Overview and Discussion of SDN Standards

Adrian Farrel : Old Dog Consulting

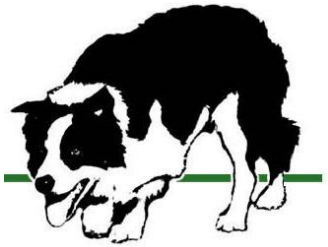
adrian@olddog.co.uk

Daniel King : Lancaster University

d.king@lancaster.ac.uk

www.isocore.com/2015

SDN MPLS



Reader Beware!

- This presentation makes no judgements
- We offer no attempt at guidance
  - Which standards are best?
  - Which standards body makes the most sense?
  - Which protocols are best for which jobs?
- This is just a “catalogue” of the work in progress
  - We deliberately blur the lines between SDN and NFV

Agenda

- Standards Development Organisations (SDO) working on SDN & NFV
  - Focus on the IRTF & IETF
- Network Controllers
  - Decomposition of the controller
  - OpenSource Initiatives
- South Bound Interfaces
  - ForCES, OpenFlow, BGP-LS, OVSDB, I2RS, PCEP, et al.
- North-Bound Interface
  - Current activity and interest
- East West Interface
  - What are the requirements?

SDN & NFV Standardisation Efforts

- Where is SDN & NFV specification and standardisation taking place?
  - What is the overlap with SDN deployment and research?
- Can we define the hot topics currently?
  - What does industry think of these efforts?
- What are the main SDOs?
  - How do they overlap and compete?
  - Which are most complete?
  - Which are easiest to participate in?
  - Which are easiest to use?
  - What does industry think of these efforts?

"The nice thing about standards is there are so many to choose from."

"You wait for years for an SDO to work on your technology and then ten come along together."

SDN & NFV Standardisation Efforts

- "SDN" Started with ONF, rapidly expanded to include:
  - ITU-T, ETSI, IRTF/IETF, ATIS, BBF, CCSA, IEEE, TMF, CEF, MEF, TTA, and 3GPP
- Note that OpenSource efforts are in some sense *de facto* standardisation
  - We will cover them later
- The volume and breadth of standards activity can be a barrier to participation
  - Need to decide which organizations are important and where the opportunities are

Open Networking Foundation

- The originators of all the current brouhaha
  - Brought SDN to the attention of the world
  - Developed and marketed OpenFlow as a South Bound interface
  - Also produced a comprehensive SDN architecture
- What are the hot areas within the ONF?
  - North Bound Interface
  - East West Interface
  - Security
  - Relationship of SDN & NFV Architecture
  - SDN Distribution: ATRIUM

ONF Atrium

- ONF provide an SDN distribution (Anchor Stack)
  - “Atrium” is the first release to help simplify controller and application integration
  - 30 minutes from download to packet passing tutorials and process
  - Build testbeds that operate with Anchor Stack
- First release of Anchor Stack (“Atrium”)
  - BGP-Routing under Quagga as the Networking App
  - ONOS as a controller (qv.)
  - Prototype Pipeline Adaptation Layer
    - OF-DPA 1.0 on Quanta LY4 Switch
    - Corsa WAN Switches
    - Dell ToR switches using non OF-DPA pipeline
  - Supports 4 Different Vendor switches
    - Already popular suppliers for OpenFlow hardware at different size and scale

SDN & NFV Standardisation Efforts

- ITU-T Joint Coordination Activity on SDN (JCA)
  - Will coordinate the work carried out by various ITU-T Study Groups and Expert Groups
  - Also acting as the first point of contact for organizations interested in contributing to ITU's SDN standardization program
- ITU-T Study Group 11 (Signalling requirements, protocols and test specifications – telephone calls and data calls)
  - Q.4 is developing a supplement (non-normative document) that describes the framework of SDN signalling
  - Q.6 is studying how to apply SDN technologies for IPv6

SDN & NFV Standardisation Efforts

- ITU-T Study Group 13 (Future networks including cloud computing, mobile and NGN)
  - Q.2 studies SDN and virtualization aspects for next generation networks (NGN)
  - Q.14 is responsible of such framework and also discusses network virtualization, and has developed Recommendation ITU-T Y.3300 (*Framework of software-defined networking*)
- ITU-T Study Group 15 (Transport, Access and Home)
  - SG15 is studying “Transport aspects of SDN”
    - It has commenced a new draft Recommendation “Architecture for SDN control of Transport Networks”, aligned with the ONF’s “SDN architecture”, Issue 1,
    - It has commenced a new draft Recommendation “Common Control Aspects” on common aspects of the interaction between the ASON control plane, SDN controller plane, management plane and transport plane

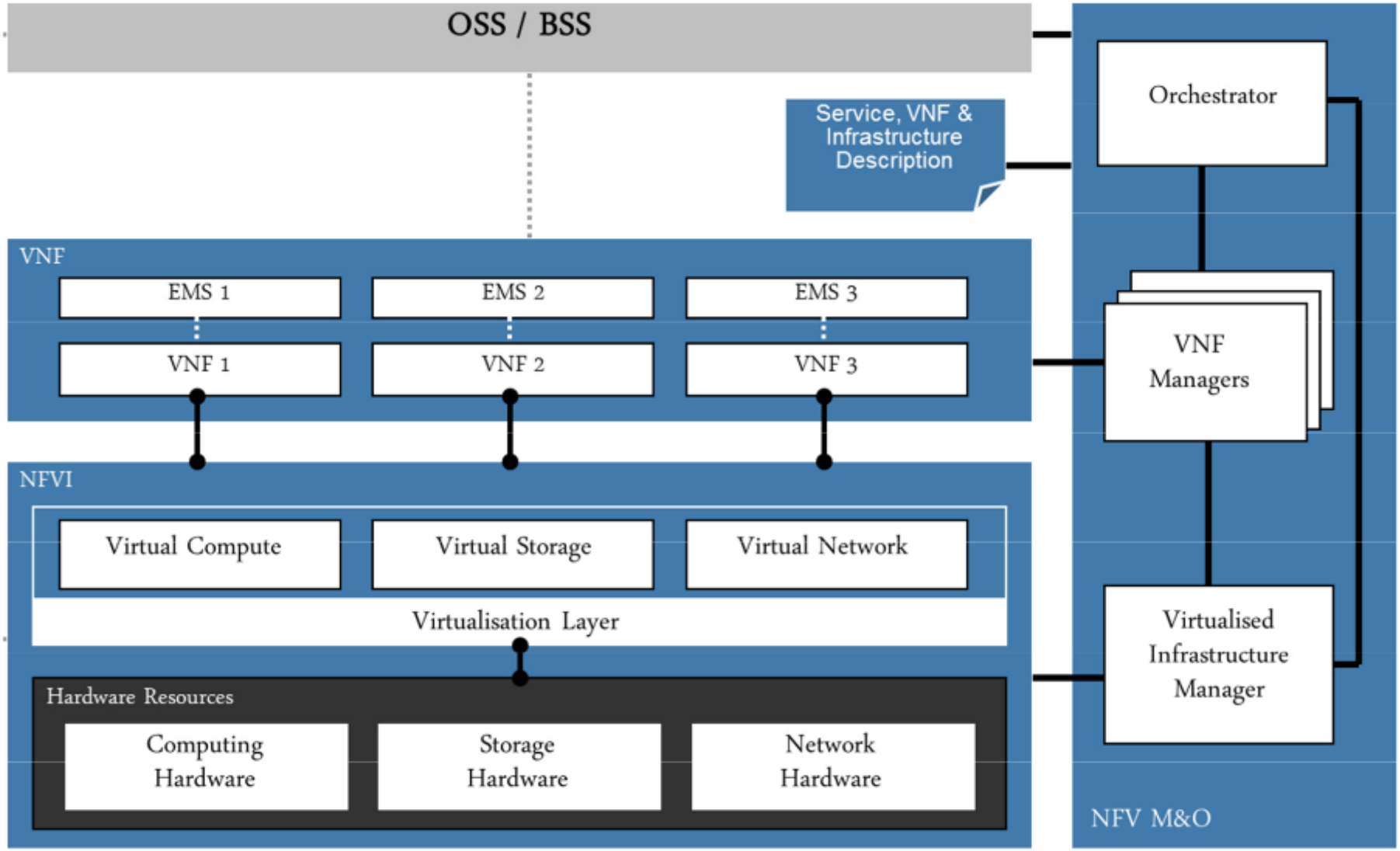
SDN & NFV Standardisation Efforts

- ITU-T Study Group 16 (Multimedia)
  - Q.3 is studying OpenFlow versus H.248 as a protocol to control packet flows
  - Q.21 is studying virtual content delivery networks
- ITU-T Study Group 17 (Security)
  - Q.6 is studying security by SDN, which covers the security services using SDN:
    - Draft Recommendation ITU-T X.sdnsec-1 is developing requirements for security services based on SDN
  - Q.2 is studying security of SDN, which covers the security architectural aspects of SDN and how to secure the SDN environment

SDN & NFV Standardisation Efforts

- ETSI Industry Specification Groups (ISGs) for Network Functions Virtualization (NFV ISG)
  - ETSI is worldwide, not just European
  - Dominated by operators not vendors
  - NFV ISG set up to achieve a consistent approach and common architecture for the hardware and software infrastructure needed to support virtualized network functions
  - NFV ISG achieved first five deliverables :
    - Four deliverables designed to align understanding about NFV across the industry, cover:
      - NFV use cases, requirements, architectural framework and terminology
    - Fifth deliverable defines a framework for coordinating and promoting public demonstrations of Proof of Concept (PoC) platforms illustrating key aspects of NFV, with the objective to encourage the development of an open ecosystem by integrating components from different players
  - NFV ISG created three more deliverables
    - Methodology to describe interfaces and abstractions for NFV infrastructure
    - Problem statement for NFV security and NFV performance
    - Portability best practises

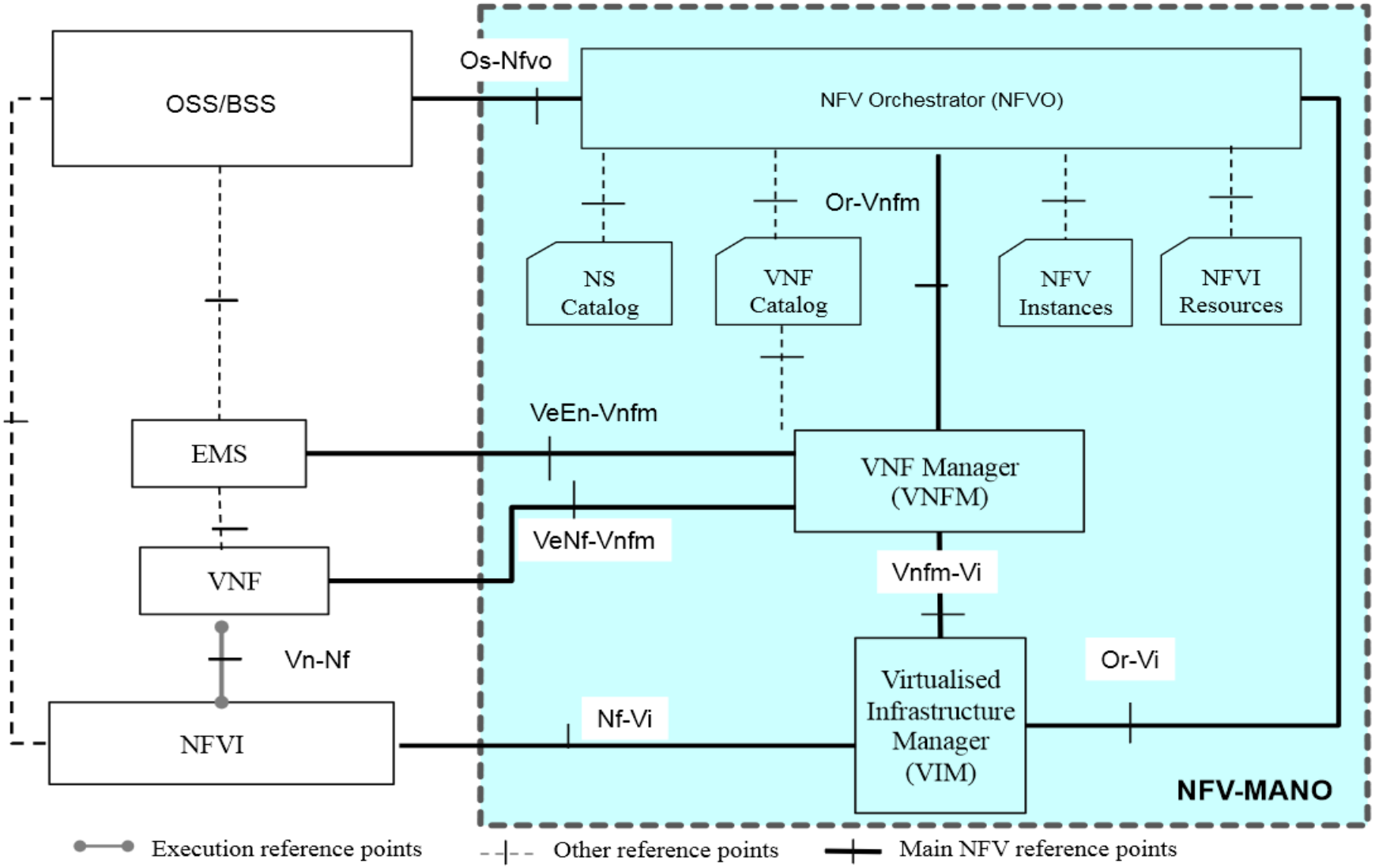
ETSI NFV ISG Architectural Framework



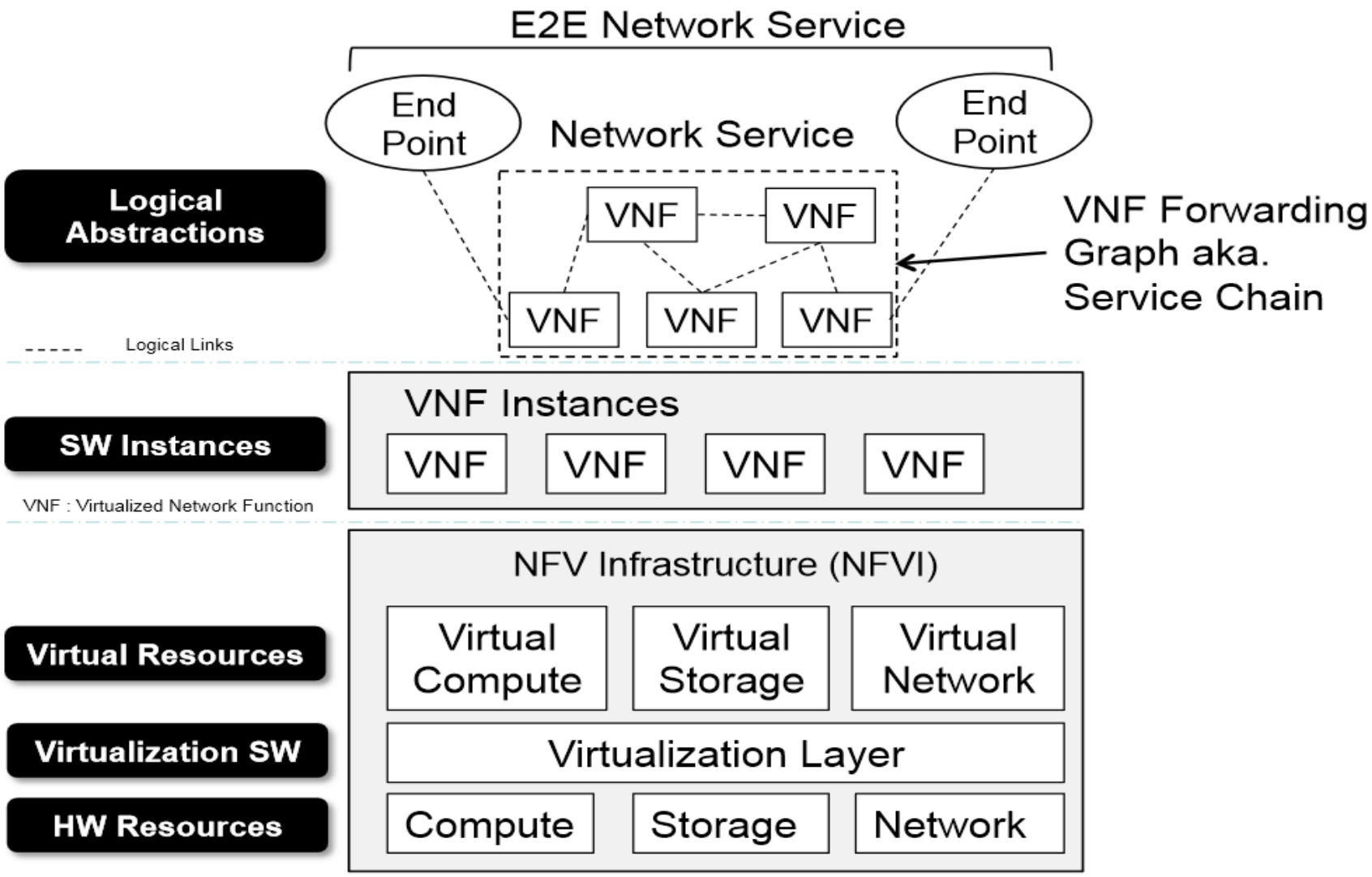
ETSI Management and Orchestration (MANO) Functional Components

- Management and Orchestration (MANO) Functional Blocks
  - NFV Orchestrator:
    - Network Service (NS) lifecycle management (including instantiation, scale-out/in, performance measurements, event correlation, termination)
    - Global resource management, validation and authorization of NFVI resource requests
    - Policy management for NS instances
  - VNF Manager:
    - Lifecycle management of VNF instances
    - Overall coordination and adaptation role for configuration and event reporting between NFVI and the E/NMS
  - Virtualized Infrastructure Manager (VIM):
    - Controlling and managing the NFVI compute, storage and network resources, within one operator's infrastructure sub-domain
    - Collection and forwarding of performance measurements and events

ETSI Management and Orchestration (MANO) Architecture



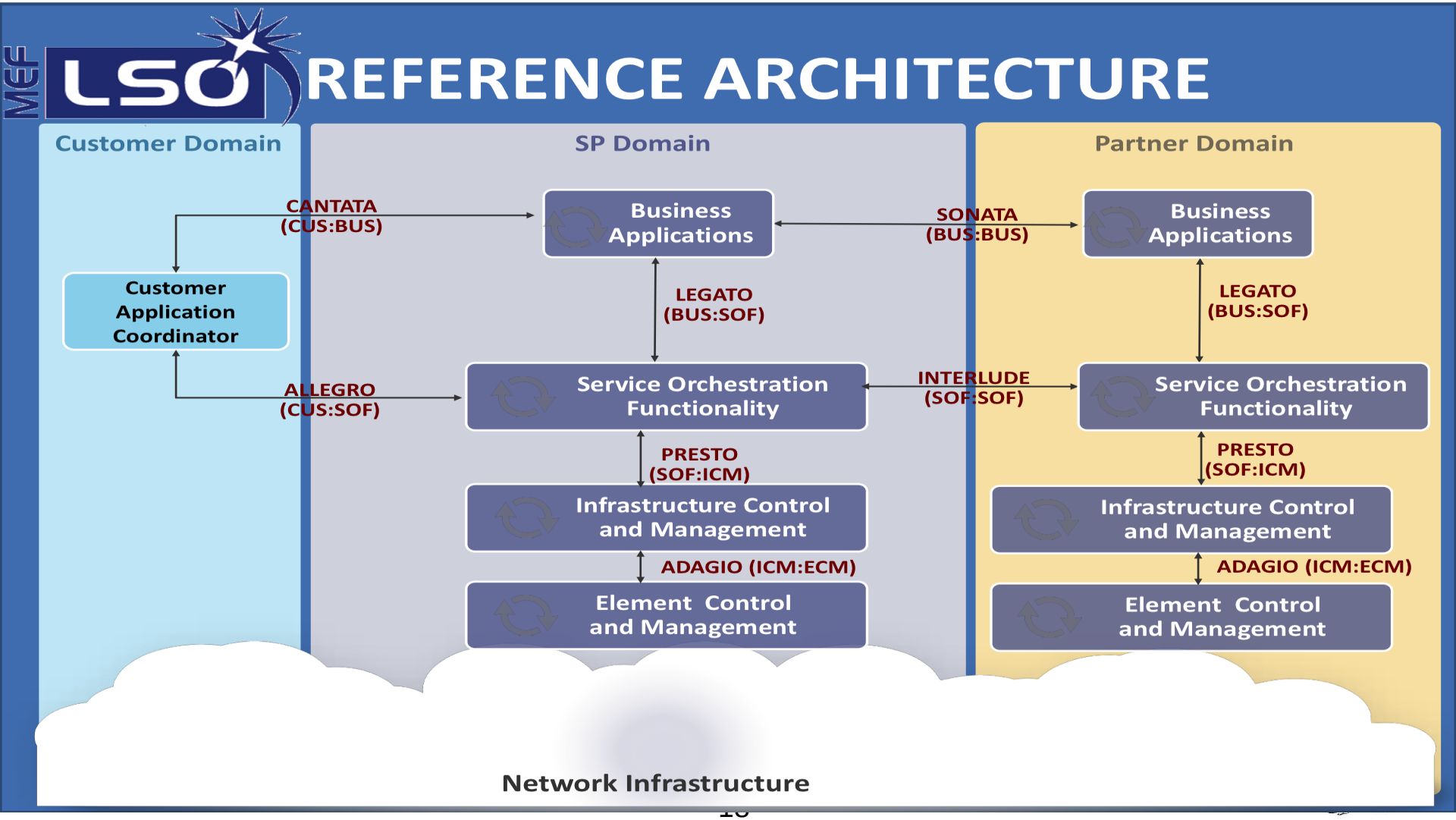
ETSI NFV E2E Service



SDN & NFV Standardisation Efforts

- MEF (No longer known as Metro Ethernet Forum)
  - Is trying to address SDN/NFV and Cloud aspects that affect Carrier Ethernet services
  - 3rd Network (previously Network as a Service (NaaS)) vision:
    - Top down solution starting with Service Order at the OSS/BSS level, addressing service fulfilment and assurance for Carrier Ethernet which includes SDN orchestration solutions
  - Started activity on service models for carrier Ethernet services

MEF's Reference Architecture



SDN & NFV Standardisation Efforts

- CEF (Cloud Ethernet Forum)
  - Mainly involved in Cloud related specifications
- Telecommunications Technology Association (TTA)
  - Future Internet project group (PG220) is a lead project group of SDN issues related to ITU-T SG13
  - PG220 is now developing TTA standards which describe common hardware and software platforms to support open programmable networking and SDN/NFV enabled services
- 3GPP
  - TSG SA WG5 (SA5) is a group for telecom management
  - Started a new study item on Network Management of Virtualized Networks
  - Their objective is to study the use cases concepts for the network management of Virtualized Networks, to identify the requirements for potential solutions, and to do gap analysis between the identified requirements and current 3GPP Management reference model

SDN & NFV Standardisation Efforts

- IEEE P1903 WG (NGSON)
  - The IEEE P1903 WG was setup to develop specifications for Next Generation Service Overlay Networks (NGSON)
  - The P1903 WG is currently working on the specification of service enabling functions which can be provided as Virtualized Network Functions (VNFs) to support NFV applications.
  - Technical aspects which may be related to SDN are mechanisms for Service Composition, Service Routing and Self Organization aiming to provide service level virtualization
- Broadband Forum (BBF)
  - End to End Architecture (E2E) WG
  - Discusses end-to-end architecture issues
  - Business Requirements and Framework for SDN in Telecommunication Broadband Networks
  - Operation of Residential Gateways and Business Gateways in association with SDN model
- TMF (No longer known as “TeleManagementForum”)
  - Focused on adapting their management concepts interfaces and tools to support SDN and NFV

SDN & NFV Standardisation Efforts

- China Communications Standards Association (CCSA)
  - TC1 SWG3
    - This Special Working Group for FDN (future data network) has developed two standards:
      - “Scenarios and requirements of FDN”
      - “Function Architecture of FDN”
  - TC3 WG1
    - This Working Group for SVN (Software Smart and virtualized Networking), mainly focuses on the standard activities on SDN-based smart network, network virtualization, and future network from carriers’ perspective. They developed two standards:
      - “General Requirements for Network Intelligent Capability Enhancement making usage of software defined networking technologies (S-NICE)”
      - “Research on VCN (Virtualization of Control Network-entity)”

SDN & NFV Standardisation Efforts

- CCA (Cloud Clean Alliance)
  - Established by Huawei to identify and coordinate against large DDoS attacks
    - Facilitate near-source cloud mitigation to ensure link availability guarantees
    - Obtain valuable data and trend analysis of the global DDoS attack status
  - Provide “alliance” security service capabilities to customers
    - Combination of specifications and hardware located in “scrubbing centers” across four continents

SDN & NFV Standardisation Efforts

- Internet Research Task Force (IRTF)
  The research arm of the IETF
  - SDN RG topics of interest
    - More info on next slides
  - NFV RG
    - VNF Elasticity
    - Policy
  - Networks Machine Learning (NML)
    - Network attack mitigation/DDOS
    - Applicability of ML to complex path computation problems

SDN Research Group : History

- Several attempts to bring SDN work into the IETF
- SDN Bar BoF at IETF 81 (July 2011)
  - SDNP mailing list created
- Software Driven Networks BoF at IETF 82 (Nov, 2011)
  - 400 participants
  - *“Enables network applications to request and manipulate services provided by the network, and allow the network to provide feedback to the network applications.“*
  - Not enough focus to form a working group
- IETF 84 (Jul, 2012) first meeting for a proposed SDN RG

IRTF: SDN Research Group Chartered

- Areas of interest include: architecture, solution scalability, abstractions, and programming languages and paradigms
- The explicit goal of the SDN RG is to provide a forum for researchers to investigate key and interesting problems in the SDN field
- Classification of SDN models, including
  - Definitions
  - Taxonomies
  - Relationship to work ongoing in the IETF and other SDOs
- SDN model scalability and applicability
- Multi-layer programmability and feedback control systems
- System Complexity
- Network description languages, abstractions, interfaces and compilers
  - Including methods and mechanisms for (on-line) verification of correct operation of network/node function.
- Security

SDN RG Holds Focussed Meetings

- IETF 85 – Seeded some “SDN-hard” Problems
- IETF 86 – SDN Trends, Experiments and NFV first mentioned
- IETF 87 – Novel SDN Applications, Operator Perspectives, State Reduction and Dependable SDNs
- IETF 88 – Defining SDN in the context of the IETF, Architectures and Terminology
- IETF 89 – SDN Hybrid Architectures and SDN Research in Action
- IETF 90 – Hardware Abstraction, Modeling Languages and Traffic Engineering
- IETF 91 – Scaling SDN, Blending SDN &NFV, NFV H-W Acceleration, and IoT
- IETF 92 – (Focused CFP) Inter-domain SDN, SDN in Mobile Networks
- IETF 93 – (Focused CFP) SDN Security
- IETF 94 – (Focused CFP) Operator Research (Challenges, Findings and Opportunities), invited Japanese & Chinese researchers

History of SDN in the IETF

- Working on SDN concepts for over 10 years
  - GSMP
  - ForCES
  - PCE (for Packet and Optical)
- “SDN” Discussion started late 2011
  - IETF BoF >400 participants (operators, vendors, academics)
  - Ended up boiling the SDN ocean
  - Led to formation of SDN RG
    - Seed ideas for new working groups in the IETF
    - Establish requirements/use cases for extending existing IETF technologies
- Other pieces of the SDN puzzle continue to be worked on in IETF

IETF SDN Efforts

- Current Working Groups
  - Path Computation Element - PCE
  - Interface to the Routing System - I2RS
  - Layer 3 Service Modeling– L3SM
  - Layer Independent OAM Management in Multi-Layer – LIME
  - Simplified Use of Policy Abstractions – SUPA
  - Interface to network Security Functions – I2NSF
- Recent Working Group Initiatives
  - Virtual Network Function Pools – VNF Pool
    - Two BoF meetings, but was premature
  - Abstraction and Control of Transport Networks – ACTN
    - Moved into the TEAS working group

Standardisation Summary

- You would be hard pressed to follow all of the SDN standards activity in detail
- “Every organisation is potentially (and dangerously) self perpetuating”
  - Few SDOs have a life-cycle plan that bound their authority and scope
- New technology study groups are exploding across multiple SDOs.
- Choose any new area of technical endeavour: Cloud, SDN, NFV, IoT, “APIs” and you will find ready examples
- This dilutes and confuses the effort for contributors and consumers
- Coordination across SDOs doesn’t really appear to be working
- Recall the purpose of standardisation
  - To enable people to make interoperable implementations
  - Diversity of standards does not aid interoperability
- It is no wonder that some people give up on standards and look to OpenSource

"If open APIs become the de-facto definition of interoperability requirements, the role of the standardization bodies, and the opportunity for operators to influence specifications, diminishes. As a result the functional interoperability (and interchangeability) of vendors and devices will decrease, potentially leading to a more proprietary and less open and global nature of the Internet."

https://tools.ietf.org/html/draft-opsawg-operators-ietf-00

Network Controllers

- Controller for Network Operations
- Decomposition of a Network Controller
- Application-Based Network Operations
  - Optical Network Use Case & Requirements
- Open Daylight
- OPNFV
- ONOS
- Open Contrail

Controller for Network Operations

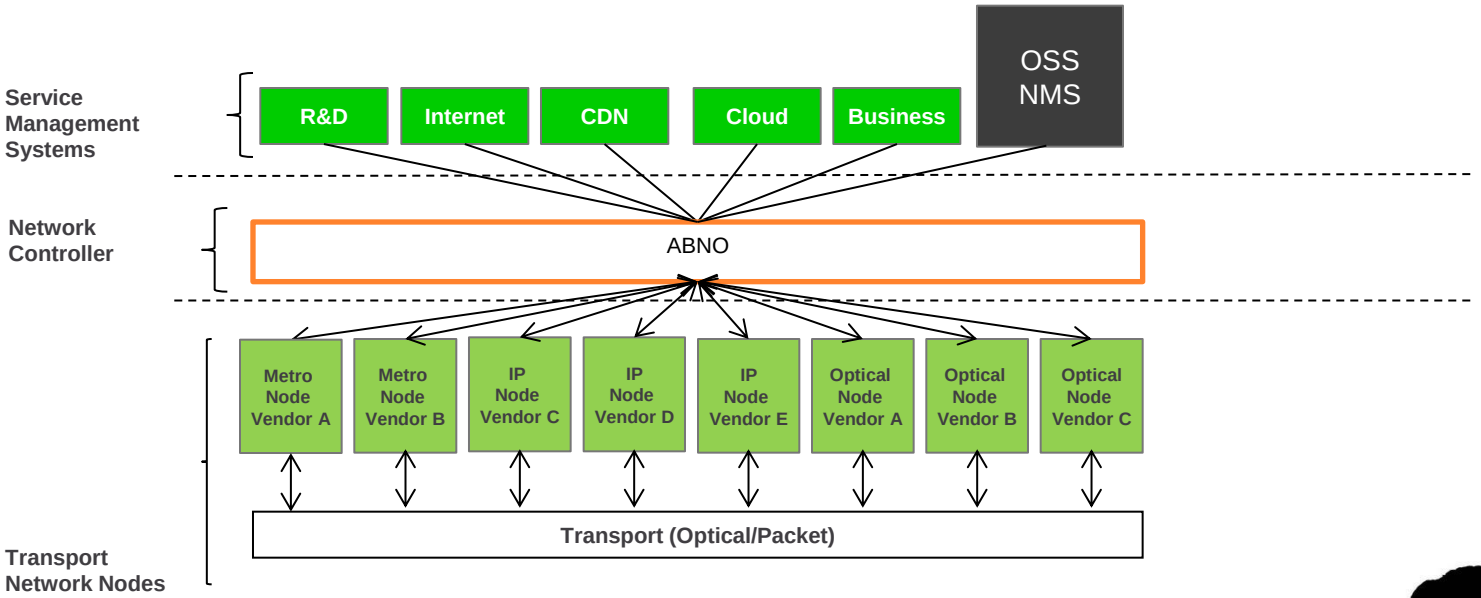
- "SDN Controller" is a contentious term: it can have many different meanings:
  - Historically the term was derived from the network domain, technology and protocol mechanism
- SDN Controller wars are ongoing:
  - Operators have an expectation of standards-based technologies for deploying and operating networks
  - SDN controller vendors rarely provide multivendor interoperability using open standards
  - Provisioning should be a compelling feature of SDN, however many SDN controllers use non-standardised APIs
  - Recent Open Source initiatives are vendor led
- Typically, SDN controllers have a very limited view of topology
  - Multi-layer and multi-domain scenarios are slowly being added
- Flexibility has been notably absent from most controller architectures both in terms of southbound protocol support and northbound application requests

Decomposition of Network Controller

- Avoiding the mistake of a single “controller” architecture
- Discovery of network resources and topology management
- Network resource abstraction, and presentation
- Routing and path computation
- Multi-layer coordination and interworking
  - Multi-domain & multi-vendor network resources provisioning through different control mechanisms (e.g., OpenFlow, ForCES)
- Policy Control
- OAM and Performance Monitoring
- Security & Resiliency
- A wide variety of southbound northbound protocol support
- Leveraging existing technologies
  - What is currently available?
  - Must integrate with existing and developing standards. Decomposition

Application-Based Network Operations (ABNO)

- Application-Based Network Operation (ABNO) framework.
- “A PCE-based Architecture for Application-based Network Operations”
  - https://tools.ietf.org/html/rfc7491
  - Old Dog Consulting, Juniper, Huawei, Telefonica, CTTC, NTT, Yale



ABNO
A PCE-enabled Network Controller

- PCE provides a set of tools for deterministic path computation
  - Prior to PCE network operators might use complex planning tools to compute paths and predict network behavior
  - PCE reduces the onerous network operation process of coordinating planning, computation, signaling and placement of path-based services
- PCE has evolved:
  - Computes single and dependant LSPs in a stateless manner
  - Concurrent optimization of sets of LSPs
  - Performing P2P and P2MP path computation
  - Hierarchical PCE Architecture
  - Stateful computation and monitoring of LSPs
    - The ***state*** in “stateful” is an LSP-DB
    - Stored information about some or all LSPs in the network
  - Active PCE, resize or recomputed based on BW or network triggers
  - PCE-initiated LSP setup
    - Delegate LSP control to the PCE
    - Recommend rerouting of LSPs

ABNO Function Components

- "Standardized" components and co-operation.
- Policy Management
- Network Topology
  - LSP-DB
  - TED
  - Inventory Management
- Path Computation and Traffic Engineering
  - PCE, PCC
  - Stateful & Stateless
  - Online & Offline
  - P2P, P2MP, MP2MP
- Multi-layer Coordination
  - Virtual Network Topology Manager
- Network Signaling & Programming
  - RSVP-TE
  - PCEP, NETCONF, ForCES and OpenFlow
  - Interface to the Routing System (I2RS)

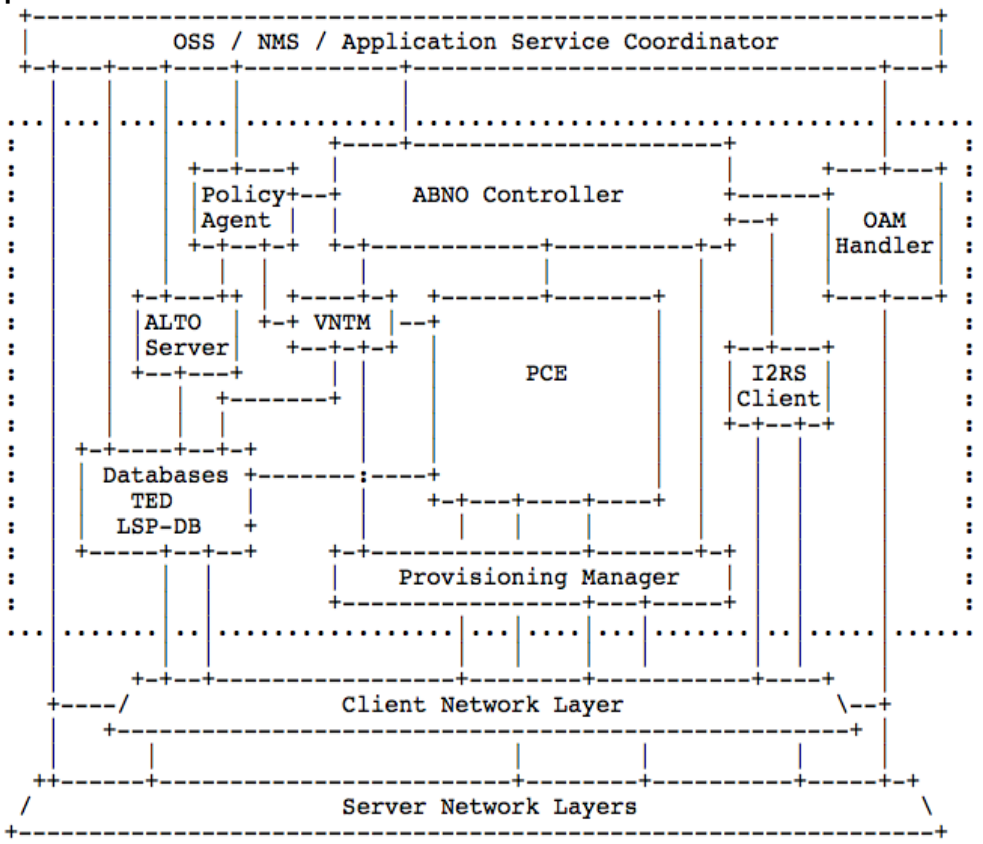


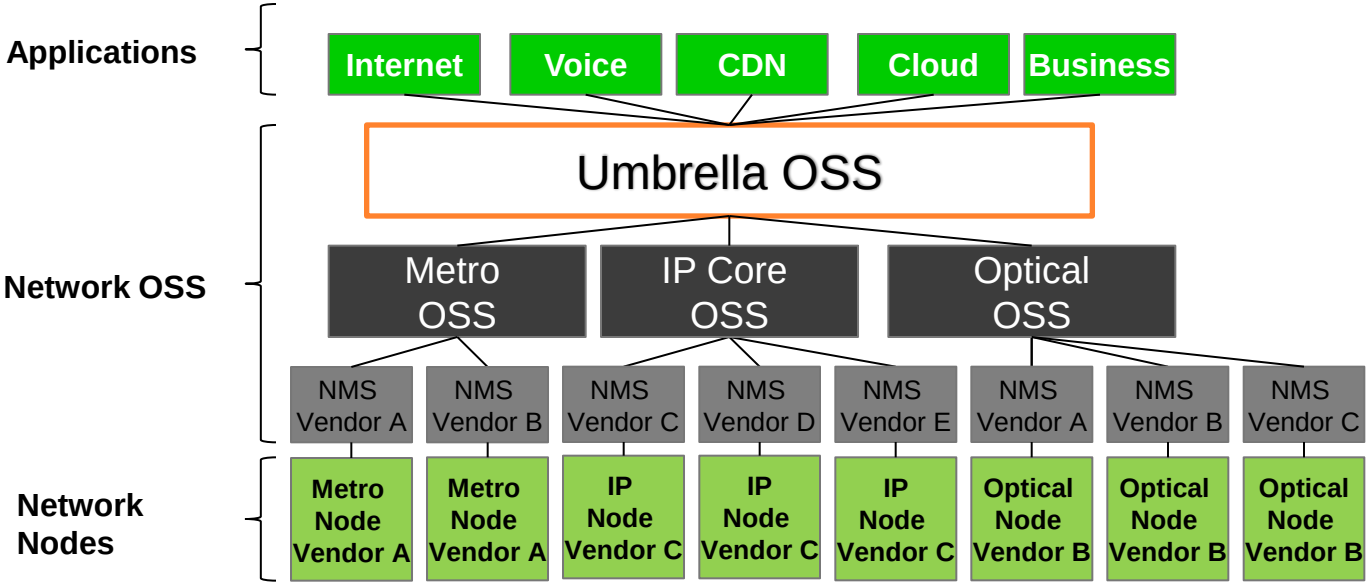
Figure 1: Generic ABNO Architecture

Optical Network Complexity

- Flexible and Elastic Optical Networks
  - Photonic Integrated Circuit (PIC) technology
    - Paving the path towards cost effective transmission schemes beyond 100Gbps.
  - Digital Coherent and SuperChannel technology solutions
    - Variable >100Gbps client signals and cost effective >100Gbps transponders
    - Capable of long reach up to 400Gbps without regeneration
  - Cost effective and flexible transponders
    - The Sliceable-Bandwidth Variable Transponder (SBVT).
      - Reduce bandwidth to extend reach
      - More spectrum to extend reach
      - More bandwidth over short reach
- Flexi-grid
  - A variable-sized optical frequency range.
  - ITU-T Study Group 15 (www.itu.int/rec/T-REC-G.694.1)

Control of Today's Optical Networks

- Current network operation is not adapted to flexible optical networking
  - Multiple manual configuration actions are needed for network nodes
  - Network solutions from different vendors typically use specific OSS/NMS implementations
  - Very long provisioning times
  - Lack of network bandwidth flexibility and inefficient use of inherent function



Optical Controller Requirements & Applications

- The network does not need to be seen any longer as a composition of individual elements
  - Applications need to be capable of interaction with the network.
- Support of the next generation of variable and dynamic optical transport characteristics
  - Multi-layer Path Provisioning
  - Network Optimization after Restoration
- Automated deployment and operation of services.
  - “Create a new transport connection for me”
  - “Reoptimize my network after restoration switching”
  - “Respond to how my network is being used”
  - “Schedule these services”
- Leveraging existing technologies
  - What is currently available
  - Must integrate with existing and developing standards
- Support for flexi-grid

Yes, but is it actually being used/developed?

- Publications & Standards
  - A PCE-Based Architecture for Application-Based Network Operations, IETF RFC7491
  - Unanswered Questions in the Path Computation Element Architecture, IETF RFC7399
  - “In-Operation Network Planning”, IEEE Communications Magazine
  - “Adaptive Network Manager: Coordinating Operations in Flex-grid Networks”, ICTON (IEEE)
  - “Experimental Demonstration of an Active Stateful PCE performing Elastic Operations and Hitless Defragmentation”, ECOC European Conference on Optical Communications
  - “Planning Fixed to Flexgrid Gradual Migration: Drivers and Open Issues”, IEEE Communications Magazine
  - “Dynamic Restoration in Multi-layer IP/MPLS-over-Flexgrid Networks”, IEEE Design of Reliable Communication Networks (DRCN)
  - “A Traffic Intensity Model for Flexgrid Optical Network Planning under Dynamic Traffic Operation”, OSA Optical Fiber Communication (OFC)
- EC-Funded Projects investigating, using and/or developing ABNO
  - XIFI, OFERTIE, DISCUS, CONTENT, PACE
- Deployments and Code Availability
  - iONE, Universitat Politècnica de Catalunya (UPC) (OpenSource)
  - ANM, Telefonica (OpenSource)
  - Infinera (Closed Source)
  - Ericsson (Closed Source)

OpenDaylight

- OpenDaylight is a Linux Foundation Collaborative Project which is structured with an open source project with a modular, pluggable, and flexible controller platform at its core
- ODL Basic Features
  - Network Apps & Orchestration
    - The top layer consists of business and network logic applications that control and monitor network behavior
    - In addition, more complex solution orchestration applications needed for cloud and NFV thread services together and engineer network traffic in accordance with the needs of those environments
  - Controller Platform
    - The middle layer is the framework in which the SDN abstractions can manifest, providing a set of common APIs to the application layer (commonly referred to as the northbound interface) while implementing one or more protocols for command and control of the physical hardware within the network (typically referred to as the southbound interface)
  - Physical & Virtual Network Devices
    - The bottom layer consists of the physical & virtual devices, switches, routers, etc., that make up the connective fabric between all endpoints within the network

OpenDaylight – Framework (1/3)

- ODL Framework Overview
  - The OpenDaylight Controller is a pure software and as a JVM it can be run on any OS and Metal as long as it supports Java
  - On the Southbound, ODL can support multiple protocols (as plugins), e.g., OpenFlow 1.0, OpenFlow 1.3, BGP-LS, etc.
    - These modules are linked dynamically into a Service Abstraction Layer (SAL)
      - The SAL exposes services to which the modules north of it are written
      - The SAL figures out how to fulfill the requested service irrespective of the underlying protocol used between the Controller and the network devices
  - For the Controller to control devices in its domain it needs to know about the devices, their capabilities, reachability, etc.
    - This information is stored and managed by the Topology Manager
    - The other components like ARP handler, Host Tracker, Device Manager and Switch Manager help in generating the topology database for the Topology Manager

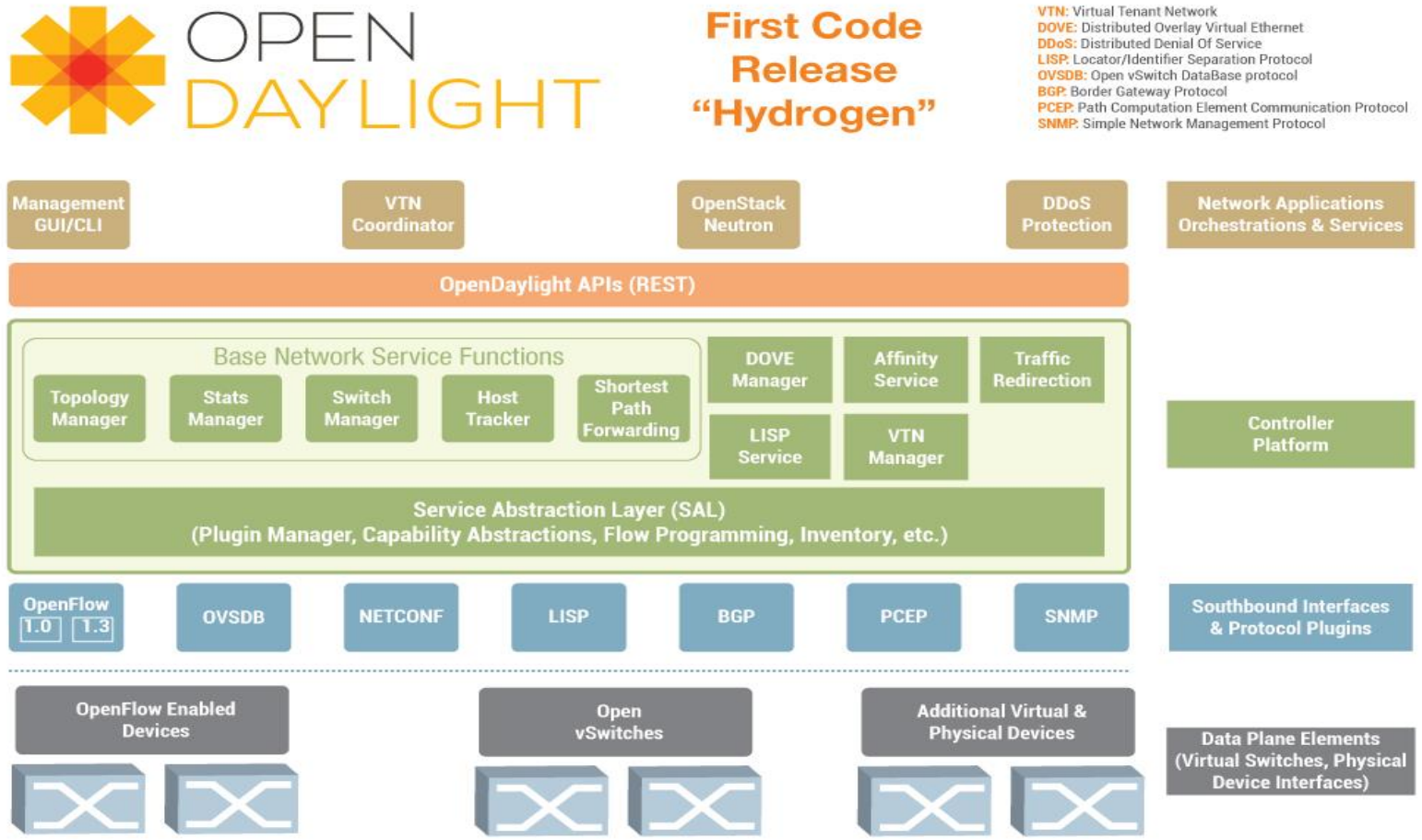
OpenDaylight – Framework (2/3)

- The Controller exposes open Northbound APIs which are used by Applications
- ODL supports the OSGi framework and bidirectional REST for the Northbound API
  - OSGi framework is used for applications that will run in the same address space as the Controller while the REST (web based) API is used for Apps that do not run in the same address space (or even the same server) as the Controller
- The business logic and algorithms reside in the Apps
  - These Apps use the Controller to gather network intelligence, runs its algorithm to do analytics and then use the Controller to orchestrate the new rules throughout the network
- The Controller has a built in GUI
  - The GUI is implemented as an application using the same Northbound API as would be available for any other user application

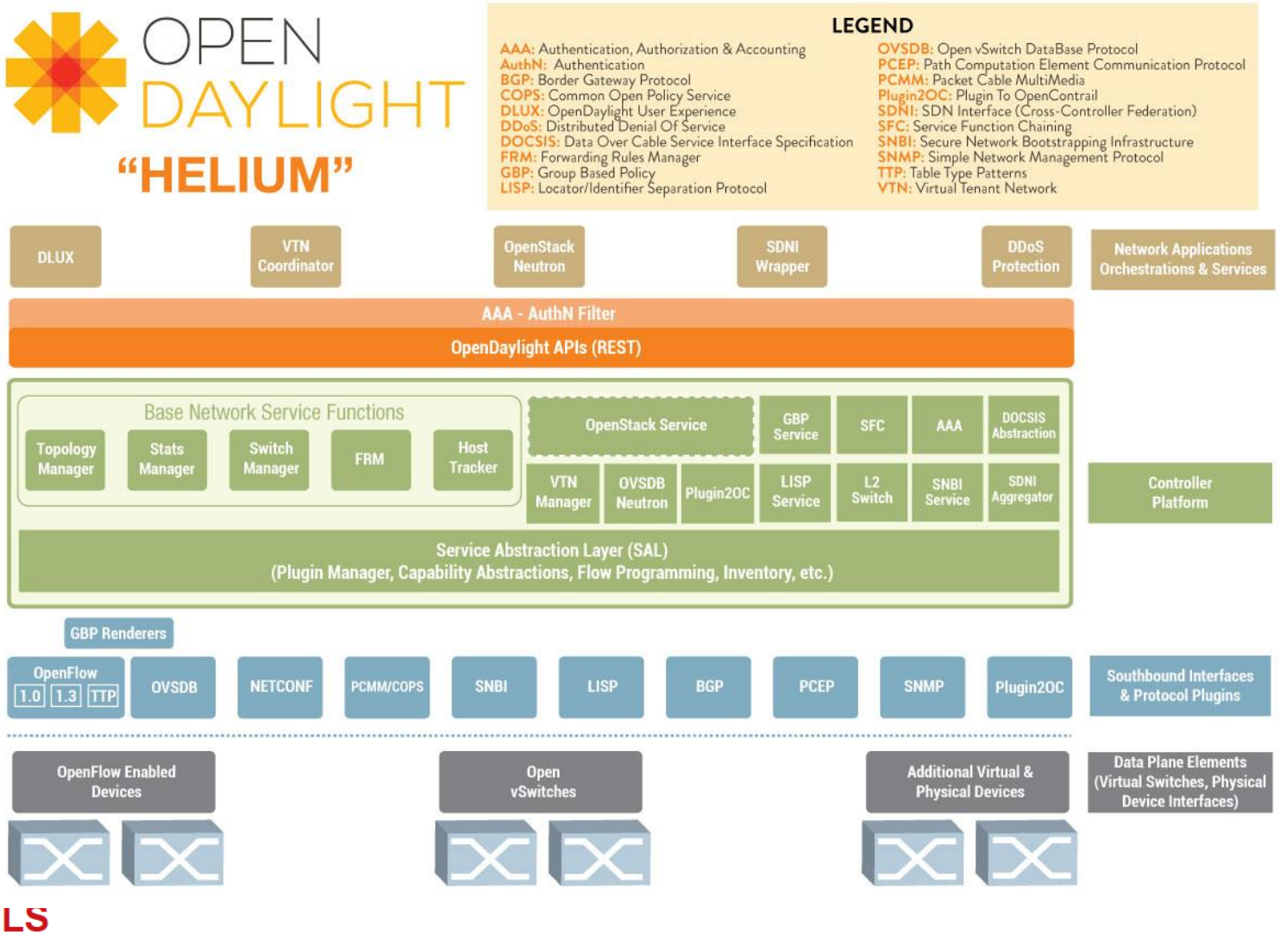
OpenDaylight – Framework (3/3)

- Switch Manager
  - The Switch Manager API holds the details of the network element
    - As a network element is discovered, its attributes (e.g. what switch/router it is, SW version, capabilities, etc.) are stored in the data base by the Switch Manager
- GUI
  - The GUI is implemented as an APP and uses the NB REST API to interact with the other modules of the Controller
    - This architecture thus ensures that whatever is possible with the GUI is also available via REST API and thus the Controller can be integrated easily into other management or orchestration systems
- High Availability
  - The OpenDaylight Controller supports a Cluster based High Availability model
    - There are several instances of the OpenDaylight Controller which logically act as one logical controller
      - This not only gives a fine grain redundancy but also allows a scale-out model for linear scalability
    - To make the Controller highly available, resilience is added at:
      - Controller level, by adding 1 or more controller instances in clustered fashion
      - Make sure the Open Flow enabled switches (OF-S elements) are multi-homed to multiple instances of the controller
      - Make sure the applications are multi-homed to the controller instances

OpenDaylight – Hydrogen (1st Release)

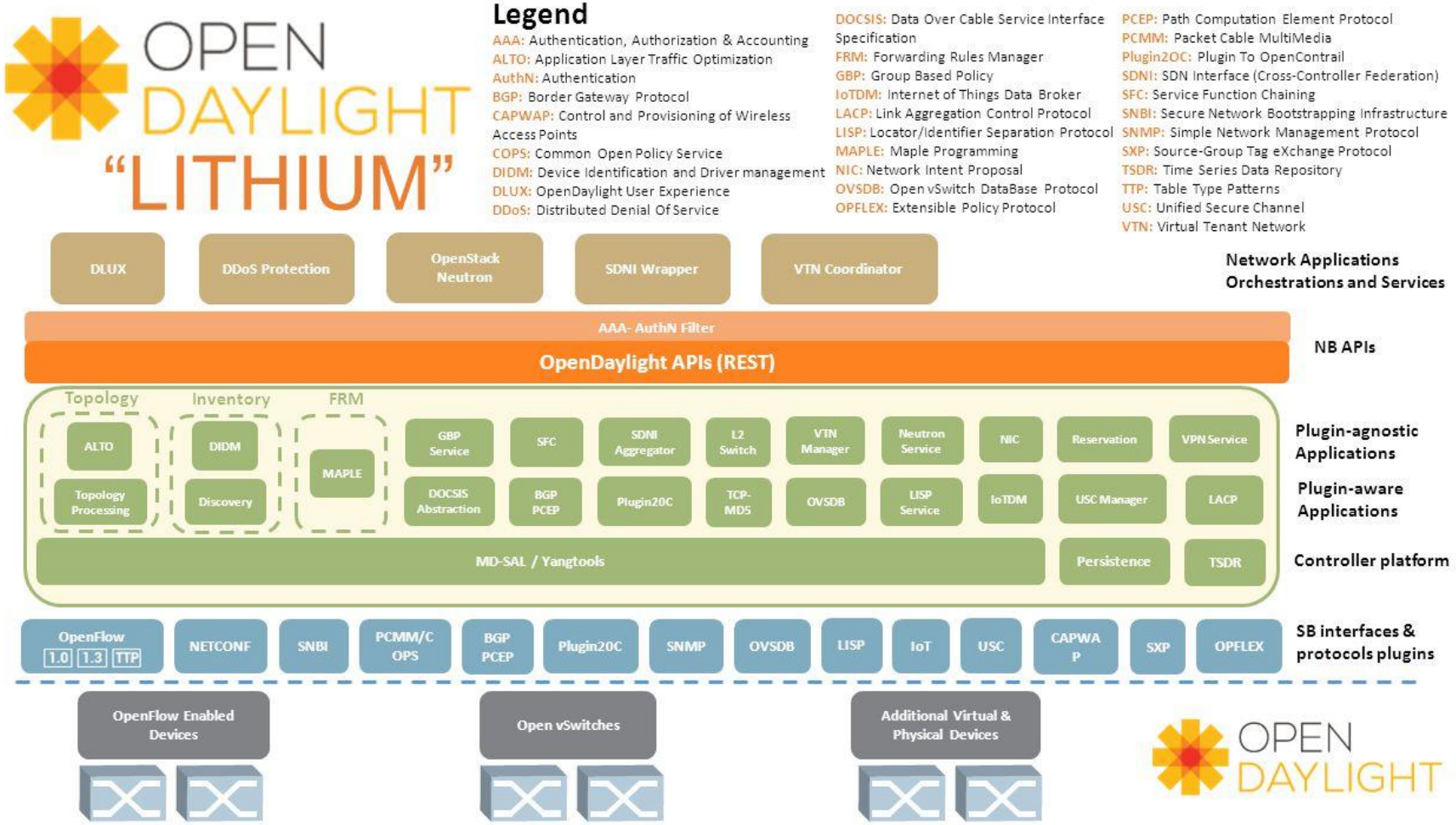


OpenDaylight – Helium (2nd Release)



SDN MPLS

OpenDaylight – Lithium (3rd Release)

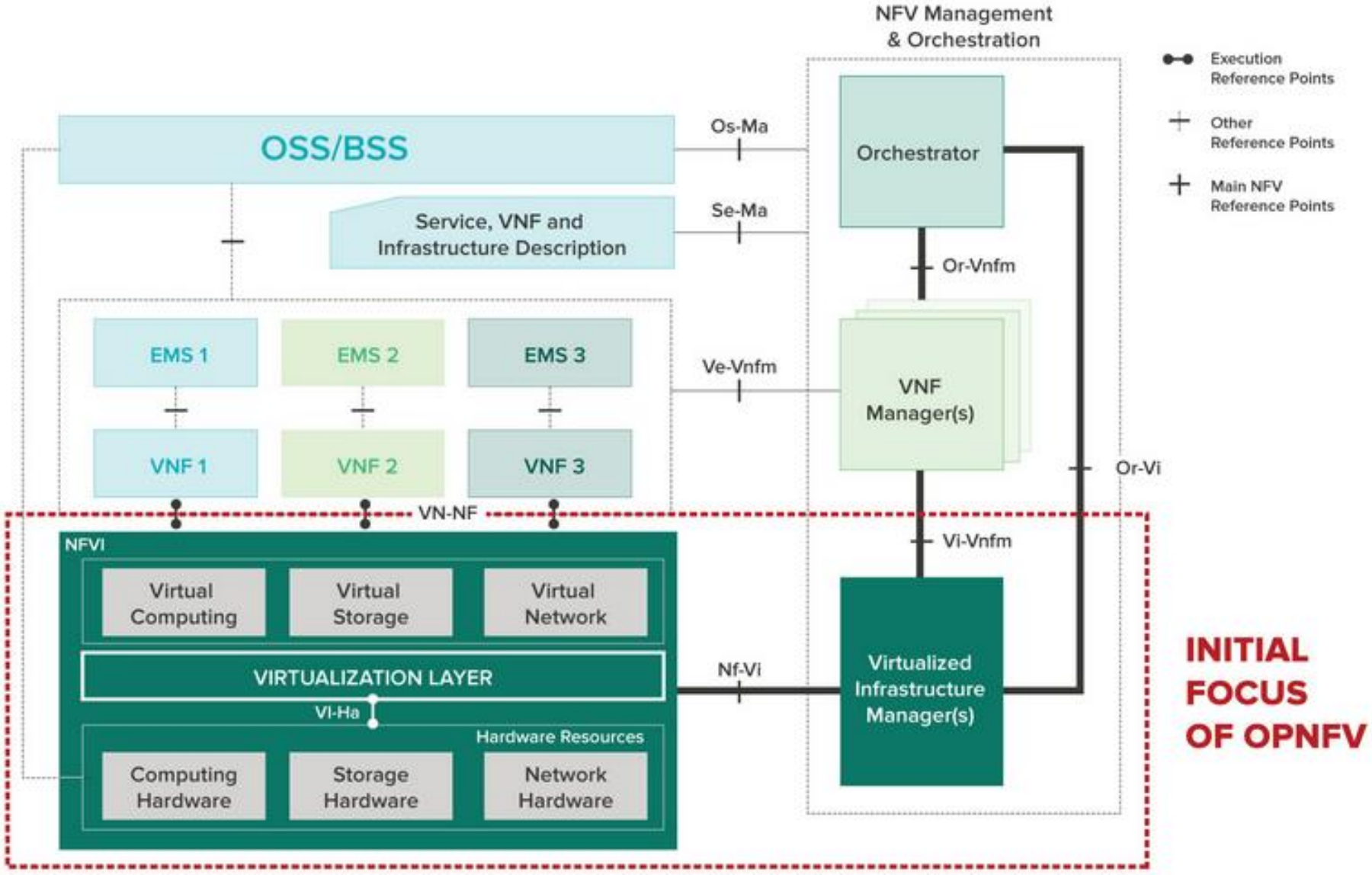


SDN MPLS

OPNFV

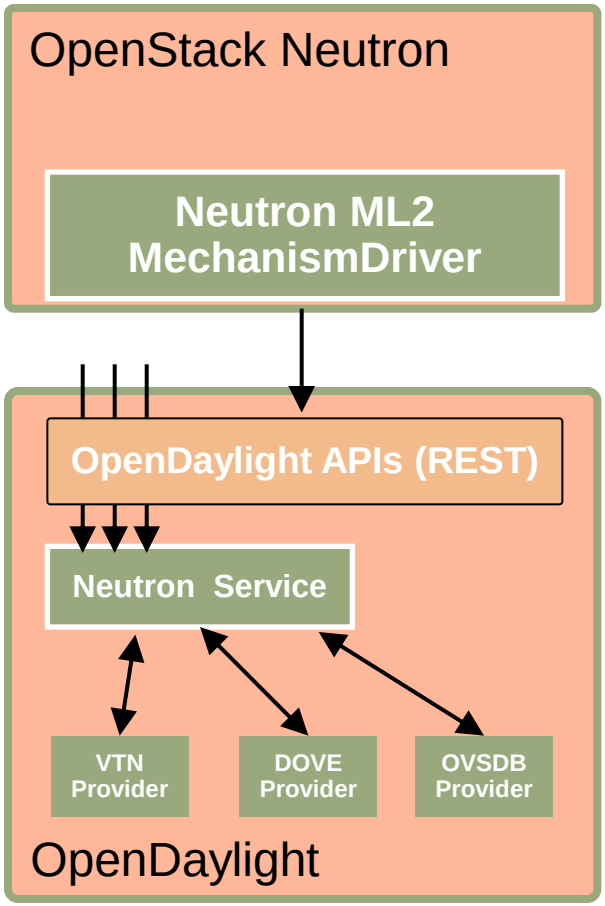
- OPNFV Mission
  - Open Platform for NFV (OPNFV) is relatively new
  - Focused on accelerating the evolution of Network Functions Virtualization (NFV)
  - Combines IT orchestration (OpenStack) with the SDN Controller (ONOS or ODL)
- OPNFV Initial Scope
  - The initial scope of OPNFV will be on building NFV Infrastructure (NFVI), Virtualized Infrastructure Management (VIM), and including application programmable interfaces (APIs) to other NFV elements
  - Blend candidate technologies for for Virtualized Network Functions (VNF) and Management and Network Orchestration (MANO) components
  - OPNFV will work closely with ETSI's NFV ISG

OPNFV Focus Area



ODL & OpenStack Integration

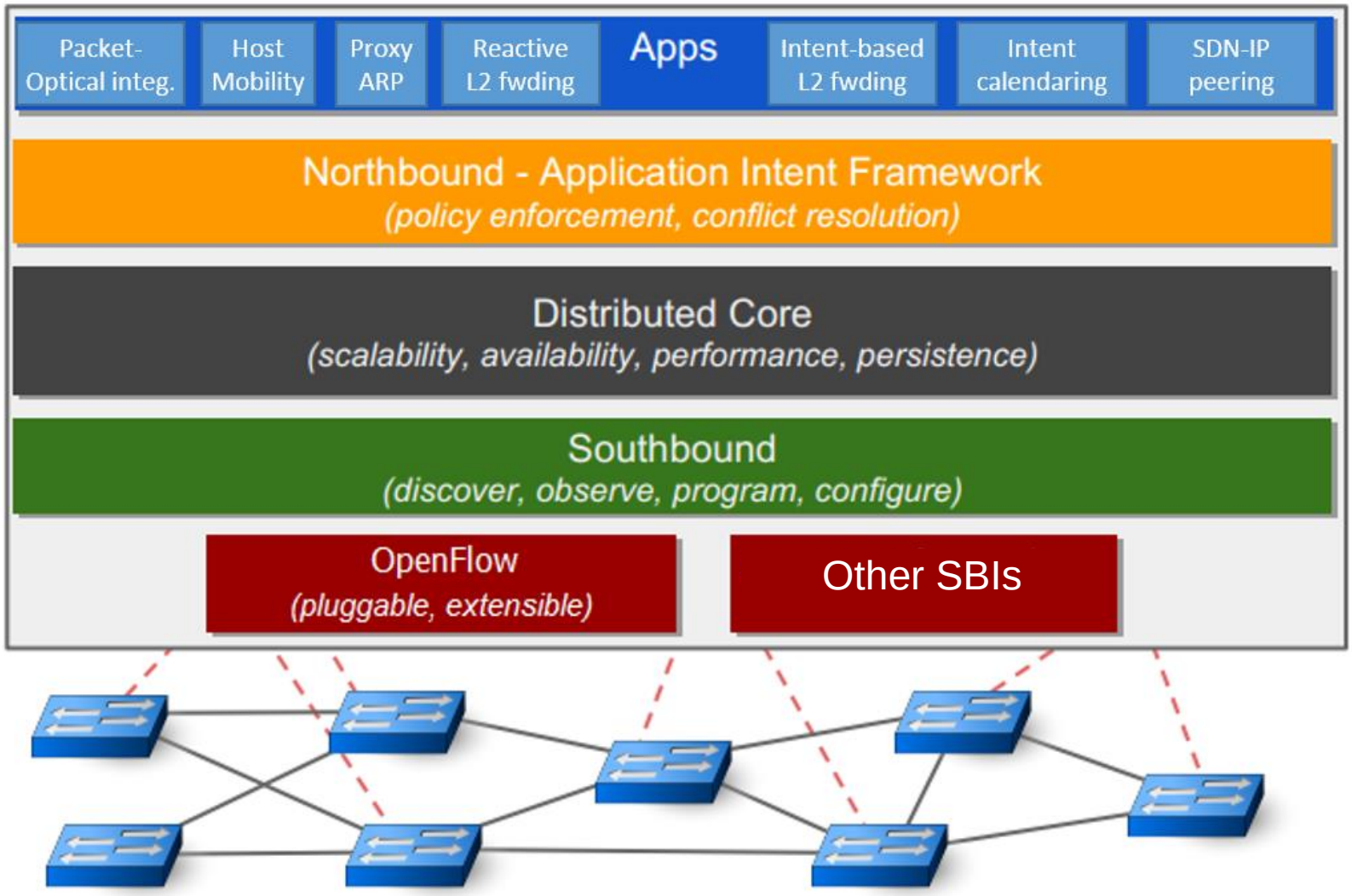
- OpenDaylight exposes a single common OpenStack Service Northbound
  - API exposed matches Neutron API precisely
  - Multiple implementations of Neutron networks in OpenDaylight
- OpenDaylight OpenStack Neutron Plugin simply passes through
  - Simplifies OpenStack plugin
  - pushes complexity to OpenDaylight



Open Networking Operating System (ONOS)

- Defining features of ONOS
  - Distributed Core
    - Provides scalability, high availability, and performance which brings carrier grade features to the SDN control plane
    - Operator can add servers incrementally, without disruption, as needed for additional control plane capacity
  - Northbound abstraction/APIs
    - Includes network graph and application intents to ease development of control, management, and configuration services
    - There are two Northbound abstractions: the Intent Framework and the Global Network View
  - Southbound abstraction/APIs
    - Enables pluggable southbound protocols for controlling OpenFlow, NETCONF and Legacy devices
  - Software Modularity
    - Open Source with key vendor support (especially Huawei!)

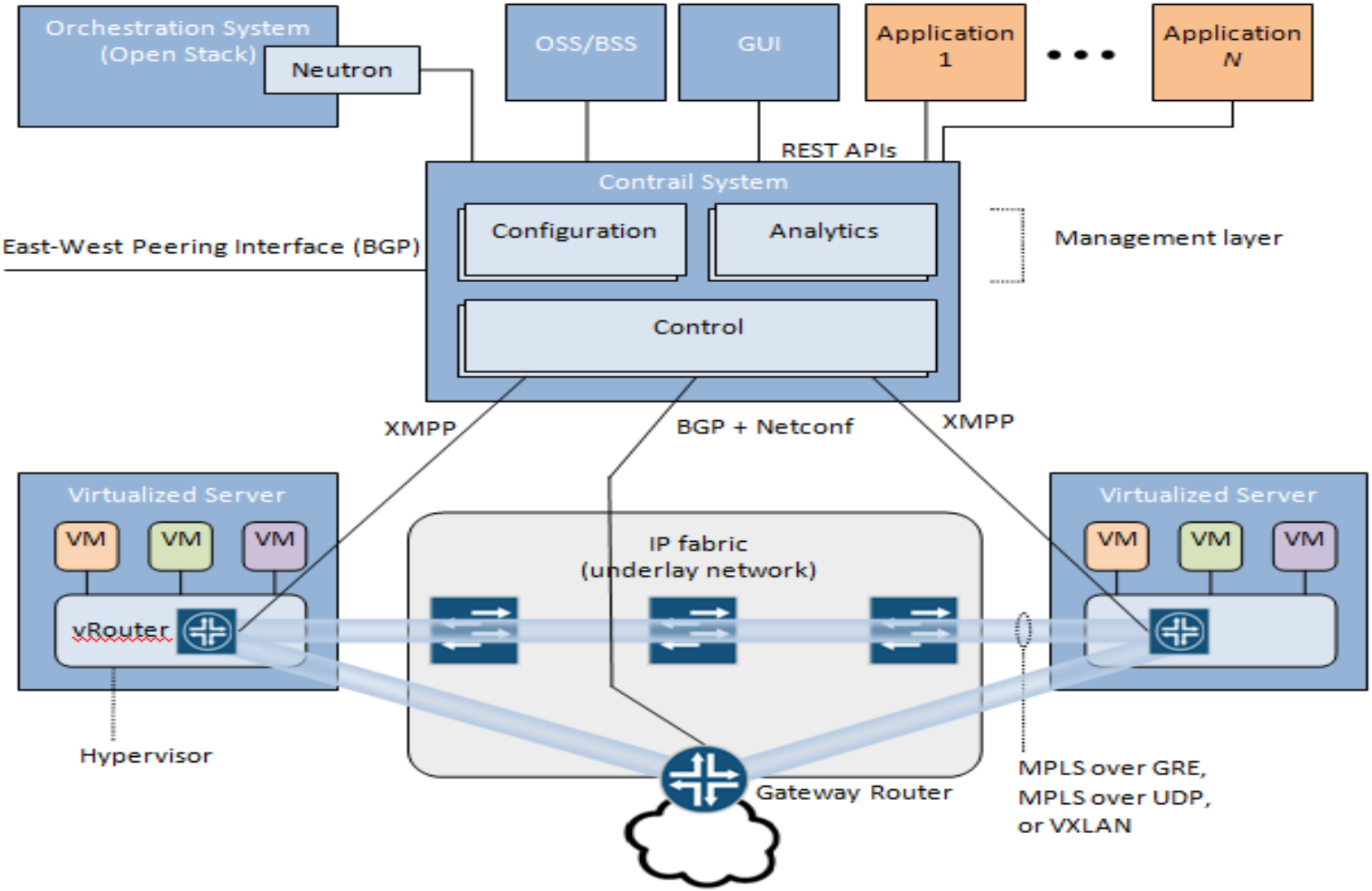
ONOS – High-level Architecture



OpenContrail

- The Contrail System consists of two main components:
  - The Contrail Controller:
    - The Contrail Controller is a logically centralized but physically distributed Software Defined Networking (SDN) controller that is responsible for providing the management, control, and analytics functions of the virtualized network
  - The Contrail vRouter
    - Contrail vRouter is a forwarding plane (of a distributed router) that runs in the hypervisor of a virtualized server
      - It extends the network from the physical routers and switches in a data center into a virtual overlay network hosted in the virtualized servers
      - The Contrail vRouter is conceptually similar to existing commercial and open source vSwitches such as for example the Open vSwitch (OVS) but it also provides routing and higher layer services (hence vRouter instead of vSwitch).
  - The Contrail Controller provides the logically centralized control plane and management plane of the system and orchestrates the vRouters

OpenContrail Architecture



SDN Controllers : Summary

- There are a number of proposed architectures
- There are also several Open Source implementation
- The architectures are relatively similar
  - Modular
  - APIs between components
    - Allows pluggable components
  - Northbound and southbound interfaces
    - In some way “standardised”

South Bound Interfaces

- A plethora of protocols may be used in the SDN environment
  - The non-exhaustive list below shows standards-based protocols being used in the telecom market.
    - OpenFlow
    - Forwarding and Control Element Separation (ForCES)
    - Path Computation Element (PCEP)
    - Extensible Messaging and Presence Protocol (XMPP)
    - Border Gateway Protocol Linkstate Distribution (BGP-LS)
    - NETCONF/RESTCONF with YANG/JSON
    - Interface to the Routing System (I2RS)
    - Open vSwitch Database Management Protocol (OVSDB)
  - Some of these protocols were not originally developed for SDN
    - You can make any protocol do anything!

OpenFlow

- Grew out of Ethane and now developed by the ONF
  - Currently at release 1.5
- Runs over TCP
- Protocol is data-structure-based
  - Makes new compatibility between releases hard
- Enables controllers to determine the path of network packets through the network of switches/router
  - Program the forwarding plane
    - Ethernet – DA, SA, EtherType, VLAN
    - MPLS – top label and BoS bit
    - IP (v4 or v6) – DA, SA, protocol, DSCP, ECN
    - TCP/UDP ports
- Various concerns about bootstrapping and security

ForCES

- FORwarding & Control Element Separation
  - IETF working group 2001-2015
- Arose from need for an open and standardized programmable interface for off-the-shelf network processor
- Protocol and data model
  - RFC 5812 Forwarding Element information/data model
  - RFC 5810 ForCES protocol
    - Master-slave protocol
    - Binary encoding of information model
- Can do everything OpenFlow can do

PCEP

- Developed as a request/response protocol
  - Communications between a client and a Path Computation Element (PCE)
    - Path computation requests and responses
  - Runs over TCP
- Assumes the presence of an active control plane
  - Therefore not a pure SDN system
- Extended to allow PCE to control the network
  - Active Stateful PCE
    - Delegation of control of established LSPs
    - PCInitiate

Possible Next Steps for PCEP

- Several new proposals for extending PCEP
  - Migration to become an SDN protocol
- Allow a controller to tell the network what an LSP is for
- Program individual forwarding steps
  - No need for an active control plane
  - Turns PCEP into a full SBI
- Retrieve topology information from network
  - No need for PCE to participate in IGP
  - Compare with BGP-LS (later)
- Retrieve data from individual network nodes
  - Inventory and network state
  - No need to run an IGP in the network
  - Compare with NETCONF/YANG (later)

XMPP

- Extensible Messaging and Presence Protocol
- Simple protocol designed to carry XML
  - As used by jabber for instant messaging
  - Use by Contrail as an SBI
- Benefits of great simplicity
- Misses features of NETCONF (qv.)
- Not RESTful like RESTCONF (qv.)

BGP Linkstate (BGP-LS)

- BGP is the glue of the Internet
- It is a session-based protocol that runs over TCP
  - Designed to move routing data about
- BGP speakers typically have access to all IGP data
- BGP-LS are extensions to allow a BGP speaker to export IGP information
  - Specifically TE data that comprises the TE database
- Use with a controller/orchestrator/PCE
  - No need to listen to the IGP

Possible Next Steps for BGP-LS

- A number of proposals move BGP-LS closer to SDN
- BGP Flowspec
  - Use it to tell the network what an LSP is for
- Use BGP-LS to report network state and inventory
  - Every network node is a BGP speaker
- BGP Path Programming
  - Use BGP to instruct network how to set up LSPs
- BGP path Programming in detail
  - Programming forwarding state in individual nodes
    - Every network node is a BGP speaker
- Oh look! Every protocol can do everything!

NETCONF Becoming RESTCONF

- NETCONF is an IETF network management tool designed to support configuration of network elements
  - Distinction between configuration and state data
  - Multiple configuration data stores:
  - Candidate, running, startup
  - Configuration change validations
  - Configuration change transactions
  - Selective data retrieval with filtering
  - Extensible Remote Procedure Call (RPC) mechanism
  - Runs over TCP with (or without) security
- RESTCONF is IETF work in progress
  - Build on everything in NETCONF
    - A simplified subset of functions
    - Still supports CRUD (create, retrieve, update, delete)
  - Add RESTful function
  - Runs over HTTP

YANG Language

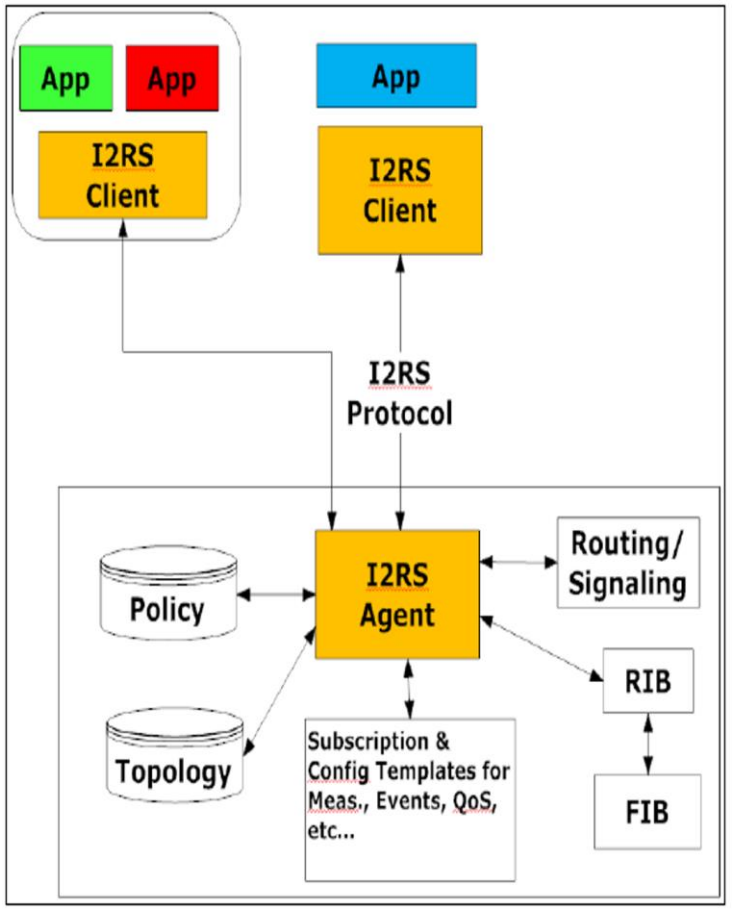
- Used to model data for standard representation
  - Used to define XML carried by NETCONF/RESTCONF
- A YANG module defines a hierarchy of data
  - Configuration, state data, RPCs, and notifications
  - Complete description of data between client and server
- Data is modelled as a hierarchical tree
  - Each node has a name
  - Each node has either a value or a set of child nodes
- Data models arranged into modules and sub-modules
  - Modules can import from other modules, and include data from sub-modules
- Augmentation allows one module to add data nodes to the hierarchy defined in another module

JSON Encoding

- JavaScript Object Notation Data Interchange Format
  - RFC 7158
- Can map from YANG automatically
  - Thus XML and JSON are interchangeable
- Design goals were for it to be minimal, portable, textual, and a subset of JavaScript
- Less verbose than XML on the wire
- Machine readable
- Semi-readable by humans
  - Not as accessible as XML

Interface to the Routing System (I2RS)

- SDN SBI has focused on programming the data plane
  - Switch programming (cross-connects)
  - Forwarding (FIB)
- There are many functions and features not covered by
  - Management of the “routing system”
  - Programming the Routing Information Base (RIB)
  - Control of routing protocols
- Existing techniques are non-standard
  - Using CLI to achieve these functions is very frustrating
  - Expensive, time-consuming, error-prone, risky
- Uses RESTCONF/YANG
- Use Cases include
  - Programming and managing the RIB
  - BGP use cases
  - Traffic steering and classification
  - DDoS mitigation
  - Topology reading, monitoring, and control

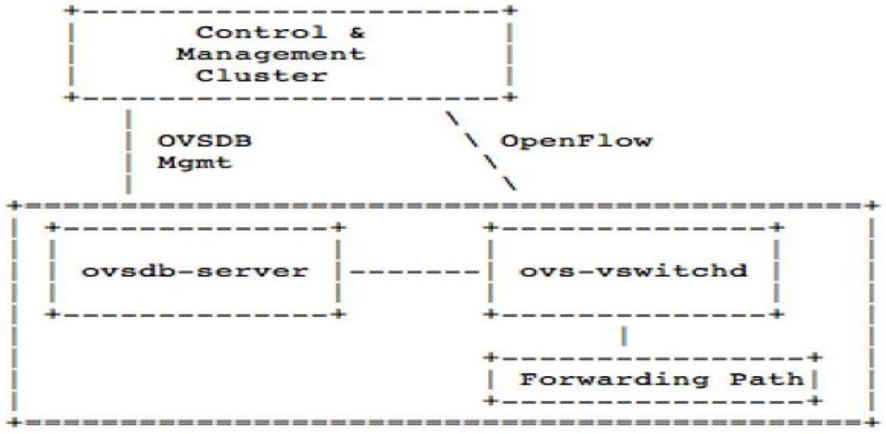


OVS Database Protocol (OVSDB)

- OVSDB is targeting a virtual switch
- Using the OVSDB Protocol, managers can specify the number of individual virtual bridges within an Open vSwitch implementation and create, configure and delete ports and tunnels from a bridge
- A manager can also create, configure and delete queues
- The OVSDB Protocol uses
  - The JavaScript Object Notation (JSON) specified in RFC 4627 for its schema and Wire protocol format
  - JSON-RPC 1.0 for its wire protocol
  - Each RPC may contain a list of configuration operations but it operates as an individual transaction
- Open vSwitch already implements the OVSDB Protocol, but other switch vendors would need to modify their switch's configuration database and implement JSON and JSON-RPC in order to adopt OVSDB

Open vSwitch (OVS)

- Open vSwitch
  - Open vSwitch supports distribution across multiple physical servers
  - Open vSwitch is a virtual switch that enables network automation, while supporting standard management interfaces and protocols, like NetFlow
  - The control and management cluster contains managers and controllers that use the OVSDB Protocol to supply configuration information to the switch database server



OVS Capabilities

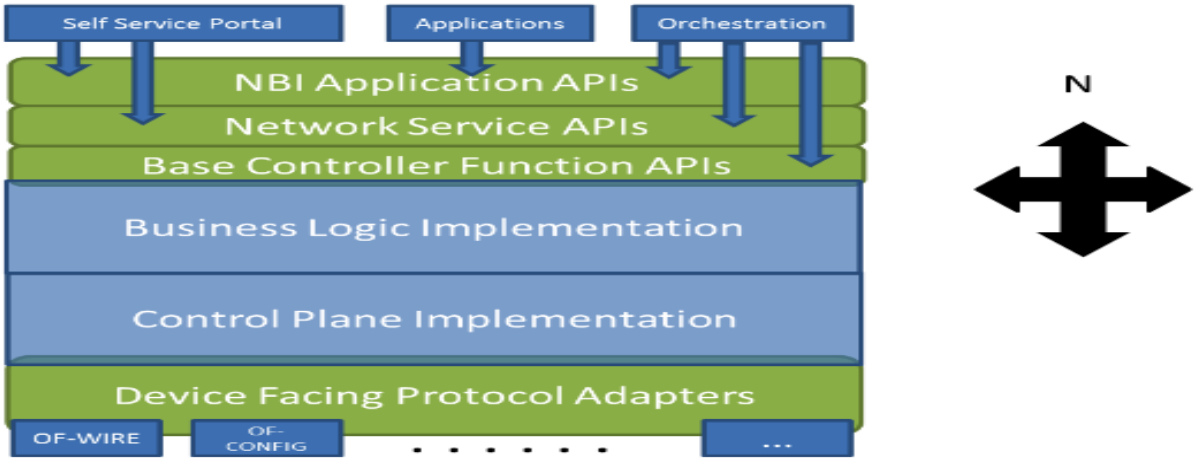
- OVS configuration is contained within several database tables, all written persistently to disk
  - These tables contain varying amounts and types of configuration data, and all refer to each other in various ways, much like a relational database would
  - ovsdb-server maintains the switch table database and external clients can talk to ovsdb-server using json rpc and json being the data format
  - ovsdb database currently contains around 13 tables and these can be extended further:
    - Bridge – Bridge configuration.
    - Port – Port configuration.
    - Interface – One physical network device in a Port.
    - Flow_Table – OpenFlow table configuration
    - QoS – Quality of Service configuration
    - Queue – QoS output queue
    - Mirror – Port mirroring
    - Controller – OpenFlow controller configuration.
    - Manager – OVSDB management connection.
    - NetFlow – NetFlow configuration.
    - SSL – SSL configuration.
    - sFlow – sFlow configuration.
    - IPFIX – IPFIX configuration

Northbound Interfaces

- North of what?
  - North of the Controller (to the Orchestrator)?
  - North of the Orchestrator (to the Application)?
- The NBI clearly exists in all of the architectures
- Largely assumed to be RESTful
- Needs standard data models
- May often appear as an internal API in implementations
  - Pluggable components in the OpenSource world
- Solutions are configuration models and service models

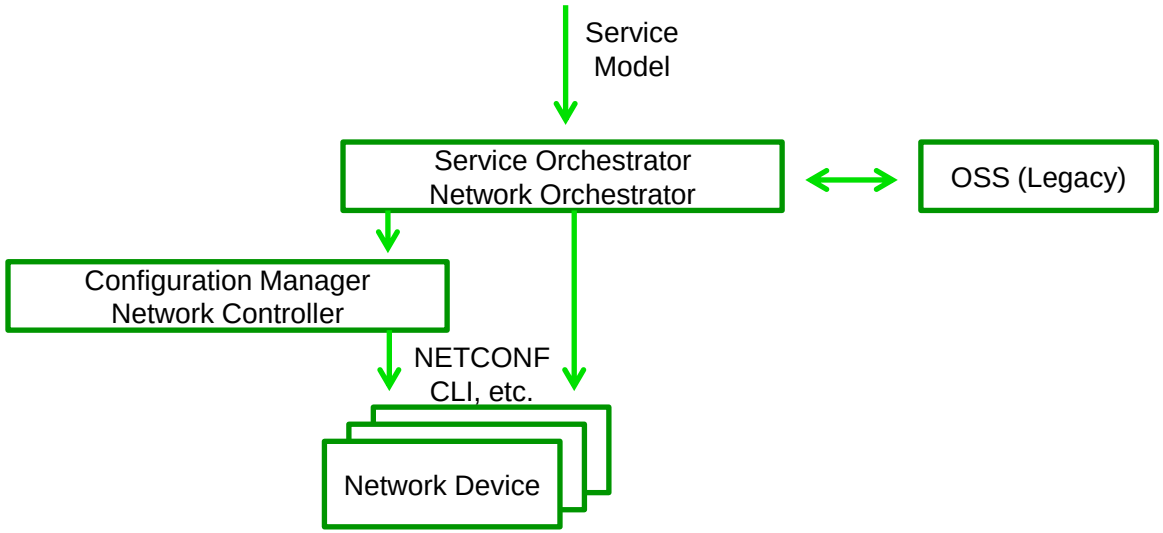
ONF: North Bound Interface

- Market Status
  - Currently there's no NBI standard in the marketplace
- ONF Approach
  - The Northbound Interfaces (NBI) working group was created to help develop concrete requirements, architecture, and working code for northbound interfaces
  - Scope of Working Group
    - The scope includes any API that adds abstraction and encapsulates implementation complexity, northbound from the SDN controller
      - Note: ONF is not going to define a standardized NBI API
  - NBI Model



Service Modelling

- Service models are not configuration models
- On the interface between customer and operator
  - Represents the operator offering in a standard way
- Allows automation of the service via a service orchestrator
- Compare with ABNO etc.



East West Interfaces

- End-to-end connectivity often spans multiple domains
- Each domain is under the command of a controller
- Each network is under the command of an orchestrator
- Building an end-to-end path needs coordination
- Two solutions
  - Hierarchical arrangement of “super-controllers” or “sub-orchestrators”
  - East-west communications between peer controllers
    - May start to look like an active control plane

ONF: East West Interface (1/3)

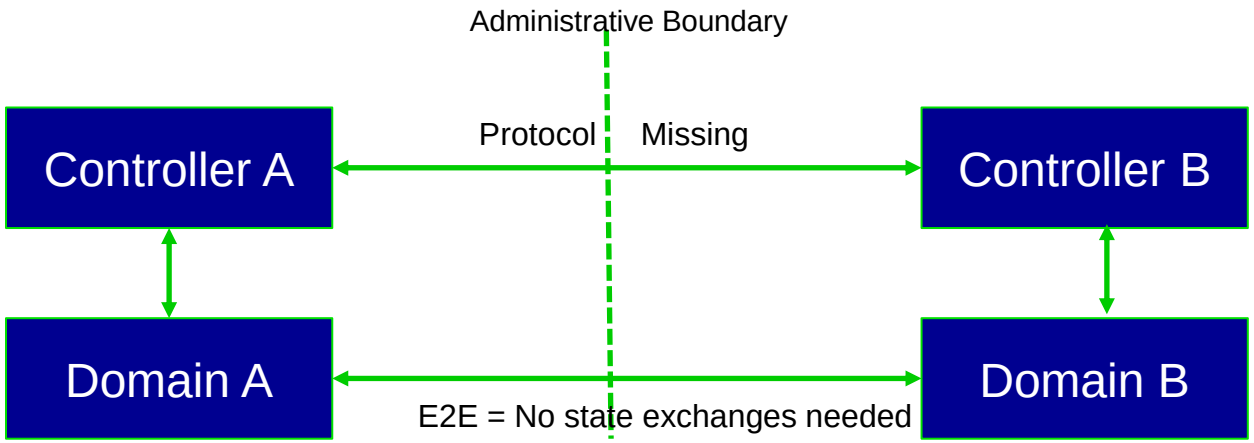
- What are the SDN multi-domain requirements?
- Do we need a protocol (beyond PCEP) to communicate between controllers for end-to-end service creation?
- Other multi-domain scenarios can be considered similar to the East West problem
  - They also require AAA, Policy, Information/Abstraction for candidate paths/topologies
  - Resource requests are passed from the application to the controller
  - Controller need not know the specific application
  - Resources are allocated according to the controller's administrative policies
  - Information abstracted and shared according to controllers policies

ONF: East West Interface (2/3)

- Information to be exchanged between controllers may include the following:
  - SDN controller adjacency and capability discovery
  - Data plane neighbour and topology discovery, to the extent agreed by policy
  - State and attribute information, including the ability to subscribe to state and attribute change notifications, as agreed by policy
  - Forwarding-relevant information, such as reachability at one or more layers
  - Path computation information such as route cost, protection or restoration policies
  - Other information such as OAM configuration, QoS assessment and reporting, usage information for billing

ONF: East West Interface (3/3)

- This interface requires two main elements
  - Trust
  - Information
- Trust
  - Is the application inside or outside my trusted boundaries?
- Information
  - What information needs to be exchanged?
  - Abstraction or scoping of information is based on trust or separation of concerns (multiple controllers, same domain)



And are we any the wiser?

What, exactly is SDN? And why bother standardising it?

"I don't know what you mean by ‘SDN'," Alice said.

Humpty Dumpty smiled contemptuously. "Of course you don't - till I tell you. I meant 'there's a nice knock-down argument for you!'"

"But ‘SDN' doesn't mean 'a nice knock-down argument'," Alice objected.

"When I use a word," Humpty Dumpty said, in rather a scornful tone, "it means just what I choose it to mean - neither more nor less."

"The question is," said Alice, "whether you *can* make words mean so many different things."

"The question is," said Humpty Dumpty, "which is to be master - that's all."

Somewhat adapted from *Through the Looking Glass* by Lewis Carr

Questions

adrian@olddog.co.uk
d.king@lancaster.ac.uk