑ Can simulate a number of end-hosts, switches, routers, links on a Linux
- ❑ Used for rapid prototyping of software define networks
- ❑ Built-in Open vSwitch, and a OpenFlow capable switch
- ❑ Command line launcher and Python API for creating networks of varying sizes, e.g., *mn -topo tree,depth=2,fanout=3*
- ❑ Useful diagnostic commands like iperf, ping, and other commands in a host, e.g., *mininet> h11 ifconfig -a*
- ❑ Mininet code for several popular commercial switches are available.

# Summary



1. SDN = Abstraction + Programmability + Centralization  
SDN = Disaggregation of h/w and s/w  
= Bare metal switches + ONIE + ONL
2. OpenFlow originated SDN but now many different southbound and northbound APIs, intermediate services and tools are being discussed and implemented by the industry,
3. OpenDaylight and ONOS are SDN Controllers.  
Differ on how much open.
4. Mininet for network simulation
5. REST=HTTP APIs  
OSGI framework for modularity

# Reading List

- ❑ Thomas D. Nadeau, Ken Gray, "SDN: Software Defined Networks," O'Reilly Media, Inc., August 2013, 384 pp., ISBN:978-1-4493-4230-2 (Safari Book).

# Wikipedia Links

- ❑ [http://en.wikipedia.org/wiki/Software-defined\\_networking](http://en.wikipedia.org/wiki/Software-defined_networking)
- ❑ [http://en.wikipedia.org/wiki/Representational\\_state\\_transfer](http://en.wikipedia.org/wiki/Representational_state_transfer)
- ❑ <http://en.wikipedia.org/wiki/OSGI>



# References

- ❑ Jim Doherty, "SDN and NFV Simplified: A Visual Guide to Understanding Software Defined Networks and Network Function Virtualization," Addison-Wesley Professional, March 2, 2016, 320 pp., ISBN:978-0-13-430739-8 (Safari Book).
- ❑ Reza Toghraee, "Learning OpenDaylight," Packt Publishing, May 2017, 336 pp., ISBN:978-1-78217-452-3 (Safari Book).
- ❑ Antonio Sanchez Monge; Krzysztof Grzegorz Szarkowicz, "MPLS in the SDN Era," O'Reilly Media, Inc., December 2015, 920 pp., ISBN:978-1-4919-0545-6 (Safari Book).

# References (Cont)

- ❑ Oswald Coker, Siamak Azodolmolky, "Software-Defined Networking with OpenFlow - Second Edition," Packt Publishing, October 2017, 246 pp., ISBN:978-1-78398-429-9 (Safari Book).
- ❑ William Stallings, "Foundations of Modern Networking: SDN, NFV, QoE, IoT, and Cloud," Addison-Wesley Professional, October 2015, 544 pp., ISBN:0-13-417539-5 (Safari Book).
- ❑ Russ White, Jeff Tantsura, "Navigating Network Complexity: Next-generation Routing with SDN, Service Virtualization, and Service Chaining," Addison-Wesley Professional, November 2015, 320 pp., ISBN:0-13-398792-2 (Safari Book).
- ❑ Abhishek Ratan, "Practical Network Automation," Packt Publishing, November 2017, 266 pp., ISBN:978-1-78829-913-8 (Safari Book).
- ❑ Scott S. Lowe, Matt Oswalt, Jason Edelman, "Network Programmability and Automation," O'Reilly Media, Inc., February 2018, 581 pp., ISBN:978-1-4919-3125-7 (Safari Book).
- ❑ Guy Pujolle, "Software Networks," John Wiley & Sons, August 2015, 260 pp., ISBN:978-1-119-00796-8 (Safari Book).

# References (Cont)

- ❑ Sriram Subramanian, Sreenivas Voruganti, "Software-Defined Networking (SDN) with OpenStack," Packt Publishing, October 2016, 216 pp., ISBN:978-1-78646-599-3 (Safari Book).
- ❑ Paul Goransson, Chuck Black, Timothy Culver, "Software Defined Networks, 2nd Edition," Morgan Kaufmann, October 2016, 436 pp., ISBN:978-0-12-804555-8 (Safari Book).
- ❑ Patricia A. Morreale, James M. Anderson, "Software Defined Networking," CRC Press, September 2015, 186 pp., ISBN:1-4822-3864-0 (Safari Book).
- ❑ V. Josyula, M. Orr, and G. Page, "Cloud Computing: Automating the Virtualized Data Center," Cisco Press, 2012, 392 pp., ISBN: 1587204347 (Safari Book).
- ❑ P. Saint-Andre, et al., "XMPP: The Definitive Guide," O'Reilly, 2009, 320 pp., ISBN:9780596521264 (Safari Book)

# References (Cont)

- ❑ <http://www.osgi.org/Technology/WhatIsOSGi>
- ❑ [http://www.sdncentral.com/sdn-use-cases /](http://www.sdncentral.com/sdn-use-cases/)
- ❑ <https://events17.linuxfoundation.org/sites/events/files/slides/OpenDaylight-Network-Programmability.pdf>
- ❑ <https://docs.opendaylight.org/en/stable-sodium/getting-started-guide/index.html>
- ❑ [https://wiki.opendaylight.org/view/Main\\_Page](https://wiki.opendaylight.org/view/Main_Page)
- ❑ OpenDaylight Components and Tools,  
<https://wiki.opendaylight.org>

# Acronyms

- ❑ ALTO Application Layer Traffic Optimization
- ❑ API Application Programming Interface
- ❑ APIC Application Policy Infrastructure Controller
- ❑ ASIC Application Specific Integrated Circuit
- ❑ BGP Border Gateway Protocol
- ❑ CDNI Content Distribution Network Interconnection
- ❑ CLI Command Line Interface
- ❑ COPS Common Open Policy Service
- ❑ CRC Cyclic Redundancy Check
- ❑ CRUD Create, Read, Update, Delete
- ❑ DLUX OpenDaylight User Interface
- ❑ DNI Name of a company
- ❑ FE Forwarding Element
- ❑ ForCES Forwarding and Control Element Separation
- ❑ HTTP Hypertext Transfer Protocol

## Acronyms (Cont)

- ❑ HW Hardware
- ❑ IaaS Infrastructure as a Service
- ❑ IDS Intrusion Detection System
- ❑ IGP Interior Gateway Protocol
- ❑ IoT Internet of Things
- ❑ IP Internet Protocol
- ❑ IPv4 Internet Protocol version 4
- ❑ IPv6 Internet Protocol version 6
- ❑ ISO International Standards Organization
- ❑ L2 Layer 2
- ❑ LDP Label Distribution Protocol
- ❑ LS Link State
- ❑ MPLS Multi-protocol Label Switching
- ❑ NetIDE Network Interactive Development Environment
- ❑ NETCONF Network Configuration

# Acronyms (Cont)

- ❑ NFV Network Function Virtualization
- ❑ NOS Network Operating System
- ❑ NVGRE Network Virtualization using Generic Routing Encapsulation
- ❑ OF OpenFlow
- ❑ OnePK Open Network Environment Platform Kit
- ❑ ONF Open Networking Forum
- ❑ ONiE Open Network Install Engine
- ❑ ONL Open Net Linux
- ❑ ONOS Open Networking Operating System
- ❑ ONV OpenDaylight Network Virtualization
- ❑ OS Operating System
- ❑ OSCP OpenDaylight SDN Controller Platform
- ❑ OSGi Open Services Gateway Initiative
- ❑ OSPF Open Shortest Path First
- ❑ OVS Open Virtual Switch
- ❑ OVSDB Open Virtual Switch Database

# Acronyms (Cont)

- ❑ PCE Path Computation Element
- ❑ PCEP Path Computation Element Protocol
- ❑ PIM Protocol Independent Multicast
- ❑ QoE Quality of Experience
- ❑ QoS Quality of Service
- ❑ REST Representational State Transfer
- ❑ RIP Routing Information Protocol
- ❑ SAL Service Abstraction Layer
- ❑ SAI Serial Audio Interface
- ❑ SDN Software Defined Networking
- ❑ SNMP4SDN SNMP for SDN
- ❑ SW Software
- ❑ TIA Telecom Industry Association
- ❑ TRILL Transparent Interconnection of Lots of Links
- ❑ URI Uniform Resource Identifier



## Acronyms (Cont)

- ❑ USB Universal Serial Bus
- ❑ VLAN Virtual Local Area Network
- ❑ VxLAN Virtual Extensible Local Area Network
- ❑ WAN Wide Area Network
- ❑ WWW World Wide Web
- ❑ XMPP Extensible Messaging and Presence Protocol

# Style Guide

Correct	Incorrect
ACL	Acl or acl
API	api
ARP	Arp or arp
IPsec	IPSEC or ipsec
IPv4 or IPv6	Ipv4, Ipv6, ipv4, ipv6, IPV4, or IPV6
Karaf	karaf
Linux	LINUX or linux
NETCONF	Netconf or netconf
Neutron	neutron
OSGi	osgi or OSGI
Open vSwitch	OpenvSwitch, OpenVSwitch, or Open V Switch.
OpenDaylight	Opendaylight, Open Daylight, or OpenDayLight.
OpenFlow	Openflow, Open Flow, or openflow.
OpenStack	Open Stack or Openstack
QoS	Qos, QOS, or qos
RESTCONF	Restconf or restconf
RPC	Rpc or rpc
URL	Url or url

Source: <http://docs.opendaylight.org/en/stable-nitrogen/documentation.html>

Washington University in St. Louis

<http://www.cse.wustl.edu/~jain/cse570-19/>

©2019 Raj Jain

# SDN Related Organizations and Projects

- ❑ Linux Foundation Open Source Networking, <https://www.linuxfoundation.org/projects/networking/>
- ❑ Open Networking Foundation (ONF): [www.opennetworking.org](http://www.opennetworking.org)
- ❑ Telecom Industry Association (TIA): [www.tiaonline.org](http://www.tiaonline.org)
- ❑ European Telecommunications Standards Institute (ETSI): [www.etsi.org/](http://www.etsi.org/)
- ❑ Association for Telecom Industry Solutions (ATIS): [www.atis.org/topsc/sdn.asp](http://www.atis.org/topsc/sdn.asp)
- ❑ Internet Engineering Task Force (IETF): [www.ietf.org](http://www.ietf.org)
- ❑ OpenStack Quantum: <https://wiki.openstack.org/wiki/Quantum>
- ❑ OpenDaylight: [www.opendaylight.org](http://www.opendaylight.org)

# SDN Web Sites

- ❑ SDN Central, <http://www.sdncentral.com>
- ❑ SDN Open Source Projects, <http://www.sdncentral.com/comprehensive-list-of-open-source-sdn-projects/>
- ❑ SDN Products and Services, <http://www.sdncentral.com/announced-sdn-products/>
- ❑ HotSDN 2012, <http://yuba.stanford.edu/~casado/of-sw.html>  
(Papers downloadable)

# Scan This to Download These Slides



Raj Jain

<http://rajjain.com>

# Related Modules



CSE567M: Computer Systems Analysis (Spring 2013),

[https://www.youtube.com/playlist?list=PLjGG94etKypJEKjNAa1n\\_1X0bWWNyZcof](https://www.youtube.com/playlist?list=PLjGG94etKypJEKjNAa1n_1X0bWWNyZcof)

CSE473S: Introduction to Computer Networks (Fall 2011),

[https://www.youtube.com/playlist?list=PLjGG94etKypJWOSPMh8Azcg5e\\_10TiDw](https://www.youtube.com/playlist?list=PLjGG94etKypJWOSPMh8Azcg5e_10TiDw)



Wireless and Mobile Networking (Spring 2016),

[https://www.youtube.com/playlist?list=PLjGG94etKypKeb0nzyN9tSs\\_HCd5c4wXF](https://www.youtube.com/playlist?list=PLjGG94etKypKeb0nzyN9tSs_HCd5c4wXF)

CSE571S: Network Security (Fall 2011),

<https://www.youtube.com/playlist?list=PLjGG94etKypKvzfVtutHcPFJXumyyg93u>



Video Podcasts of Prof. Raj Jain's Lectures,

<https://www.youtube.com/channel/UCN4-5wzNP9-ruOzQMs-8NUw>