Διαχείριση δικτύων

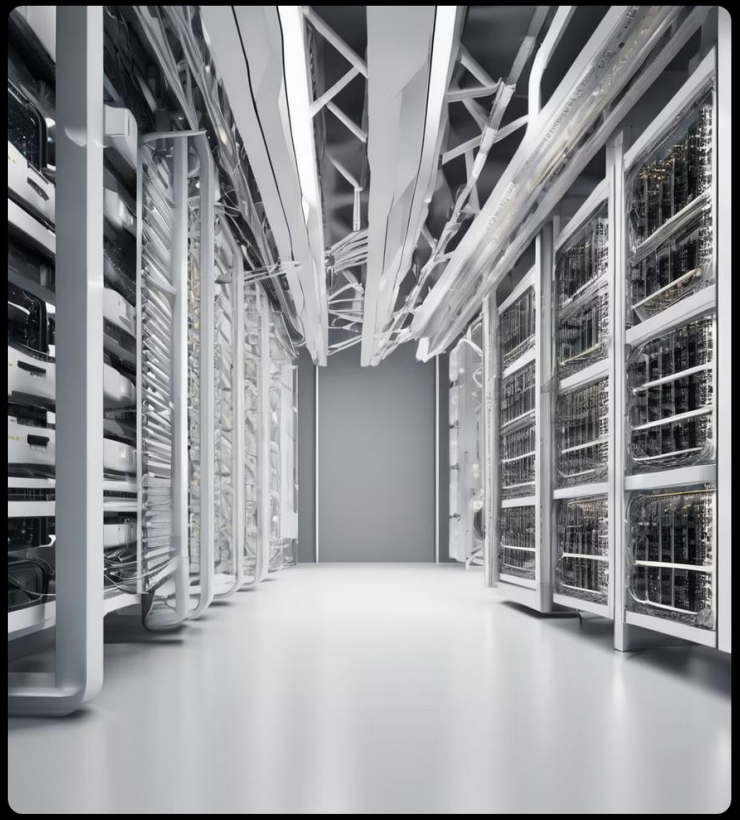
Καθ. Νάνσυ Αλωνιστιώτη

nancy@di.uoa.gr

PROJECT-BASED

- ΕΙΣΑΓΩΓΗ
- ΠΕΡΙΓΡΑΦΗ **PROJECT**
- **SUPPORT LECTURES** -ΤΡΙΤΕΣ
- **PROJECT-TEAMS Q&A** – ΠΑΡΑΣΚΕΥΕΣ (ΚΑΤΟΠΙΝ ΗΛΕΚΤΡΟΝΙΚΗΣ ΥΠΟΒΟΛΗΣ ΕΡΩΤΗΣΕΩΝ)

Network Management: Challenges and Innovations



Introduction

- Overview of Network Management
- Evolving Landscape: Software Defined Networking (SDN), AI, Digital Twins

Introduction

- As technology advances, so does the complexity of managing modern networks.
- Traditional approaches face challenges in scalability, adaptability, and efficiency.

Traditional Network Management

- SNMP-based Management
- Challenges in Legacy Approaches

Overview of Network Management

Network management is crucial for maintaining the performance, reliability, and security of computer networks. It involves monitoring, configuring, and optimizing network resources to ensure efficient and uninterrupted communication.

Key Concepts

- Network Monitoring: Continuously monitoring network devices, traffic, and performance to identify issues and ensure optimal network operation.
- Configuration Management: Managing network device configurations, including updates, backups, and compliance with security policies.
- Performance Optimization: Analyzing network performance data to identify bottlenecks and optimize network resources.
- Security Management: Implementing measures to protect the network from unauthorized access, threats, and vulnerabilities.
- Fault Management: Detecting, isolating, and resolving network faults to minimize downtime and ensure network availability.
- Network Documentation: Maintaining accurate documentation of network infrastructure, configurations, and changes.

Traditional Network Management

- **SNMP-based Management**
- In the early days of network management, the Simple Network Management Protocol (SNMP) emerged as a standard for monitoring and managing network devices.
- SNMP allowed administrators to collect and organize information from various network devices, facilitating tasks such as fault detection, performance monitoring, and configuration management.

Traditional Network Management

- **Challenges in Legacy Approaches**
- While SNMP and similar protocols have served as the foundation for network management, they come with inherent challenges in the face of evolving network architectures and requirements:

1. **Limited Programmability:**
    1. Traditional methods lack the programmability needed to adapt quickly to changing network conditions and demands.

Traditional Network Management

- Challenges of Traditional Network Management
- Traditional network management relies on manual configurations and static policies, leading to inefficiencies and errors.
- Lack of centralized control and visibility makes it difficult to adapt to dynamic network conditions and evolving business needs.
- Scaling traditional network management becomes increasingly complex and costly, hindering agility and innovation.

Traditional Network Management

1. **Inefficient Resource Allocation:**
    1. Traditional approaches may lead to suboptimal resource allocation, hindering the network's ability to meet the evolving needs of applications and services.

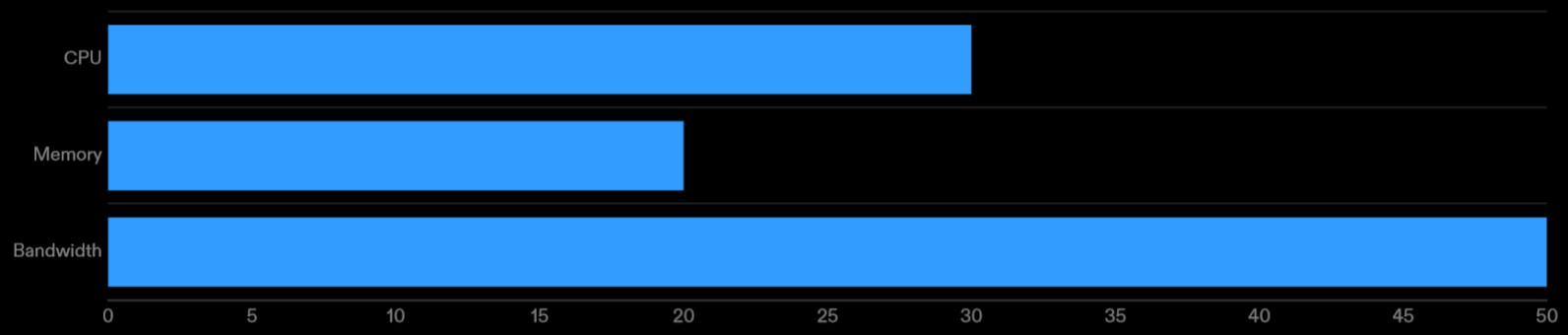
- In response to these challenges, the industry has embraced innovative approaches, such as Software Defined Networking (SDN), to overcome the limitations of traditional network management. SDN, AI, and digital twins are reshaping the way we manage and optimize networks for the future.

Inefficient Resource Allocation

Traditional Network Management

- In traditional network management, resources are allocated based on static configurations and predefined policies.
- This approach often leads to inefficient resource allocation, as the allocation does not dynamically adapt to the changing needs of applications and services.

Resource Allocation



Impact on Applications and Services

- Inefficient resource allocation can lead to performance bottlenecks and suboptimal utilization of network resources.
- This can result in degraded application performance, increased latency, and poor user experience.

Traditional Network Management

1. **Scalability Issues:**
    1. As networks grow in complexity and scale, traditional approaches struggle to efficiently manage an increasing number of devices and dynamic configurations.
2. **Manual Configuration:**
    1. Manual configuration and management are time-consuming, error-prone, and do not align with the agility required in today's rapidly changing technological landscape.

Challenges of Traditional Network Management

Manual Configurations

- Traditional network management requires manual configurations for each device and network component.
- This process is time-consuming and prone to human errors, leading to inefficiencies and network downtime.

Lack of Centralized Control

- Traditional network management lacks a centralized control system, making it difficult to monitor and manage the entire network.
- Network administrators have to access and configure each device individually, leading to increased complexity and inefficiency.

Scalability Issues

- Traditional network management struggles to scale with the increasing size and complexity of modern networks.
- Adding new devices or expanding the network requires significant manual effort and can lead to performance issues.

Intent-Based Networking (IBN)

- IBN translates high-level business policies into network configurations automatically.
- By aligning network behavior with business objectives, IBN enhances agility, security, and reliability.
- IBN leverages artificial intelligence and machine learning to continuously optimize network performance and ensure compliance with policy requirements.

Introduction to Software Defined Networking (SDN)

- Definition and Key Concepts
- Separation of Control Plane and Data Plane

Introduction to Software Defined Networking (SDN)

- **Software Defined Networking (SDN)** represents a revolutionary shift in network architecture, offering unprecedented flexibility and control. At its core, SDN separates the traditional network control plane from the data plane, introducing a centralized intelligence that orchestrates network resources dynamically.

Introduction

1. **Software Defined Networking (SDN):**

   An innovative paradigm that separates the control plane from the data plane, providing programmability, centralized control, and unprecedented flexibility in managing network resources.

Introduction

1. **Artificial Intelligence (AI) for Network Management:**
    1. Harnessing the power of AI, we can achieve intelligent, adaptive network management. Machine learning algorithms enable predictive analysis, anomaly detection, and dynamic resource allocation, optimizing network performance.

Introduction

1. **Digital Twins for Network Management:**

   A concept borrowed from industrial IoT, digital twins create virtual replicas of network components. This technology enhances real-time monitoring, predictive analysis, and optimization, providing a holistic view of the network's health and performance.

Software-Defined Networking (SDN)

- SDN decouples the control plane from the data plane, allowing for centralized control and programmability of network devices.
- Advantages include dynamic traffic management, simplified network configuration, and improved scalability.
- SDN facilitates network automation and orchestration, enhancing efficiency and flexibility in network management.

Network Function Virtualization (NFV)

- NFV replaces traditional network appliances with software-based virtualized instances running on standard hardware.
- This enables greater agility, scalability, and cost-effectiveness in network deployment and management.
- NFV allows for on-demand service chaining and rapid deployment of network functions, improving resource utilization and reducing operational complexities.

Introduction to Software Defined Networking (SDN)

- **SDN Controller: The Brain of the Operation**
  - At the heart of Software Defined Networking (SDN) is the SDN Controller, a centralized intelligence that transforms network management. The brain orchestrating the entire network.
  - It communicates with network devices, translating high-level network policies into low-level instructions for individual devices.

- **Dynamic Adaptation:**
  1. Enables dynamic adaptation to network changes, traffic patterns, and application demands by issuing real-time commands to network devices.

Introduction to Software Defined Networking (SDN)

1. **Southbound and Northbound APIs:**
    1. *Southbound APIs* connect the SDN controller to the data plane, allowing it to communicate with switches and routers. On the other hand, *northbound APIs* facilitate communication between the SDN controller and applications, enabling the implementation of network policies.

- By decoupling the control plane from the underlying infrastructure, SDN empowers network administrators to programmatically control network behavior and respond dynamically to changing requirements.

Components of SDN

1. **Southbound APIs - OPENFLOW:**
    1. Connect the SDN Controller to the data plane, allowing it to communicate with switches and routers. These APIs enable the translation of high-level policies into actionable instructions.
2. **Northbound APIs – e.g., JSON:**
    1. Facilitate communication between the SDN Controller and applications or higher-layer software. These APIs empower external applications to interact with and control the network, implementing policies and utilizing network resources.

Advantages of SDN

- Network Programmability
- Centralized Management and Control

AI in Network Management

- Role of Artificial Intelligence
- Machine Learning for Anomaly Detection

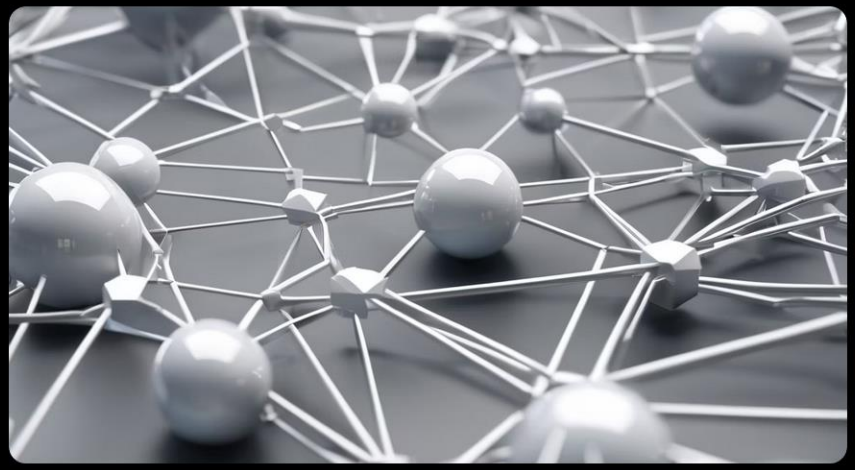
Use Cases of AI in Network Management

- Predictive Maintenance
- Dynamic Resource Allocation
- Energy Efficiency
- Security
- Predictive resource allocation
- Crisis management

Traffic Engineering in SDN

- SDN for Efficient Traffic Engineering
- Benefits of Dynamic Traffic Management
- Cloud RAN
- Slicing
- Multitenancy

Integration of SDN, AI, and Digital Twins



Digital Twins

Digital twins are virtual representations of physical network infrastructure. They enable real-time monitoring, simulation, and analysis of network behavior. By creating a digital replica, network managers can gain insights, optimize performance, and identify potential issues before they occur.

Integration of SDN, AI, and Digital Twins

- How SDN, AI, and Digital Twins Work Together
- Synergy in Network Management
- Definition and Concept
- Creating Digital Twins of Network Components
- Predictive Analysis
- Real-time Monitoring and Optimization

Evolving Landscape: Software Defined Networking (SDN), AI, Digital Twins

Software Defined Networking (SDN)

SDN is a network management approach that separates the control plane from the data plane, enabling centralized control and programmability of network infrastructure. It offers flexibility, scalability, and automation, revolutionizing network management.

Artificial Intelligence (AI)

AI is transforming network management by leveraging machine learning algorithms to analyze network data, detect anomalies, and optimize performance. It enables proactive network monitoring, predictive maintenance, and intelligent decision-making.

Innovative Approaches: Software Defined Networking (SDN), AI, Digital Twins

Software Defined Networking (SDN)

Software Defined Networking (SDN) is an innovative approach to network management that separates the control plane from the data plane. It allows for centralized control and programmability of network infrastructure, providing greater flexibility, scalability, and automation.

Artificial Intelligence (AI)

Artificial Intelligence (AI) is revolutionizing network management by enabling intelligent automation, predictive analytics, and proactive troubleshooting. AI algorithms can analyze vast amounts of network data in real-time, identify patterns, and make intelligent decisions to optimize network performance and security.

Digital Twins

Digital twins are virtual replicas of physical network infrastructure, allowing for real-time monitoring, simulation, and analysis. By creating a digital twin of a network, organizations can gain deep insights into network behavior, identify potential issues, and test network changes before implementation.

Future Trends

- Emerging Technologies and Trends in Network Management
- 6G and Beyond
- Joint Sensing and Communications
- TN/NTN integration

Case Studies

- Emerging Technologies and Trends in Network Management
- 6G and Beyond
- **Challenges and Considerations**
- Security Concerns
- Skillset Requirements

```python
from pysnmp.hlapi import *

# SNMPv2c parameters
snmp_community = 'public'
snmp_host = 'router_ip_address'

# OID for system description
oid_sys_descr = ObjectIdentity('SNMPv2-MIB', 'sysDescr', 0)

# SNMP GET operation
errorIndication, errorStatus, errorIndex, varBinds = next(
    getCmd(SnmpEngine(),
           CommunityData(snmp_community),
           UdpTransportTarget((snmp_host, 161)),
           ContextData(),
           ObjectType(oid_sys_descr))
)
# Check for errors
if errorIndication:
    print('SNMP GET operation failed: %s' % errorIndication)
elif errorStatus:
    print('SNMP GET operation failed: %s at %s' % (
        errorStatus.prettyPrint(),
        errorIndex and varBinds[int(errorIndex) - 1][0] or '?'
    ))
else:
    # Print system description
    for varBind in varBinds:
        print('System Description:', varBind[1].prettyPrint())

# OID for system description (to set)
oid_sys_descr_set = ObjectIdentity('SNMPv2-MIB', 'sysDescr', 0)

# New system description
new_sys_descr = 'New system description'

# SNMP SET operation
errorIndication, errorStatus, errorIndex, varBinds = next(
    setCmd(SnmpEngine(),
           CommunityData(snmp_community),
           UdpTransportTarget((snmp_host, 161)),
           ContextData(),
           ObjectType(oid_sys_descr_set, OctetString(new_sys_descr)))
)

# Check for errors
if errorIndication:
    print('SNMP SET operation failed: %s' % errorIndication)
elif errorStatus:
    print('SNMP SET operation failed: %s at %s' % (
        errorStatus.prettyPrint(),
        errorIndex and varBinds[int(errorIndex) - 1][0] or '?'
    ))
else:
    print('System Description set successfully to:', new_sys_descr)
```

SNMP to configure a router using Python. Retrieve and set the router's system description using SNMP

```python
from easysnmp import Session

# SNMP parameters
snmp_host = '127.0.0.1'
snmp_port = 1161
snmp_community = 'public'

# OID for system description
oid_sys_descr = '1.3.6.1.2.1.1.1.0'

# Create SNMP session
session = Session(hostname=snmp_host, community=snmp_community, version=2, port=snmp_

# Perform SNMP GET operation
sys_descr = session.get(oid_sys_descr)

# Print system description
print('System Description:', sys_descr.value)

# Perform SNMP SET operation (example)
# OID for system description (to set)
oid_sys_descr_set = '1.3.6.1.2.1.1.1.0'

# New system description
new_sys_descr = 'New system description'

# Perform SNMP SET operation
session.set(oid_sys_descr_set, new_sys_descr)

# Verify the change by performing another SNMP GET operation
sys_descr_new = session.get(oid_sys_descr_set)

# Print updated system description
print('Updated System Description:', sys_descr_new.value)
```

SNMP-based configuration actions on a simulated network using Mininet.

Configure SNMP on the simulated network devices and then interact with them using SNMP commands

```python
from mininet.net import Mininet
from mininet.topo import SingleSwitchTopo
from mininet.node import RemoteController, OVSSwitch
from mininet.cli import CLI

# Create network topology
topo = SingleSwitchTopo(2)
net = Mininet(topo=topo, switch=OVSSwitch, controller=RemoteController)

# Start network
net.start()

# Add flow entries to switch
switch = net.switches[0]
switch.dpctl('add-flow', 'in_port=1,actions=output:2')  # Forward traffic from port 1
switch.dpctl('add-flow', 'in_port=2,actions=output:1')  # Forward traffic from port 2


# Verify flow entries
print("Flow entries added:")
switch.dpctl('dump-flows')

# Start Mininet CLI
CLI(net)

# Stop network
net.stop()
```

OpenFlow-based configuration actions on a simulated network using Mininet.

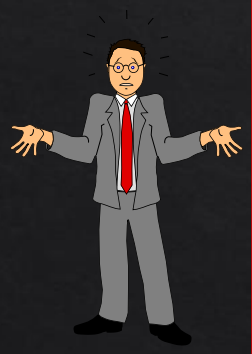
Mininet Python API to interact with OpenFlow-enabled switches

Network manager - Διαχειριστής Δικτύου: super-hero or super-engineer?

Απο την απόγνωση στην επίγνωση

The notion of network management….

- "autonomous" systems (aka "network"): 100s or 1000s of interacting hardware/software components
- other complex systems requiring monitoring, control:
  - jet airplane
  - nuclear power plant
  - others?



"Network management includes the deployment, integration and coordination of the hardware, software, and human elements to monitor, test, poll, configure, analyze, evaluate, and control the network and element resources to meet the real-time, operational performance, and Quality of Service requirements at a reasonable cost."

Requirements in new generation networks -Οι απαιτήσεις στα σύγχρονα δίκτυα

Support up to 1000 times more traffic

Enable Gbps peak speeds

Improve energy efficiency

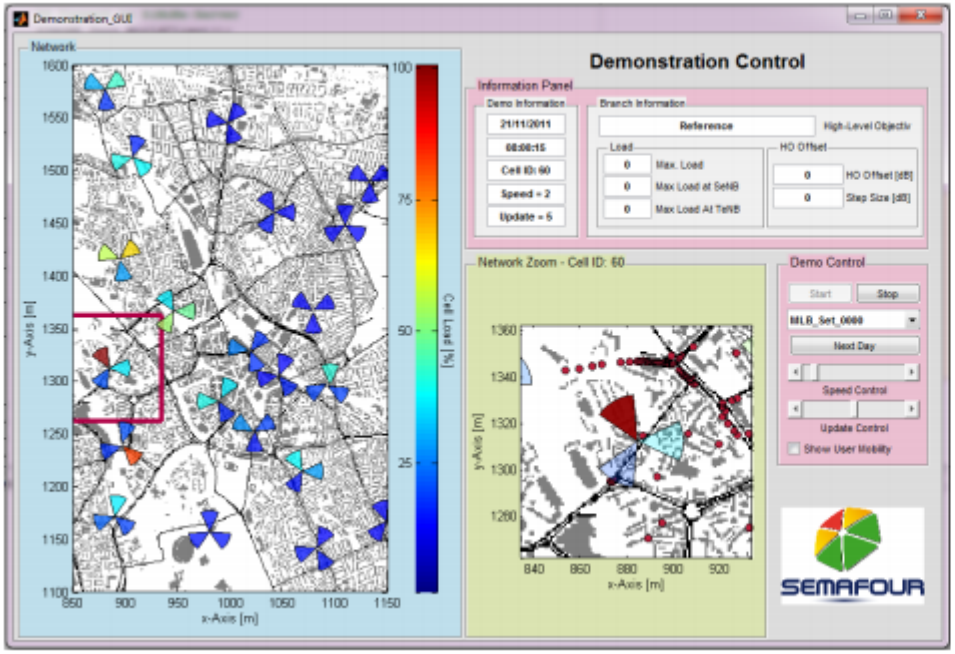
Deliver safe superior customer experience

Manage up to 10 times more users

Reduce latency to milliseconds

Make networks self-aware, self-adaptable, and intelligent

Example of traditional network management tools



What are we talking about? FCAPS model

- Network Management Tasks
  - fault management
  - configuration management
  - accounting management
  - performance management
  - security management
  - inventory management

Technologies in Traditional Network Management

- Simple Network Management Protocol (SNMP): SNMP is a standard protocol used for network management and monitoring of network-attached devices.
- Command Line Interface (CLI): CLI allows administrators to interact with networking devices through text-based commands.
- Remote Monitoring (RMON): RMON provides advanced monitoring capabilities, allowing for remote monitoring and analysis of network traffic.

Technologies in Traditional Network Models in Traditional Network Management

- FCAPS Model: FCAPS (Fault, Configuration, Accounting, Performance, Security) is a network management framework used for organizing different functions of network management.
- OSI Management Framework: The OSI (Open Systems Interconnection) model provides a reference framework for understanding and implementing network management protocols and systems.
- ITIL Framework: ITIL (Information Technology Infrastructure Library) offers best practices for managing IT services, including network management processes and procedures.

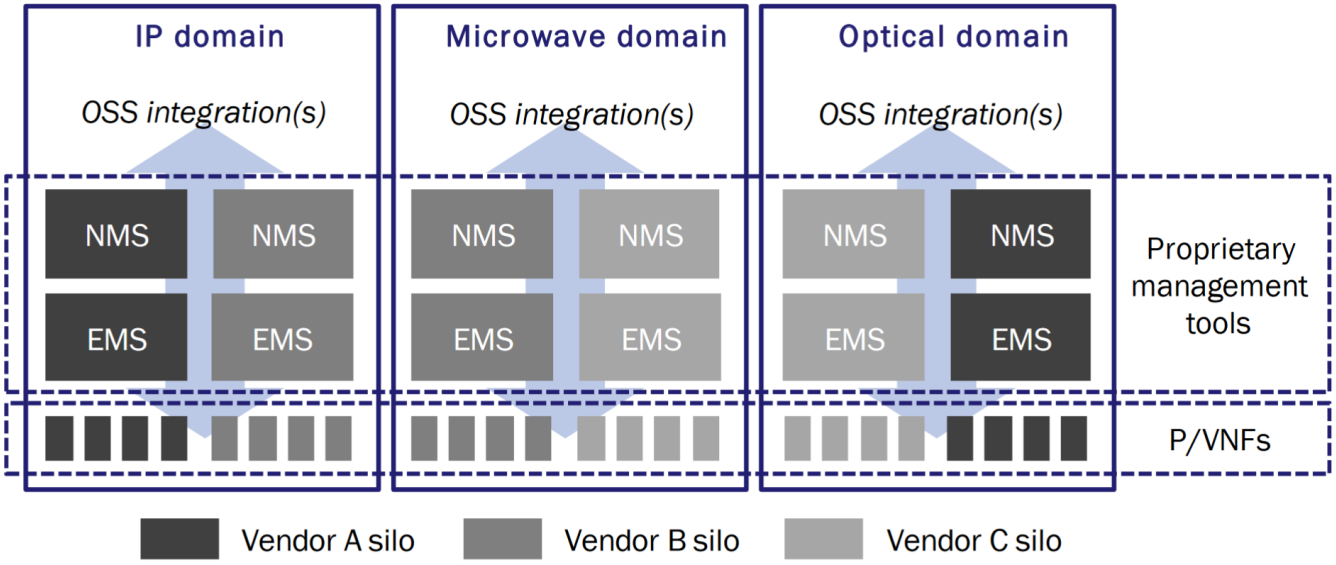
Functionalities in Traditional Network Management

- Fault Management: Detecting, isolating, and resolving network faults to ensure continuous network operation.
- Configuration Management: Managing network configurations, including device settings, policies, and access controls.
- Performance Management: Monitoring and optimizing network performance to meet service level agreements (SLAs) and user expectations.
- Accounting Management: Tracking network resource usage for billing, auditing, and capacity planning purposes.
- Security Management: Implementing and maintaining security measures to protect network assets and data from unauthorized access and malicious attacks.

Legacy Networks

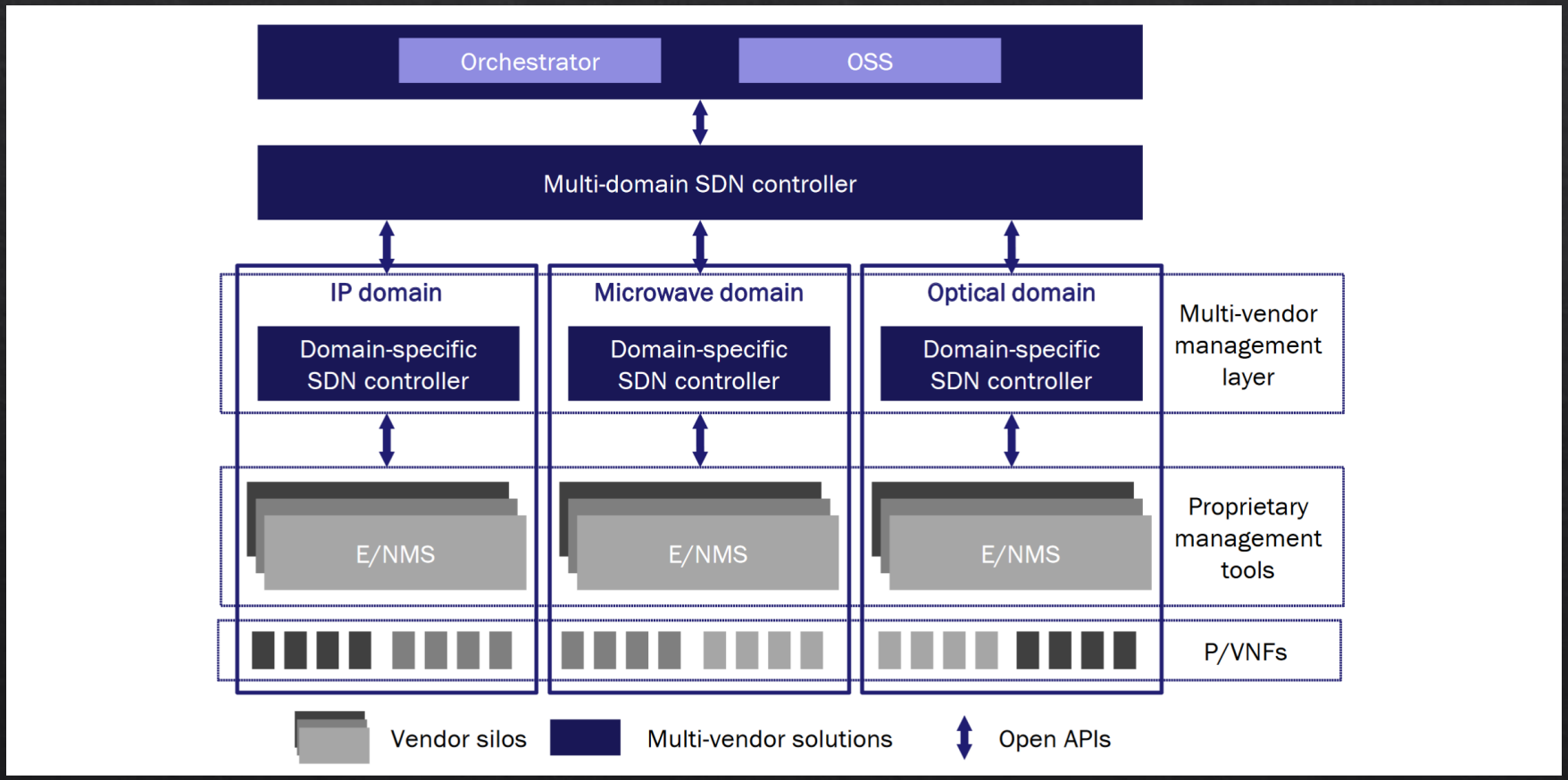
Figure 1 shows an example of a disaggregated network where the CSP has selected different vendors for different solutions and deployed best of breed physical and virtual network functions (P/VNFs) across network domains. Many vendors continue to build their applications without open APIs that require the vendor's own EMSs and NMSs, creating a layer of proprietary management tools made up of individual vendor silos. Each silo will then need to be integrated with the CSP's OSS and will typically result in domain-specific interfaces (for example, CORBA, SNMP, MTOSI, XML, FTP, REST, CLI).

Figure 1: Example of management silos across network domains and different vendors' solutions



Source: Analysys Mason, 2021

Hierarchical multi-vendor, SDN-based network management architecture for the transport network



Trends

- **ZERO-TOUCH AUTOMATION**: Intensive Automation is providing rapid provisioning; what used to take weeks and months now takes seconds and minutes. More complex network implementations including reactive network changes, zero downtime upgrades, and automatic threat response.
- Frees up network engineers' time, projects that generate revenue vs. just keeping the lights on. The same person who used to manage 10 network devices can now manage 1,000 network devices or start working on a next-generation, self-tuning monitoring system.

Trends

- **IoT and hyper connectivity will fundamentally disrupt traditional Network Management and security safeguards**
- **Network Management and security will ultimately be driven by machine learning and AI.**
- **User experience will be leveraged as a competitive differentiator.** Today, the value of a customer facing service is measured in high availability, security and performance. While these are important, what isn't emphasized is the user experience of that service, but this is because it is difficult to measure. Service providers will begin to quantify user sentiment, which is typically subjective, through the use of Natural Language Processing technology that can interpret human communication channels (e.g. Twitter, Facebook, message boards, etc.) and measure satisfaction.

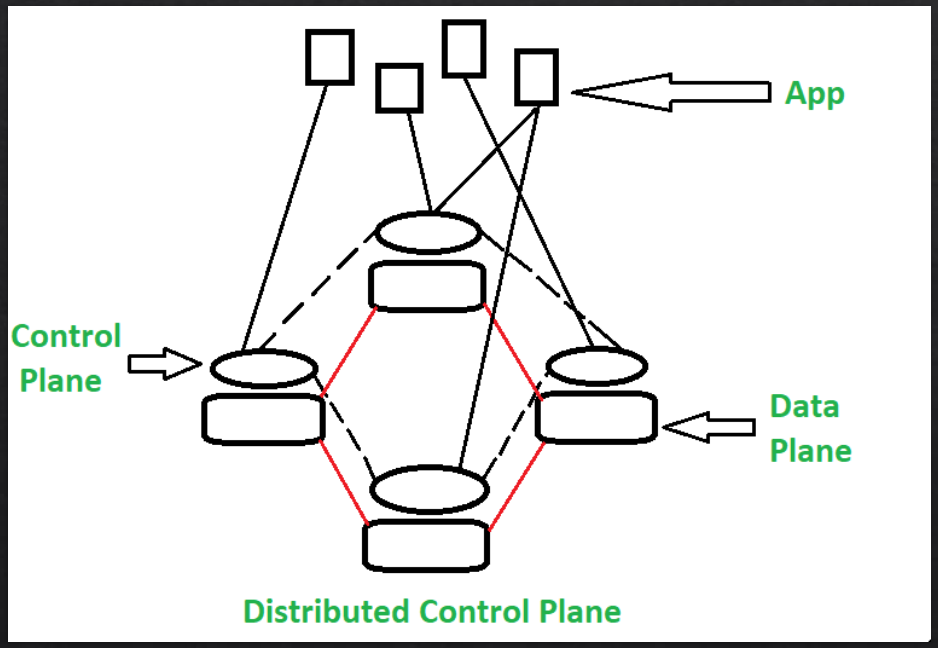
Trends

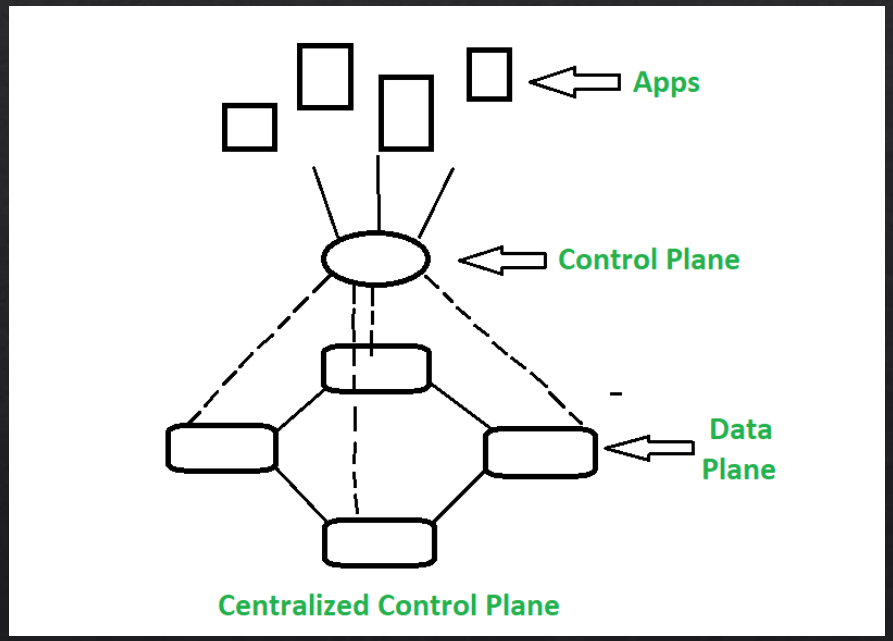
- **Deployment of next generation networks**
  - The traditional network is hardware dependent, and runs on fragmented and sometimes inefficient technologies that can result in performance inequalities from location to location
  - Next generation networks, which will largely be software defined and have a management plane will provide IT with the ability to leverage the right network paths, assign appropriate priority to network traffic and ensure the health of network at all locations. These networks will also incorporate an integrated, end-to-end view of the user experience from the data center to the end devices at the edge so that anything that may jeopardize performance is identified and managed before the end user is impacted.
- **Emerging technologies such as augmented and virtual reality, as well as IoT, will drive the need for scaled and automated network management.**

Trends

- Networks becoming
  - More programmatic
  - Defined by owners and operators, not vendors
  - Faster changing, to meet operator needs
  - Lower opex, capex and power
- Abstractions
  - Will shield programmers from complexity
  - Make behavior more provable

Legacy vs Software Defined Network Arxhitecture





Impact on Network Management

- Traditional network refers to the old conventional way of networking which uses fixed and dedicated hardware devices such as routers and switches to control network traffic.
- Inability to scale and network security and Performance are the major concern now a days in the current growing business situation so that SDN is taking control to traditional network.
- Traditional network is static and based on hardware network appliances.

Differences in Network Management

S.No.	SDN	TRADITIONAL NETWORK
01.	Software Defined Network is virtual networking approach.	Traditional network is the old conventional networking approach.
02.	Software Defined Network is centralized control.	Traditional Network is distributed control.
03.	This network is programmable.	This network is non programmable.
04.	Software Defined Network is open interface.	Traditional network is closed interface.
05.	In Software Defined Network data plane and control plane are decoupled by software.	In traditional network data plane and control plane are mounted on same plane.
06.	It supports automatic configuration so it takes less time.	It supports static/manual configuration so it takes more time.

Differences in Network Management

07.	It can prioritize and block specific network packets.	It leads all packets in the same way no prioritization support.
08.	It is easy to program as per need.	It is difficult to program again and to replace existing program as per use.
09.	Cost of Software Defined Network is low.	Cost of Traditional Network is high.
10.	Structural complexity is low in Software Defined Network.	Structural complexity is high in Traditional Network.
11.	Extensibility is high in Software Defined Network.	Extensibility is low in Traditional Network.
12.	In SDN it is easy to troubleshooting and reporting as it is centralized controlled.	In Traditional network it is difficult to troubleshoot and report as it is distributed controlled.

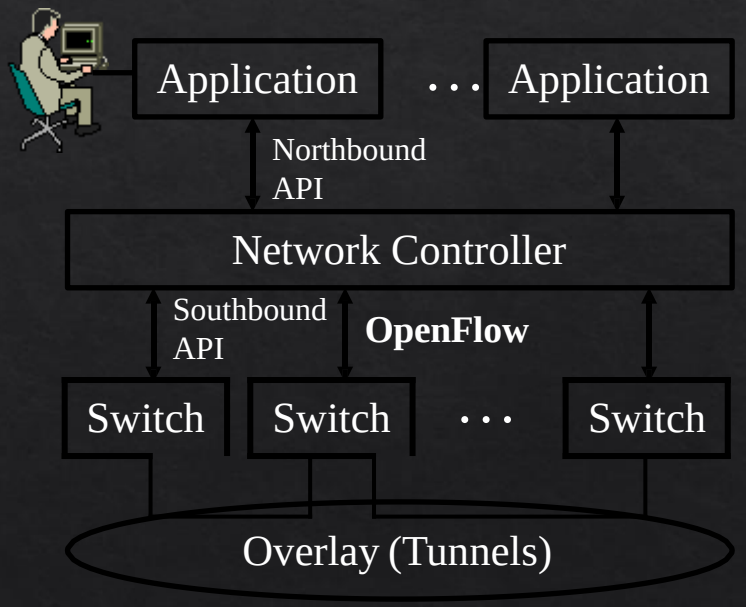
Software Defined Network (SDN)

- SDN stands for Software Defined Network which is networking architecture approach. It enables the control and management of network using software applications. Through Software Defined Network (SDN) networking behavior of entire network and its devices are programmed in centrally controlled manner through software applications using open APIs.

- Software Defined Network improves performance by network virtualization. In SDN software controlled applications or APIs work as basis of complete network management that may be directing traffic on network or to communicate with underlying hardware infrastructure. So in simple we can say SDN can create virtual network or it can control traditional network with the help of software.

Origins 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.



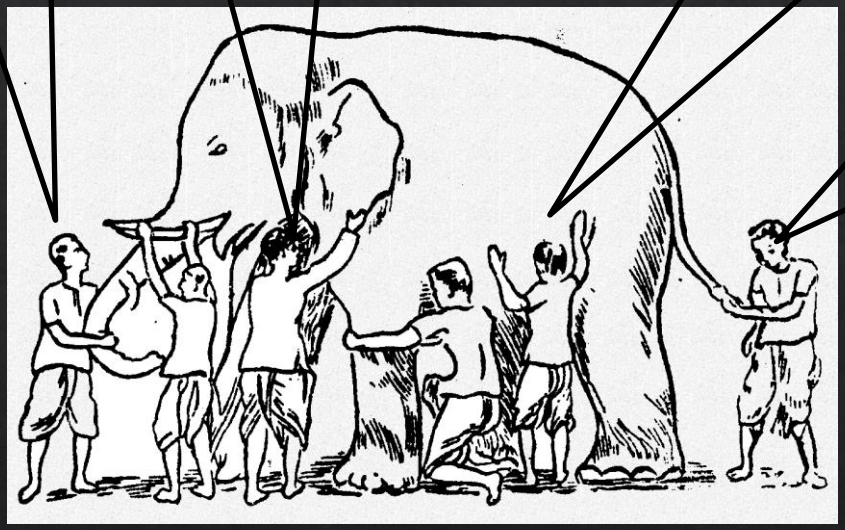
What is SDN?

SDN=OpenFlow

SDN=Standard Southbound API

SDN = Centralization of control plane

SDN = Separation of Control and Data Planes



- All of these are mechanisms.
- SDN is *not* a mechanism.
- It is a framework to solve a set of problems ⇒ Many solutions

Original Definition of SDN

1. **"*What is SDN?***
2. *The physical separation of the network control plane from the forwarding plane, and where a control plane controls several devices."*
3. Directly programmable
4. Agile: *Abstracting control from forwarding*
5. Centrally managed
6. Programmatically configured
7. Open standards-based vendor neutral

The above definition includes *How*.

8. Now many different opinions about *How.*
9. ⇒SDN has become more general. Need to define by *What*?

What = Why We need SDN?

1. **Virtualization**: Use network resource without worrying about where it is physically located, how much it is, how it is organized, etc.
2. **Orchestration**: Should be able to control and manage thousands of devices with one command.
3. **Programmable**: Should be able to change behavior on the fly.
4. **Dynamic Scaling**: Should be able to change size, quantity
5. **Automation**: To lower OpEx minimize manual involvement
   - Troubleshooting
   - Reduce downtime
   - Policy enforcement
   - Provisioning/Re-provisioning/Segmentation of resources
   - Add new workloads, sites, devices, and resources

Why We need SDN? (Cont)

6. **Visibility**: Monitor resources, connectivity
7. **Performance**: Optimize network device utilization
   - Traffic engineering/Bandwidth management
   - Capacity optimization
   - Load balancing
   - High utilization
   - Fast failure handling
8. **Multi-tenancy**: Tenants need complete control over their addresses, topology, and routing, security
9. **Service Integration**: Load balancers, firewalls, Intrusion Detection Systems (IDS), provisioned on demand and placed appropriately on the traffic path

Why We need SDN? (Cont)

10. **Openness**: Full choice of "How" mechanisms
    ⇒ Modular plug-ins
    ⇒ Abstraction:
    - Abstract = Summary = Essence = General Idea
      ⇒ Hide the details.
    - Also, abstract is opposite of concrete
      ⇒ Define tasks by APIs and not by how it should be done.
      E.g., send from A to B. Not OSPF.

Ref: http://www.networkworld.com/news/2013/110813-onug-sdn-275784.html

Ref: Open Data Center Alliance Usage Model: Software Defined Networking Rev 1.0," http://www.opendatacenteralliance.org/docs/Software_Defined_Networking_Master_Usage_Model_Rev1.0.pdf

SDN Definition

- SDN is a *framework* to allow network administrators to *automatically* and dynamically manage and control a *large number* of network devices, *services*, topology, traffic paths, and packet handling (quality of service) policies using high-level languages and APIs. Management includes provisioning, operating, *monitoring*, optimizing, and managing FCAPS (faults, configuration, accounting, *performance*, and security) in a *multi-tenant* environment.
- Key: Dynamic $\Rightarrow$ Quick
  Legacy approaches such as CLI were not quick particularly for large networks

Examples Alternative APIs

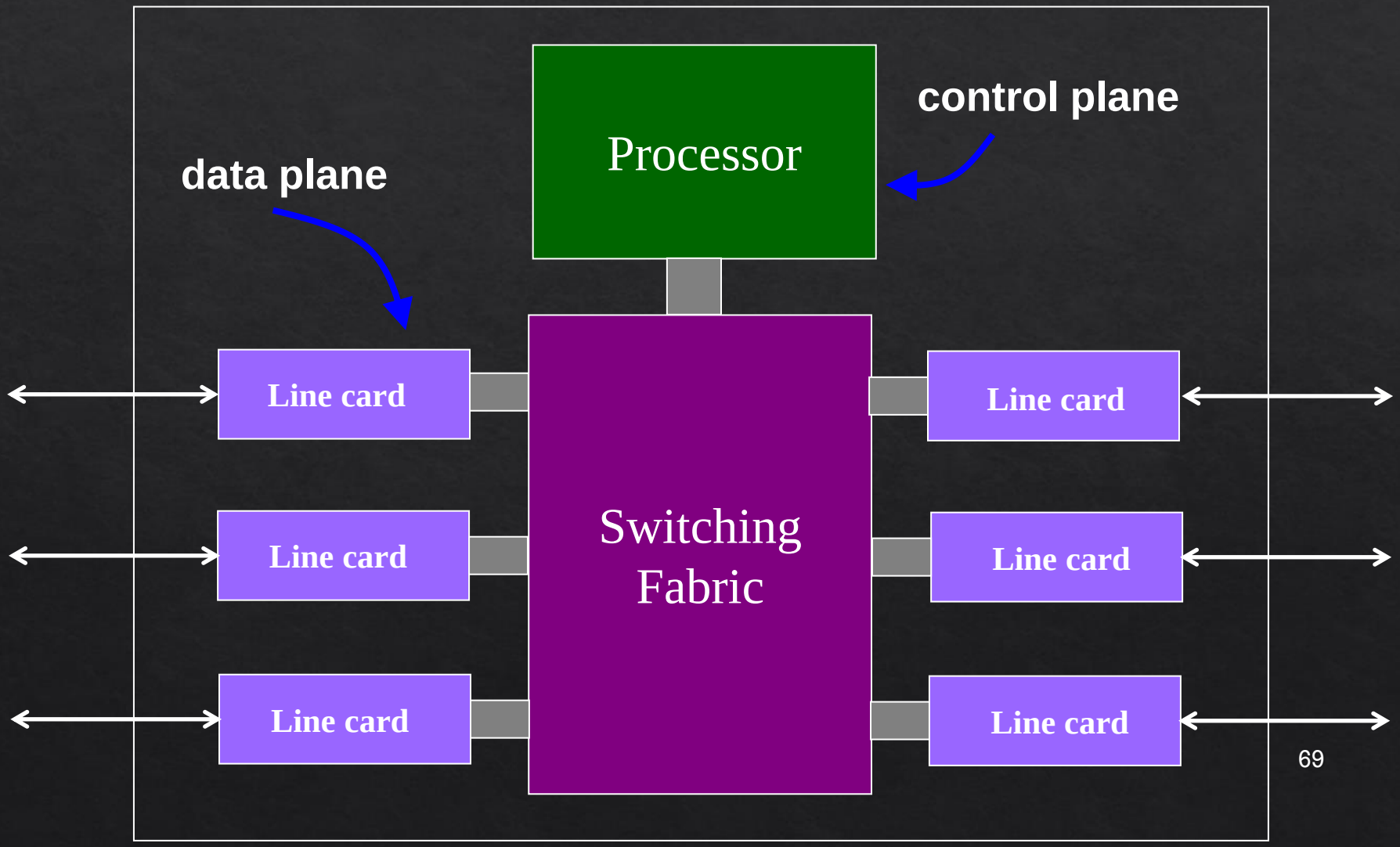
- Southbound APIs: XMPP (Juniper), OnePK (Cisco)
- Northbound APIs: I2RS, I2AEX, ALTO,
- Overlay: VxLAN, TRILL, LISP, STT, NVO3, PWE3, L2VPN, L3VPN
- Configuration API: NETCONF
- Controller: PCE, ForCES

Data, Control, and Management Planes

Timescales

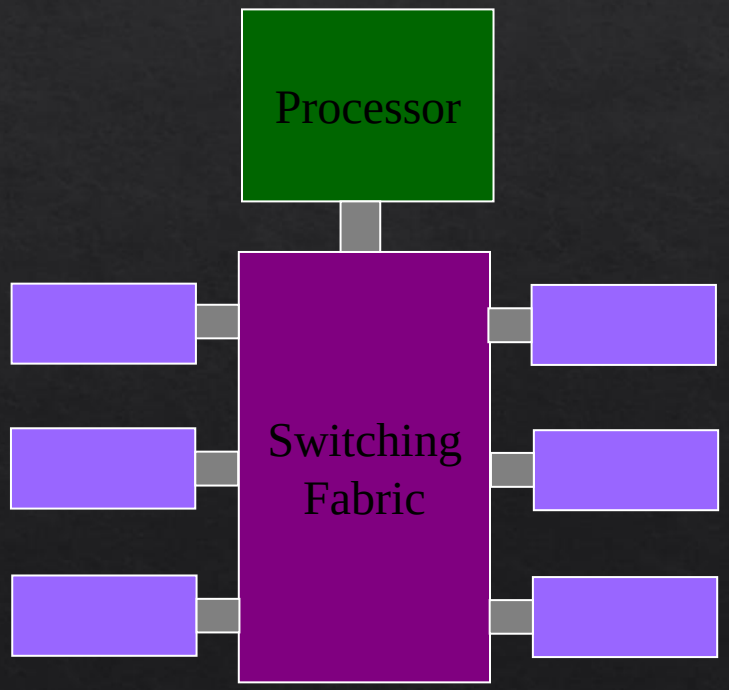
	Data	Control	Management
Time-scale	Packet (nsec)	Event (10 msec to sec)	Human (min to hours)
Tasks	Forwarding, buffering, filtering, scheduling	Routing, circuit set-up	Analysis, configuration
Location	Line-card hardware	Router software	Humans or scripts

Data and Control Planes



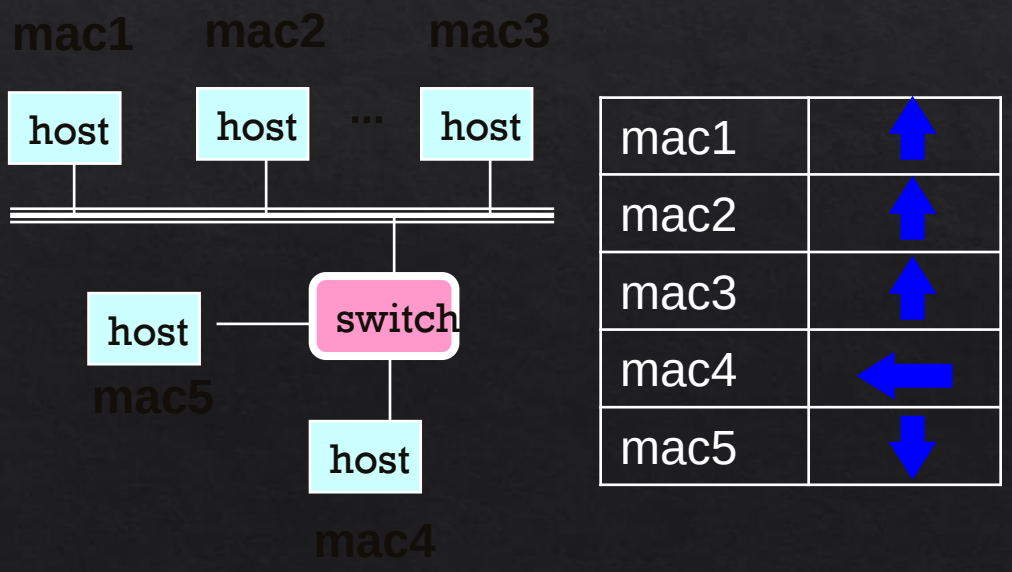
Data Plane

- Streaming algorithms on packets
  - Matching on some bits
  - Perform some actions
- Wide range of functionality
  - Forwarding
  - Access control
  - Mapping header fields
  - Traffic monitoring
  - Buffering and marking
  - Shaping and scheduling
  - Deep packet inspection



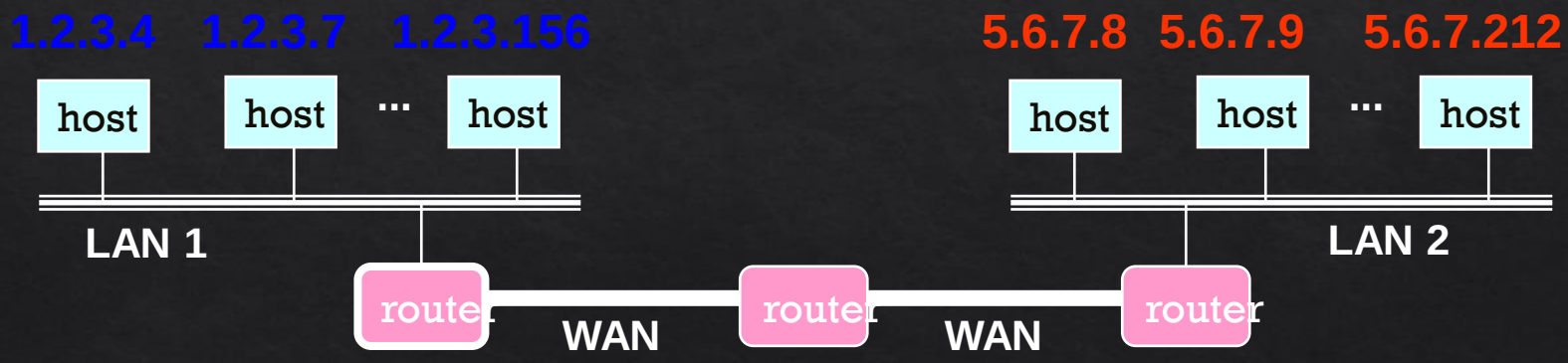
Switch: Match on Destination MAC

- MAC addresses are location independent
  - Assigned by the vendor of the interface card
  - Cannot be aggregated across hosts in LAN



Router: Match on IP Prefix

- IP addresses grouped into common subnets
  - Allocated by ICANN, regional registries, ISPs, and within individual organizations
  - Variable-length prefix identified by a mask length



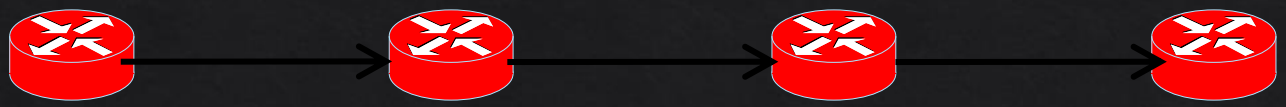
1.2.3.0/24	←
5.6.7.0/24	→

forwarding table

Prefixes may be nested. Routers identify the *longest matching* prefix.

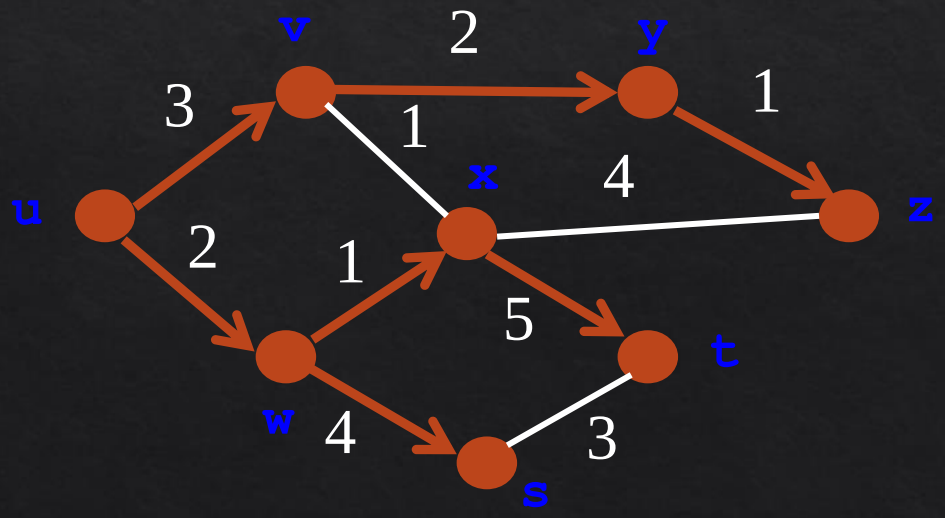
Forwarding vs. Routing

- Forwarding: data plane
  - Directing a data packet to an outgoing link
  - Individual router *using* a forwarding table
- Routing: control plane
  - Computing paths the packets will follow
  - Routers talking amongst themselves
  - Individual router ***creating*** a forwarding table



Distributed Control Plane

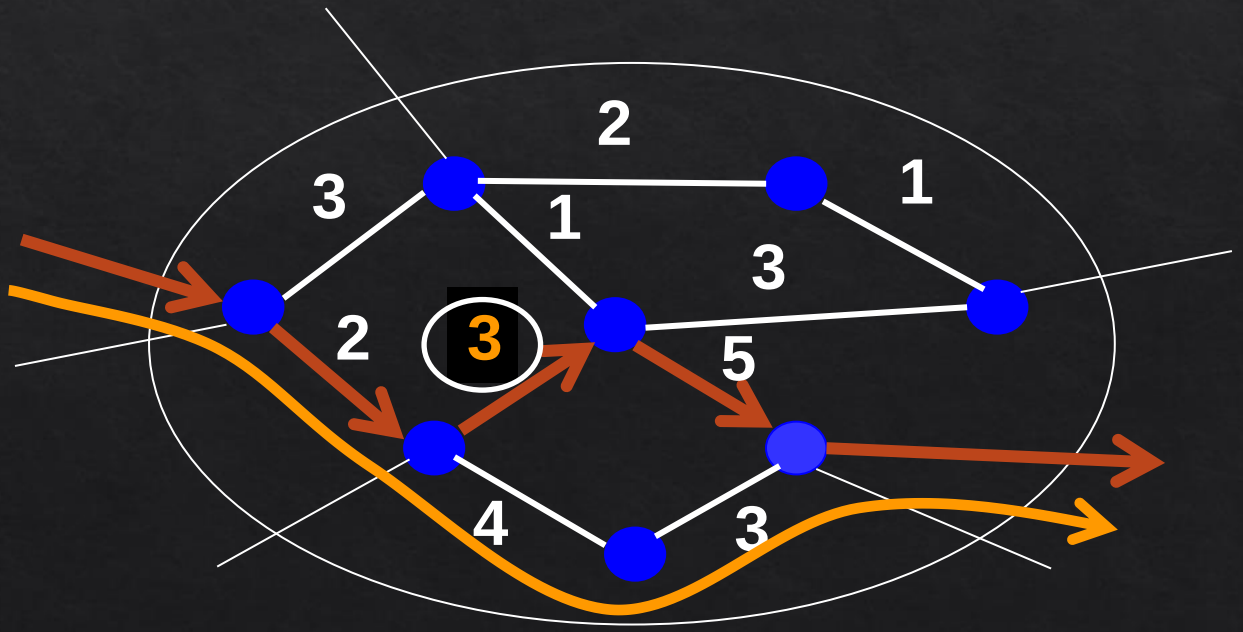
- Link-state routing: OSPF, IS-IS
  - Flood the entire topology to all nodes
  - Each node computes shortest paths
  - Dijkstra's algorithm



	link
v	(u,v)
w	(u,w)
x	(u,w)
y	(u,v)
z	(u,v)
s	(u,w)
t	(u,w)

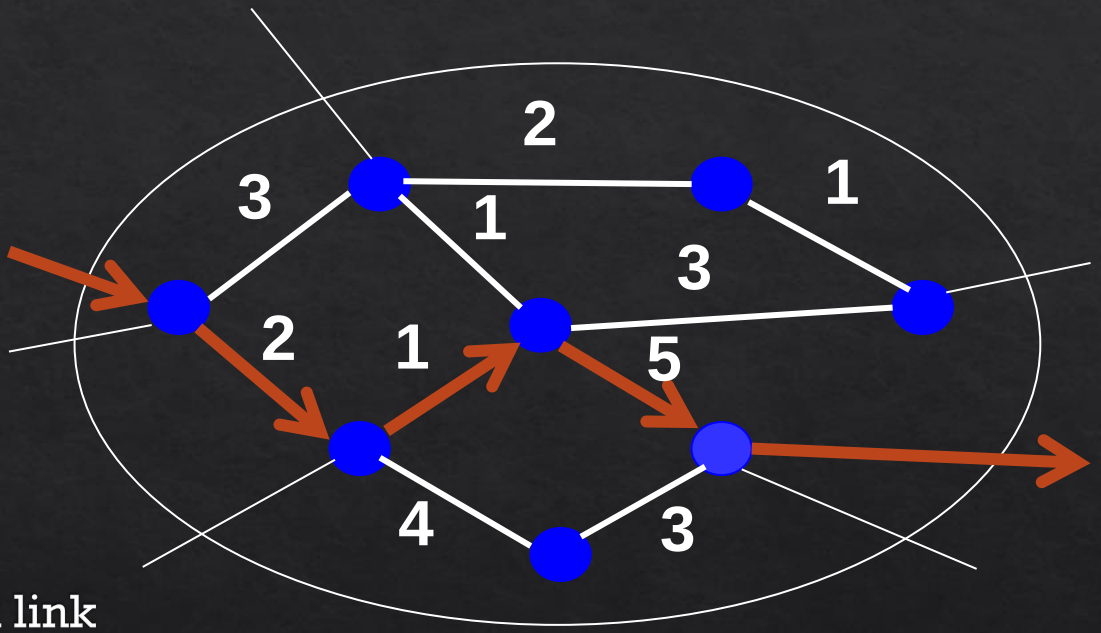
Traffic Engineering Problem

- Management plane: setting the weights
  - Inversely proportional to link capacity?
  - Proportional to propagation delay?
  - Network-wide optimization based on traffic?



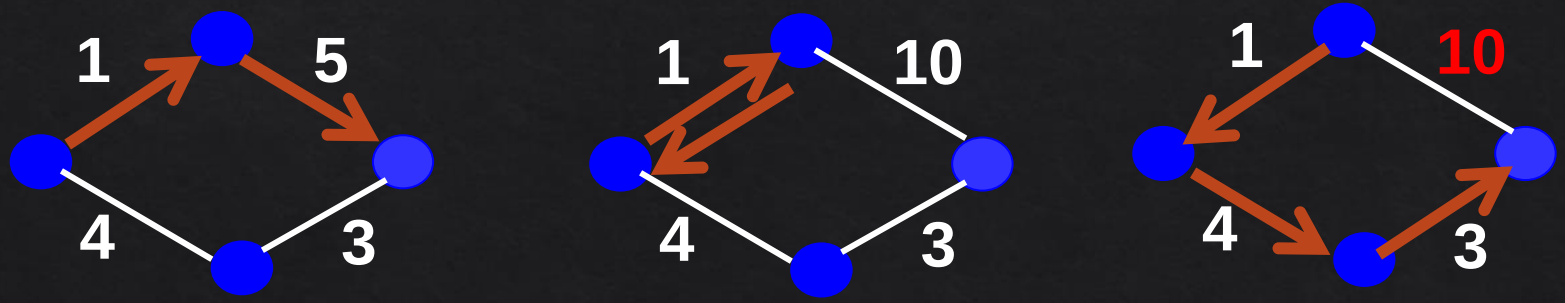
Traffic Engineering: Optimization

- Inputs
  - Network topology
  - Link capacities
  - Traffic matrix
- Output
  - Link weights
- Objective
  - Minimize max-utilized link
  - Or, minimize a sum of link congestion



Transient Routing Disruptions

- Topology changes
  - Link weight change
  - Node/link failure or recovery
- Routing convergence
  - Nodes temporarily disagree how to route
  - Leading to transient loops and blackholes



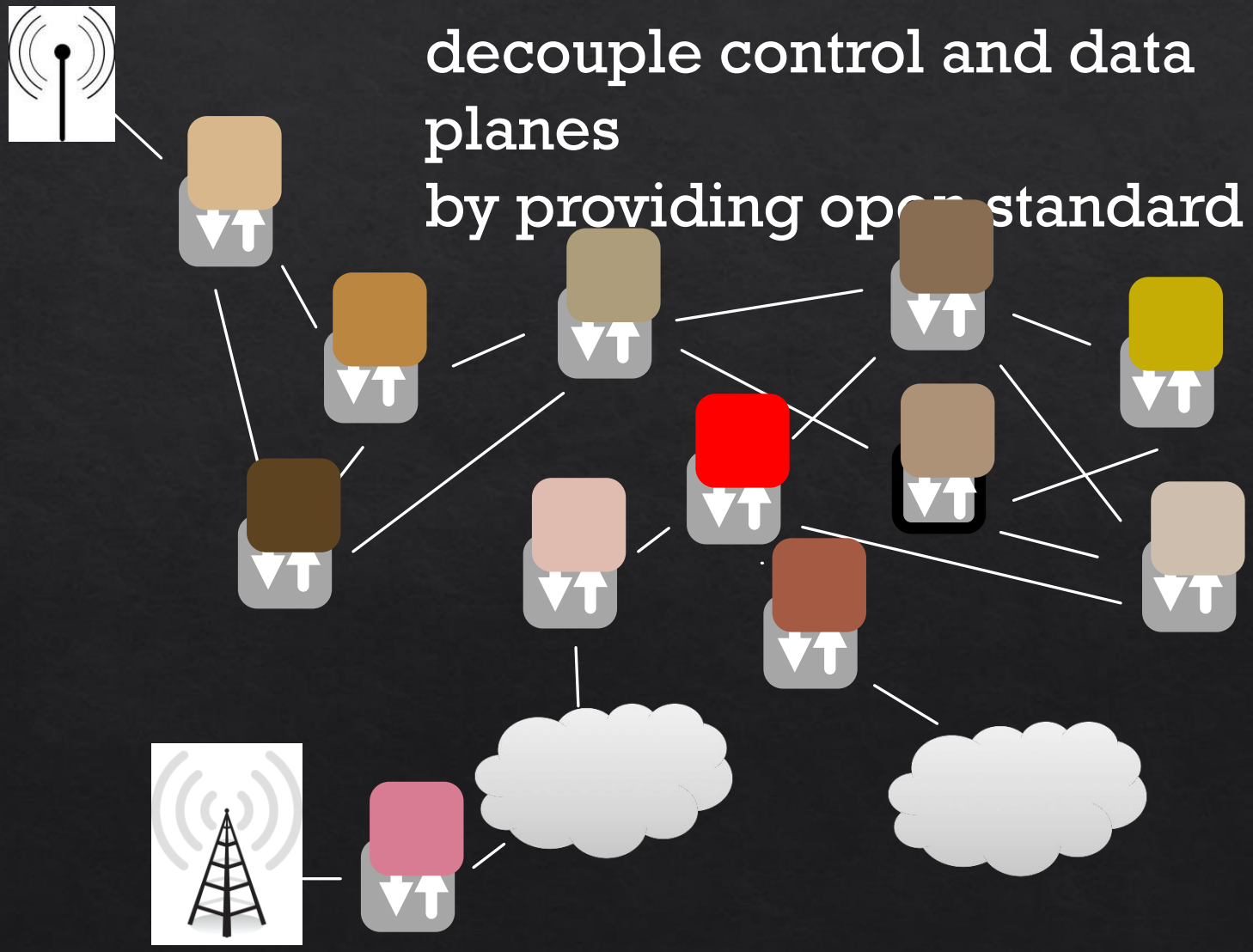
Management Plane Challenges

- Indirect control
  - Changing weights instead of paths
  - Complex optimization problem
- Uncoordinated control
  - Cannot control which router updates first
- Interacting protocols and mechanisms
  - Routing and forwarding
  - Naming and addressing
  - Access control
  - Quality of service
  - …

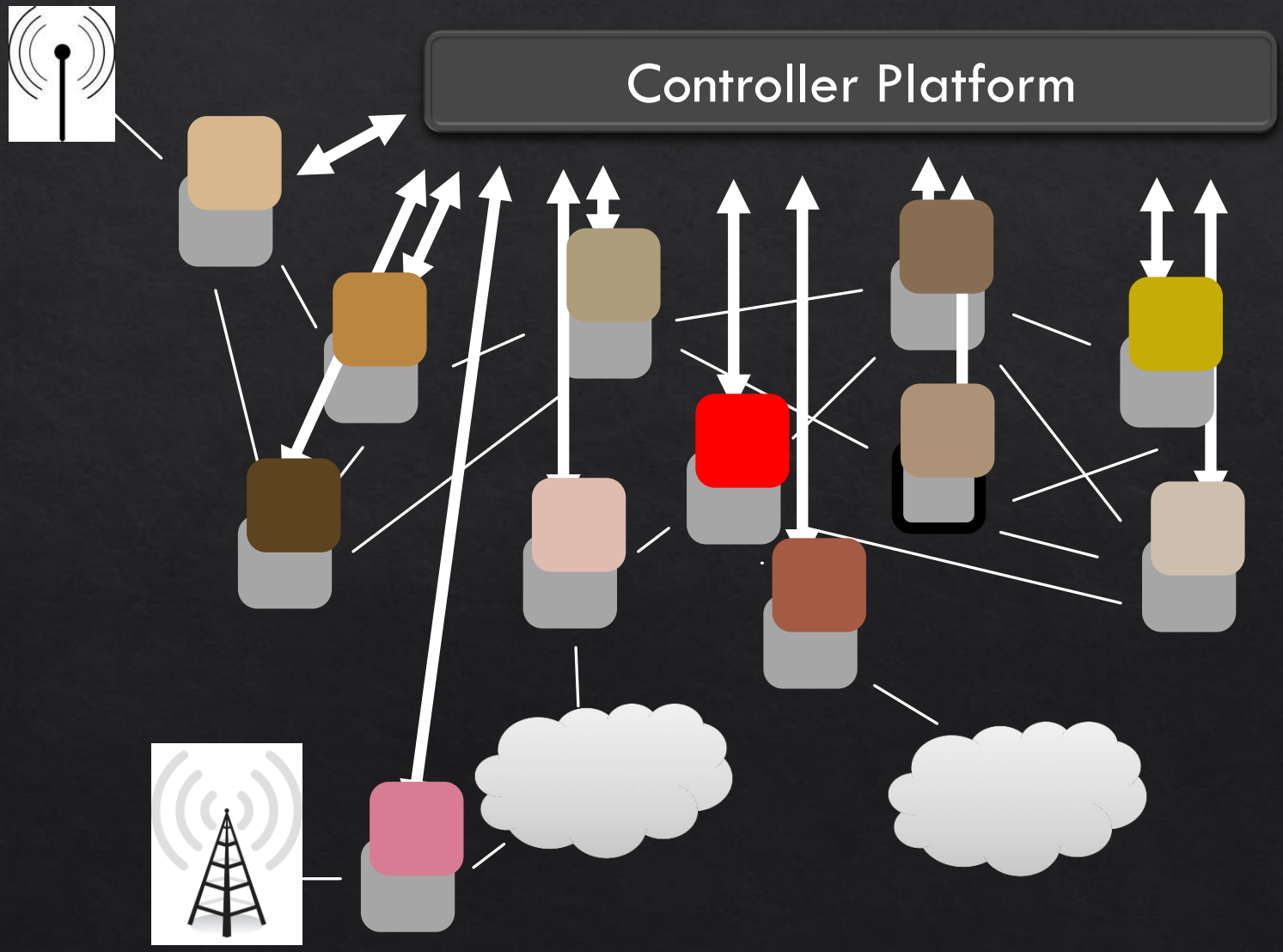
Software Defined Networking (high level view)

Control/Data Separation

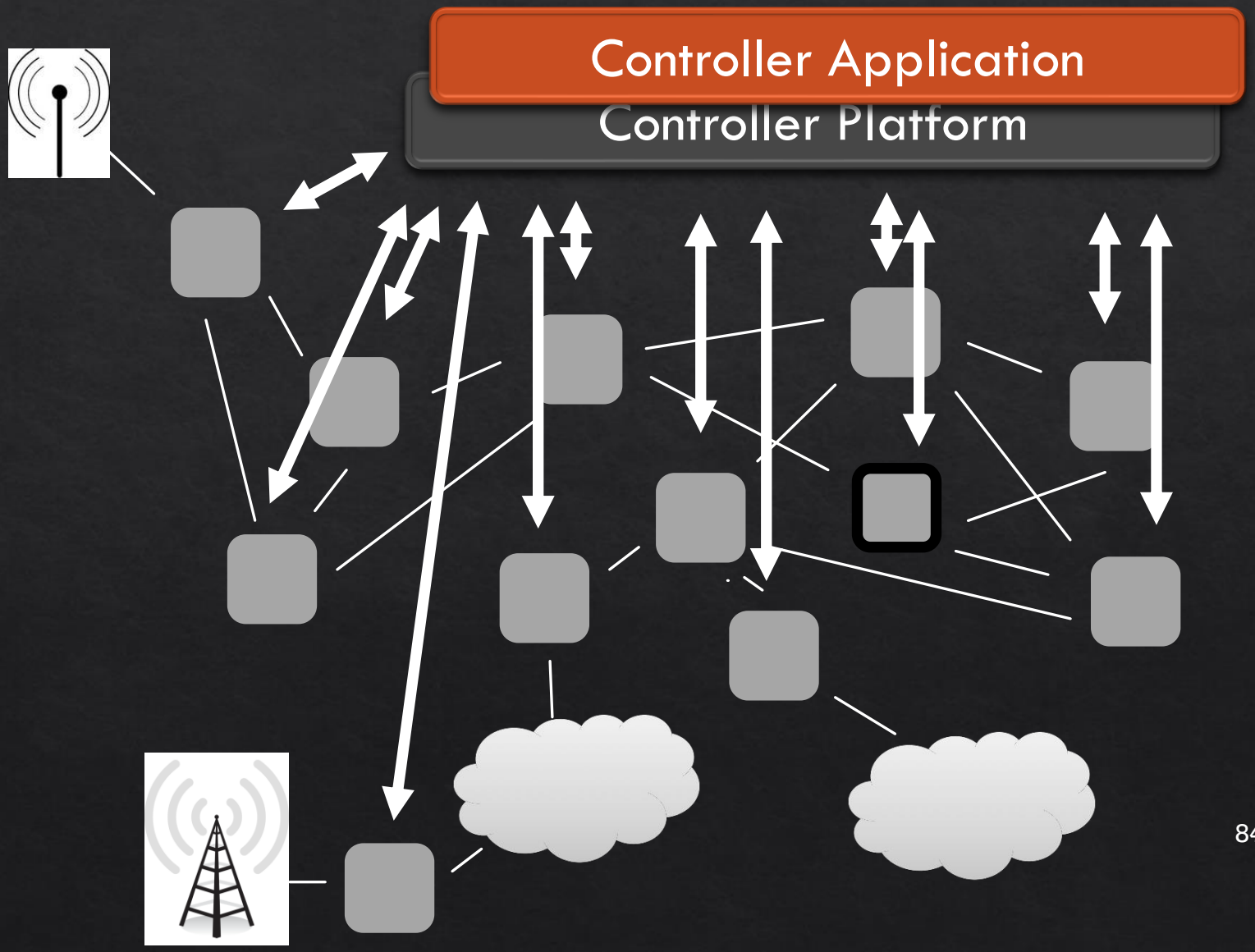
decouple control and data planes
by providing open standard API

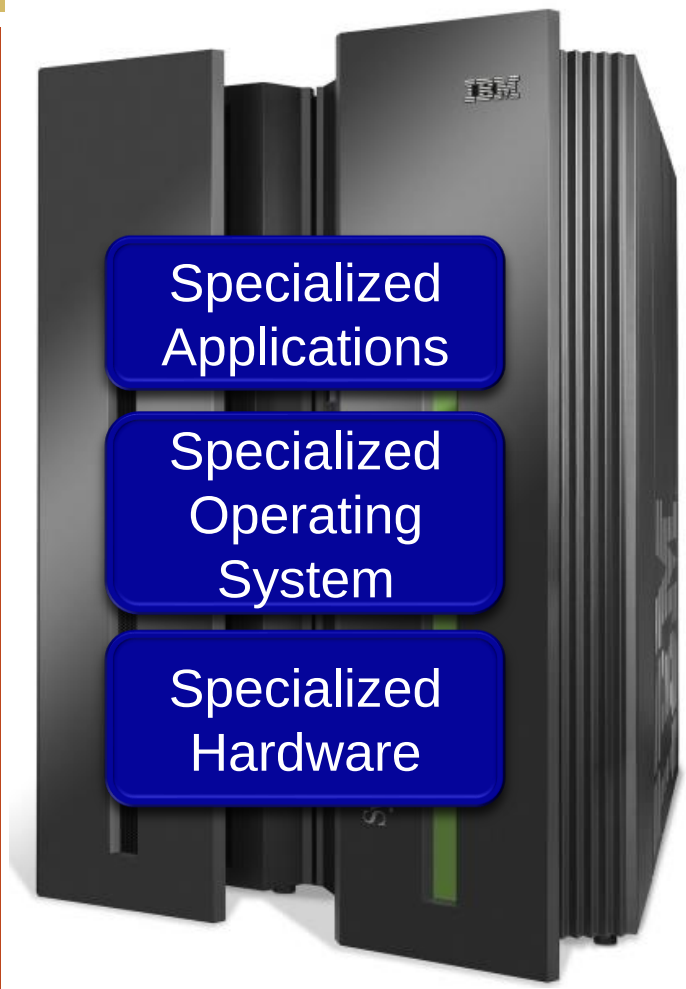


(Logically) Centralized Controller

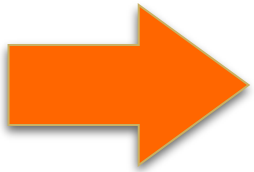


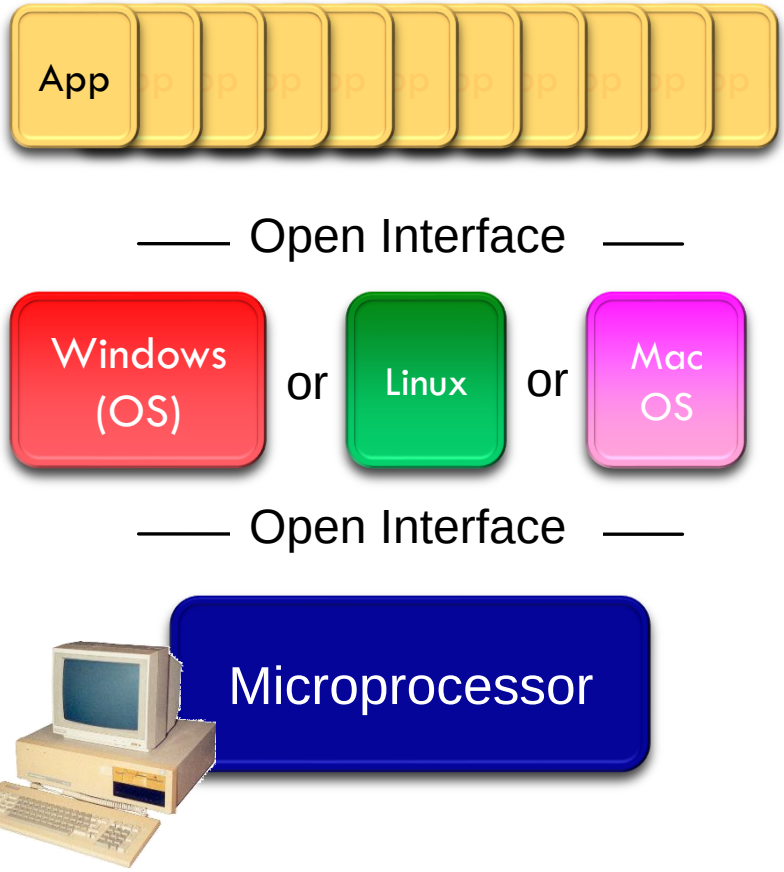
Protocols → Applications

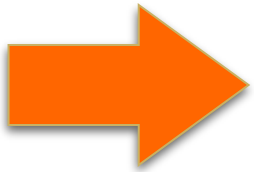




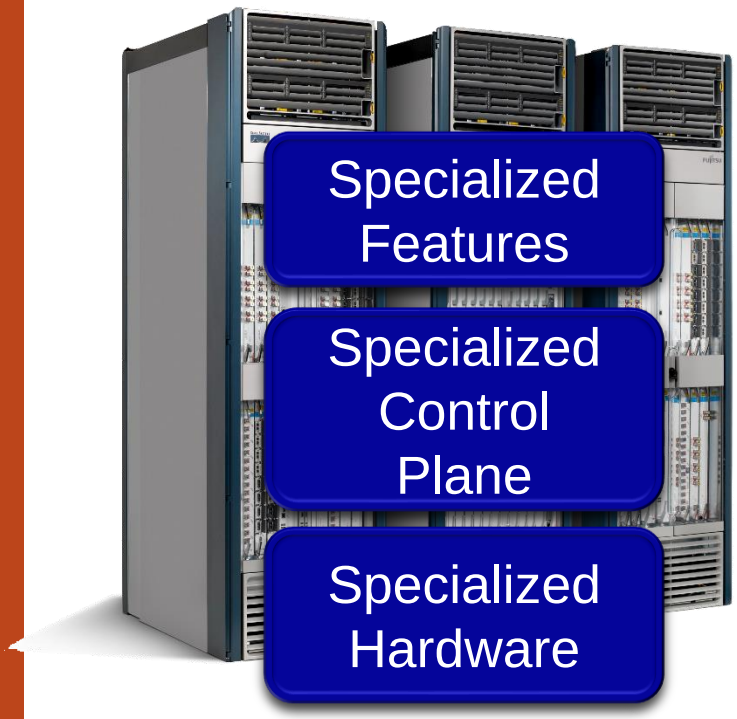
Vertically integrated
Closed, proprietary
Slow innovation
Small industry

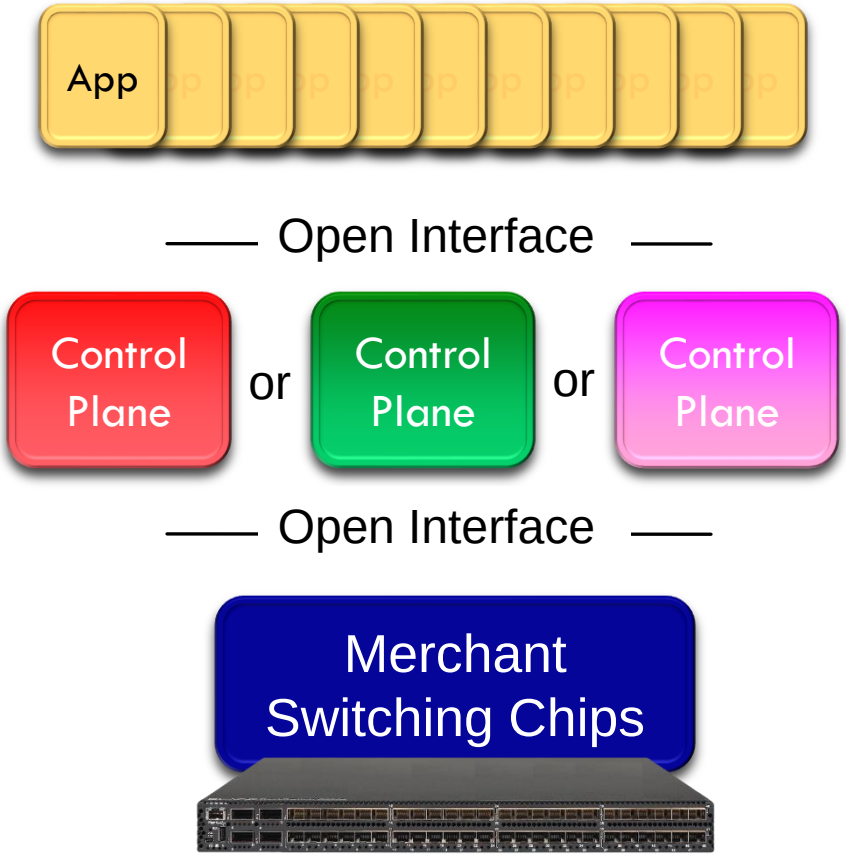






Horizontal
Open interfaces
Rapid innovation
Huge industry

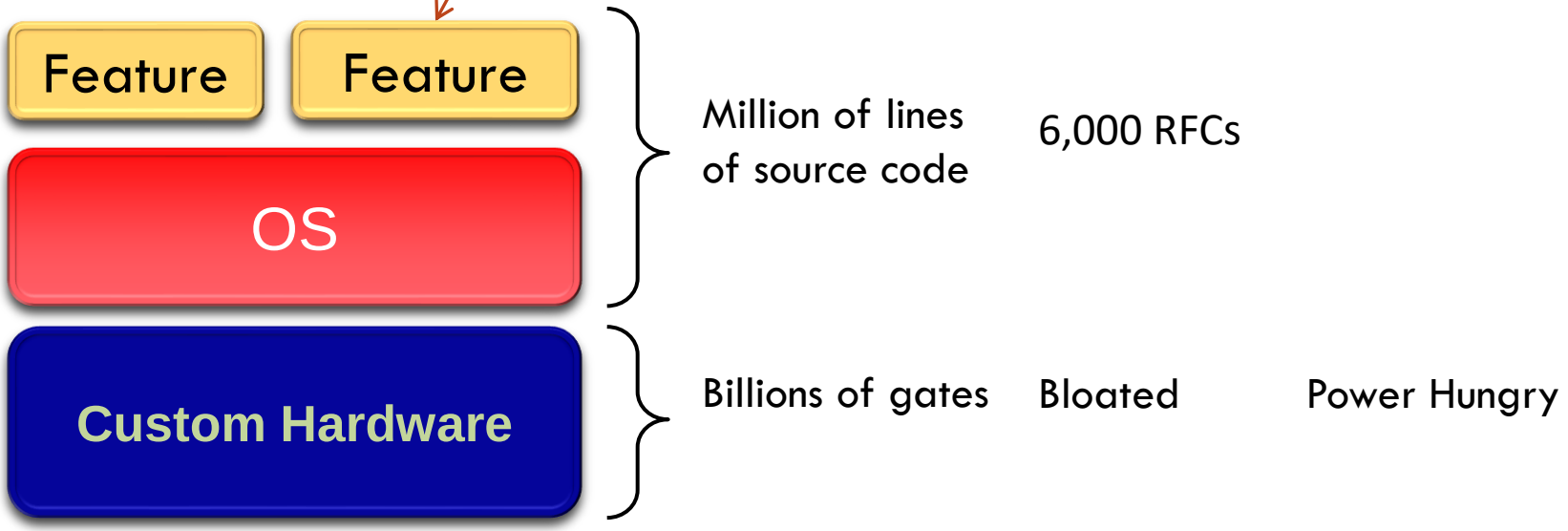




Vertically integrated
Closed, proprietary
Slow innovation

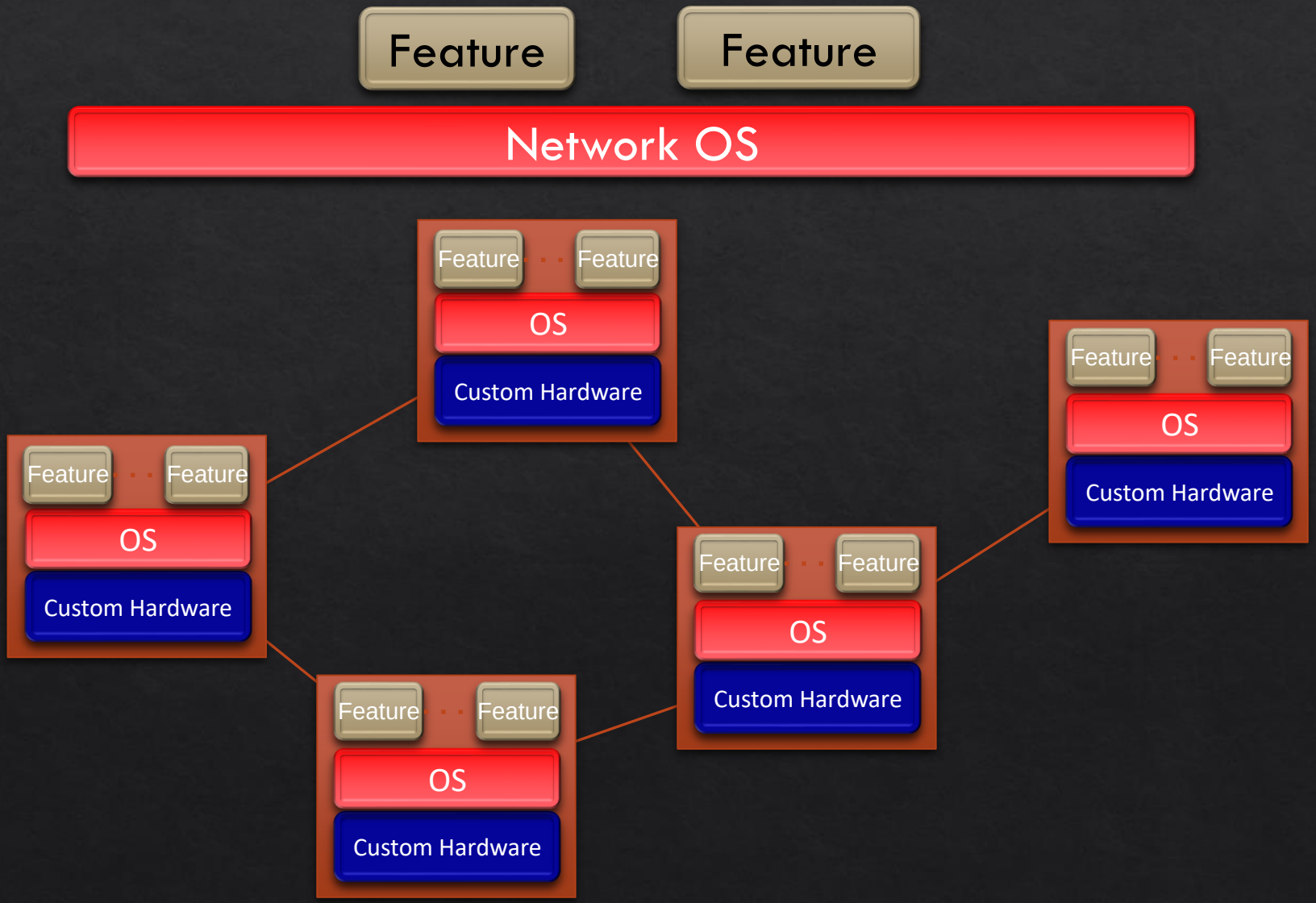
Horizontal
Open interfaces
Rapid innovation

Routing, management, mobility management, access control, VPNs, …

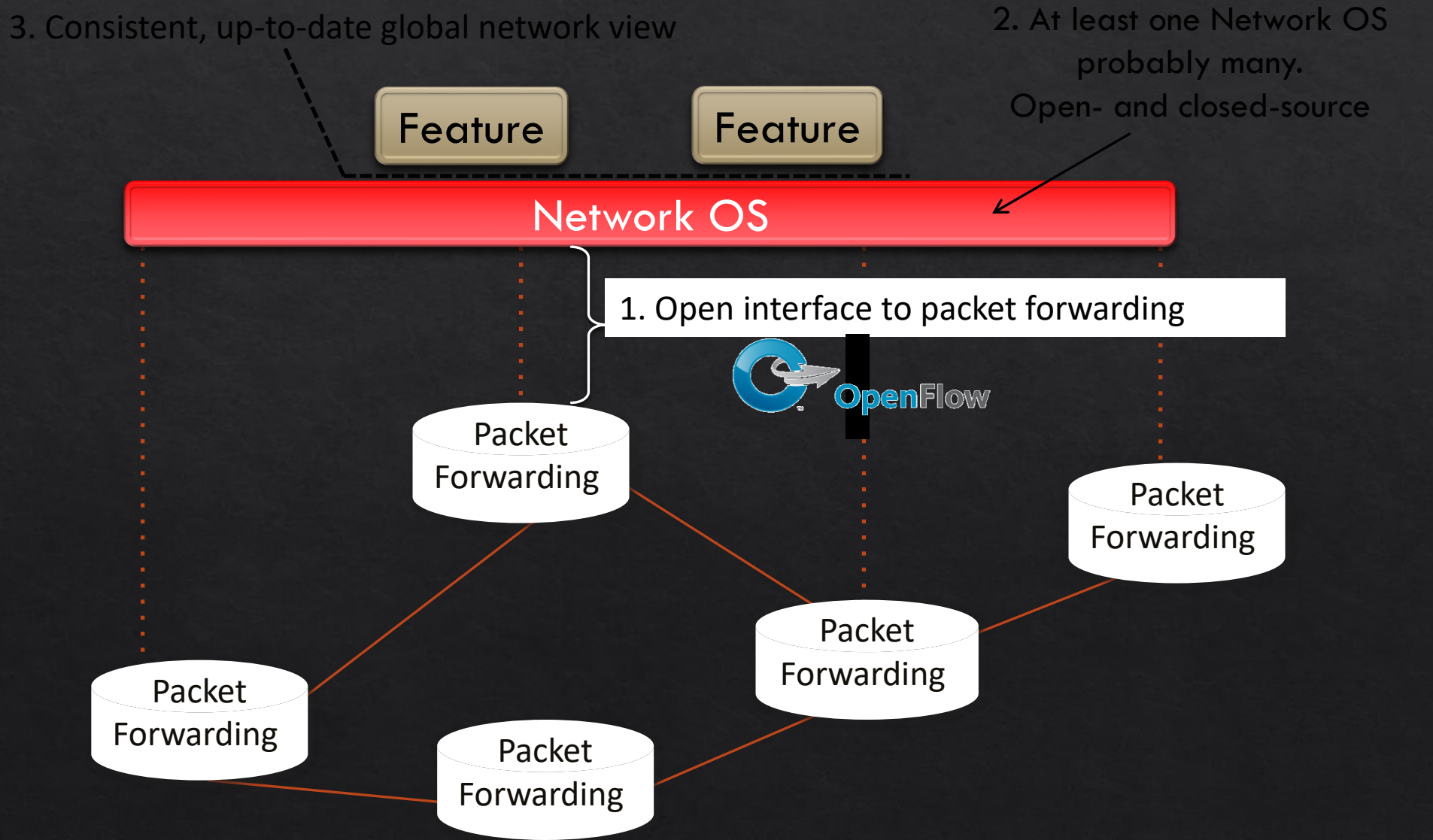


- Vertically integrated, complex, closed, proprietary
- Networking industry with "mainframe" mind-set

The network is changing



Software Defined Network (SDN)



Network OS

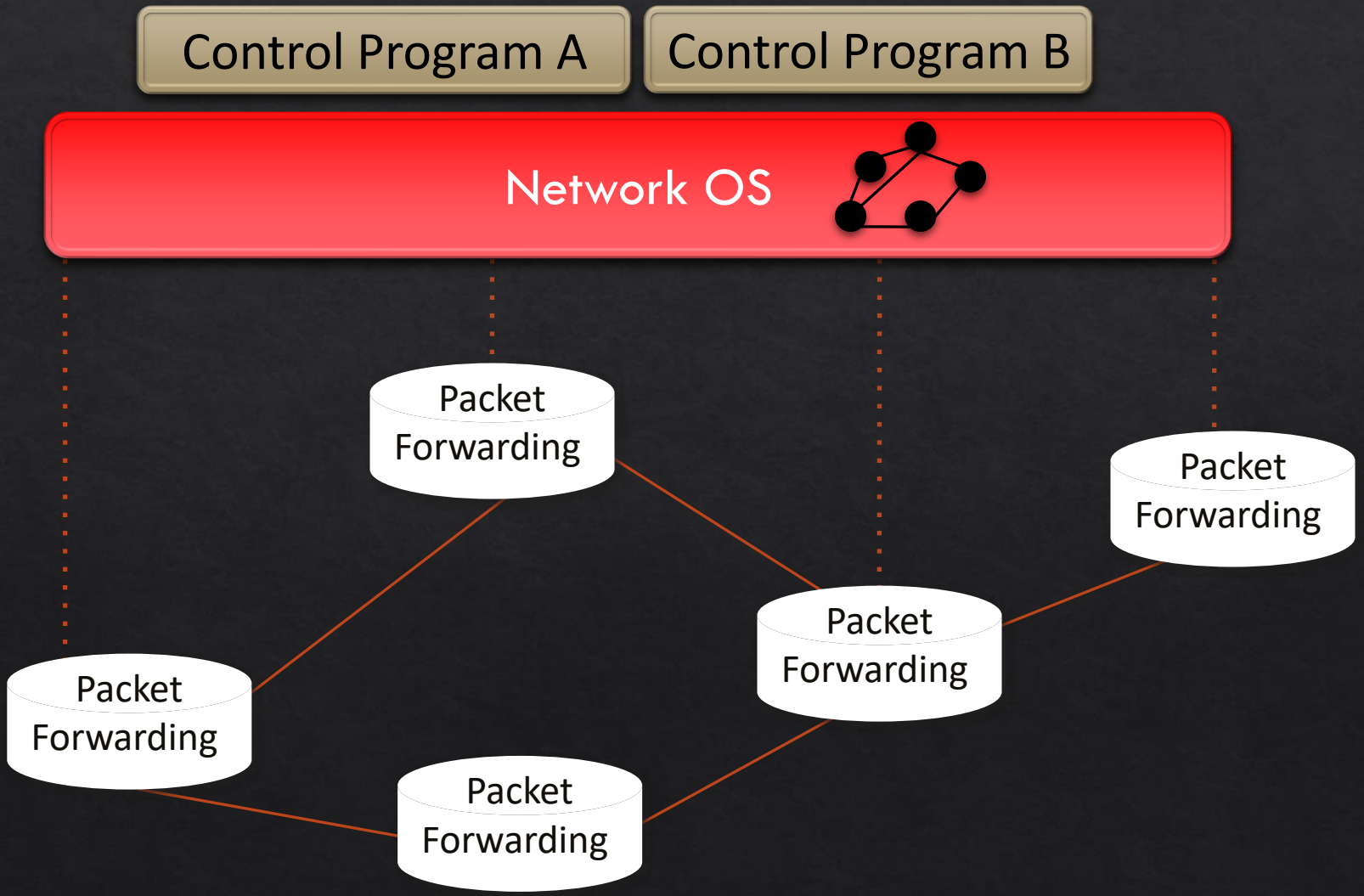
Network OS: distributed system that creates a consistent, up-to-date network view

- Runs on servers (controllers) in the network
- NOX, ONIX, Trema, Beacon, Maestro, … + more

Uses forwarding abstraction to:

- Get state information **from** forwarding elements
- Give control directives **to** forwarding elements

Software Defined Network (SDN)



Control Program

- Control program operates on view of network
  - **Input**: global network view (graph/database)
  - **Output**: configuration of each network device
- Control program is not a distributed system
  - Abstraction hides details of distributed state

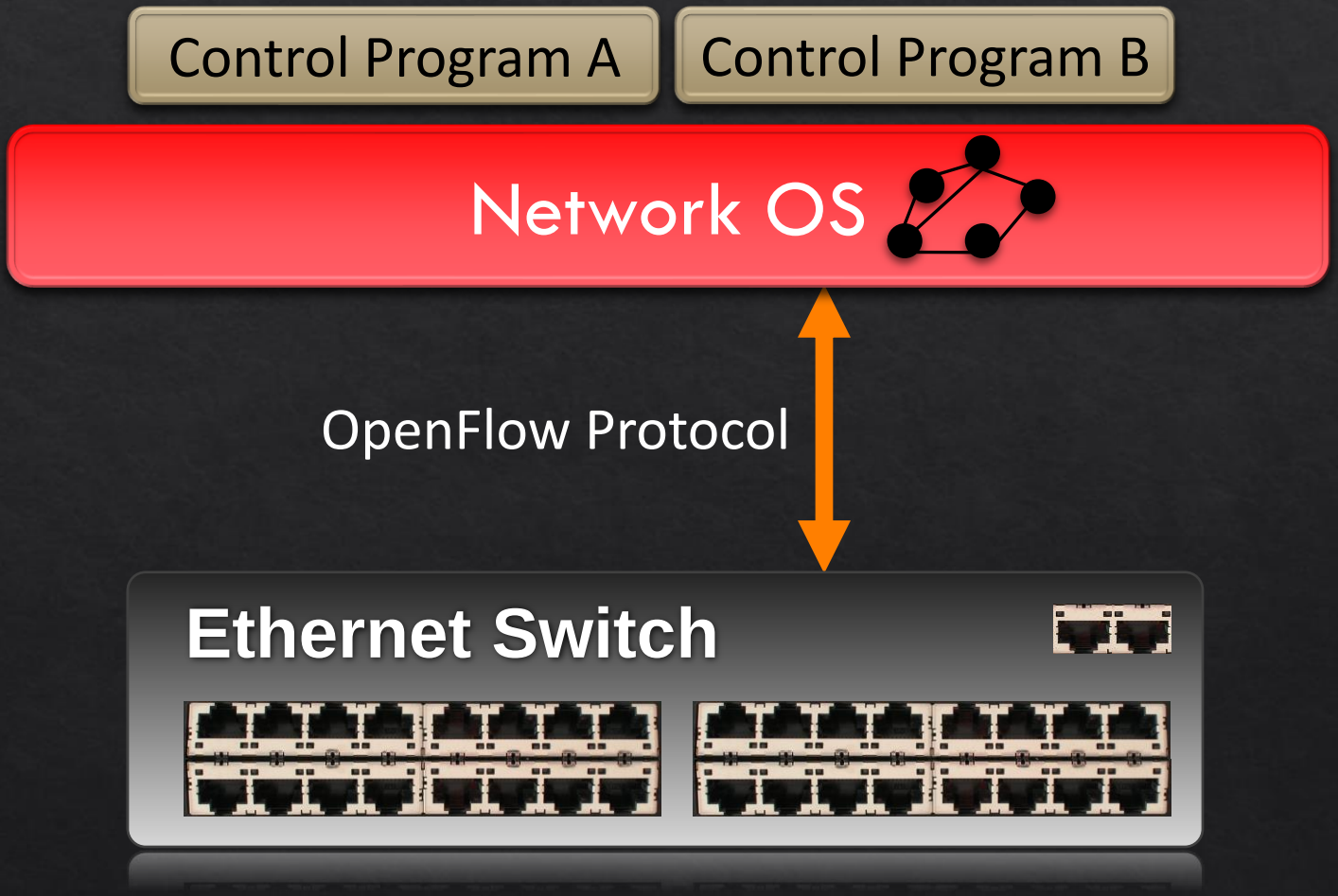
Forwarding Abstraction

Purpose: Abstract away forwarding hardware

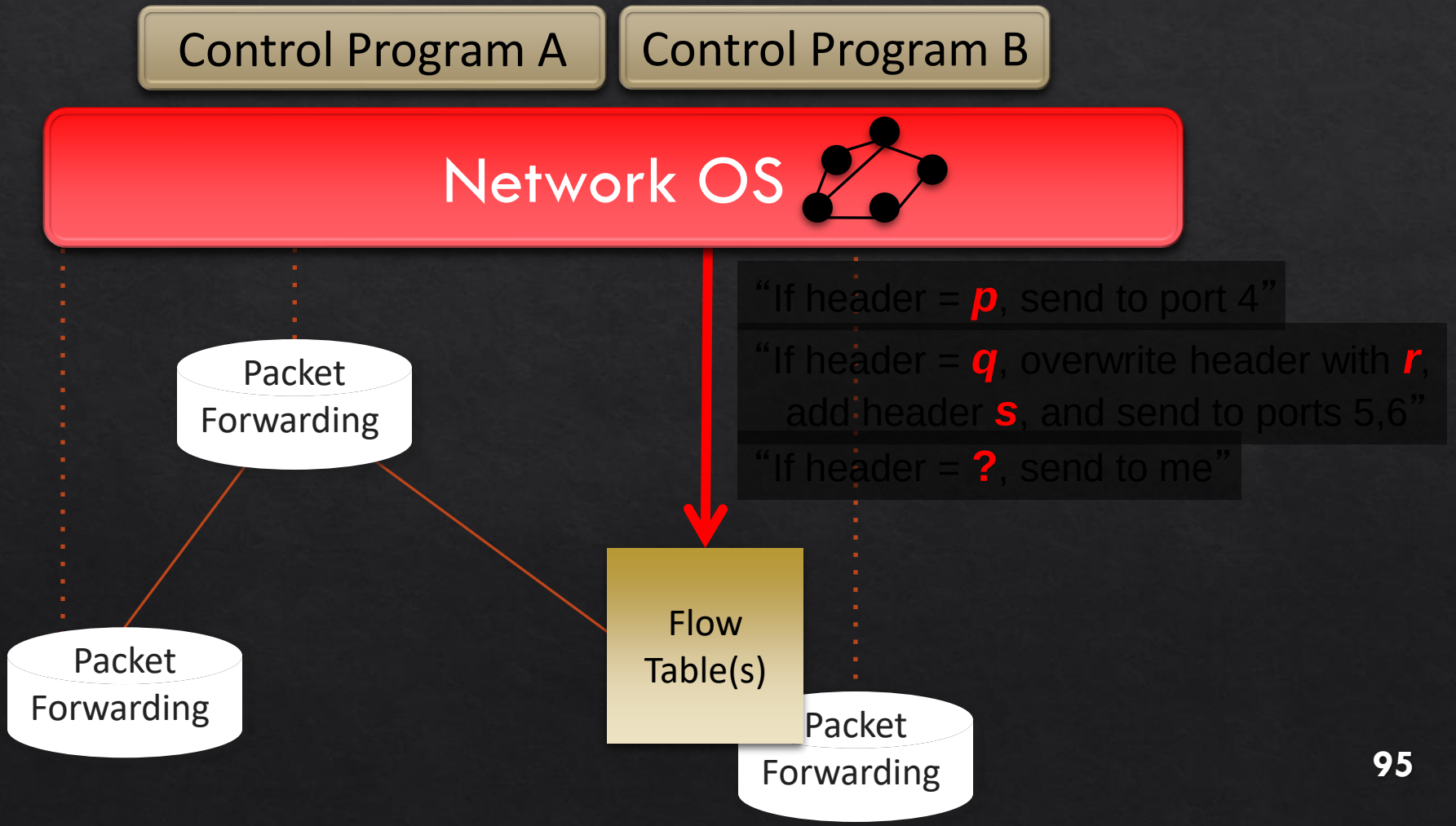
- Flexible
  - Behavior specified by control plane
  - Built from basic set of forwarding primitives
- Minimal
  - Streamlined for speed and low-power
  - Control program not vendor-specific

- OpenFlow is an example of such an abstraction

OpenFlow Basics

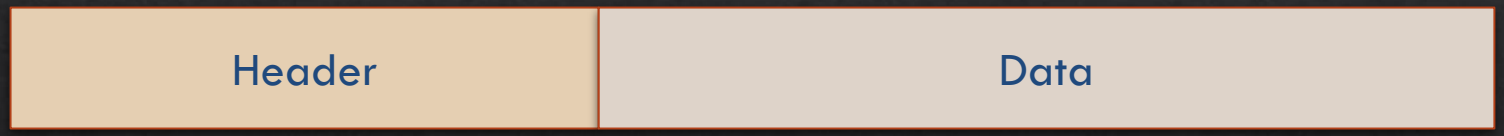


OpenFlow Basics



Plumbing Primitives

- Primitive is *<Match, Action>*
- *Match* arbitrary bits in headers:



Match: 1000x01xx0101001x

  - Match on any header, or new header
  - Allows any flow granularity
- *Action*
  - Forward to port(s), drop, send to controller
  - Overwrite header with mask, push or pop
  - Forward at specific bit-rate

General Forwarding Abstraction

Small set of primitives
"Forwarding instruction set"

Protocol independent
Backward compatible

Switches, routers, WiFi APs,
basestations, TDM/WDM

Example

Open Shortest Path First (OSPF) is a routing protocol

Intermediate System-to-Intermediate System (IS-IS) routing protocol is an Interior Gateway Protocol (IGP) and commonly used in large Service Provider networks

Networking

- Networking is
  - “Intellectually Weak”
  - behind other fields
  - about the mastery of complexity

Good abstractions tame complexity

- Interfaces are instances of those abstractions

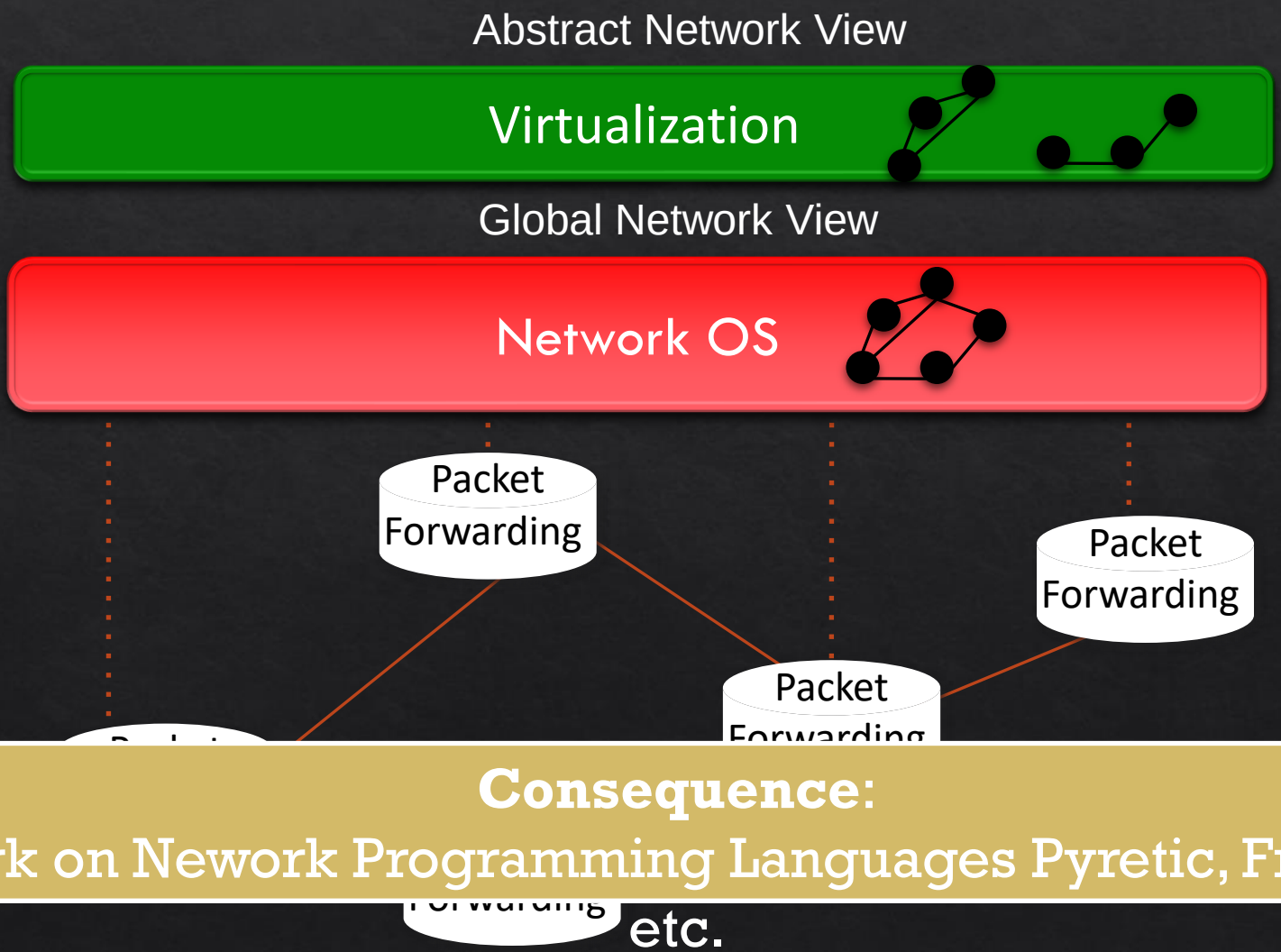
No abstraction => increasing complexity

- We are now at the complexity limit

Specification Abstraction

- Network OS eases implementation
  - E.g., Helps manage distributed state
- Next step is to ease specification
  - E.g., How do you specify what the system should do?
- Key goals
  - Provide abstract view of network map
  - Control program operates on abstract view
  - Develop means to simplify specification

Software Defined Network (SDN)



SDN in development

Domains

- Data centers
- Enterprise/campus
- Cellular backhaul
- Enterprise WiFi
- WANs

Products

- Switches, routers: About 15 vendors
- Software: vendors and startups

New startups. Lots of hiring in networking.

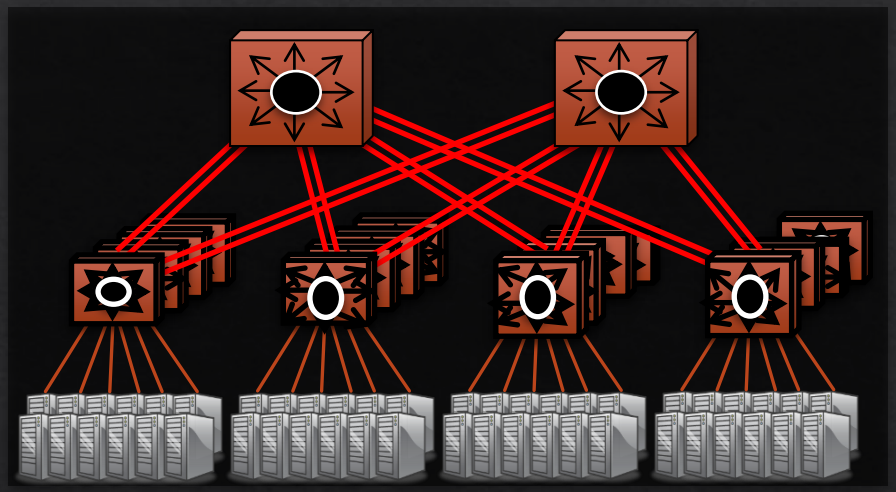
Telco Operators

- Global IP traffic growing 40-50% per year
- End-customer monthly bill remains unchanged
- Therefore, CAPEX and OPEX need to reduce 40-50% per Gb/s per year
- But in practice, reduces by ~20% per year

SDN enables industry to reduce OPEX and CAPEX

…and to create new differentiating services

Example: New Data Center



Cost

200,000 servers
Fanout of 20 ➔ 10,000 switches
$5k vendor switch = $50M
$1k commodity switch = $10M

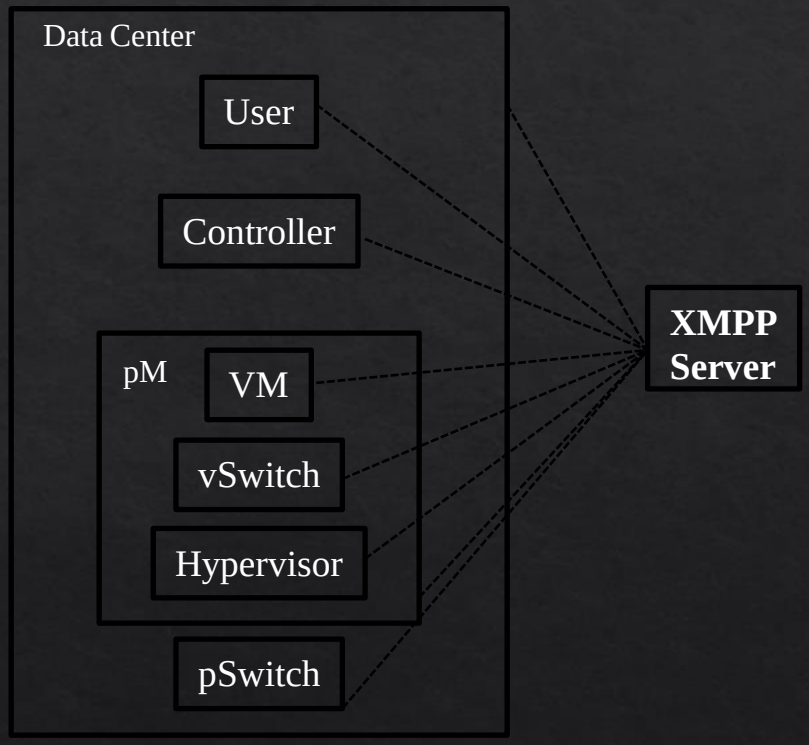
Savings in 10 data centers = $400M

Control

More flexible control
Tailor network for services
Quickly improve and innovate

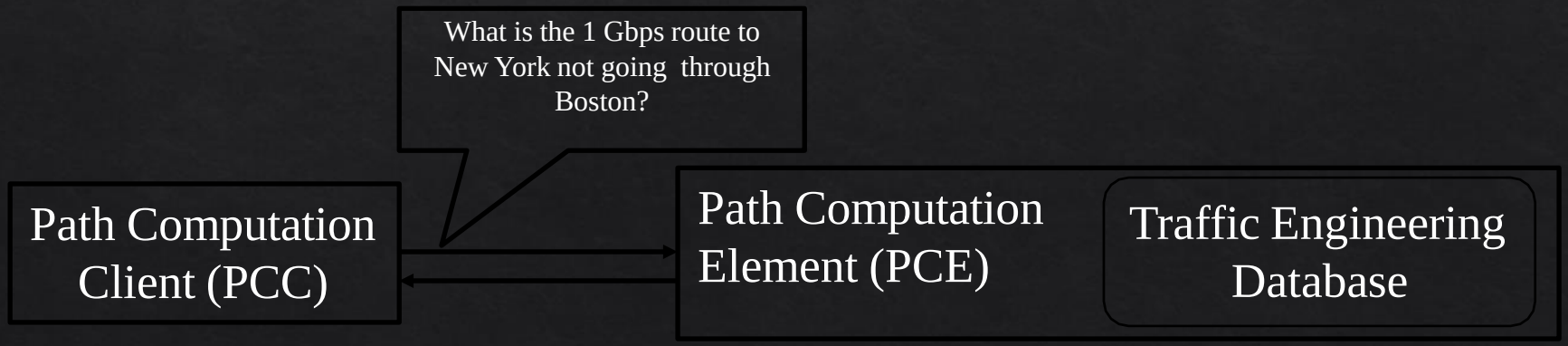
XMPP in Data Centers

- Everything is an XMPP entity.
  - It has its own contact list and authorizations.



Path Computation Element (PCE)

- MPLS and GMPLS require originating routers
  - to find paths that satisfy multiple constraints including not using any backup routers and having a given bandwidth etc.
- This may require more computer power or network knowledge than a router may have.
- IETF PCE working group has developed a set of protocols that allow a Path computation client (PCC), i.e., router to get the path from path computation element (PCE)
- PCE may be centralized or may be distributed in many or every router.

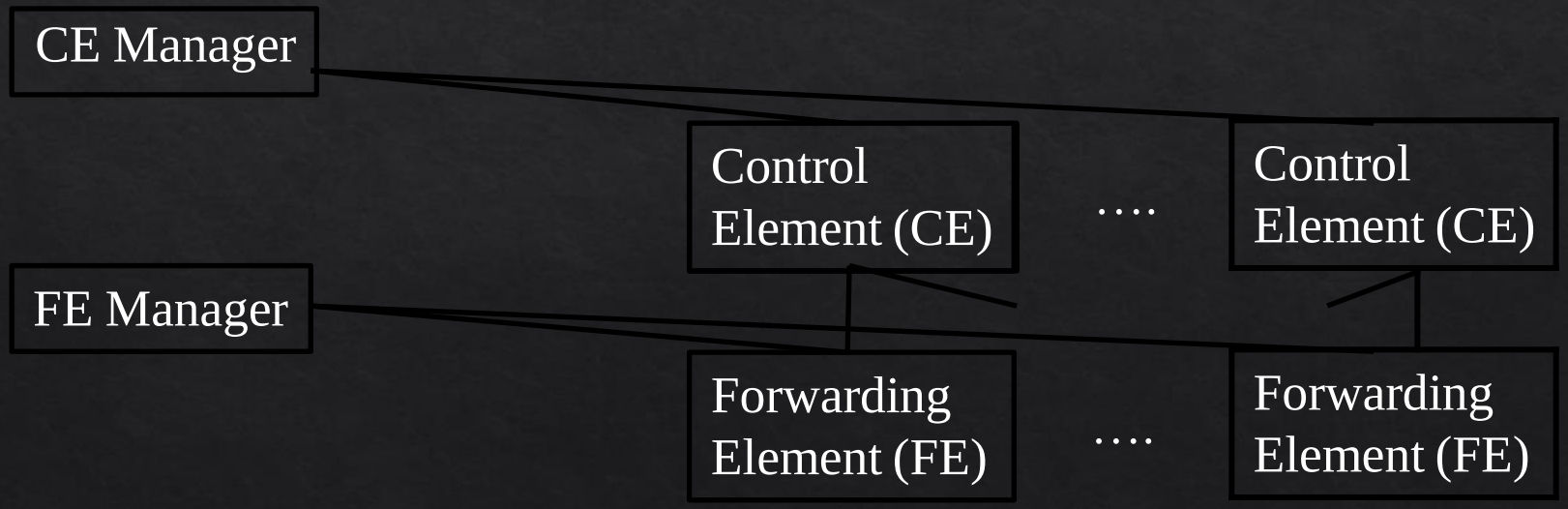


PCE (Cont)

- PCE separates the route computation function from the forwarding function.
- Both functions may be resident in the same box or different boxes.
- 25+ RFCs documenting protocols for:
  - PCE-to-PCC communication
  - PCE-to-PCE communication (Multiple PCEs)
  - PCE discovery

Forwarding and Control Element Separation (ForCES)

- IETF working group since July 2001
- Control Elements (CEs) prepare the routing table for use by forwarding elements (FEs).
- Each CE may interact with one or more FEs
- There may be many CEs and FEs managed by a CE manager and a FE manager



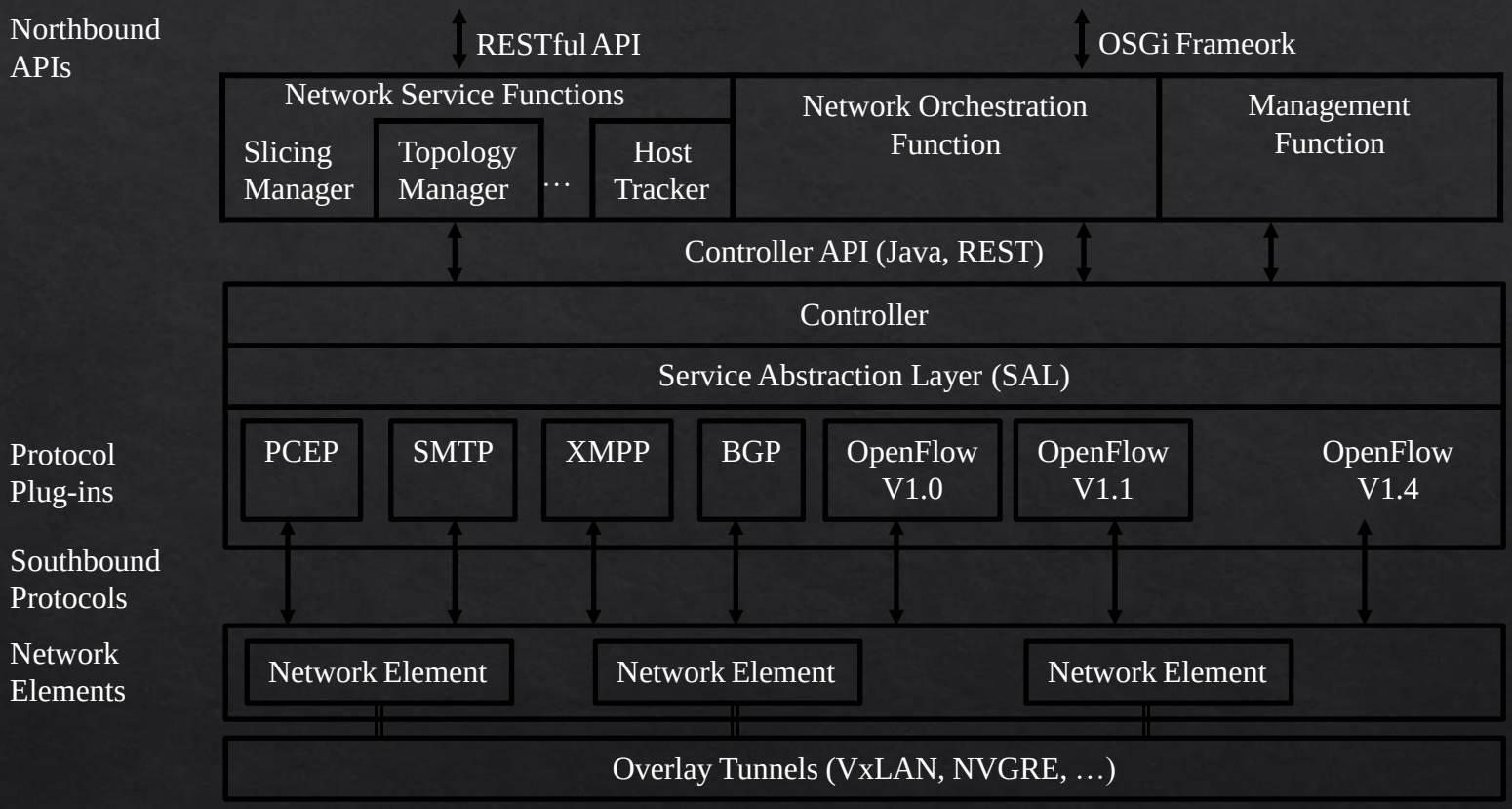
ForCES (Cont)

- Idea of control and data plane separation was used in BSD 4.4 *routing sockets* in early 1990s. It allowed routing tables to be controlled by a simple command line or by a route daemon.
- ForCES protocol supports exchange of:
  - Port type, link speed, IP address
  - IPv4/IPv6 unicast/multicast forwarding
  - QoS including metering, policing, shaping, and queueing
  - Packet classification
  - High-touch functions, e.g., Network Address Translation (NAT), Application-level Gateways (ALG)
  - Encryptions to be applied to packets
  - Measurement and reporting of per-flow traffic information

Current SDN Debate: What vs. How?

- SDN is easy if control plane is centralized but not necessary. Distributed solutions may be required for legacy equipment and for fail-safe operation.
- Complete removal of control plane may be harmful.
  - Exact division of control plane between centralized controller and distributed forwarders is yet to be worked out
- SDN is easy with a standard southbound protocol like OpenFlow but one protocol may not work in all cases
  - Diversity of protocols is a fact of life.
  - There are no standard operating systems, processors, routers, or Ethernet switches.

SDN Controller Functions



RESTful APIs

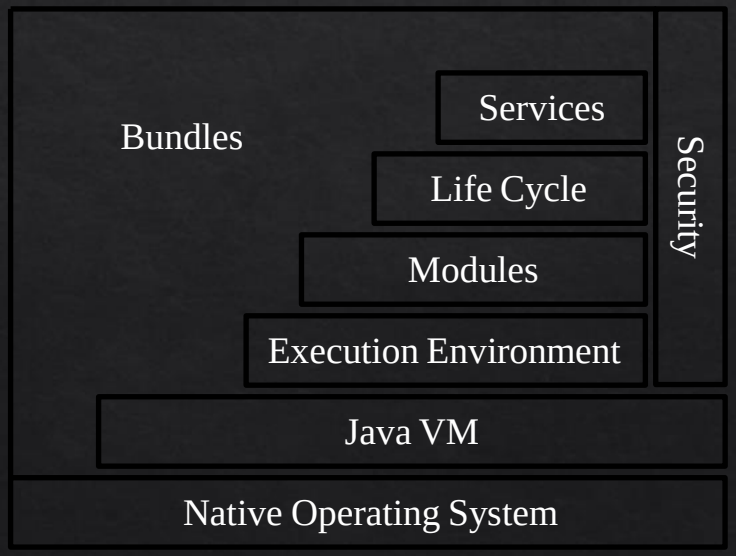
- Software architecture style developed by W3C.
- WWW uses this sytle. Very popular in other applications.
- Goals: Scalability, Generality, Independence, and allow intermediate components
- Client-Server Model: Clients and servers can be developed undependably.
- Server is stateless
- Responses can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

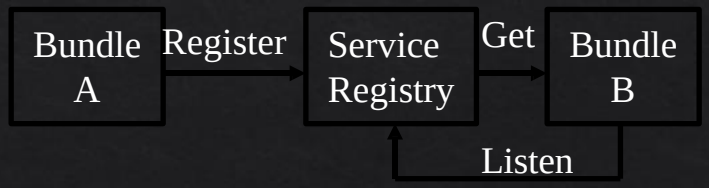
REST (Cont)

- 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





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
  - $\Rightarrow$ Bundles can be installed and uninstalled on the fly.

OpenDaylight SDN Controller Platform (OSCP)

- Multi-company collaboration under Linux foundation
- Many projects including OpenDaylight Controller
- **NO-OpenFlow** (Not Only OpenFlow): 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

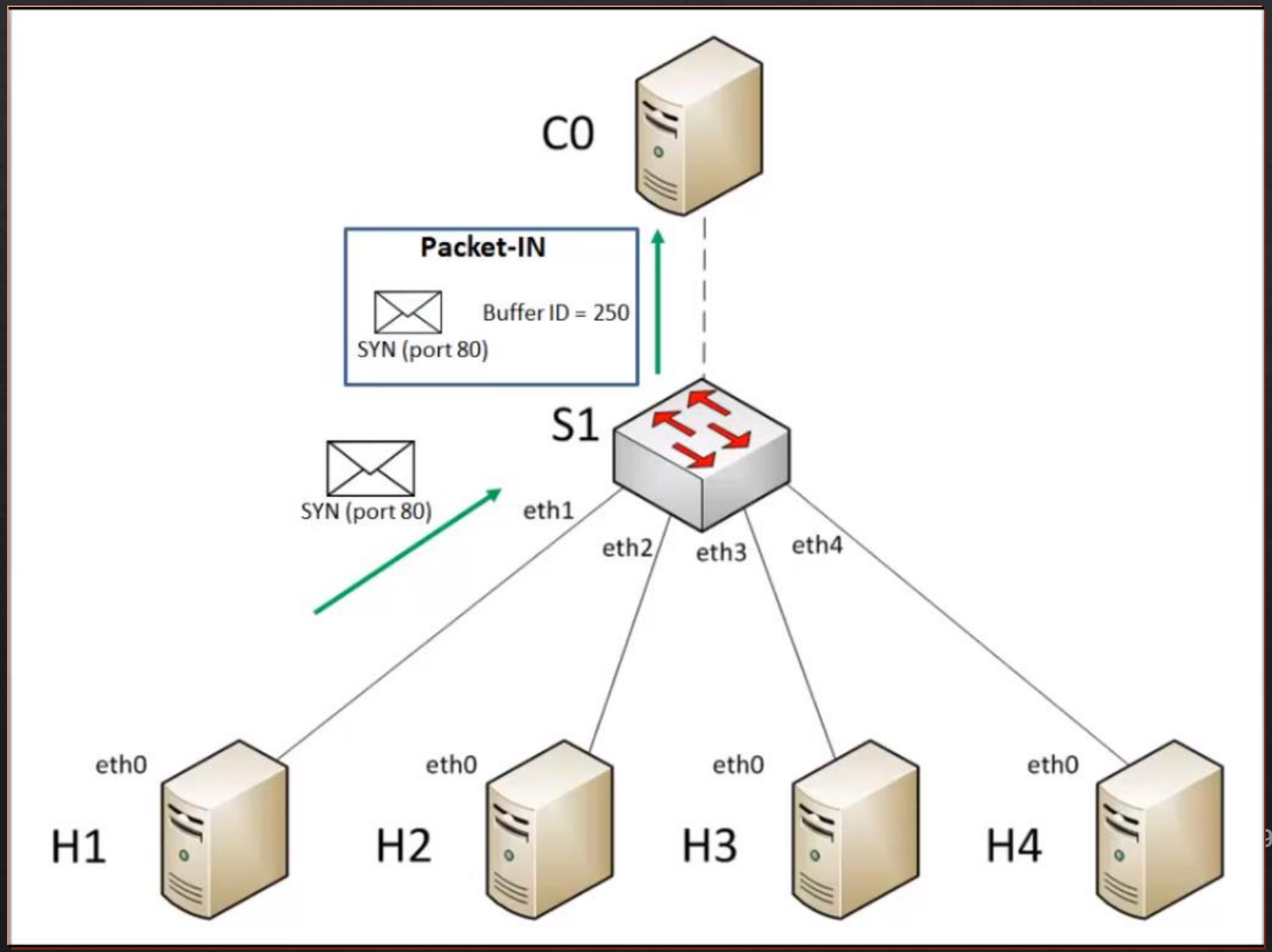
OpenDaylight Tools

1. **Applications**: Provides Virtual Network Segments (VNS) for each tenant
    1. OpenDaylight Network Virtualization (ONV):
    2. OpenDaylight Virtual Tenant Network (VTN)
2. **Services**:
    1. Defense4All: Security
3. **Northbound APIs**:
    1. **REST**
    2. **Dlux**: Northbound API using AngularJS, an extension of HTML by Google for dynamic views

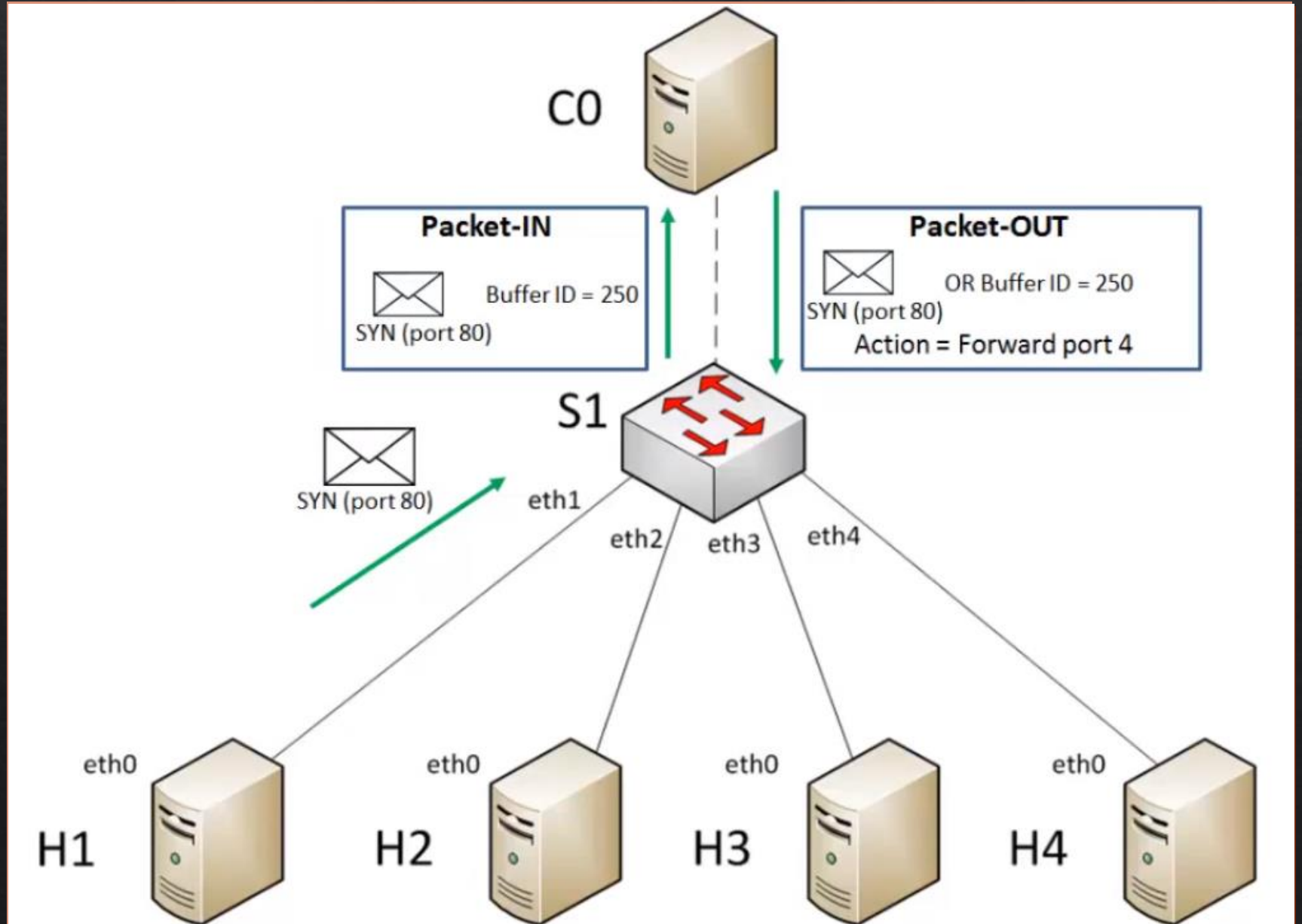
OpenDaylight Tools (Cont)

4. **Southbound APIs**:
    1. OpenFlow Plug-in + Protocol Library (V1.0, V1.1,…)
    2. Locator ID Separation Protocol (LISP) Mapping Service
    3. SNMP4SDN
    4. BGP Link State Path Control Element Protocol
5. **Overlay**:
    1. Open Distributed Overlay Virtual Ethernet (DOVE): Like VxLAN but does not use IP Multicast
6. **Configuration**:
    1. OpenDaylight YANG Tools: NETCONF
    2. Open vSwitch Database (OVSDB) Integration
    3. Affinity Metadata Service

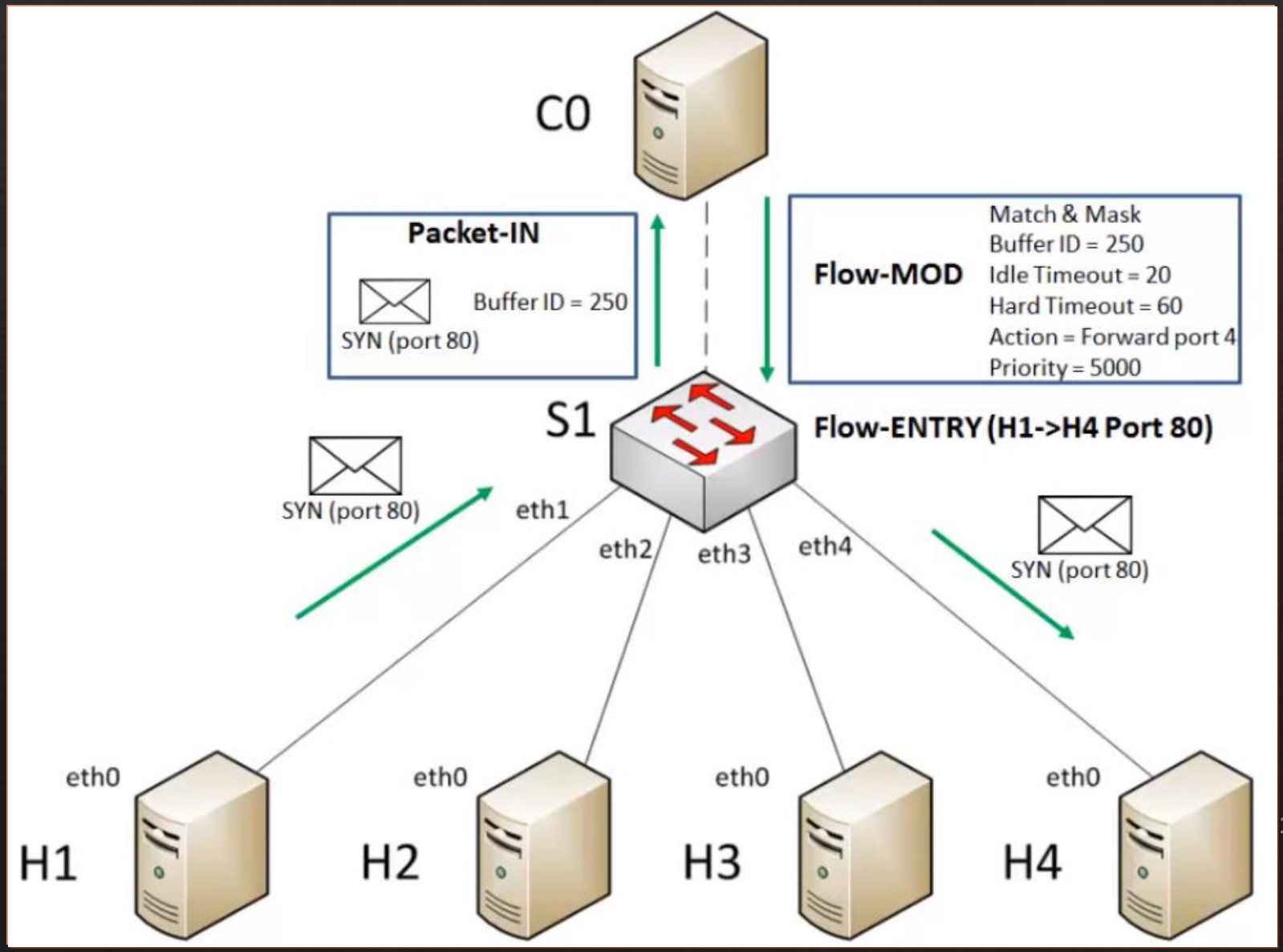
Initiation of a flow: Packet FW to SDNC to identify policy rules for the openflow flow tables



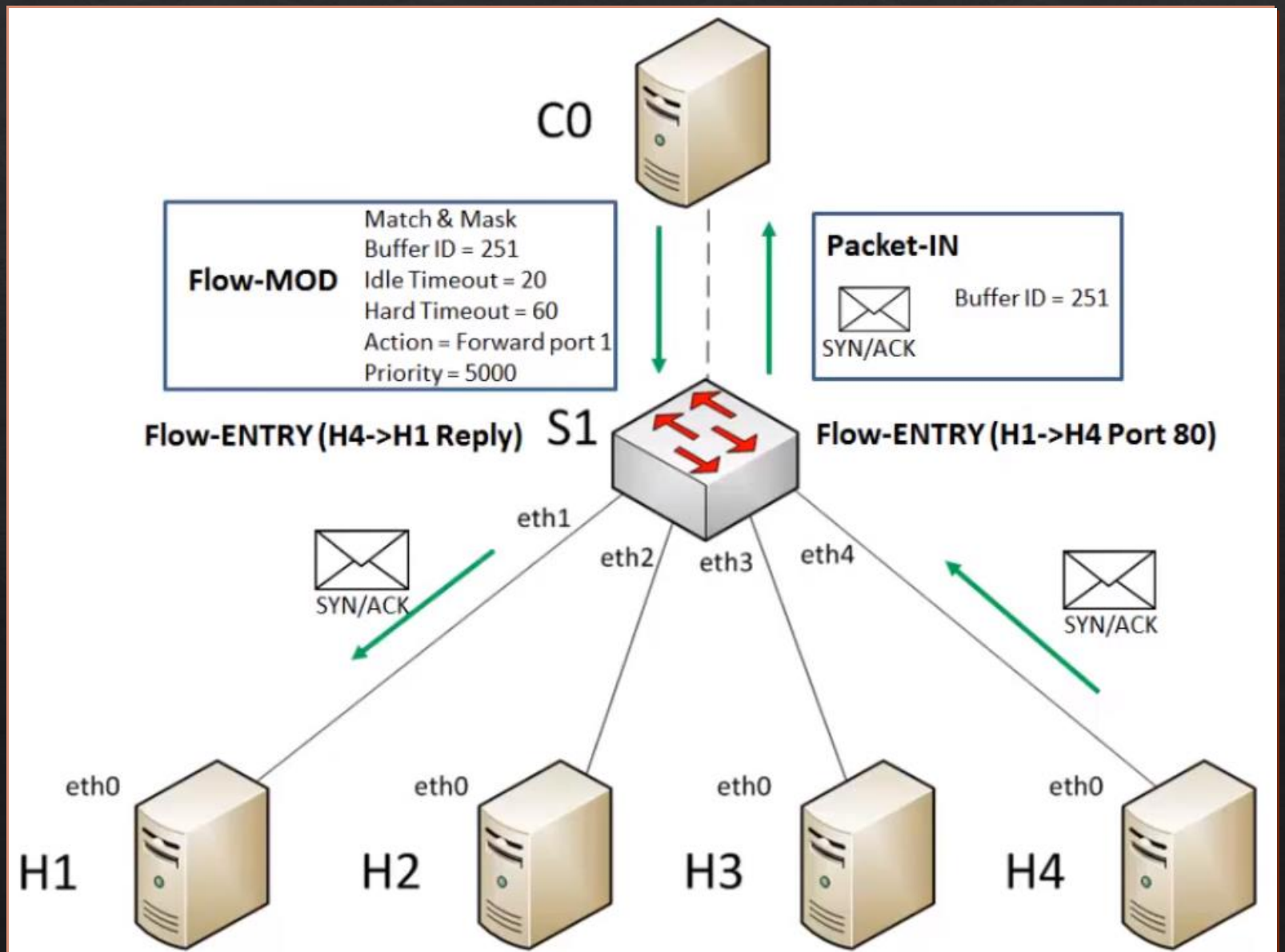
SDNC determines the ACTION for the packet/flow



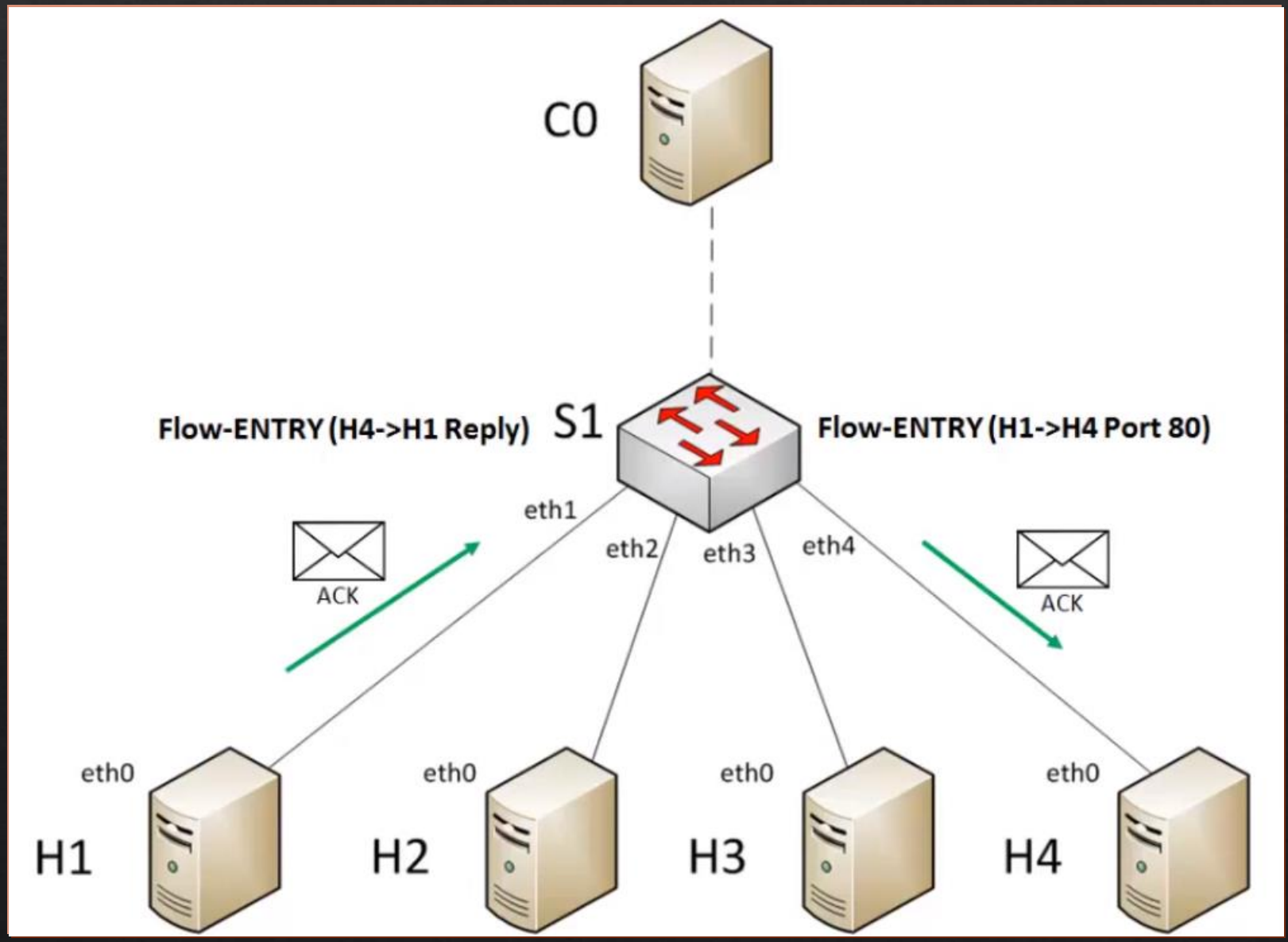
Alternatively, SDNC may provide a synthetic rule for the flow entry



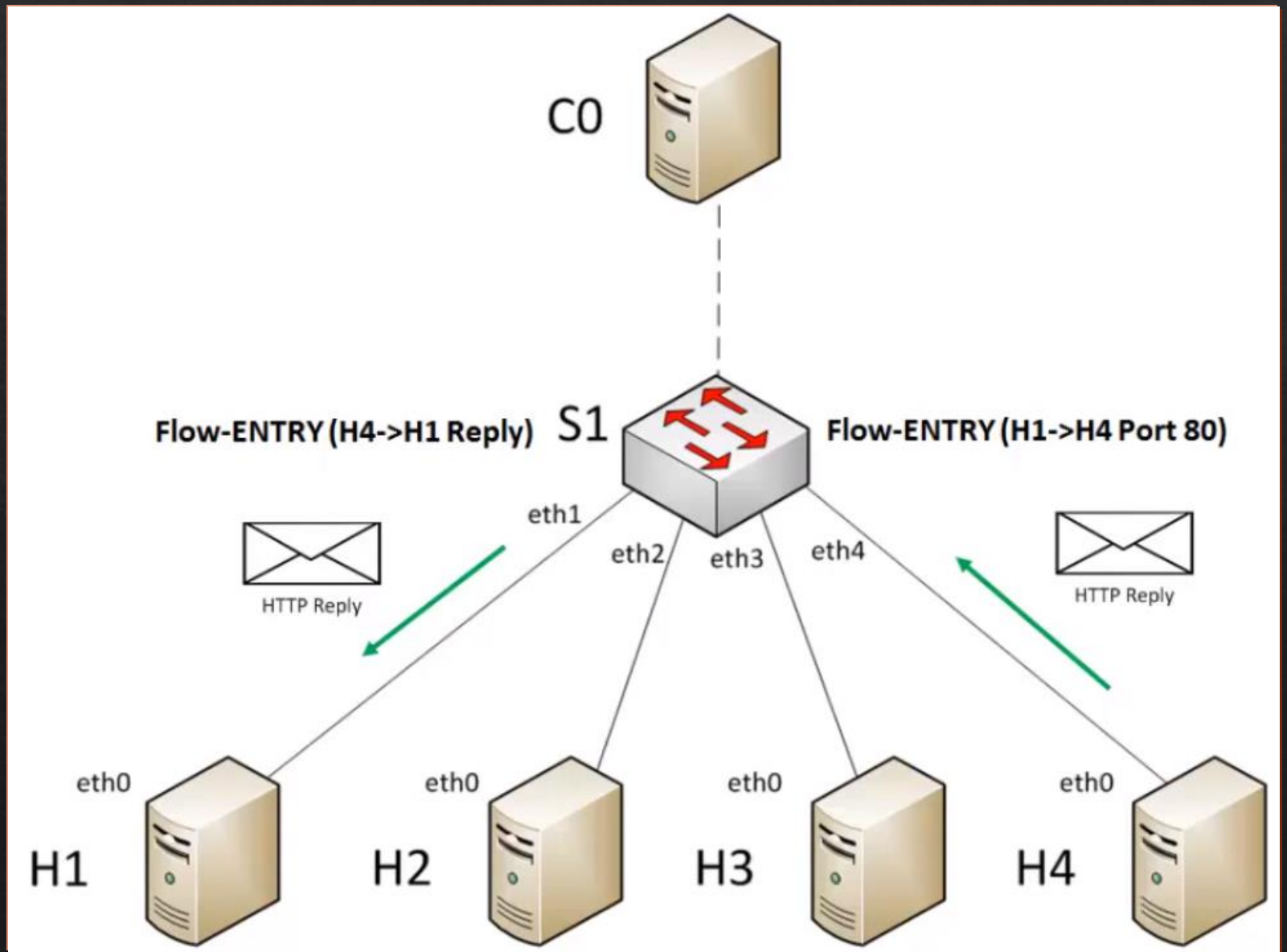
ACK packet FW to SDNC as the 1st packet of the flow from H4→H1



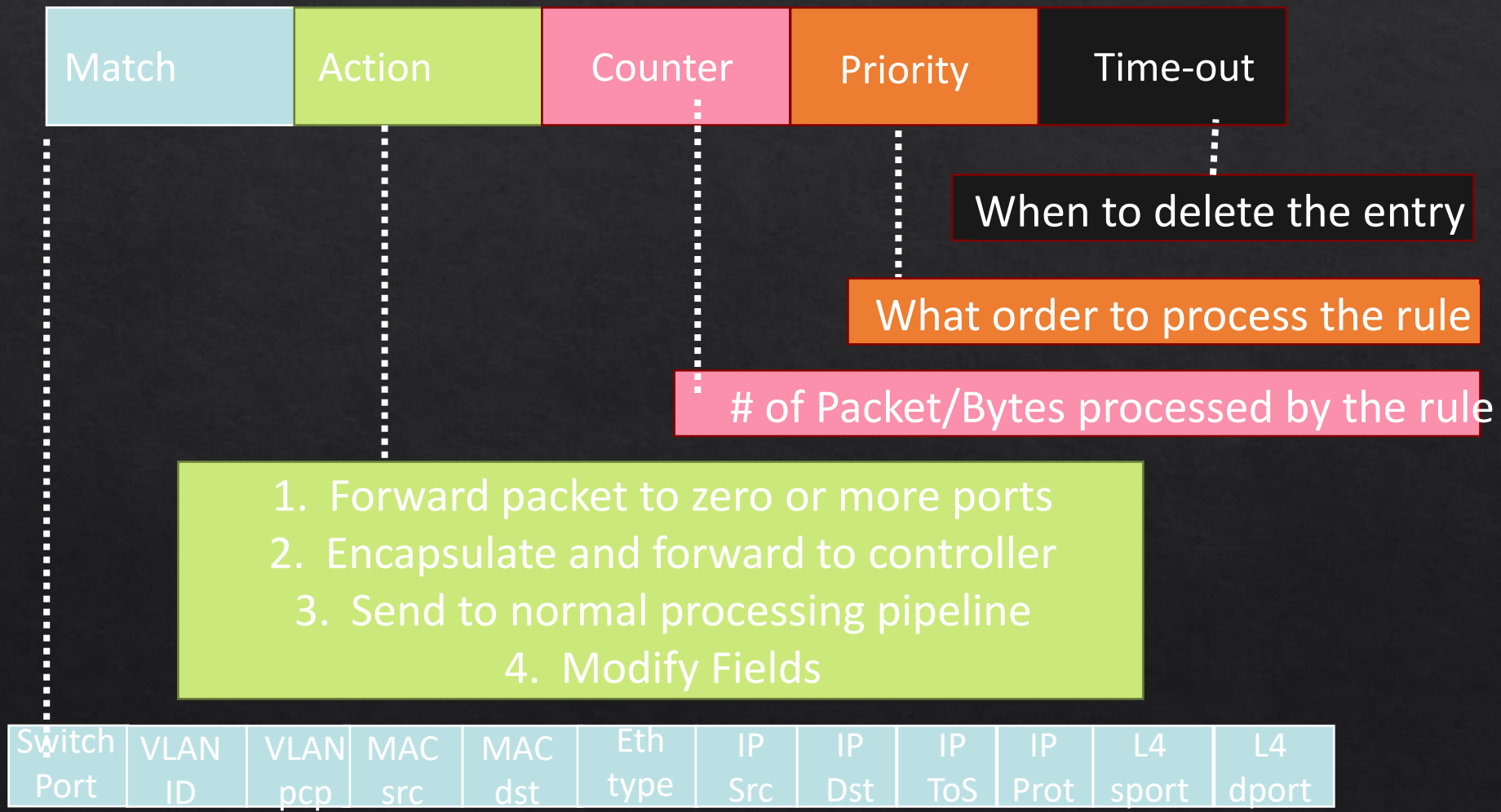
The rest of the packets flow through the switch S1 following the flow table rules set out by SDNC



The rest of the packets flow through the switch S1 following the flow table rules set out by SDNC



OpenFlow: Anatomy of a Flow Table Entry



Examples

Switching

Switch Port	MAC src	MAC dst	Eth type	VLAN ID	IP Src	IP Dst	IP Prot	TCP sport	TCP dport	Action
*	*	00:1f:..	*	*	*	*	*	*	*	port6

Flow Switching

Switch Port	MAC src	MAC dst	Eth type	VLAN ID	IP Src	IP Dst	IP Prot	TCP sport	TCP dport	Action
port3	00:20..	00:1f..	0800	vlan1	1.2.3.4	5.6.7.8	4	17264	80	port6

Firewall

Switch Port	MAC src	MAC dst	Eth type	VLAN ID	IP Src	IP Dst	IP Prot	TCP sport	TCP dport	Action
*	*	*	*	*	*	*	*	*	22	drop

OpenFlow: Types of Messages

- Asynchronous (Controller-to-Switch)
  - Send-packet: to send packet out of a specific port on a switch
  - Flow-mod: to add/delete/modify flows in the flow table
- Asynchronous (initiated by the Controller)
  - Read-state: to collect statistics about flow table, ports and individual flows
  - Features: sent by controller when a switch connects to find out the features supported by a switch
  - Configuration: to set and query configuration parameters in the switch
- Asynchronous (initiated by the switch)
  - Packet-in: for all packets that do not have a matching rule, this event is sent to controller
  - Flow-removed: whenever a flow rule expires, the controller is sent a flow-removed message
  - Port-status: whenever a port configuration or state changes, a message is sent to controller
  - Error: error messages
- Symmetric (can be sent in either direction without solicitation)
  - Hello: at connection startup
  - Echo: to indicate latency, bandwidth or liveliness of a controller-switch connection
  - Vendor: for extensions (that can be included in later OpenFlow versions)

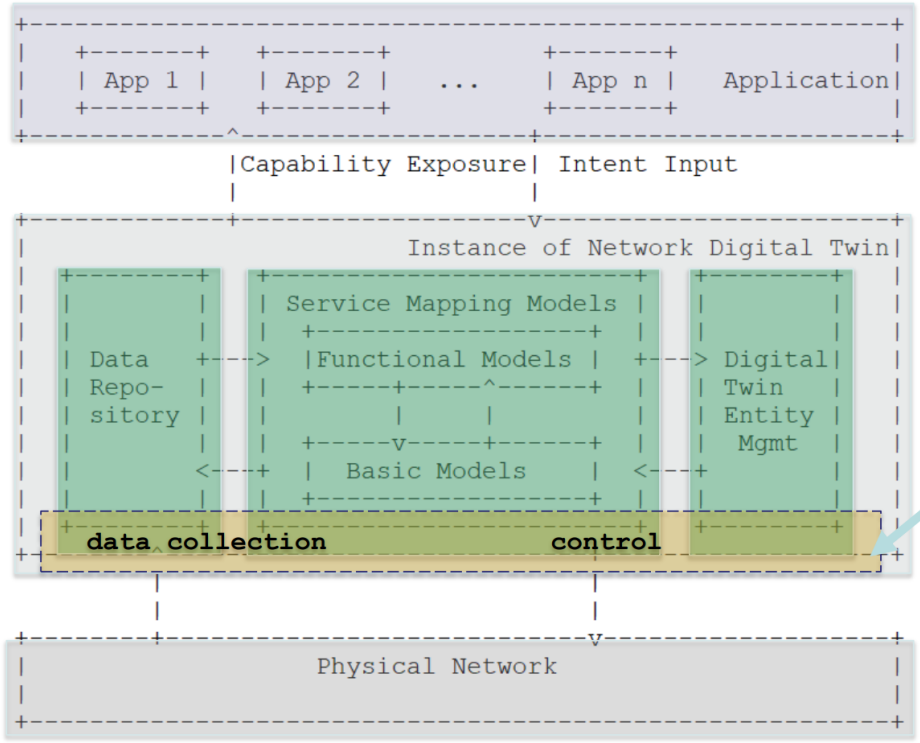
Digital twins

- Tower management/field service management: Various data including proximity, image, touch, temperature, motion and position, can be collected from telecom sites using sensor networks. This data can be fed into a digital twin of the tower to give operations and field service management key information before they go on site. When on site, experts can also assist field workers from the command center by observing the digital twin.

- Network planning and design: Maintaining an accurate inventory of network elements and keeping track of changing configuration has always been a challenge for operators. CSPs use a variety of tools in network modelling, planning, simulation, deployment and operation support activities. A digital twin could bring together all these tool capabilities to provide accurate network inventory and user data from live operations.

- Programmable network DevOps: Every new technology wave (e.g. SDN/NFV, 5G) requires testing the interworking of multiple vendor devices and solutions. A digital twin of the network and associated services together with all functionalities and behaviors could become the DevOps sandbox, where new services are simulated, tested and adjusted before being deployed on the real network.

Refine the Reference Management Architecture



Three-layer DTN reference architecture

- **The Lowest Layer:** Physical Network
- **Top Layer:** Network Application
- **The Intermediate Layer:** Network Digital Twin
  - Core layer of DTN system
  - 3 key subsystems: Data repository, Service mapping models, and Digital Twin entity mgmt
- **Optional sub-layer**

'Data collection' and 'change control' are regarded as southbound interfaces between virtual and physical networks. From an implementation perspective, they can **optionally** form a sub-layer or sub-system to provide common functionalities of data collection and change control, enabled by a specific infrastructure supporting bi-directional flows and facilitating data aggregation, action translation, pre-processing and ontologies.

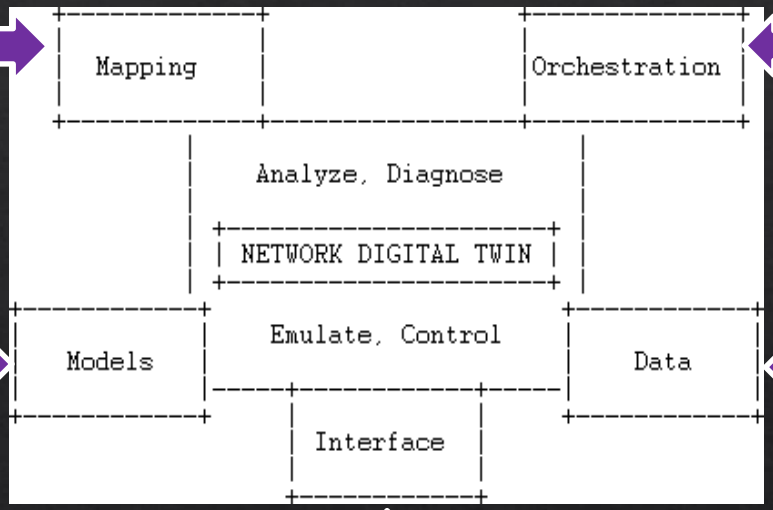
Digital twins

Key Enabling Technologies for Building DTNs

- **Data Collection**
  - Diverse existing tools (e.g., SNMP, NETCONF, Telemetry, INT, etc.) can be used to collect different type of network data
  - Innovative new tools (e.g., sketch-based measurement) can be explored
  - Semantic aggregation mechanisms for data integration and action translation
- **Data storage and services**
  - Unified data repository to effectively store large-scale and heterogeneous network data
  - To provide data services including fast search, batch-data handling, conflict avoidance, data access interfaces, etc.
- **Data Modeling**
  - For small scale network, network simulating tools (e.g., NS-2, GNS3) can be an option
  - For large scale network, low-cost solution is required to create network element and topology models
  - AI/ML can be used to build complex functional models in twin entity.
- **Visualization**
  - Display the network topology, operational status in multiple dimensions and fine granularity
  - The interactive visualizing the execution of models to help users better understand, deduce and explore the network Interfaces and protocols
- **Interfaces**
  - **Twin interfaces** between the physical network and its twin entity: existing interfaces (SNMP, NETCONF, etc.) or new interfaces
  - **Application-facing interfaces** between the network digital twin and applications, e.g., Intent, “what-if” planning app, ...
  - **Internal interfaces** within network digital twin: Interfaces of high-speed, high-efficiency, high-concurrency, etc.

Digital Twin Network Composition

Provides real-time interactive mapping between physical network and virtual twin network or mapping between two virtual twin networks



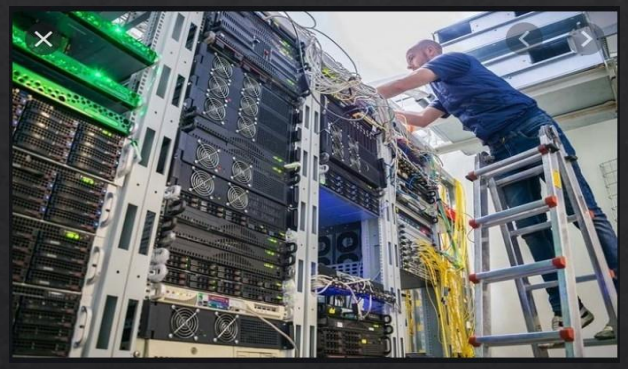
lifecycle management of components

Comprise the models based on physical network simulation, statistical data, performance metrics, inventory information, log information, and Artificial Intelligence

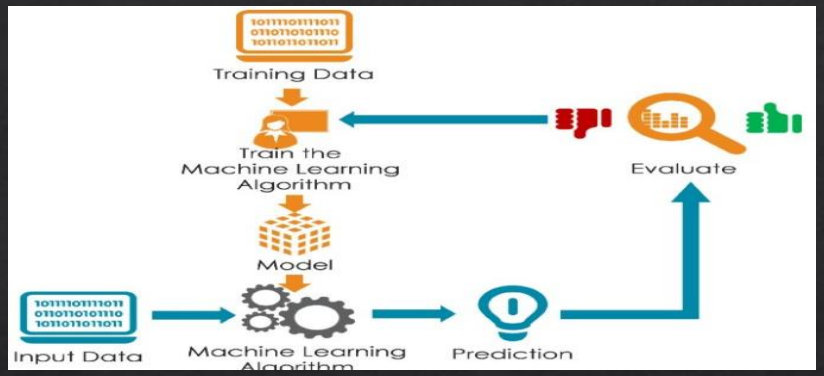
Data about its real-world twin that are required by the models to represent and understand the states and behaviors of the real-world twin

Service interfaces for network applications or other digital twins to access data and invoke capabilities. Telemetry interface for digital twin to populate and cache data

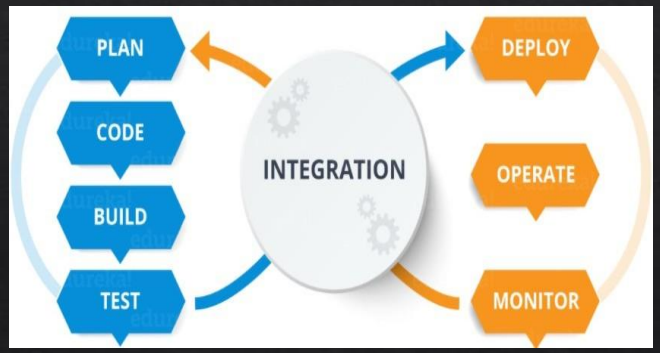
Sample Application Scenarios



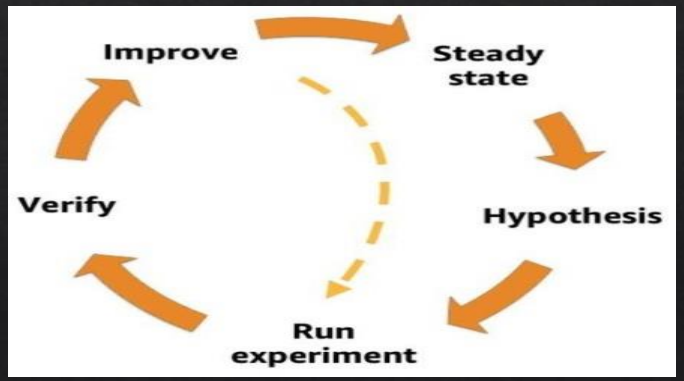
Network Maintenance Engineer Training



Machine Learning Training



DevOps oriented Certification



Network Fuzzing

Issue1: Why distinguish data from model?

- Data and models are usually the two common and separated components/elements in other industrial digital twin entities (e.g., manufacturing, factory-floor).
  - Of course, data can be structured to follow a set of well-known data models.
- Data and Model in digital twin networks
  - *Data*: a digital twin should contain data about its real-world twin (i.e., physical network) that are required by the models to represent and understand the states and behaviors of the real-world twin.
  - *Models*: A digital twin should contain computational or analytic models that are required to describe, understand, and predict the twins' operational states and behaviors, and models that are used to prescribe actions based on service logic and objectives about the corresponding real-world object.
  - In brief, data is cornerstone for constructing a DTN system; and various models are the power and source to Analyze, Diagnose, **Emulate** and Control the physical network.

Issue 2: How Orchestration is different from other components?

Basically orchestration component aims to control and manage the twin entity, then helps to provide an integrated service to various applications.

- Two main orchestration features can be provided:

1. Control the digital twin network environment and its components to derive the required/expected behavior
2. Deal with the dynamic lifecycle management of these components by providing
   - Repeatability: Replicate network conditions on demand
   - Reproducibility: Replay successions of events, possibly under controlled variations

Issue 3: How should the interfaces be defined?

- Three types of interfaces were identified:
  1) Twin interface: between the physical network and its twin entity
     - It can be implemented using a variety of existing tools (telemetry, SNMP, NETCONF, etc.) or new ones.
  2) Application-facing interface: between the network digital twin and applications that make use of the emulated network.
     - For example, Intent, “what-if” planning app, ...
  3) Internal interfaces between components within network digital twin
- We need to first define or choose the first two types of interfaces, then focus on the internal interfaces to build the twin image.
  - The first two interfaces should be open, standardized, real-time, secure, and reliable.
  - Internal interfaces should be with capability of high-speed, high-efficiency, high-concurrency etc.

Issue 4: Which component is responsible for checking for deviation of the underlay network vs. the image?

- Mapping component is responsible for such checking
- From traditional simulation to emulation, with real-time interactive mapping.
  - Digital twin network provides **real-time interactive mapping** between physical network and virtual twin network, that **emulates** the behavior of a network by calculating the deviation between the different network entities (routers, switches, nodes, access points, links, etc.) in the physical network and corresponding entities in the virtual twin network.

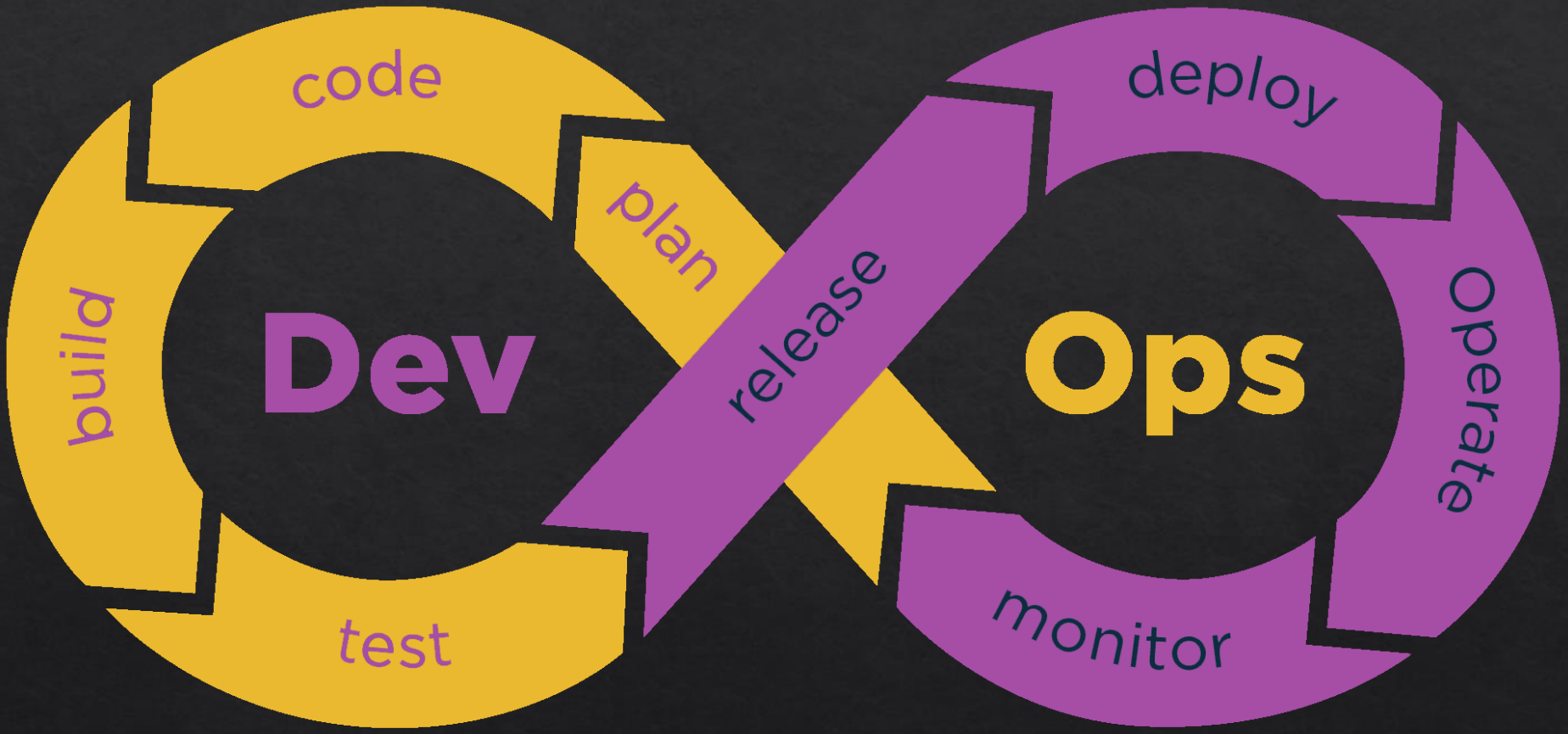
- Mapping can be:
  - One to one mapping (pairing, vertical): Synchronize between a physical network and its virtual twin network **with continuous flow**
  - One to many mapping (coupling, horizontal): Synchronize among virtual twin networks with **occasional data exchange**

Issue 5: Continuous Verification vs CI/CD

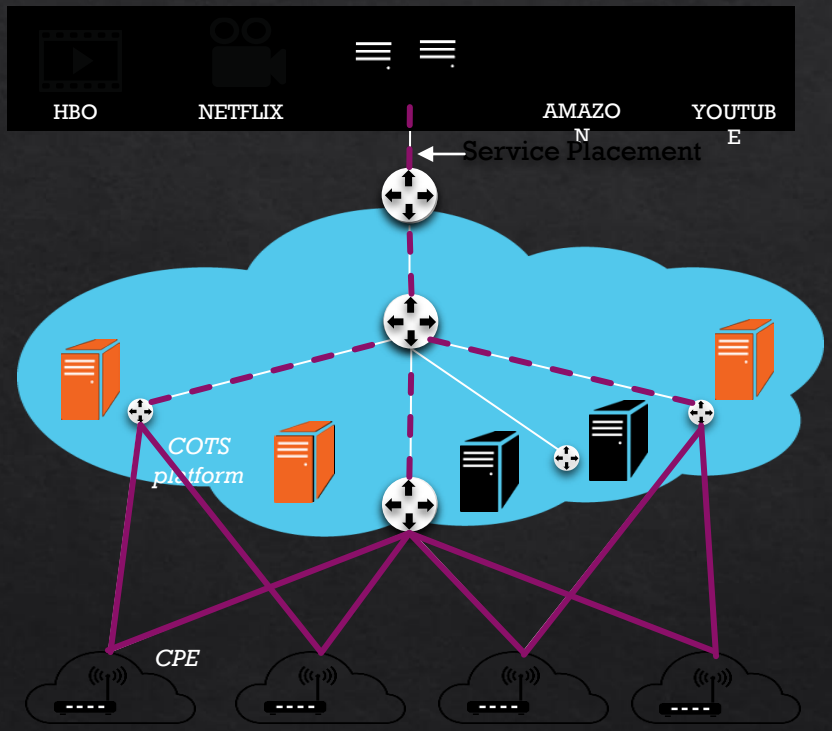
- Modern DevOps practices involve continuous development/testing/integration, /deployment/monitoring of software applications throughout its development life cycle:
  - ***Continuous Integration (CI)*** allow implement small changes and check in code to version control repositories frequently.
    - E.g., committing all your application code in a single repository
  - ***Continuous Delivery (CD)*** automates the delivery of applications to selected infrastructure environments. (e.g., network digital twin)
    - E.g., Travis CI allows automatically run CI tasks like unit tests and push your code to a hosting platform every time you push new changes to a branch.

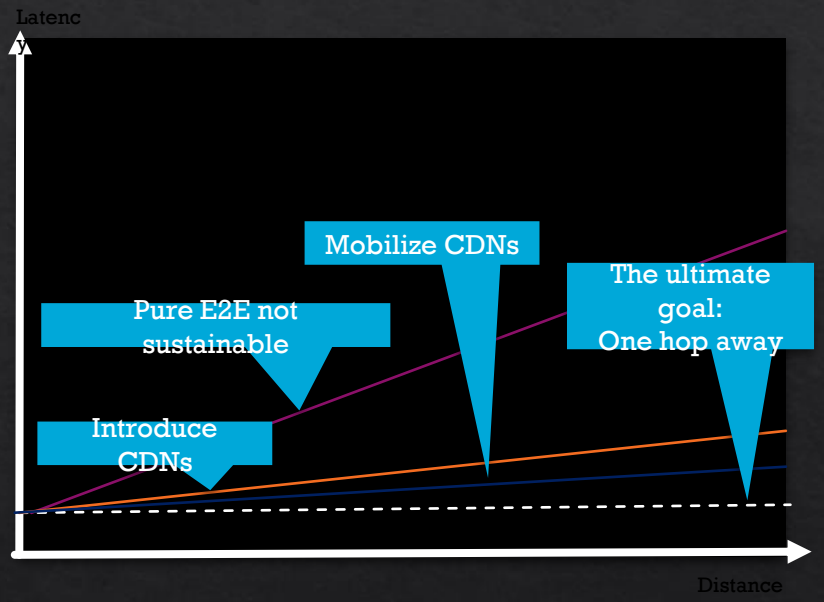
- Continuous Verification (CV) is an extension of DevOps practices that are concerned with verifying the system as a whole.
  - The application of CI/CD models in network management operations increases the risk associated to deployment of non-validated updates
  - CV can be used in **DevOps-oriented certification application** to address it.

Dev Ops


code
plan
build
Dev
release
Ops
deploy
operate
test
monitor

Meeting 5G KPIs
Your Service Just One Hop Away





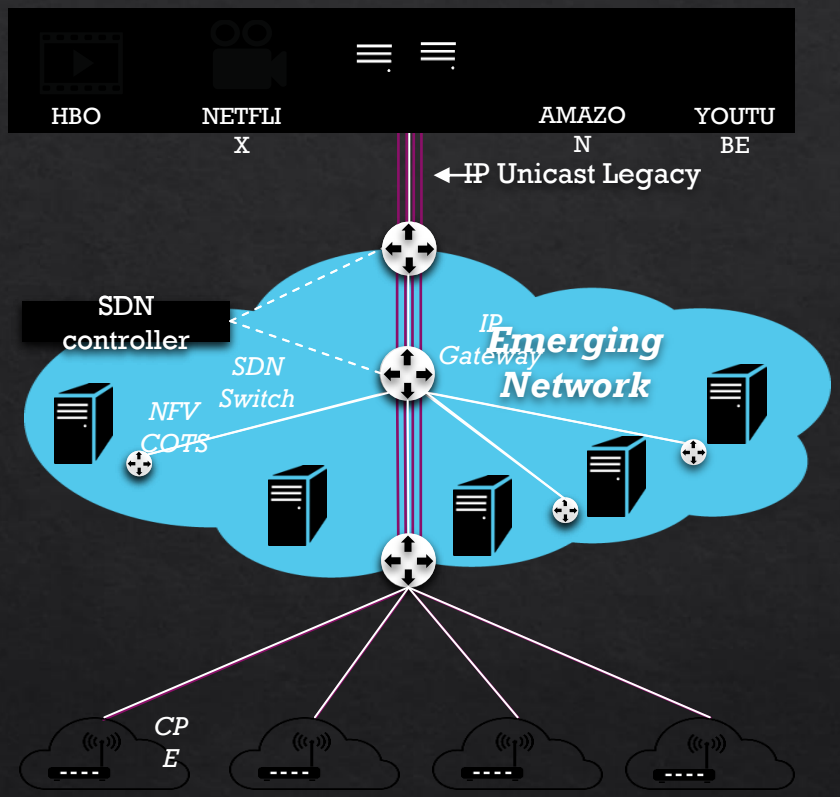
5G + SDN/NFV

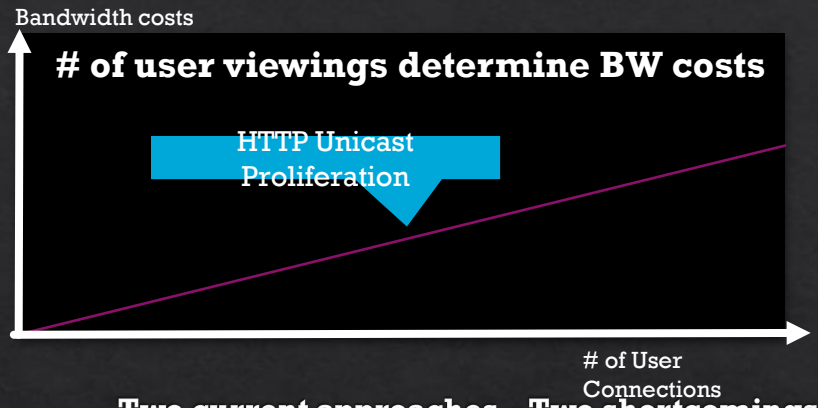
Reinventing the approach to IP based services through a backward compatible introduction of new methodologies supported by an SDN/NFV enabled network fabric & designed to meet challenging 5G KPIs

It looks like IP, it smells like IP, BUT with this technology inside networks will simply work better…

The target for this tech: Telcos & Switch Vendors

The Problem & Current Approaches



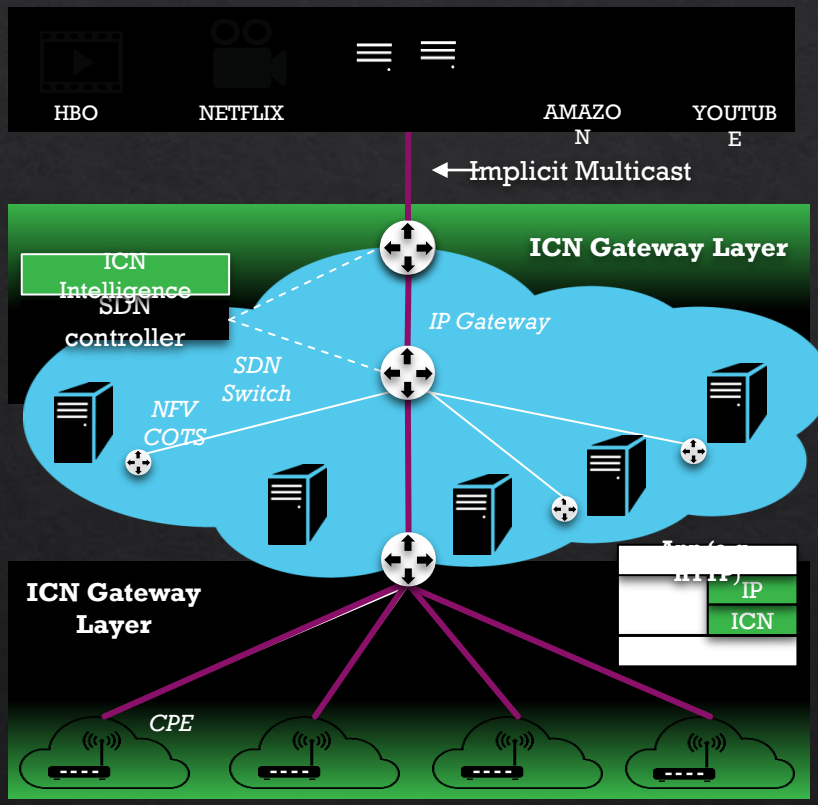


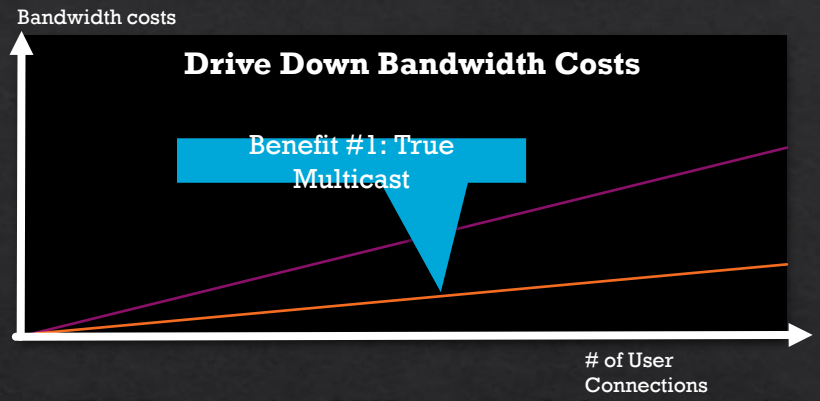
Two current approaches – Two shortcomings

- CDNs are currently used for popular content but this is overly complex and results in inefficiencies associated with indirections
- Overprovisioning of resources drives unsustainable spiraling costs

Both shortcomings are unsustainable for 5G

Re-Introducing Multicast



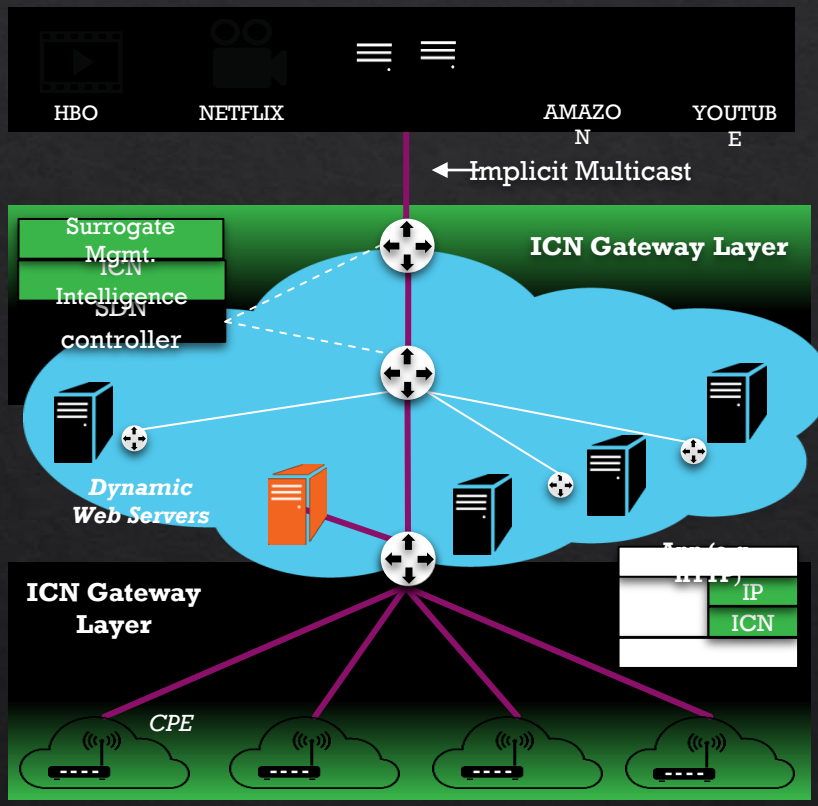


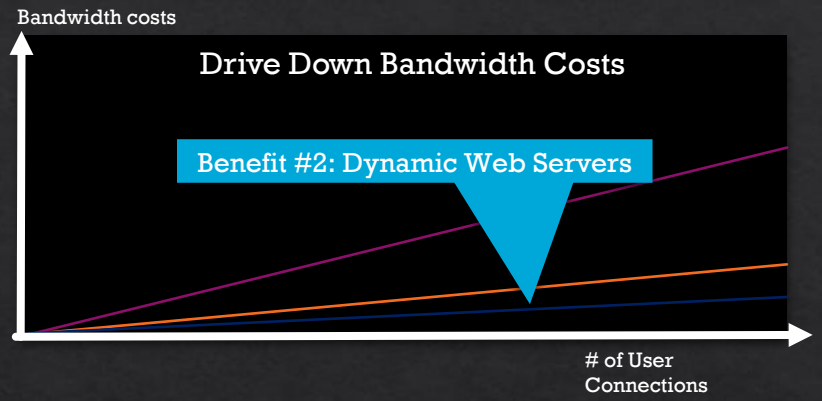
The Innovative ICN approach for competitive 5G (or before) operator networks

- Re-introduce multicast into world of predominantly personalized web experience
  -> **higher network utilization**
- Flexible routing at runtime through *cloudifiable* software elements
  -> **increased resilience, latency reduction**

Information-centric networking

Localize Communication





The next logical step: Dynamic Web Servers, spun up possibly just one hop away

- Creates new service possibilities for operators, utilizing in-network NFV-based computing capabilities
- Helps meeting challenging 5G KPIs, such as 5ms service-level latency & 1000x capacity increase

Attack Detection

- Trends and automation allow you to know when you are under attack.
- The tools in use can help you to mitigate attacks:
  - Flows across network interfaces
  - Load on specific servers and/or services
  - Multiple service failures

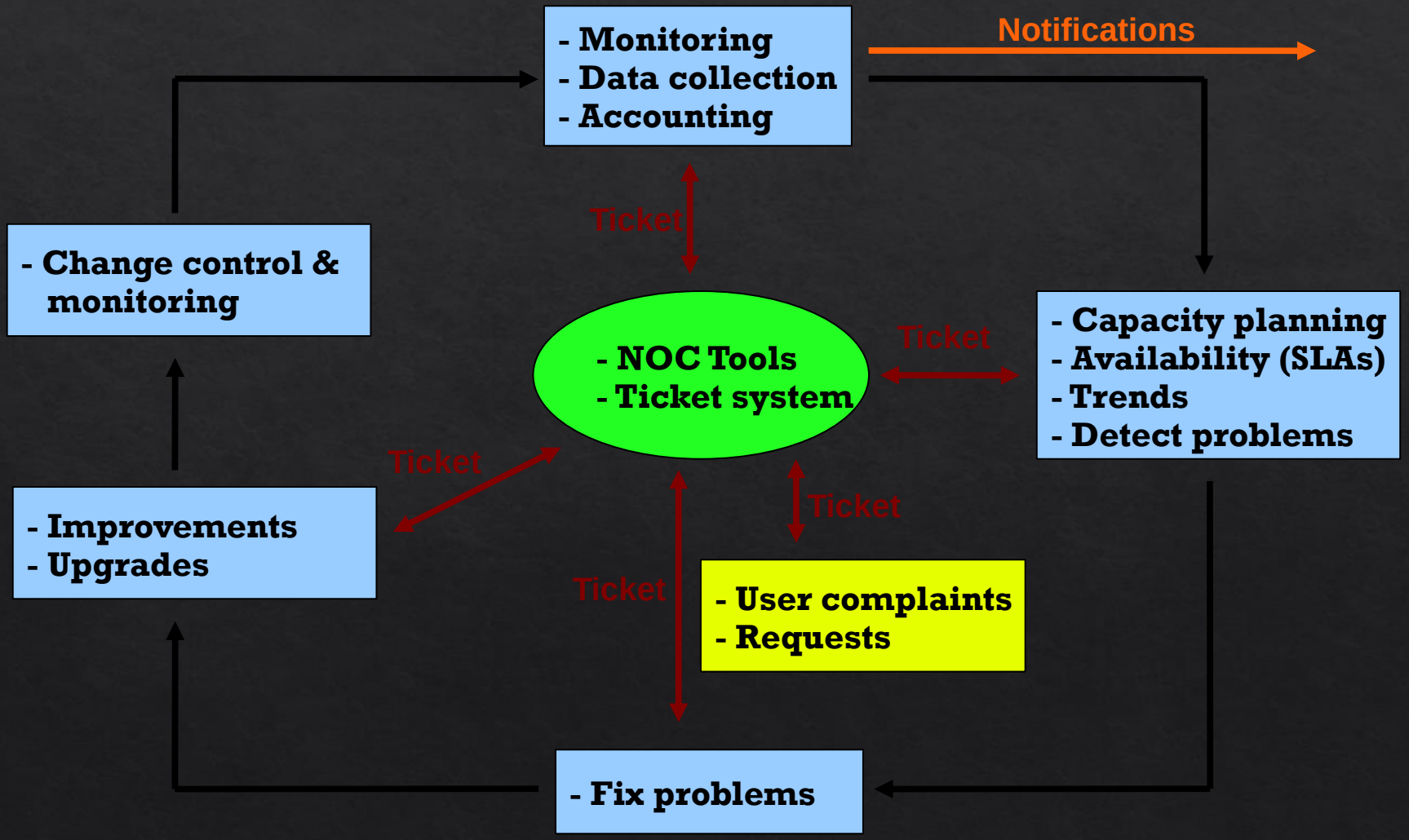
Consolidating the data

The Network Operations Center (NOC)

“Where it all happens”

- Coordination of tasks
- Status of network and services
- Fielding of network-related incidents and complaints
- Where the tools reside (”NOC server”)
- Documentation including:
  - Network diagrams
  - database/flat file of each port on each switch
  - Network description
  - Much more as you'll see a bit later.

The big picture



The notion of Internet management

- The design of Future Internet network elements with the aim of mastering the increasing complexity of communication networks
- The network should be capable of real-time, secure and cost-effective delivery of data. It is of utmost importance to increase the user's perceived quality of life anywhere and anytime.
  - human-to-human
  - human-to-machine
  - machine-to-machine

Network Management
Basics

- Network management requirements
- OSI Management Functional Areas
  - Network monitoring: performance, fault, accounting
  - Network control: configuration, security
- Standardization in network management

- Practical issue: introduction to SNMP

Network Management

Requirements

Example of approach

- Controlling strategic assets
- Controlling complexity
- Improving service
- Balancing various needs: performance, availability, security, cost
- Reducing downtime
- Controlling costs

What are we talking about? FCAPS model

- Network Management Tasks
  - fault management
  - configuration management
  - accounting management
  - performance management
  - security management
  - inventory management

Network Management
OSI functional areas

- Fault management
  - Detect the fault
  - Determine exactly where the fault is
  - Isolate the rest of the network from the failure so that it can continue to function
  - Reconfigure or modify the network in such a way as to minimize the impact
  - Repair or replace the failed components

  - Tests: connectivity, data integrity, response-time, ….

Fault Management

- detection
- exception alarm generation
- investigation and analysis
- statistics for steady state behaviour characterisation

Fault Management Sub-categories

Prioritization	• Prioritize faults in the order in which they should be addressed • Use in-band management packets to learn about important faults • Identify which fault events should cause messages to be sent to the manager • Identify which devices should be polled and at what intervals • Identify which device parameter values should be collected and how often • Prioritize which messages should be stored in the manager's database
Timeliness Required	• Management Station is passive and only receives event notifications • Management Station is active and polls for device variable values at required intervals • Application periodically requests a service from a service provider
Physical Connectivity Testing	• Using a cable tester to check that links are not broken
Software Connectivity Testing	• Using an application that makes a request of another device that requires a response. ❑ The most often application for this is Ping.Exe. It calls the Internet Control Message Protocol (ICMP) which sends periodic Echo Request messages to a selected device on a TCP/IP network ❑ Application on one device makes a request of an application on another device
Device Configuration	• Devices are configured conservatively to minimize chances of dropped packets.
SNMP Polls	• Devices are periodically polled to collect network statistics
Fault Reports Generated	• Thresholds configured and alarms generated • Text media used for report • Audio media used for report • A color graphical display used to show down devices • Human manager is notified
Traffic Monitored	• Remote Monitors used • Protocol analyzers used • Traps sent to Network Management Station • Device statistics monitored
Trends	• Graphical trends generated to identify potential faults

Fault Management notification cycle

Understanding the need for Fault Management

Fault management is usually mentioned as the first concern in network management.

Its main role is to ensure high availability of a network

Hence, involving a procedure to anticipate and avoid network failures

In the case where a failure cannot be avoided, the necessary steps are required to contain the damage and resolve the effects on the network

Defining fault management

The goal of Fault Management

The goal of *fault management*

- detect,
- log,
- notify users of, and (to the extent possible)
- automatically fix

network problems to keep the network running effectively.

Because faults can cause downtime or unacceptable network degradation, fault management is perhaps the most widely implemented of the ISO network management elements.

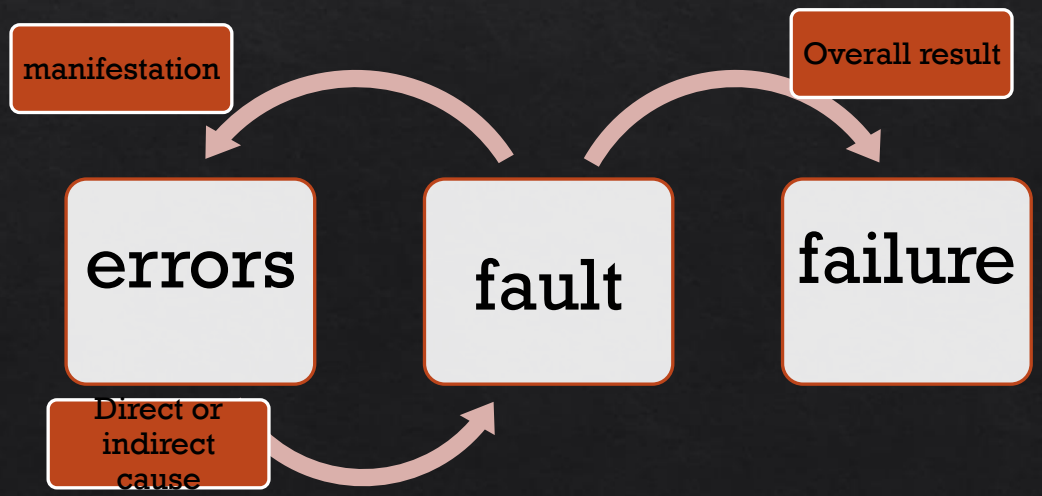
Fault Management

- Fault management involves first.

First,	• determining symptoms and isolating the problem
Second,	• Then the problem is fixed and the solution is tested on all-important subsystems
Finally,	• The detection and resolution of the problem is recorded.

FAULT MANAGEMENT – Defining the terms

- A *fault* is a software or hardware defect in a system that disrupts communication or degrades performance
- An *error* is the incorrect output of a system component. If a component presents an error, we say the component fails → This is a *component failure*



Fault symptoms can be associated to four types of error

These symptoms may take one of the following forms:

These symptoms may take one of the following forms:

- **timing error** — *An output with an expected value comes either too early or too late*
- **timely error** — *An output with an unexpected value within the specified time interval*
- **commission error** — *An output with an unexpected value outside the specified time interval → response produced*
- **omission error** — *An output with an unexpected value outside the specified time interval → no response is produced*

network faults can be generally divided according to their duration into three groups:
permanent, intermittent and transient:

1st: A permanent faul exist in the network until a repair actior en taken. This resul in permanent max gradation of the service

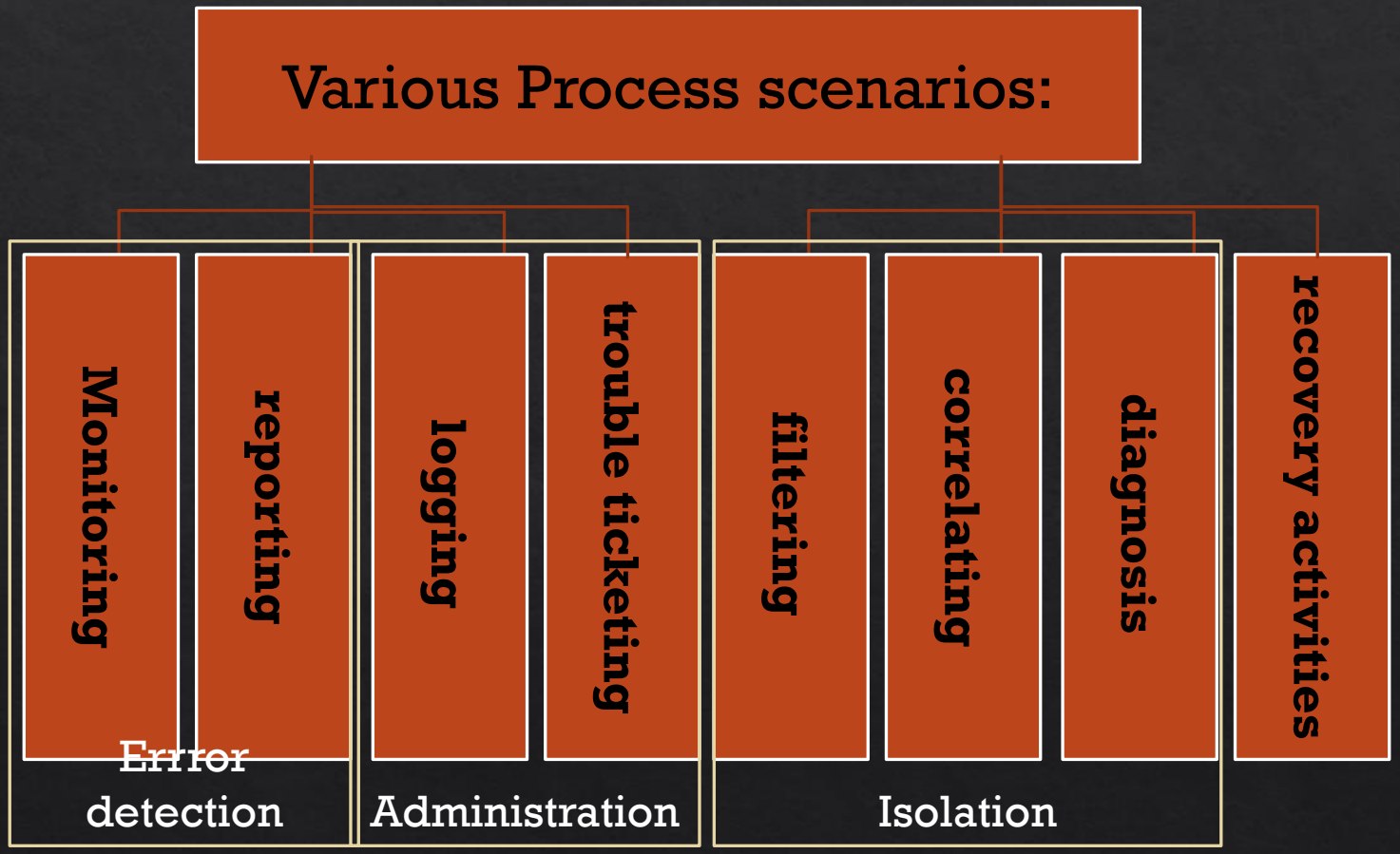
2nd :An intermittent fault occurs in a discontinuous and a periodic manner. I outcome will be failures in current proc This implies maximum degradation of t service level for a short period of time.

3rd :A transient fault will momentarily cause a minor degradation of the service.

FAULT:

- For faults of the first type, it will cause an event report to be sent out and changes made in the network configuration to prohibit further utilisation of this resource.
- For a fault of the second type, the severity of the fault may transfer from being intermittent to being permanent if an excessive occurrence of this kind of fault becomes significant.
- For a fault of the third type, it will usually be masked by the error recovery procedures of network protocols and therefore may not be observed by the users.

Fault Management Process



Typical fault

- **Observable fault**
- **Unobservable fault**
  - For example, the existence of a deadlock between co-operating distributed processes may not be observable locally.
  - Other faults may not be observable because the vendor equipment is not instrumented to record the occurrence of a fault
- **Too many related observation**:
  - A single failure can affect many active communication paths. The failure of a WAN back-bone will affect all active communication between the token-ring stations and stations on the Ethernet LANs, as well as voice communication between the PBXs.
- **Propagation of failures**
  - a failure in one layer of the communications architecture can cause degradation or failures in all the dependent higher layers.

Network Management
OSI functional classification

- Performance management:
  - What is the level of capacity (χωρητικότητα) utilization?
  - Is there excessive traffic?
  - Has throughput been reduced to unacceptable levels?
  - Are there bottlenecks?
  - Is response time increasing?

- Indicators: availability, response time, accuracy ← service
  throughput, utilization ← efficiency

Performance Management

- Availability and Reliability metrics
- Quality metrics
- real-time measurement
- historical analysis

Performance Management Sub-Categories

Collecting Baseline Utilization Data	• Measuring link utilization using a probe • Counting packets received/transmitted by a specific device • Measuring device processor usage • Monitoring device queue lengths • Monitoring device memory utilization • Measuring total response times
Collecting a History of Utilization Data	• Measuring utilization and response times at different times of the day • Measuring utilization and response times on different days over an extended period
Capacity Planning	• Manually graphing or using a network management tool to graph utilization as a function of time to detect trends • Preparing trend reports to document projected need for and the cost of network expansion.
Setting Notification Thresholds	• Having a network management tool poll devices for values of critical parameters and graphing these values as a function of time • Setting polling intervals • Setting alarms/alerts on those parameters when the threshold is reached or a percentage of it is reached • Initiating an action when the threshold is reached such a sending a message to the network manager.
Building Databases	• Having the network management tool create a database of records containing device name, parameter, threshold and time for off-line analysis. • Using the database to extract time dependence of utilization • Using the time dependence of parameters to decide when network upgrades will be necessary to maintain performance
Running Network Simulations	• Using a simulation tool to develop a model of the network • Using the model's parameters and utilization data to optimize network performance
Latency	• Query/Response time interval

Network Management
OSI functional classification

- Configuration and Name Management:
  - Installation of new hardware/software
  - Tracking changes in control configuration
  - Who, what and why? - network topology
  - Revert/undo changes
  - Change management
  - Configuration audit
  - Does it do what was intended

Configuration Management

- installation of new hardware/software
- tracking changes in control configuration
  - who, what and why!
- revert/undo changes
- change management
- configuration audit
  - does it do what was intended?

Configuration Management Sub-categories

Configuration (Local)	• Choice of medium access protocol • Choice of correct cabling and connectors • Choice of cabling layout • Determining the number of physical interfaces on devices • Setting device interface parameter values ❑ Interrupts ❑ I/O Addresses ❑ DMA numbers ❑ Network layer addresses (e.g. IP, NetWare, etc) • Configuration of multiport devices (e.g. hubs, switches and routers) • Use of the Windows Registry • Comparing current versus stored configurations • Checking software environments • SNMP service
Configuration (Remote)	• From the network management station • Disabling device ports • Redirecting port forwarding • Disabling devices • Comparing current versus stored configurations • Configuring routing tables • Configuring security parameters such as community strings and user names • Configuring addresses of management stations to which traps should be sent • Verifying integrity of changes

Configuration Management Sub-categories

Configuration (Automated)	• Using the Dynamic Host Configuration Protocol (DHCP) to configure IP addresses • Using Plug and Play enabled NICs for automatic selection of interrupts and I/O addresses • Domain Name Services (DNS) addresses • Trap messages from agents
Inventory (Manual)	• Maintaining records of cable runs and the types of cables used • Maintaining device configuration records • Creating network database containing for each device: • Device types ❑ Software environment for each device ❑ operating systems ❑ utilities • drivers • applications ❑ versions ❑ configuration files (.ncf, .ini, .sys) • vendor contact information • IP address • Subnet address
Inventory (Automated)	• Auto-discovery of devices on the network using an NMS • Auto-determination of device configurations using an NMS • Creation of a network database • Auto-mapping of current devices to produce a network topological map • Accessing device statistics using an NMS and the Desktop Management Protocol

Network Management
OSI functional classification

- Security management
  - Security services: generating, distributing, storing of encryption keys for services
  - Exception alarm generation, detection of problems
  - Uniform access control to resources
  - Backups, data security
  - Security logging

Security Management

- exception alarm generation
- detection
- uniform access controls to resources
- backup

Security Management Sub-categories

Applying Basic Techniques	• Identifying hosts that store sensitive information • Management of passwords • Assigning user rights and permissions • Recording failed logins • Setting remote access barrier codes • Employing virus scanning • Limiting views of the Enterprise network • Tracking time and origin of remote accesses to servers
Identifying Access Methods Used	• Electronic Mail • File Transfer • Web Browsing • Directory Service • Remote Login • Remote Procedure Call • Remote Execution • Network Monitors • Network Management System
Using Access Control Methods	• Encryption • Packet filtering at routers • Packet filtering at firewalls • Source host authentication • Source user authentication
Maintenance	• Audits of the activity at secure access points • Executing security attack programs (Network Intrusion Detection) • Detecting and documenting breaches
Accessing Public Data Networks	• No restrictions - hosts are responsible for securing all access points • Limited access - only some hosts can interface with the Public Data Network using a proxy server
Using an Automated Security Manager	• Queries the configuration database to identify all access points for each device. • Reads event logs and notes security-related events. • Security Manager shows a security event on the network map. • Reports of invalid access point attempts are generated daily for analysis

Network Management - FCAPS

OSI functional classification

- Accounting management
  - Identifying consumers and suppliers of network resources - users and groups
  - Mapping network resources consumption to customer identity
  - Billing

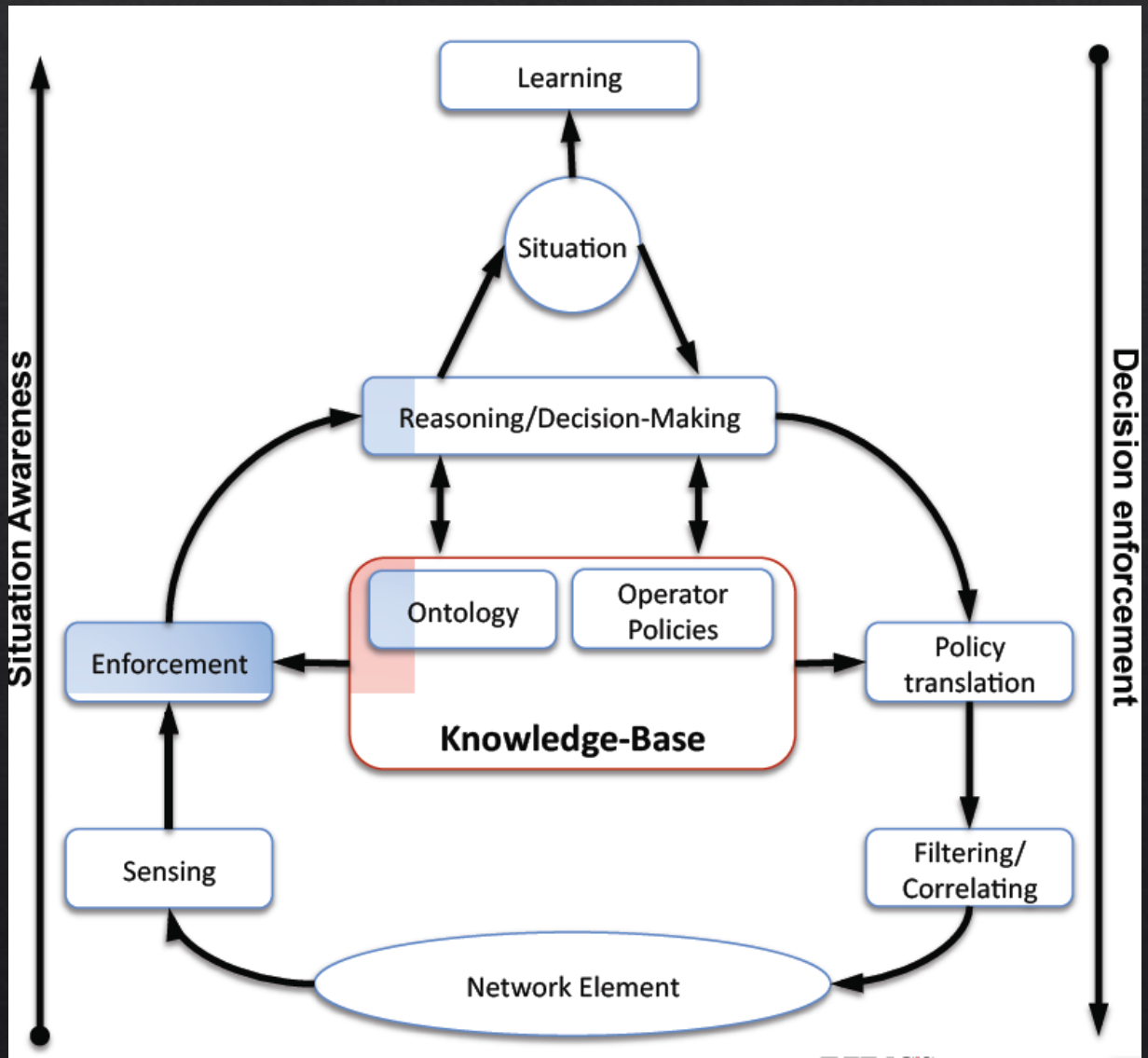
Accounting Management

- identifying consumers and suppliers
  - of network resources
- mapping network resources to customer identity
- charge back
  - volumetric data
  - time data
  - date time of day

Accounting Management Sub-categories

Gather Network Device Utilization Data	• Measure usage of resources by cost center • Set quotas to enable fair use of resources • Site metering to track adherence to software licensing
Bill Users of Network Resources	• Set charges based on usage. • Measure one of the following ❑ Number of transactions ❑ Number of packets • Number of bytes • Set charges on direction of information flow
Use and Accounting Management Tools	• Query usage database to measure statistics versus quotas • Define network billing domains • Implement automatic billing based on usage by users in the domain • Enable billing predictions • Enable user selection of billing domains on the network map
Reporting	• Create historical billings trends • Automatic distribution of billing to Cost Centers • Project future billings by cost center

Situation awareness and decision making



Knowledge fusion

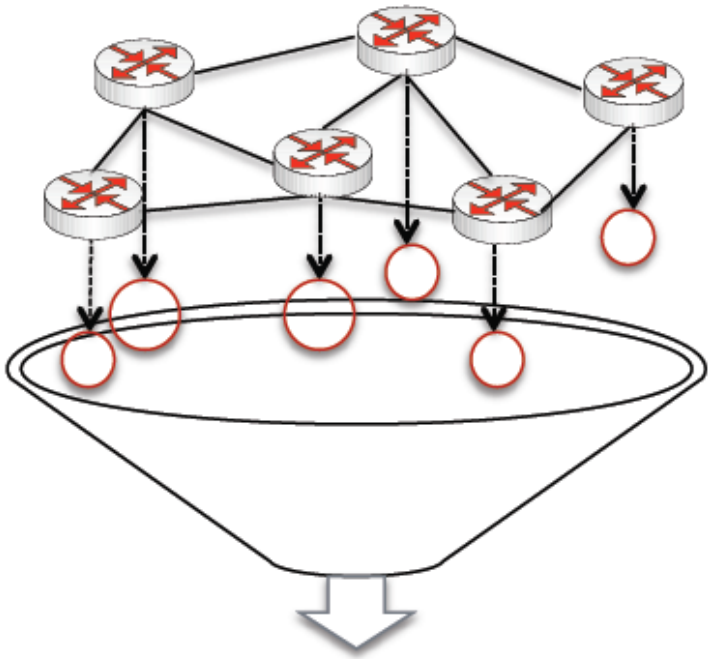
- Combine -> Transform -> Manipulate

Network monitoring tools

- NETCONF: Network Configuration Protocol
- Simple Network Management Protocol (SNMP)

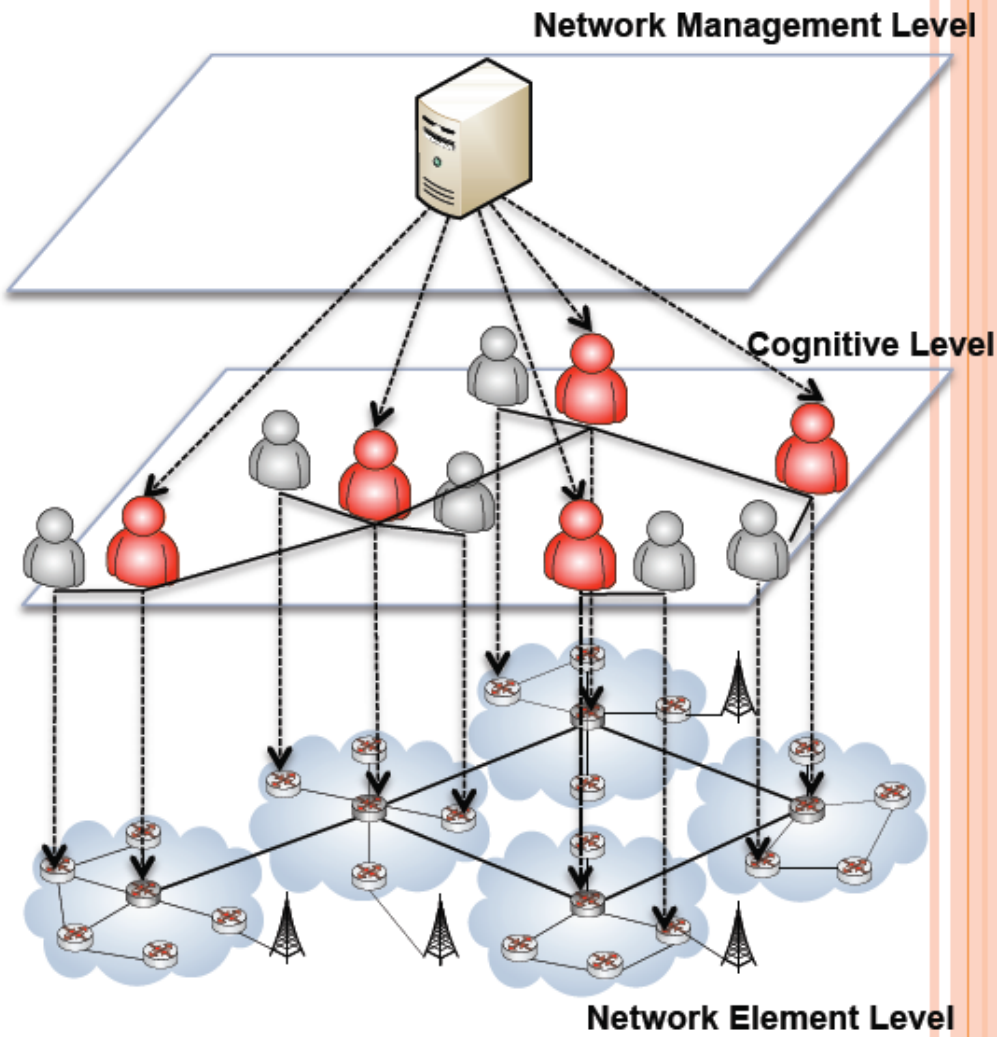
Data Fusion techniques

- Mathematical Methods
  - Dempster–Shafer's Evidential Theory
  - etc...
- Logic Programming
  - Inductive logic programming
  - Constraint logic programming
  - Abductive logic programming
  - etc...



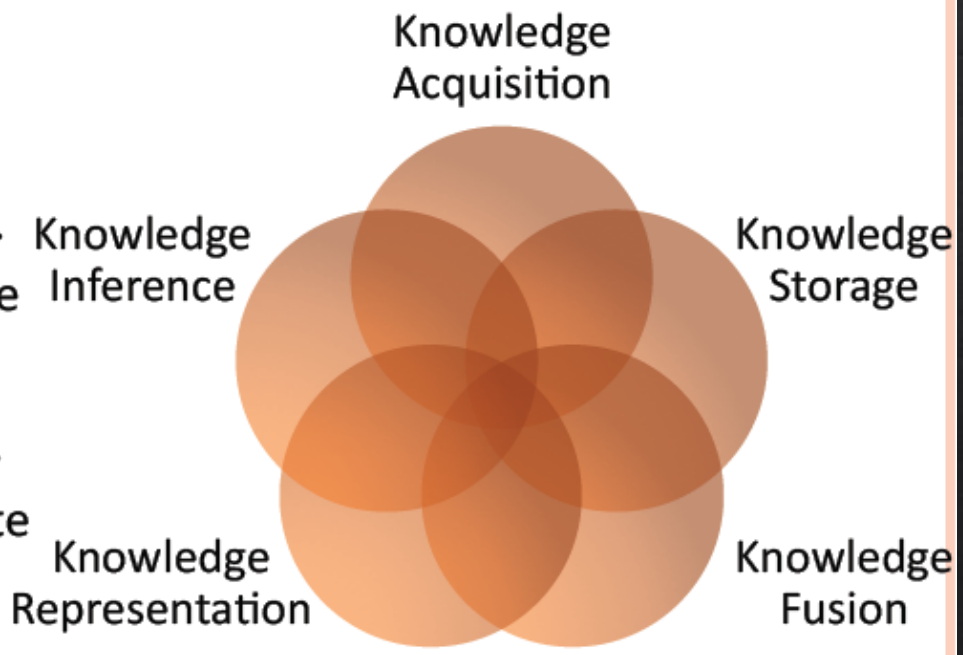
Cognitive network management

- Governing all intelligences in the network
- Tackling all the complexity tasks, performing tasks, monitoring events and making decisions on behalf of the network administrator
- Monitoring on all individual intelligence capabilities, interconnect all individual intelligence functionalities and manage all individual operation invocations in the system etc.



Cognitive Network Knowledge Tools

- Knowledge Acquisition
  - Collecting the right information, at the right time and the right location
- Knowledge Storage
  - Remembering data -> information -> object -> event then retrieve or reuse
- Knowledge Fusion
  - Collecting information -> category -> combine and transform -> manipulate
- Knowledge Representation
  - Organising information
- Knowledge Inference
  - Deriving a logical consequence conclusion



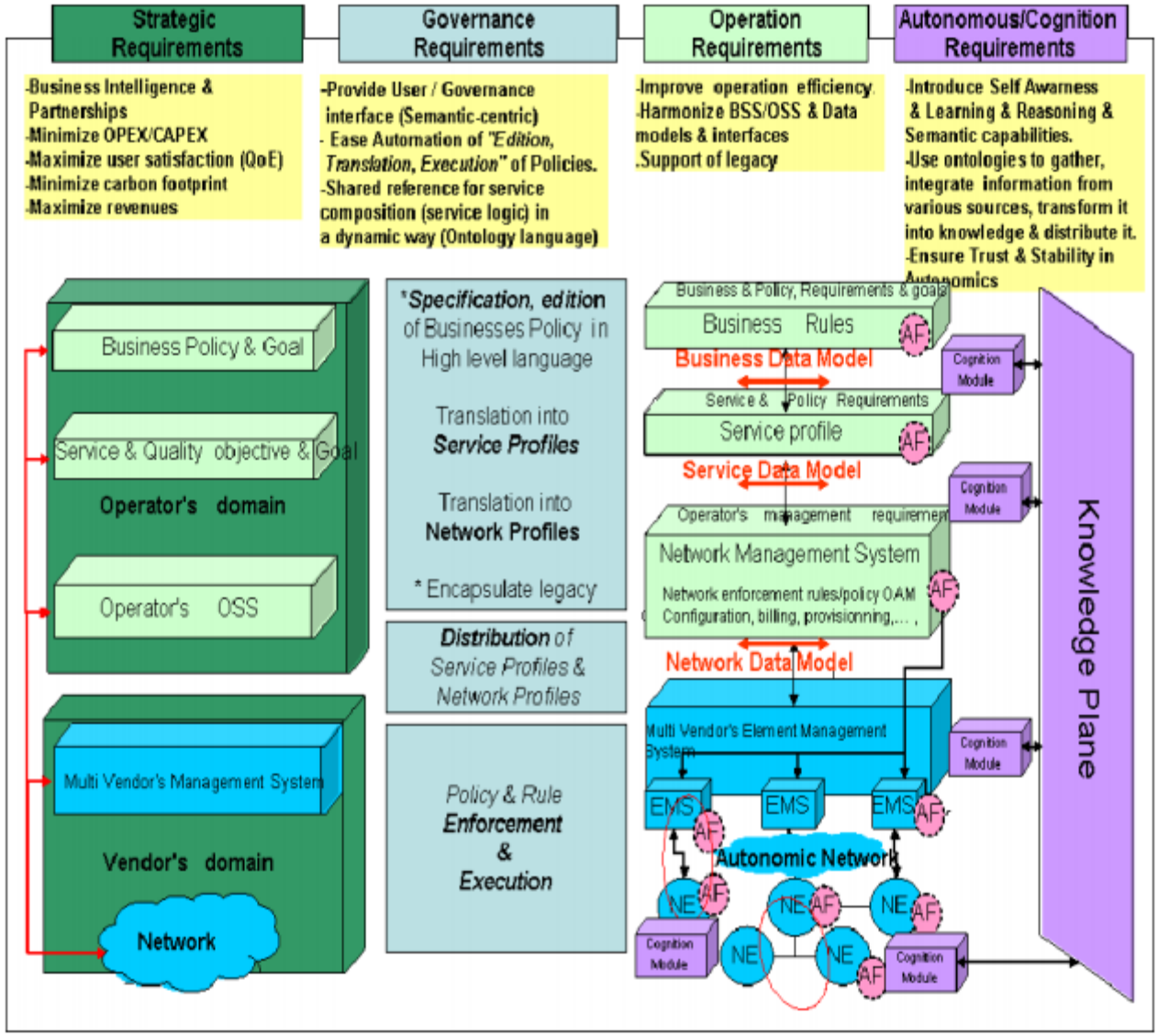


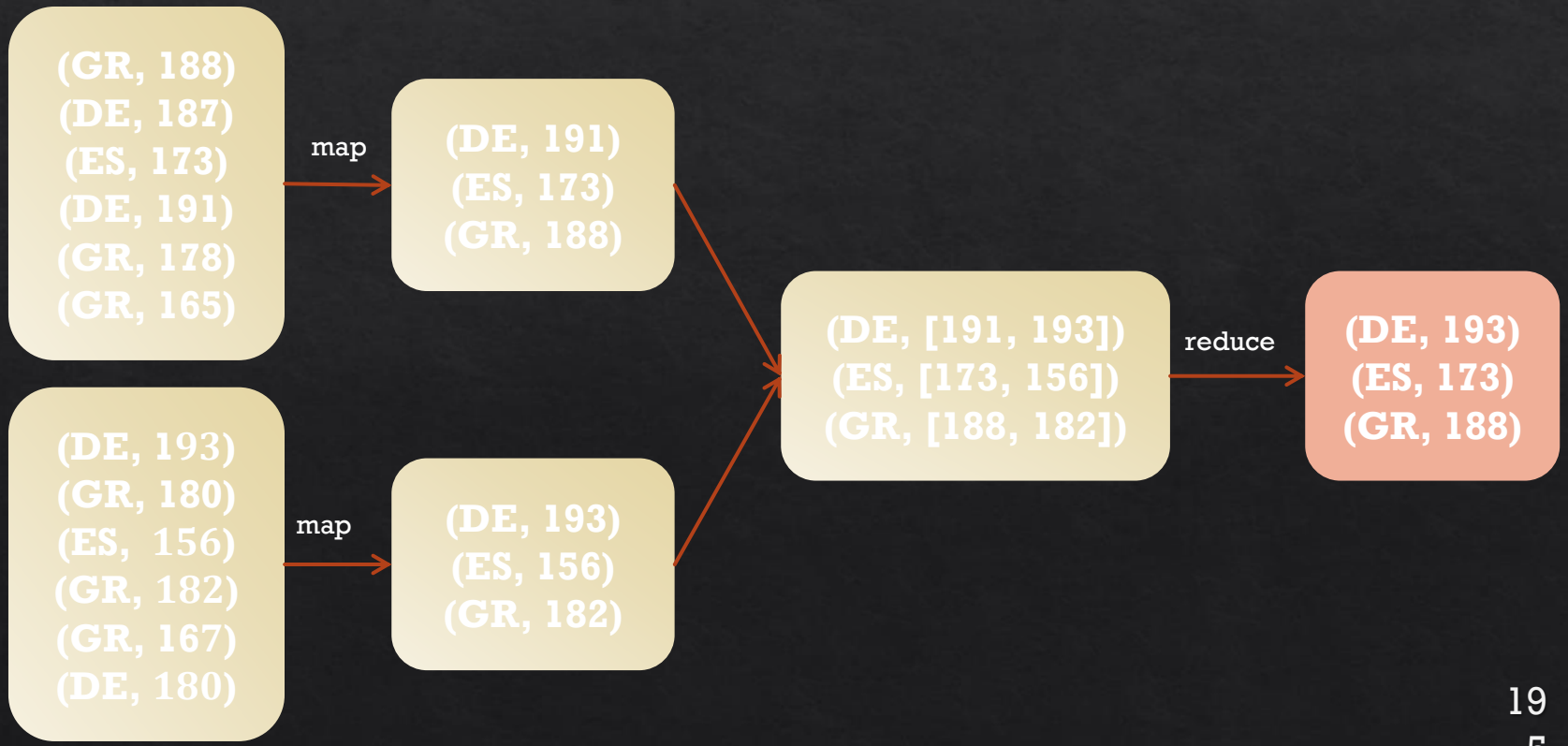
Figure 2: Requirement framework for a Policy- based management of an Autonomics Network

Big Data :
Why Distributed Dimensionality Reduction?

- Can handle very large datasets – processing billions of records cannot take place on a single device.
- If data are dispersed in a number of devices, it is resource consuming to transmit all information to one single node.

MapReduce Programming Paradigm

- Distributed data processing model
- Two phases: map & reduce
- Both phases have key – value pairs as input and output



Apache Hadoop

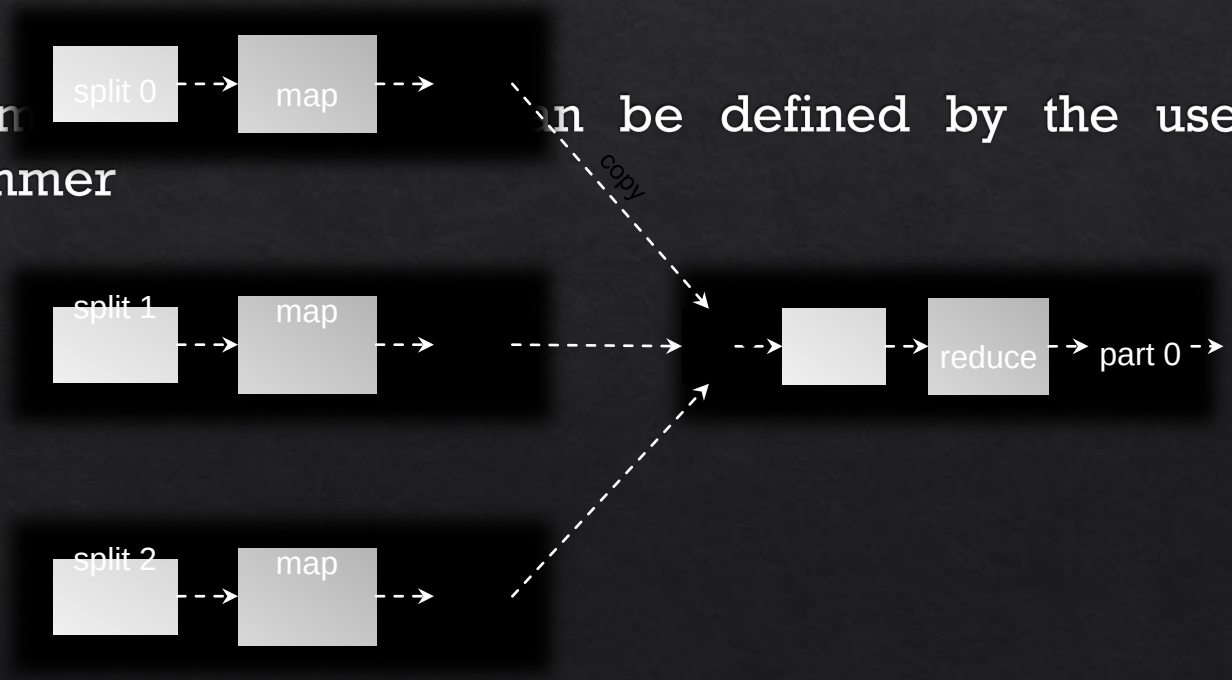
- Open source framework from Apache
- Designed for distributed processing on large scale datasets
- Written in Java, but supports other languages
- Two main subprojects:
  - HDFS → Hadoop Distributed File System
  - MapReduce framework → the most popular implementation of the MapReduce programming paradigm
- Other subprojects:
  - Pig → Distributed environment for data processing
  - HBase → Distributed database (column oriented)
  - ZooKeeper → Contains utilities for distributed processes

Apache Hadoop

- Runs on commodity hardware
- Designed to handle very large files (Gigabytes, Terabytes)
- Block size 64 Mb (default)
- Optimized for fast access to the whole dataset, not the first row
- Not a good choice for many small files
- Does not support simultaneous writers in a file, nor modifications in a random spot of a file

Apache Hadoop – MapReduce

- As many map tasks as the number of blocks of a file (input splits)
- After map phase, the mappers output is sorted and grouped by key
- The num[illegible]n be defined by the user – programmer



The Managed Object

- A network may be managed by representing network resources as managed objects. Each MO is a data variable representing one aspect of the managed resource e.g. on/off status, number of packets sent, etc.
- A collection of MOs is called the MIB (Management Information Base), that is, a collection of access points at the agent(s) for the network management system (NMS).

The Managed Object

- Monitoring equates to retrieving values from MIB objects in agents.
- Controlling equates to setting values within the MIB objects in agents. MOs are standardized across systems.
- The Structure of Management Information (SMI) defines syntax (format) and semantics (meaning) of management information stored in the Management Information Base (MIB). Abstract Syntax One (ASN.1) is a formal language standardized by ITU-T (X.208 and X.680) and ISO 8824 that clarifies how data are arranged , what meaning they have and the expected data type.

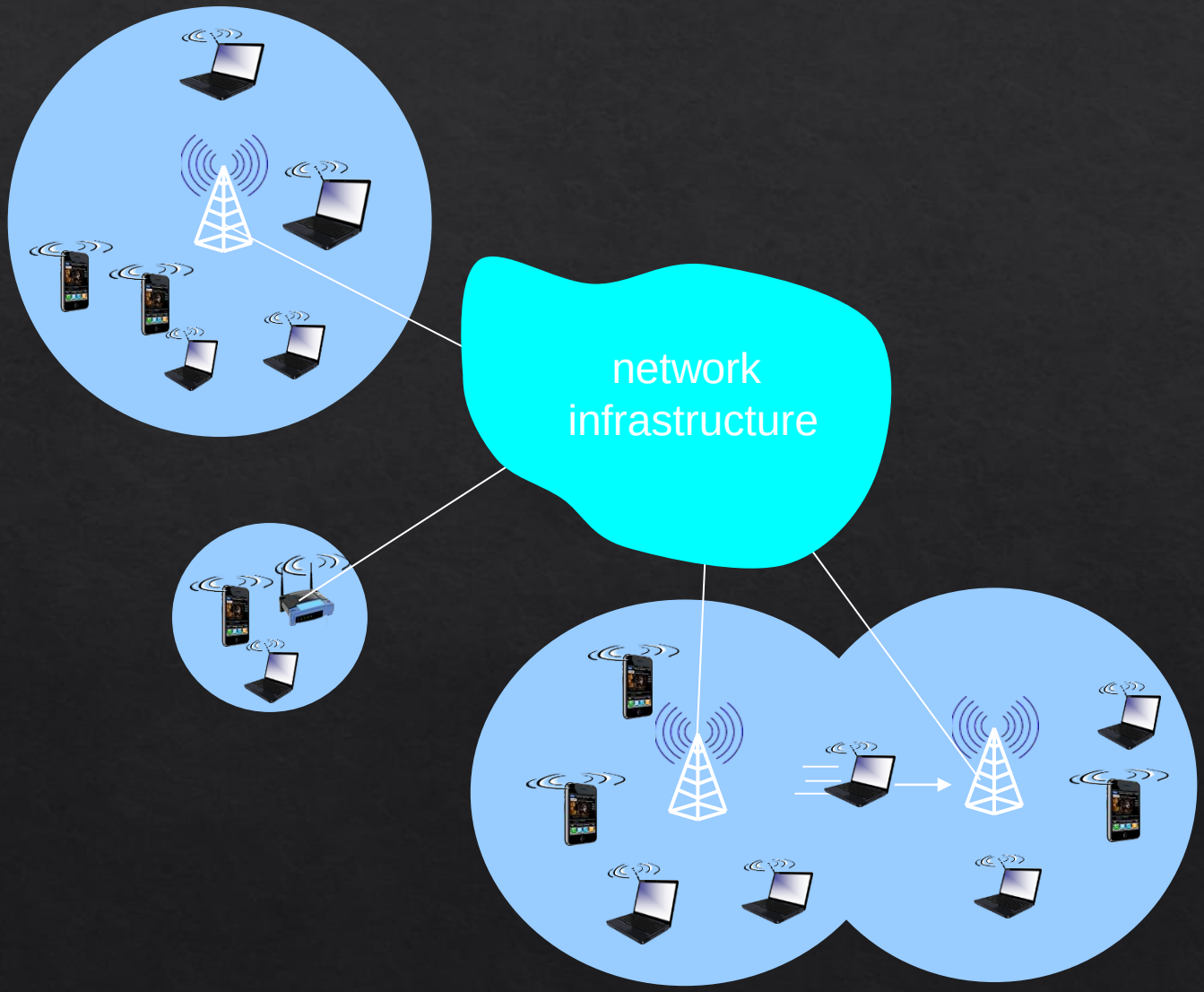
Cellular Wireless Networks

Wireless and Mobile Networks

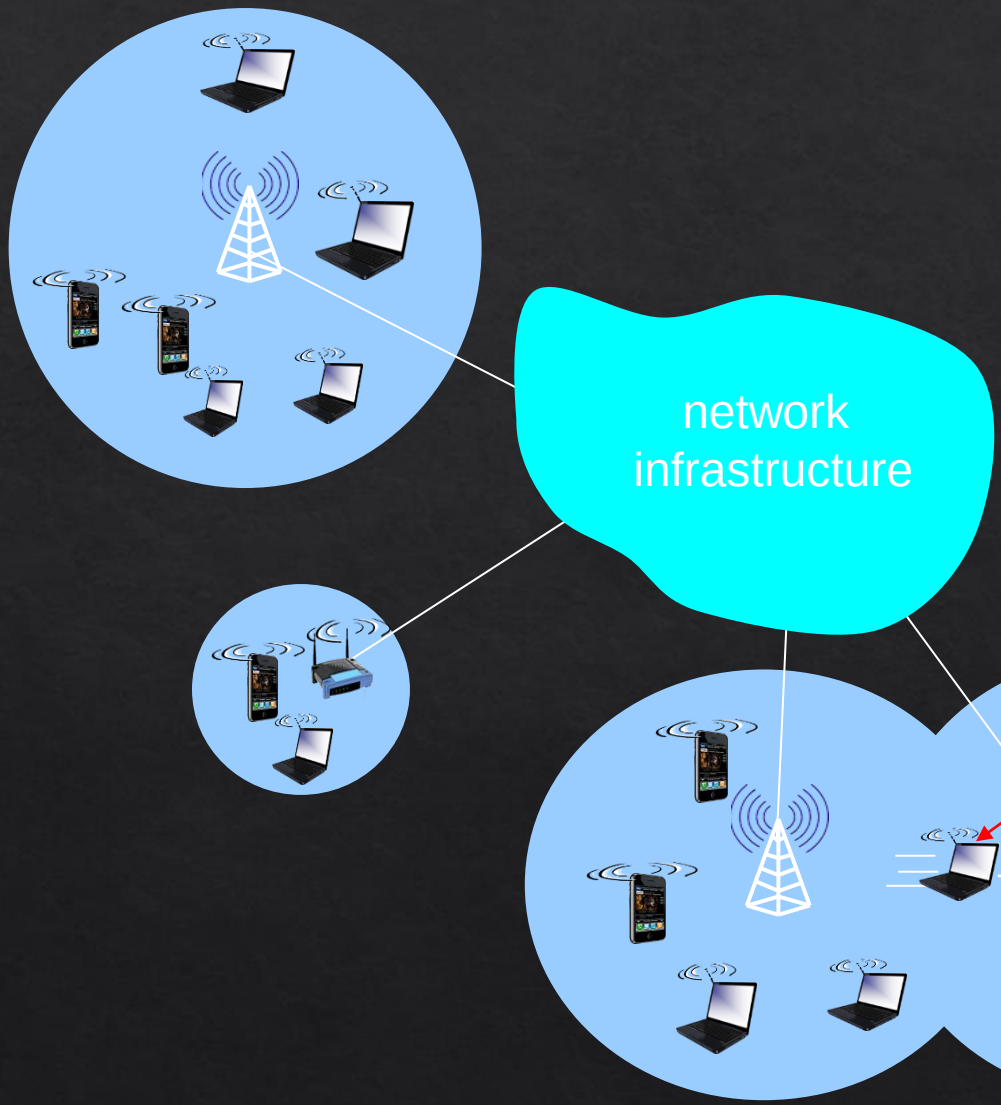
Background:

- # wireless (mobile) phone subscribers now exceeds # wired phone subscribers (5-to-1)!
- # wireless Internet-connected devices equals # wireline Internet-connected devices
  - laptops, Internet-enabled phones promise anytime untethered Internet access
- two important (but different) challenges
  - *wireless:* communication over wireless link
  - *mobility:* handling the mobile user who changes point of attachment to network

Elements of a wireless network



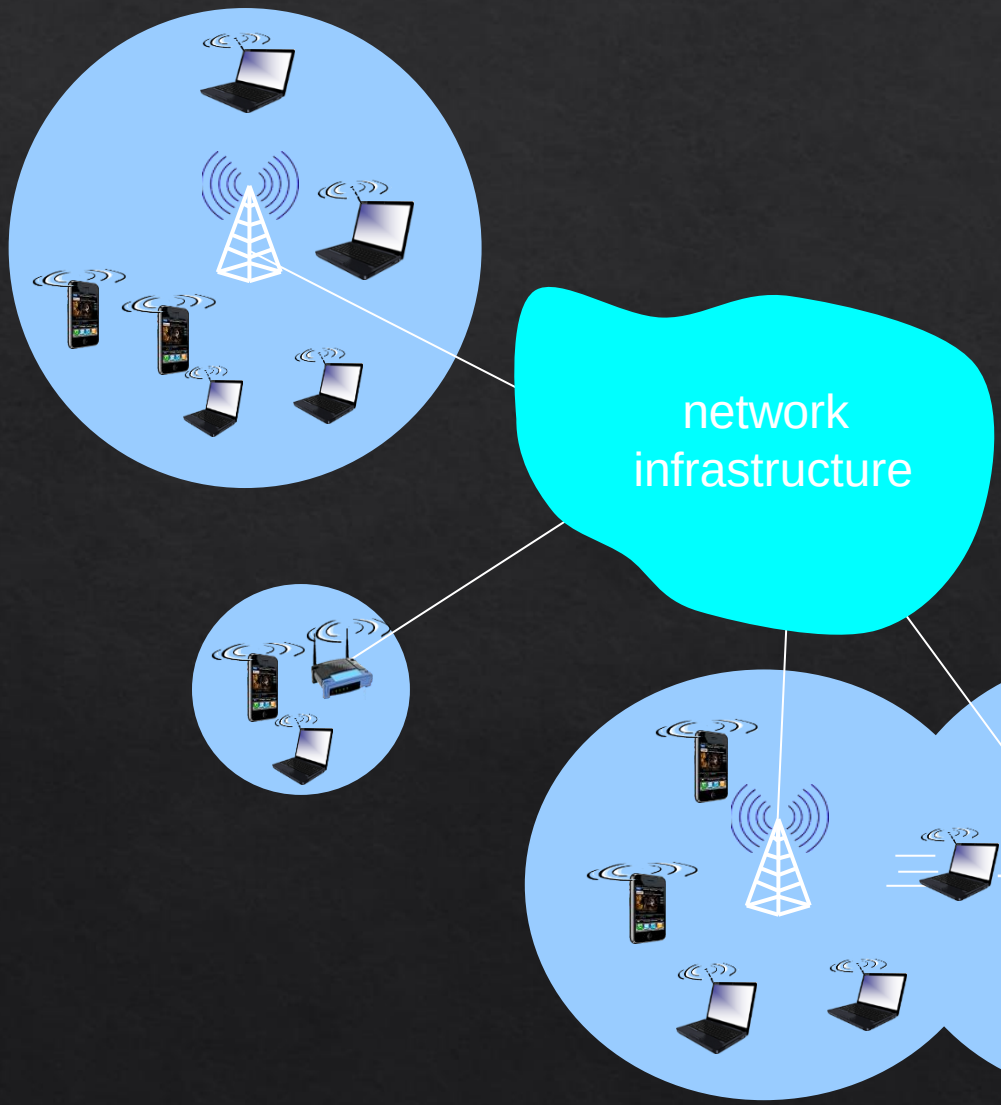
Elements of a wireless network

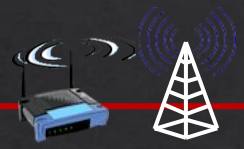


wireless hosts

- laptop, smartphone
- run applications
- may be stationary (non-mobile) or mobile
  - wireless does *not* always mean mobility

Elements of a wireless network

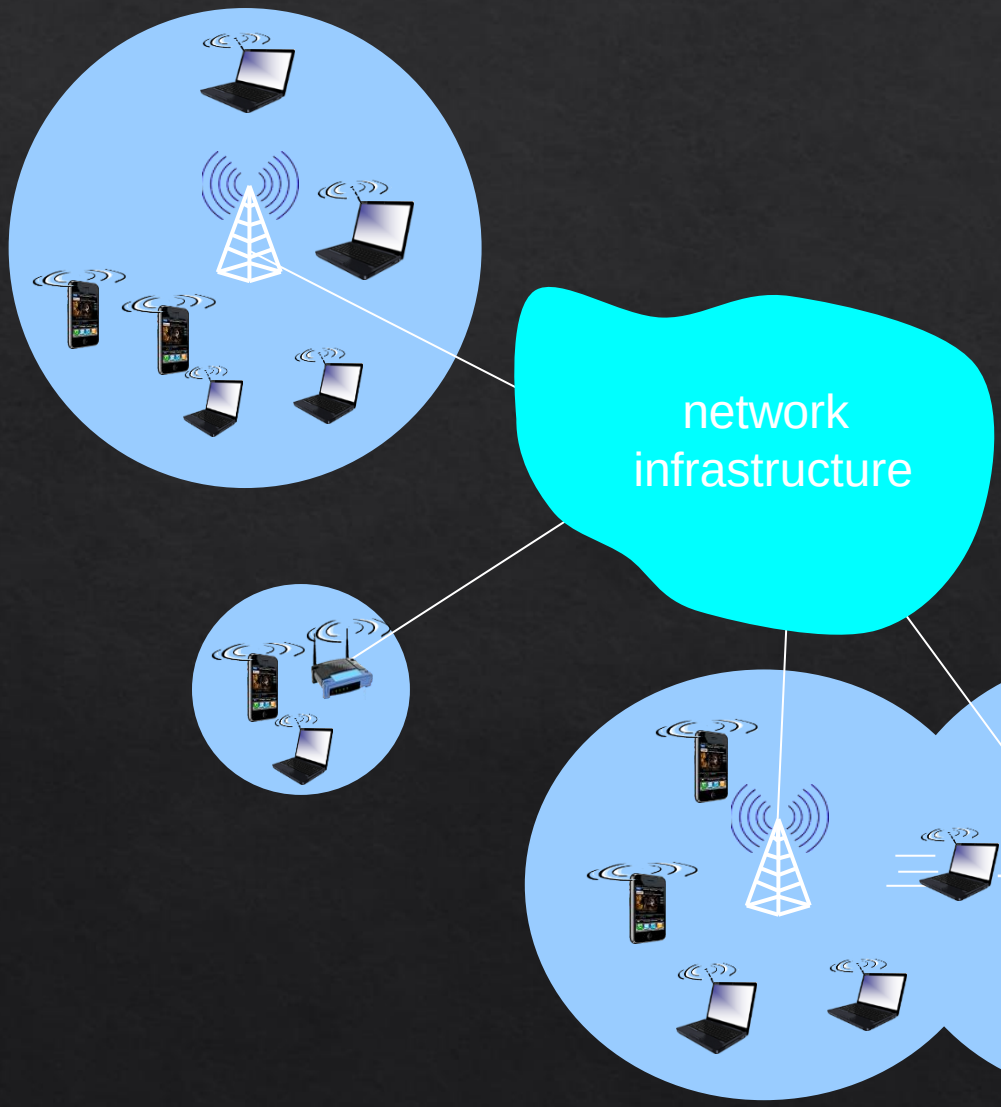




base station

- typically connected to wired network
- relay - responsible for sending packets between wired network and wired network and wireless host(s) in its "area"
  - e.g., cell towers, 802.11 access points

Elements of a wireless network



wireless link

- typically used to connect mobile(s) to base station
- also used as backbone link
- multiple access protocol coordinates link access
- various data rates, transmission distance

Wireless Link Characteristics

important differences from wired link ….

- *Spectrum:* the scarced resource – major issue for the NM
- *decreased signal strength:* radio signal attenuates as it propagates through matter (path loss)
- *interference from other sources:* standardized wireless network frequencies (e.g., 2.4 GHz) shared by other devices (e.g., phone); devices (motors) interfere as well
- *multipath propagation:* radio signal reflects off objects ground, arriving at destination at slightly different times

…. make communication across (even a point to point) wireless link much more "difficult"

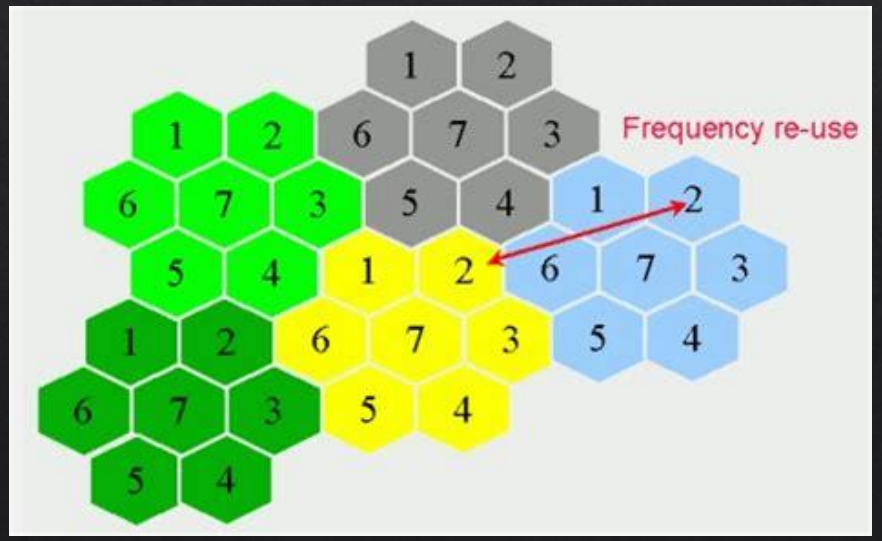
Frequency Reuse

The concept of frequency reuse is based on assigning to each cell a group of radio channels used within a small geographic area

- Cells are assigned a group of channels that is completely different from neighbouring cells
- The coverage area of cells is called the footprint and is limited by a boundary so that the same group of channels can be used in cells that are far enough apart

Frequency Reuse

- Cells with the same number have the same set of frequencies



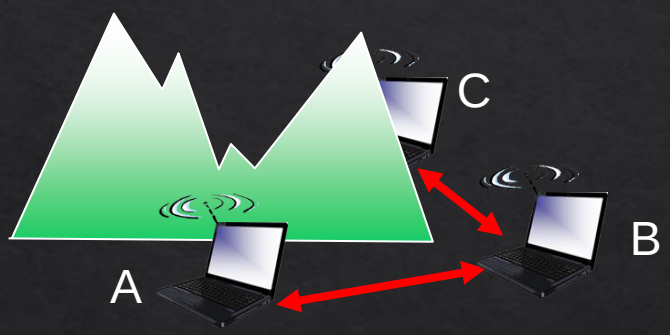
Frequency Reuse

Frequency Reuse using 7 frequencies allocations

Each cell is generally 6437 to 12874 m in diameter with a lower limit around 3218 m.

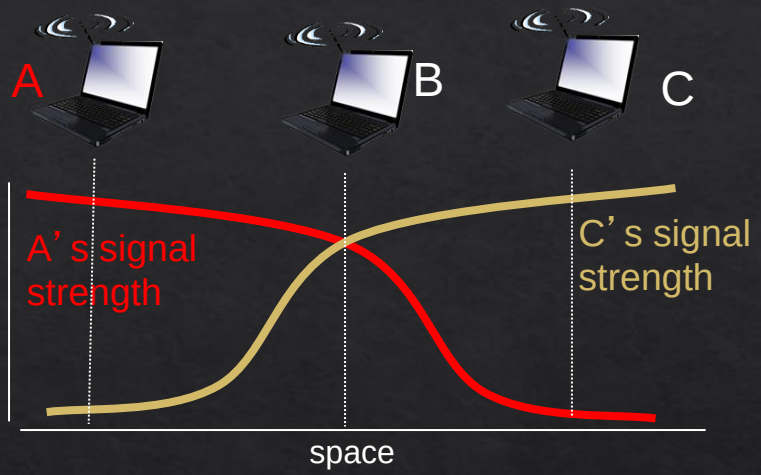
Wireless network characteristics

Multiple wireless senders and receivers create additional problems (beyond multiple access):



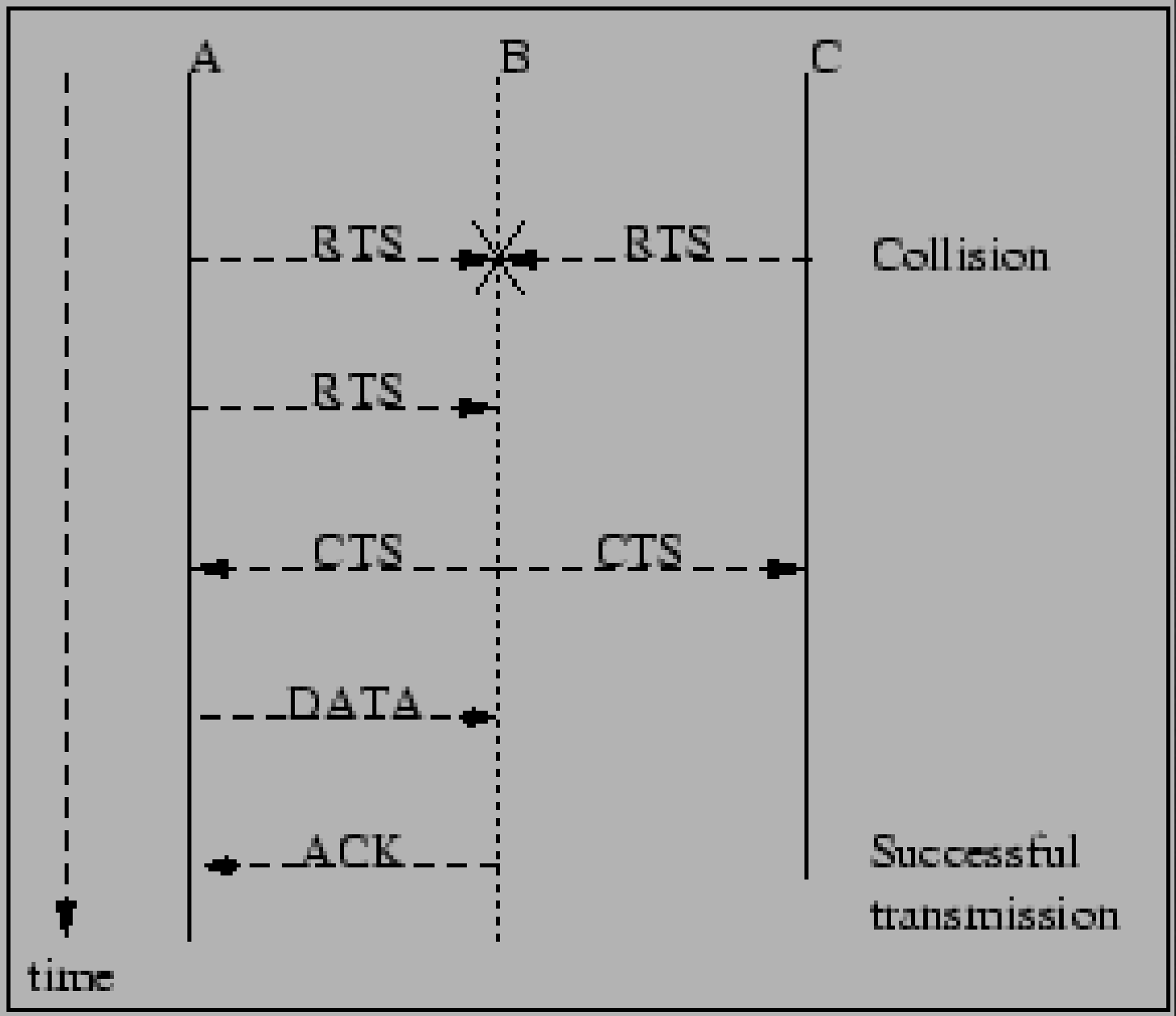
Hidden terminal problem

- B, A hear each other
- B, C hear each other
- A, C can not hear each other means A, C unaware of their interference at B



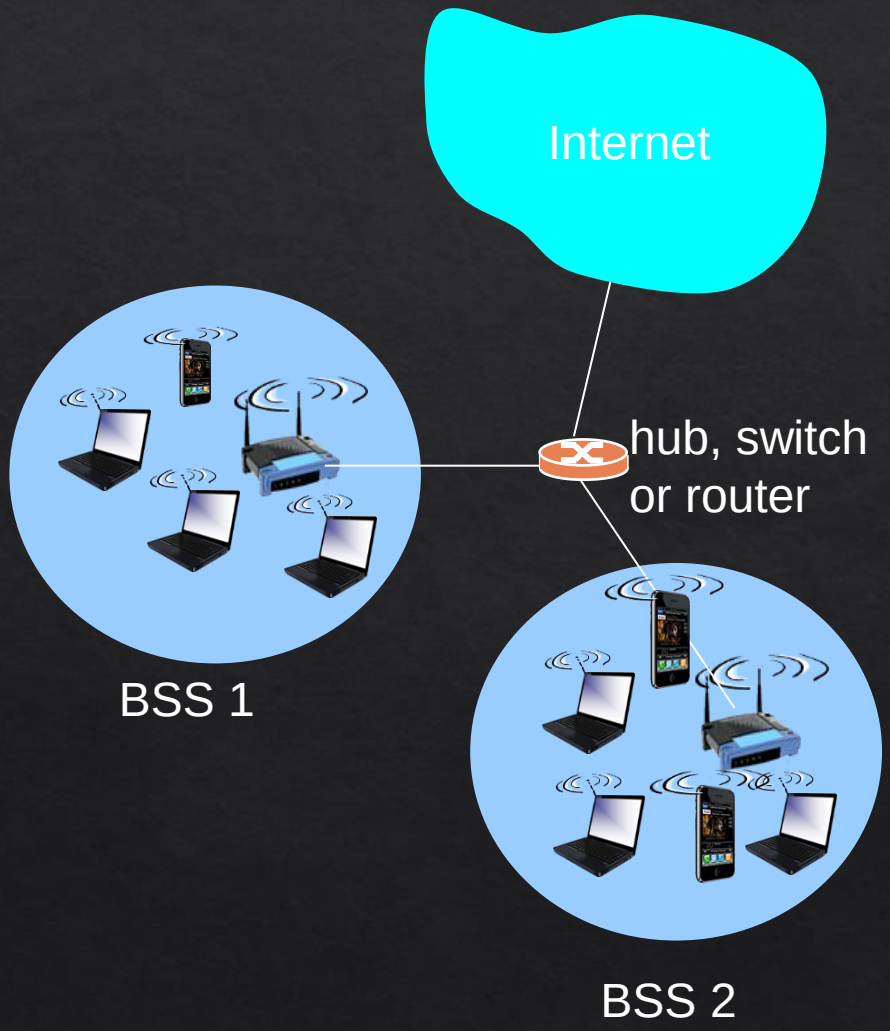
Signal attenuation:

- B, A hear each other
- B, C hear each other
- A, C can not hear each other interfering at B



A request to send (RTS) and clear to send (CTS) scheme. First, A and C transmits a packet simultaneously, causing a packet collision at B. Then A retransmits the packet before C does, thus capturing the channel.

802.11 LAN architecture

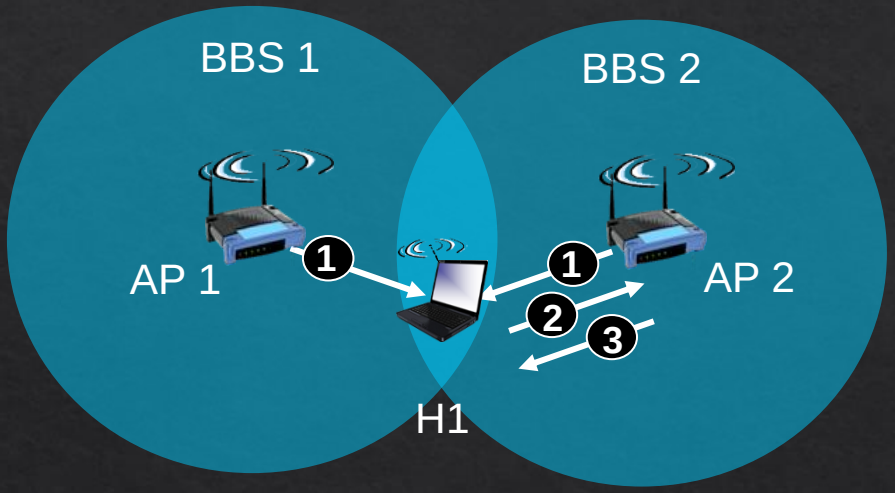


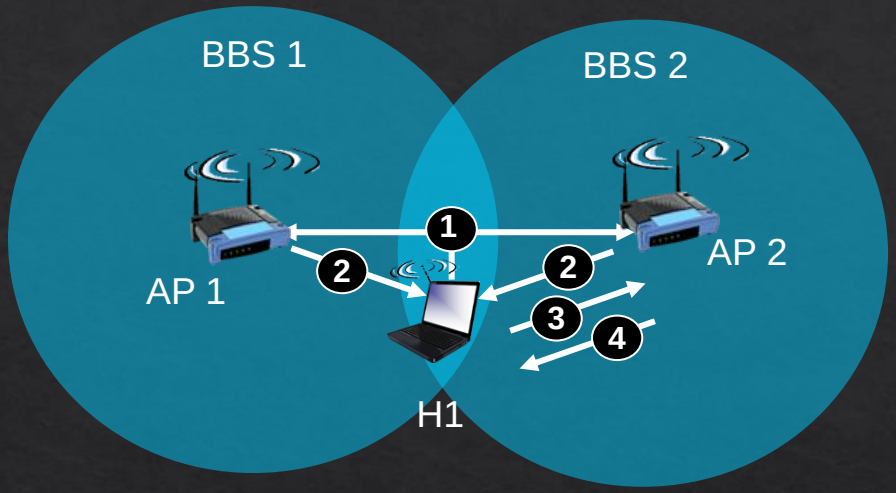
- wireless host communicates with base station
  - base station = access point (AP)
- Basic Service Set (BSS) (aka "cell") in infrastructure mode contains:
  - wireless hosts
  - access point (AP): base station
  - ad hoc mode: hosts only

802.11: Channels, association

- 802.11b: 2.4GHz-2.485GHz spectrum divided into 11 channels at different frequencies
  - AP admin chooses frequency for AP
  - interference possible: channel can be same as that chosen by neighboring AP!
- host: must *associate* with an AP
  - scans channels, listening for *beacon frames* containing AP's name (SSID) and MAC address
  - selects AP to associate with
  - may perform authentication
  - will typically run DHCP to get IP address in AP's subnet

802.11: passive/active scanning





passive scanning:

(1) beacon frames sent from APs

(2) association Request frame sent: H1 to selected AP

(3) association Response frame sent from selected AP to H1

active scanning:

(1) Probe Request frame broadcast from H1

(2) Probe Response frames sent from APs

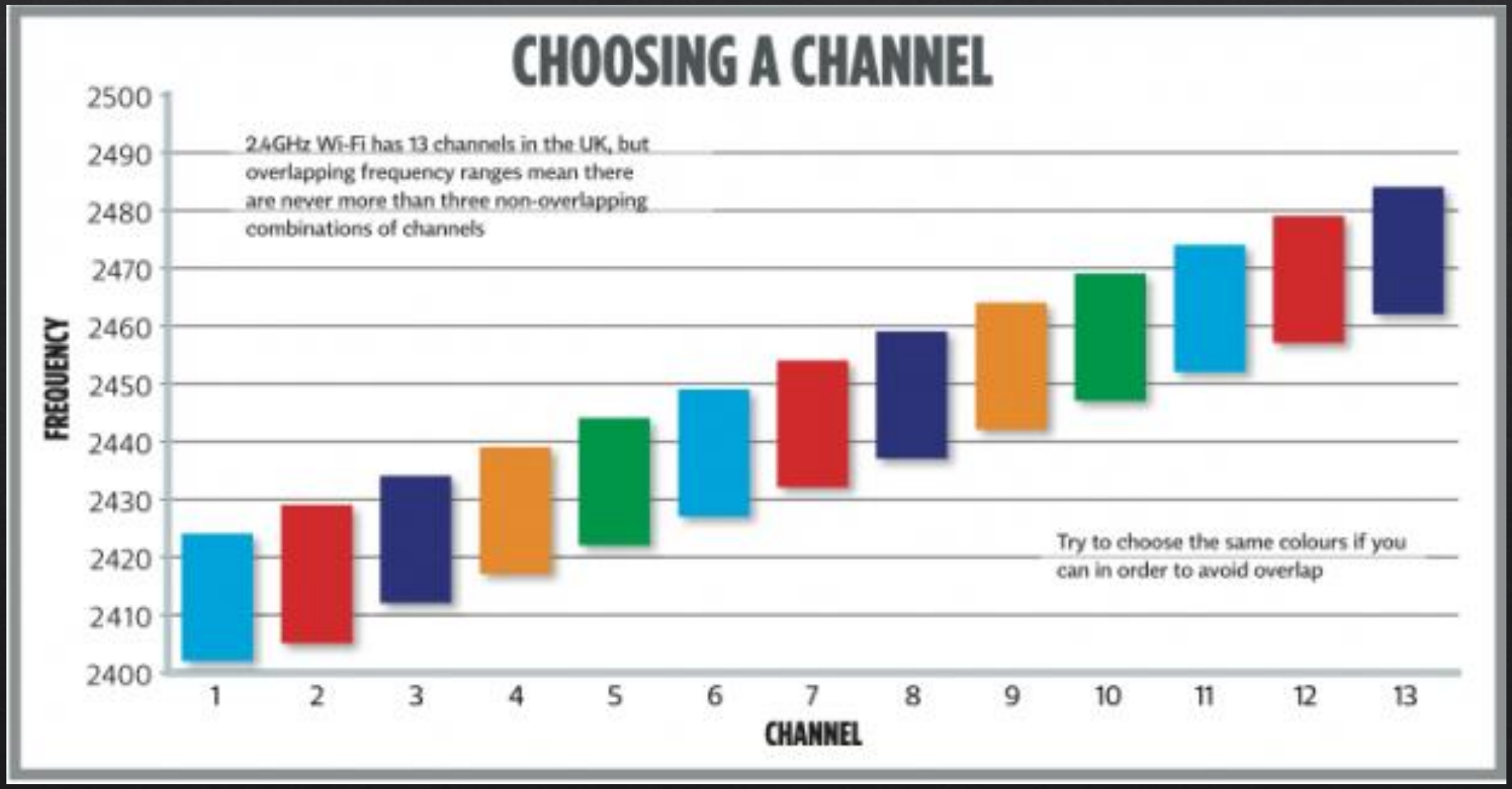
(3) Association Request frame sent: H1 to selected AP

(4) Association Response frame sent from selected AP to H1

802.11: advanced capabilities

power management

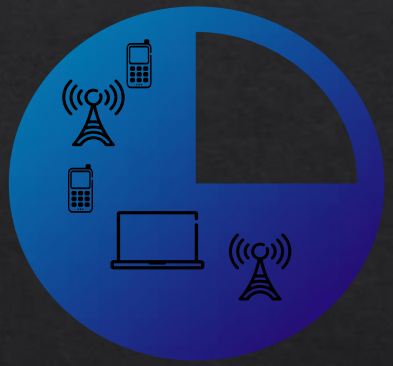
- node-to-AP: "I am going to sleep until next beacon frame"
  - AP knows not to transmit frames to this node
  - node wakes up before next beacon frame
- beacon frame: contains list of mobiles with AP-to-mobile frames waiting to be sent
  - node will stay awake if AP-to-mobile frames to be sent; otherwise sleep again until next beacon frame


CHOOSING A CHANNEL
2.4GHz Wi-Fi has 13 channels in the UK, but overlapping frequency ranges mean there are never more than three non-overlapping combinations of channels
Try to choose the same colours if you can in order to avoid overlap
FREQUENCY
2500
2490
2480
2470
2460
2450
2440
2430
2420
2410
2400
1
2
3
4
5
6
7
8
9
10
11
12
13
CHANNEL

LTE ”cost efficiency”

Wider pipe advantage

Self Organizing Networks



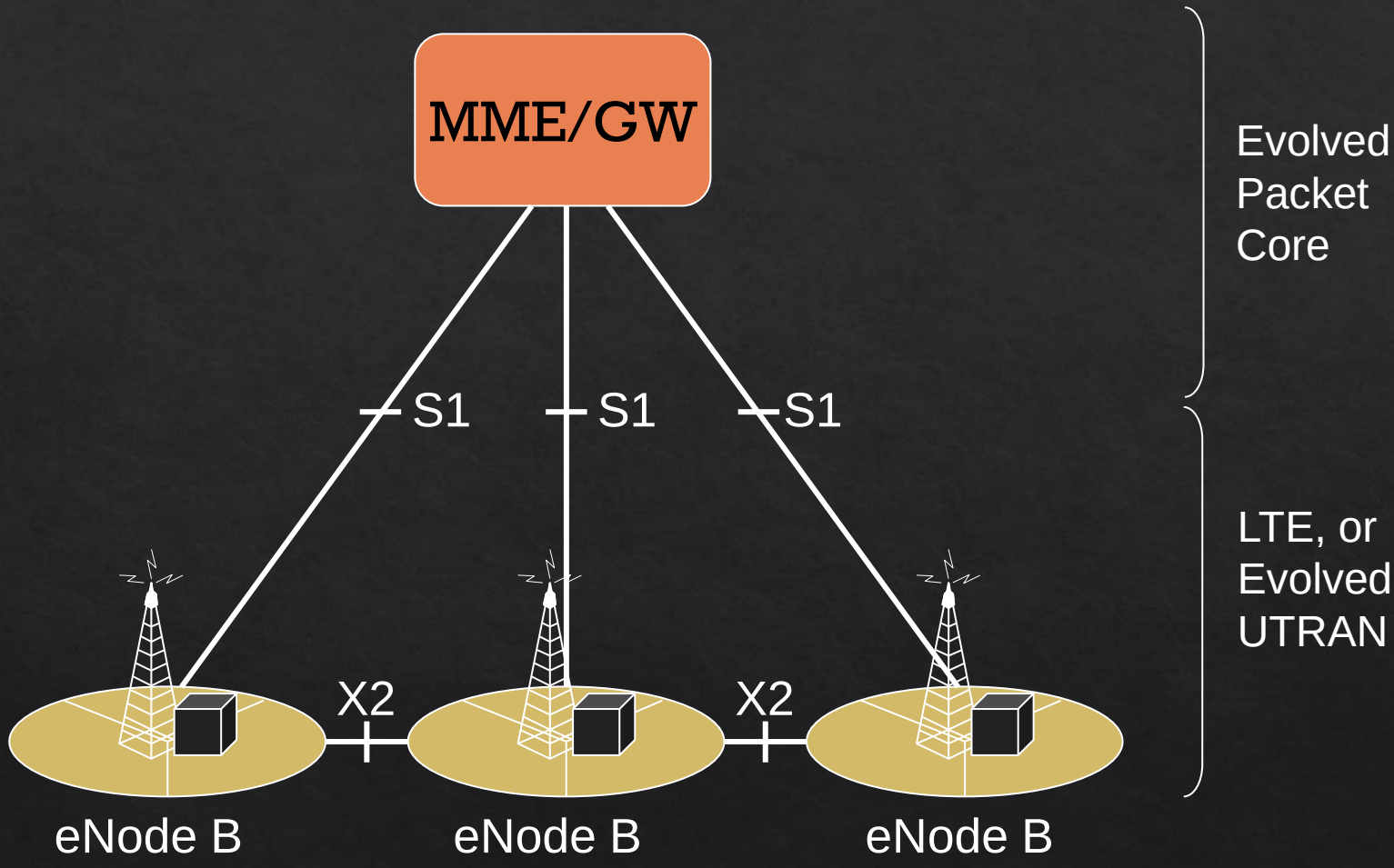
All-IP architecture

Economies of Scale

Low total cost of ownership

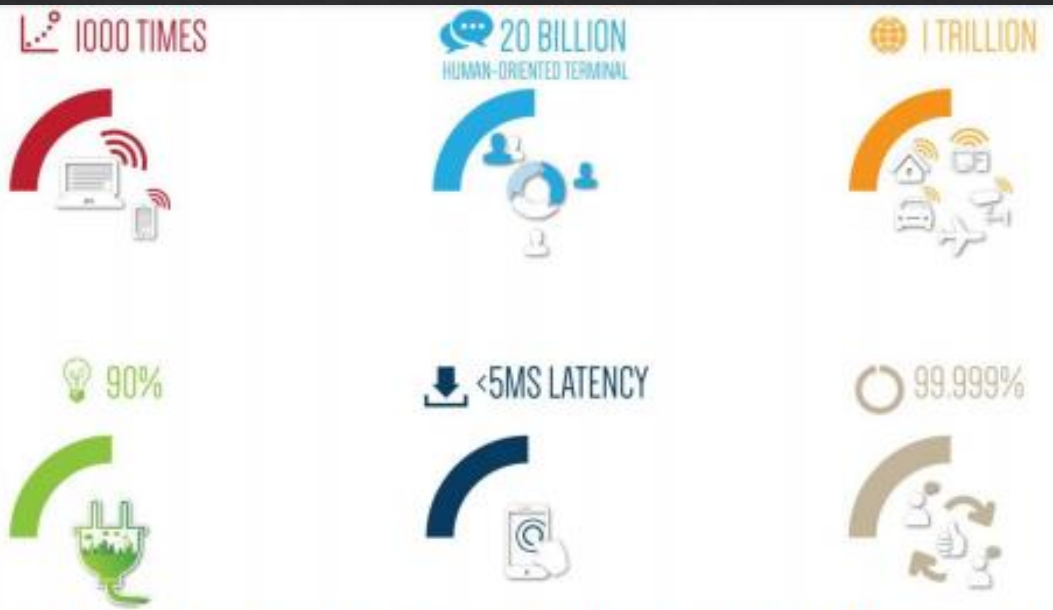
LTE interfaces

Logical view



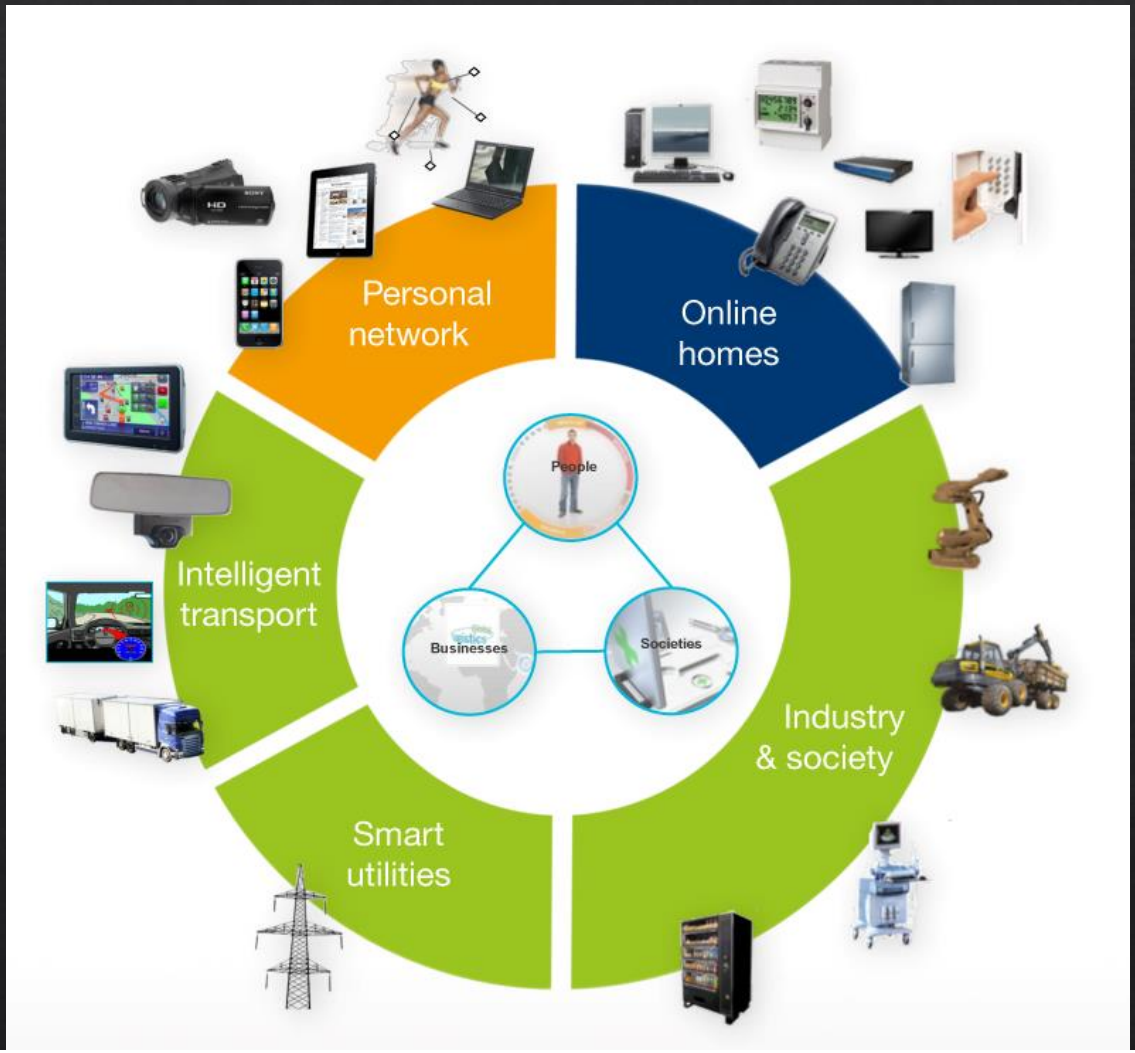

SERGEANT, I CAUGHT A SMARTPHONE AT 20Gbps!!
DID YOU GET HIS REGISTRATION?
NOT THE FULL NUMBER, BUT IT STARTED WITH 5G!!

5G



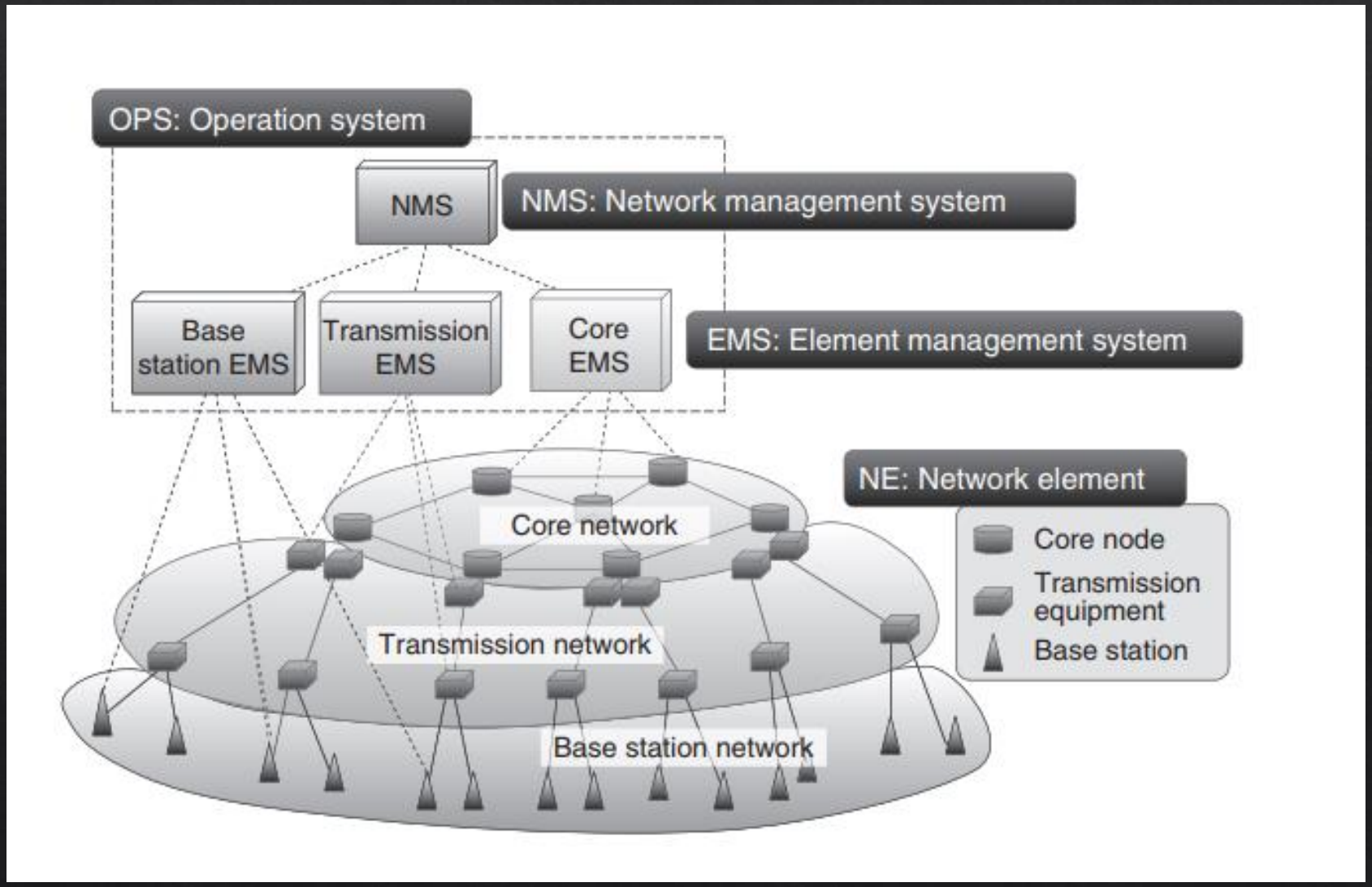
- 1,000 X in mobile data volume per geographical area reaching a target ≥ 10 Tb/s/km^2
- 1,000 X in number of connected devices reaching a density ≥ 1M terminals/km2
- 100 X in user data rate reaching a peak terminal data rate ≥ 10Gb/s
- Guaranteed user data rate >50Mb/s
- 1/10 X in energy consumption compared to 2010
- 1/5 X in end-to-end latency reaching 5 ms for e.g. tactile Internet and radio link latency reaching a target ≤ 1 ms for e.g. Vehicle to Vehicle communication
- 1/5 X in network management OPEX
- 1/1,000 X in service deployment time reaching a complete deployment in ≤ 90 minutes
- Mobility support at speed ≥ 500km/h for ground transportation
- Accuracy of outdoor terminal location ≤ 1m

50 billion connected devices vision – Ultra Dense Networks



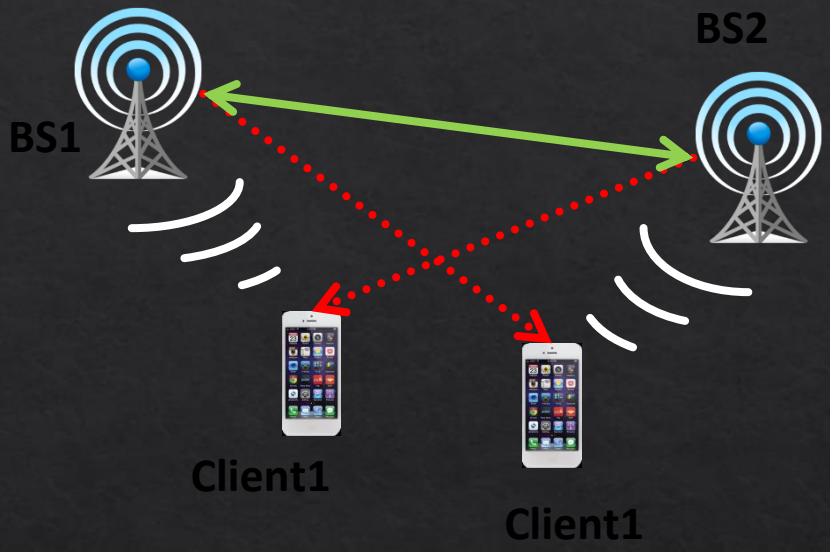
Everything that benefits from being connected will be connected

Mobile Network management



Network Programming

Radio Resource Management : Mobility



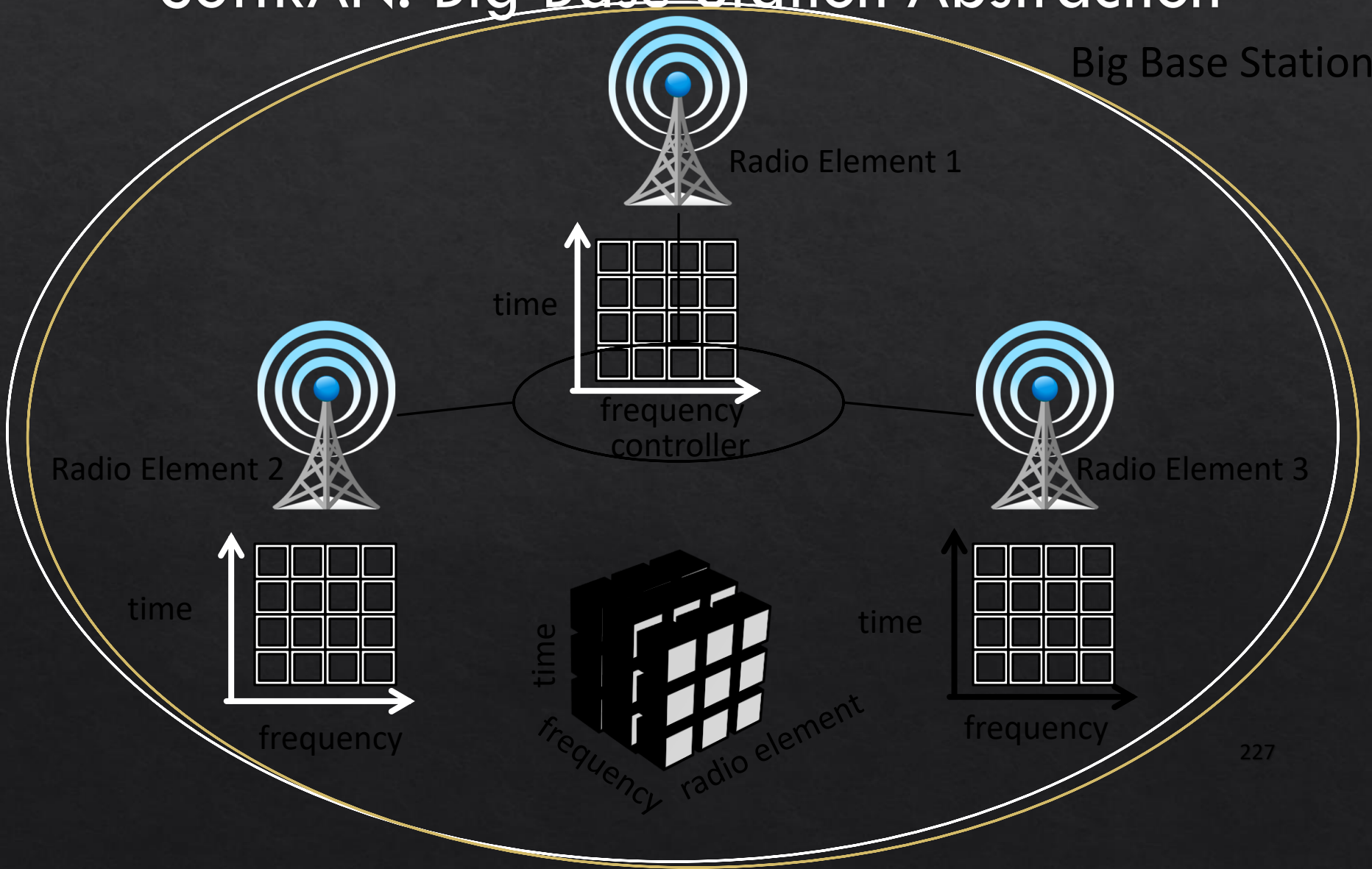
- Coordination required to decide handovers
- Load at BS1 reduces and load at BS2 increases

LTE-RAN: Current Architecture

- Distributed control plane
  - Control signaling grows with density
  - Inefficient RRM decision making
  - Harder to manage and operate the network
  - Clients need to resynchronize state at every handover

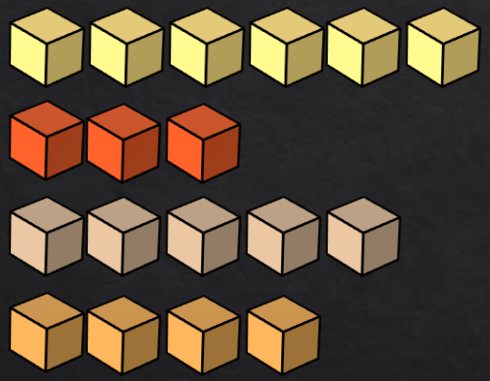
- Works fine with sparse deployments, but problems compound in a dense network

SoftRAN: Big Base Station Abstraction

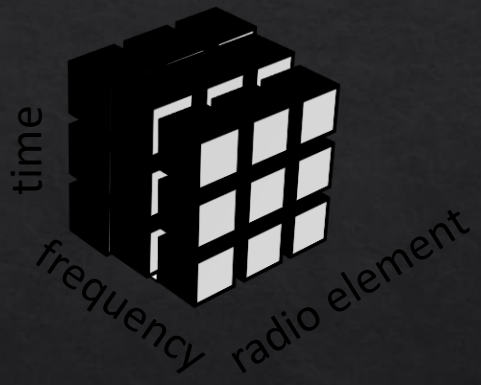


Radio Resource Allocation

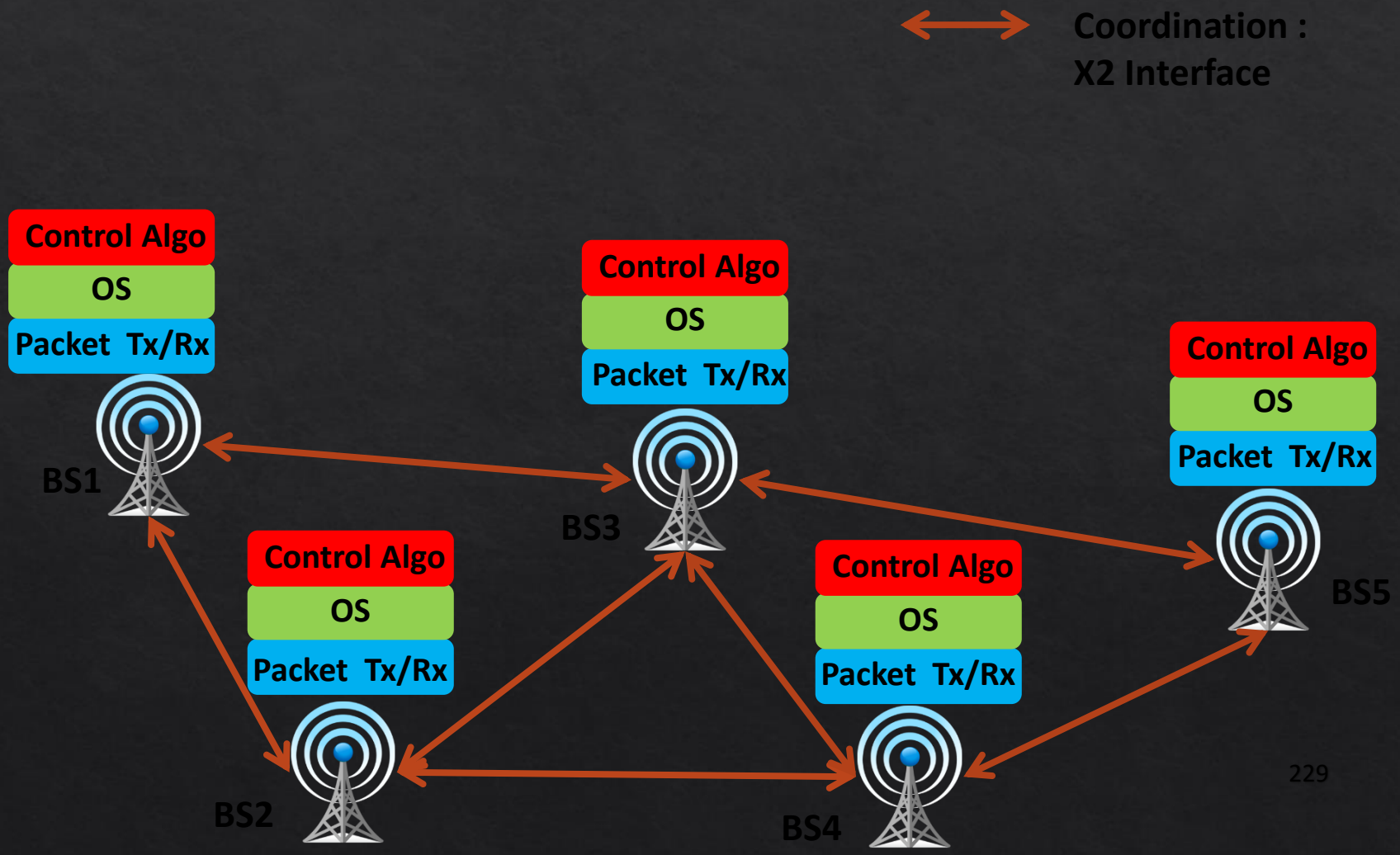
Flows



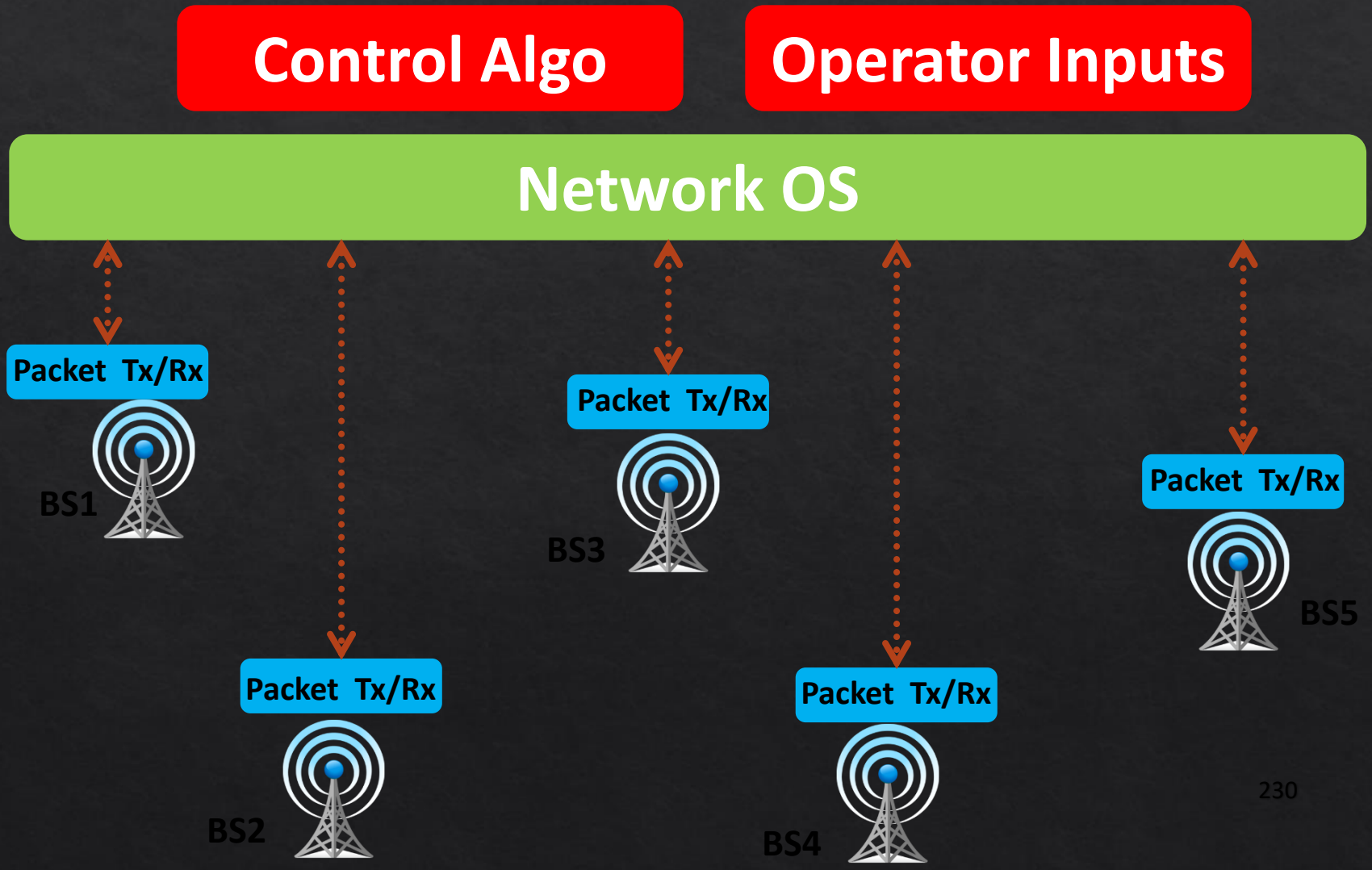
3D Resource Grid



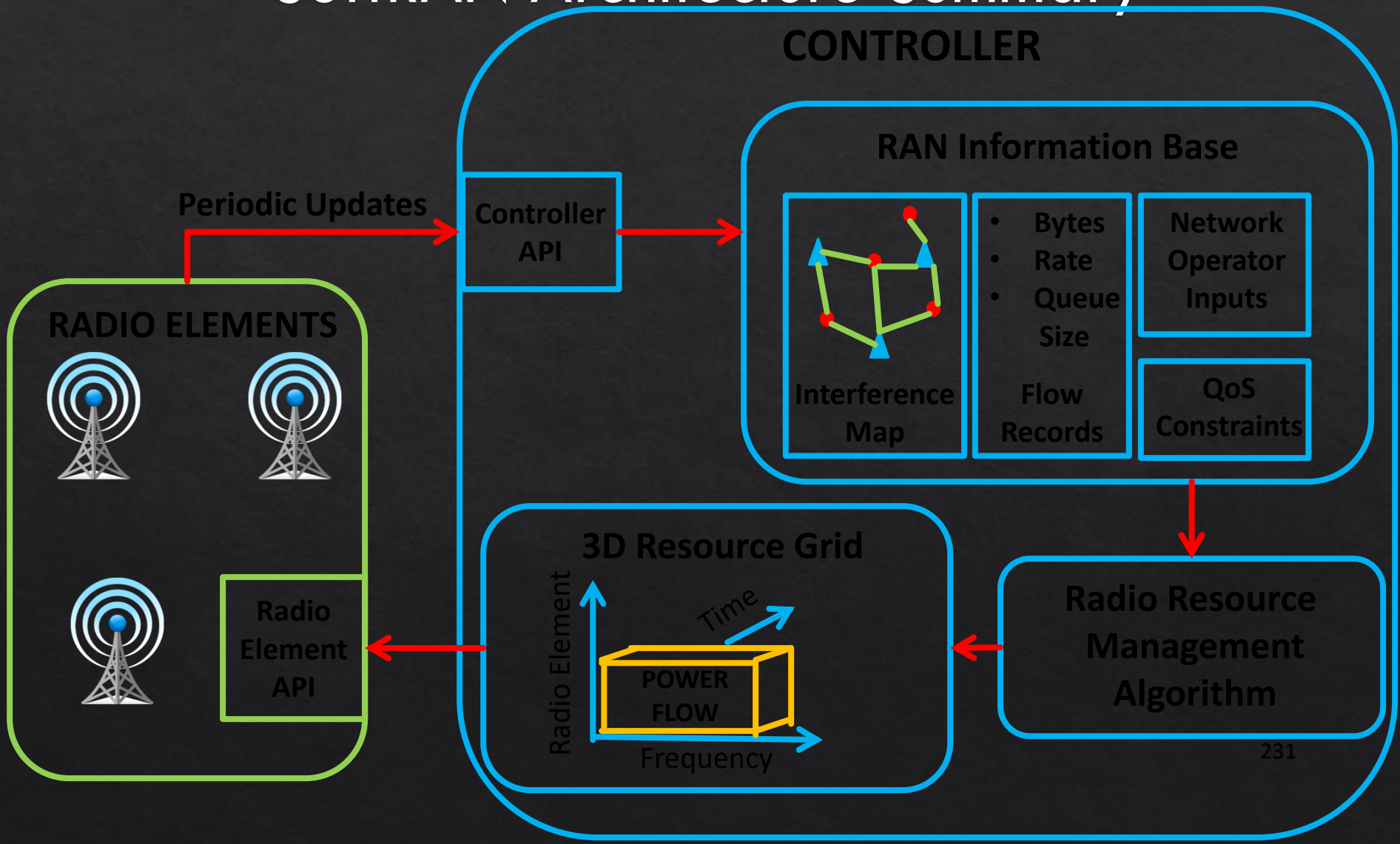
SoftRAN: SDN Approach to RAN



SoftRAN: SDN Approach to RAN



SoftRAN Architecture Summary



SoftRAN Architecture: Updates

- Radio element -> controller (updates)
  - Flow information (downlink and uplink)
  - Channel states (observed by clients)
- Network operator -> controller (inputs)
  - QoS requirements
  - Flow preferences

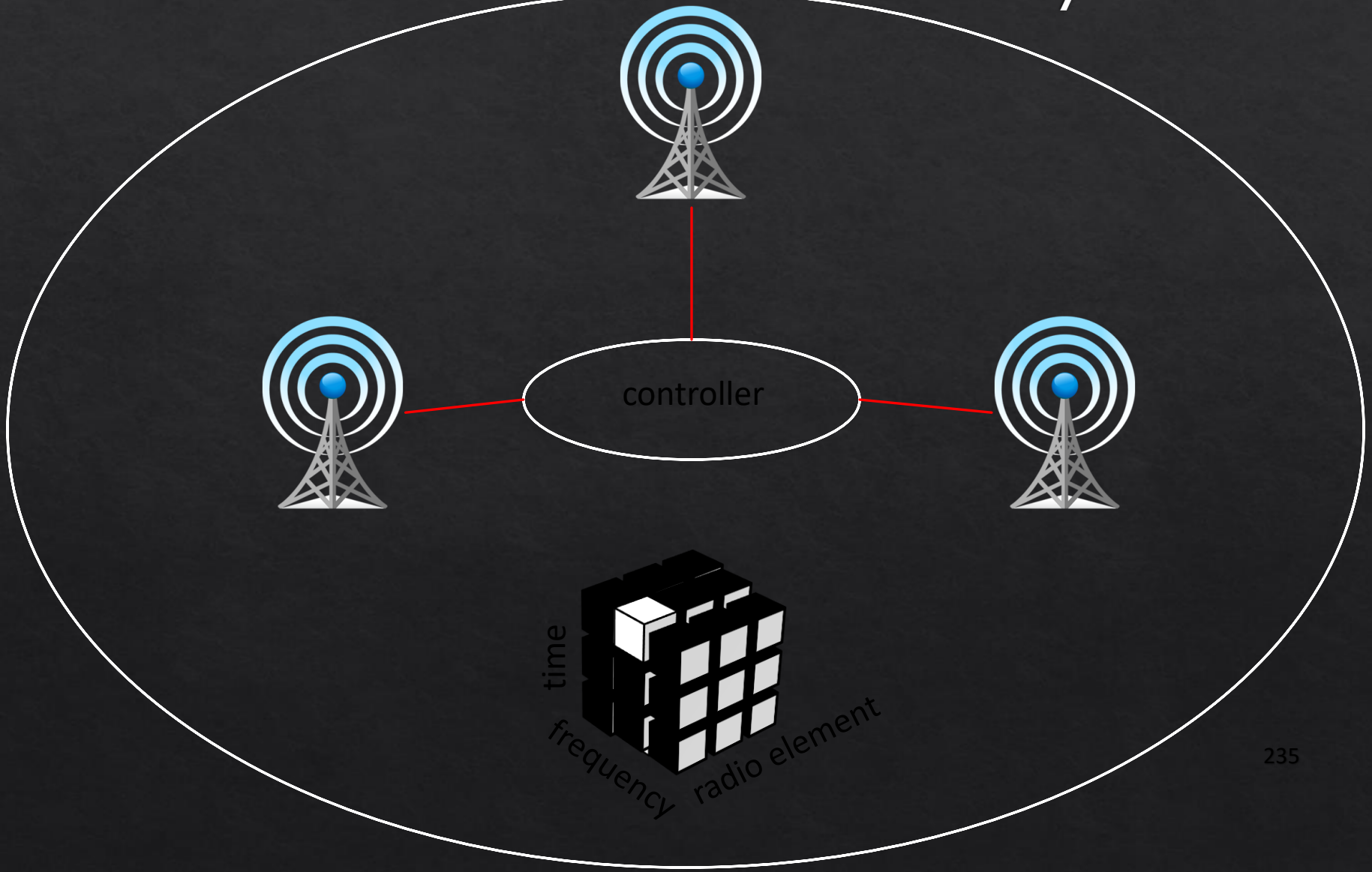
SoftRAN Architecture: Controller Design

- RAN information base (RIB)
  - Update and maintain global network view
    - Interference map
    - Flow records
- Radio resource management
  - Given global network view: maximize global utility
  - Determine RRM at each radio element

SoftRAN Architecture: Radio Element API

- Controller -> radio element
  - Handovers to be performed
  - RF configuration per resource block
    - Power allocation and flow allocation
  - Relevant information about neighboring radio elements
    - Transmit Power being used

SoftRAN: Backhaul Latency



Refactoring Control Plane

- Controller responsibilities:
  - Decisions influencing global network state
    - Load balancing
    - interference management
- Radio element responsibilities:
  - Decisions based on frequently varying local network state
    - Flow allocation based on channel states

SoftRAN Advantages

- Logically centralized control plane:
  - Global view on interference and load
    - Easier coordination of radio resource management
    - Efficient use of wireless resources
  - Plug-and-play control algorithms
    - Simplified network management
  - Smoother handovers
    - Better user-experience

SoftRAN: Evolving the RAN

- Switching off radio elements based on load
  - Energy savings
- Dynamically splitting the network into Big-BSs
  - Handover radio elements between Big-BSs

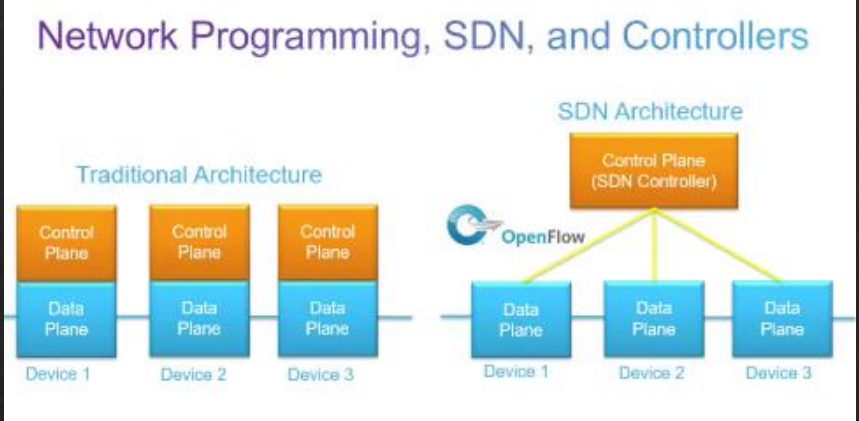
Implementation: Modifications

- SoftRAN is incrementally deployable with current infrastructure
  - No modification needed on client-side
  - API definitions at base station
    - Femto API : Standardized interface between scheduler and L1 (http://www.smallcellforum.org/resources-technical-papers)
    - Minimal modifications to FemtoAPI required

Implementation: Controller

- Floodlight : controller implementation
- Radio resource management algorithm
  - Load balancing
  - Interference management
  - QoS constraints
  - Network operator preferences

Network Programming, SDN, and Controllers



- Network Programming - used to program and virtualize the network
  - Protocols – OpenFlow, NETCONF, RESTCONF
  - Programming languages – C, C++, Java
  - Scripting languages – Python, Ruby, Lua
  - Data modeling languages – YANG
  - Markup Languages - XML
  - Architectures and APIs – REST, Java

- SDN - Software Defined Networking
  - Open Network Foundation
  - OpenFlow - a protocol that separates the control plane from the forwarding plane
  - OpenStack – Platform for Cloud Computing and providing IaaS

- Network Controllers – SDN control plane device which is a programmable point of automation to manage, configure, monitor, and troubleshoot virtual and physical network infrastructures. Automates the configuration of the network infrastructure.
  - Cisco ACI – ANP policy model, APIC-EM controller, and Nexus 9000 series switches

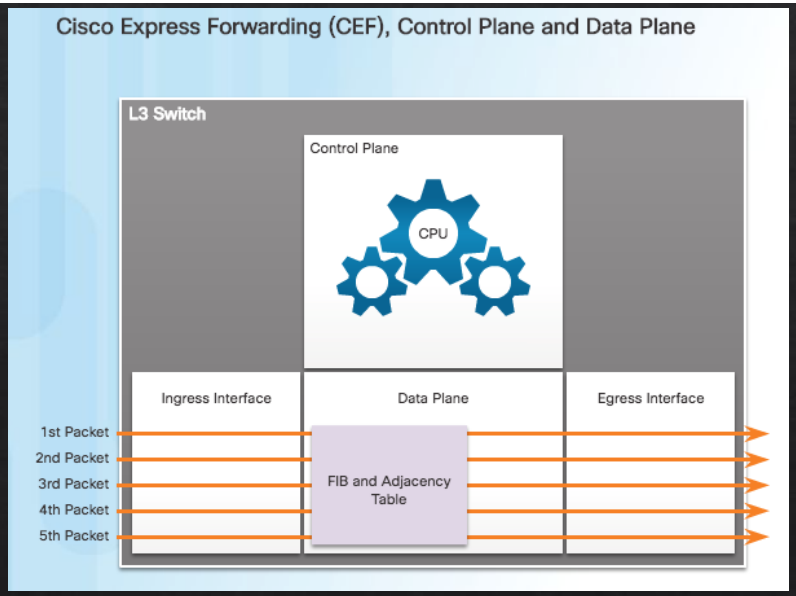
Software-Defined Networking

Control Plane and Data Plane

A network device contains the following planes:

Control plane - Regarded as the brains of a device. Used to make forwarding decisions. Information sent to the control plane is processed by the CPU.

Data plane - Also called the forwarding plane, this plane is the switch fabric connecting the various network ports on a device. The data plane of each device is used to forward traffic flows.



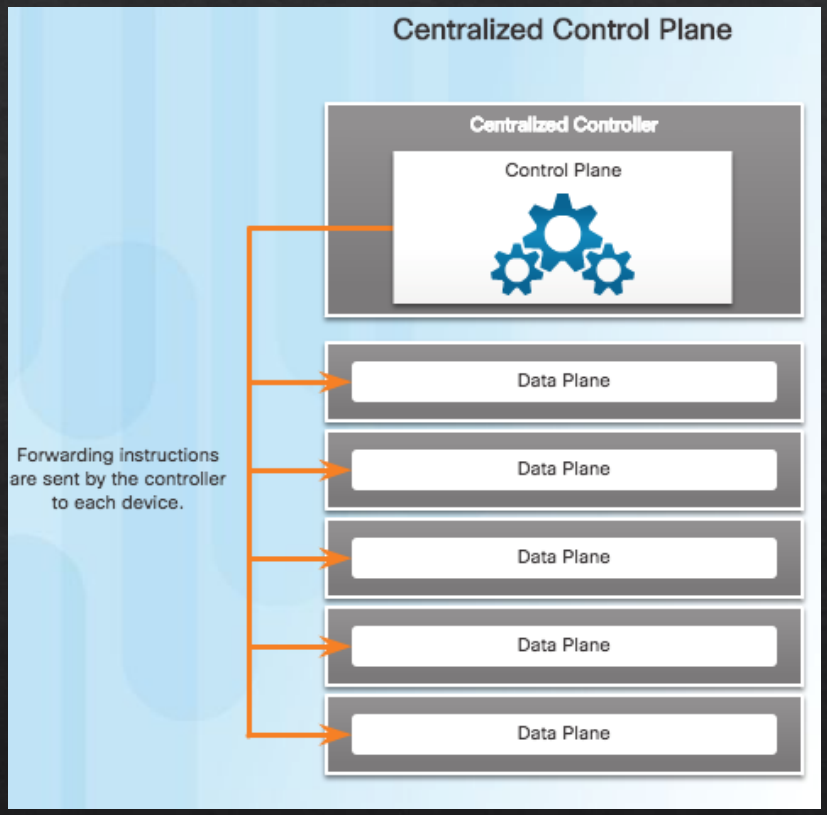
CEF is an advanced, Layer 3 IP switching technology that enables forwarding of packets to occur at the data plane without consulting the control plane. Packets are forwarded directly by the data plane based on the information contained in the Forwarding Information Base (FIB) and adjacency table, without needing to consult the information in the control plane.

Control Plane and Data Plane (Cont.)

To virtualize the network, the control plane function is removed from each device and is performed by a centralized controller.

The centralized controller communicates control plane functions to each device.

Each device can now focus on forwarding data while the centralized controller manages data flow, increases security, and provides other services.



Virtualizing the Network

Two major network architectures have been developed to support network virtualization:

- Software Defined Networking (SDN) - A network architecture that virtualizes the network.
- Cisco Application Centric Infrastructure (ACI) - A hardware solution for integrating cloud computing and data center management.

These are some other network virtualization technologies, some of which are included as components in SDN and ACI:

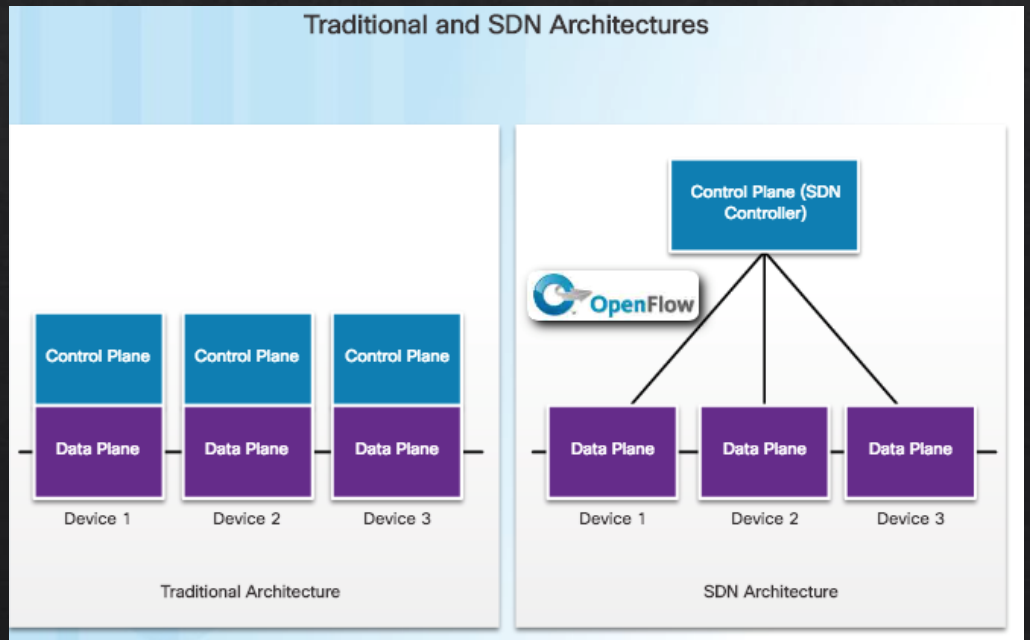
- OpenFlow - The OpenFlow protocol is a basic element in building SDN solutions.
- OpenStack - This approach is a virtualization and orchestration platform available to build scalable cloud e ovide an infrast a service (IaaS) solution.

Software-Defined Networking

SDN Architecture

In a traditional router or switch architecture, the control plane and data plane functions occur in the same device. Routing decisions and packet forwarding are the responsibility of the device operating system.

Software defined networking (SDN) is a network architecture that has been developed to virtualize the network. SDN can virtualize the control plane. SDN moves the control plane from each network device to a central network intelligence and policy-making entity called the SDN controller.



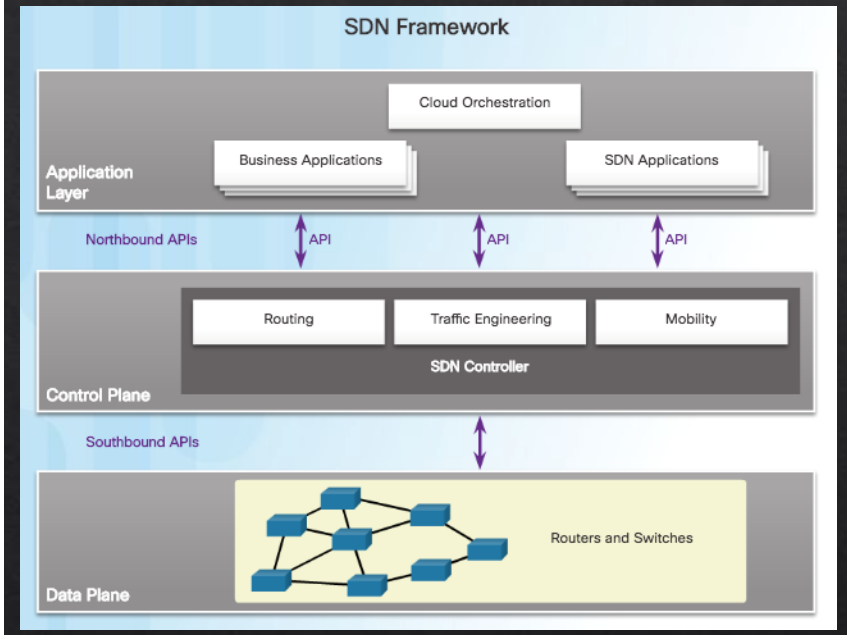
SDN Architecture (Cont.)

The SDN controller enables network administrators to manage and dictate how the data plane of virtual switches and routers should handle network traffic.

The SDN controller uses northbound APIs to communicate with the upstream applications. These APIs help network administrators shape traffic and deploy services.

The SDN controller also uses southbound APIs to define the behavior of the downstream virtual switches and routers.

An API is a set of standardized requests that define the proper way for an application to request services from another application.



- OpenFlow is the original and widely implemented southbound API.

Controllers

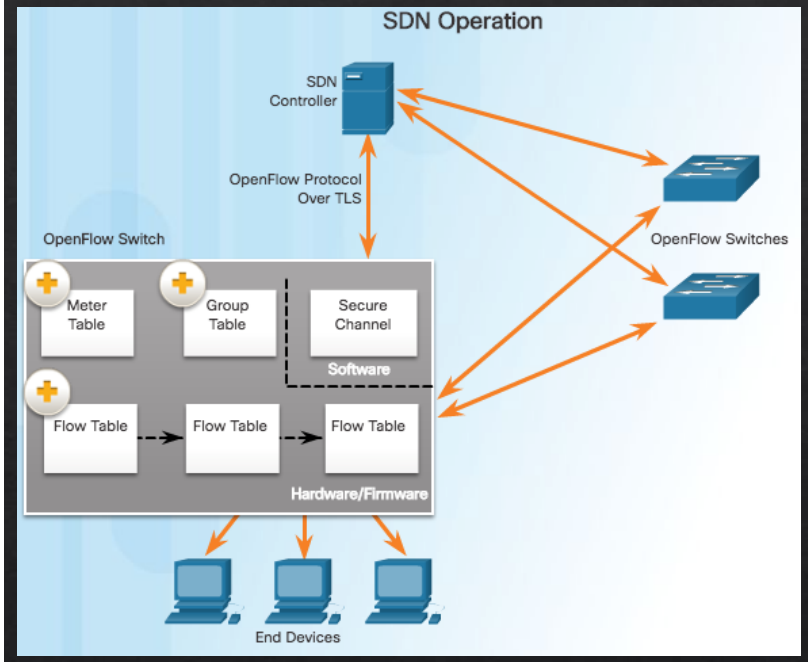
SDN Controller and Operations

SDN controller defines the data flows that occur in the SDN Data Plane. A flow could consist of all packets with the same source and destination IP addresses, or all packets with the same VLAN identifier.

Each flow traveling through the network must first get permission from the SDN controller. If the controller allows a flow, it computes a route for the flow to take and adds an entry for that flow in each of the switches along the path.

The controller populates and the switches manage the flow tables. Each OpenFlow switch connects to other OpenFlow switches. They can also connect to end-user devices that are part of a packet flow.

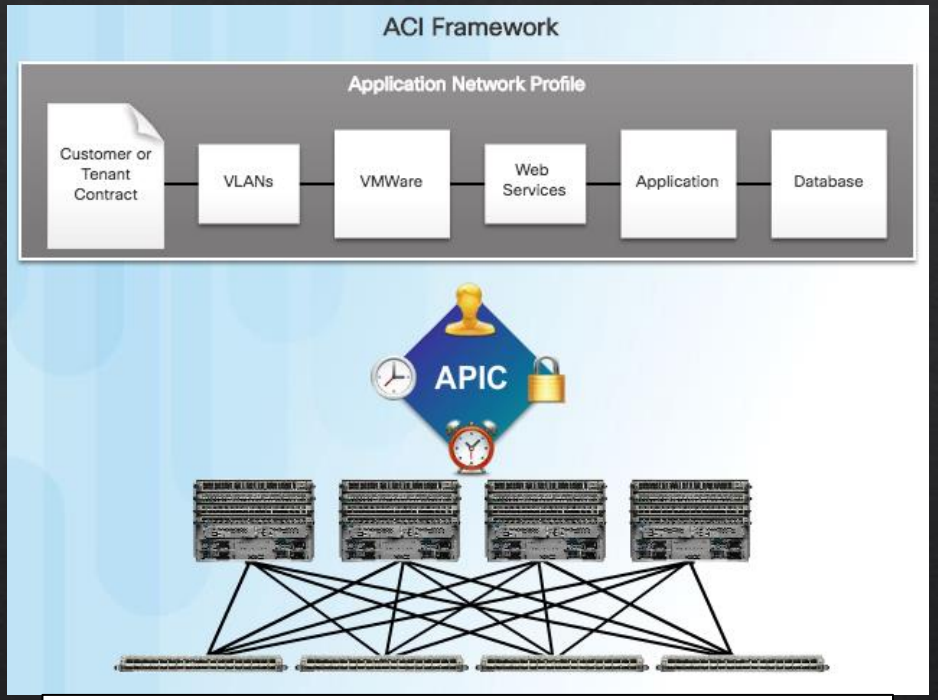
To the switch, a flow is a sequence of packets that matches a specific entry in a flow table.



Core Components of ACI

Three core components of the ACI architecture:

- Application Network Profile (ANP) - Collection of end-point groups (EPG), their connections, and the policies that define those connections.
- Application Policy Infrastructure Controller (APIC) – The brains of the ACI architecture. A centralized software controller that is designed for programmability and centralized management. Translates application policies into network programming.
- Cisco Nexus 9000 Series switches – Provide an application-aware switching fabric and work with an APIC to manage the virtual and physical network infrastructure.



APIC is positioned between the ANP and the ACI-enabled network infrastructure. The APIC translates the application requirements into a network configuration to meet those needs.

Controllers

Spine-Leaf Topology

Cisco ACI fabric is composed of the APIC and the Cisco Nexus 9000 series switches using two-tier spine-leaf topology, as shown in the figure.

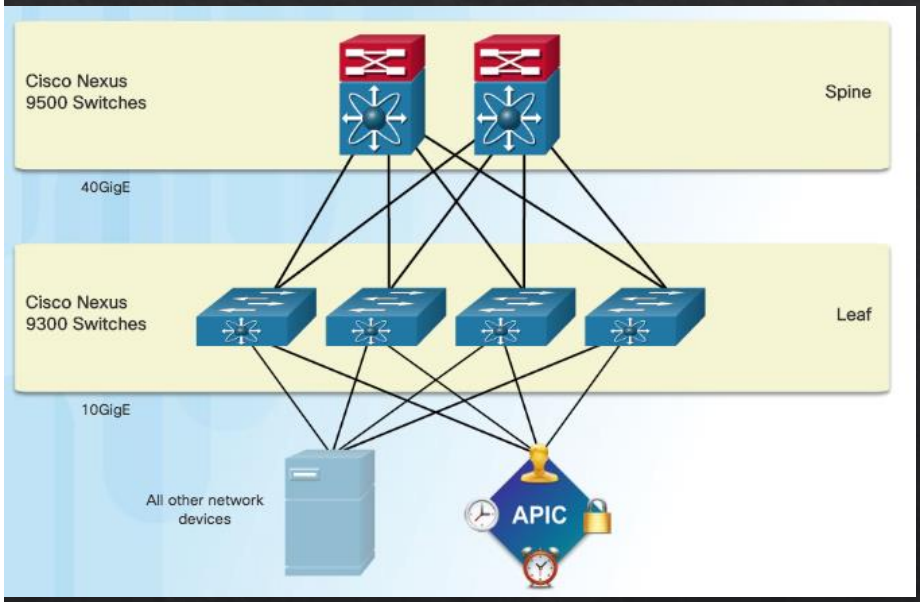
Leaf switches always attach to the spines, but they never attach to each other.

Spine switches only attach to the leaf and core switches (not shown).

Cisco APICs and all other devices in the network physically attach to leaf switches.

When compared to SDN, the APIC controller does not manipulate the data path directly.

The APIC centralizes the policy definition and programs the leaf switches to forward traffic based on the defined policies.



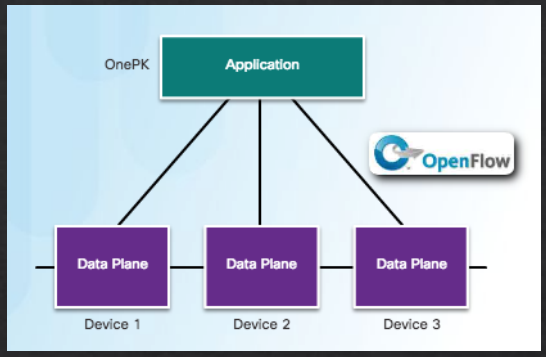
Controllers

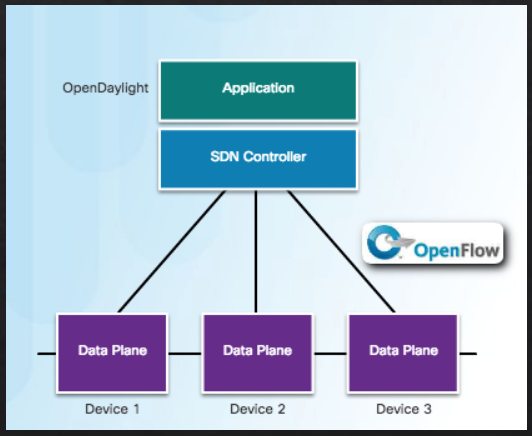
SDN Types

To better understand APIC-EM, it is helpful to take a broader look at the three types of SDN:

Device-based SDN - The devices are programmable by applications running on the device itself or on a server in the network. Cisco OnePK is an example of a device-based SDN. It enables programmers to build applications to integrate and interact with Cisco devices.

Controller-based SDN - Uses a centralized controller that has knowledge of all devices in the network. The applications can interface with the controller responsible for managing devices and manipulating traffic flows throughout the network. The Cisco Open SDN Controller is a commercial distribution of OpenDaylight.



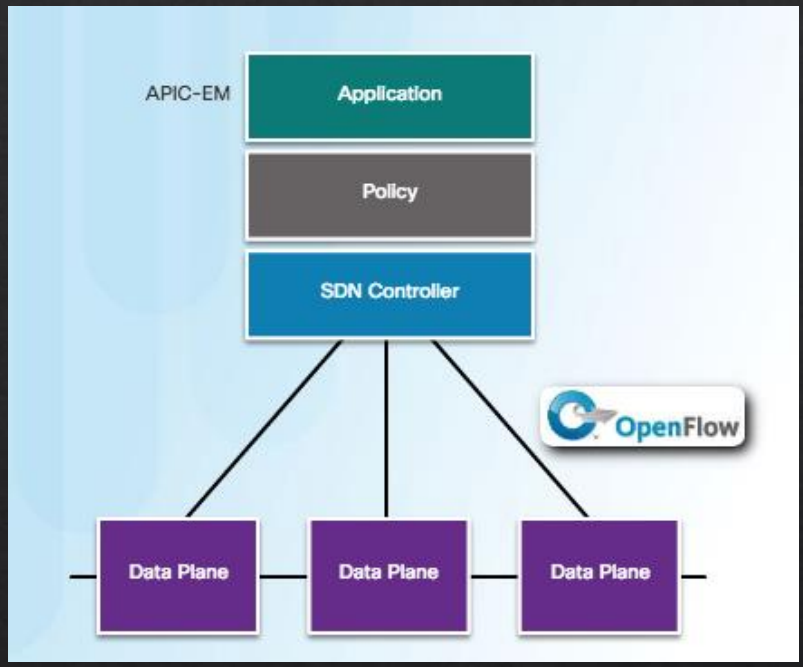


Controllers

SDN Types (Cont.)

Policy-based SDN - Similar to controller-based SDN where a centralized controller has a view of all devices in the network. Includes an additional Policy layer. Uses built-in applications that automate advanced configuration tasks via a guided workflow and user-friendly GUI. No programming skills are required. Cisco APIC-EM is an example of this type of SDN.

Policy-based SDN is the most robust, providing for a simple mechanism to control and manage policies across the entire network.



Controllers

APIC-EM Features

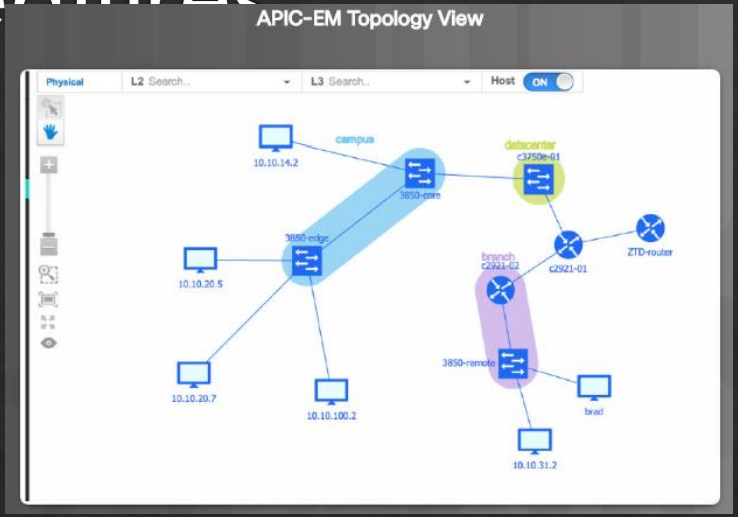
Cisco APIC-EM provides the following features:

Discovery - Supports a discovery functionality that is used to populate the controller's device and host inventory database.

Device Inventory - Collects detailed information from devices within the network including device name, device status, MAC address, IPv4/IPv6 addresses, IOS/Firmware, platform, up time, and configuration.

Host Inventory - Collects detailed information from hosts with the network including host name, user ID, MAC address, IPv4/IPv6 addresses, and network attachment point.

Policy - Ability to view and control policies across the entire network including QoS.



Topology - Supports a graphical view of the network (topology view).

Policy Analysis - Inspection and analysis of network access control policies. Ability to trace application specific paths between end devices to quickly identify ACLs in use and problem areas.

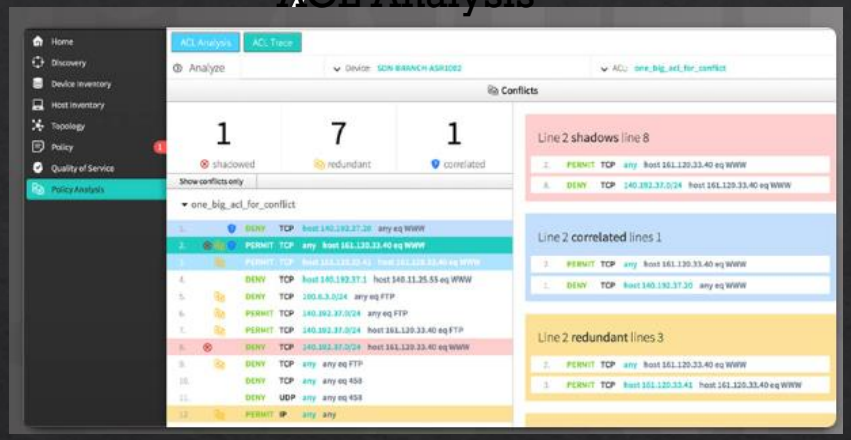
APIC-EM ACL Analysis

One of the most important features of the APIC-EM controller is the ability to manage policies across the entire network.

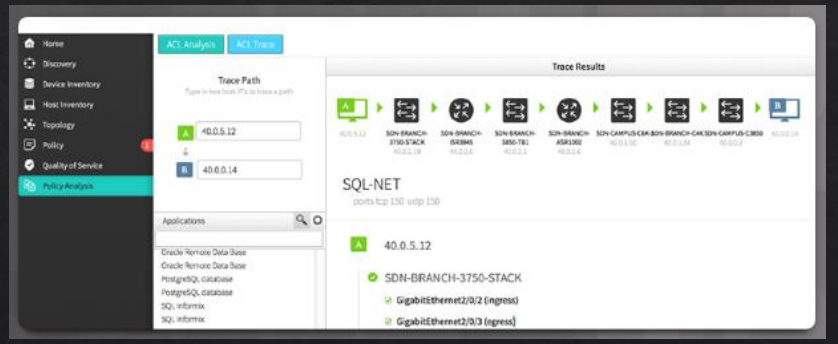
APIC-EM ACL Analysis and Path Trace provide tools to allow the administrator to analyze and understand ACL policies and configurations.

- ACL Analysis Tool - Enables ACL inspection and interrogation across the entire network, exposing any problems and conflicts.
- ACL Path Trace - This tool examines specific ACLs on the path between two end nodes, displaying any potential issues.

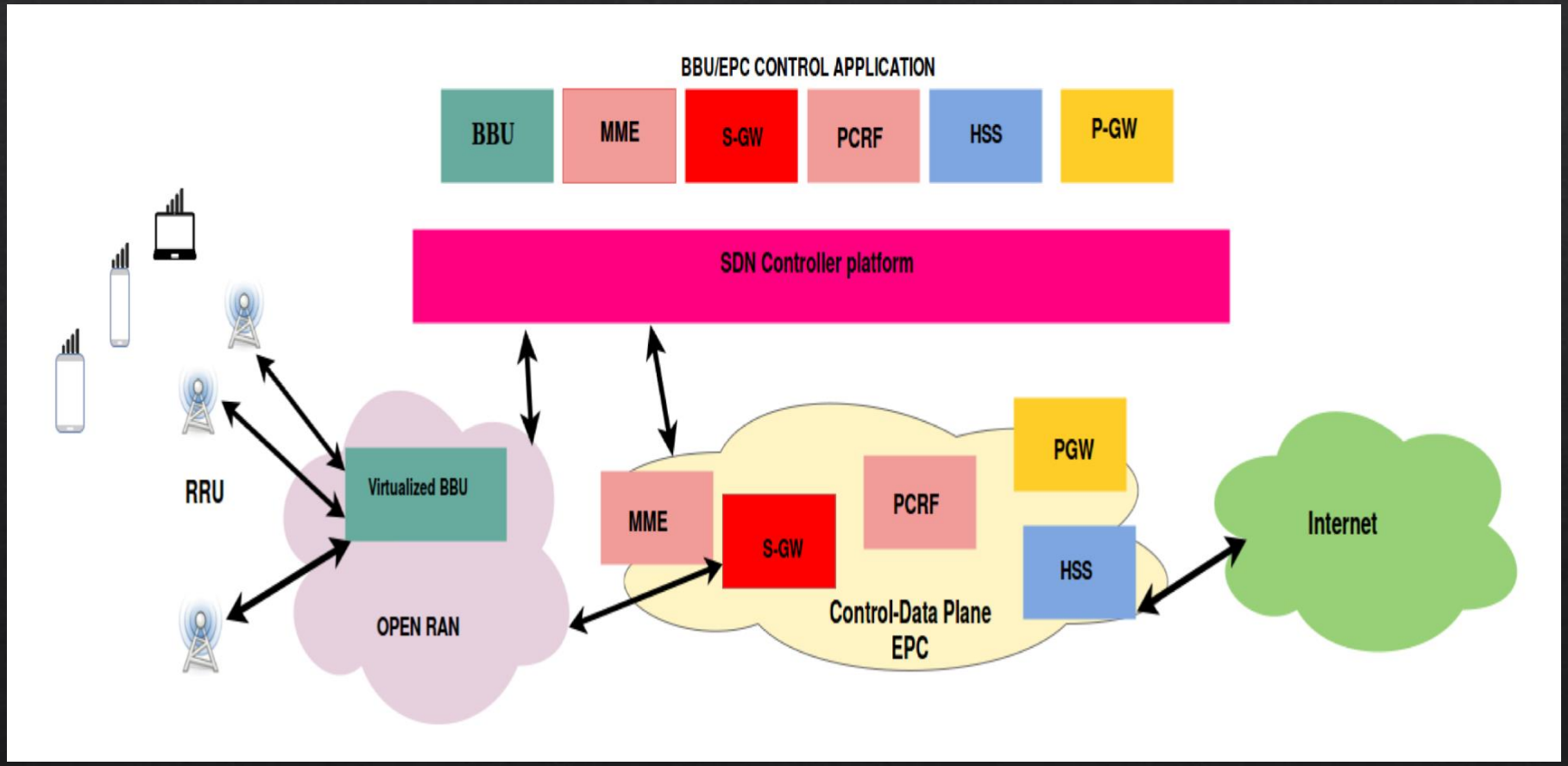
ACL Analysis



ACL Path Trace



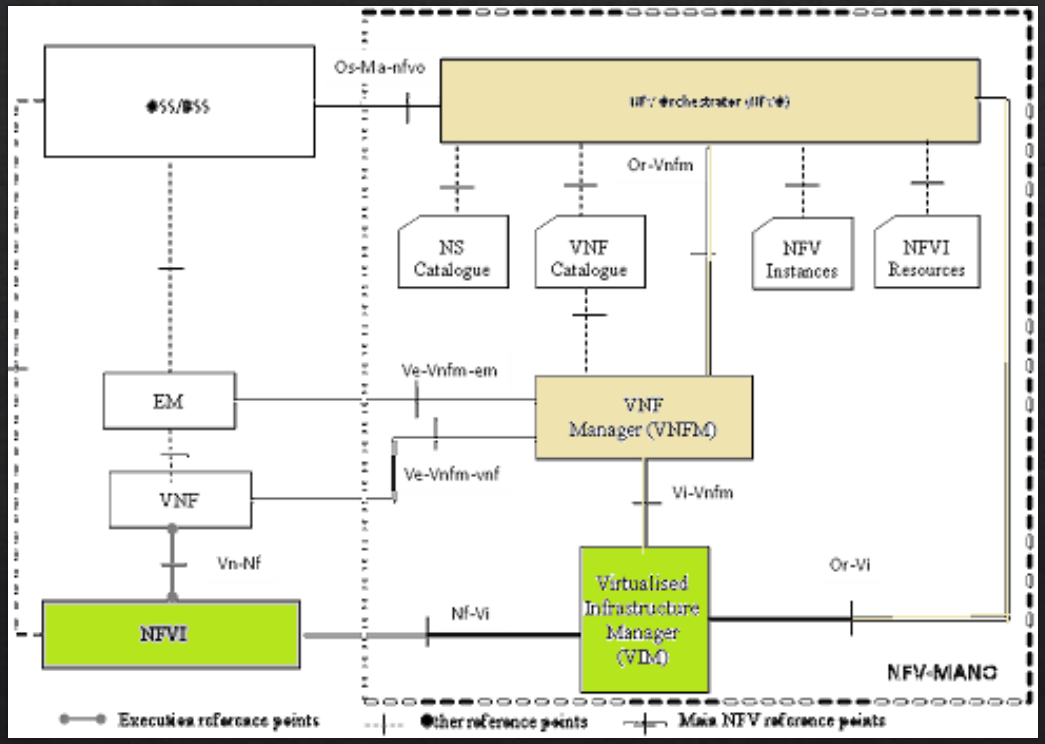
Mobile Network management – SDN approach



Policy and Charging Rules Function (PCRF)

ETSI NFV MANO Standards and OPNFV MANO push up the stack

- Network Functions Virtualization (NFV); Management and Orchestration (ETSI GS NFV-MAN 001 V1.1.1 (2014-12)).
- OSS/BSS needs NFVO for Network Services Orchestration & Management
- NFVO : O&M of Network Service Descriptor (NSD)
- VNFM : O&M of VNF Descriptor (VNFD)
- VIM : O&M NFVI(POPs) and Resources (VM,VN.VS)
- Other FCAPS for Service
- Priority 1 to come up with common minimum for Generic VNFM and flexibility to allow NFVO to also directly Orchestrate and Manage for NFVI through Or-Vi
- RAS in another aspect besides FCAPS to be noted.

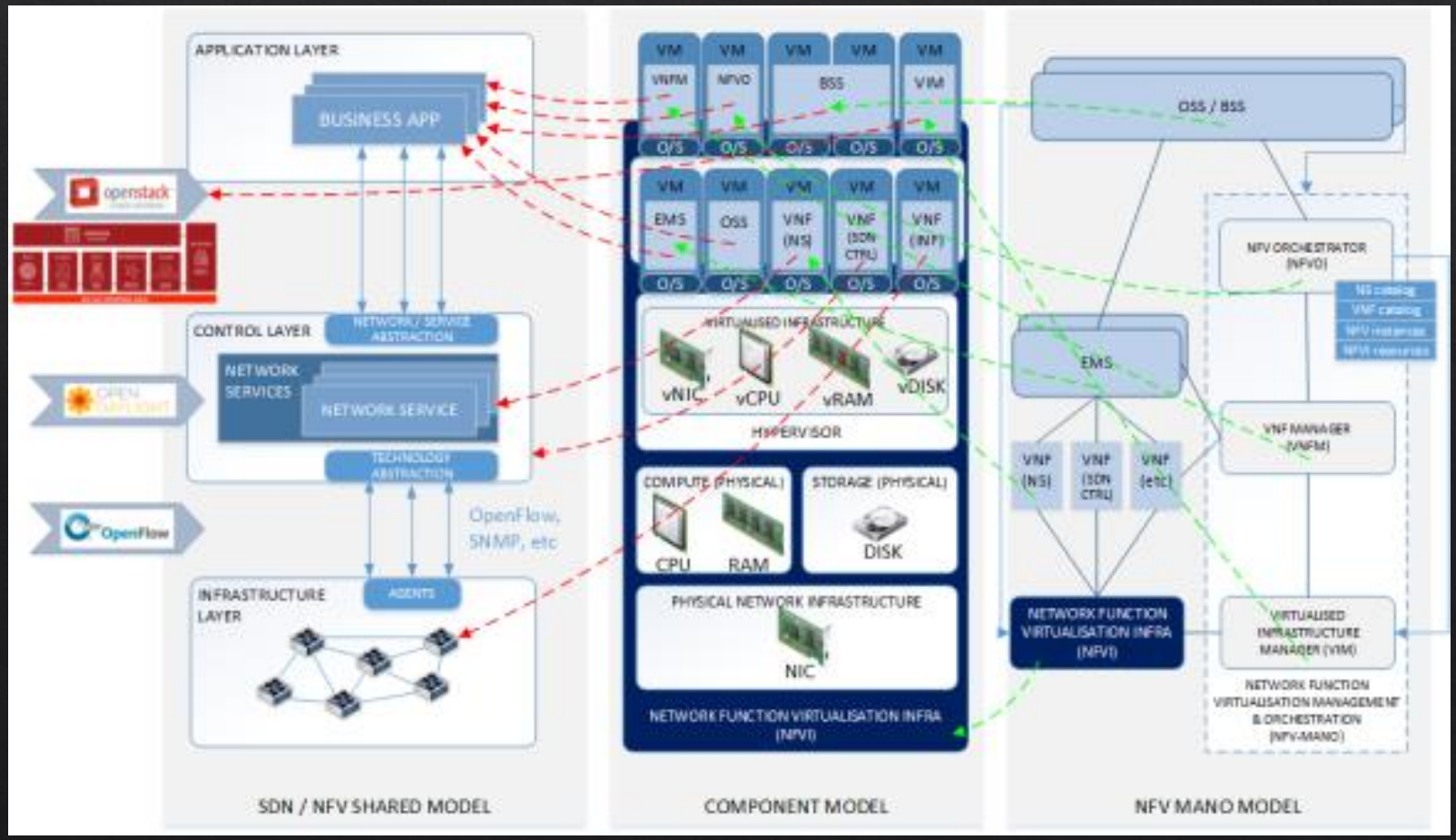


The three panels of the triptych are as follows:

SDN / NFV Shared Model (Left) – This is basically the shared model developed by ETSI and ONF to describe how the elements of their respective standards work together.

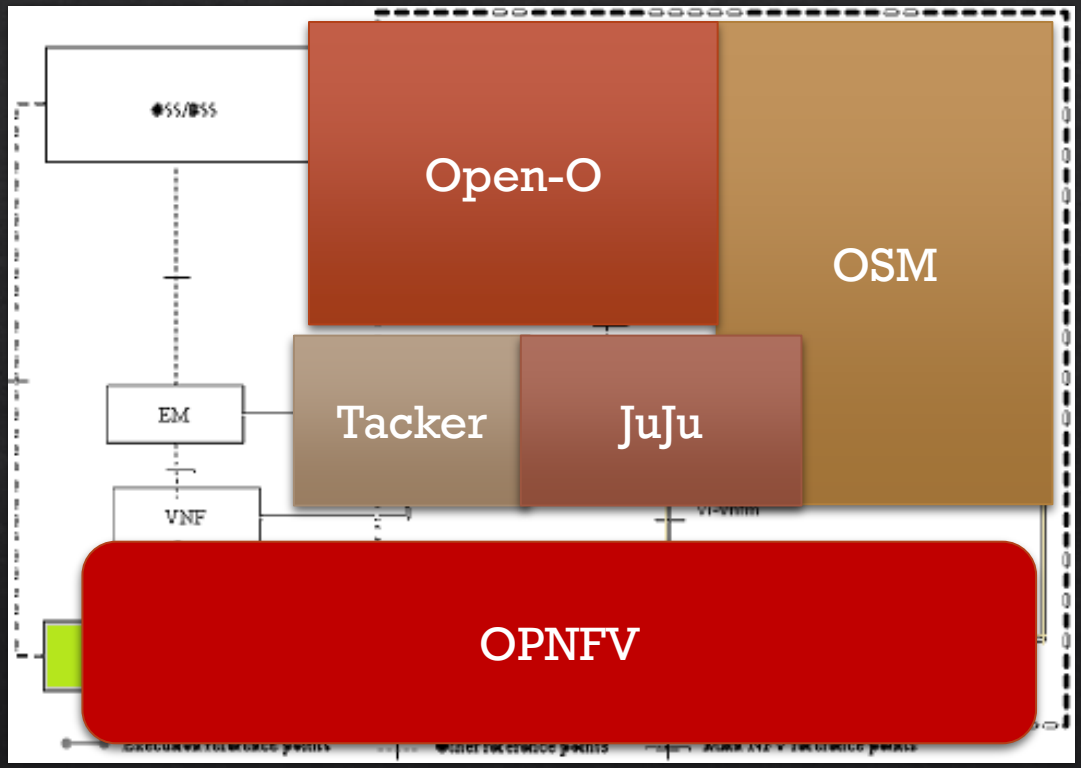
NFV MANO Model (Right) – This is a slightly re-worked version of ETSI's MANO (Management and Orchestration) topology

The component Model (Middle) – This is an attempt at trying to visualise the physical and virtual components that deliver upon the two other topologies / visions from ONF and ETSI

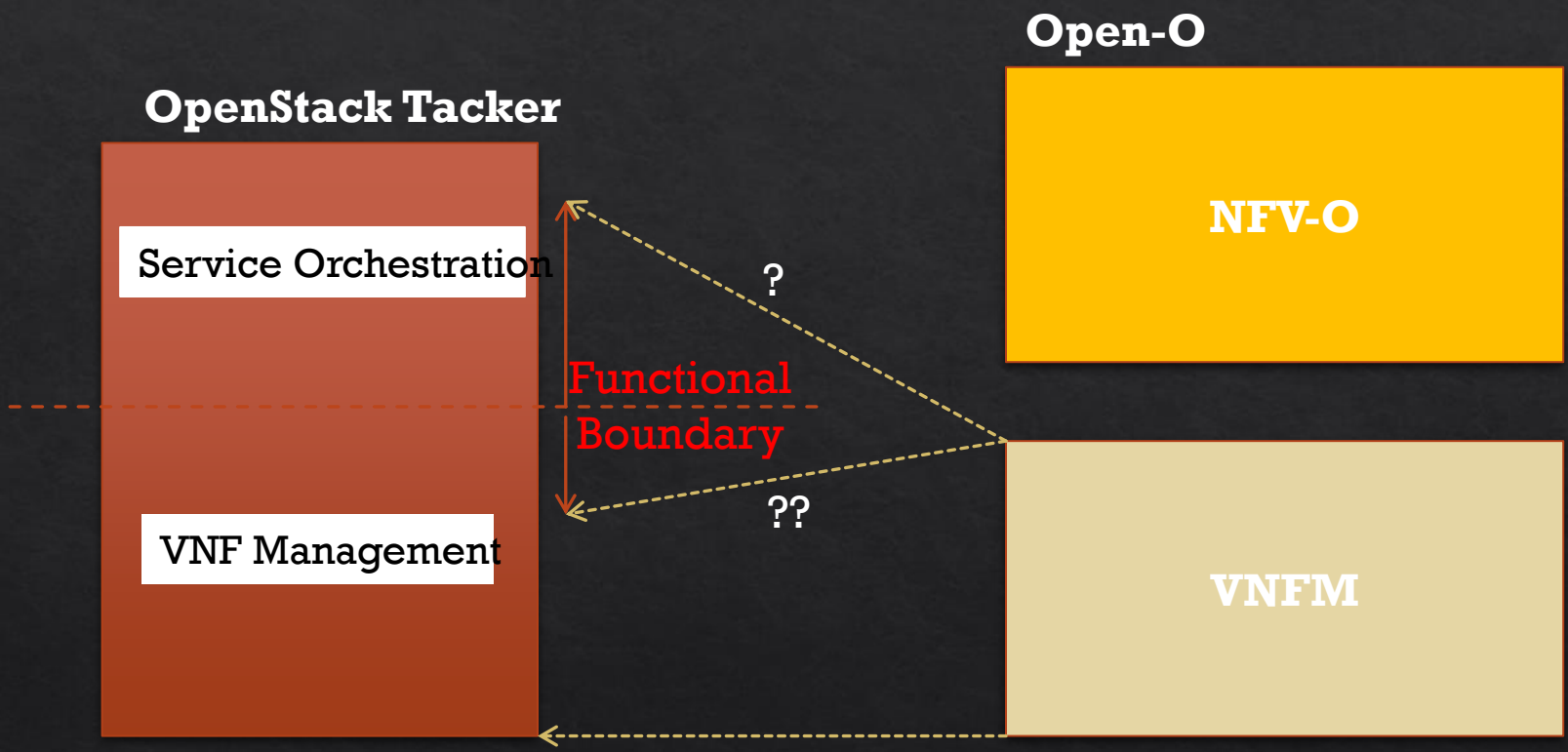


Challenge: Multiple Implementations

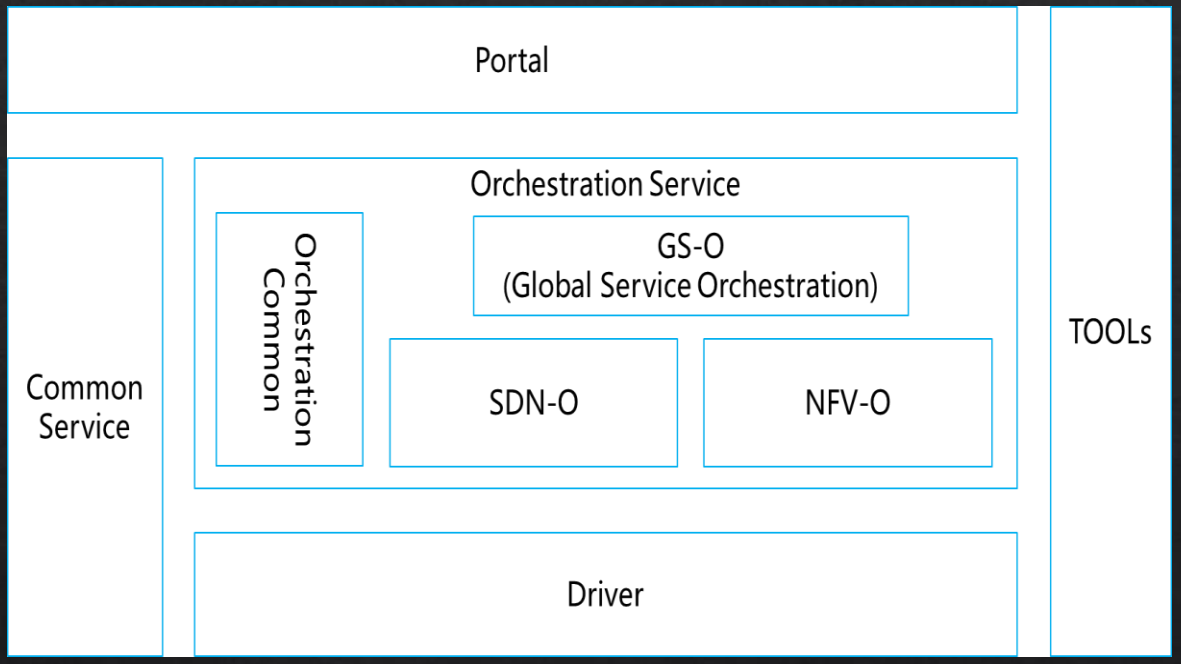
- Several Open Source MANO projects
  - Open-O
  - Open Source MANO (OSM)
  - Tacker
  - Juju
  - Etc.
- Many differences
  - Scope, data models, interfaces, etc.
- OPNFV as reference platform for NFV
  - Common integration point to help with consistency
  - Drive alignment on data models, events, etc.



Pain Point Example:
Open-O views Tacker as VNFM, but is it just a VNFM?

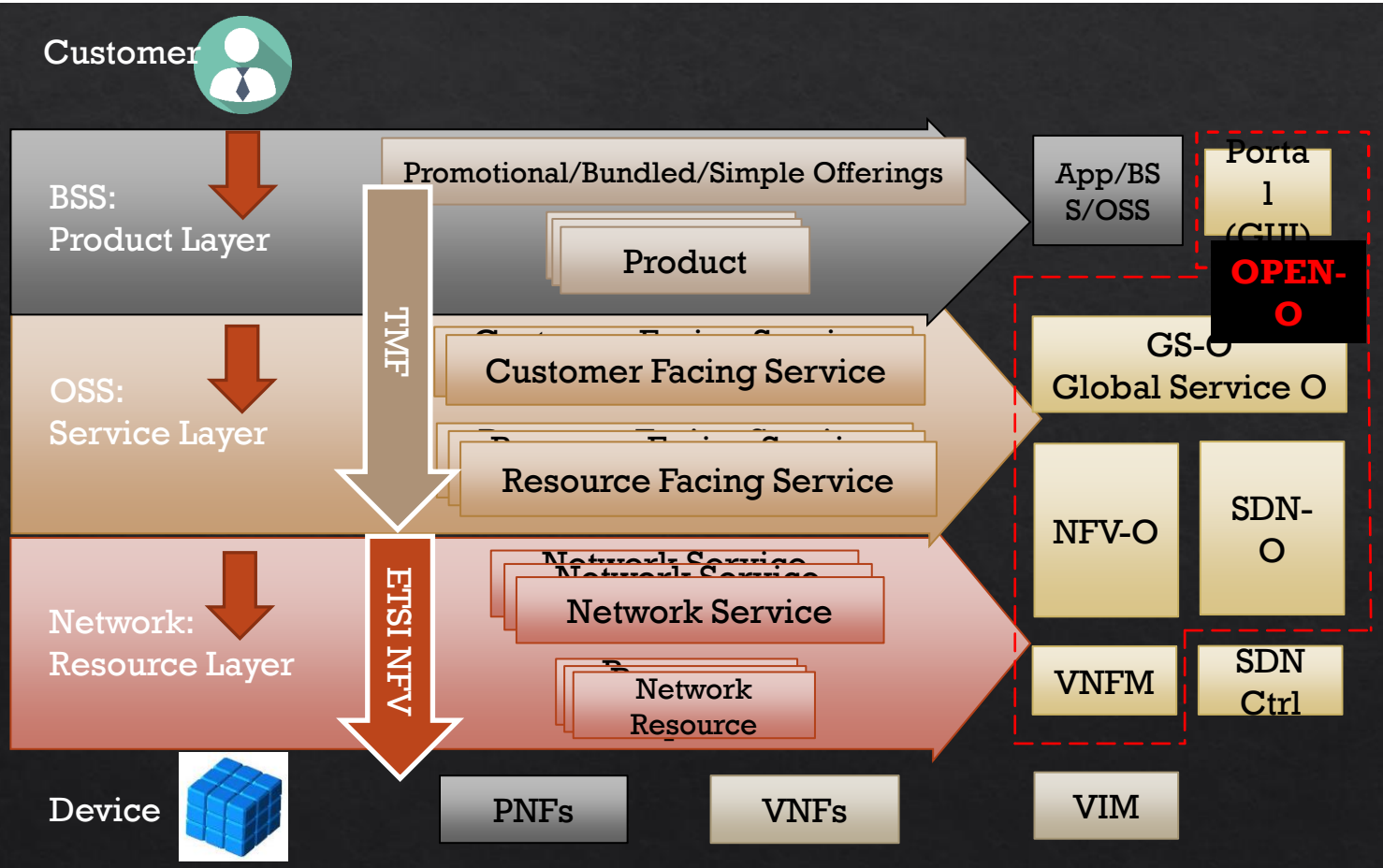


Introducing Open-O



- *Orchestration Service* consists of GS-O/SDN-O/NFV-O and Orchestration Common. GS-O coordinates orchestration across domains.
- Open-O connects to SDN controllers/VFNM/VIM/EMS through *Drivers.*
- *Common Service* provides platform service(Messaging etc.)
- *Tools* consist of model designer &VNF deployment tool.
- *Portal* provides GUI mgmt. interface for Open-O.

Where OPEN-O fits in Carrier network



Biz Layer: Transformation to APP ecosystem.

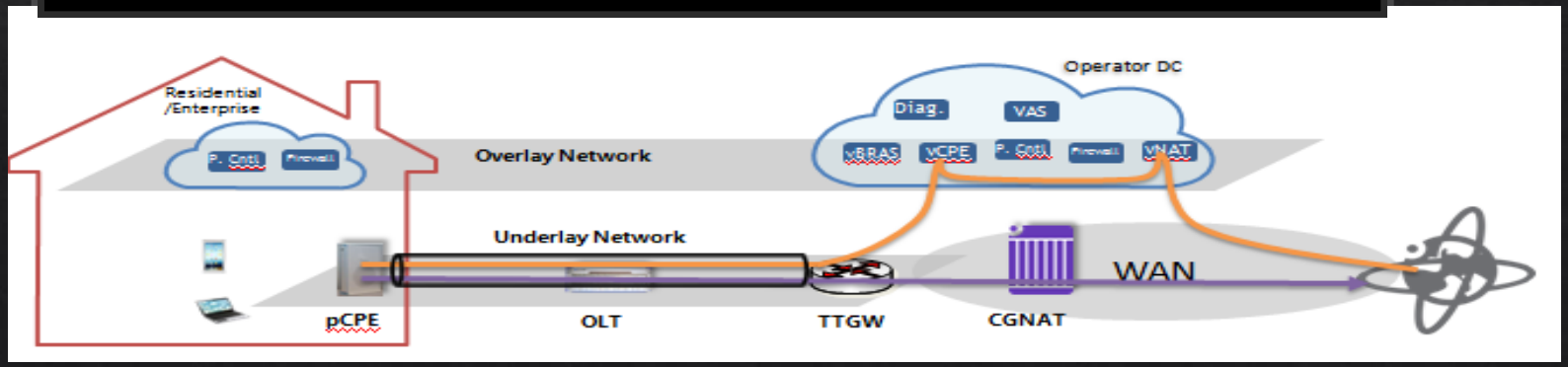
Operation Layer: Orchestration takes key role.

Network Layer: Virtualized resources make agility happen.

Modeling Use Case: virtual CPE/CE

Catalogue

Overlay NS	Internet Access Service 500M, with Firewall and NAT
Overlay NS (Enterprise)	Internet Access Service 500M with Firewall and Policy Cntl
Overlay NS (DC)	DC Service Chain with vCPE and vNAT
Underlay NS	Underlay Connectivity (complex, multiple CoSs, hierarchical)



Virtual CPE/CE Use Case (One Approach)

Design Time
Catalogue

Global-Service-O (GS-O)

Overlay NS

Overlay NS Topology/Graph

NFV-O (Enterprise)

Overlay NS (Enterprise)

Underlay NS DC Topology/Graph

Firewall

Policy Control

Enterprise Transport

SDN-O

Underlay NS

Underlay NS Topology/Graph

EMS1: pCE Controller 1: Connectivity 1

Controller 2: Connectivity n

EMS2: TTGW

Transport

NFV-O (DC)

Overlay NS (DC)

Underlay NS DC Topology/Graph

vCPE

vNAT

DC Transport

Enterprise VIM

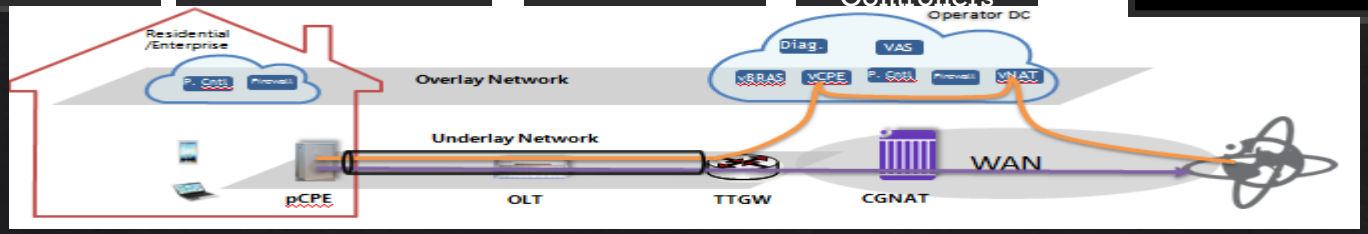
Enterprise SDN Controller

EMSs

SDN Controllers

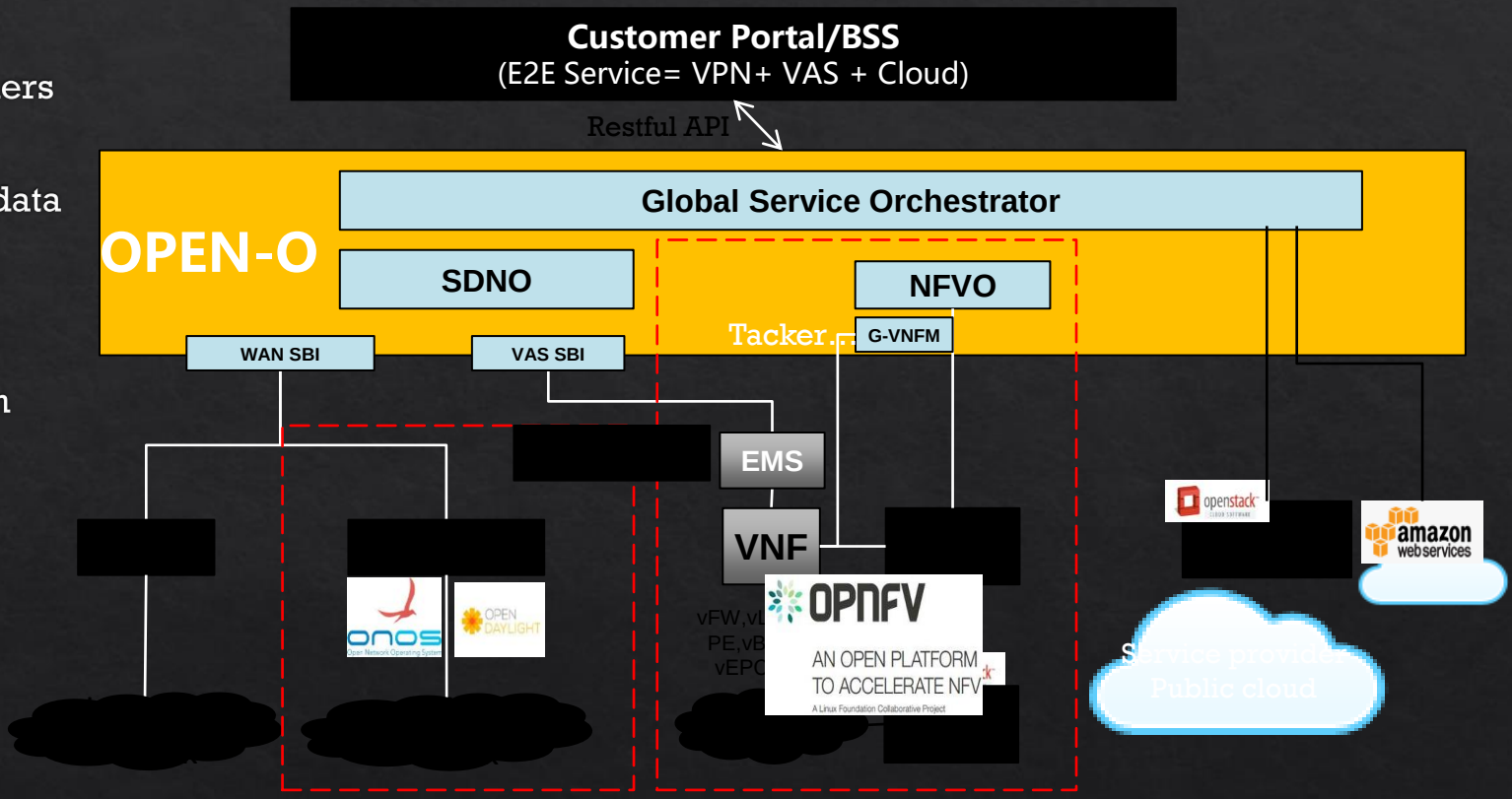
DC VIM

DC SDN Controller

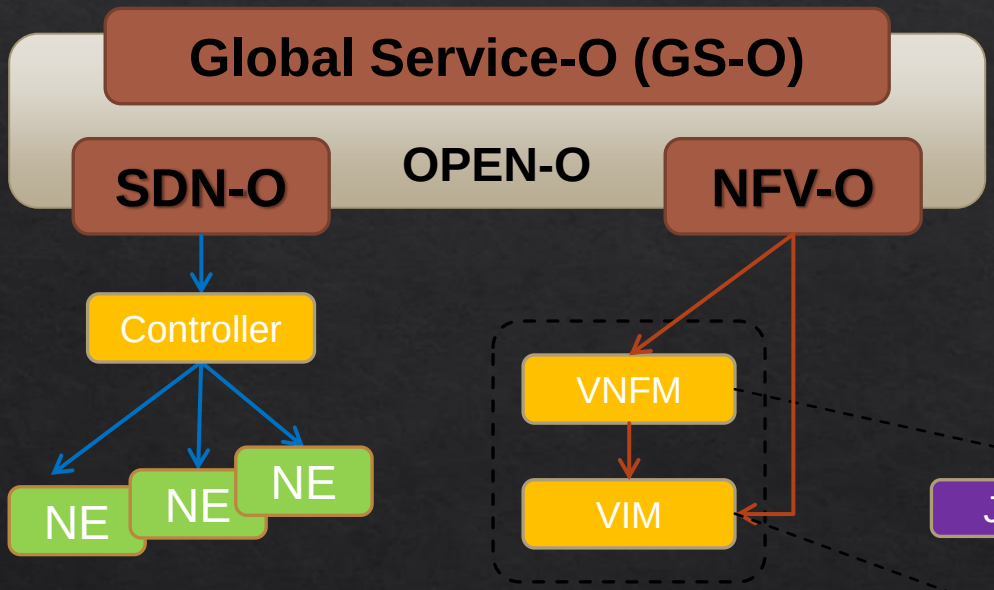


Relationship with other standard bodies

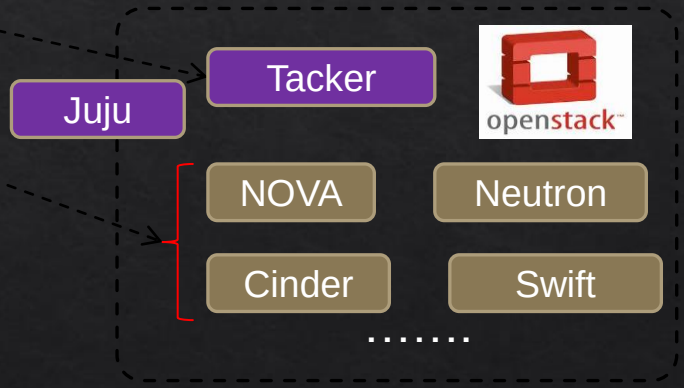
- Open-O wants to integrate with others
- ETSI NFV ISG – architecture and data models
- Open Source projects - OPNFV, Open Stack, Open Daylight, ONOS



Relationship with OPNFV



- Customer facing services are provided by Open-O
- OPNFV provides *VIM* services
- Both OPNFV and Open-O integrate VNFM services



Juju or Openstack Tacker provides VNF Manager functions

- Catalog
- Basic life-cycle of VNF
- Performance and Health monitoring
- Auto Healing
- Configuration

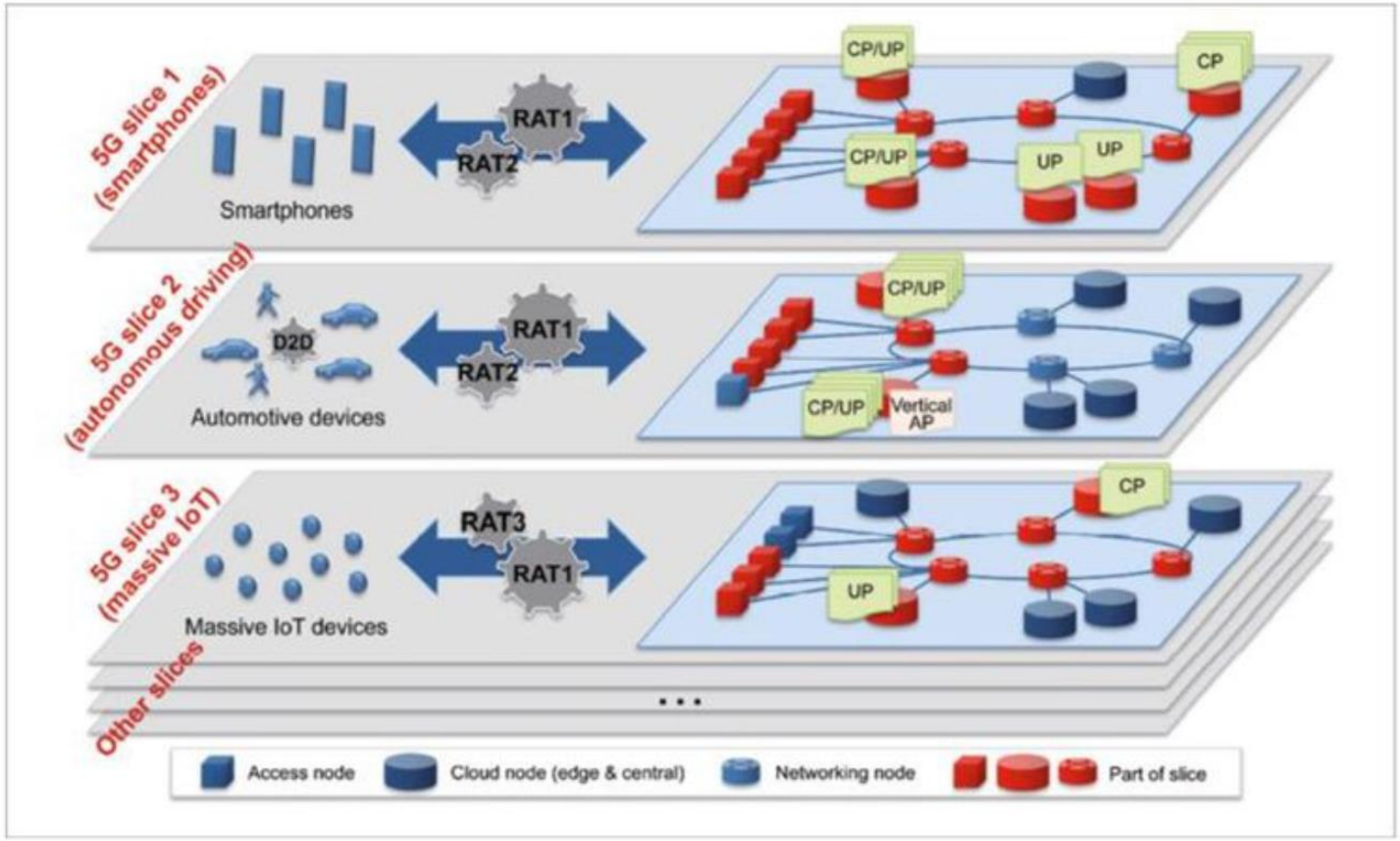
OPNFV/Openstack Nova/Neutron/etc provides VIM service

Relationship with other OPNFV Projects

- **Domino** in OPNFV handles template distribution and can help contribute any-to-any delivery of templates for Orchestration for top down implementation for OPNFV-MANO integration.
- **Models** in OPNFV defines various Models and MOVIE is another. If Tacker & Juju can work with them on use cases from NFVO (Open-O/OSM), there is possibility of encouraging them to contribute.
- **VNF Event Streaming (VES)** has a VNF event Collector and VES standardization of notifications is one of the key aspects that can be absorbed by OPNFV OPEN-O integration efforts.
- **Parser** defines libraries for Tosca translations for VNFD and NSD to Yang, HOT, and possible others used by different controllers and orchestrators.
- **Doctor** is a project whose key feature is immediate notification of unavailability of virtualized resources from VIM, to process recovery of VNFs on them. This may need to add Service Assurance for VM & VNF recovery along with some standardization efforts done by Doctor team independently. Doctors team and MANO workgroup to work out the details if there is any common agreement on this.
- **Promise** is a project whose key feature is Virtual Resource reservation and allocation, and thus it may need Service Assurance to reserve, allocate and manage resources on-behalf of NOFVO and again this along RS may depend on documents available to communicate between teams to get to a common ground.

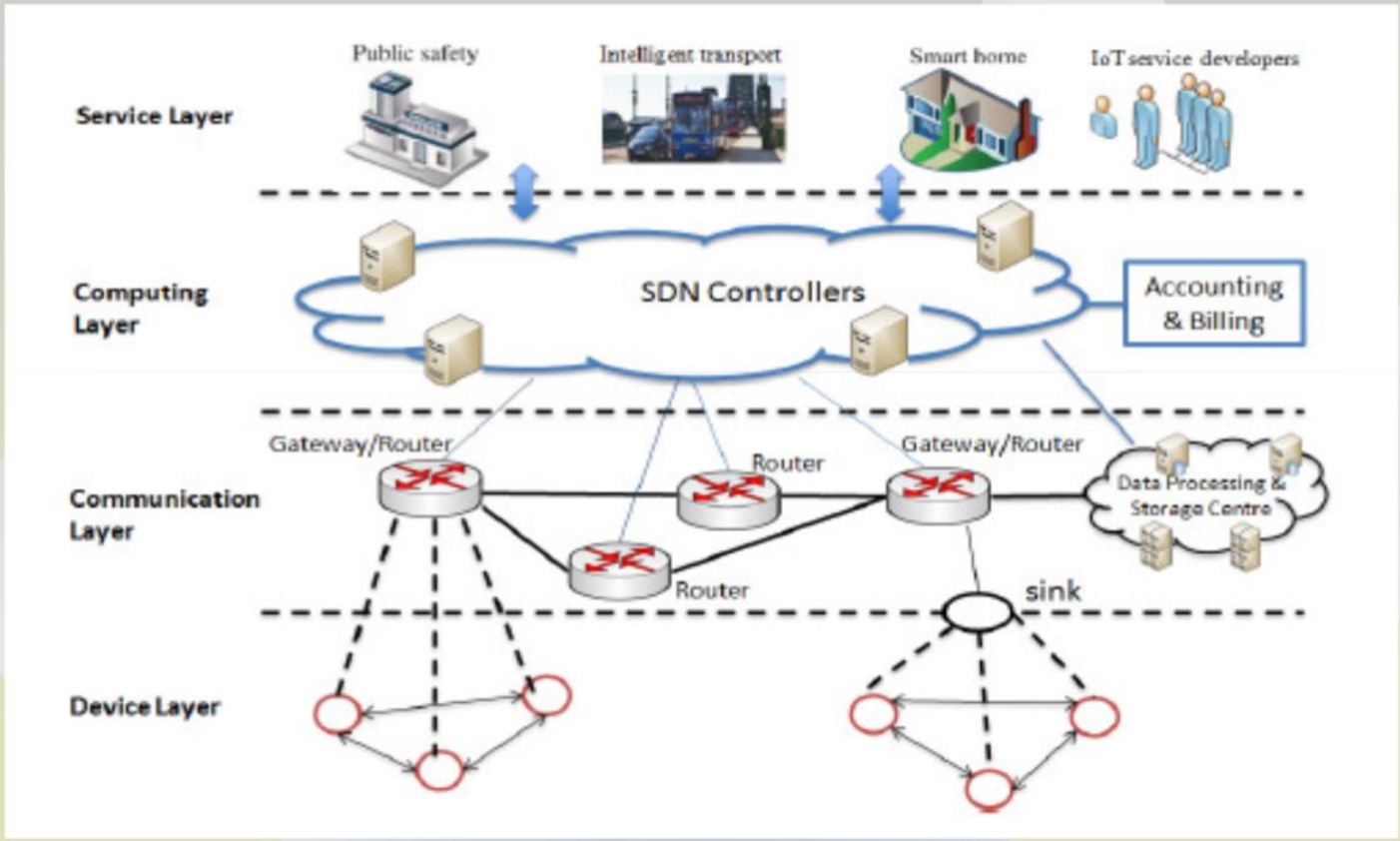
SDN & IoT

Network Slicing and IoT



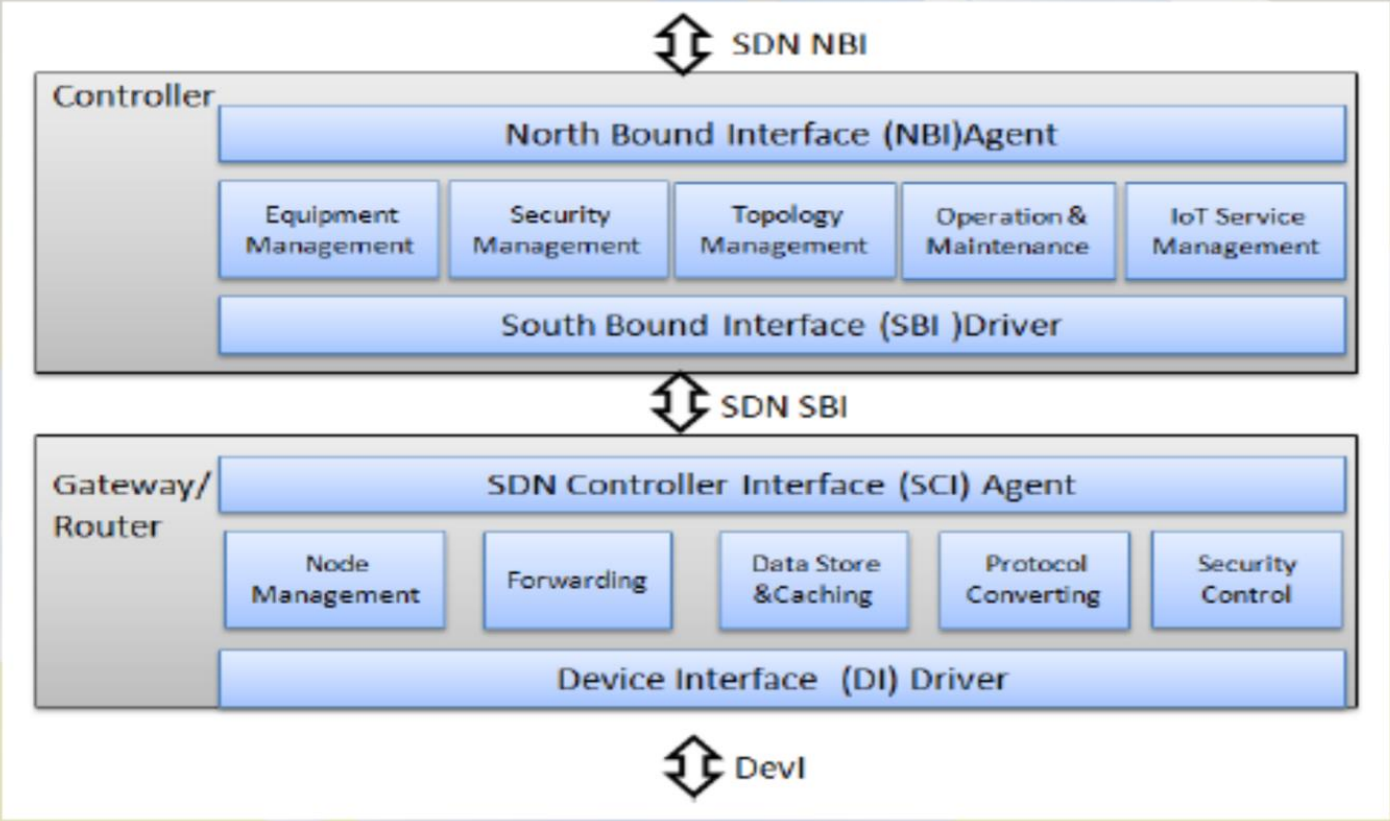
Convergence SDN - IoT

- **SDN control of IoT- example 1**



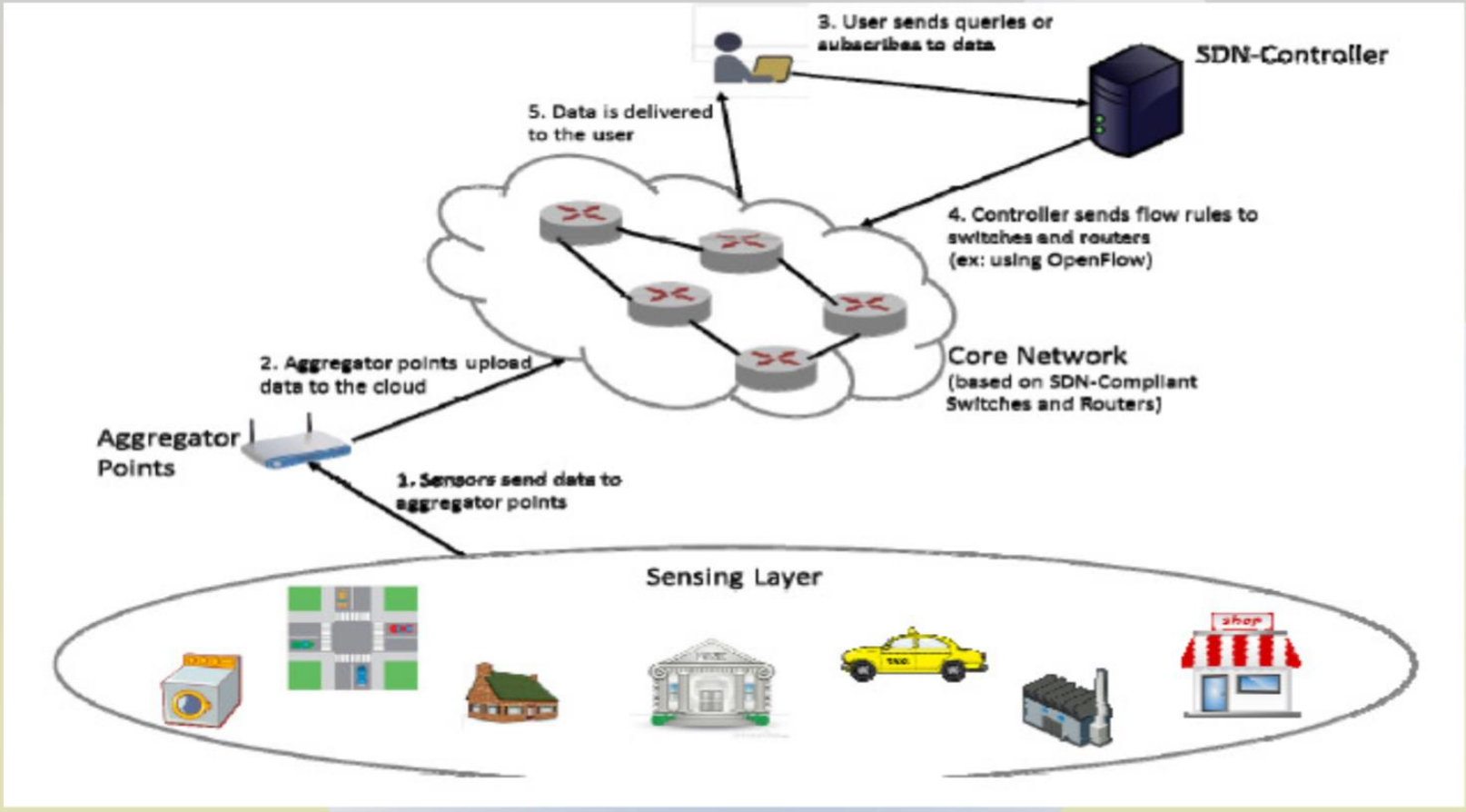
SDN-IoT

- **SDN control of IoT- example 1 (cont'd)**
- **Functional modules of the controller and gateways**



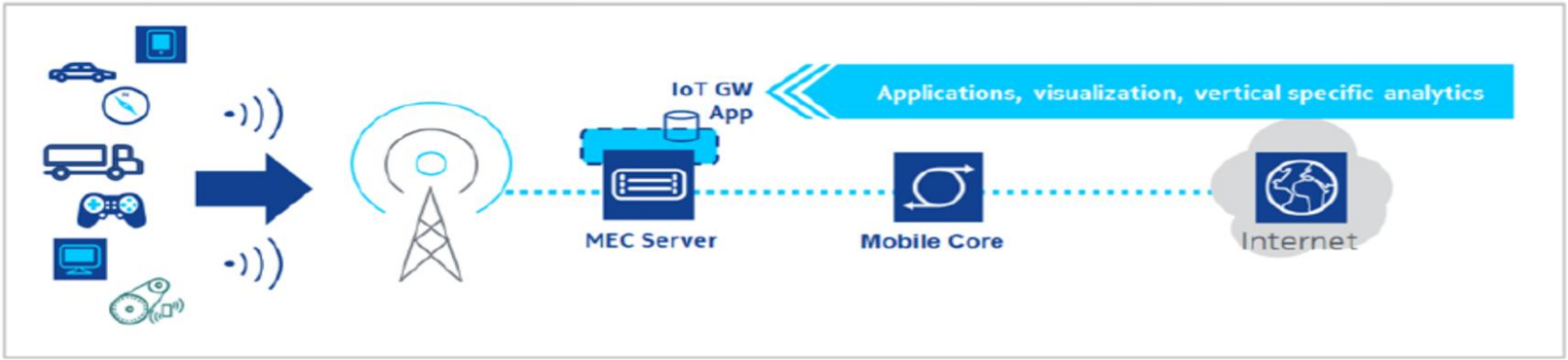
SDN-IoT

- **SDN control of IoT- example 2 (ICN-style architecture)**

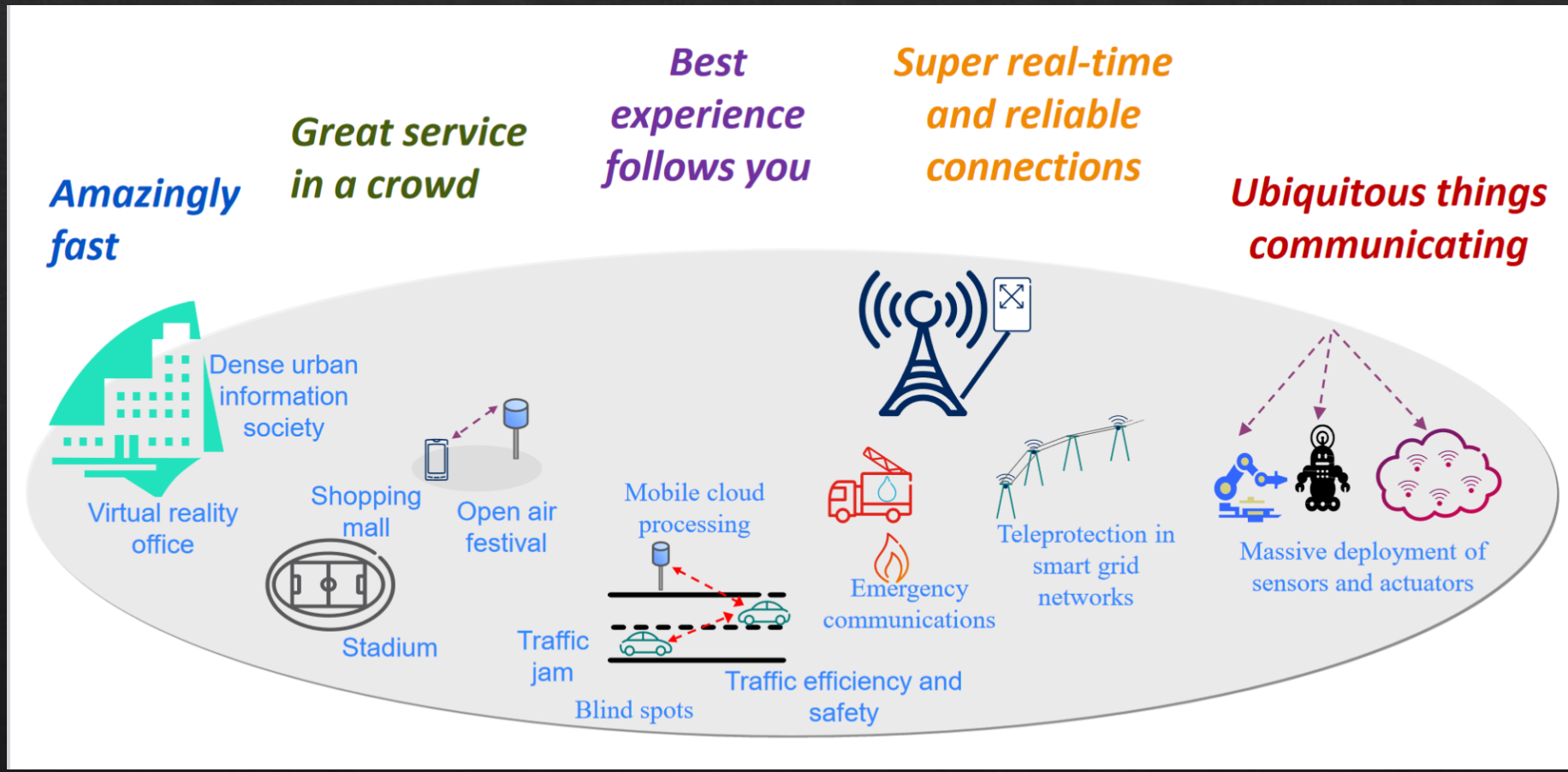


SDN-MEC-IoT

- **MEC Use Cases example- IoT**
- Internet of Things (IoT)
  - IoT devices: Often limited (processor, memory capacity)→ need for messages aggregation , security , low latency ..
  - r.t. capability → grouping of sensors and devices is needed for efficient service.
  - Possible Solutions:
    - IoT manipulated close to the devices (e.g., MEC server)
    - This also provides an analytics processing capability and a low latency response time.

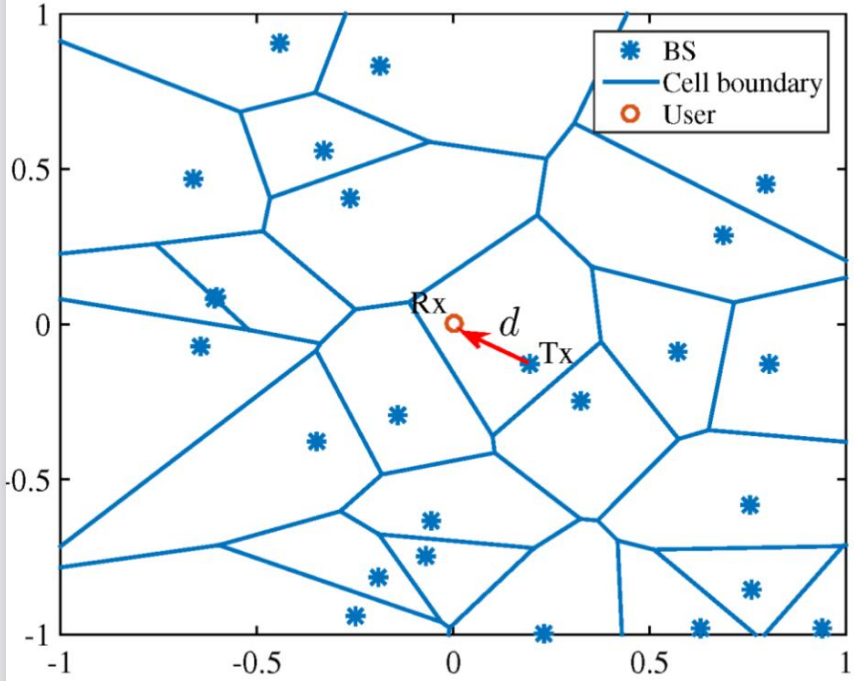


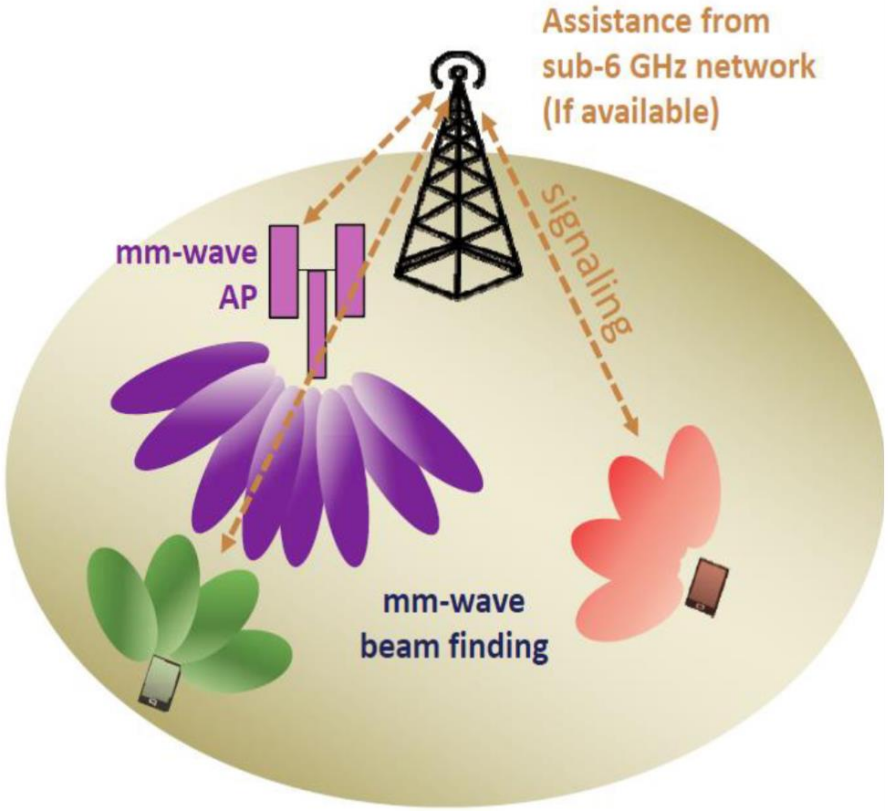
SDN & 5G


Amazingly fast
Great service in a crowd
Best experience follows you
Super real-time and reliable connections
Ubiquitous things communicating
Dense urban information society
Virtual reality office
Shopping mall
Open air festival
Stadium
Mobile cloud processing
Traffic jam
Blind spots
Traffic efficiency and safety
Emergency communications
Teleprotection in smart grid networks
Massive deployment of sensors and actuators

Convergence SDN-mmWave-5G

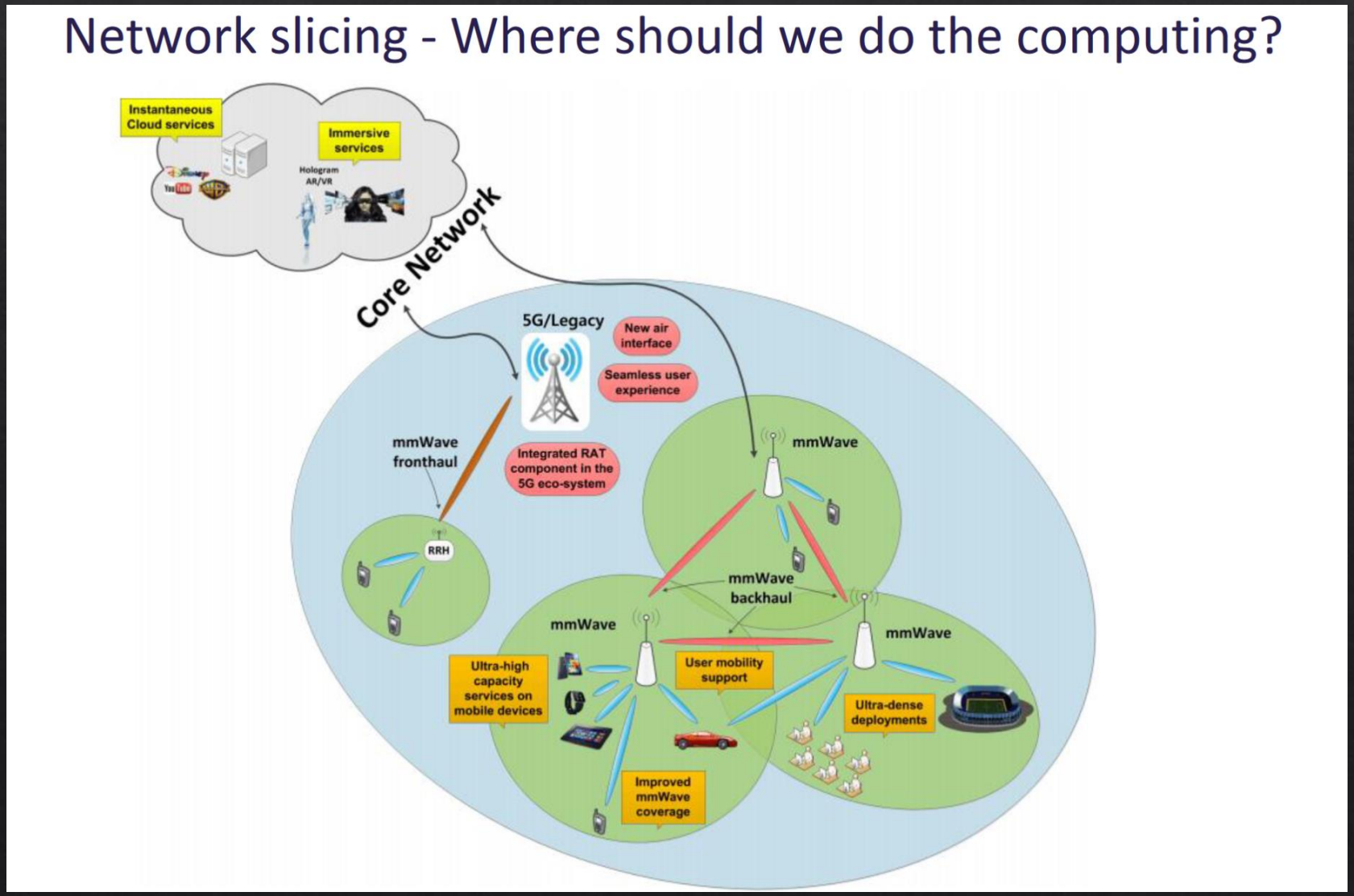
From hexagonal cells to unstructured beam spaces





Source: mmMAGIC WP4 presentation, ETSI workshop, Sophia-Antipolis, Jan 28, 2016

Network slicing in converged networks



Integrated Moving networks

Challenges and Opportunities with Demanding Verticals "Integrated Moving Networks"

- **Mutual benefits!**
- **Better mobile systems efficiency**: Vehicles collect side information to improve the resource allocation and performance of the mobile network
- **More reliable V2X links**: Connect non-vehicular users to the Traffic Safety/Traffic Efficiency protocols (Pedestrians, cyclists, pets, …)
- **New disruptive business opportunities:** exploiting vehicle sensed data

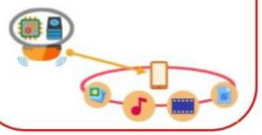
mmWAVE MEC

- Main research directions:
  - Focus on the ultra High-Speed and Low Latency Communications (uHSLLC) use cases and related technology enablers
  - Synergize between mmWave and MEC technologies

Data rate	Connectivity	E2E latency	CAPEX/OPEX
mmWave Massive MIMO	HetNet C/U splitting	MEC Novel ARQ	Shared license RAN virtualization

mmWave Edge Cloud

Prefetch/cach user data/ application and transfer them instantaneously between MiEdge AP & UE



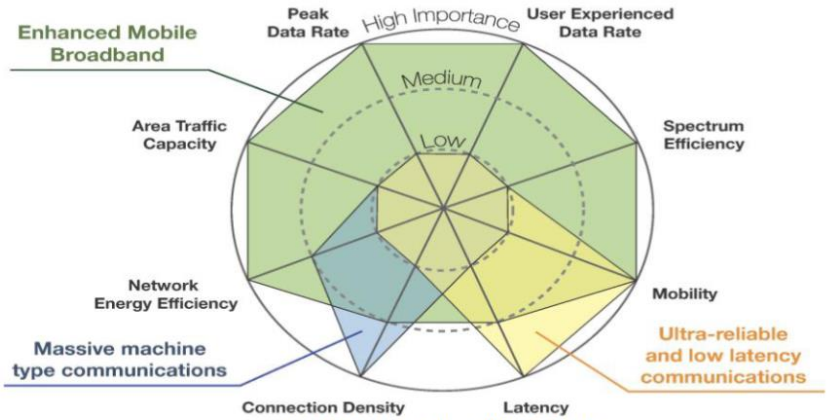
Liquid RAN C-plane

Collect context (location, action, etc.) information and provide traffic forecast to users and application providers

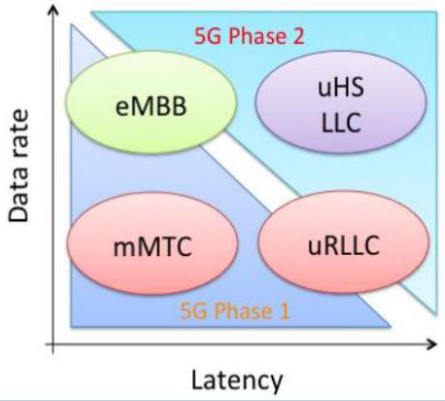
User/application centric orchestration

Network orchestration of MiEdge AP (SDN) by user or application provider

Technology enablers for uHSLLC and related KPIs



5G KPI defied in IMT-2020



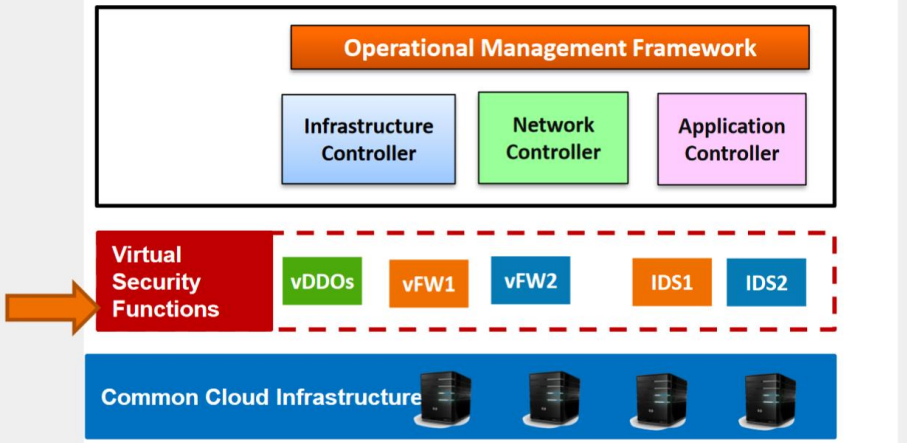
SDN/NFV and Security

An Example - Security Transformation – Virtual Firewall/Virtual DDOS/Virtual IPS

Non-Virtualized Security

- Wide variety of vendor specific security hardware
- Requires vendor specific FW management platforms
- Requires hands-on customized physical work to install
- Multiple support organizations
- No single operations model or database of record

Virtualized Security Function

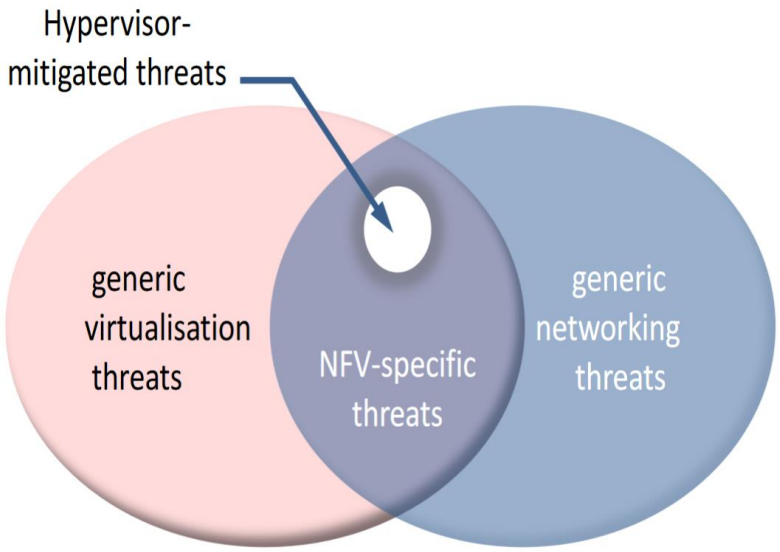


- Security functions will be cloud-based
- Security dynamically orchestrated in the cloud as needed
- Streamlined supplier integration
- Centralized common management platform
- Creates a standard operations/support model

ETSI and IEEE security challenges

Security Challenges in SDN/NFV Environment ETSI Problem Statement Draft

- Hypervisor Vulnerability
- API security
- Orchestration Vulnerability
- Virtual monitoring
  - Limited visibility to Mobility/EPC interfaces (e.g. S6a, S11, S8)
- Virtualized firewalls
- Secure boot
- Secure crash
- User/tenant authentication, authentication and accounting
- Topology validation and enforcement
- Performance isolation
- Authenticated Time Service
- Private Keys within Cloud Images
- Detection of attacks on resources in virtualization infrastructure
- Security monitoring across multiple administrative domains (i.e., Lawful Interception)

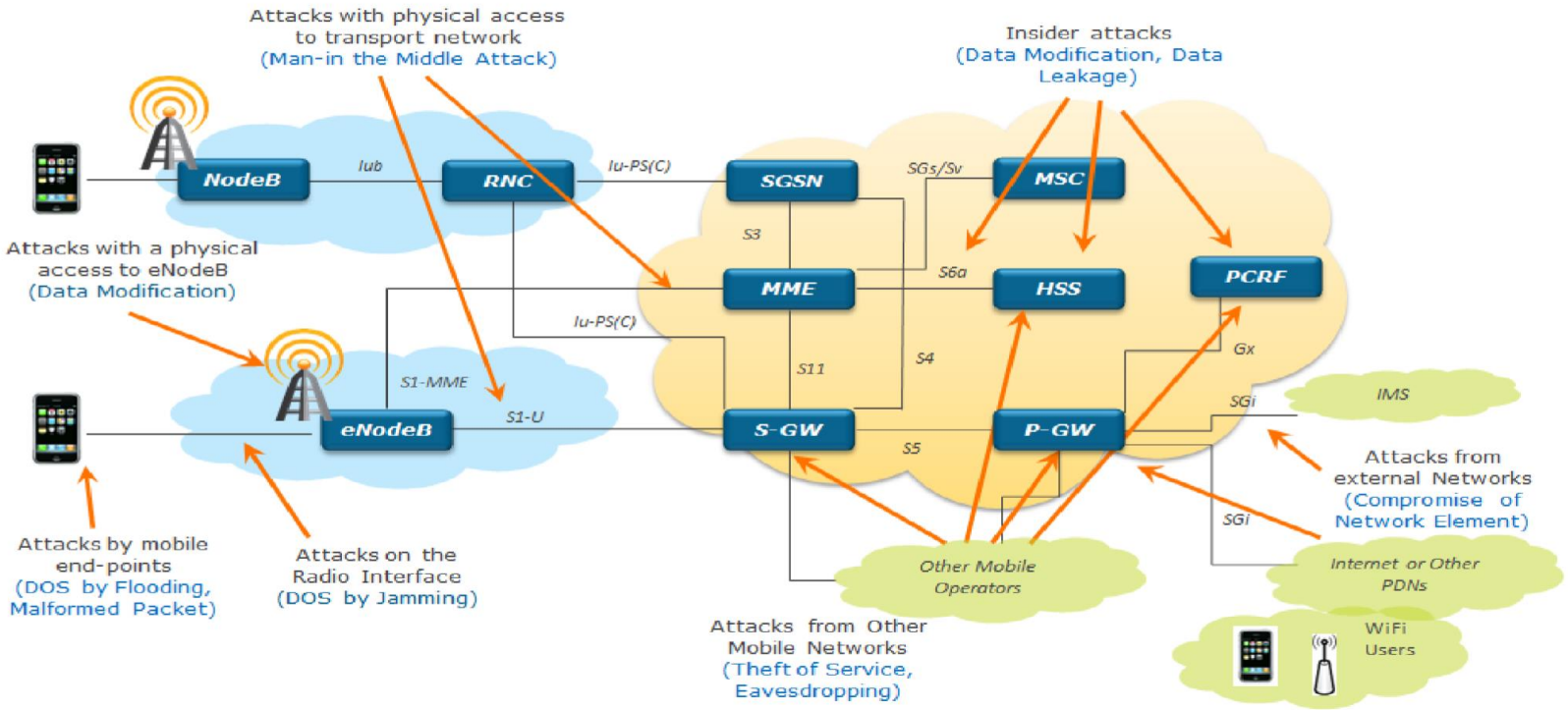


IEEE

Threats categories

General Threat Taxonomy (EPC) – Ref. ETSI/NFV Monitoring and Management (Draft 13)

LTE/EPC Security Threats Categories



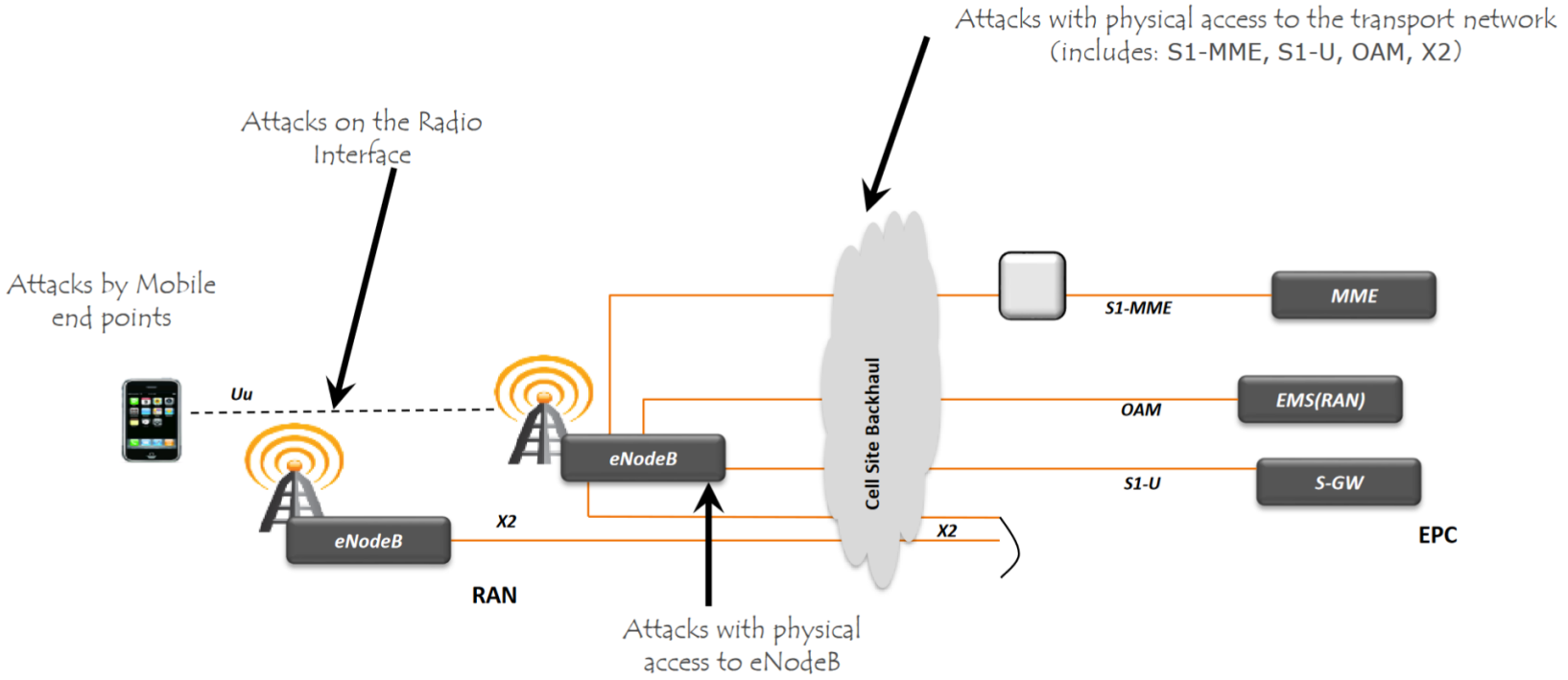
Threat Categories

Mobile Network Security - EPC
Threat Categories

	Category	Threat	Description
T1	Loss of Availability	Flooding an interface	Attackers flood an interface resulting in DoS condition (e.g. multiple authentication failure on s6a, DNS lookup)
T2		Crashing a network element	Attackers crash a network element by sending malformed packets
T3	Loss of Confidentiality	Eavesdropping	Attackers eavesdrop on sensitive data on control and bearer plane
T4		Data leakage	Unauthorized access to sensitive data on the server (HSS profile, etc.)
T5	Loss of Integrity	Traffic modification	Attackers modify information during transit (DNS redirection, etc.)
T6		Data modification	Attackers modify data on network element (change the NE configurations)
T7	Loss of Control	Control the network	Attackers control the network via protocol or implementation flaw
T8		Compromise of network element	Attackers compromise of network element via management interface
T9	Malicious Insider	Insider attacks	Insiders make data modification on network elements, make unauthorized changes to NE configuration, etc.
T10	Theft of Service	Service free of charge	Attackers exploits a flaw to use services without being charged

S6a is an LTE 4G mobile-related **interface** between the MME and HSS used for authentication, location & service information about the subscriber. It uses Diameter over TCP, UDP, SCTP transport

RAN threats


Attacks with physical access to the transport network
(includes: S1-MME, S1-U, OAM, X2)
Attacks on the Radio Interface
Attacks by Mobile end points
Uu
eNodeB
eNodeB
X2
RAN
Cell Site Backhaul
S1-MME
MME
OAM
EMS(RAN)
S1-U
S-GW
X2
EPC
Attacks with physical access to eNodeB

RAN Threats

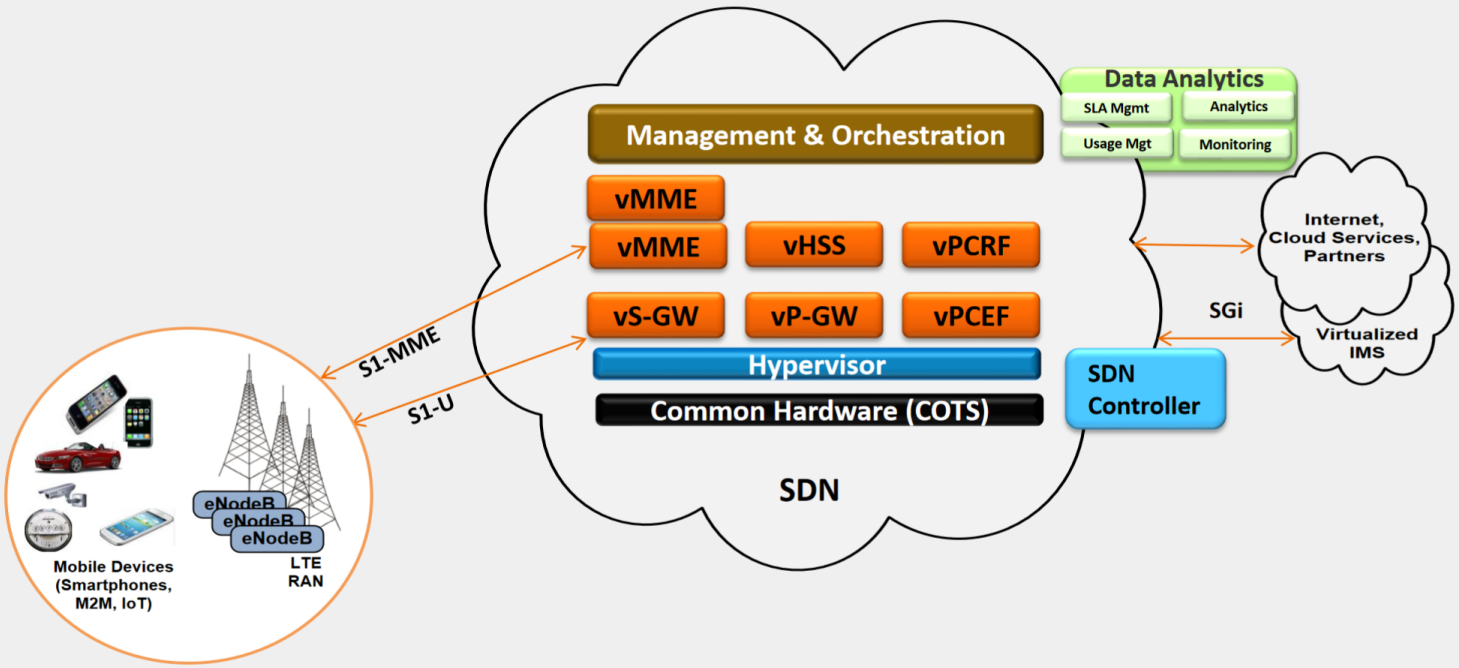
RAN Threat Categories

	Category	Threat	Description
T1	Loss of Availability	Flooding an interface	DOS on eNodeB via RF Jamming
T2		Crashing a network element	DDOS on eNodeB via UE Botnets
T3	Loss of Confidentiality	Eavesdropping	Eavesdropping on S1-MME/S1-U interfaces
T4		Data leakage	Unauthorized access to sensitive data on the eNodeB
T5	Loss of Integrity	Traffic modification	Man-in-the-Middle attack on UE via false eNodeB
T6		Data modification	Malicious modification of eNodeB configuration data
T7	Loss of Control	Control the network	Attackers control the eNodeB via protocol or implementation flaw
T8		Compromise of network element	Attackers compromise the eNodeB via management interface
T9	Malicious Insider	Insider attacks	Malicious Insider makes unauthorized changes to eNodeB configuration
T10	Theft of Service	Service free of charge	Theft of Service via Spoofing/Cloning a UE

The **S1-MME** carries **S1** application protocol (**S1**-AP) messages, using SCTP 2 over IP to provide guaranteed data delivery. Each SCTP association between an eNB and an **MME** can support multiple UEs. The **S1-MME** is responsible for EPC bearer setup and release procedures, handover signaling,

Main SDN/NFV EPC architecture

SDN/NFV-based Evolved Packet Core



Advantages of SDN/NFV for security challenges

Security Advantages of SDN/NFV

A Comprehensive View of SDN/NFV Security Advantages

Performance Improvements:

Streamline and Reduce Incident Response Cycle Time

Streamline and Reduce Patching Cycle Time

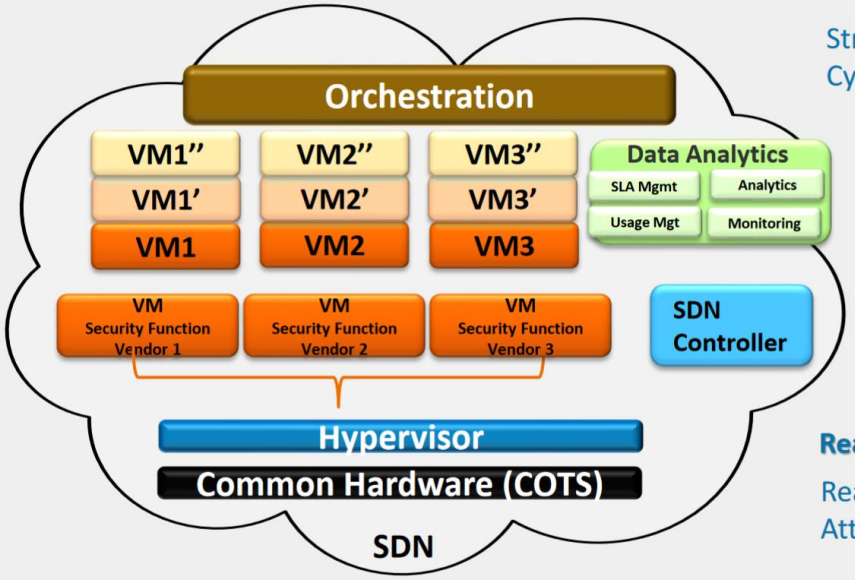
Design Enhancements:

Centralize Control and Management Functions

Security Embedded at Design Time

Security that Exceeds Existing Perimeter

Multivendor Security Service

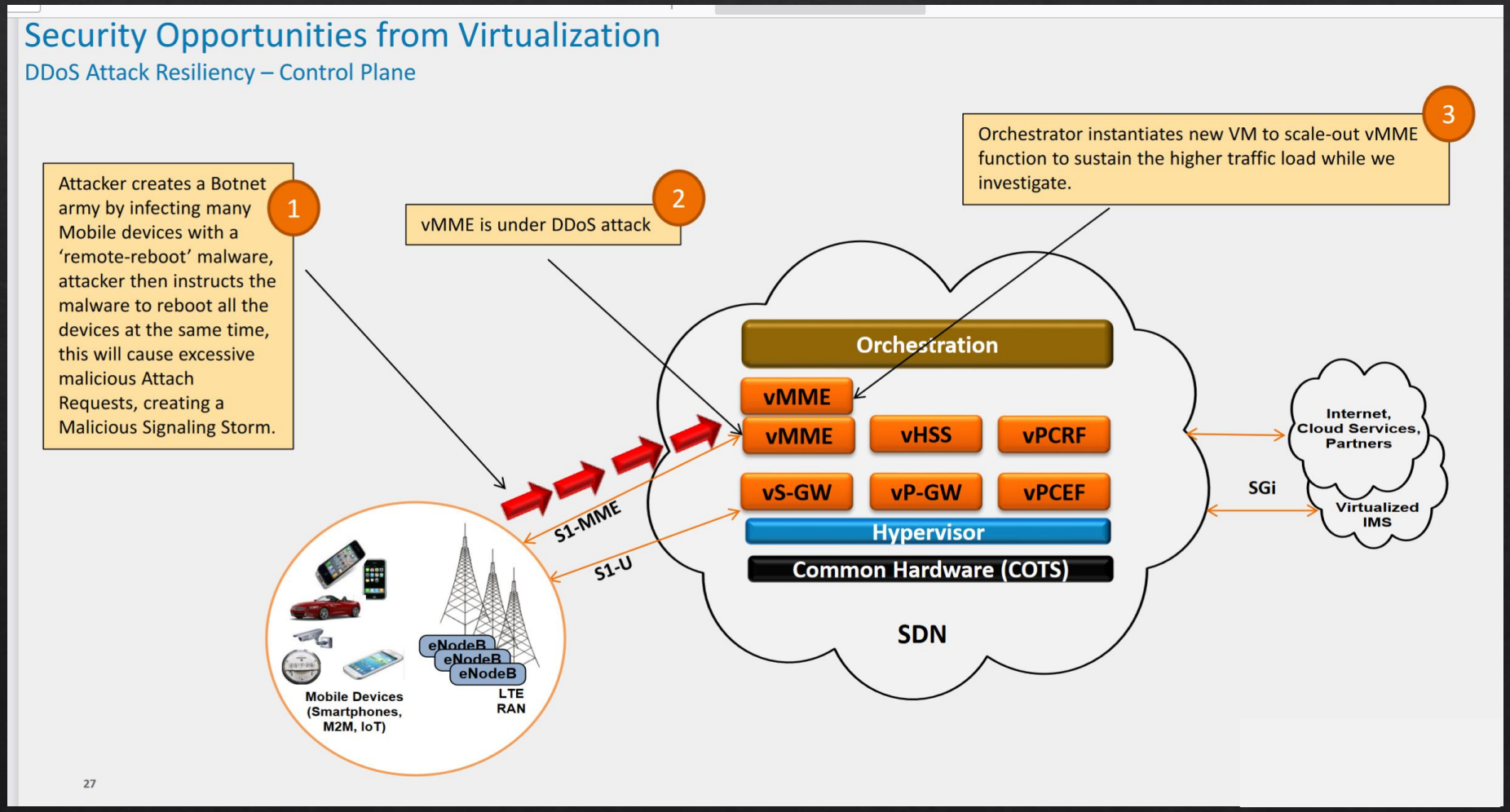


Real-Time capabilities:

Real-Time Scaling to Absorb DDOS Attacks

Real-Time Integration of "Add-on" Security Functions

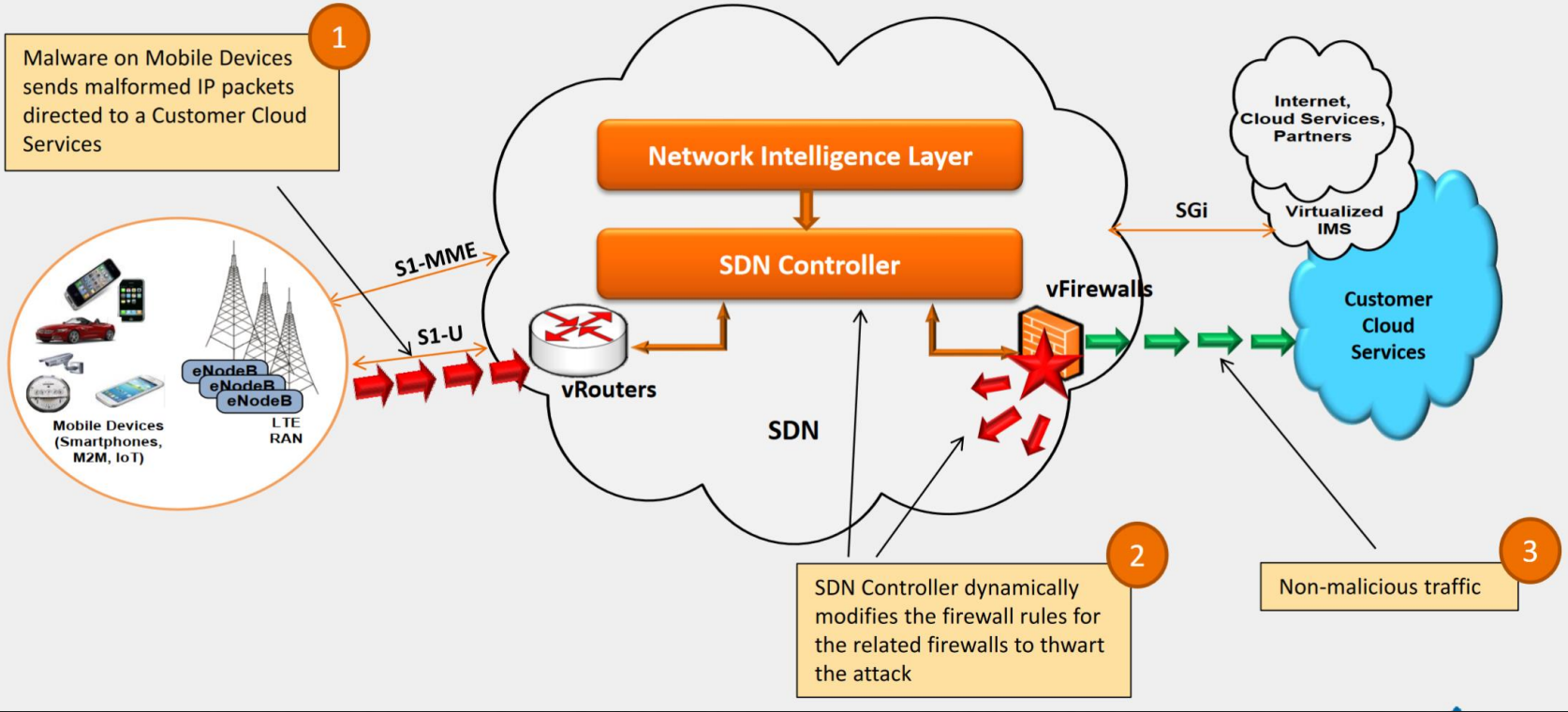
Example for Security optimization



Example for Security optimization

Security Opportunities from Virtualization

SDN Controller Dynamic Security Control – Data Plane



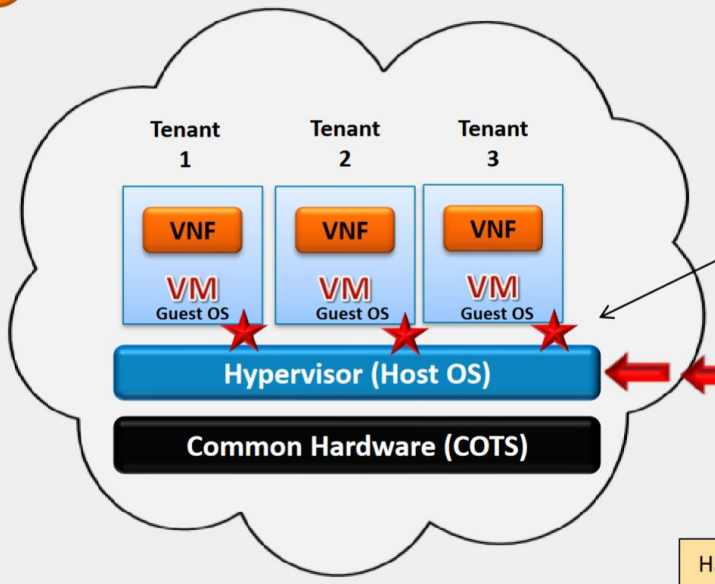
Example for Security challenges

Security Challenges from Virtualization

Hypervisor Vulnerabilities

To prevent this type of attack, we must:
- ✓ Conduct security scans and apply security patches
- ✓ Ensure the Hypervisor is hardened and minimized (close vulnerable ports)
- ✓ Ensure the access to the Hypervisor is controlled via User Access Management,

3



Malware compromises VMs:
- VM/Guest OS manipulation
- Data exfiltration/destruction

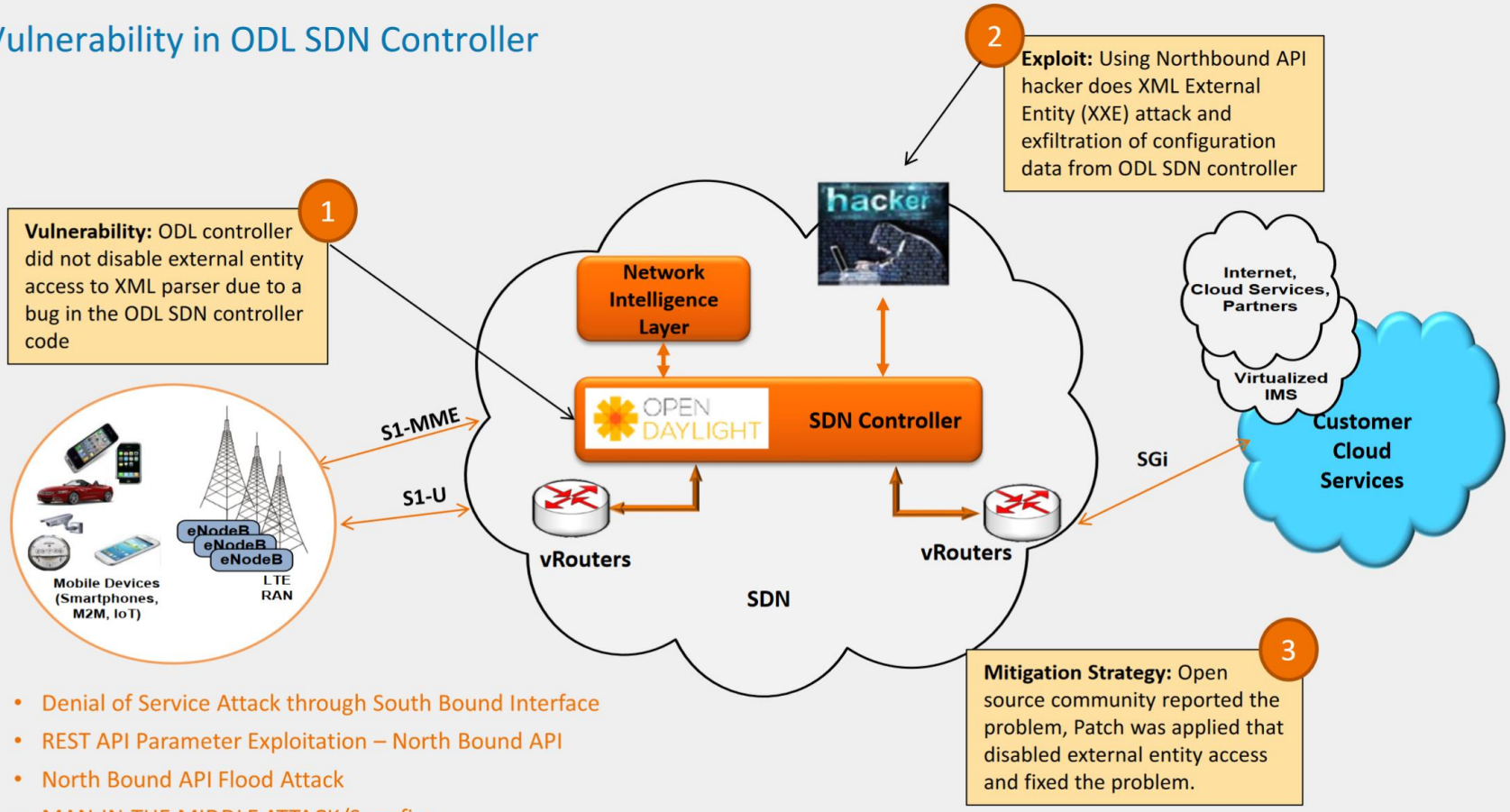
2

Hacker exploits a vulnerability in the Open Source code and infects the Hypervisor with a Malware

1

SDN Controller security challenges

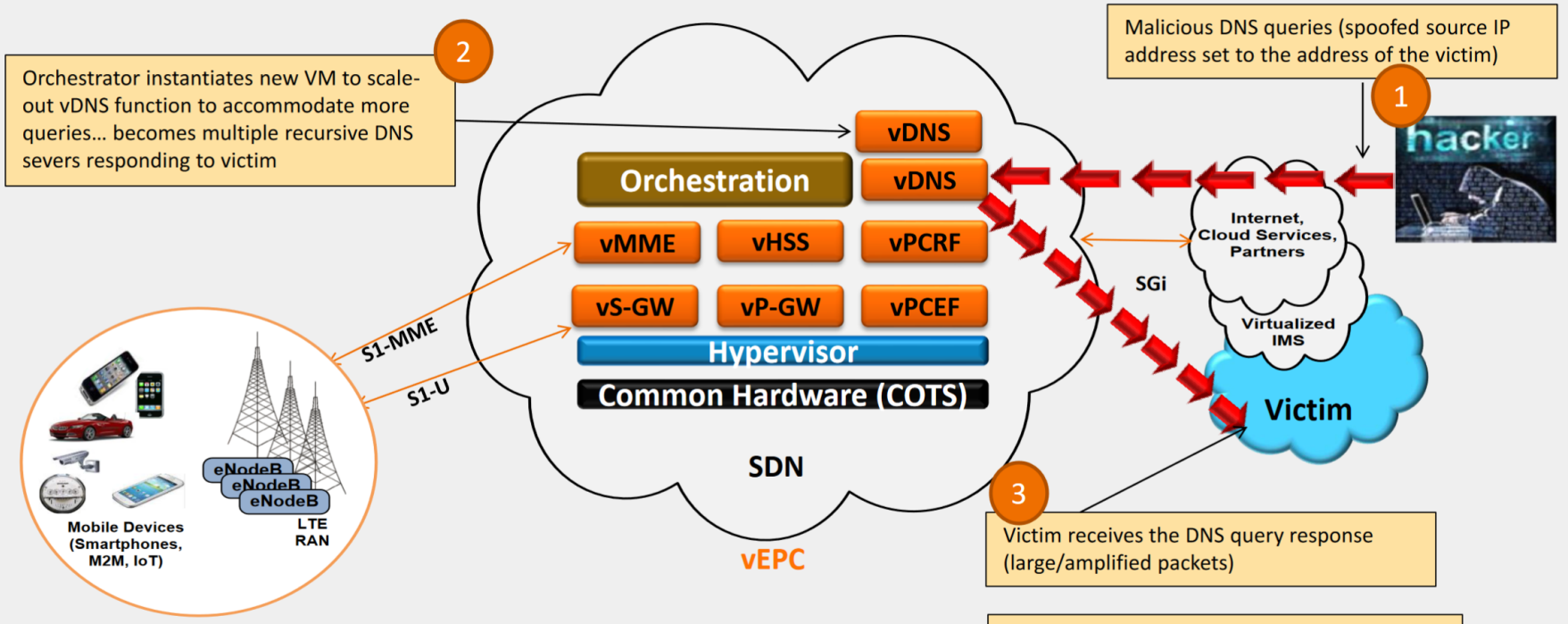
Security Vulnerability in ODL SDN Controller



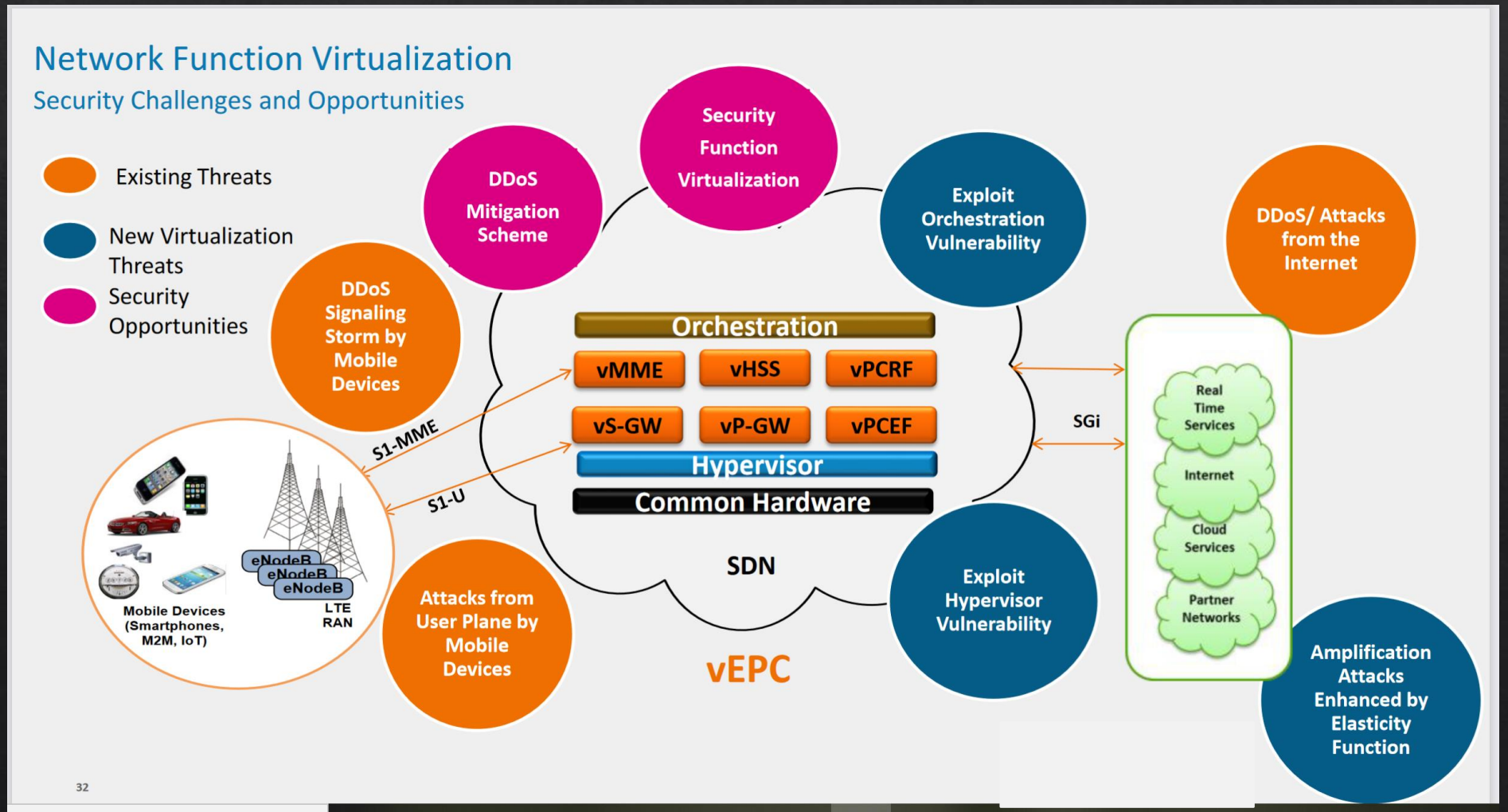
- Denial of Service Attack through South Bound Interface
- REST API Parameter Exploitation – North Bound API
- North Bound API Flood Attack
- MAN-IN-THE MIDDLE ATTACK/Spoofing
- Protocol Fuzzing – South Bound
- Controller Impersonation – South Bound

Threats Mitigation

DNS Amplification Attacks Enhanced by Elasticity Function



SDN/NFV Security overview

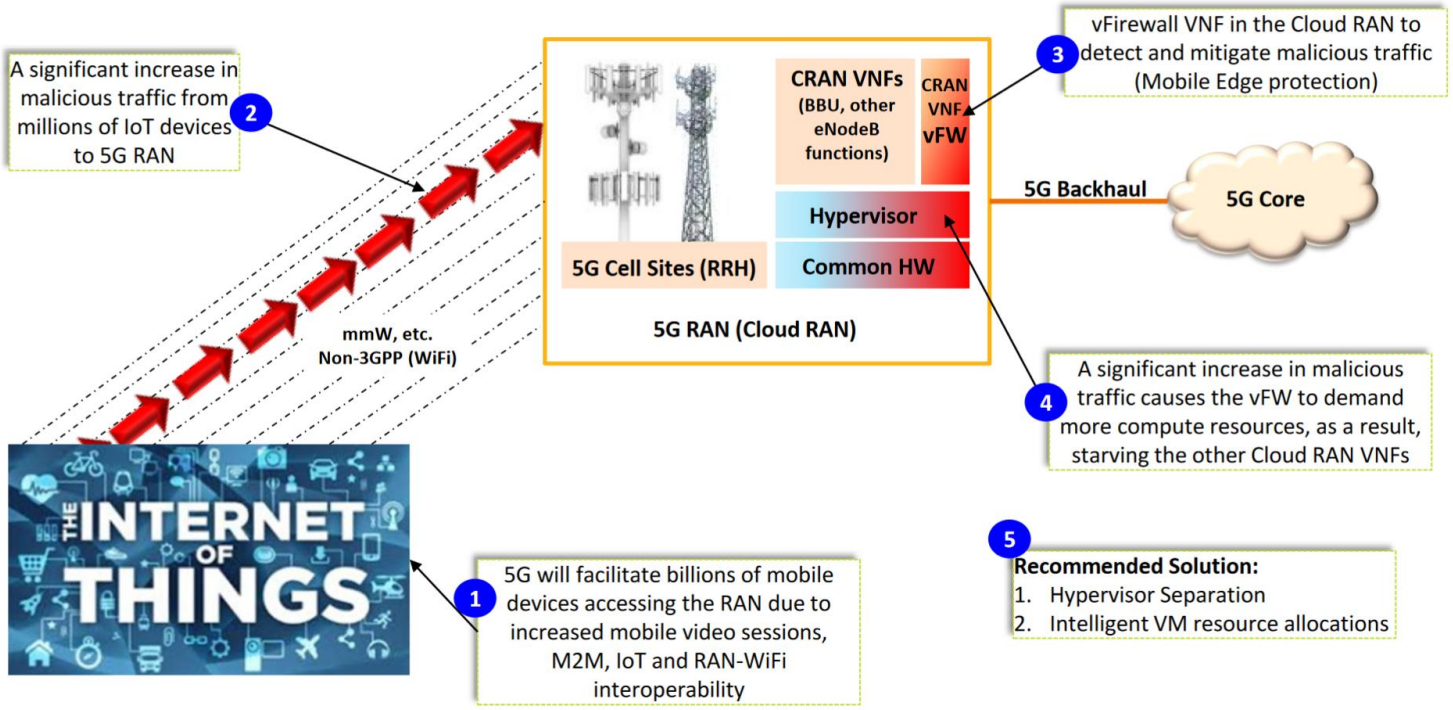


Security challenges for 5G & SDN

Virtualization (NFV and SDN) is the Foundation upon which 5G will be Built

Opportunities and Risks associated with Virtualization will apply to 5G VNFs

Use Case: CRAN (Cloud RAN) Resource Starvation due to 5G RAN Firewall Functions

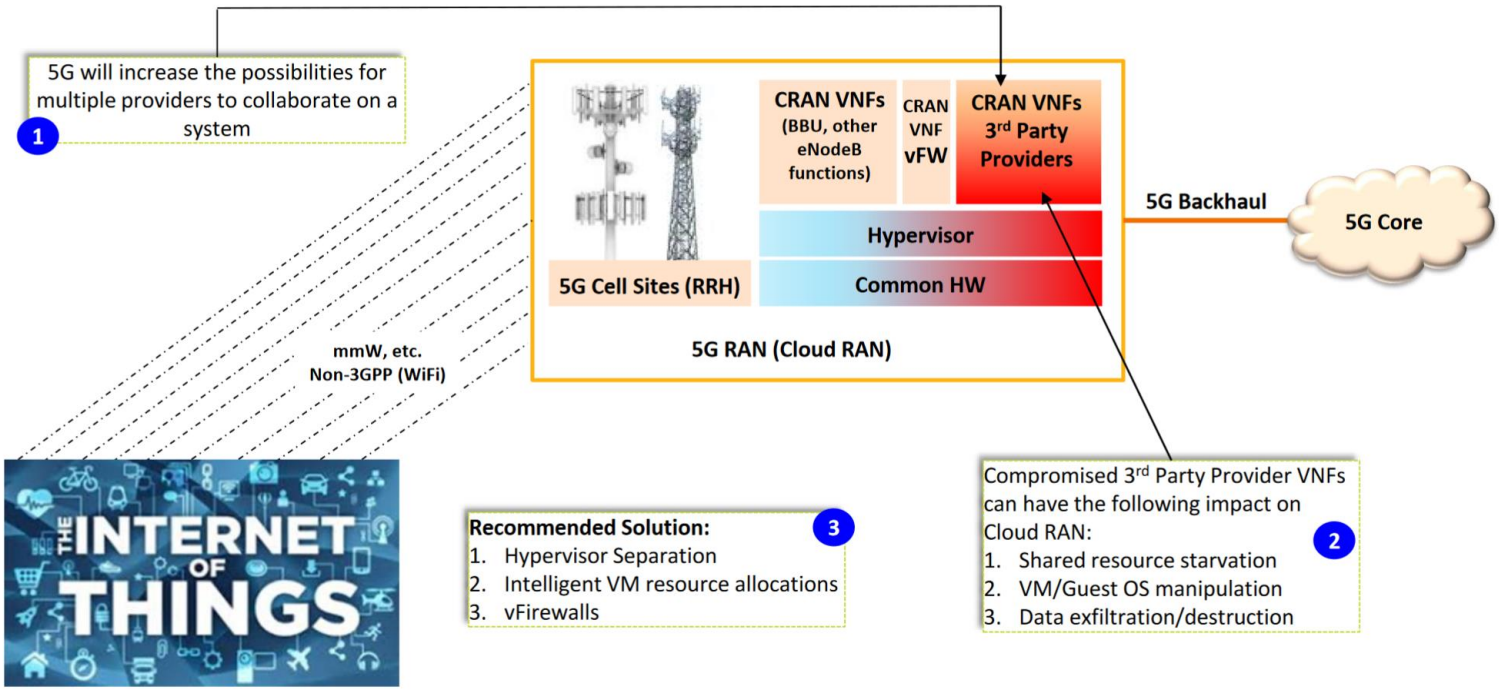


Multi-tenancy challenges for security

5G will Increase the Possibilities for Multiple Providers to Collaborate on a System

Increase the Risk of Compromise Shared Resources

Use Case: Compromise Shared Resources



Use Cases for security

Security Use Cases for Mobile Edge Computing

- Storage of Sensitive Security Assets at the Edge
- Third party applications on the same platform as network functions
- User Plane attacks in Mobile Edge Computing Environment
- Exchange of Sensitive Security Assets between core and Mobile Edge
- Trust establishment between functions at the core and at the edge
- Subscriber authentication within the visited network
- Secure storage of credentials to access IMS network
- Access to 5G core over non-3GPP network access
- User plane data security over less trusted 3GPP network accesses
- Management of credentials to access non-3GPP network access

Mobile Edge Computing – Use Case

Storage of Sensitive Security Context at the Mobile Edge

- IOT type applications require low latency, faster authentication and hence, need security context to be stored at the edge of the network

- Sensitive Security Assets stored at the mobile edge should be encrypted
- threat to sensitive assets while temporarily decrypted also needs to be addres

- Sensitive security assets are compromised at virtualized functions at the edge
- An attacker could maliciously reuse them to gain connectivity or carry out a spoofing, eavesdropping or data manipulation attack.

Recommended Solutions:
1. Virtual Firewalls at MEC
2. Encryption at the Edges
3. IDS/IPS to detect and mitigate spoofing and eavesdropping

1

MEC Server
Edge
Cloud

3

vMME
vHSS
vS/PGW-C

Network
Slice 1
Control Plane

2

4

5G Core

MEC Server
Edge
Cloud
VNF

vP-CSCF
vS-CSSCF
vHSS

Network
Slice 4
vIMS

vSGW-U
vPGW-U
vDNS

Network
Slice 2
vEPC Data Plane

VNF

Edge
Cloud
MEC Server

Internet

Network Slicing Use Case

Side channel attacks across Network slices

- Each slice is dedicated for a specific operator.
- These slices run on the same hardware controlled by the hypervisor

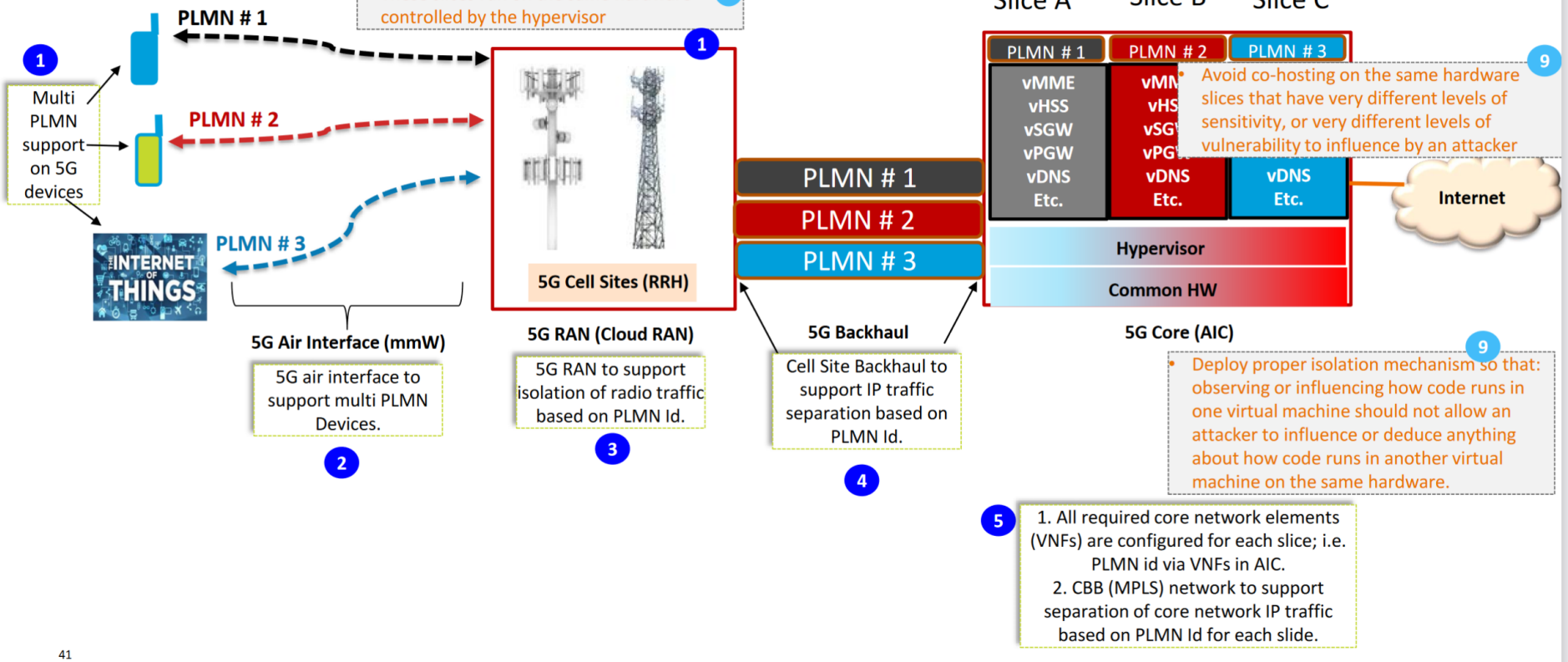
6

- If an attacker can observe or influence how code runs in functions in slice A, she/he may be able to affect the running of code in functions in the slice B machine, or extract information about the running of code in slice B.

7

- This may allow side channel attacks – in particular, timing attacks – that extract information about cryptographic keys or other secrets in slice B.

8



- Avoid co-hosting on the same hardware slices that have very different levels of sensitivity, or very different levels of vulnerability to influence by an attacker

9

5G air interface to support multi PLMN Devices.

2

5G RAN to support isolation of radio traffic based on PLMN Id.

3

Cell Site Backhaul to support IP traffic separation based on PLMN Id.

4

- Deploy proper isolation mechanism so that: observing or influencing how code runs in one virtual machine should not allow an attacker to influence or deduce anything about how code runs in another virtual machine on the same hardware.

9

5

1. All required core network elements (VNFs) are configured for each slice; i.e. PLMN id via VNFs in AIC.
2. CBB (MPLS) network to support separation of core network IP traffic based on PLMN Id for each slide.

Public Land Mobile Network Identity (PLMN-ID) = three digit mobile country code (MCC) + a two or three digit mobile network code (MNC

Security use cases

Security Use Cases for Network Slicing

- Controlling Inter-Network Communications
- Instantiation time Impersonation attacks against Network Slice Manager
- Impersonation attacks against a Network Slice instance within an Operator Network
- Impersonation attacks against different Network Slice managers within an Operator Network
- Different Security Protocols or Policies in different slices
- Denial of Service to other slices
- Exhaustion of security resources in other slices
- Side Channel Attacks Across Slices
- Hybrid Deployment Model
- Sealing between slices when UE is attached to several slices

3GPP Management Framework

Architectural framework-Management reference model and interfaces

A number of management interfaces in a PLMN are identified, namely:

1) between the Network Elements (NEs) and the Element Manager (EM) of a single PLMN Organisation;
2) between the Element Manager (EM) and the Network Manager (NM) of a single PLMN Organisation;
3) between the Network Managers and the Enterprise Systems of a single PLMN Organisation;
4) between the Network Managers (NMs) of a single PLMN Organisation;

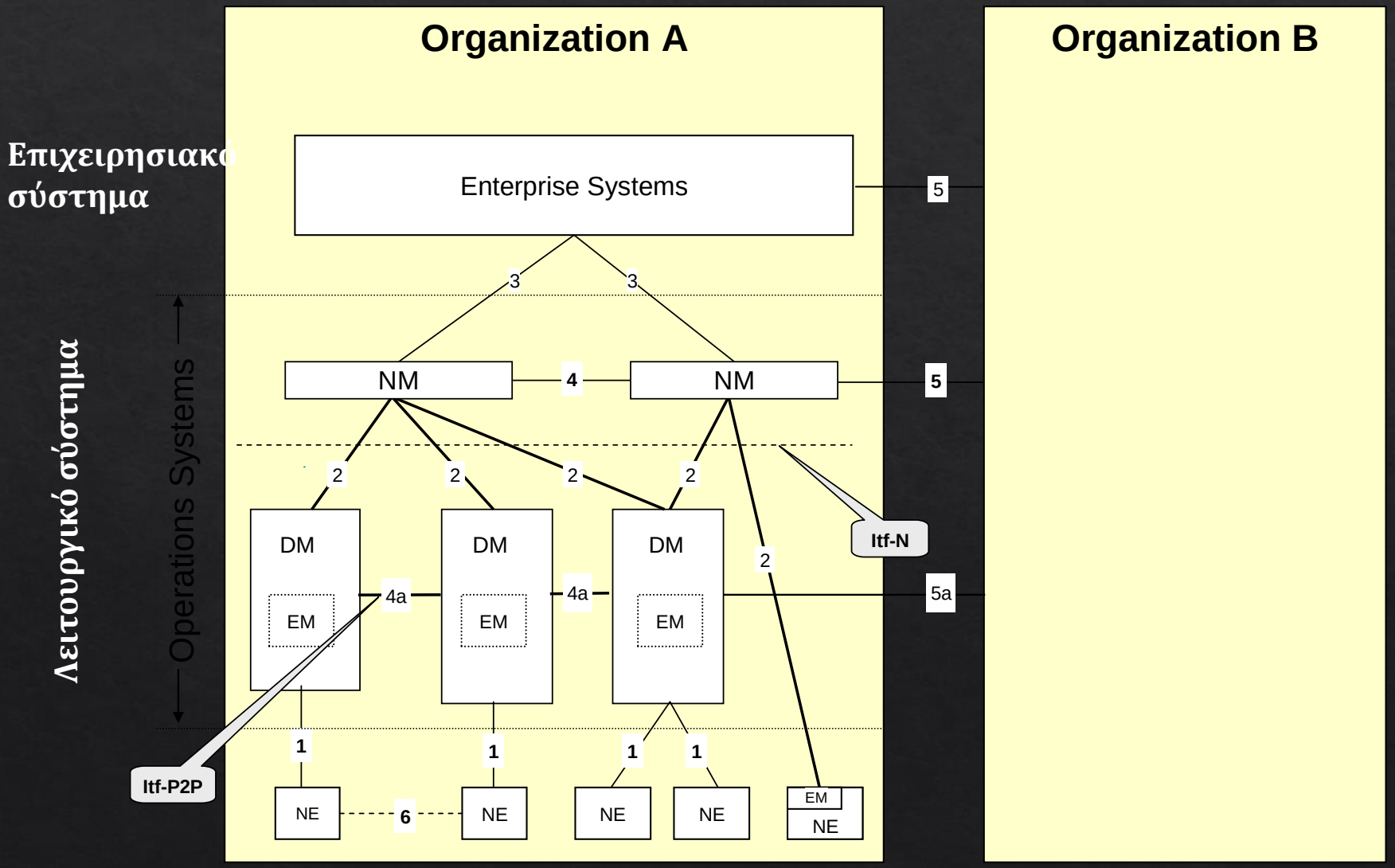
4a) between the Domain Managers (DMs) of a single PLMN Organisation.

5) between Enterprise Systems & Network Managers of different PLMN Organisations;

5a) between the Domain Managers (DMs) of different PLMN Organisations.

6) between Network Elements (NEs).

Management reference model



Types of interfaces

Interfaces from EM Operations Systems to NEs (Type 1)

- The approach for NE management interfaces of Type 1 will be to allow the use of certain management application protocol suites (SNMP, CORBA IIOP, SOAP)

Interfaces from NM Operations Systems to NEs (Type 2)

- The approach for NE management interfaces of Type 2 will be to concentrate on protocol independent information models, allowing a mapping to several protocol suites

Types of interfaces

Interfaces to Enterprise Systems (Type 3)

- Enterprise Systems are those Information Systems that are used in the telecommunication organisation but are not directly or essentially related to the telecommunications aspects (Call Centres, Fraud Detection and Prevention Systems, Invoicing etc.).
- Standardising Enterprise Systems is out of the scope of 3GPP. However, it is essential that the requirements of such systems are taken into account and interfaces to the Operations Systems are defined, to allow for easy interconnection and functional support.

Interface between Network Managers (Type 4)

- Interface type 4 (where Itf-P2P is between Domain Managers of different PLMN Organisations) could have additional requirements over interface type 4a .

Types of interfaces

Interface between Domain Managers (Type 4a) - the Itf-P2P Interface

- The approach for Interfaces of type 4a (the Itf-P2P interface) is the same as for interfaces of type 2 (the Itf-N interface).
- The Itf-P2P should as much as possible re-use the interface definitions of the Itf-N interface.

Interfaces to Operations Systems in other organisations (Type 5)

- PLMN management considers integrally the interaction with the Operations Systems of other legal entities for the purpose of providing Mobile services.
- There are two major types of interfaces to other management systems:
  - 1) To the Operations Systems of another PLMN Organisation;
  - 2) To the Operations Systems of a non-PLMN Organisation.
- The first type deals with co-operation to provide Mobile services across a number of PLMN networks (e.g. roaming related interactions). The second type deals with client-server relationship

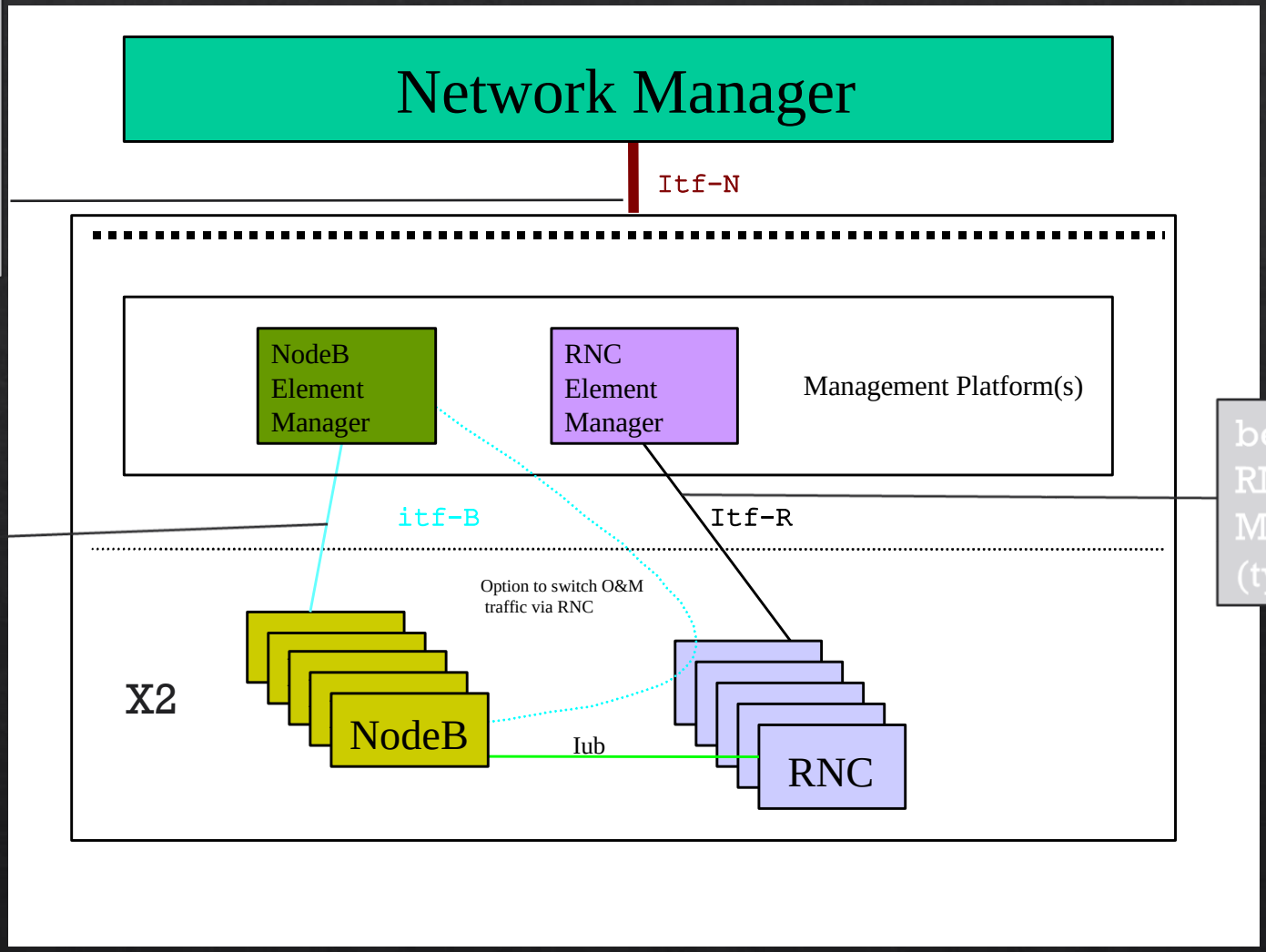
Inter-NE interfaces (Type 6)

- Interfaces between Network Elements are sometimes used to carry management information even though this may not be the primary purpose of the interface. An example in a 3G network is the Iub interface between Node-B and RNC

Radio Network management interfaces

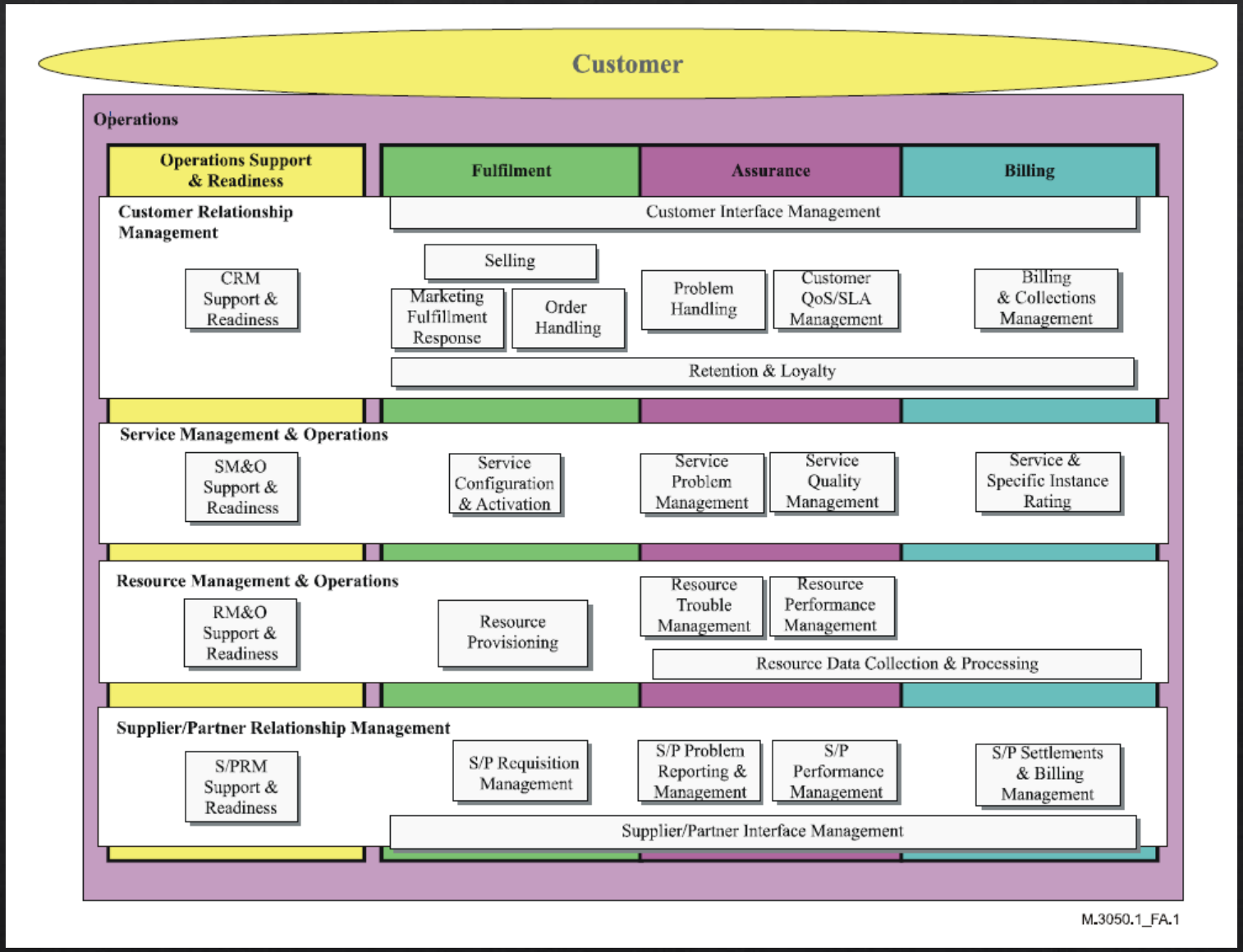
between the Network (Element Manager or NEs with an embedded EM) & Network Manager (type 2)

between Node B & its Manager (type1)



between RNC & its Manager (type 1)

Enhanced Telecom Operations Map Business Process Model (TeleManagement Forum)



CRM

Customer QoS/SLA Management

- Customer QoS/SLA Management processes encompass monitoring, managing and reporting of delivered vs contractual Quality of Service (QoS), as defined in the enterprise's Service Descriptions, customer contracts or product catalogue.
- They are also concerned with the performance of the enterprise and its products and services in relation to its Service Level Agreements (SLAs) for specific product instances, and other service-related documents.
- They include operational parameters such as resource performance and availability, but also encompass performance across all of a product's contractual or regulatory parameters, e.g., % Completion on Time for Order Requests, time to repair commitments, customer contact performance. Failure to meet a contracted SLA may lead to billing adjustments, which are handled by Billing and Collections Management.

Billing & Collections Management)

- Billing & Collections Management processes encompass creating and maintaining a customer's billing account, sending bills to customers, processing their payments, performing payment collections, monitoring the status of the account balance, and the handling of customer generated or systems reported billing and payment exceptions.
- These processes are accountable for assuring that enterprise revenue is billed and collected.

Service Management & Operations (SM&O) Processes

Service Configuration & Activation

- Service Configuration & Activation processes encompass the installation and configuration of the service for customers. They also support the reconfiguration of the service (either due to customer demand or problem resolution) after the initial service installation.

Service Problem Management

- The purpose of the Service Problem Management processes is to respond immediately to customer-affecting service problems or failures in order to minimize their effects on customers, and to invoke the restoration of the service, or provide an alternate service as soon as possible. They encompass the reporting of problems, making a temporary fix or work-around, isolating the root cause and finally recovering the complete functionality of the service and providing information for future enhancements.

Service Management & Operations (SM&O) Processes

Service Quality Management

- The purpose of the Service Quality Management processes encompasses monitoring, analysing and controlling the performance of the service perceived by customers. These processes are responsible for restoring the service performance for customers to a level specified in the SLA descriptions as soon as possible.

Service & Specific Instance Rating

- Service & Specific Instance Rating processes manage service events by correlating and formatting them into a useful format. These processes include the service level rating of usage information. Investigation of service related billing event problems is also part of these processes. These processes provide information on customer-related and Service-related events to other process areas.

Resource Management & Operations (RM&O) Processes

RM&O Support & Readiness

- RM&O Support & Readiness processes manage classes of resources, ensuring that appropriate application, computing and network resources are available and ready to support the Fulfilment, Assurance and Billing processes in instantiating and managing resource instances. This includes, but is not limited to:
  - Managing the Resource Knowledge base;
  - Configuring the resources and provisioning of logical resources to be able to support specific service classes;
  - Analysing availability and performance over time on resources or groups of resources, including trend analysis and forecasting;
  - Demand balancing in order to maintain resource capacity and performance;
  - Performing pro-active maintenance and repair activities.

Resource Provisioning

- Resource Provisioning processes encompass allocation and configuration of resources to individual customer service instances in order to meet the service requirements. This includes activation as well as testing to ensure the expected performance of the service. Responsibilities of the Resource Provisioning processes include, but are not limited to:
  - Verifying whether appropriate resources are available as part of pre-order feasibility checks;
  - Allocating the appropriate resources to support the customer service instance;
  - Reserving the resources (if required by the business rules) for a given period of time until the customer confirms the order;
  - Possibly delivering the physical resource to the central office or customer premise;
  - Configuring and activating physical and/or logical resources, as appropriate;
  - Testing the resource to ensure the resource is working correctly and meets the performance requirements implied by the service's Key Quality Indicators;
  - Updating of the Resource Inventory Database to reflect that the resource is being used for a specific customer.

Resource Management & Operations (RM&O) Processes

Resource Trouble Management

- RTM processes are responsible for the management of troubles with allocated resources. The objectives of these processes are to report resource failures, to isolate the root cause and act to resolve them. They include, but are not limited to:
  - Detect, analyse and report Resource Failure Events;
  - Fault localization analysis;
  - Correcting Resource Faults;
  - Resource trouble reporting to amongst others Service Problem Management processes;
  - Resource trouble administration to ensure repair activities are assigned and tracked efficiently.
  - On one hand, resource troubles relate to Problems in the Service and hence the customer domain. On the other hand, they relate to resource failures, which are caused by resource faults.
  - As such, the Resource Trouble Management processes work with resource failure events received from Resource Data Collection & Processing, resource quality problem notifications from Resource Performance Management, and potential resource failure notifications from Support Resource Trouble Management.
  - Resource Trouble Management processes perform analysis, decide on the appropriate actions/responses and carry them out. However, these activities need to interact with the Service Problem Management processes, as the latter have a view on service impact.

Resource Management & Operations (RM&O) Processes

Resource Performance Management

- Resource Performance Management processes encompass monitoring, analysing, controlling and reporting on the performance of resources. They work with basic information received from the Resource Data Collection & Processing processes.
- If the analysis identifies a resource quality problem, information will be passed to Resource Trouble Management and/or Service Quality Management. The latter processes are responsible for deciding on and carrying out the appropriate action/response. This may include requests to the Resource Performance Management processes to install controls to optimize the resource performance.
- The Resource Performance Management processes will continue to track the resource performance problem, ensuring that resource performance is restored to a level required to support services.
- Depending on the resource class, the Resource Performance Management processes might send an abatement message to Resource Trouble Management once the resource performance problem has been cleared.

Resource Management & Operations (RM&O) Processes

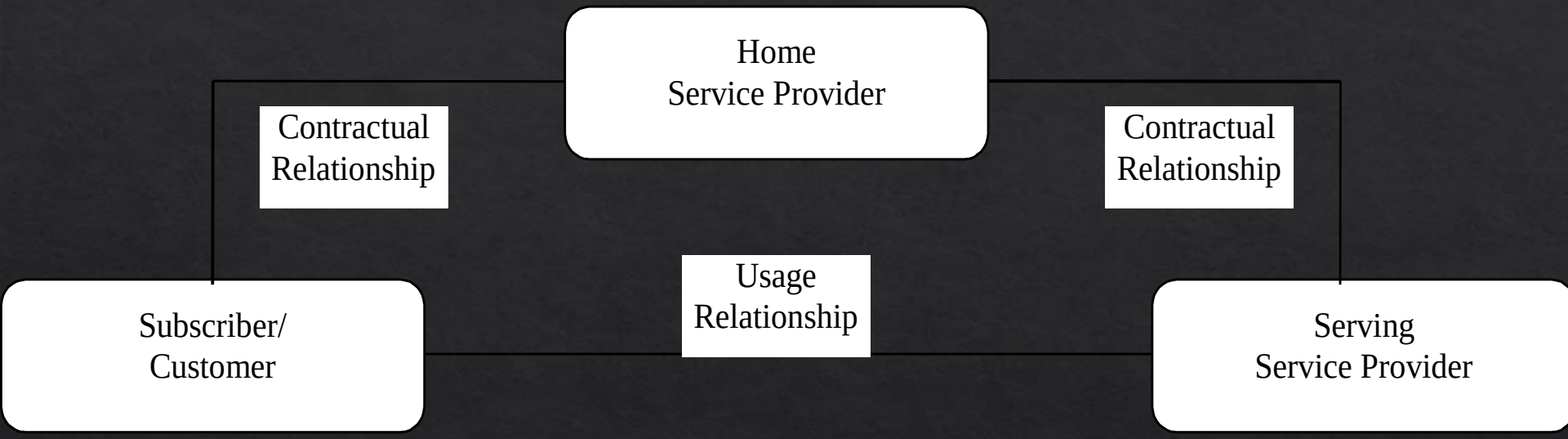
Resource Data Collection & Processing

- Resource Data Collection & Processing processes interact with the resources to collect usage, network and information technology events and performance information for distribution to other processes within the enterprise.
  The responsibilities also include processing the data through activities such as filtering, aggregation, formatting and correlation of the collected information before presentation to other processes. Client processes for this information perform usage reporting and billing activities, as well as Fault and Performance analysis of resources and services. These include Resource Performance Management, Service Quality Management and Service & Specific Instance Rating.

3GPP management functions

- Performance management;
- Roaming management;
- Fraud management;
- Fault management;
- Security management;
- Software management
- Configuration management;
- Accounting management;
- Subscription management;
- Quality of Service (QoS) management;
- User equipment management.

Roaming management overview



Roaming is a service provided by Mobile Service Providers. Customers of a Home Service Provider may use the infrastructure of another, a Serving Service Provider to give its customer the ability to make calls when outside the home service provider' s territory. The goal is to have a customer receive the same service (or as close to the same service) when travelling in an area supported by another network as the customer receives when in their home service provider's area.

Fraud management overview

Fraud and all the activities to detect and prevent fraud are quite common to any network. Nonetheless, mobility and roaming, two integral mobile services, make fraud detection and fraud prevention more complicated and more urgent.

The mobile service provider does not know the location of the "end of the wire," which would lead to the home of a fraudulent customer. For roaming, the situation is demonstrably worse. For a roaming visitor the caller is not the service provider's customer and therefore, the service provider does not have complete information to assess fraud. In the reverse case, the service provider has little control when its customers are roaming, e.g., potentially going over credit limits or using service after being suspended. In this case, the fraudulent customer uses the network facilities of another provider (the serving service provider) meaning the home service provider has to rely on the serving service provider for some level of fraud protection support. This means to a large extent that fraud prevention is largely out of the control of the home service provider when one of its customers roams on another network and out of the control of a serving service provider when being visited by another provider's roamer.

Fault Management

- Is accomplished by means of several processes/sub-processes like fault detection, fault localisation, fault reporting, fault correction, fault repair, etc. These processes/sub-processes are located over different management layers, however, most of them (like fault detection, fault correction, fault localisation and fault correction) are mainly located over the Network Element and Network Element Management layers, since this underlying network infrastructure has the 'self healing' capabilities.
- Network data management logically collects and processes performance and traffic data, as well as usage data.
- While the Fault Management triggered within the Network Element and NE management layers is primarily reactive, the Fault Management triggered within the Network and Systems Management layer is primarily proactive. Meaning triggered by automation rather than triggered by the customer; and this is important for improving service quality, customer perception of service and for lowering costs.
- Focusing on the Network and Systems Management layer, when a fault/problem is detected, no matter where and how, several processes are implicated.

Fault Management