SDN APIs for Communications

When Applications and the Network Talk with Each Other

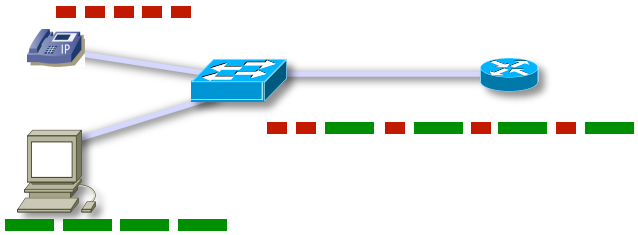
Terry Slattery
Principal Architect
NetCraftsmen
CCIE #1026

Agenda

- History of Traditional Networking
- SDN and UC
- REST API
- Future Directions and Summary

Traditional Networking

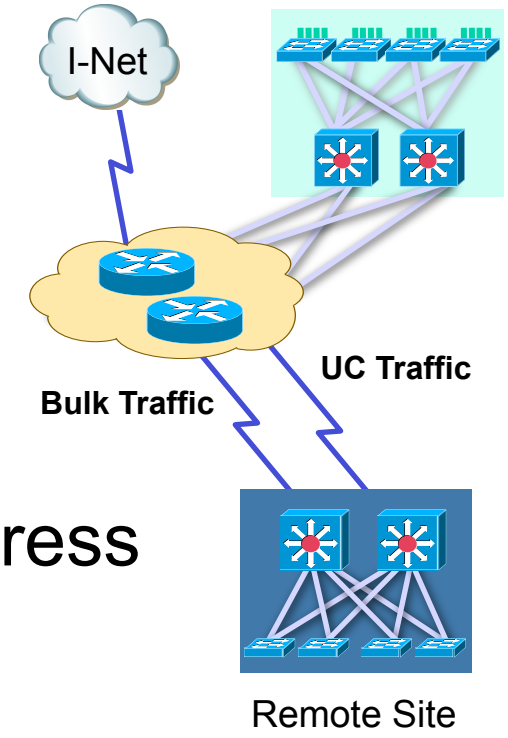
- Complex configurations
  - Low-level CLI commands
  - Non-intuitive interactions
  - QoS example:




```
ip access-list extended QOS-LOW-LATENCY-DATA
 remark Latency sensitive Data application traffic
 permit tcp host 10.1.1.2 any any
 permit tcp host 10.1.1.4 any any
 deny ip any any
…
class-map match-any OUT-LOW-LATENCY-DATA
 description Low-Latency Data
 match access-group QOS-LOW-LATENCY-DATA
 match ip dscp af21
…
policy-map OUT-QUEUING-REMOTE1
  description Outbound queuing and scheduling
  class OUT-LOW-LATENCY-DATA
    bandwidth percent 20
    queue-limit 100
    random-detect
```

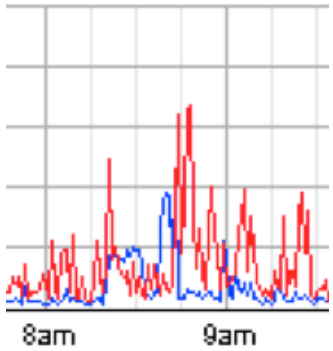
Traditional Networking

- Relatively static configuration
  - Difficult to synchronize with business needs
  - Not responsive to applications
- Forwarding based on destination address
  - Drives equal cost multi-path topologies
  - Complex policy routing configuration to implement policy routing



Traditional Networking

- Low network utilization – 30-40%
  - Reserve bandwidth for traffic bursts
  - Unable to manage traffic with sufficient granularity



- Applications and the network don't communicate
  - Apps can't ask the network for special service
  - The network can't inform apps of network changes
  - App and network teams often don't work well together
  - Need special application performance monitoring systems

The Network Is Not Agile

- Compute and storage are very agile

  - VMs can be created and moved within minutes
  - Containers will accentuate the difference (they activate in seconds)

  - Network changes require days or weeks
    - Change control systems induce delays
    - Changes implemented by manual processes
    - Network staff is often reluctant to use automation

We Need Something Better

- Bidirectional communications between applications and the network
- Faster configuration through automation
- Simplify configurations with powerful abstractions
- Better security (built-in and provable)
- New forwarding path selection mechanisms

Agenda

- Traditional Networking
- SDN and UC
- REST API
- Future Directions and Summary

SDN Overview

- What is Software Defined Networking?
  - A new form of networking
  - Decouples control from packet forwarding
  - Software control of the network
  - Abstractions hide details of the infrastructure layer
  - Network and applications communicate with each other

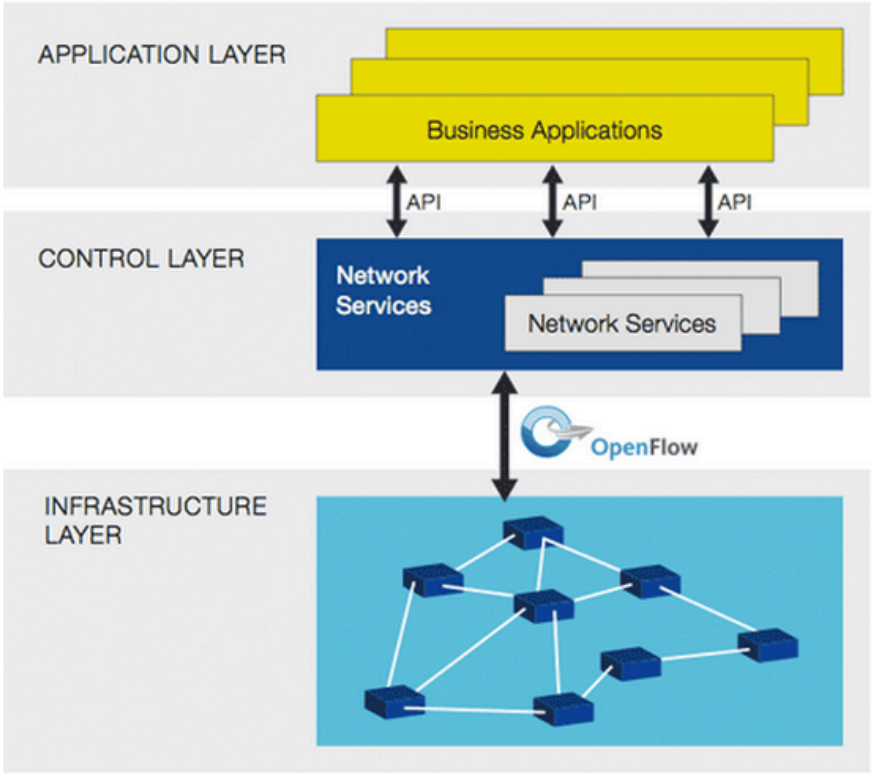
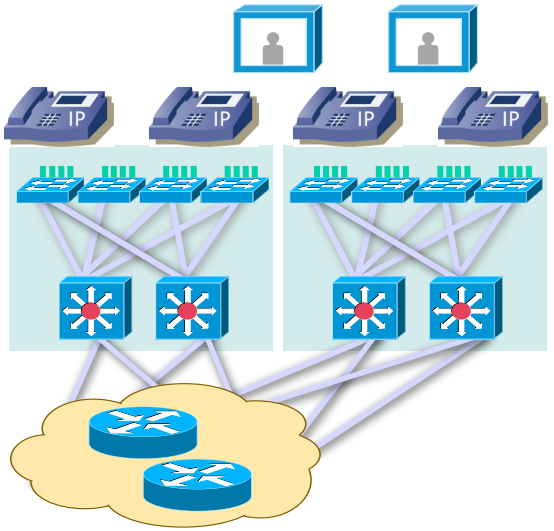


Image: Open Networking Foundation

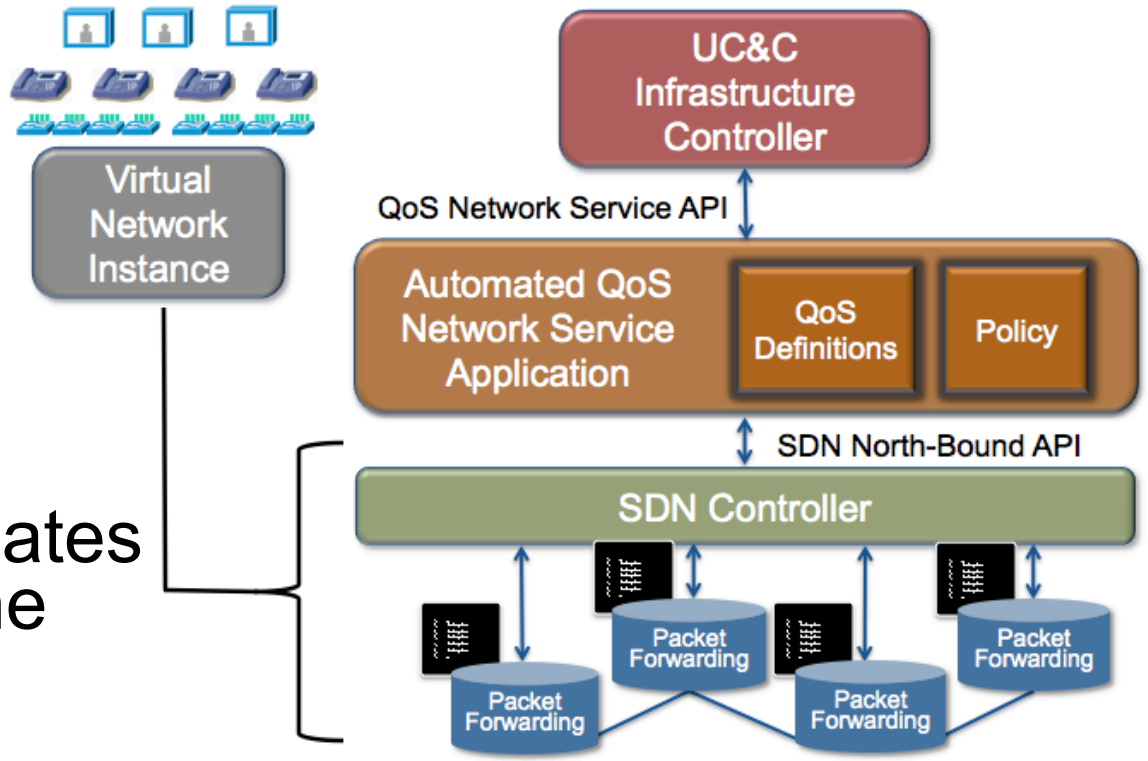
The Advantages of SDN

- Applications and Network communicate
- Networks become more dynamic and agile
- Centralized control system makes better decisions
  - Programming is easily done across multiple devices
- Packet forwarding based on more than destination address
- New path selection protocols



Architecture of SDN for UC

- UC controller talks with QoS Service App
- QoS Service App talks with the SDN controller
- SDN controller creates a VNI to contain the UC infrastructure



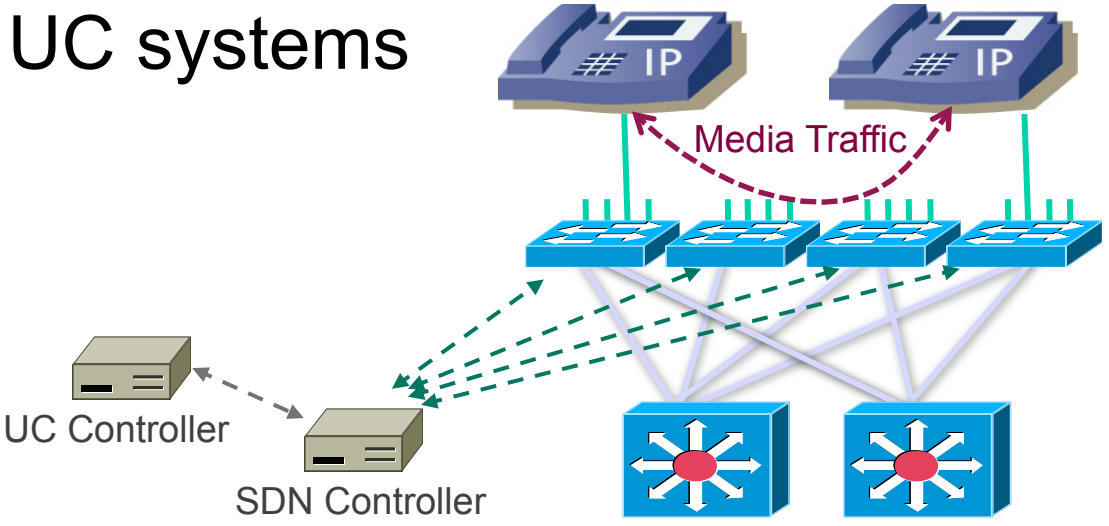
SDN Functions for Unified Communications

- Dynamic QoS
  - Apply QoS classification and marking at call setup
- Call Admission Control
  - Integrated CAC across multiple UC applications
- Dynamic traffic engineering
  - Dynamic path selection
- Policy control
  - Controls details of the SDN automation system

Ref: IMTC “Automating Unified Communications Quality of Experience using SDN”

Dynamic QoS Classification and Marking

- UC controller identifies media traffic endpoints
  - Based on 5-tuple: Src IP+port, Dst IP+port, Protocol (UDP/TCP)
- Works with encrypted media traffic (Lync & Skype)
- Handles multiple UC systems
  - Lync + Polycom



HP Unified Communications SDN

Manfred Arndt, Chief Technologist - UC & Mobility

March 18, 2015

Today's UC&C Challenges

Network issues cause 60% to 80% of poor end-user QoE

Legacy Networks have poor visibility into real-time traffic

- Lync uses encryption by default, making DPI difficult and unreliable
- Skype tries to hide itself from networks

Traffic engineering & QoS is complex…easily broken

- Requires brute force static policies that must match application server settings
- Intermittent problems are tedious to diagnose, especially for Softphones and BYOD

QoS engineering has to be managed consistently *end-to-end*, or it can have a negative impact for *all voice and video traffic*

Software Defined Network (SDN) Vision

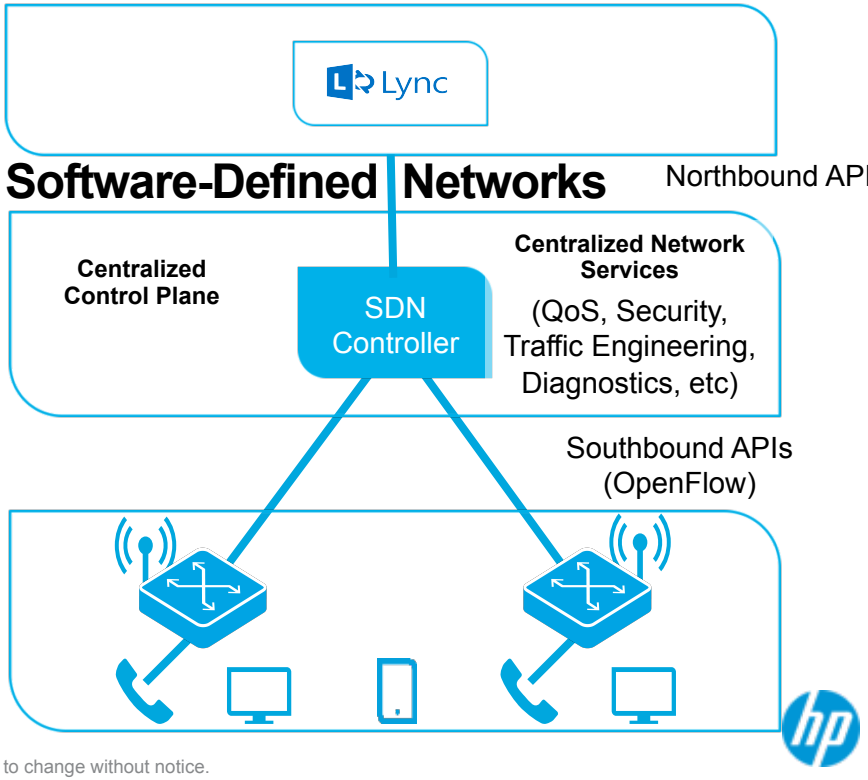
Make Applications & Networks Play Better Together

"Higher-layer application functions will become integrated with lower layers of the network, leading to two-way application awareness. The network will be able to adapt to changing application requirements efficiently and effectively."

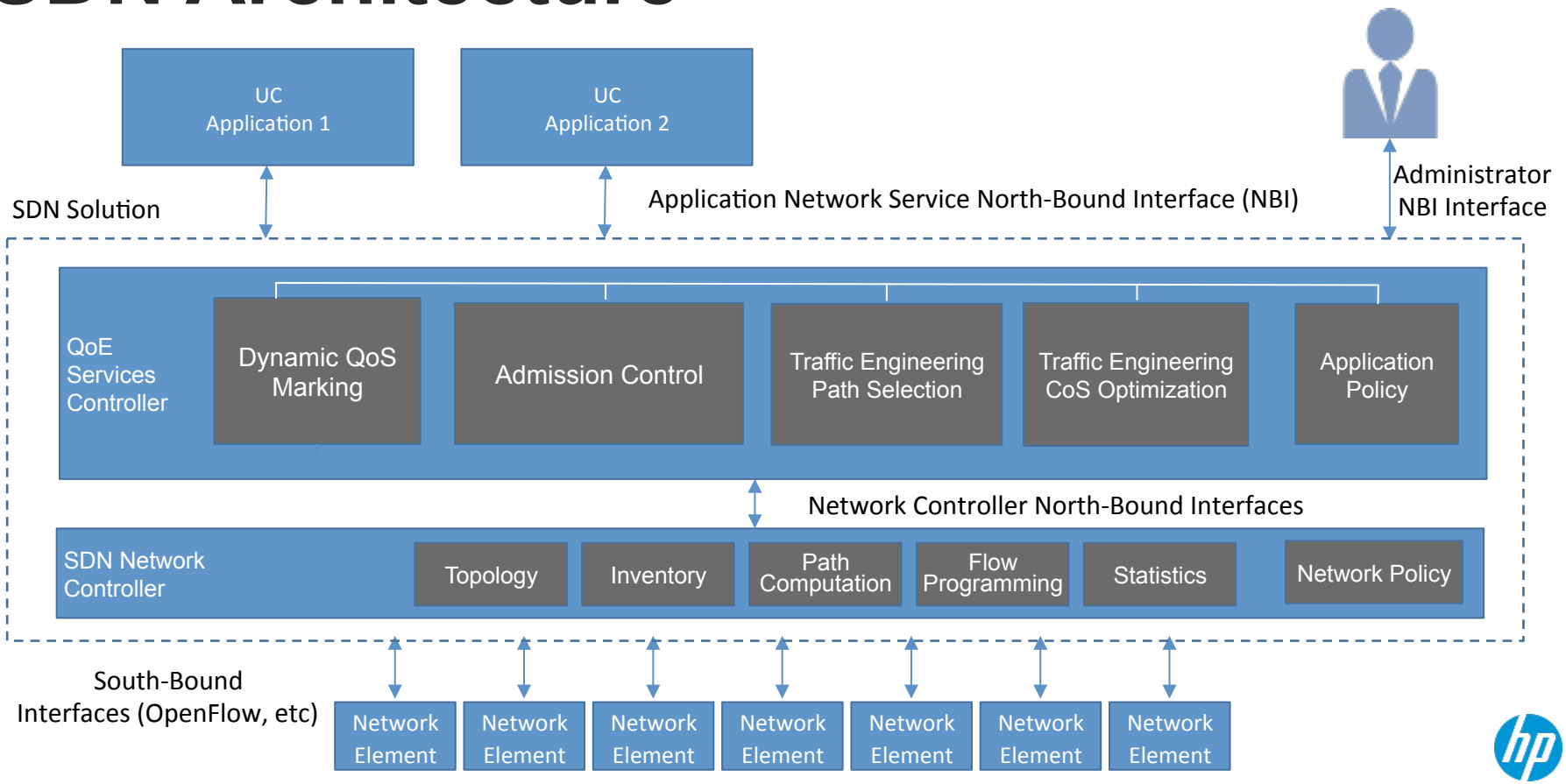
– Julie Kunstler, Ovum Research
March 18, 2013

In other words, they will work together collaboratively to create
application directed networks

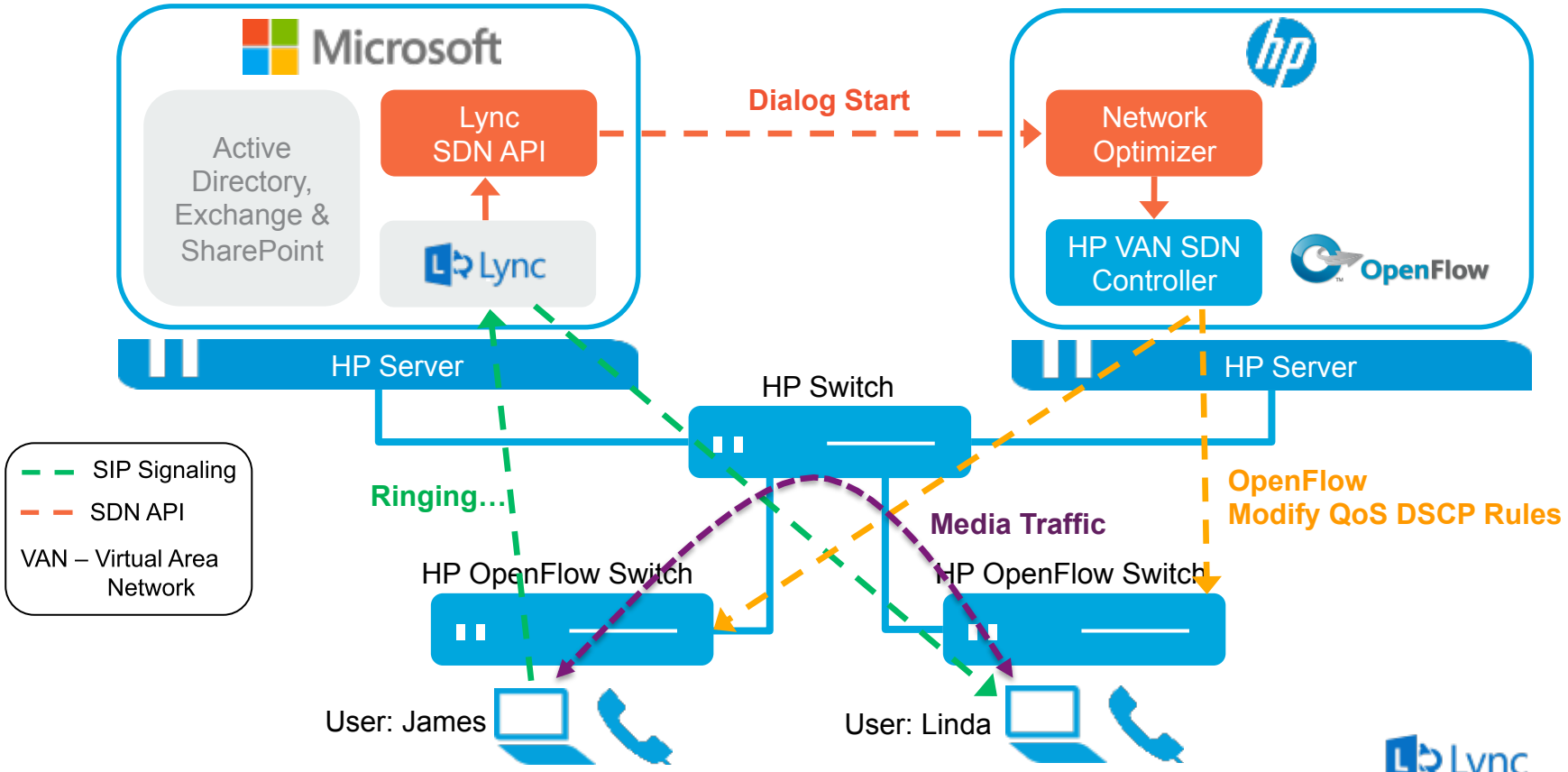
End-User Applications Talking to Networks



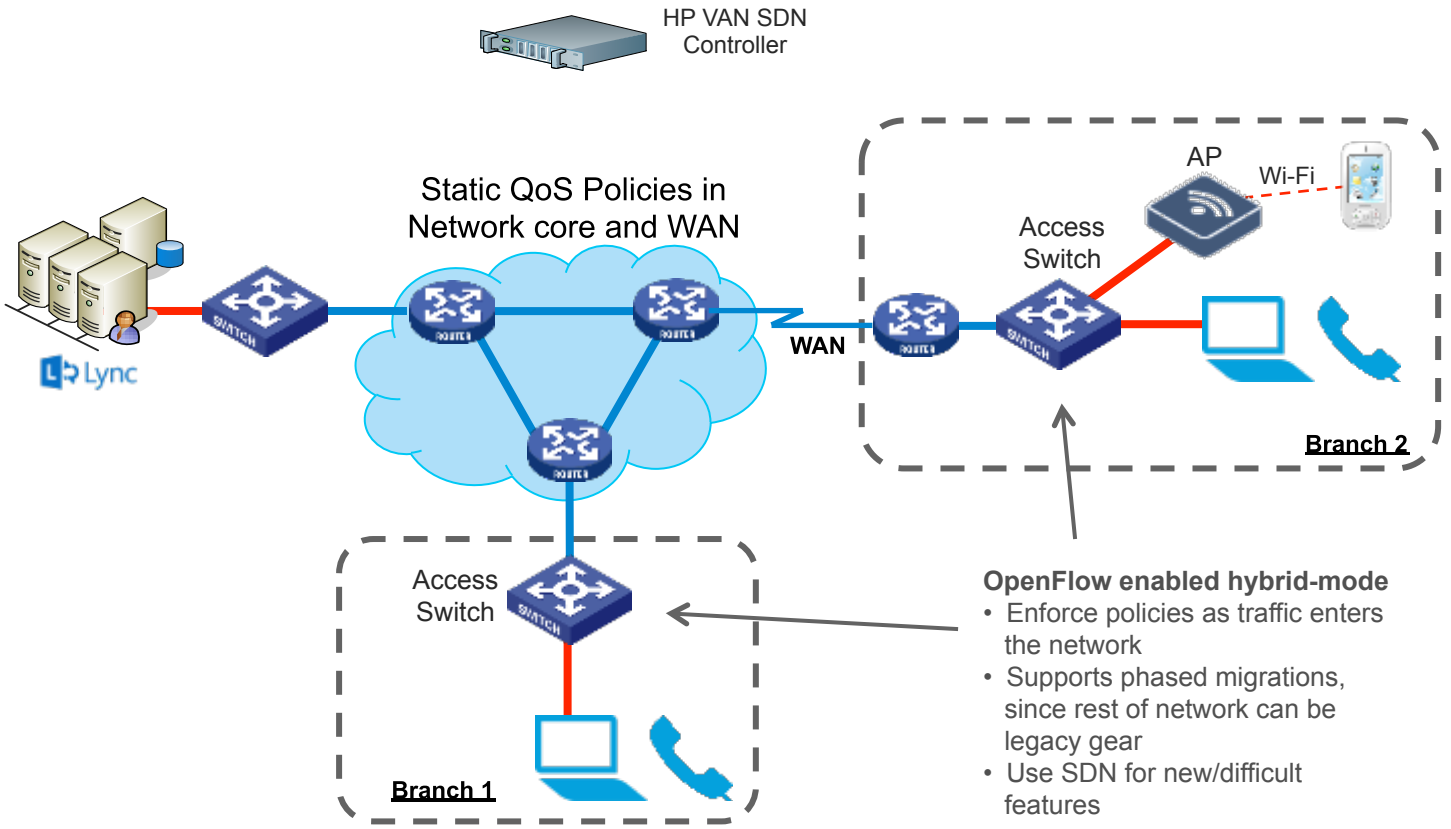
SDN Architecture



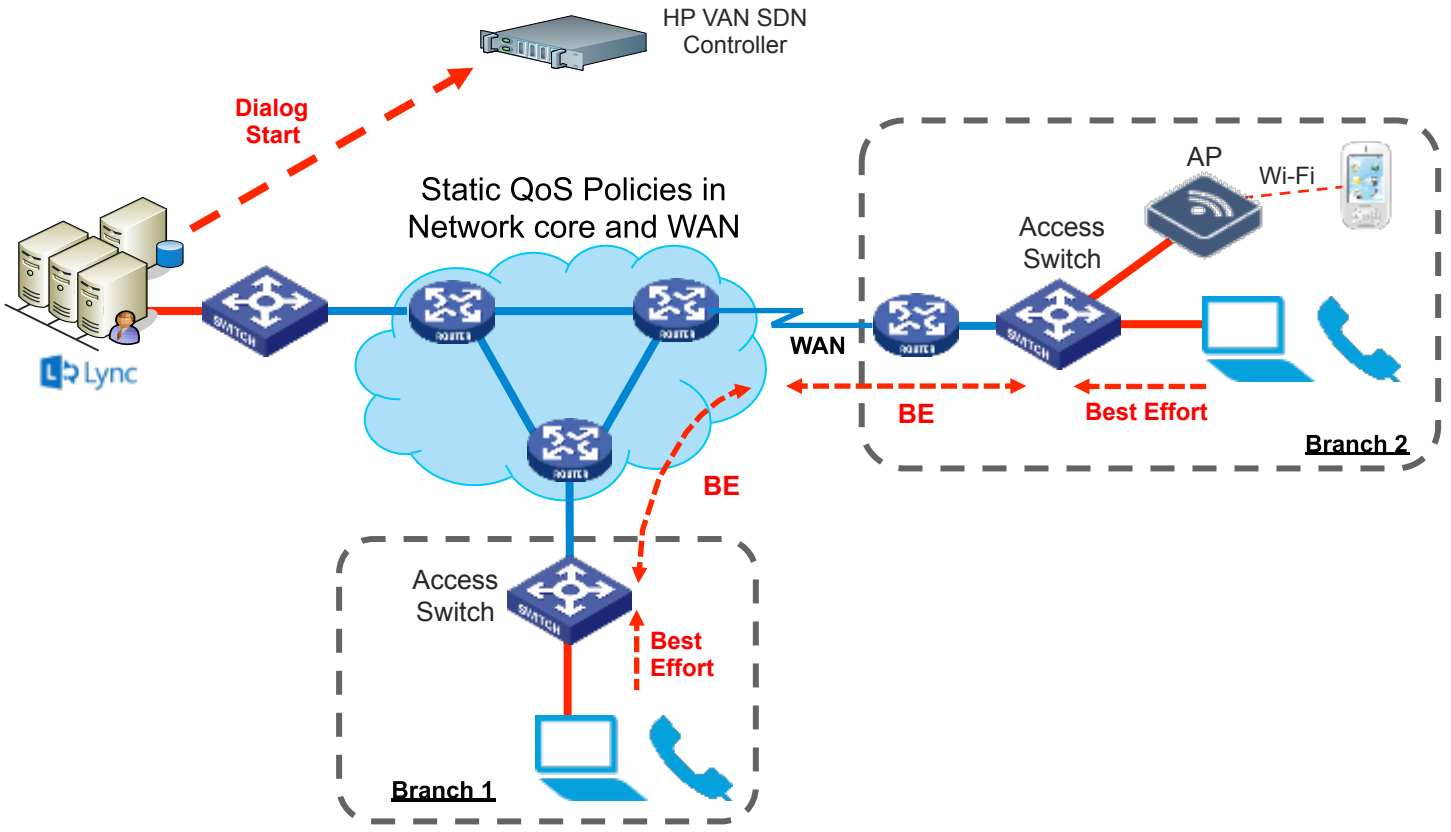
HP Network Optimizer SDN App - Microsoft Lync



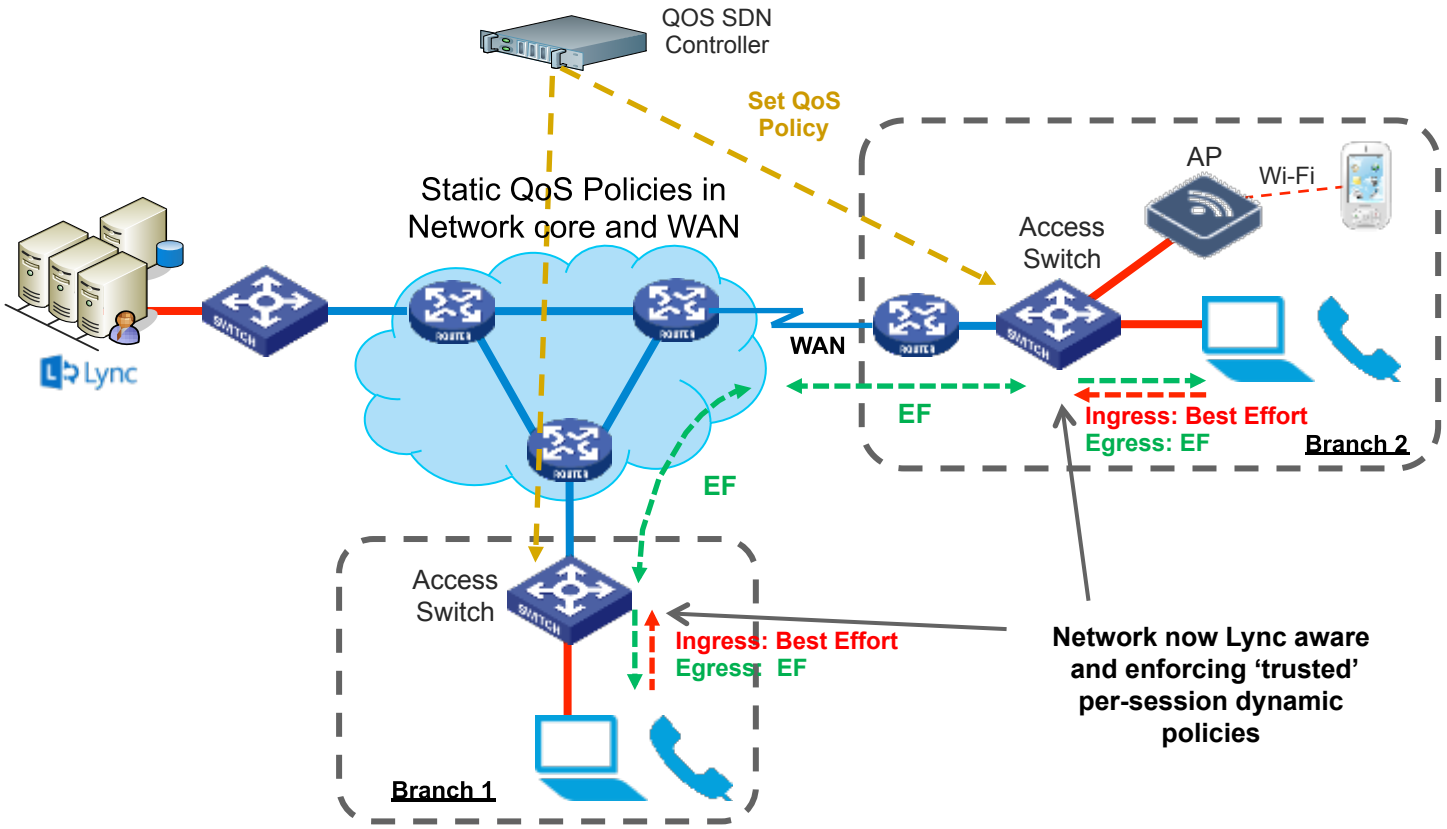
HP's Hybrid SDN Approach



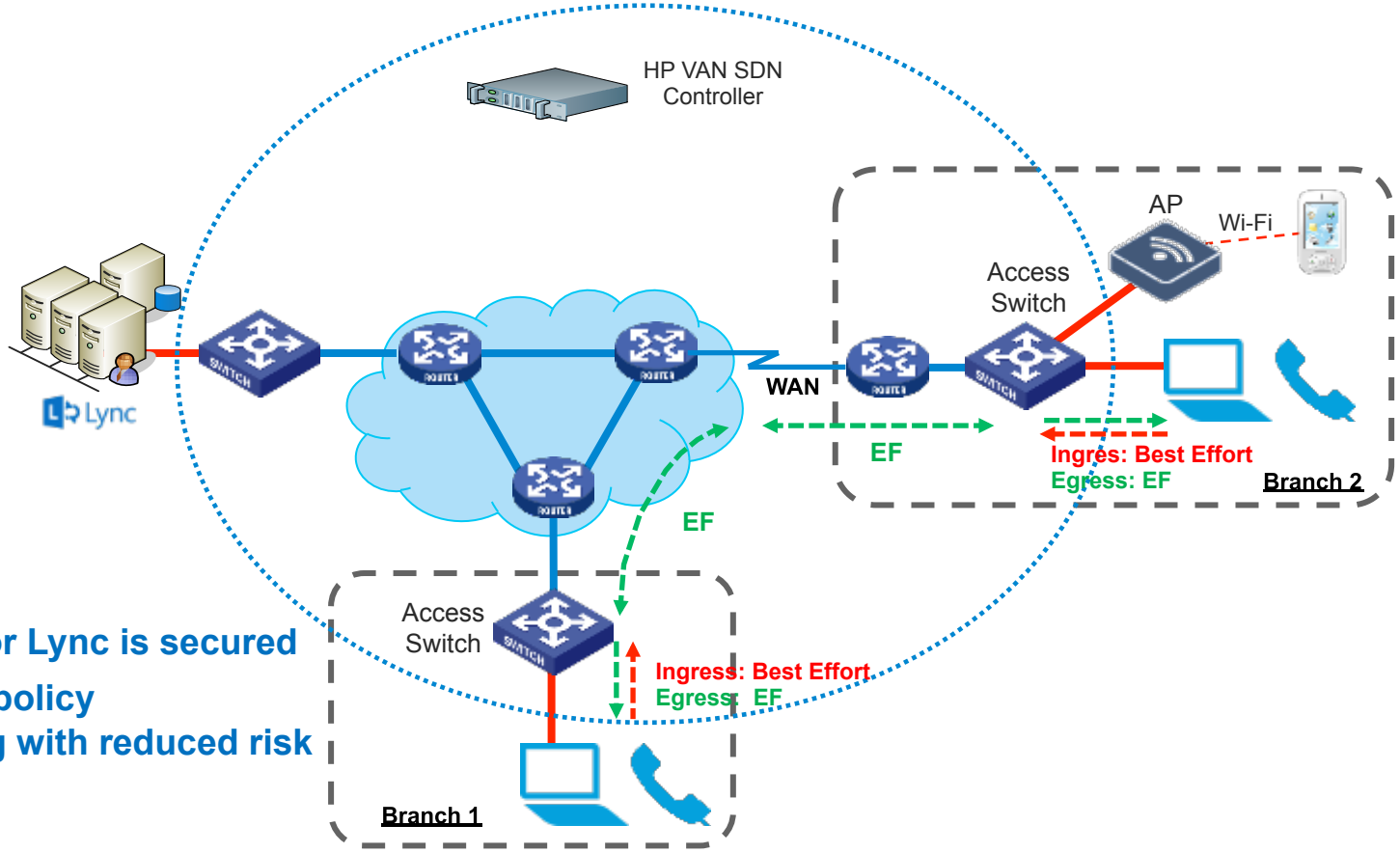
SDN QoS Model for Lync



SDN QoS Model for Lync



SDN QoS Model for Lync



- **Auto QoS for Lync is secured**
- **Automated policy provisioning with reduced risk**

Standards Activity

International Multimedia Telecommunications Consortium (IMTC)

- UC SDN - Use Cases and Data Model specifications
  - ***Dynamic QoS*** : dynamically assign QoS (Network Optimizer v1.2)
  - ***Admission Control*** : prevent voice and video from exceeding available bandwidth capacity
  - ***Dynamic Traffic Engineering*** : route media along path best able to meet performance requirements (dynamic policy based routing)

Open Networking Foundation (ONF)

- Working with IMTC liaison to define standard North-Bound API schema

Thank You to Manfred Arndt and HP

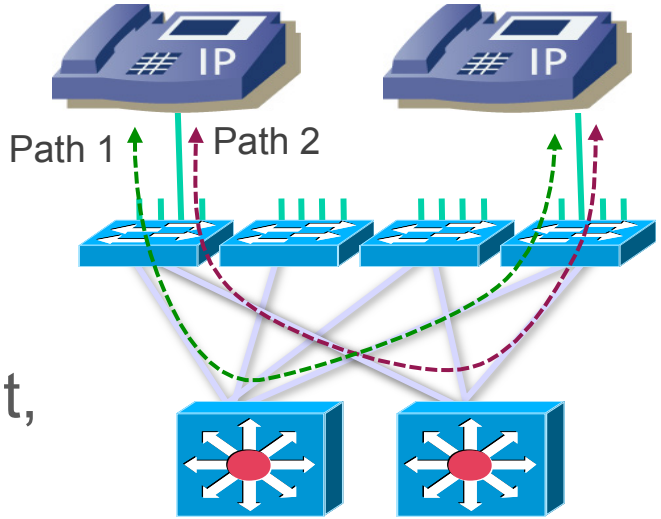
SDN and Call Admission Control

- CAC across multiple UC systems
  - Multi-vendor (Cisco and Avaya due to a merger or acquisition)
  - Multiple media sources (e.g., Lync + Polycom + Skype)
- Policy: handling queue oversubscription
  - Deny the call – communicated back to the UC controller
  - Drop the traffic – but looks like a network failure
  - Mark down – to what class?
  - Have the UC controller make room for the new call

Note: CAC isn't currently supported in UC/SDN systems

SDN and Dynamic Traffic Engineering

- Dynamically select a media traffic path
  - Based on media traffic type
  - Driven by current network loading and characteristics
- Path selection protocols
  - IS-IS or OSPF – shortest path to the destination IP
  - Constraint-based SPF
  - Segment routing – source routing using MPLS tags
  - An area of research & development, ideally suited to SDN



Empowered by Innovation NEC

NEC's *SDN Ready* UC Platform

ProgrammableFlow (PFlow) SDN Controller
Integration with UNIVERGE UC Platforms

Empowering the
Smart Enterprise

SDN Ready Platforms Integration Summary

- NEC's UNIVERGE UC Platforms are now powered by NEC's ProgrammableFlow (**PFlow**) SDN Controller

- NEC's "*SDN Ready*" Platforms dynamically allocate/manage/provision/ secure SDN Network resources

- NEC's "*SDN Ready*" Platforms are tightly integrated with the SDN Controller (NEC's **PFlow**) to ready the data infrastructure for various UC events
  - Some of those events described in next slide

SDN Ready Platforms Areas of Integration

- **Network Provisioning (Deployment/Setup)**
  - Voice, Video Priority
  - End Point Auto Provisioning
  - On-Demand Meeting (Web/Video/Audio Collaboration/Conference)
- **Priority Communications (Crisis)**
  - Precedence calling (Government/DOD)
  - Emergency Call (Public Safety)
  - VIP Call (Hospitality)
  - Emergency Broadcast or Notifications (Education/Public Safety)
  - Nurse Calls (Healthcare)
  - Employee Meeting/Video Conference (Enterprise)

SDN Ready Platforms Areas of Integration

- **Disaster Recovery (Backups & Failures)**
  - Auto Re-allocation of Traffic & Priorities (QoS)
  - Servers/Applications Backup/Synchronization

Network Provisioning

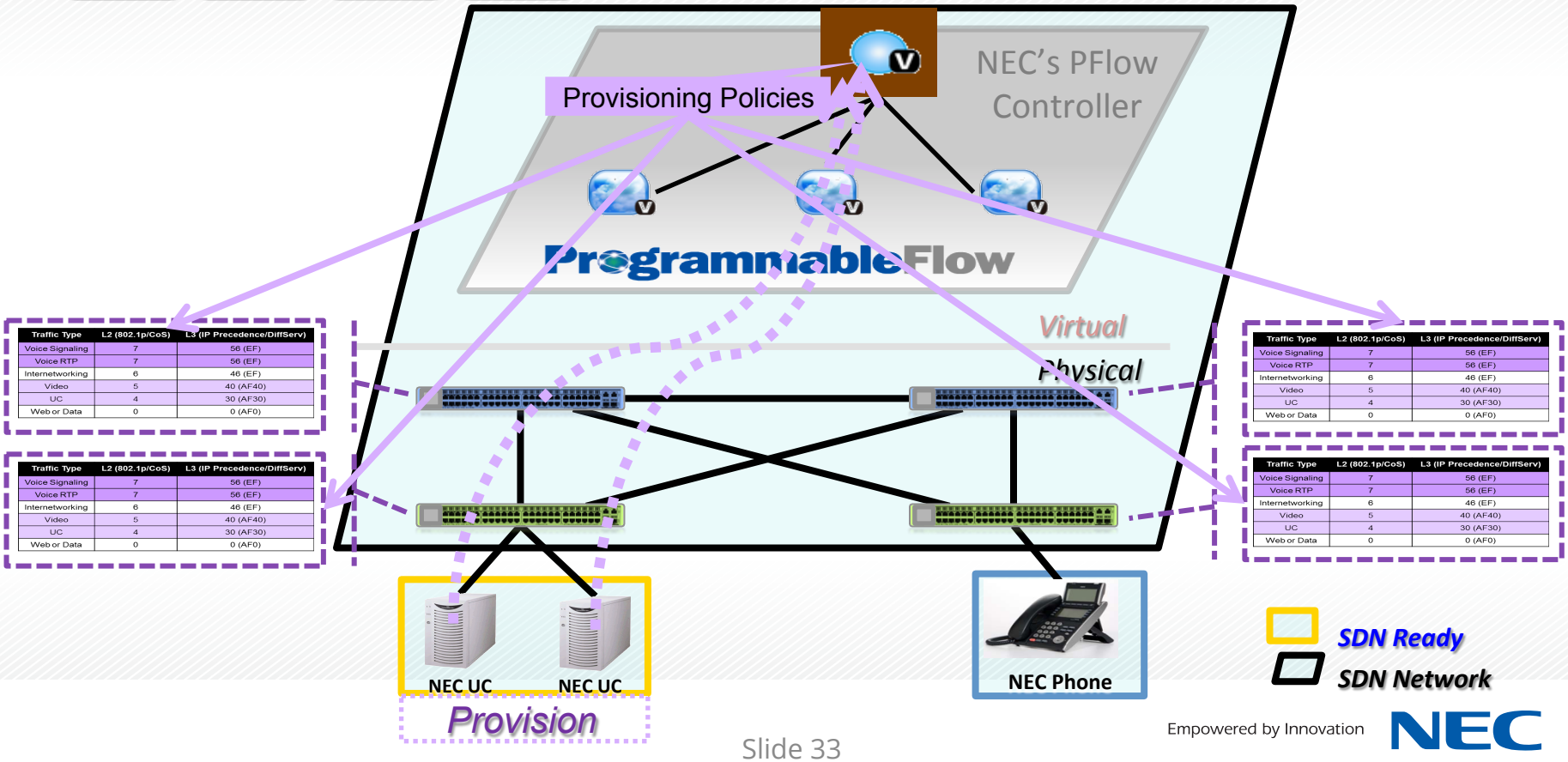
Empowered by Innovation NEC

Provision Network *SDN Ready* Platforms (1/2)

- Provision SDN Network QoS, Bandwidth & Policies
  - UNIVERGE Voice/UC & Collaboration Applications
- Advantages
  - Simplify setup of communication system(s)
  - Allocate proper resources across network for critical applications
  - Centralized Communications Management

Traffic Type	L2 (802.1p/CoS)	L3 (IP Precedence/DiffServ)
Voice Signaling	7	56 (EF)
Voice RTP	7	56 (EF)
Internetworking	6	46 (EF)
Video	5	40 (AF40)
UC	4	30 (AF30)
Web or Data	0	0 (AF0)

Provision Network *SDN Ready* Platforms (2/2)



Priority Communications

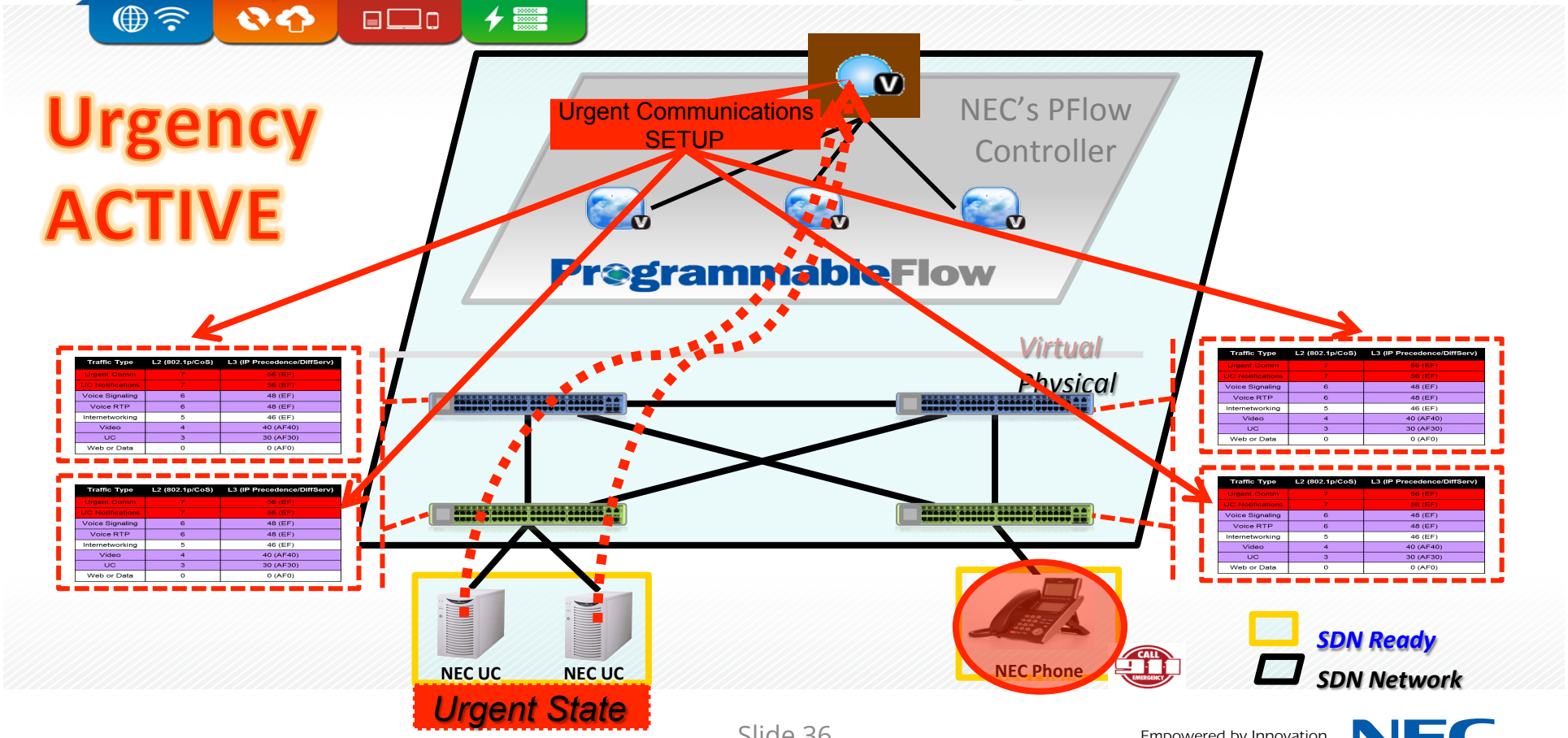
Empowered by Innovation NEC

Priority Communications *SDN Ready* Platforms (1/3)

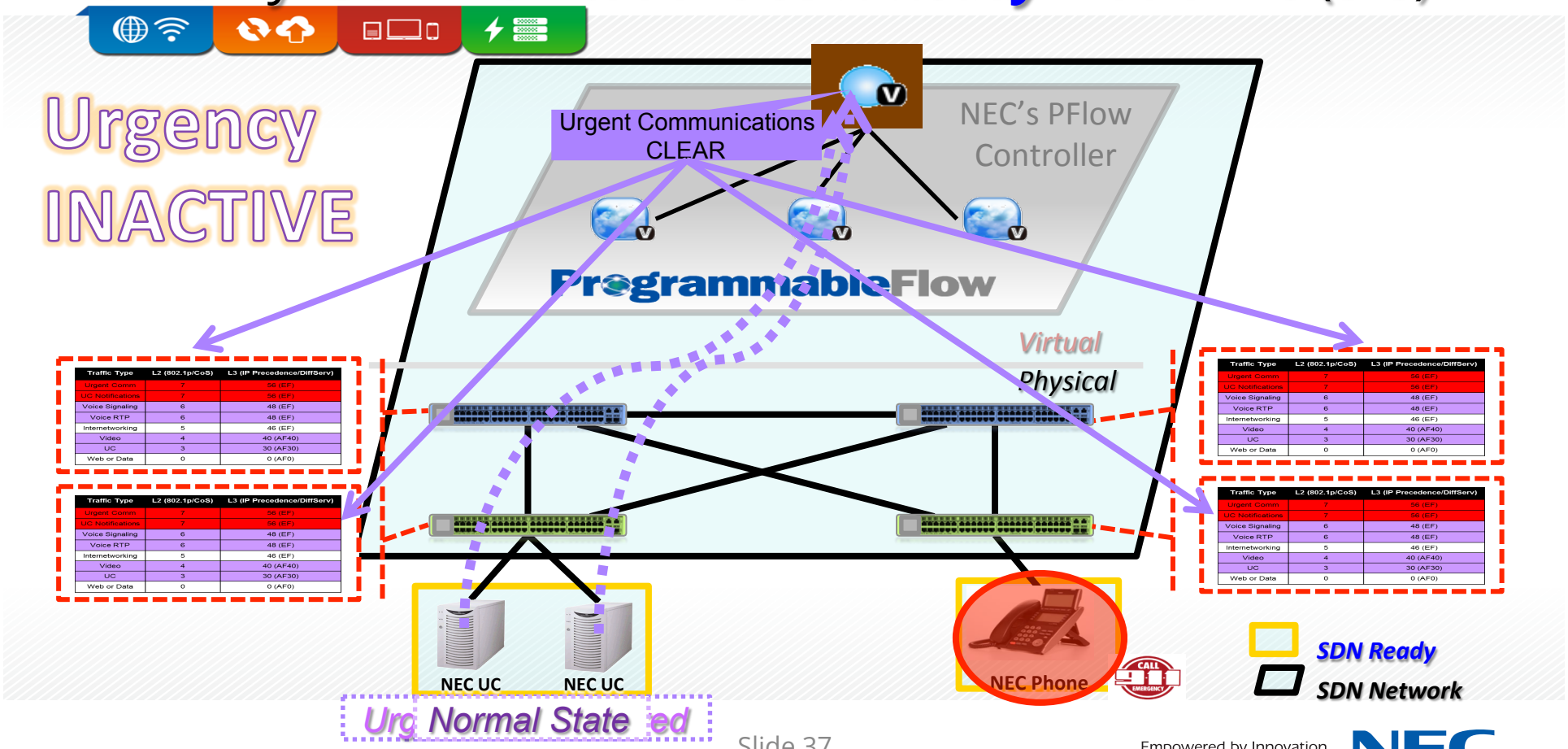
- Prioritize Urgent Communication
  - Precedence Calling (Government/DOD)
  - Emergency Calls (911)
  - Emergency Call (Public Safety)
  - VIP Call (Hospitality)
  - Emergency Broadcasts (Education/ Public Safety)
  - Nurse Calls (Healthcare)
- Advantages
  - Efficient use of SDN Resources
  - Dynamic Allocation of Network Resources for Urgent Communications
  - Ensure end to end prioritization, not just in the communications system
  - Centralized Urgent Communications Control

Traffic Type	L2 (802.1p/CoS)	L3 (IP Precedence/DiffServ)
Urgent Comm	7	56 (EF)
UC Notifications	7	56 (EF)
Voice Signaling	6	48 (EF)
Voice RTP	6	48 (EF)
Internetworking	5	46 (EF)
Video	4	40 (AF40)
UC	3	30 (AF30)
Web or Data	0	0 (AF0)

Priority Communications *SDN Ready* Platforms (2/3)



Priority Communications *SDN Ready* Platforms (3/3)



Disaster Recovery

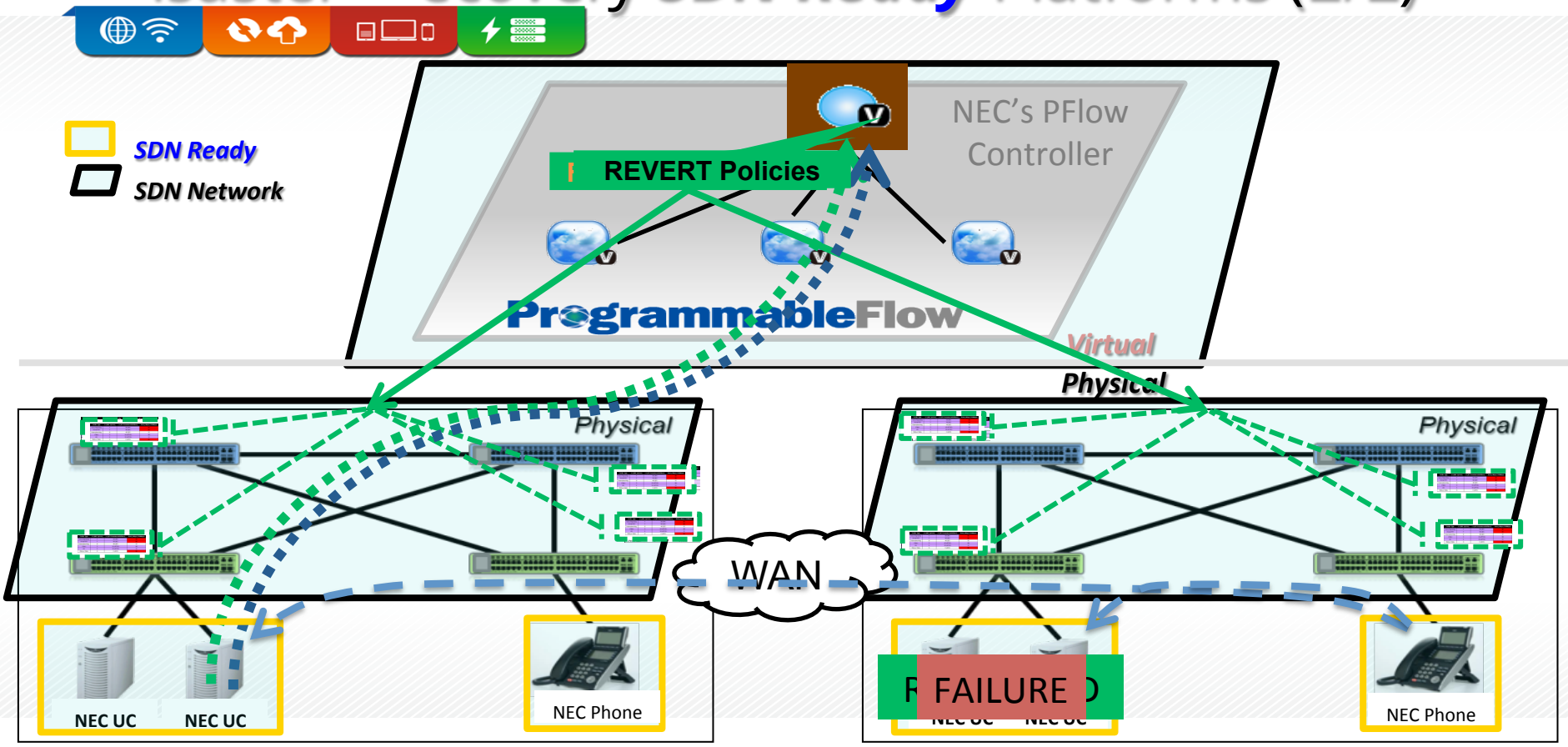
Empowered by Innovation NEC

Disaster Recovery *SDN Ready* Platforms (1/2)

- Provision Failover Scenario Bandwidth Policy
  - UNIVERGE Voice/UC & Collaboration Applications
- Advantages
  - Provide Dynamic Disaster Recovery Communication(s)
  - Re-allocate resources across network in Disaster Recovery State
  - Centralized Disaster Recovery Management

Traffic Type	L2 (802.1p/CoS)	L3 (IP Precedence/DiffServ)	Policy Mbps (100Mbps)
Voice Signaling	7	56 (EF)	**20**
Voice RTP	7	56 (EF)	**40**
Internetworking	6	46 (EF)	**5**
Video	5	40 (AF40)	**10**
UC	4	30 (AF30)	**5**
Web or Data	0	0 (AF0)	**20**

Disaster Recovery *SDN Ready* Platforms (2/2)

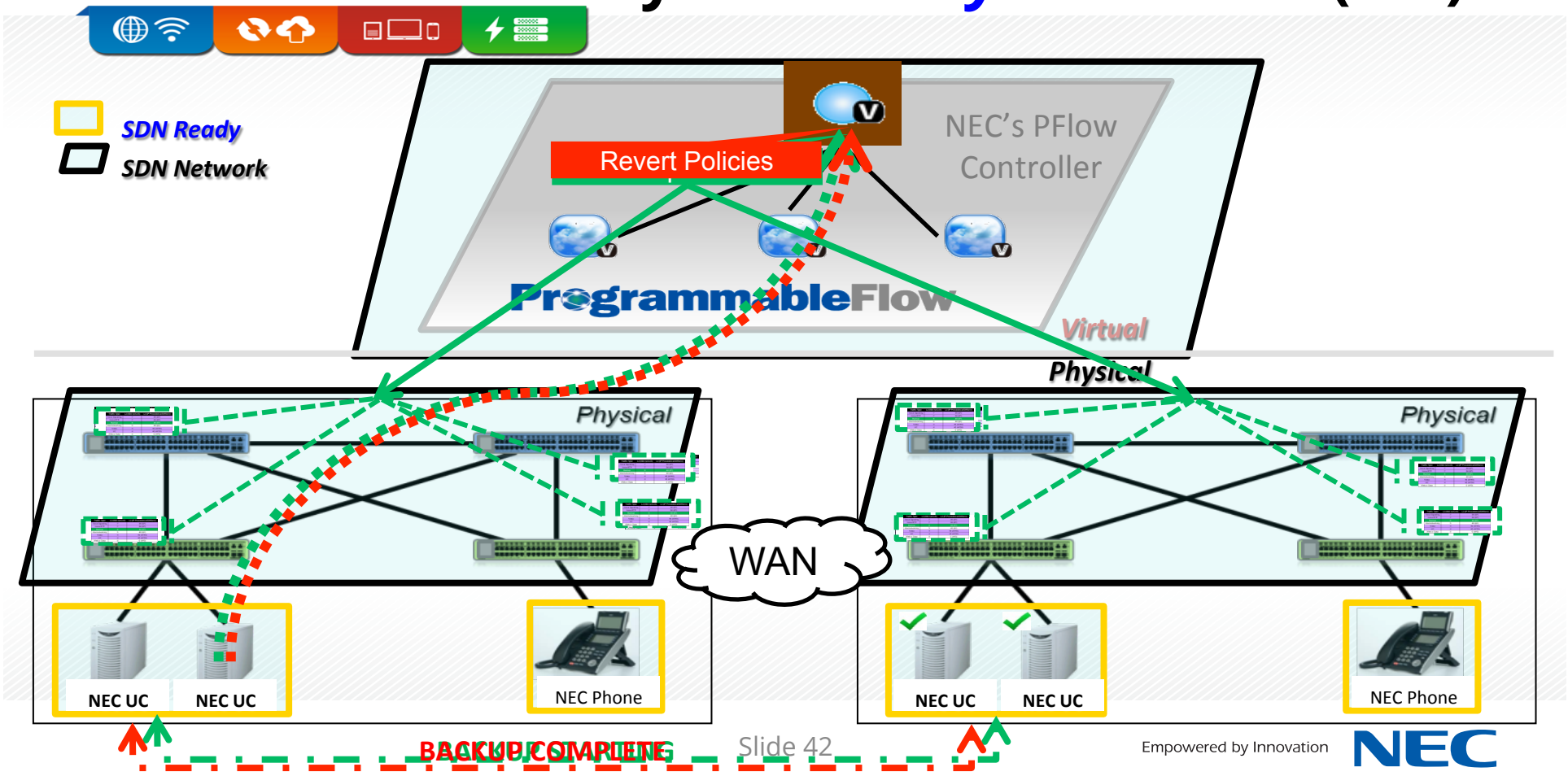


Disaster Recovery *SDN Ready* Platforms (1/2)

- Provision **Recovery** Network QoS, Bandwidth & Policies
  - UNIVERGE Voice/UC & Collaboration Applications
- Advantages
  - Provide Dynamic Disaster Recovery Communication(s)
  - Re-allocate resources across network in Disaster Recovery State
  - Centralized Disaster Recovery Management

Traffic Type	L2 (802.1p/CoS)	L3 (IP Precedence/DiffServ)
Voice Signaling	7	56 (EF)
Voice RTP	7	56 (EF)
Backup	6	48 (EF)
Internetworking	5	46 (EF)
Video	4	40 (AF40)
UC	4	40 (AF30)
Web or Data	0	0 (AF0)

Disaster Recovery *SDN Ready* Platforms (1/2)

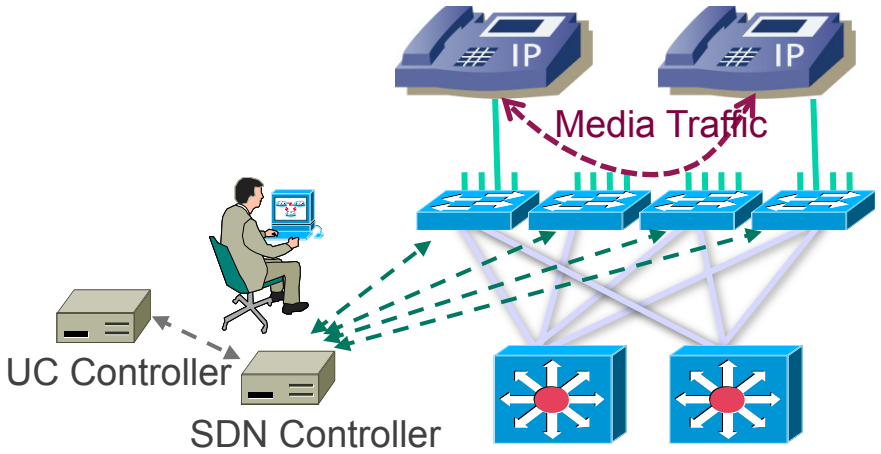


Thank You to NEC!

NEC

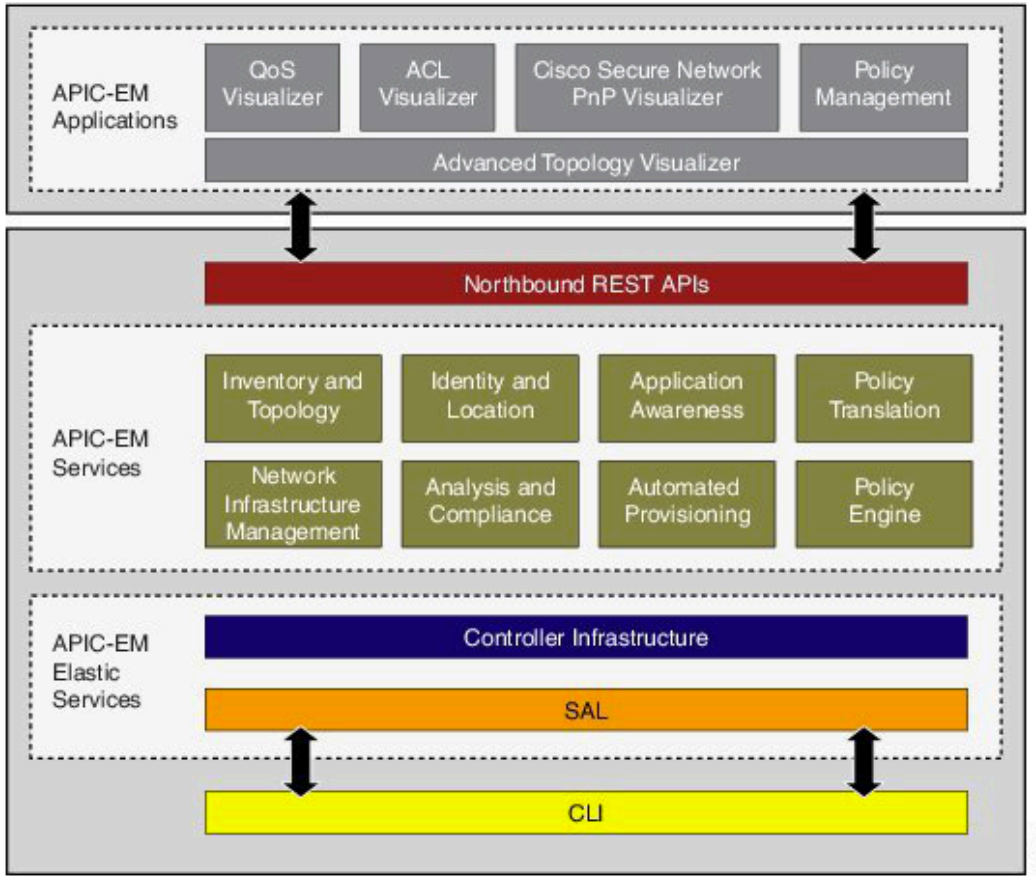
Policies Provide Administrator Control

- Specifying path selection mechanism and criteria
- How to handle oversubscription
  - Tell UC controller to adjust codec on existing calls
  - Deny call or drop packets
  - Mark down media traffic
  - Move traffic to other paths
- Assign application traffic priorities
  - Healthcare: health monitoring apps vs UC

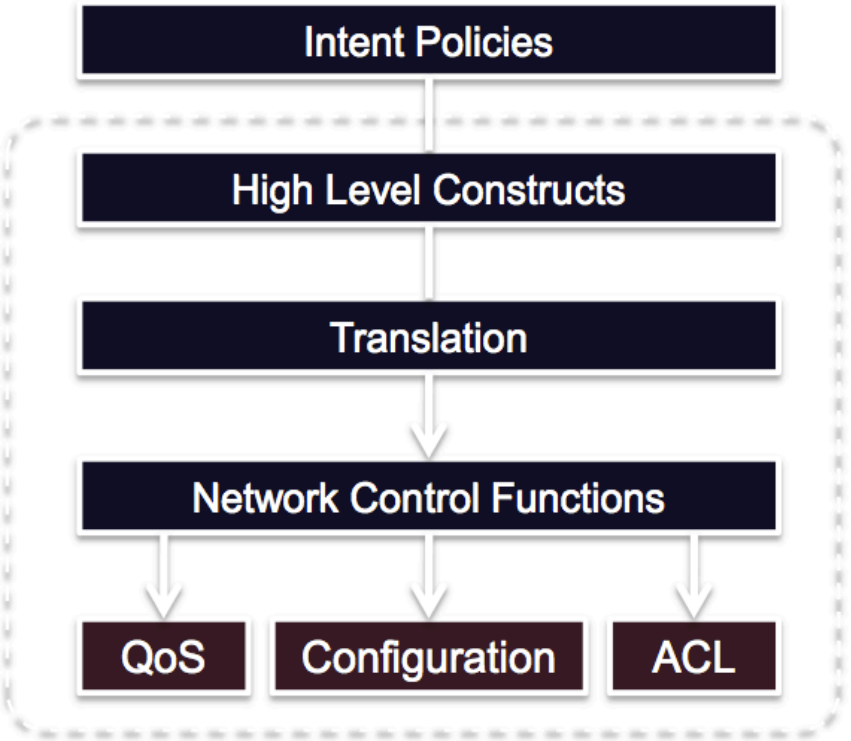


Cisco: APIC Enterprise Module V2 Beta

- Application Policy Infrastructure Controller (APIC)
- Works with traditional network equipment
  - Applications or app interface modules
  - Basic services layer
  - Device control via Service Abstraction Layer (SAL)



Intent Policy Management Service



Translation of high level constructs to network control functions reduces skills gaps and clarifies policy procedures

Cisco live!

Agenda

- Traditional Networking
- SDN and UC
- REST API
- Future Directions and Summary

Representational State Transfer (REST) API

- Simple stateless data transfer mechanism
  - Generally operates over HTTP
  - XML or JavaScript Object Notation (JSON) encoding
- Four functions (CRUD):

Create	Post
Read	Get
Update	Put
Delete	Delete

JSON Format:

```
{
    "id": 1,
    "name": "Foo",
    "price": 123,
    "tags": [ "Bar", "Eek" ],
    "stock": {
        "warehouse": 300,
        "retail": 20
    }
}
```

Header: Content-Type: Application/JSON

Simple Policy

http://10.10.10.10:8081/api/v0/policy POST

```
{
 "actions": ["DENY"],
  "policyOwner":"admin",
  "policyName": "deny_all",
  "networkUser": {"userIdentifiers": ["10.10.20.7"]}, # src IP
  "resource": {"applications": ["0,0,TCP"]} # dst ports (optional)
}
```

userIdentifier can be an IP address || user-id ||group-id

Response

```
{
    "version": "0.0",
    "response": "16cbd3f9-cb02-49cb-bbcd-c661dfc75d5e"
}
```


Cisco live!

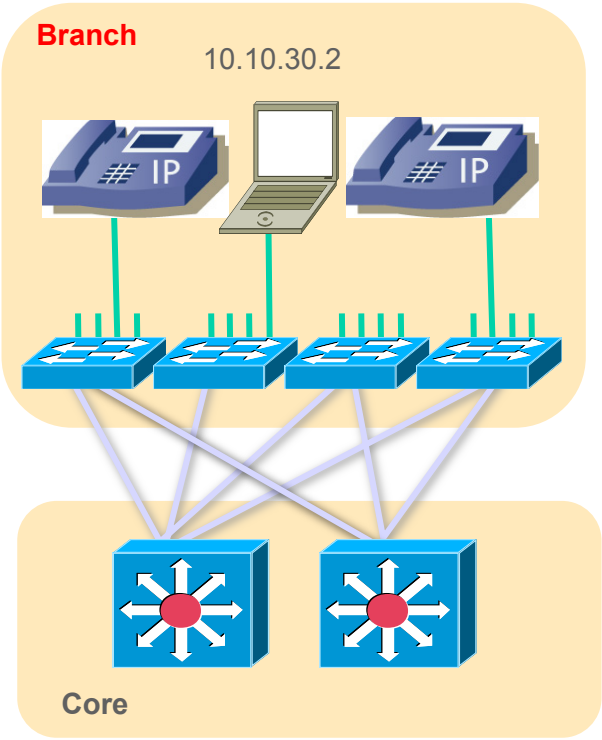
Policy created on controller

http://10.10.10.10:8081/api/v0/policy/16cbd3f9-cb02-49cb-bbcd-c661dfc75d5e GET


```
{
    "version": "0.0",
    "response": {"policyPriority": 64,
                 "actions": ["DENY”],
                 "policyId": "16cbd3f9-cb02-49cb-bbcd-c661dfc75d5e",
                 "policyName": "deny_all",
                 "policyOwner": "admin",
                 "networkUser": {"userIdentifiers": ["10.10.20.7”]},
                 "resource": {"applications": ["0,0,TCP”]},
                 "state": "Active”
                 }
}
```


Device Tagging

- Group devices by function or role
  - Automation aid
  - Group configuration
  - Defines the scope of policies
- Example
  - Configure Core for queuing and forwarding
  - Configure Branch for classification, marking, and queuing



Scope of policy

http://10.10.10.10:8081/api/v0/policy POST

```
{
  "actions": ["DENY"],
  "policyName": "denyweb",
  "policyOwner": "Admin",
  "networkUser": {"userIdentifiers": ["10.10.30.2"],
                  "applications": ["80,80,TCP"]},
  "scope" : "branch"
}
```

Only applied to devices with tag “branch”

Marking Traffic

http://10.10.10.10:8081/api/v0/policy POST

```
{
  "actions": ["PERMIT"],
  "policyName": "src-marking",
  "policyOwner": "Admin",
  "actionProperty": {"priorityLevel": "46"}, #DSCP bits
  "networkUser": {"userIdentifiers": ["10.10.20.5"]} #src IP
}
```


Cisco live!

API Maturity

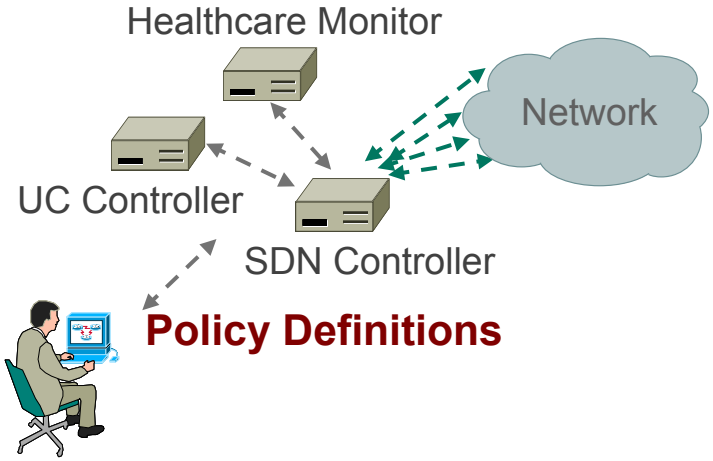
- Most APIs are low-level
  - Some emulate CLI-level functionality
  - Higher-level abstractions are being developed
    - Scope (tagging) and End-Point-Groups
    - Abstract QoS definitions (hides details of classification/marking)
  - Documentation is often vague
- North-Bound Interfaces are maturing
  - Middleware is a valuable component (e.g., QoE Services)
  - Use-cases are helping define required functionality

Agenda

- Traditional Networking
- SDN and UC
- REST API
- Future Directions and Summary

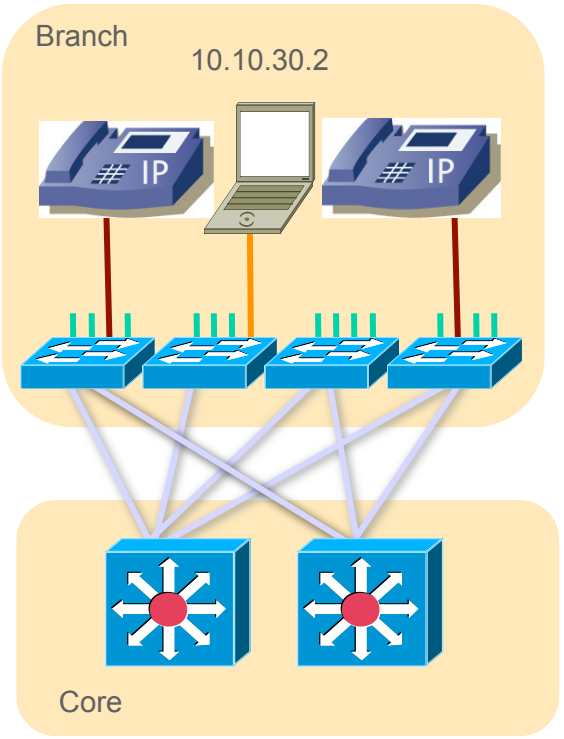
Future Directions

- Network must communicate back to the application
  - Communicate bandwidth changes (failure or new capacity)
  - Handling oversubscription
  - Monitoring and diagnostics
- Smart policy engines
  - Help resolve conflicts between applications
  - Simplify policy creation mechanisms



Summary

- Need high level abstractions
  - Simplify how we think about networks
  - Device groups (Cisco: device tags)
  - Interface groups
    (Cisco: End Point Groups – EPG)
  - Device-independent QoS definitions
  - L3 forwarding domains (multi-tenancy)
- Prediction
  - Today: One application per VM
  - Future: One application per L3 network domain
    - Simplify service chaining



Summary

- SDN is happening!
  - Progress has been fast
  - Production rollout is limited by equipment refresh
- Plan for 2 to 4 times the voice-data convergence effort
  - Cultural changes must occur
  - Start with a small pilot program
  - Cross-discipline team (network, security, IT)
  - SDN resources at http://www.netcraftsmen.com/resources/sdn-resources/

Questions?

SDN APIs for Communications

When Applications and the
Network Talk with Each Other

Terry Slattery
Principal Consultant
NetCraftsmen
CCIE #1026