IEEE 802-ec-13-0055-00-00EC

IEEE 802 Tutorial – Wireless SDN in Access and Backhaul

12 November 2013
Dallas, TX, USA

Paul Congdon (Tallac Networks)
Luis Contreras (Telefónica)
Serge Manning (Huawei)
Roger Marks (EthAirNet Associates)
Antonio de la Oliva (Universidad Carlos III de Madrid)
Juan Carlos Zúñiga (InterDigital Communications)

Abstract

Software-Defined Networking (SDN) is becoming widely deployed throughout the network. This tutorial addresses applications of SDN in wireless access and backhaul networks, with a focus on issues relevant to IEEE 802 technologies. Tutorial background on the basics of OpenFlow SDN will be provided as an introduction to the subject.

Outline

SDN Concepts

Paul Congdon, Tallac Networks

OpenFlow for Mobile & Wireless Networks

Serge Manning, Huawei

Operator’s Vision

Luis Contreras, Telefónica

SDN in IEEE 802 Network Reference Model

Juan Carlos Zúñiga, InterDigital Communications

SDN in Wireless Access

Antonio de la Oliva, Universidad Carlos III de Madrid

SDN in Wireless Backhaul Service

Roger Marks, EthAirNet Associates

Looking Forward

Questions and Answers

SDN Concepts

Paul Congdon
Tallac Networks

Defining SDN
(Software Defined Networking)

- **Definition (Wikipedia paraphrased):**
  - An approach to networking that decouples the control plane from the data plane and abstracts lower level functionality in order to simplify the management and delivery of network services.
- **Historical Roadmap (always debatable):**
  - **2003: FORCES architecture (RFC 3654):**
    (**FOR**warding and **C**ontrol **E**lement **S**eparation)
  - **2005: Clean State Project (Stanford University)**: re-inventing networking
  - **2006: Ethane Project (Stanford University)**: early implementation (*campus*)
  - **2007:** First experimental OpenFlow protocol support in **vendor** devices
  - **2008: OpenFlow v0.8.9:** initial published specification
  - **2009:** The term **SDN** is associated with this movement. http://bit.ly/108bBJT
  - **2011**: OpenFlow development turned over to the newly formed Open Networking Foundation (ONF)
  - **Today**: Depends on who you talk to due to market 'definition' refactoring

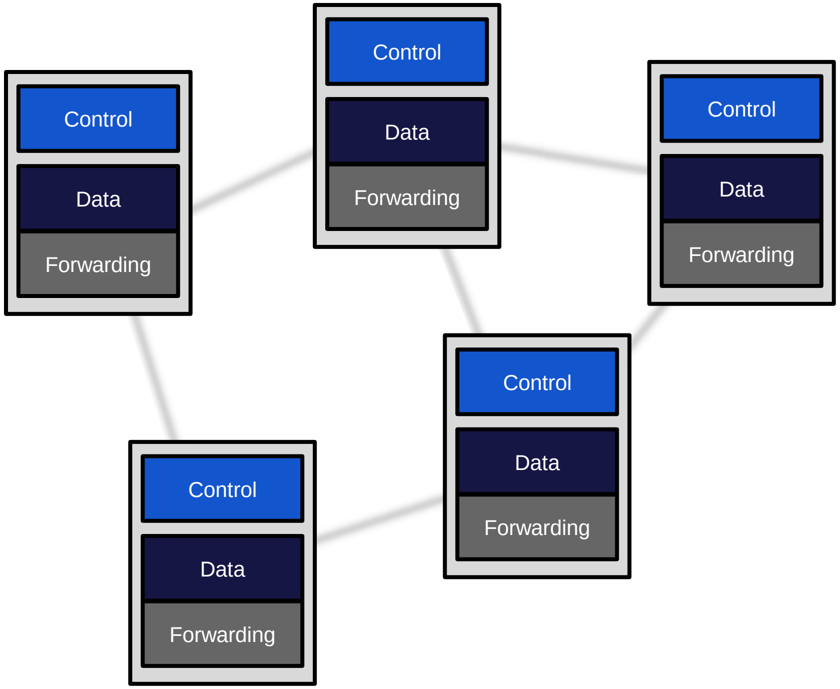
Why SDN?

What's wrong with the architecture we've had for the last 20 years?

It Has:

- Distributed network intelligence
- Independent and autonomous devices
- End-to-End principals and empowered clients
- Network control functionality in individual devices

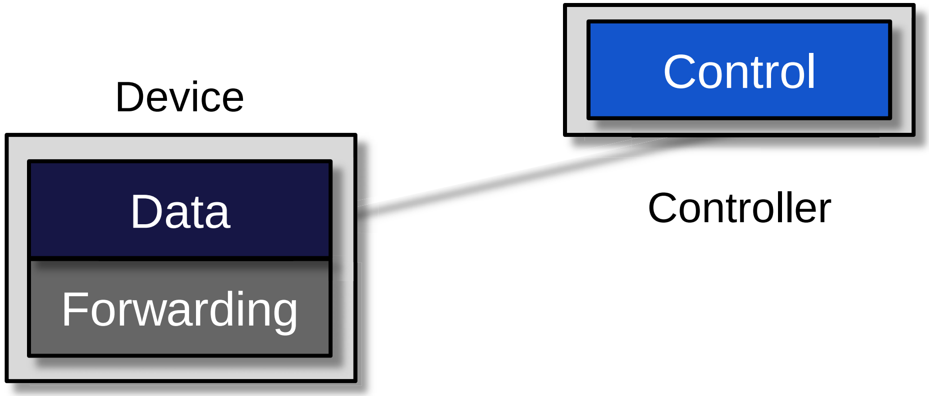
  NOTE: Wireless system architectures have found reasons to deviate from this



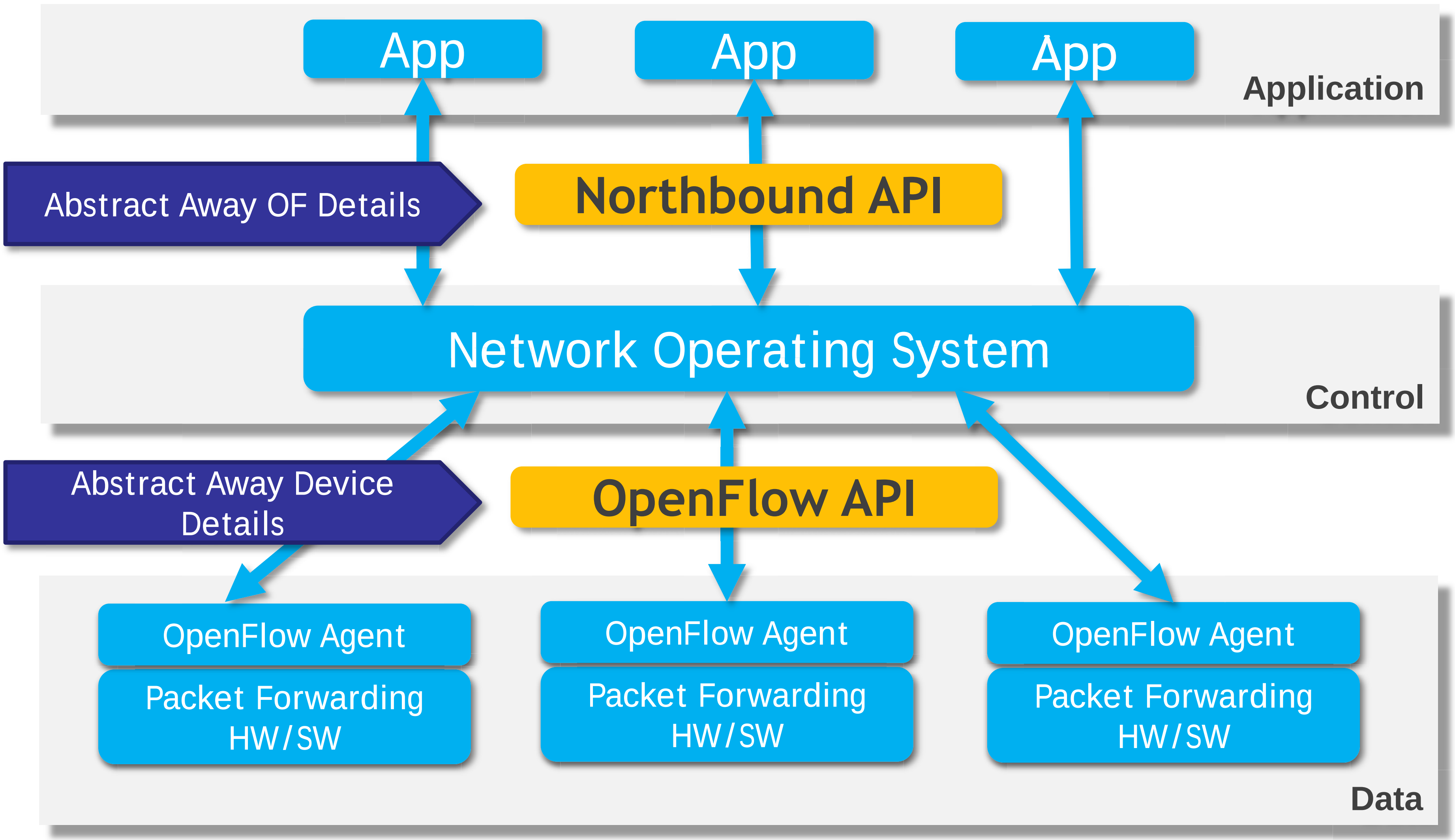
Why SDN

- Many believe that networks today are:
  - Hard to manage
  - Hard to evolve/change
  - Not designed on formal principals, thus hard to understand

- Watch Scott Shenker's Talk on SDN
  - http://www.youtube.com/watch?v=WabdXYzCAOU
  - "The root cause of this difficulty is the lack of proper abstractions in the control plane"



SDN: The Vision



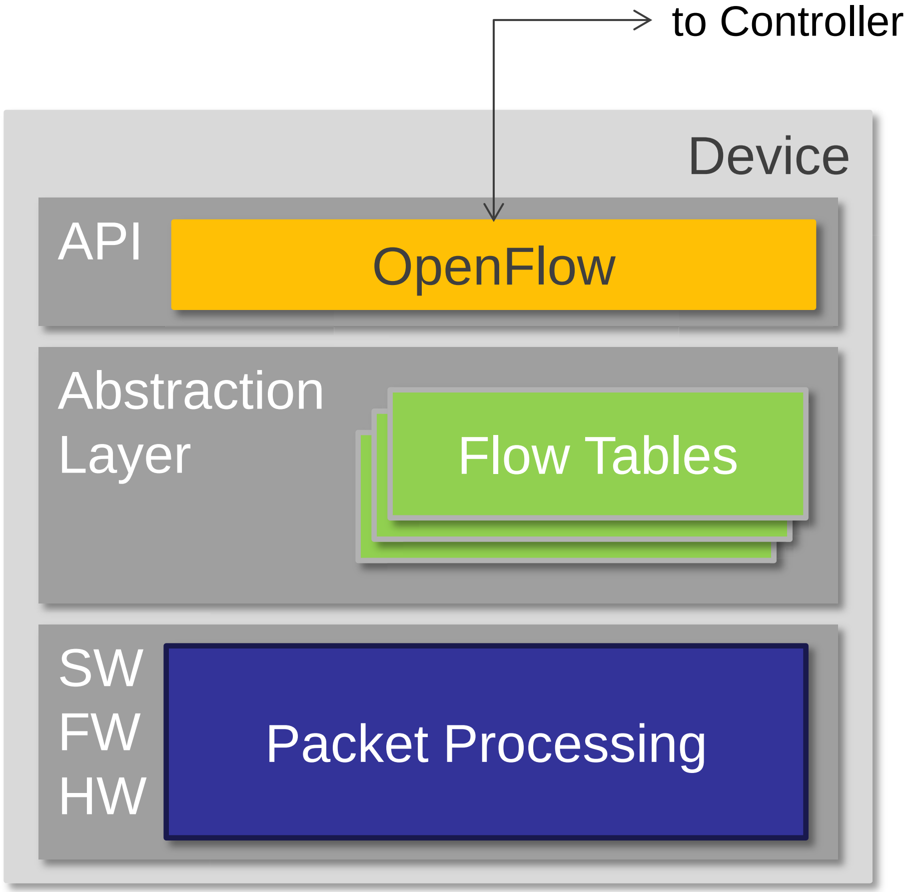
Anatomy of an SDN Device

API
- OpenFlow or proprietary

Abstraction Layer
- Flow Table(s)

Forwarding
- Forwarding tables (using L2, L3, L4 headers)
- TCAMs for fast searching on wildcard match fields

Operation
- Handle matching flows locally
- Forward non-matching flows to controller and await instructions



Anatomy of an SDN Controller

Northbound API:

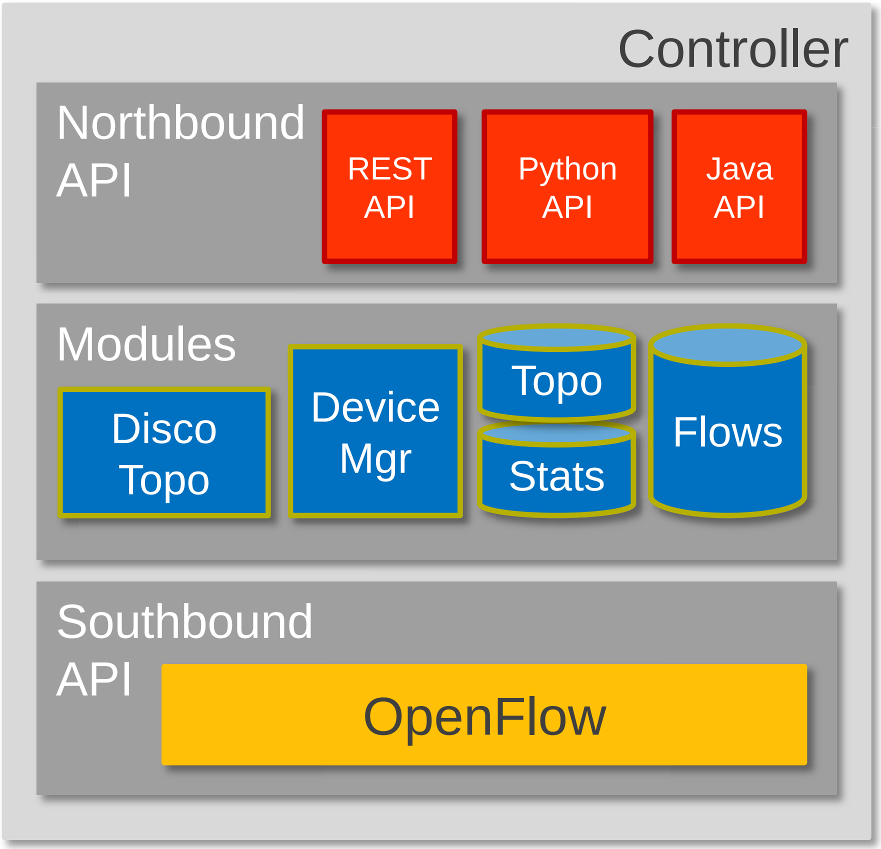
- Non-standard
- Java, Python, REST
- Handle Events (network, packet miss)
- Program Methods (program flows and actions)

Modules:

- Switch discovery & topology
- End user device manager

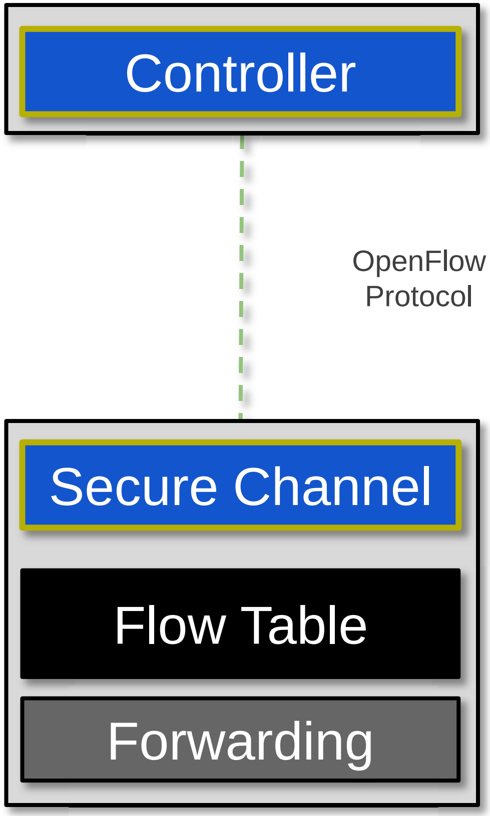
Southbound API:

- OpenFlow or proprietary, or both

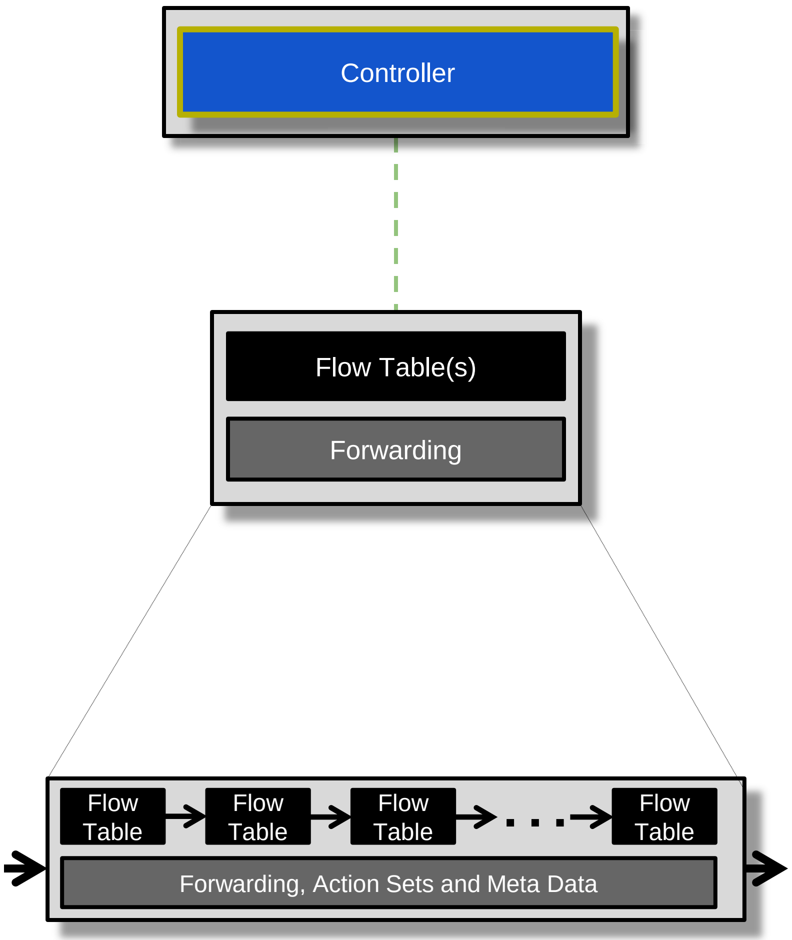


OpenFlow Communication

- OpenFlow Wire Protocol
  - Specified in the OpenFlow Switch Specification (latest version 1.4.0)
  - Implements a communication channel between device and controller typically using TCP (secured via TLS)
  - Can be initiated by either Controller or Switch. Allows for flexible separation of control and data planes.
  - Controller manages device and it’s flow table, receives events (packets) and injects packets



OpenFlow Basics: Flow entries and tables



Match Fields	Counters	Actions

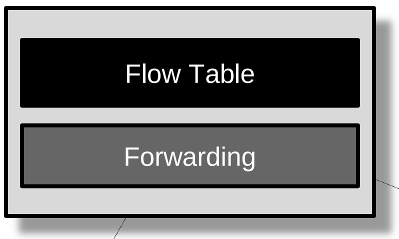
Flow Entries

- Match fields: *matching incoming packets*
- Counters: *keeping tally of packet matches*
- Instructions: *what to do if the packet matches*

Flow Tables

- Match: perform associated action/instruction
- No match: forward to controller
- Actions: Forward, Drop, Flood, Normal, ...
- Multiple flow tables may be arranged in a pipeline.

OpenFlow Basics: Match fields



OpenFlow building blocks:

- Matching fields
- e.g. MAC src/dst, IP src/dst, VLAN, TCP/UDP ports, physical switch port
- Allows wildcards

Ingress Port	MAC Src	MAC Dst	Eth Type	VLAN Id	VLAN Prior	IP Src	IP Dst	IP Prot	IP ToS	TCP/UDP sport	TCP/UDP dport

Common Criticisms

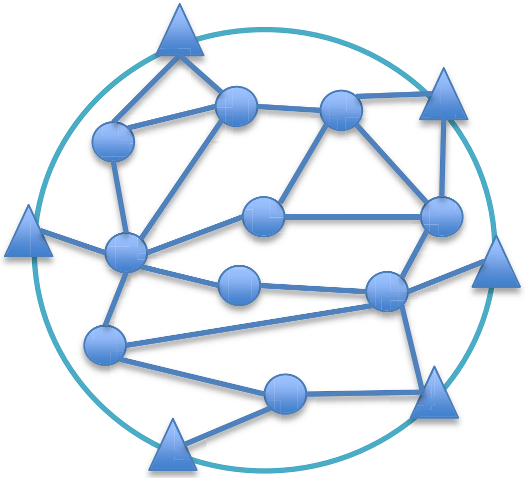
- SDN is too much change for my network…
- The central controller is a single point of failure…
- SDN is to slow and doesn’t scale…
- SDN Packet operations aren’t ‘rich’ enough to do what I want…

SDN Posturing… SDN Approaches

- SDN via existing APIs
  - Making traditional solutions more programmable
- SDN via Overlay Networks
  - Dynamic tunnels originating at the edge
- SDN via native OpenFlow Implementations
  - OpenFlow only mode
  - Hybrid mode

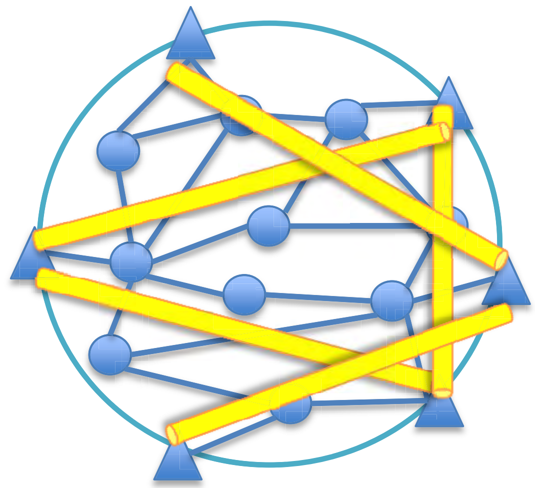
SDN: Potential Evolution

Phase 0
SDN via APIs



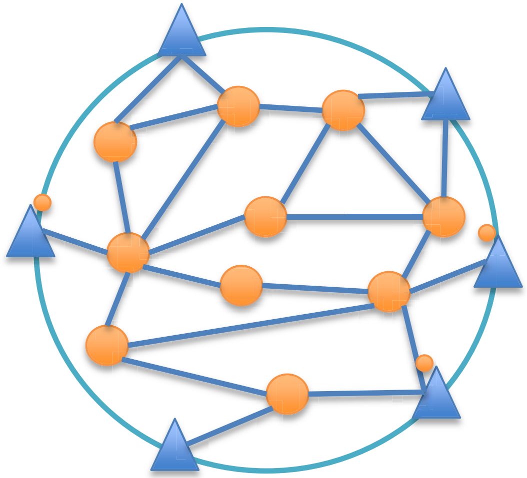
- Addresses some programmer needs on legacy environments
- Still mainly distributed control with hop-by-hop forwarding decisions
- Well known and proven
- Perceptions are slow to change, Inflexible, closed, proprietary, etc.

Phase 1
SDN Overlay



- Central control of tunnel end-points (e.g. in hypervisor). Data Center focused environments
- Independent of network fabric
- Software forwarding at edge
- Difficult multi-path, traffic engineering, diagnostics

Phase 2
Native SDN Devices



- Central control of forwarding
- High performance, multi-path, flexible, low-cost devices
- Aggregation (flow classifier) required in core of network
- Device upgrades, new ASICs needed to benefit scale, flexibility and performance

Environments for SDN

• **Datacenter**	Virtualization, multi-tenancy, recovery from failures, traffic engineering and load-balancing
• **WAN/Backbone**	Resiliency, reliability, determinism, traffic engineering and load-balancing
• **Campus**	Network access control, guest access, BYOD, monitoring malicious behavior
• **Security**	Firewalls, intrusion detection and prevention, blacklists, enforced quarantine
• **Wireless**	Mobile wireless backhaul, heterogenous wireless access, service chaining, load balanced packet core.

OpenFlow for Mobile & Wireless Networks

Serge Manning
Huawei

Open Networking Forum (ONF)

Board Members:
Google
Facebook
Microsoft
Yahoo
NTT Communications
Deutsche Telekom
Verizon
Goldman Sachs
Stanford University
UC Berkeley

ONF Board

Technical Leadership
- Executive Director (Dan Pitt)
- Technical Advisory Group
- Council of Chairs

Working Groups
- Arch & Framework
- Extensibility
- Config & Management
- Testing & Interop
- Forwarding Abstractions
- Market Education
- Optical Transport
- Migration
- Wireless & Mobile "WMWG"
- Northbound Interface

- Over 85 members
- Technical Advisory Group and Board review and approve all specifications
- Some of the WGs have different kinds of roles
- Wireless & Mobile and Northbound Interface created in October 2013

Wireless & Mobile WG (WMWG) Goals and Deliverables

- Mission
  - Identify enhancements to ONF technologies to improve operation of mobile and wireless networks.
    - ONF technologies include OpenFlow Switch Specification, OpenFlow-Config, and Northbound interfaces
- Participation
  - Started as a Discussion Group in March 2013, with more than 50 member companies signed up.
  - Active contribution (so far) by China Mobile, Huawei, NEC, Ceragon, NSN, Orange, Telefonica, and Goldman Sachs. So end users are engaged.
- Output of WG
  - Finalize Use Cases
    - During the Discussion Group phase, more than 15 use cases were collected
    - Merge/split & prioritize these use cases now
  - Dig deeper into Architectural and Protocol Requirements w.r.t. enhancing OpenFlow and other ONF technologies based on use cases
  - Work with other ONF WGs to realize architecture and protocol changes and support the corresponding running code
  - Help ONF support wireless & mobile technologies from other SDOs (like IEEE!)
    - WMWG will be careful not overlap or conflict with work in other SDOs

Current Use Cases
(work in progress)

- EPC: Separation of control and data planes and traffic steering
  - EPC control plane and SDN controller separate from data plane implemented by enhanced OpenFlow switches
    - Flexible and Scalable Packet Core
    - SDN Enhanced Distributed P/S-GW
    - Serving GW virtualization
  - Place and move the routing of flows through EPC data plane using enhanced OpenFlow while supporting the needs of the wireless network
    - Mobile Traffic Management (offloading)
    - Service Chaining in Mobile Service Domain
    - Network Based Mobility Management
    - SDN-Based Mobility Management in LTE
- Wireless Backhaul Optimization
  - Central SDN controller optimizes radio parameters in data plane using enhanced OpenFlow
    - Dynamic resource management for wireless backhaul (ACM)
    - Energy Efficiency in Mobile Backhaul Network
    - Security and Backhaul Optimization

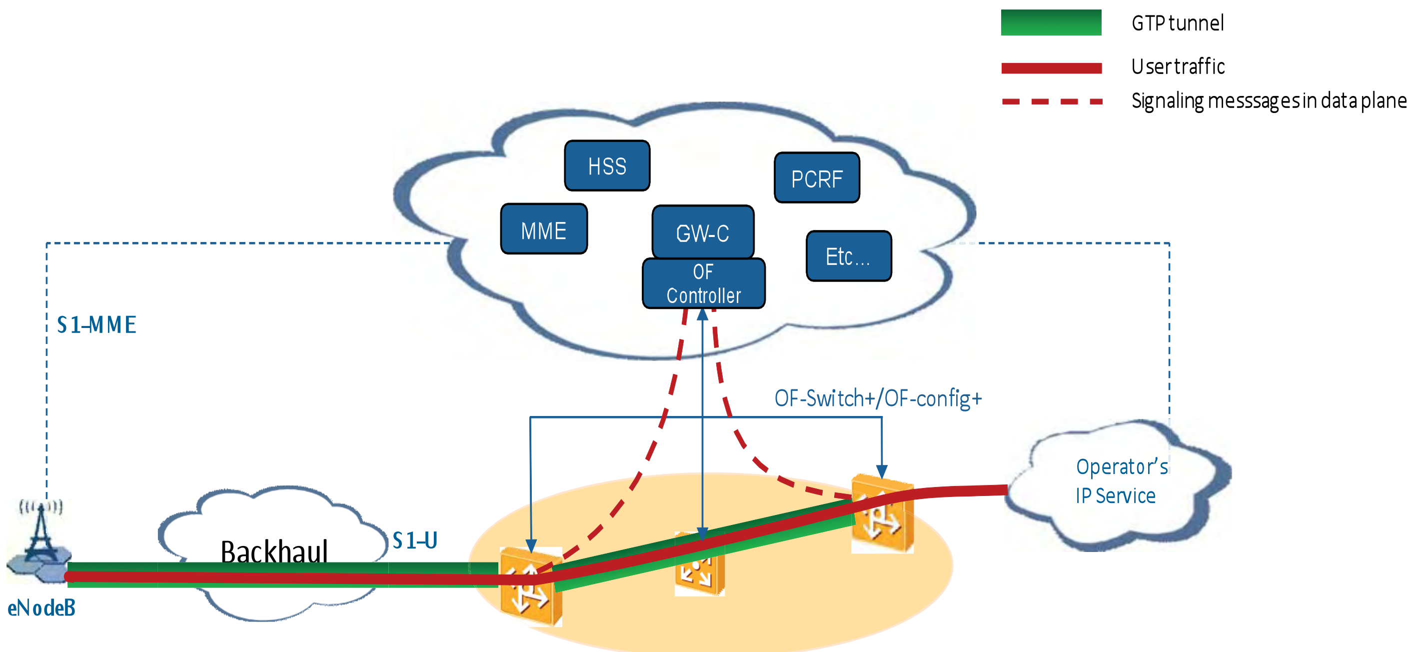
Current Use Cases (2)
(work in progress)

- IEEE-standards Related
  - Leverage work in IEEE 802 to manage devices and links using enhanced OpenFlow
    - IEEE 802.16r Connection-Oriented SDN for Wireless Small Cell Backhaul
    - IEEE 802.21 Media-Independent Handover
    - IEEE OmniRAN architecture for connecting multiple technologies
- Mobile Network Security
  - Enhance OpenFlow to support the security requirements of wireless networks
    - Management of secured flows in LTE

WMWG Projects

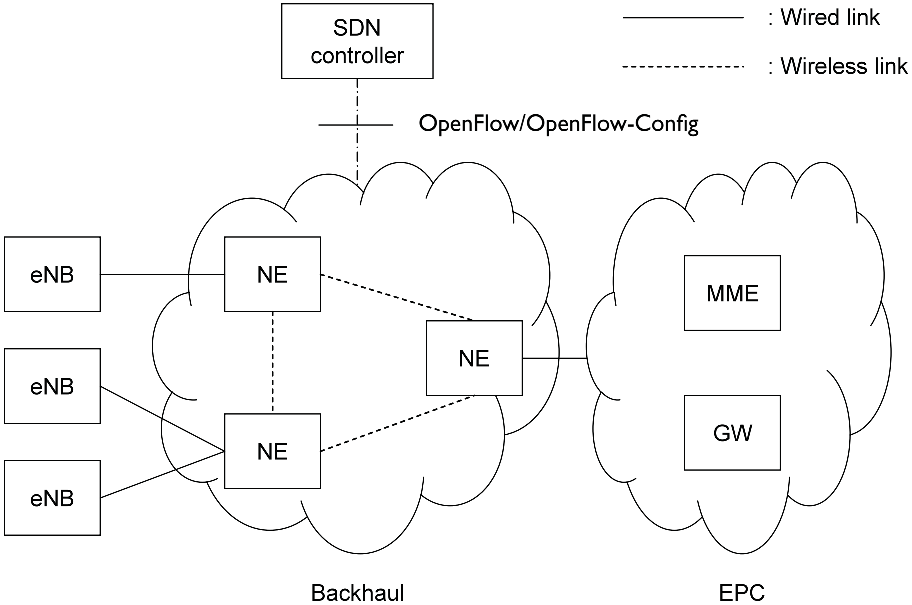
- Work within WMWG is divided into technical areas as Projects
- Completed Projects will include:
  - Set of use cases and scenario requirements
    - 1 month, end of year 2013
  - Architectural view and any new wireless impacts
    - 3-6 months
  - Protocol requirements
    - 6-12 months
  - Demo/Proof of concept
    - 12+ months
- Currently 3 Projects:
  - Cellular Evolved Packet Core (EPC)
  - Wireless Backhaul
  - Enterprise: Unified Management of Fixed/Wireless Networks

Cellular EPC



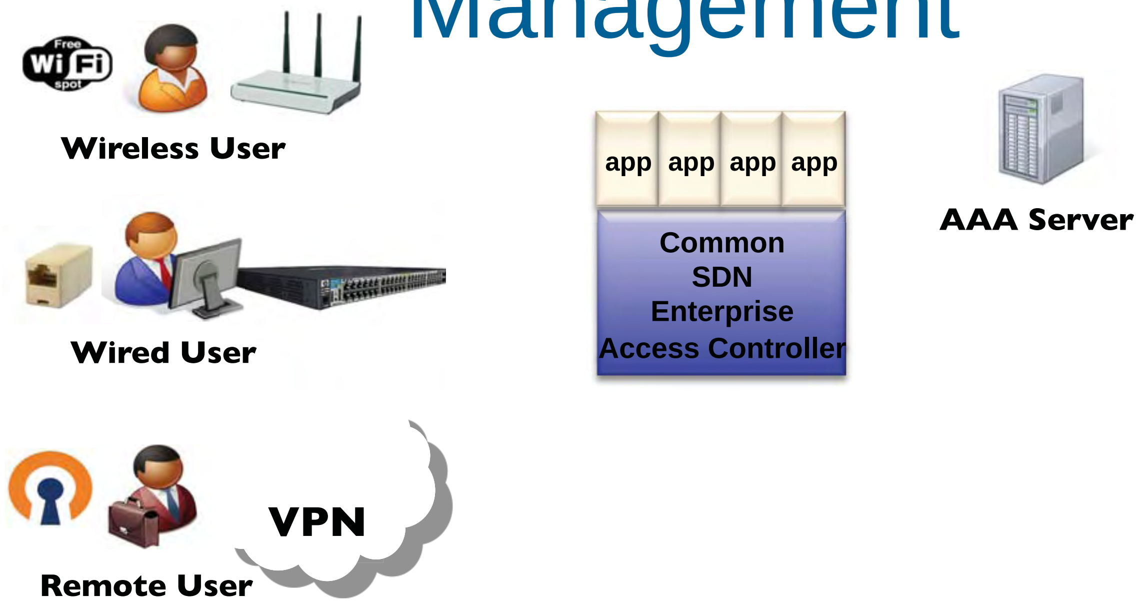
- EPC control plane and SDN controller separated from data plane implemented by OpenFlow switches
- Place and move the routing of flows through EPC data plane using OpenFlow while supporting the needs of the wireless network
- OpenFlow extensions may be required to support:
  - GTP/non-GTP tunneling, Policy Control, Online/Offline charging, and Lawful Interception

Wireless Backhaul – Dynamic Resource Management



- Backhaul with wireless transport links
- As demand for backhaul resources change, the SDN controller calculates the path and assigns the backhaul resources
  - Several parameters such as the SLA of the traffic and available backhaul resource may be taken into account (e.g., guaranteed resources for high SLA traffic)
  - When calculating the path, the SDN controller considers the availability of the link according to the modulation/capacity, and validates that there is enough capacity on the path
  - The SDN controller may periodically collect traffic statistics to estimate the actual throughput
- Can also accomplish other things such as Energy Efficiency

Enterprise: Unified Wired/Wireless Management



- Use same controller to manage both wired switches and wireless access points in standardized manner
  - SDN Controller leverages OpenFlow as well as other wireless management protocols
- A unified architecture and consistent means of managing across the enterprise:
  - User role/location based policy (who/where you are, what apps to access)
  - Rogue Detection (malicious or illegal devices connecting to network)
  - Troubleshooting/Monitoring (collect information centrally)
- Need to support
  - Strong authentication of endpoints, fast roaming, and IEEE 802 technologies
  - Potentially, use IEEE OMNIRAN architecture (R2 interface) to operate across access technologies

How Do IEEE Technologies Fit In?

- The goal is to enhance IEEE based networks using SDN and OpenFlow.
- Two approaches:
- (1) WMWG may start one or more IEEE specific Projects
  - Build upon work in IEEE 802.16r and/or 802.21
  - Introduce SDN elements based on architectures such as IEEE OmniRAN
- (2) Consider IEEE technology support into existing Projects
  - Probably for the Backhaul and/or Enterprise Projects
- Other choices?

Summary

- SDN/OpenFlow expanding out into wireless networks, IEEE 802 should be involved
- Potential IEEE Technologies to Consider
  - OMNIRAN, 802.16r, 802.21
  - Are there others?
- IEEE 802 participants could drive a new Project in ONF WMWG
  - The alternative is to try and address something in the existing Projects.
- Need help from IEEE 802 participants to make either happen
  - WMWG welcomes your input
  - How can ONF/WMWG help?

Operator’s Vision

Luis Contreras
Telefónica

What is the reality of an operator like Telefónica

- Telefónica is an operator present in 24 countries across Europe and Latam with more than 317,3 million customer
- This means that Telefónica is …
  - ***Multinational***: different regulations and tax policies (applicable to network infrastructure, spectrum usage and cost, etc)
  - ***Multicultural:*** different ways of consuming services leading to a different use of network resources
  - ***Multiservice:*** fixed, mobile, enterprise, convergent
  - ***Multi-role:*** incumbent, alternative (typically renting network capacity)
  - ***Multi-origin:*** own new branded, acquisitions
  - ***Multivendor:*** great variety of providers in network equipment, OSS/BSS, etc
  - ***Multi-technology:*** great diversity of access and transport technologies

This reality translates into …

- Complex procedures for service delivery
- Difficulties on defining and standardizing services across the group
- High customization during service creation
- Slow adaptation of the network to changing demands
- Long time-to-market

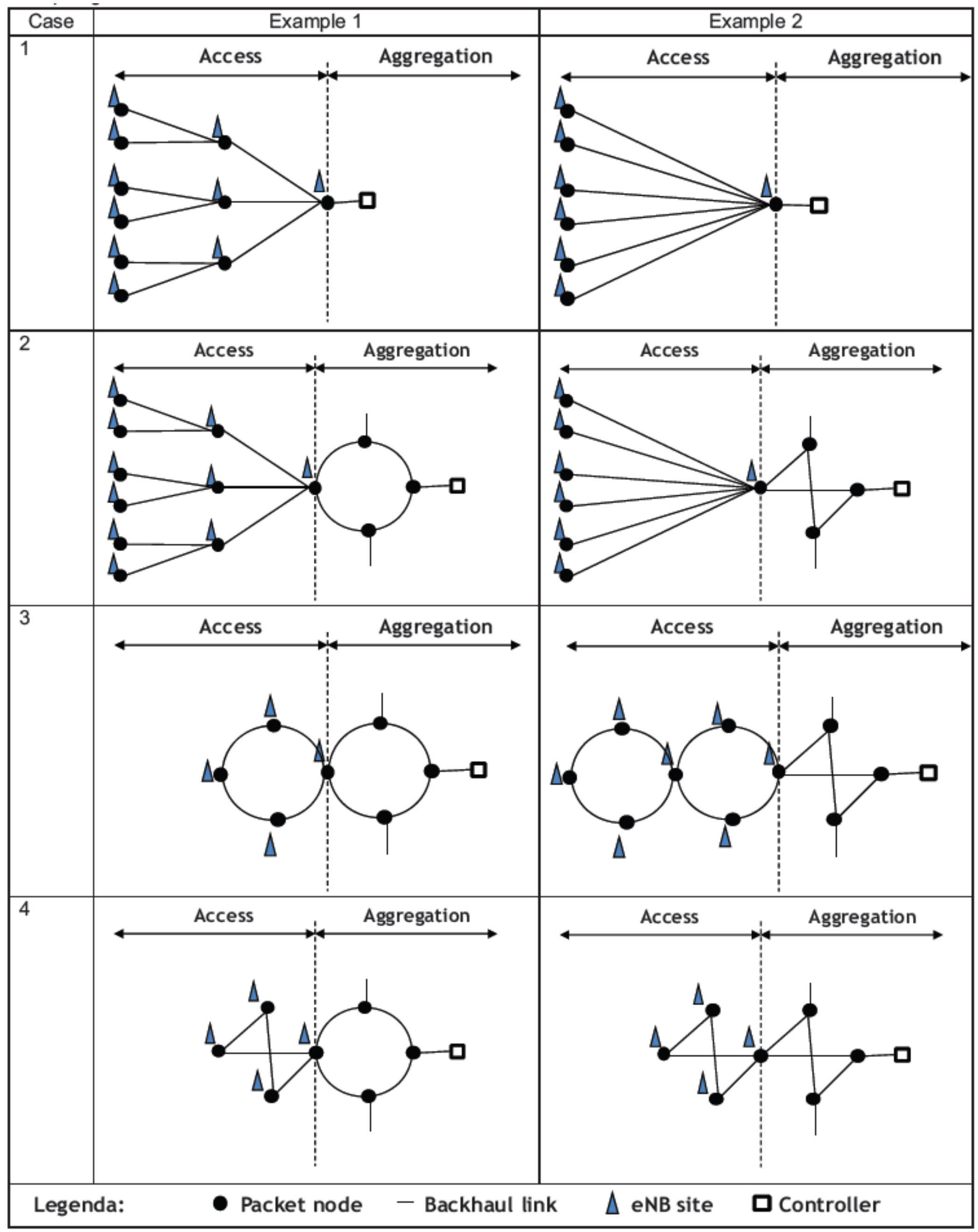
SDN as a way towards network programmability

- Existing service innovation cycles are tightly coupled to the product innovation cycles
  - High variety of vendors in operators like Telefónica, each of them presenting different operating systems, SDKs and APIs, in multiple node families and functionalities
  - Closed solutions per vendor, difficult to port
- Lack of automation in service delivery
  - Complex procedures for service scaling in and out

- Network programmability as the capability of installing and removing network behavior, in real time
  - This is not just to populate software line code to simple switches or offering APIs
  - End-to-end network abstraction is required for true technology and vendor integration
- Network services to be realized by programming instead of rearchitecting the network
  - Leveraging on existing and deployed network capacities (control plane functionalities)
  - Managing the network in an integrated/coordinated way, not as a collection of individual boxes/layers
  - Stress on service modeling and network modeling, lately propagated through standard interfaces (Netfconf, Yang, OpenFlow) cooperating with existing control plane capabilities (GMPLS, etc.)

Case description
Mobile backhaul

- Mobile service provided over
  - Multi-technology network
    – Optical, microwave, Ethernet
  - Multi-topology architecture
    – Ring, star



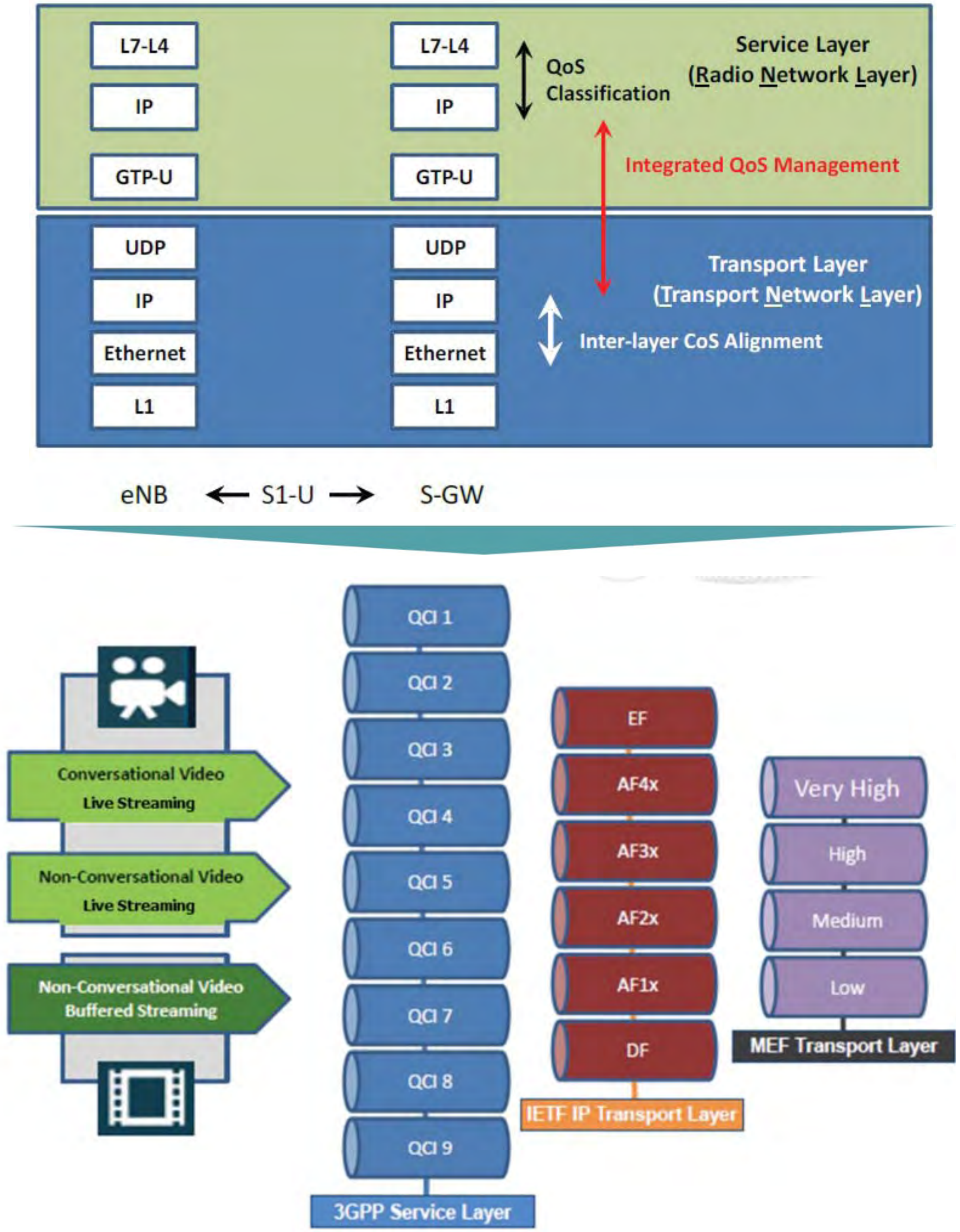
Source: NGMN, *"LTE Backhauling Deployment Scenarios"*

Case description
Mobile backhaul

- Need for providing homogeneous service QoS over heterogeneous transport networks

- Need for dynamically controlling network resources according to radio conditions

  ✓ For instance, adaptive modulation in a wireless backhaul path could impact the network resources for a flow both in the radio access and the aggregation switches in backhaul



Source: NGMN, *"Integrated QoS Management"*

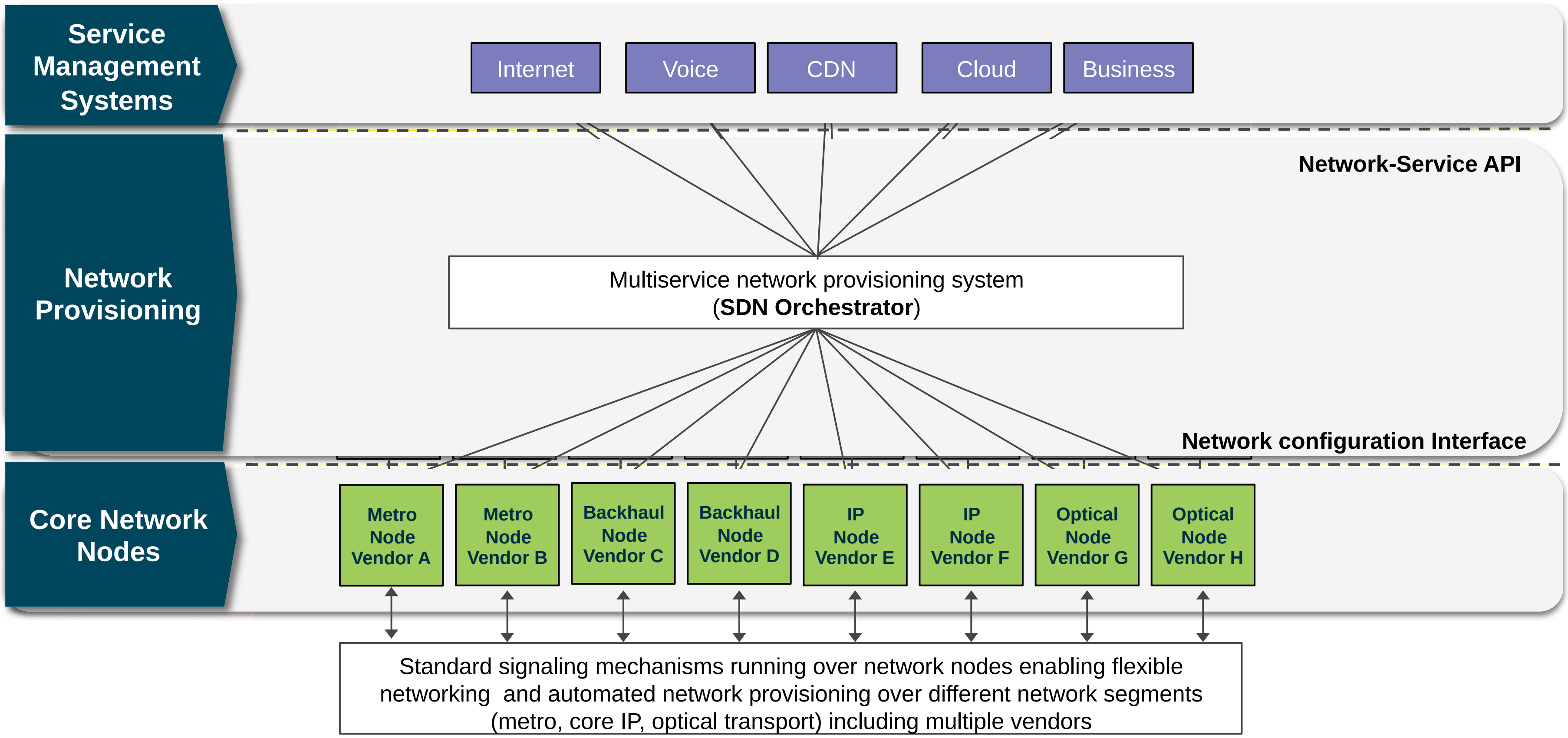
Case description
Mobile backhaul

- SDN control will allow the orchestration of the network resources
  - Separate, independent layers do not allow overall optimization
    - Multilayer approach for performing combined optimization
  - Separated domains complicate the service provisioning and network adaptation
    - Interconnection of controllers for e2e optimization
  - Separated control and management mechanisms require multiple interventions
    - Standard interfaces simplify heterogeneous device management

Some other cases

- Network sharing
  - The higher granularity of the access networks make them subject of higher investments
  - Sharing network infrastructures with other operators reduce that investments
  - Agile and flexible mechanisms for sharing network resources are needed
- Traffic offloading / steering
  - Variations on traffic load, network status, or service node location could lead to flow redirection over vacant infrastructure

Carrier SDN - Unified network provisioning architecture



- Key building blocks for unified network provisioning architecture are:
  - **Network configuration interface**: Multivendor edge nodes configuration (e.g. OLT and BRAS, IP core routers, optics and microwave transport in backhaul, radio access, etc.) by standard interfaces (e.g. OpenFlow)
  - **IT and network SDN orchestration**: Coordinated network and datacenter resources control according to service
  - **Network-Service API**: Application level API hiding details of the network

Expected benefits

- Lower OpEx thanks to the automation of provision processes in different layers and domains, in a highly dynamic environment
- Lower CapEx because of device simplification and a better usage of network resources respect to what is observed today
  - E.g., IP: traffic being send through congested shortest path links
  - E.g., Ethernet: no traffic on blocked links due to spanning tree algorithm
- Application awareness thanks to the fine-grained flow management
  - It can take advantage of network analytics for a (logically) centralized traffic engineering
- Network abstraction to homogenize services, procedures and operations

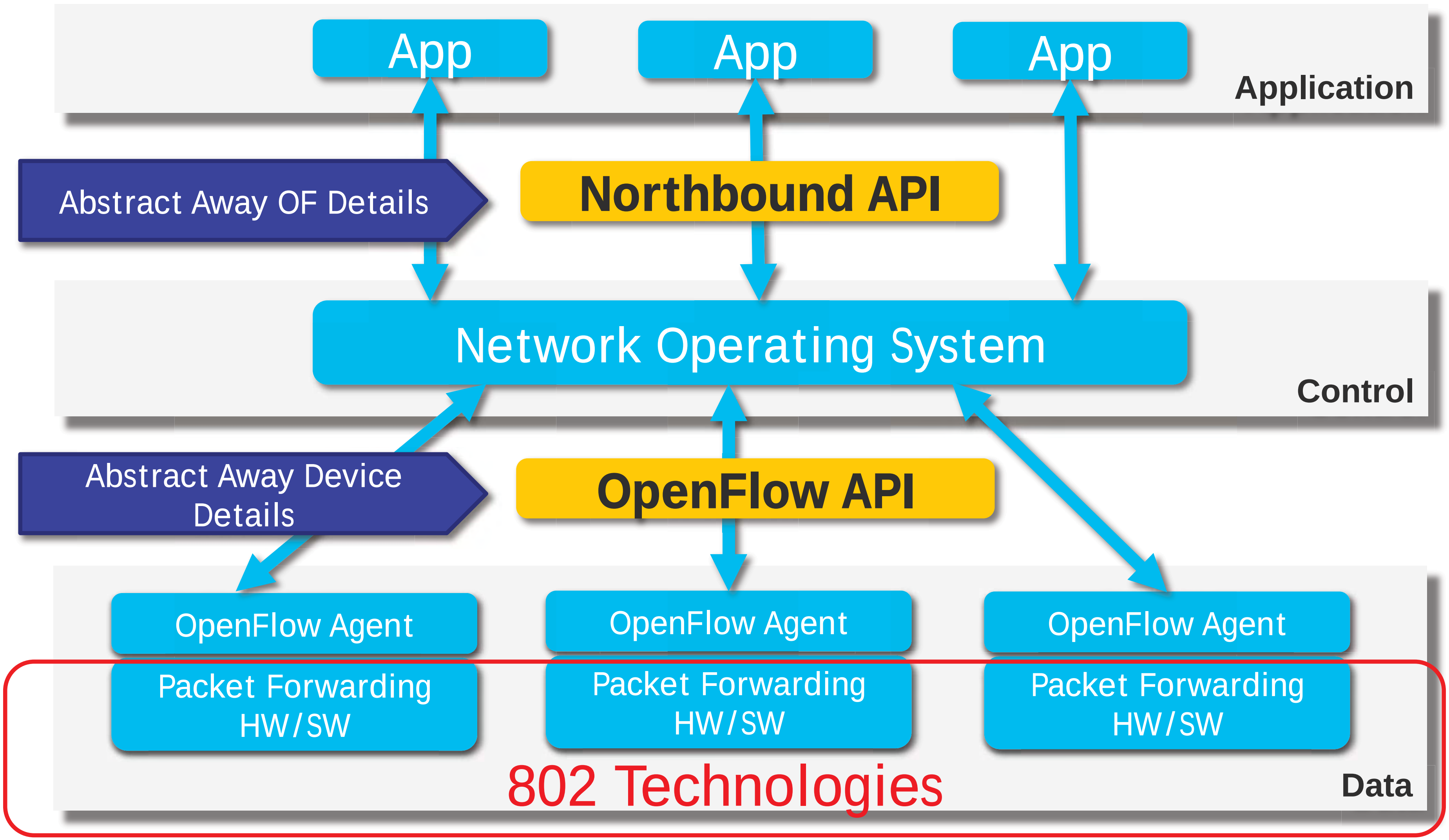
SDN challenges

- Fine-grained management of network flows
  - New possibilities arise for a richer management of traffic flows to provide advanced functionalities (e.g., offloading, caching, etc.)
  - Opportunity for achieving a better network utilization
  - Fixed/Mobile mix of traffics with distinct QoS and different impact in terms of network costs and revenue
- SDN-based control mechanisms
  - Scalability of the decoupled control elements
  - Reliability of a logically centralized system
- Co-existence with fixed and mobile legacy infrastructure
  - Put in value the control plane in current network equipment already deployed e.g. E2E MPLS (integration vs. substitution)
  - Interoperability with conventional equipment in the field (IP routing, MPLS signaling, OAM mechanisms, etc.)
- Enabler for network virtualized functions
  - Rapid programmatic re-configuration to support de-localized network functions from fixed and mobile (even moving network functions to the cloud)
- New forms of operating the network
  - Organizational changes and new skills are required

SDN in IEEE 802 Network Reference Model

Juan Carlos Zúñiga
InterDigital Communications

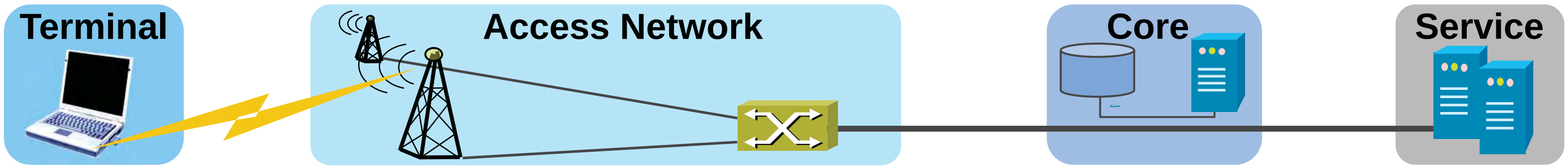
SDN Network Model

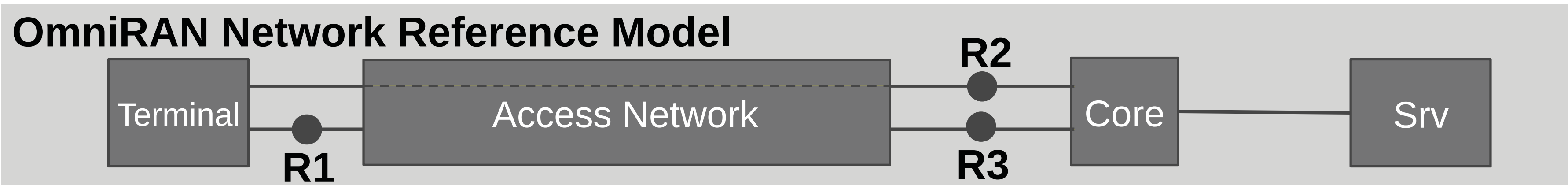


IEEE 802 OmniRAN Project Proposal

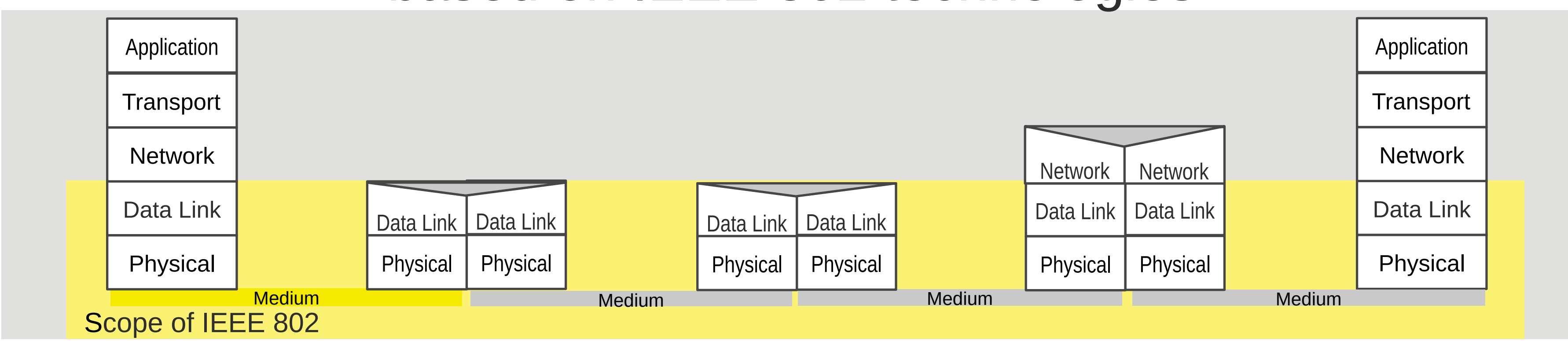
- Scope
  - Network Reference Model and Functional Description of IEEE 802 Access Network
    - Based on the family of IEEE 802 Standards
    - Including entities and reference points along with behavioral and functional descriptions of communications among those entities (e.g. Stage-2)
  - Anticipating active participation from existing 802 WGs
- Use cases considered by Study Group
  - **SDN-based Model**
  - Smart Grid
  - 3GPP WLAN EPC access, etc
- Status
  - PAR/5C being considered by 802 EC this week

Access Network Abstraction by OmniRAN

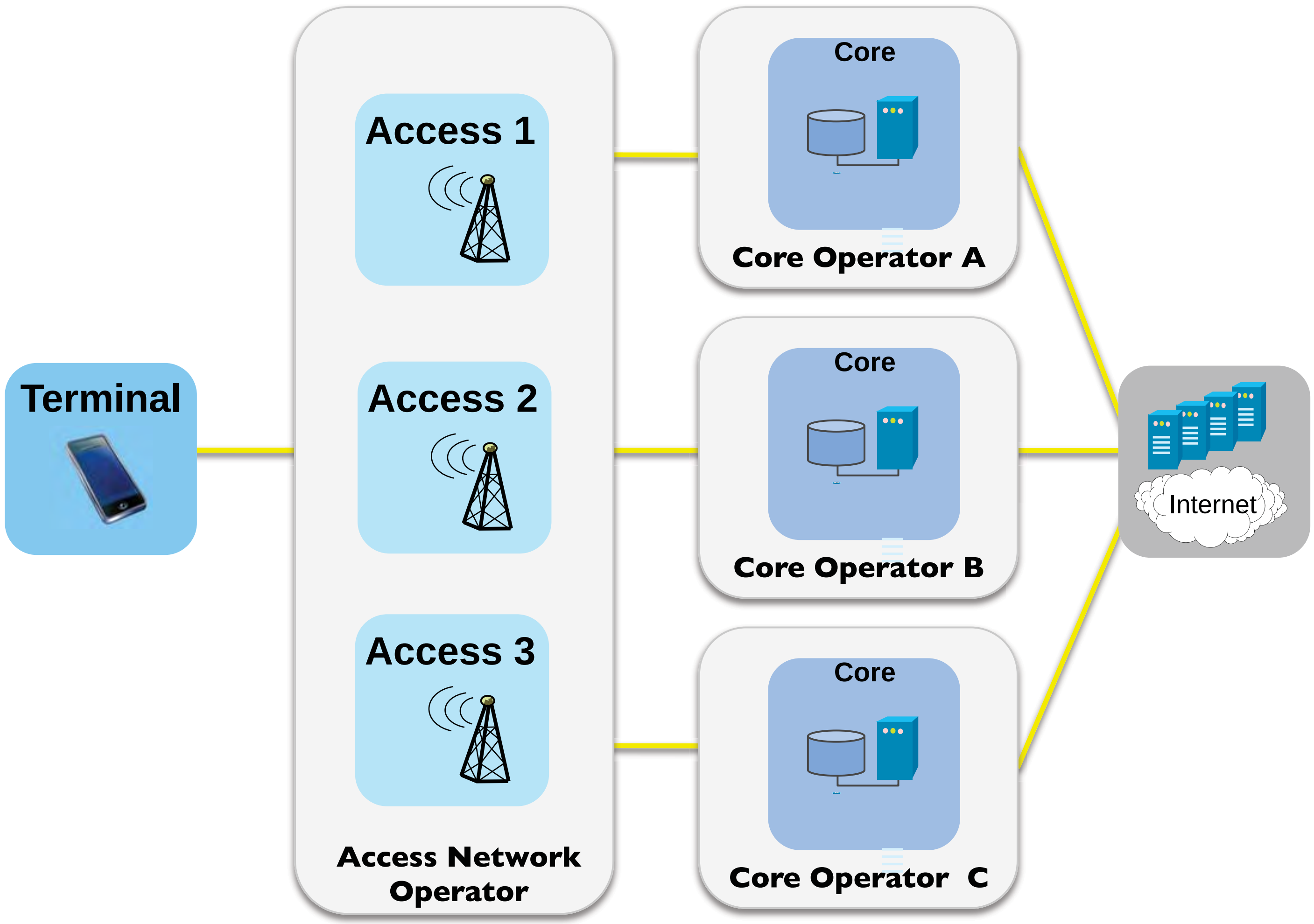




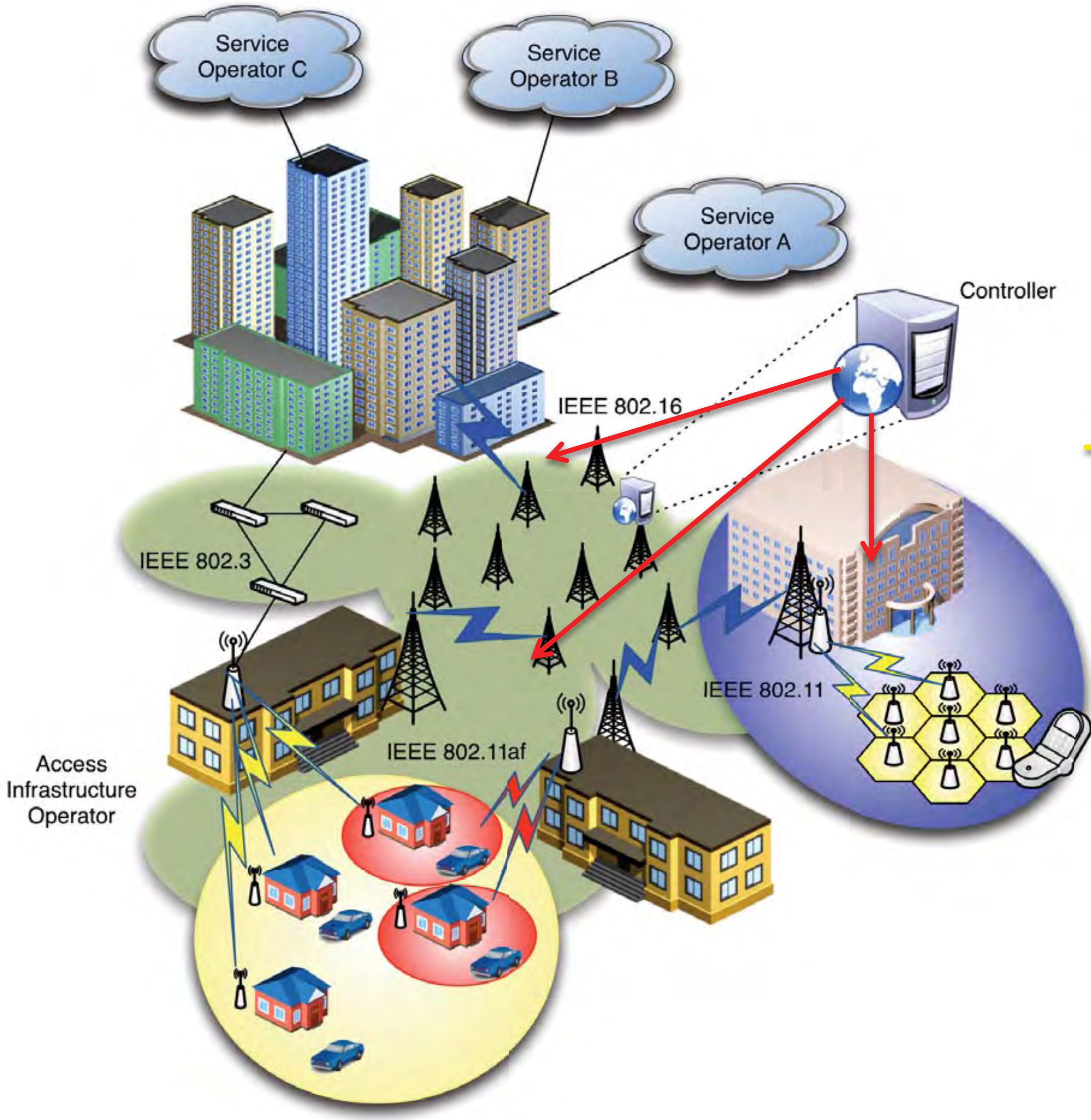
OmniRAN provides a generic model of an access network based on IEEE 802 technologies



OmniRAN Network Model with multiple Cores

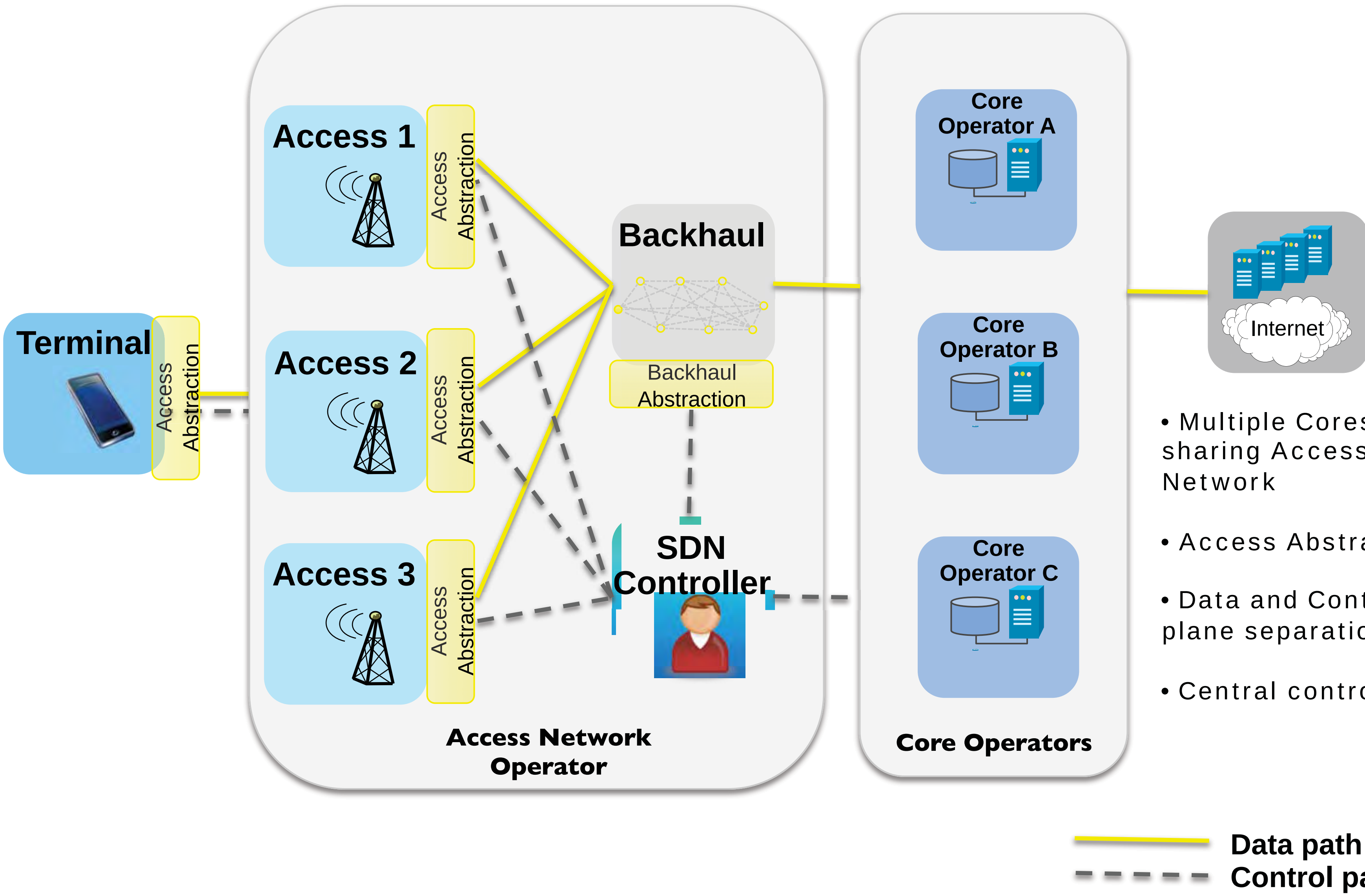


SDN-based OmniRAN Use Case



- Centrally controlled configuration, from Core to Terminal, of heterogeneous IEEE 802 links
- Dynamic creation of data paths with dynamic reconfiguration and mapping to the terminal at flow granularity
- Clean separation of data and control planes

SDN-based OmniRAN Network Model



- Multiple Cores sharing Access Network
- Access Abstraction
- Data and Control plane separation
- Central control

IEEE 802 functionalities for SDN control

- Control of data forwarding plane, common to 802 technologies
  - Southbound interface enabling the communication between the 802 technologies and the central controller (e.g. access abstraction)
  - Clearly defined interfaces, SAPs (APIs) and behaviors
  - Ability to modify data path and link parameters based on arbitrary but bounded selection parameters
    - Packet classification mechanisms based on templates (á la OpenFlow)
    - End-to-end packet flow and QoS
- Radio configuration mechanism for access and backhaul links
  - With defined metrics and reporting
- Data plane management of Terminals with multiple-interfaces
  - Notion of 802 logical interface facing L3
- Generic 802 access authorization and attachment

SDN in Wireless Access

Antonio de la Oliva
Universidad Carlos III de Madrid

SDN in Wireless Access

- These slides focus on discussing how the SDN concept can be applied to the access network
- We focus on IEEE 802.11, although similar ideas can be applied to other technologies for the access
- Topics to be tackled:
  - Current OpenFlow support of WLAN
  - Key issues
  - Related work

OpenFlow support of IEEE 802.11 Interfaces

- Currently OpenFlow can be used to control an IEEE 802.11 interface, but with some limitations
- The properties of the port as seen by the controller are unknown:

```
>>dpctl show tcp:192.168.20.3:6633
features_reply (xid=0xd03029cb): ver:0x1, dpid:2320e7afd5
n_tables:2, n_buffers:256
features: capabilities:0xc7, actions:0xeff
 1(wlan0): addr:00:1c:10:44:32:ca, config: 0, state:0
 2(eth0.0): addr:00:1c:10:44:32:c8, config: 0, state:0
     current:    100MB-FD COPPER AUTO_NEG
     advertised: 10MB-HD 10MB-FD 100MB-HD 100MB-FD AUTO_NEG AUTO_PAUSE AUTO_PAUSE_ASYM
     supported:  10MB-HD 10MB-FD 100MB-HD 100MB-FD AUTO_NEG
 3(eth0.1): addr:00:1c:10:44:32:c8, config: 0, state:0
     current:    100MB-FD COPPER AUTO_NEG
     advertised: 10MB-HD 10MB-FD 100MB-HD 100MB-FD AUTO_NEG AUTO_PAUSE AUTO_PAUSE_ASYM
     supported:  10MB-HD 10MB-FD 100MB-HD 100MB-FD AUTO_NEG
 4(eth0.2): addr:00:1c:10:44:32:c8, config: 0, state:0
     current:    100MB-FD COPPER AUTO_NEG
     advertised: 10MB-HD 10MB-FD 100MB-HD 100MB-FD AUTO_NEG AUTO_PAUSE AUTO_PAUSE_ASYM
     supported:  10MB-HD 10MB-FD 100MB-HD 100MB-FD AUTO_NEG
 5(eth0.3): addr:00:1c:10:44:32:c8, config: 0, state:0
     current:    100MB-FD COPPER AUTO_NEG
     advertised: 10MB-HD 10MB-FD 100MB-HD 100MB-FD AUTO_NEG AUTO_PAUSE AUTO_PAUSE_ASYM
     supported:  10MB-HD 10MB-FD 100MB-HD 100MB-FD AUTO_NEG
get_config_reply (xid=0x3f4af6c): miss_send_len=128
```

OpenFlow support of IEEE 802.11 Interfaces

- Current specification (OF 1.4) defines three kind of port properties: Ethernet, Optical and Experimental.
  - No concept of wireless properties
  - Way behind Ethernet or Optical ports

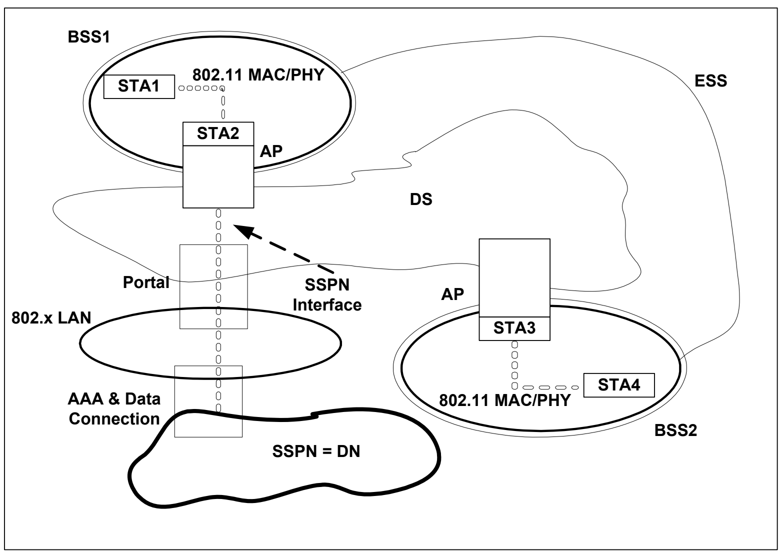
```
/* Optical port description property. */
struct ofp_port_desc_prop_optical {
        uint16_t type; /* OFPPDPT_3OPTICAL. */
        uint16_t length; /* Length in bytes of this property. */
        uint8_t pad[4]; /* Align to 64 bits. */
        uint32_t supported; /* Features supported by the port. */
        uint32_t tx_min_freq_lmda; /* Minimum TX Frequency/Wavelength */
        uint32_t tx_max_freq_lmda; /* Maximum TX Frequency/Wavelength */
        uint32_t tx_grid_freq_lmda; /* TX Grid Spacing Frequency/Wavelength */
        uint32_t rx_min_freq_lmda; /* Minimum RX Frequency/Wavelength */
        uint32_t rx_max_freq_lmda; /* Maximum RX Frequency/Wavelength */
        uint32_t rx_grid_freq_lmda; /* RX Grid Spacing Frequency/Wavelength */
        uint16_t tx_pwr_min; /* Minimum TX power */
        uint16_t tx_pwr_max; /* Maximum TX power */
};
```

- Optical properties recently added (OF 1.4, Oct'13)
- It allows the controller to configure physical properties of the port, such as Tx power or wavelength used
- It can also monitor this properties
- Other actions such as turning iface on/off are also supported

4

Problems of the OpenFlow view of WLAN

- OpenFlow sees the IEEE 802.11 interface as an unknown interface connected to the switch
- The real view of the switch corresponds, in the IEEE 802.11 architecture, to the portal

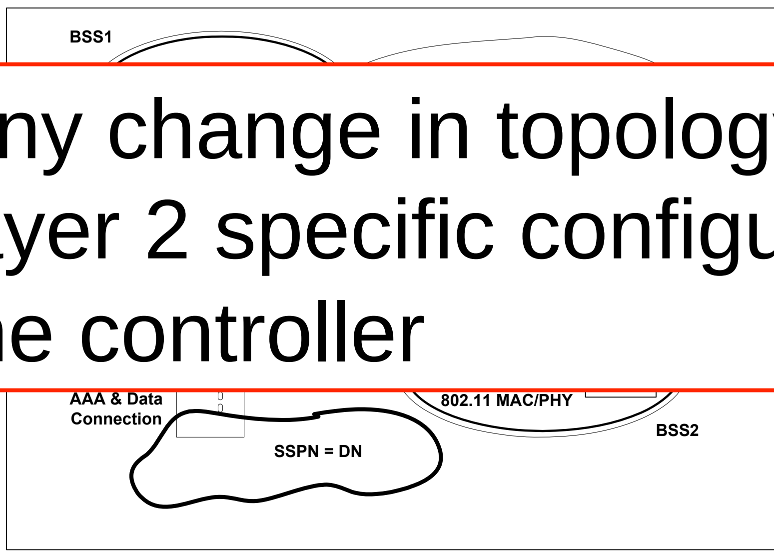


From IEEE 802.11-2012:
"The most important distinction is that a portal has only one "port" (in the sense of IEEE Std 802.1D, for example) through which it accesses the DS. This renders it unnecessary to **update bridging tables inside a portal** each time a STA changes its association status. In other words, the details of distributing MSDUs inside the IEEE 802.11 WLAN need not be exposed to the portal."

Note that IEEE 802.11ak and IEEE 802.1Qbz are working on improving the connection among IEEE 802.11 and bridged systems

Problems of the OpenFlow view of WLAN

- OpenFlow sees the IEEE 802.11 interface as an unknown interface connected to the switch
- The real view of the switch corresponds in the IEEE 802.11 architecture to the portal



From IEEE 802.11-2012:

Any change in topology, mobility, QoS, or any layer 2 specific configuration is hidden from the controller

Specific Issues

- Match fields do not consider IEEE 802.11 specific frame format:
  - OpenFlow defines operation in the switches based on matching and actions.
    - E.g., If packet comes from port X, apply VLAN Y and forward through port Z
  - Match filters defined for fields of Ethernet frame, IPv4/IPv6 packet, TCP segments, ARP frames, ICMP, MPLS..

Specific Issues

– Matching of specific Ethernet frame fields

Preamble	S F D	Dst Address	Src Address	VLAN TAG	Type/ Length	Payload	CRC

Field	Bits	Mask	Pre-requisite	Description
OXM_OF_ETH_DST	48	Yes	None	Ethernet destination MAC address
OXM_OF_ETH_SRC	48	Yes	None	Ethernet source MAC address
OXM_OF_ETH_TYPE	16	No	None	Ethernet type of the OpenFlow packet payload, after VLAN tags.
OXM_OF_VLAN_VID	12+1	Yes	None	VLAN-ID from 802.1Q header. The CFI bit in- dicate the presence of a valid VLAN-ID, see be- low.
OXM_OF_VLAN_PCP	3	No	VLAN VID!=NONE	VLAN-PCP from 802.1Q header.

Specific Issues

- Currently there is no support for the matching of IEEE 802.11 specific fields (Portal view)

IEEE 802.3

Preamble	SFD	Dst Address	Src Address	VLAN TAG	Type/ Length	Payload	CRC

Critical aspects:

- Matching of addresses
- BSSID (RAN Sharing)
- Type of Payload
- No management or control frames are visible

IEEE 802.11

Frame Control	Duration /ID	Address 1	Address 2	Address 3	Seq. Control	Address 4	QoS Control	HT Control	Payload	CRC

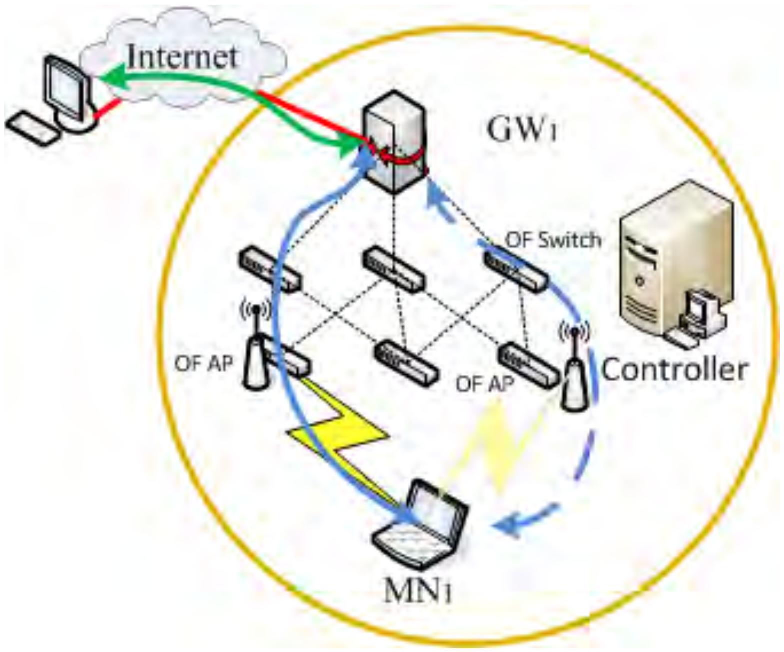
Protocol Version	Type	Subtype	To DS	From DS	More Fragments	Retry	Power Management	More Data	Protected Frame	Order

Specific Issues

- Configuration of interfaces
  - OpenFlow supports an extension to configure port properties
  - OF-CONFIG protocol supports:
    - current-speed
    - no-recv
    - no-fwd
    - no-packet-in
    - link-down
    - blocked
    - live
    - speed
    - duplex-mode
    - copper-medium
    - fiber-medium
    - auto-negotiation
    - pause
    - asymmetric-pause
- Wireless Interfaces have huge variety of properties that can be configured
  - Radio properties (channel, tx power, RSSI levels)
  - Transmission characteristics (number of antennas, high throughput characteristics)
  - MAC layer (QoS, groupcast)
- Some other SDOs have already look at this, e.g., CAPWAP at IETF
  - ONF W&M studying a CAPWAP use case

Specific Issues

- Mobility at the WRAN is hidden from the OpenFlow point of view
- Use of SDN approaches can benefit RAN deployments by improving their flexibility
- Need of matching rules for STA attachment, currently done by SNMP traps
- Extensions for handling terminal behavior



11

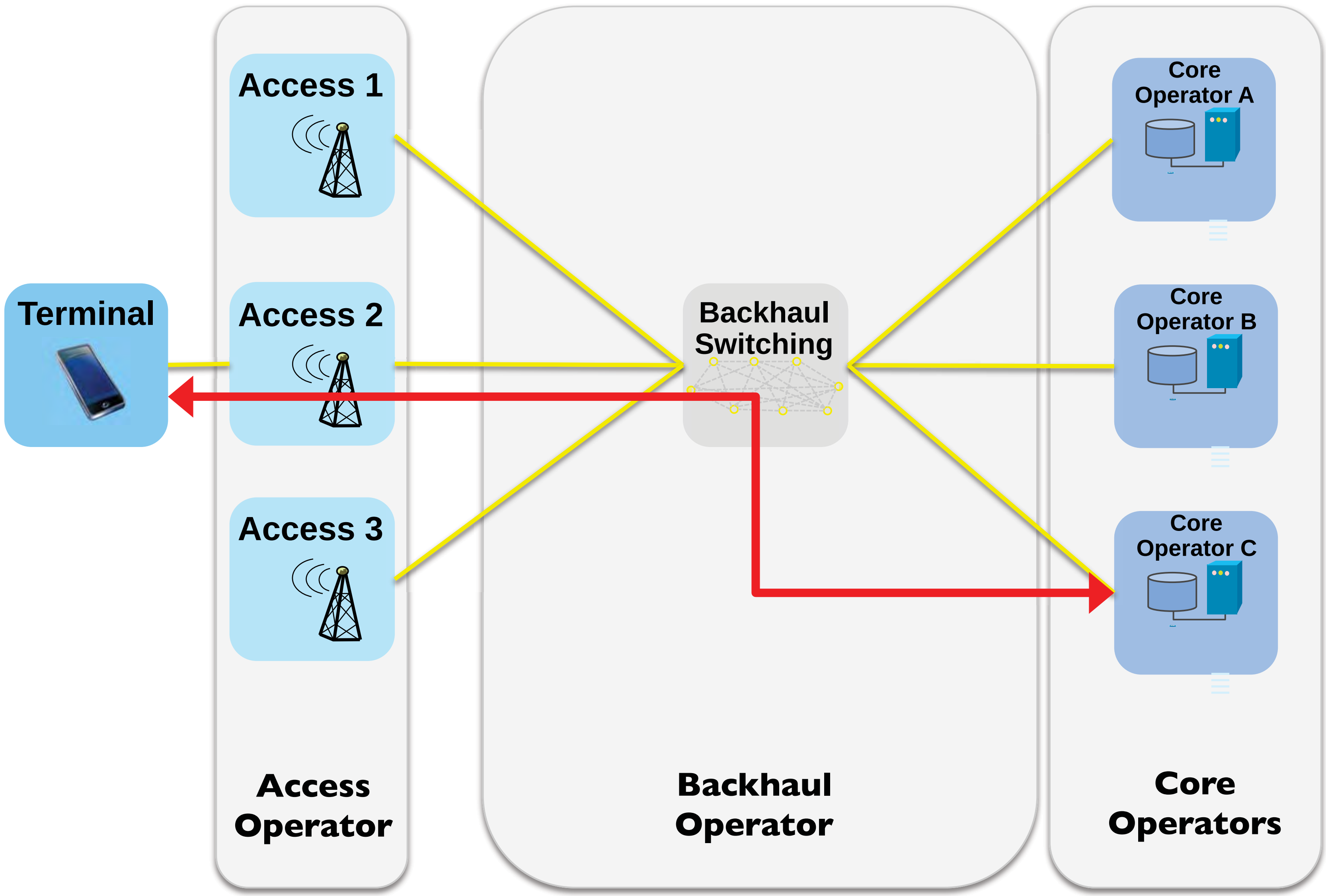
Summary of issues

- OpenFlow currently lacks of a real view of a wireless interface
- Seen as unknown Ethernet port, without properties
- No specific match rules, no specific actions, no access to wireless management or control
- Extending this support can greatly expand the flexibility of systems:
  - Implementation of e.g., ANQP/GAS as an SDN service, no firmware update, instantaneous support in a network
  - Other examples: PMIP/DMM deployments

SDN in Wireless Backhaul Service

Roger Marks
EthAirNet Associates

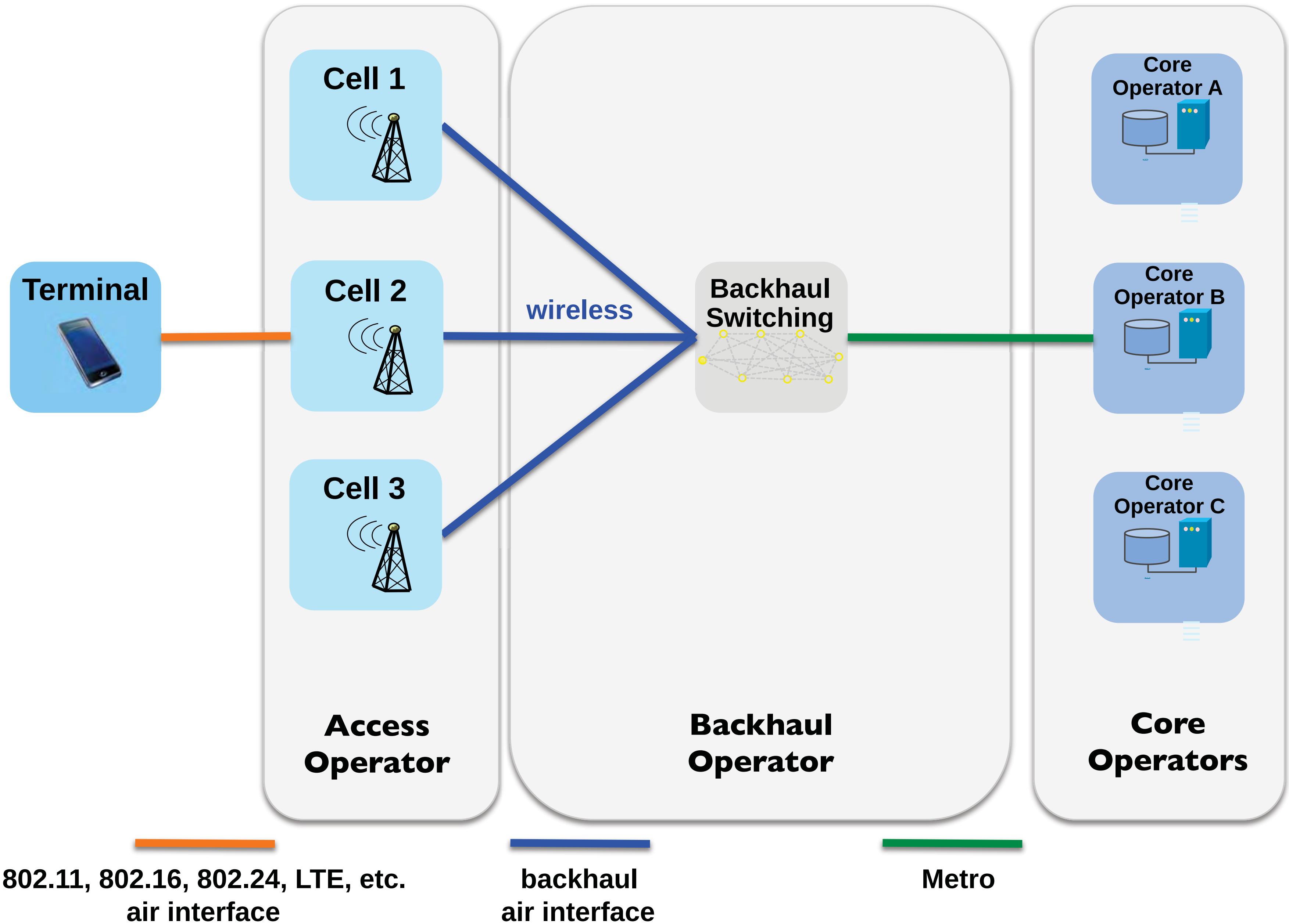
Shared Backhaul and Access



Wireless Backhaul Service Demands

- small cells
  - small cells (LTE, Wi-Fi, etc.) are needed to address local end-user capacity demands
  - small cell backhaul must be ubiquitous (and therefore wireless)
- Backhaul may be shared
  - backhaul service to multiple mobile operators simultaneously
  - possibility of independent backhaul service provider
- Access may be shared
  - access service to terminals of multiple mobile operators simultaneously
  - possibility of independent access service provider
- Capacity must be shared
  - capacity sliced among mobile operators and services, with SLA commitments
  - customer traffic is comprised of multiple flows with varying QoS requirements to satisfy

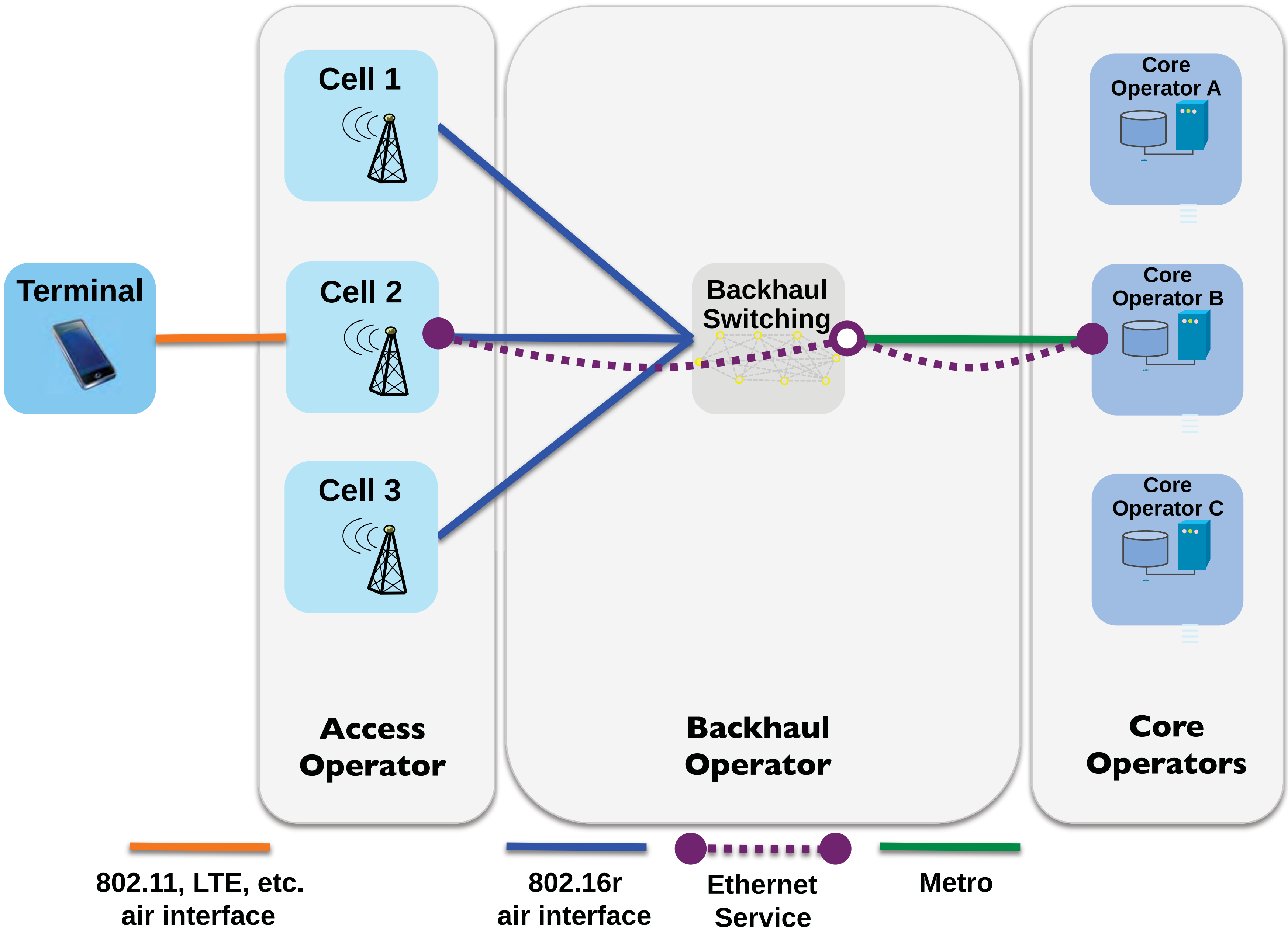
Wireless Backhaul



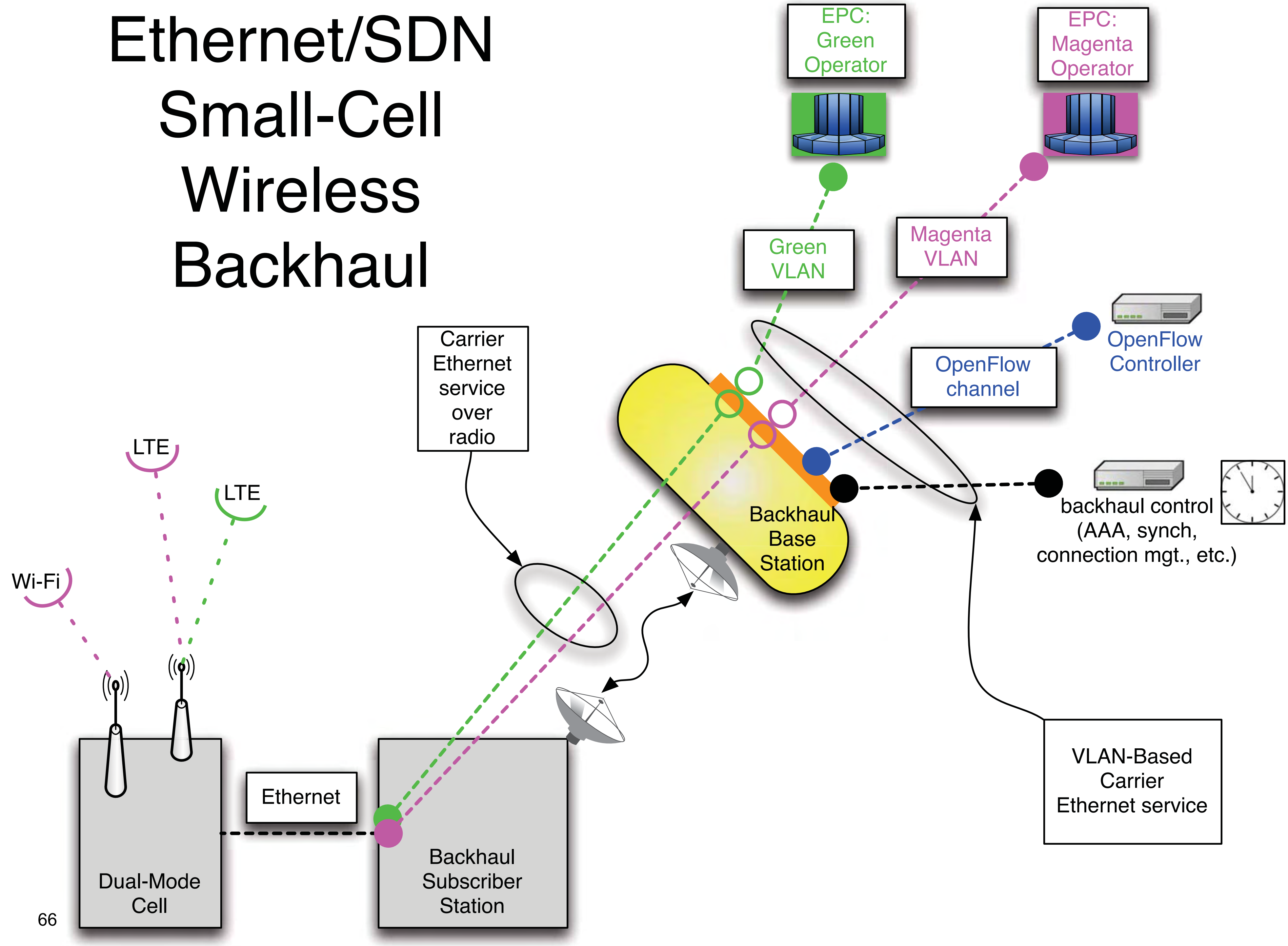
Small Cell Backhaul with P802.16r

- IEEE Project 802.16r: **Small Cell Backhaul**
  - *for effective use in wireless fixed and nomadic Ethernet transport, including small cell backhaul applications*
  - *small cell air interface could be, for example, WirelessMAN-OFDMA, IEEE 802.11, or 3GPP LTE*
  - *out-of-band wireless backhaul to the small cells, allowing those cells to be positioned for optimal performance without regard to the local availability of high-capacity wired backhaul*
  - *providing core network services to radio access networks*
  - *support of Carrier Ethernet 2.0 backhaul requirements*

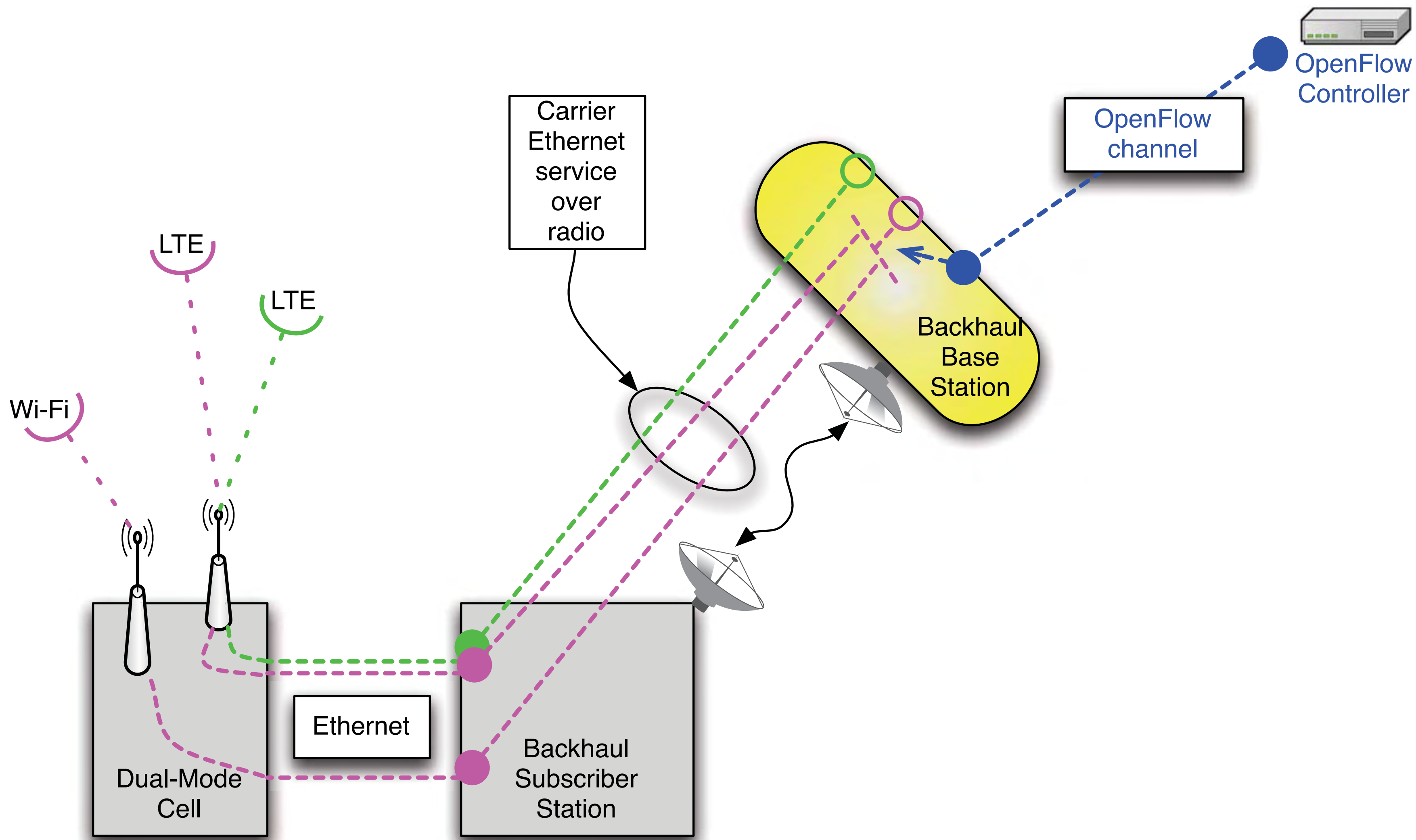
Wireless Backhaul per P802.16r



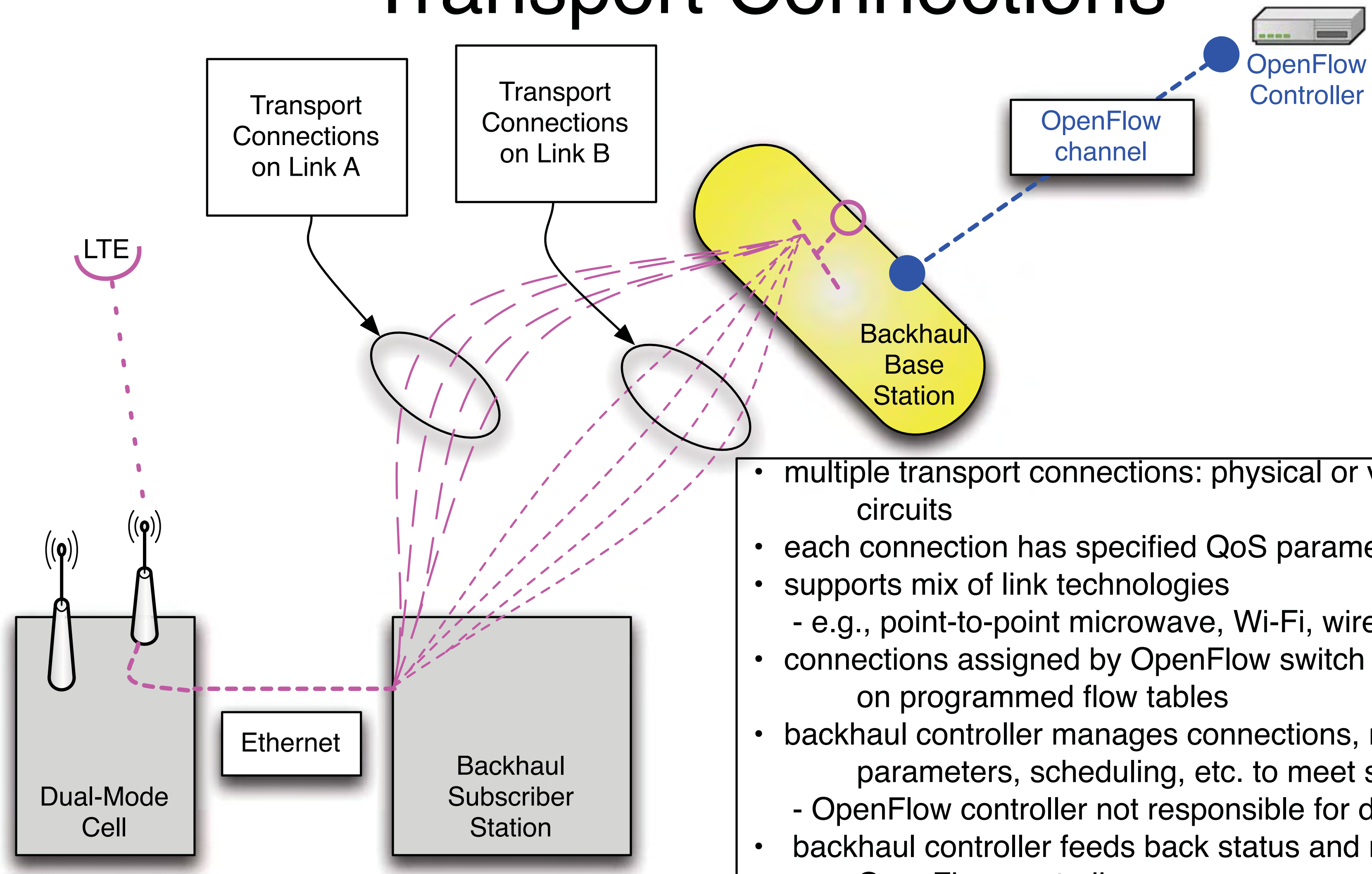
Ethernet/SDN Small-Cell Wireless Backhaul



Highlight:
OpenFlow Switch and Controller



Highlight: Transport Connections



- multiple transport connections: physical or virtual transport circuits
- each connection has specified QoS parameters
- supports mix of link technologies
  - e.g., point-to-point microwave, Wi-Fi, wired Ethernet
- connections assigned by OpenFlow switch per flow based on programmed flow tables
- backhaul controller manages connections, radio parameters, scheduling, etc. to meet specified QoS
  - OpenFlow controller not responsible for details
- backhaul controller feeds back status and reporting to OpenFlow controller
- see "Integration of IEEE 802.16 with OpenFlow Software-Defined Networking," R. B. Marks <802.16-13-0084>

802.16 Wireless Backhaul Directions

- P802.16r is in development for small-cell backhaul
- 802.16 Working Group has communicated with ONF Wireless and Mobile Discussion Group regarding P802.16r for coordinated effort
  - expects to continue discussions with ONF Wireless and Mobile Working Group (WMWG)

Looking Forward

Questions and Answers

Looking Forward

- Abstract: “This tutorial addresses applications of SDN in wireless access and backhaul networks, with a focus on issues relevant to IEEE 802 technologies.”
- We encourage 802 Working Groups to consider SDN implications on future standards developments.

Questions and Answers