
University of Colorado Boulder

Network Management and Automation

Fundamentals of Software-Defined Networking (SDN) and Network Functions Virtualization (NFV)

Levi Perigo, Ph.D.
University of Colorado Boulder
Department of Computer Science
Network Engineering

Review

- **Network Automation**
- **DevOps**

Lecture

- **Overview**
- **Internships/Full-time**
- **No SDN Lab**
- **New way of thinking**
- **Definition of SDN?**

Software-Defined Networking (SDN)

- **Separating intelligence from hardware to software**
  - SANS – data storage with a management system
  - Computers - Virtualization of OS

- **Networking equipment was left behind**
  - Routers, switches, routing protocols
    - ***Faster, more throughput, bigger backplane, etc.; basic intelligence are still the same***
  - Making networking equipment "dumb," but creating a management system to make the overall system more intelligent
    - ***One control panel for all networking hardware***
      - Control the entire network via "intent"

History

- **20 years ago, all we cared about was size of the "pipe"**
  - Transferring files
  - Speed!
- **15 years ago**
  - Youtube
  - Skype
  - VoIP
  - Realtime Communications
    - ***Latency / jitter – (not files; but packets)***
      - 10 Mbps vs 20 ms
    - ***QoS – prioritize packets***
- **10 years ago**
  - More complicated
    - ***FTP vs SIP (prioritize)***
      - Sometimes! Variables; QoS isn't dynamic
- **Currently**
  - Automation
  - Scale
- **Major Roadblocks**
  - Practical security
  - Performance concerns
  - Vendor lock-in

Now

- **Dynamically shape global traffic based on what we need to do**
- **Separating out the different components so we can deal with them differently**
- **Programmability and Scale**
- **Separate networking**
  - Control plane (“Brains”)
    - ***Management servers that communicate with all of the hardware***
    - ***How should data move “at this second”***
      - VoIP gets priority over FTP
      - Now video gets priority over VoIP
  - Data plane (“Muscle”)
    - ***Switches and router hardware***

SDN Concept

- **Originally created by researchers and data center architects to scale and automate data center networks**

- **Centralized SDN Controller**
  - Controls how Network Devices forward packets, & provides an abstract, centralized view of the overall network

- **Flow Switching**
  - The new paradigm to underpin classic L2/3 switching/routing, VPNs, etc.

- **Network devices move packets between ports according to policies/tables from SDN Controllers**
  - Packets that require special handling go to the SDN controller for processing

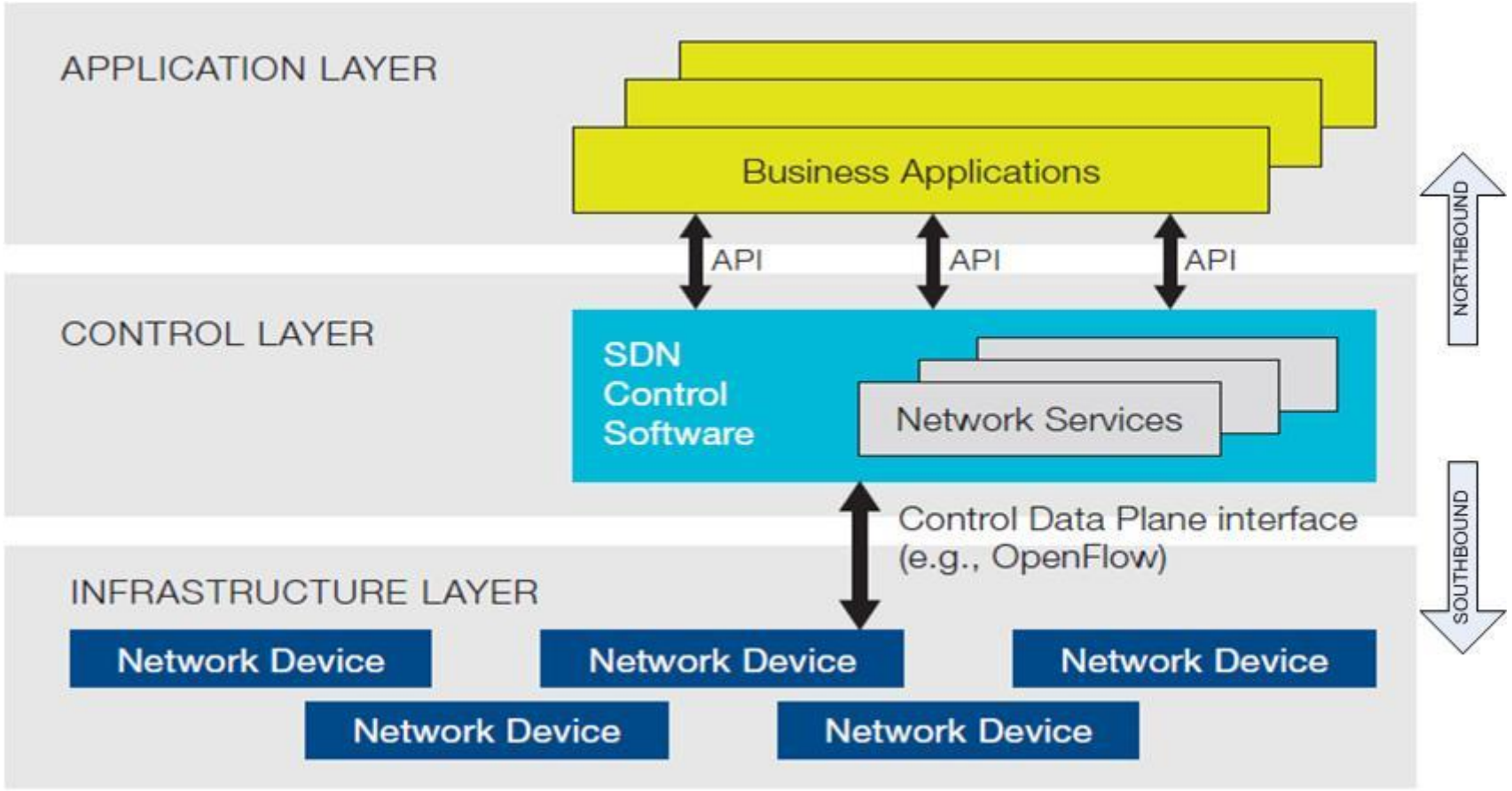
SDN Benefits

- **Multi-Tenancy**
- **Automated Deployment & Operations**
  - Reduces OpEx
- **Enable Innovation**
- **Rapidly Deploy New Applications, Services and Infrastructure**
  - Add new application (software)
    - ***TTM***

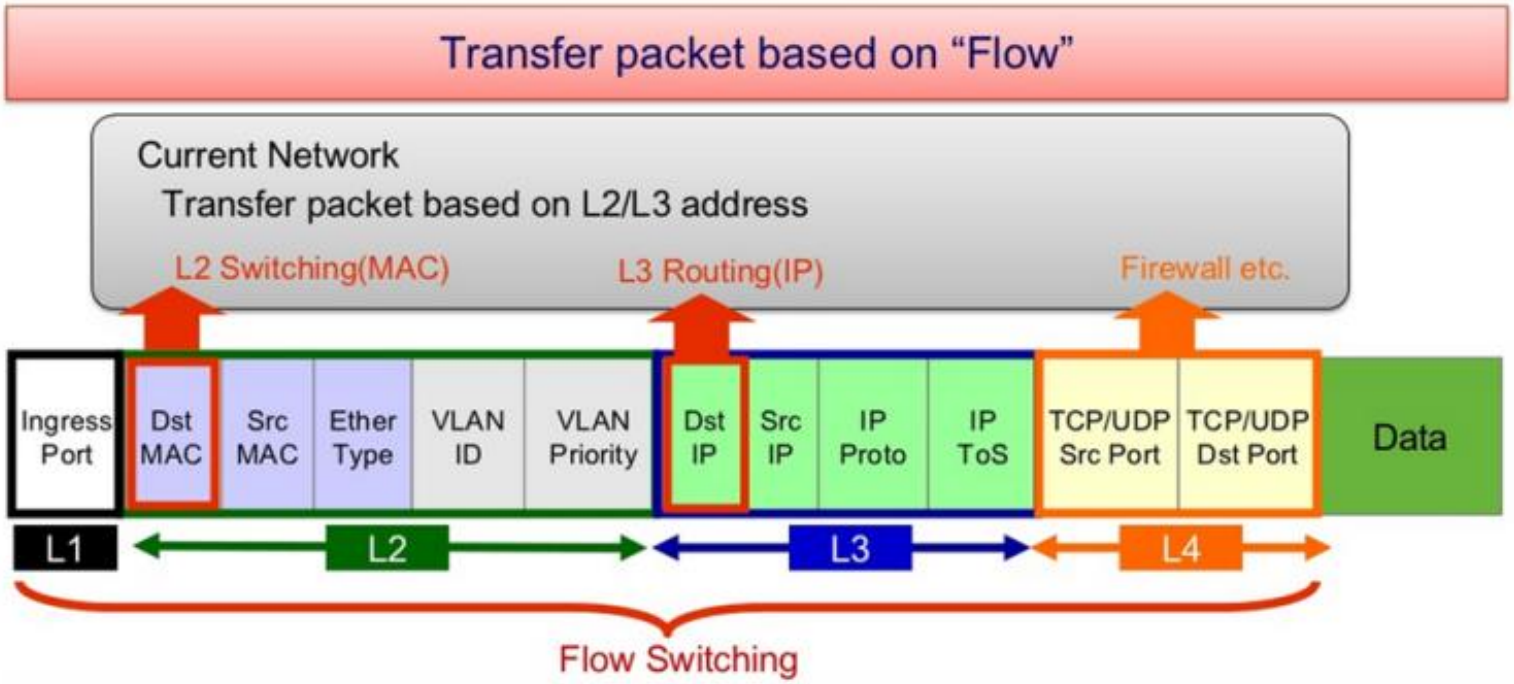
SDN Benefits

- **Whitebox/bare metal switches IaaS**
  - Reduces CapEx
- **Improved network efficiency and control**
- **Reduce CapEx/OpEx, TTM, & TCO**
- **Global view of network**

SDN Diagram


APPLICATION LAYER
Business Applications
API
API
API
NORTHBOUND
CONTROL LAYER
SDN
Control
Software
Network Services
SOUTHBOUND
Control Data Plane interface
(e.g., OpenFlow)
INFRASTRUCTURE LAYER
Network Device
Network Device
Network Device
Network Device
Network Device

Flow Switching



Key Terms and Buzzwords

- **Control Plane**
- **Data / Forwarding Plane**
- **OpenFlow**
  - protocol used to install flows, flow tables in Ternary Content Addressable Memory (TCAMs) and ASIC
- **OpenDaylight**
  - an open, standards-based SDN controller platform project hosted by the Linux Foundation
- **Open vSwitch (OvS)**
  - an open-source virtual switch
- **NFV - Network Functions Virtualization**
- **MEF (Metro Ethernet Forum)**
- **ONF - Open Networking Foundation (ONF)**

Key Terms and Buzzwords

- **OpenStack**
  - an open source “cloud operating system” used to create and manage pools of cloud resources (orchestrator)
- **Orchestration**
  - a system to automatically create, initialize, coordinate & manage physical and virtual resources for cloud service delivery
- **Mininet**
  - An instant, virtual overlay network on your machine

SDN Protocols

- **OpenFlow**
  - a fairly low-level protocol used to configure flows, flow tables and TCAMs in network devices.
- **NETCONF**
  - a device (physical or virtual) configuration and management protocol. Allows for configuration, data retrieval and event notification.
  - **OF-Config**
    - The ONF specification for configuration of devices in an OpenFlow network. Specifies NETCONF over TLS with XML data models (specifically recommends YANG).
- **XMPP**
  - Extensible Messaging and Presence Protocol is used by some (i.e. Juniper) as an alternative to OpenFlow. Preferred by some due to its maturity as a protocol.
- **OpFlex**
  - A Cisco proprietary protocol proposed as an alternative to OpenFlow. Departs from a basic tenant of SDN in that control intelligence remains in the networking infrastructure.

SDN Definition - ONF

- SDN is an emerging network architecture where network control is decoupled from forwarding and is directly programmable. This migration of control, formerly tightly bound in individual network devices, into accessible computing devices enables the underlying infrastructure to be abstracted for applications and network services, which can treat the network as logical or virtual entity. Network intelligence is (logically) centralized in software-based SDN controllers, which maintain a global view of the network.

Traditional, Open SDN Definition

- **Key Themes**
  - Decoupling the control from the forwarding plane
  - Software programmability of network elements (abstraction and agility)
  - Centralized network control
  - Open standards-based and vendor neutral

Traditional Router

- Each router has own “brain”
  - Example: Each router has own routing protocol FIB, and needs to communicate to all the other routers (or neighbors)
    - Link-state: knows about “every” link in the network
- Each hardware device has OS features it supports; load-balancing, firewall, NAT, VPN ($), MPLS ($), voice ($), etc.
- Instability in the control plane, which results in long convergence times when changes occur in the network
  - One slow node/processor in the network can slow rest of network
    - STP, OSPF, MPLS

Data and Control Plane

- **Data plane**
  - Hardware

- **Control plane**
  - Servers that manage all the services and data
    - ***More powerful options when using "super" servers/processors***
  - "Centralized/global" view of the network
    - ***Logically centralized, physically distributed***

OpenFlow

- "Control" protocol
  - ***Communication between Control and Data plane (southbound)***
  - ***Non-proprietary protocol for programming the data plane flows***
- Standardized way of describing the data plane actions
- Device reference model that describes the separation of the control and data plane
- Allows direct access to and manipulation of the forwarding plane of network devices such as switches and routers
- Vendor-neutral, Open standard managed by ONF

OpenFlow Overview (diagram)

Why would we implement SDN?

- **What if we have big powerful networking equipment?**
  - Boundaries/boxes
    - ***VoIP network has limited amount of bandwidth***
    - ***Video surveillance has limited amount of bandwidth***
    - ***Computers/Servers has limited amount of bandwidth***
    - ***Can't use bandwidth that isn't being used***
  - One pipe that is allocated to us all the time, then depending on our needs, different services can use different speeds at different times

Why would we implement SDN?

- **Ossification**
  - Tendency or state of being molded into a rigid, convention, sterile, or unimaginative condition
- **Open Source / vendor neutral**
- **Reduce vendor lock-in**

Why would we implement SDN?

- **Cost**
  - CapEx
    - ***Physical devices and multiple-use devices***
  - OpEx
    - ***Automation/programmability and orchestration***
  - TTM
  - TCO

Real World SDN Example

- **4K video**
  - 1,000 nodes
  - malicious intent
  - avoid person
  - video alert system streams to every device on network in 4K video/audio what the person is doing at that second, and where they are

  - Tremendous amount of bandwidth!
    - ***QoS, division of bandwidth, etc.***
    - ***Turn entire network over to one "emergency" service (i.e. "best effort" services (HTTP) etc.)***
      - Now switch priority back to VoIP/wireless so they can call loved ones

Current Status

- **The "new VoIP"**
  - SDN is what VoIP was 15-20 years ago
  - Everyone wants SDN and is using SDN, but everyone has a different view of how to implement it and what it is (i.e. VoIP - SIP eventually won)
    - ***SDN via OpenFlow; via APIs; via Overlays***

- **Buying equipment**
  - Early proprietary VoIP protocols failed
  - Juniper's view is different from Cisco's
    - ***They will solidify on a model***
  - Caution on "SDN equipment"

Control Plane

- **Big deal?**

- **Server or cluster of servers**
  - Management console
    - ***Commands to all devices to change real-time***
    - ***Automated applications***
    - ***APIs***
      - Software/applications on network connects to control plane and tells it to run more efficiently
      - Individual devices/apps can automatically make the network function in a different way!!!

Security

- **Hardware devices (routers, switches, firewalls, etc.)**
  - Stable
- **Services**
  - Servers (Web, Mail, DNS, etc.)
- **SDN**
  - Devices can control pieces of your network!
    - ***Con***
      - "SDN App Store" - Hacker: I want my application to have 100% priority; and nothing for mission critical traffic
    - ***Pro***
      - Think of what you can do with a global view!

Security Challenges

- **BYOD**
  - Consumer electronics on your network
  - SDN has API
    - ***Devices connect and control network***
    - ***Viruses and Malware***
      - iPad example
        - » User downloads “SDN App” from App store
          - Not just the device; the entire infrastructure!

- **Routers don’t get viruses!**

ISPs

- **WAN**
  - Priced based on time of use and quality of use

- **ISP has costs associated with services**
  - Bandwidth - 10 Mbps; 100 Mbps; 1 Gbps
  - Latency and Quality
    - ***Large bandwidth/high latency vs. small bandwidth/low latency***
    - ***Stock Market Traders example (latency)***
      - 6am – 6 pm
        - » What happens at 7 pm? They need bandwidth, but not low latency

- **SDN Fixes this!**
  - Low bandwidth, high latency after hours
  - High bandwidth, low latency during the day

Dynamically Used

- **No longer all about bandwidth**
- **More like electricity**
  - Pay for use

Networking Technology Limitations

- **Network complexity that leads to “status”**
- **Inconsistent policies**
  - Different vendors, configurations, etc.
- **Inability to scale**
  - OSPF, MAC address table
- **Vendor dependence (lock-in)**

New Network Architecture

- **Changing traffic patterns**
- **Consumerization of IT (BYOD)**
- **Cloud services**
- **Big data = more bandwidth**

SDN Architecture

- **Directly programmable**
- **Agile**
- **Centrally managed**
  - Logically centralized, but physically distributed
- **Programmatically configured (automation)**
- **Open standards-based and vendor-neutral**

SDN & NFV

- **SDN programs network flows**
- **NFV programs network functions**

Network Functions Virtualization (NFV)

- While SDN is focused on network programmability, NFV focuses on taking network functions and converting them to software applications that run in compute resources.

Services Plane

- **Services are on their own "plane" on a SERVER**
  - Firewall, router, load-balancer, etc.
    - ***Cisco proprietary***
  - x86 – Intel platform
    - ***Open standards***
    - ***Programming in different languages (Java, Python, etc.)***
      - Doing things/network services you've never done before!!
        - » You know what service I think we need on our network?! . . . . IMPLEMENT IT!

NFV

- Eliminates proprietary, high-cost hardware while enabling instant activation of network services
- Virtual Network Functions (VNFs) offer great flexibility in service offerings while improving network automation
- NFV Infrastructure (NFVI) can be hosted in any IaaS environment and shared with cloud services, provided by 3rd parties

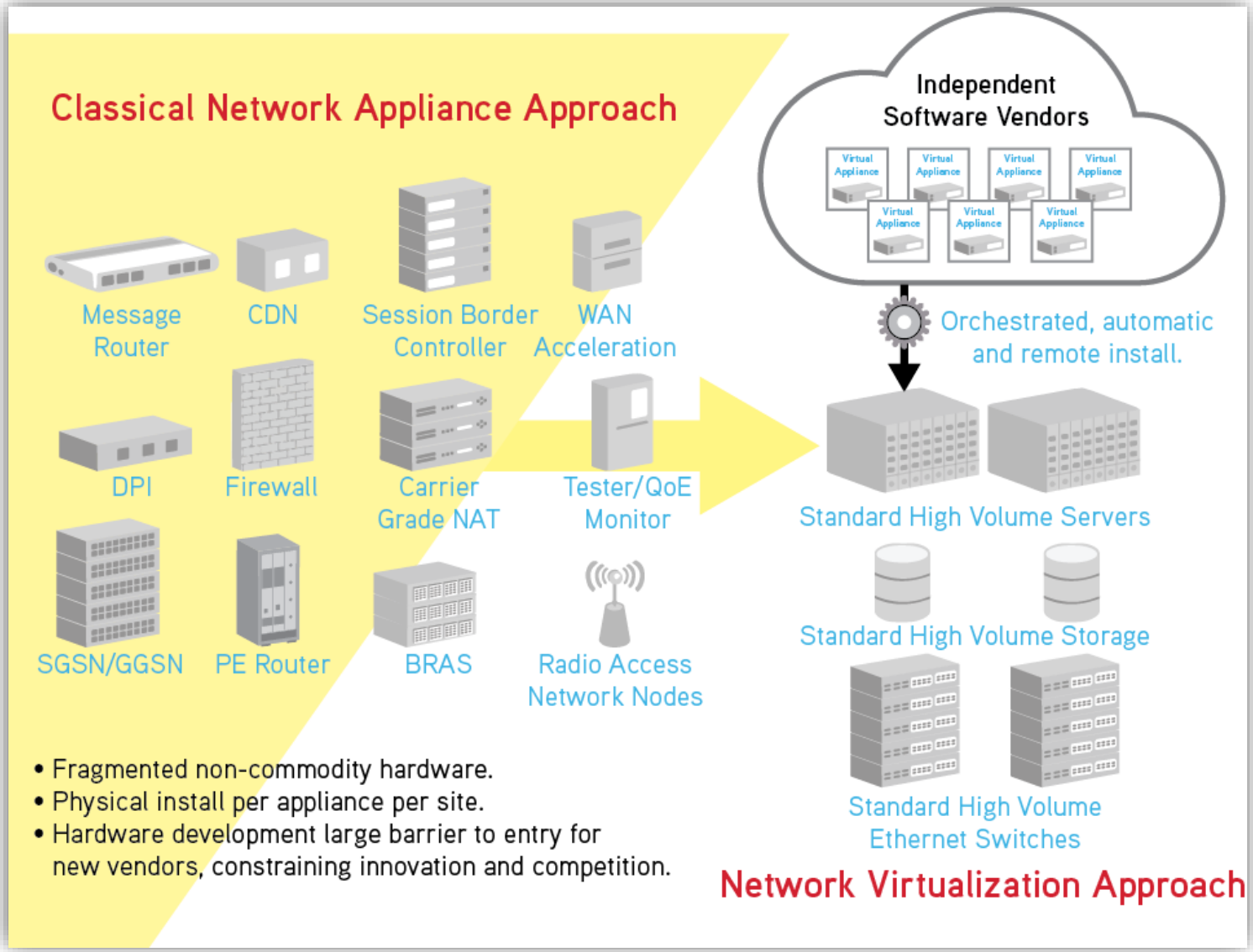
NFV Cost Comparison

NFV Definition - (ETSI)

- Network Functions Virtualization aims to transform the way that network operators architect networks by evolving standard IT virtualization technology to consolidate many network equipment types onto industry standard high-volume servers, switches, and storage, which could be located in datacenters, network nodes and in the end user premises. It involves the implementation of network functions in software that can run on a range of industry standard server hardware, and that can be moved to, or instantiated in, various locations in the network as required, without the need for installation of new equipment.

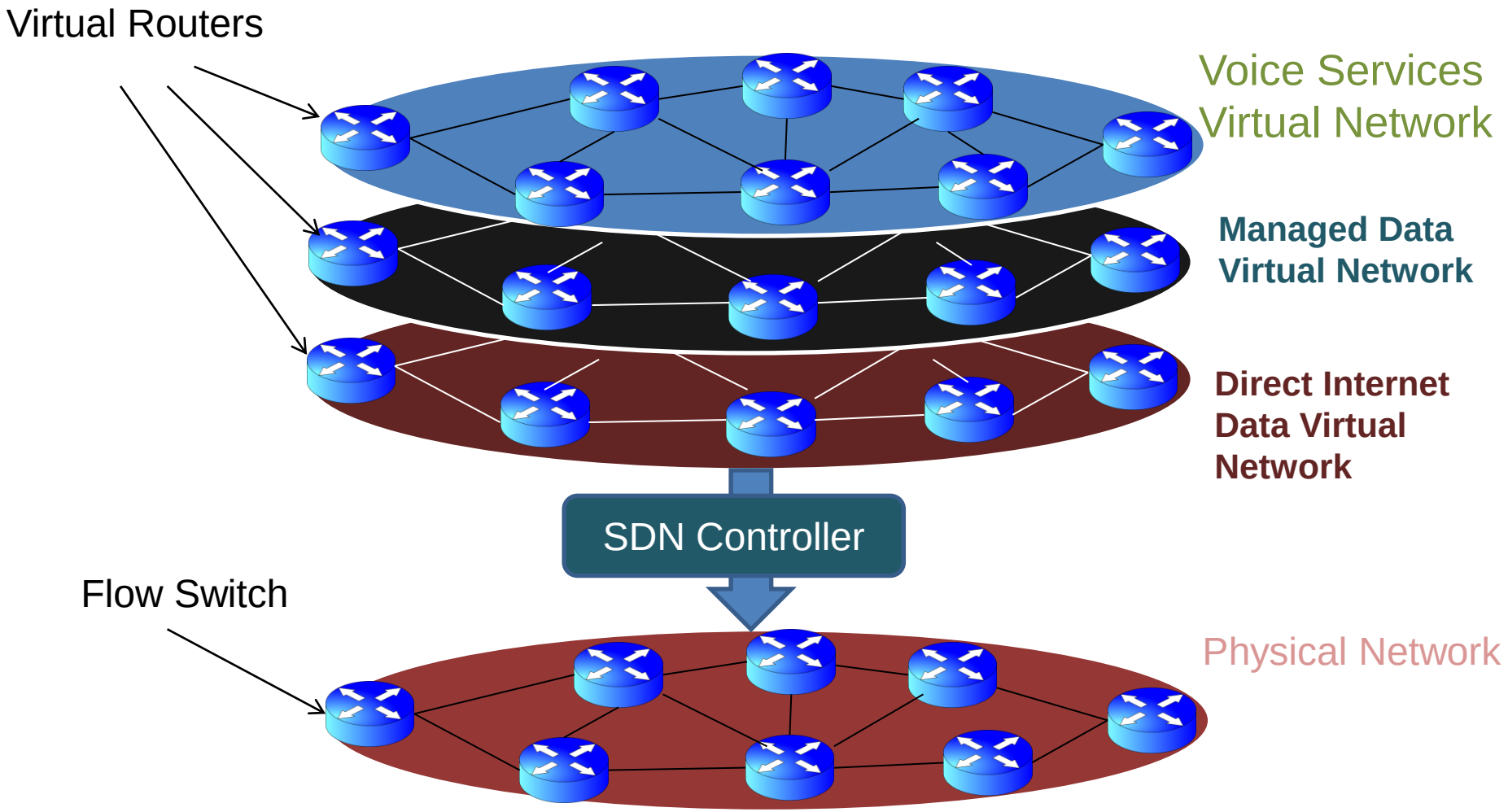
NFV Definition

- **Key Themes**
  - Consolidate network equipment onto industry standard servers, switches, storage (data centers)
  - Network functions in software
  - Located in various locations in the network; without installation of new equipment


Classical Network Appliance Approach
Message Router
CDN
Session Border Controller
WAN Acceleration
DPI
Firewall
Carrier Grade NAT
Tester/QoE Monitor
SGSN/GGSN
PE Router
BRAS
Radio Access Network Nodes
• Fragmented non-commodity hardware.
• Physical install per appliance per site.
• Hardware development large barrier to entry for new vendors, constraining innovation and competition.
Independent Software Vendors
Virtual Appliance
Orchestrated, automatic and remote install.
Standard High Volume Servers
Standard High Volume Storage
Standard High Volume Ethernet Switches
Network Virtualization Approach

Central Office Re-architected as Data Center (CORD)

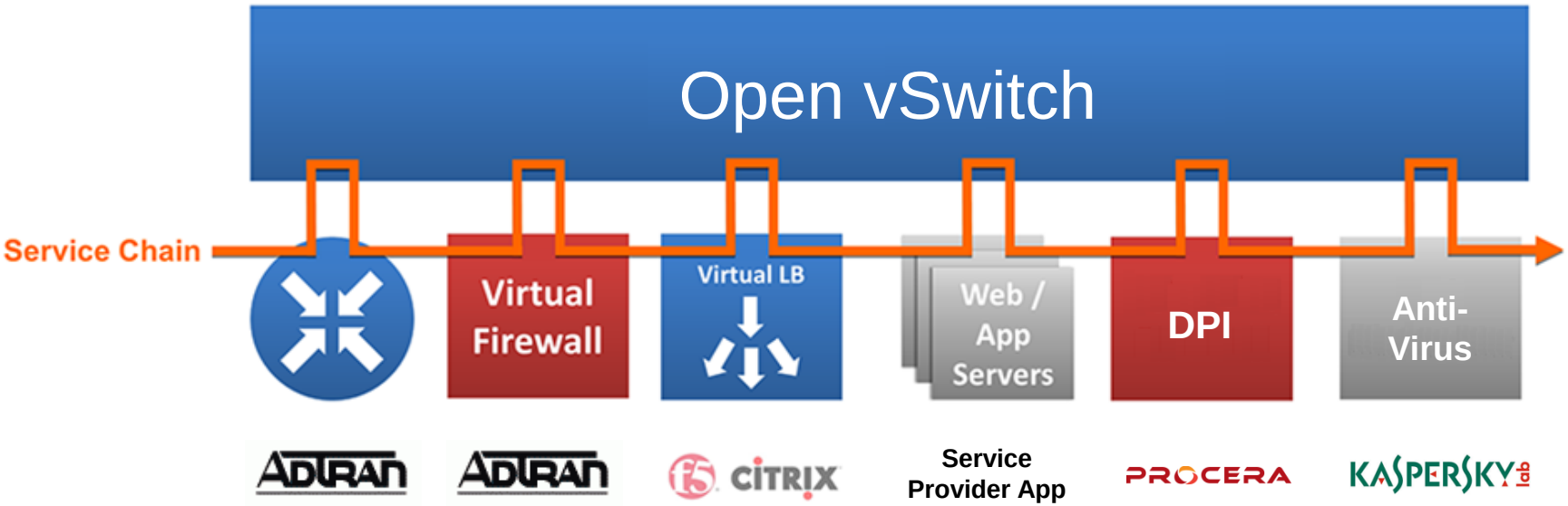
Network Virtualization (NV)



Service Chaining

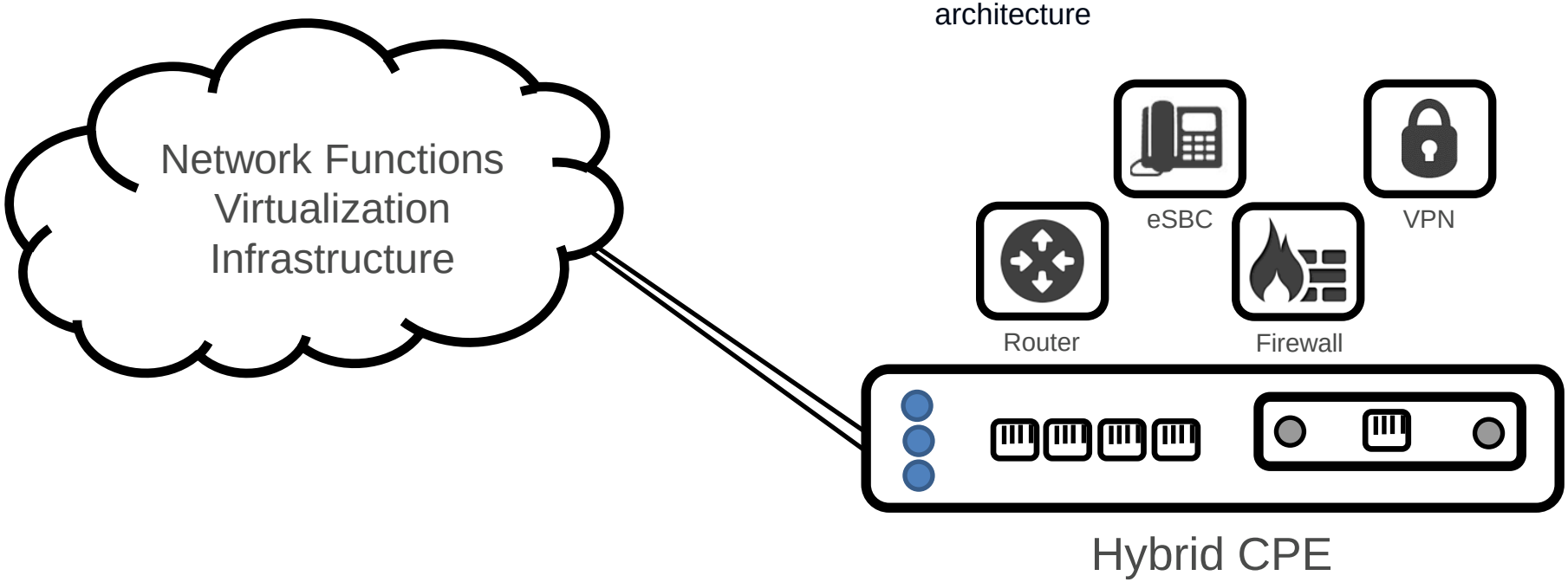
- **Network service chaining allows providers to define services for customers**
  - Allows providers to select and link Virtual Network Functions (VNFs) from multiple vendors
  - Utilizes orchestration systems and SDN to automatically turn up new services

Service Chaining

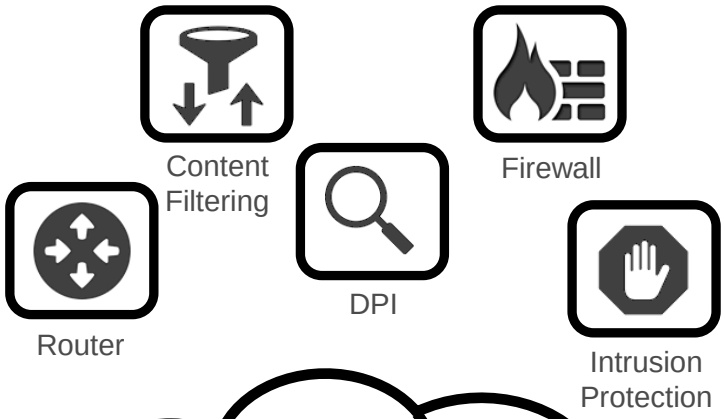


Hybrid CPE

- **Benefits**
  - Allows for the eventual migration of networking functions to the cloud
  - Providers can leverage today's CPE investment into tomorrow's NFV architecture

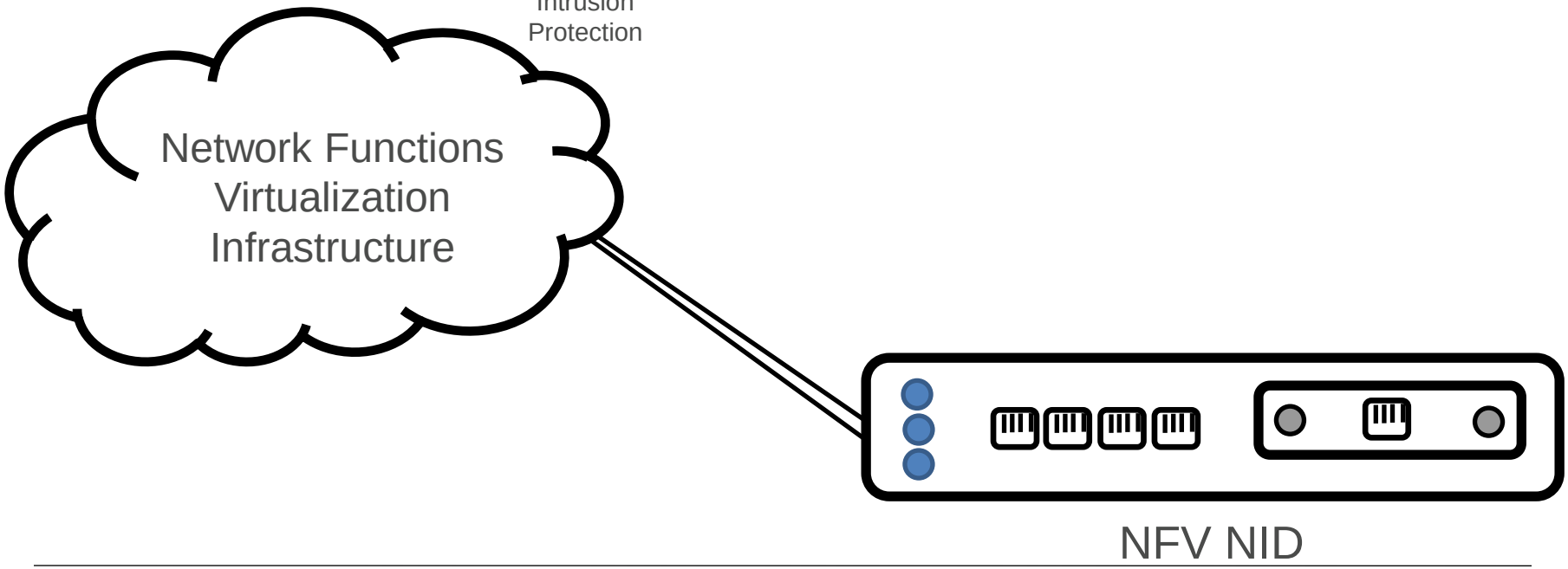


NFV NID



Benefits

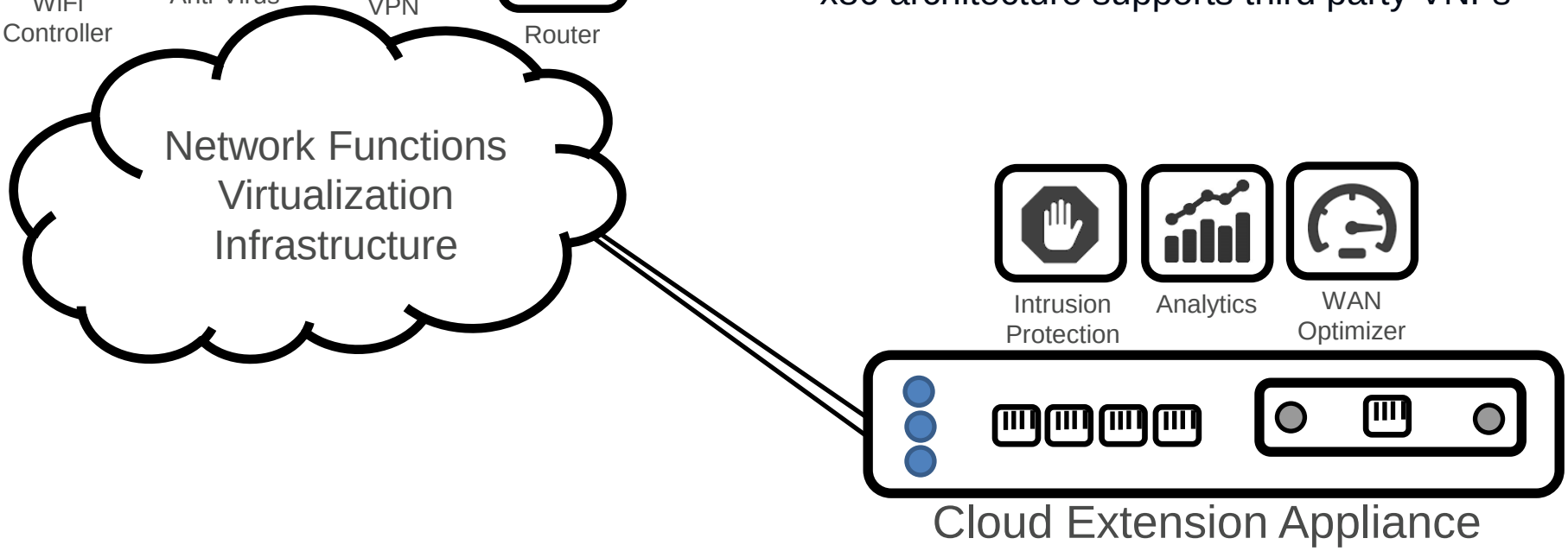
- Centralizes VNFs in the cloud
- Increases provider agility and by enabling expansion of business services through service chaining
- Lowest cost solution for Greenfield NFV installations



Cloud Extension Appliance

Benefits

- Maximum Flexibility – allows provider to deploy VNFs and applications in the optimal location
- Bolsters managed service offerings by providing for on-premises compute and storage resources
- x86 architecture supports third party VNFs



Challenges for SDN and NFV

- **Evolving standards, competing schools of thought and industry politics**
  - Needs to ensure open standards
- **Migration, integration, and coexistence of SDN and NFV models into existing networks**
- **Virtual Network Function (VNF) testing and deployment**
  - Multi-vendor forwarding graph/service chaining
  - Ability to specify attributes of individual VNFs
  - Testing, measurement, and validation of service performance
- **Security**
  - One device vs. Many; which is better?

Summary

- SDN and NFV are game changing technologies that will impact the way we think about network and service design
- These technologies will allow service providers to automate, reduce OpEx and become more agile in service definition and deployment
- SDN implementations are mainly in the data center and service provider networks but will find their way into broader networking applications over the next few years
- There is still a lot of work to be done but vendors and service providers are partnering together to define open standards

Summary

- The control and data planes are decoupled
- Forwarding decisions are flow-based instead of destination-based
- Control logic is moved to an external entity, called SDN controller
- Network is programmable through applications running on top of the controller (the fundamental characteristic of SDN)

Questions?